

## СИСТЕМА ВИРТУАЛЬНОГО ОКРУЖЕНИЯ VIRSIM ДЛЯ ИМИТАЦИОННО-ТРЕНАЖЕРНЫХ КОМПЛЕКСОВ ПОДГОТОВКИ КОСМОНАВТОВ

М.В. Михайлюк, А.В. Мальцев, П.Ю. Тимохин, Е.В. Страшнов,  
Б.И. Крючков, В.М. Усов

Докт. физ.-мат. наук, профессор М.В. Михайлюк; А.В. Мальцев;  
П.Ю. Тимохин; Е.В. Страшнов (ФГУ ФНЦ НИИСИ РАН)  
Докт. техн. наук Б.И. Крючков; докт. мед. наук, профессор В.М. Усов  
(ФГБУ «НИИ ЦПК имени Ю.А. Гагарина»)

Современные технологии виртуальной реальности (VR) могут быть использованы при разработке космических имитационно-моделирующих и тренажерных комплексов нового типа, которые позволят повысить качество технических средств подготовки космонавтов (ТСПК), расширить область моделируемых на имитационных стендах задач. В статье рассматриваются основные возможности системы виртуального окружения (СВО) VirSim и ее использования на базе космических тренажеров и имитационно-моделирующих стендов.

**Ключевые слова:** система виртуального окружения, космические тренажеры, имитационные стенды, подготовка космонавтов.

### **The VirSim Virtual Environment System for the Simulation Complexes of Cosmonaut Training. M.V. Mikhailyuk, A.V. Maltsev, P.Yu. Timokhin, E.V. Strashnov, B.I. Kryuchkov, V.M. Usov**

Existing virtual reality (VR) technologies can be used in designing the new type simulation complexes that will allow improving quality of the technical facilities for cosmonaut training and expanding the range of modeling tasks. The paper considers the main potentials of the VirSim system and its application for space purposed simulators and simulation-modeling stands.

**Keywords:** virtual environment system, space purposed simulators, modeling stands, cosmonaut training.

Имитационно-тренажерные комплексы занимают важное место в системе подготовки космонавтов к выполнению запланированных миссий, выполнении конкретных задач и при проектировании деятельности в новых миссиях. Опасная среда, непривычная обстановка, невесомость, наличие скафандра и т.д. приводят к необходимости использования имитационного моделирования на Земле. Создание необходимых для тренировок натуральных экспонатов большой величины может представлять реальные трудности. С помощью реальных объектов трудно смоделировать многие условия и процессы, происходящие в космосе. Использование видеотренажеров, на которых окружающая обстановка моделируется и визуализируется с по-

мощью компьютеров, представляет существенный шаг вперед в обучении и подготовке операторов.

На сегодняшний день существует ряд программных комплексов, созданных для трехмерного моделирования в области исследования космоса и тренировок космонавтов. Одной из таких систем является NASA's Eyes Visualization [1], позволяющая, например, имитировать полет на космическом корабле Voyager, моделировать движение по орбите Марса, посадку на планеты и перемещение по их поверхностям с помощью специальных технических средств, таких, как марсоход. Данная система основана на использовании коммерческого графического движка Unity [2] для визуализации трехмерной виртуальной среды.

Еще одним примером может служить программный комплекс EDGE (Engineering DOUG Graphics for Exploration [3]), созданный на базе графического движка DOUG (Dynamic Onboard Ubiquitous Graphics) – собственной разработки NASA. Комплекс используется для подготовки астронавтов к внекорабельной деятельности. Он обеспечивает визуализацию трехмерной виртуальной модели Международной космической станции и космических кораблей в масштабе реального времени, отработку задач перемещения по внешней поверхности станции, обслуживания оборудования, спасения астронавта с помощью устройства SAFER и т.д. Преимуществом EDGE является широкий функционал решаемых задач и высокая детализация геометрии используемых виртуальных моделей. Но взаимодействие, например, модели астронавта с моделью МКС реализовано не в полном объеме: выполняется только контроль столкновений рук с поручнями станции и некоторыми приборами, участвующими в тренировке. Существуют также профессиональные системы визуализации более широкого профиля, например, UNIGINE 2 [4], одним из направлений использования которых могут являться космические тренажеры. Однако такие продукты, как правило, имеют весьма высокую стоимость лицензии для использования в тренажерных комплексах.

Отдельное направление в области систем виртуального окружения (СВО) космического назначения занимают системы визуализации трехмерных моделей Земли и других планет. Широко известным является проект Google Планета Земля [5], отличительной чертой которого является беспрецедентно высокое разрешение моделируемой земной поверхности. Изображения Земли синтезируются на основе различных по дате и времени данных спутников и аэрофотосъемки, которые могут существенно отличаться от одной области к другой, как, например, в областях стыковки суши и водных поверхностей. Серьезными ограничениями проекта являются жесткая лицензионная политика и отсутствие возможности расширения и модификации ядра визуализатора сторонними разработчиками. Противоположный подход продемонстрирован в программных комплексах NASA World Wind [6] и Celestia [7]. Комплексы являются открытыми и спроектированы так, чтобы их можно было эффективно дополнять и модифицировать. World Wind

обладает широкими возможностями по моделированию поверхности Земли и используется во многих видеотренажерах NASA. Система работает через виртуальную машину Java (для поддержки веб-приложений и мобильных устройств), что ограничивает ее производительность по сравнению с аналогичными решениями, создаваемыми под видеотренажерные платформы. Проект Celestia ориентирован на 3D-симуляцию объектов в масштабе Вселенной – от искусственных спутников до полных галактик, и используется NASA и ESA в образовательных программах. Реалистичная визуализация планет в масштабе реального времени является важной, но не приоритетной задачей проекта, что критично для космических видеотренажеров. Кроме программных пакетов, решающих комплексно задачу моделирования визуальных и динамических свойств Земли, существуют проекты, в которых хорошо проработаны только отдельные направления. Например, система Orbiter [8] ориентирована на качественное моделирование межпланетных полетов и орбитальной динамики, а в графическом движке Outerra [9] большое внимание уделяется детализации и реалистичности моделирования поверхности планеты. Такие проекты являются ориентирами для дальнейшего развития тренажерных систем визуализации.

В области моделирования динамики виртуальных объектов существуют готовые программные библиотеки (так называемые физические движки) Nvidia PhysX [10], Bullet [11], ODE [12], Havok [13] и другие. Эти программные продукты ориентированы на игровые и развлекательные приложения, где не требуется физическая точность вычислений, а определяющим фактором является игровая реалистичность поведения объектов. Существуют также профессиональные решения для промышленности, в которых осуществляется виртуальное моделирование сложных машин, механизмов и изделий в сборе. Такие решения представляют собой CAE-системы (Computer-Aided Engineering) и предлагаются следующими компаниями: Adams [14], Simpack [15], проект «Универсальный механизм» [16], ЗАО «Автомеханика» (проект «Эйлер» [17]) и другие. Их характеризует высокая стоимость и чаще всего они ориентированы на офлайн-моделирование.

В области систем управления виртуальными объектами известны такие разработки, как Simulink [18], Scicos [19], Dymola [20], LabView [21] и другие. Несмотря на обилие существующих программных продуктов, остается актуальной задача создания более удобной и доступной системы моделирования на основе функциональных схем для СВО имитационно-тренажерных комплексов. Также существует несколько программных комплексов (симуляторов), которые используют физические движки и предназначены для моделирования роботов, такие, как USARSim [22], Gazebo [23], V-REP [24], Webots [25], ARGoS [26] и др.

Использование сторонних бесплатных движков с закрытым исходным кодом в этих симуляторах приводит к тому, что они обладают ограниченным функционалом и плохой расширяемостью. Также недостатком этих систем

является то, что они больше ориентированы на моделирование динамики роботов, в то время как для управления роботами требуется дополнительное написание программ или сценариев.

В настоящей работе представляется СВО VirSim, разработанная в ФГУ ФНЦ НИИСИ РАН в части ее применимости в имитационно-моделирующих и тренажерных комплексах для подготовки космонавтов.

## Общая структура СВО

Компоненты разработанной СВО взаимодействуют следующим образом. Оператор видит виртуальную сцену на экране монитора или на носимых устройствах виртуальной реальности (VR) и осуществляет управление выбранным виртуальным объектом с помощью одного из средств управления (виртуального или реального пульта, жестов, голосом, экзоскелета и т.д.).

Действия оператора поступают в систему управления, в которой с помощью специальной функциональной схемы вычисляются управляющие сигналы (например, величины напряжений, подаваемых на электрические двигатели, флаги включения или выключения фар и т.д.). Подсистема динамики, получая эти сигналы, осуществляет моделирование движения или других изменений динамических объектов и вычисляет их новые положения и ориентации. Эти данные передаются в подсистему визуализации, которая синтезирует изображения виртуальной сцены из текущего средства наблюдения сцен VR, которые использует оператор.

Общая структура СВО показана на рисунке 1а. На рисунке 1б представлены этапы проектирования интерактивных 3D-сцен с помощью СВО.

Весь цикл должен занимать не более 40 мсек, чтобы обеспечить режим реального времени, при котором в секунду выводится не менее 25 кадров. Это позволяет получить плавное движение объектов в процессе визуализации. С другой стороны, это накладывает серьезные требования к скорости работы всех подсистем с учетом необходимости обеспечения высоко реалистичной визуализации и адекватного поведения виртуальных объектов.

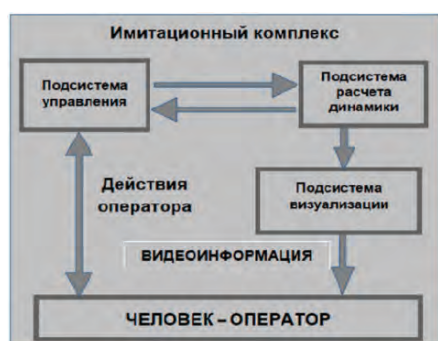


Рис. 1а. Структура имитационно-тренажерного комплекса

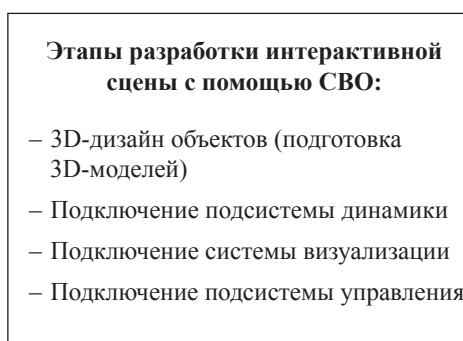


Рис. 1б. Этапы разработки интерактивной сцены с помощью СВО

## Подготовка 3D-моделей

Первым этапом при разработке СВО является создание виртуальной сцены. Для этого можно использовать системы трехмерного (3D) моделирования, например, Autodesk 3ds Max, Maya и т.д. В 3ds Max виртуальные сцены создаются из набора стандартных графических примитивов (параллелепипедов, сфер, конусов, цилиндров и т.д.), для каждого из которых задаются размеры и материал, включающий набор текстур и параметров. Зная формы и размеры частей моделируемого объекта, из этих примитивов можно сконструировать его виртуальную модель. Текстуры для этой модели создаются с помощью графических редакторов (например, Adobe Photoshop) и накладываются на соответствующие ее части. Примитивы можно связать друг с другом таким образом, чтобы одному объекту (родителю) были подчинены один или несколько других объектов (детей). Тем самым получается дерево всей сцены.

Если родитель перемещается или поворачивается, то за ним также перемещаются или поворачиваются все его дети и далее все объекты вниз по иерархии. Однако в стандартном наборе примитивов системы моделирования 3ds Max нет многих необходимых для виртуального окружения элементов (динамических объектов, датчиков и т.д.). Для расширения стандартного набора примитивов был разработан специальный конструктор в виде плагина, подключаемого к 3ds Max и позволяющего добавлять на панель инструментов новые элементы с требуемым списком параметров, а также создавать их в виртуальной сцене. Так, например, данный конструктор поддерживает динамические объекты типа электрический и ракетный двигатели, шарниры различного вида, колеса и гусеницы, подвески колес, винты, специальные системы частиц и т.д. Также реализованы датчики различных типов: положения, ориентации, сило-моментные, расстояния, линейных и угловых скоростей, радиоактивного излучения и др.

С помощью 3ds Max создаются также виртуальные камеры (с ортографической или перспективной проекцией) и источники освещения различных типов. Опыт использования виртуальных сцен в космических тренажерах рассмотрен в [27].

Файл с виртуальной сценой загружается с помощью специально разработанного конвертора во внутренние структуры подсистем динамики и визуализации.

## Подсистема динамики

Подсистема динамики осуществляет моделирование поведения динамических объектов в масштабе реального времени. Это значит, что для вычисления новых положений и ориентаций всех объектов отводится не более 10 миллисекунд. Основными составляющими подсистемы динамики являются:

- моделирование динамики абсолютно твердого тела и систем шарнирно связанных тел (открытых и замкнутых кинематических цепей);
- моделирование двигателей различного типа (электрических, реактивных и т.д.);
- моделирование колесных и гусеничных роботов;
- определение и разрешение коллизий (столкновений) объектов;
- реализация моделей различных видов трения (скольжения, качения и верчения);
- моделирование динамики специальных типов объектов (системы частиц, тросы, пружины, демпферы и др.);
- моделирование датчиков различных типов.

Динамика абсолютно твердого тела описывается дифференциальными уравнениями Ньютона-Эйлера [28], позволяющими вычислить линейную и угловую скорости тела через действующие на него силы и моменты. Интегрирование этих уравнений осуществляется с помощью полуневявной схемы Эйлера [29], в которой сначала вычисляются новые скорости тела, а затем его положение и ориентация.

Сложные объекты могут состоять из простых, соединенных шарнирами различного типа (осевой, сферический, Гука, призматический, цилиндрический), образуя тем самым систему шарнирно связанных тел. Такая система может иметь структуру в виде открытой или замкнутой кинематической цепи. С помощью подобных цепей описываются манипуляторы, механизмы захватного устройства, подвески на двойных поперечных и продольных рычагах, опорные устройства и др.

Моделирование динамики систем шарнирно связанных тел осуществляется путем их разбиения на множество твердых тел и задания голономных ограничений [30] относительно координат тел, описывающих действие шарниров. Эти ограничения формулируются таким образом, чтобы обеспечить неразрывность шарниров. Реализация ограничений осуществляется на основе импульсного подхода, в рамках которого последовательно вычисляются импульсы, изменяющие скорости тел так, чтобы ограничения были выполнены. Этот метод позволяет производить расчет как открытых, так и замкнутых кинематических цепей, и моделировать динамику как колесных, так и гусеничных роботов.

Математическая модель вращательного (или линейного) электрического двигателя позволяет вычислить крутящий момент (или силу) двигателя в зависимости от его паспортных параметров и напряжения, подаваемого от системы управления. Этот момент (соответственно, сила) передается на объекты, присоединенные к двигателю (колеса, шарниры и др.). Динамическая модель колеса включает как контактное взаимодействие колеса с рельефом, так и его движение с учетом типа подвески, соединяющей колеса с шасси.

В подсистеме динамики смоделированы следующие типы подвесок: с качающимися полуосями (осевая поперечная), на продольных рычагах

(осевая продольная), на двойных поперечных и продольных рычагах (трапециевидные), свечная (амортизатор) и подвеска Макферсона (на направляющих стойках – осевая с амортизатором).

На рисунке 2а показано моделирование динамики робота-кентавра на поверхности Луны на примере решения задачи захвата камня и его перемещения в тележку робота.

На рисунке 2б показано выполнение операции взятия грунта на Марсе с помощью модели марсохода, в которой колеса крепятся к корпусу с помощью подвески с качающимися полуосями.



Рис. 2а. Моделирование робота-кентавра

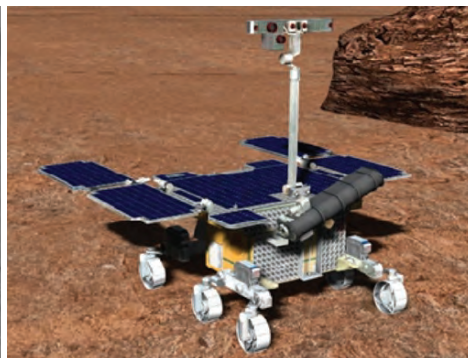


Рис. 2б. Моделирование динамики марсохода

Помимо моделирования перемещения робота по поверхности система позволяет моделировать перемещение в открытом космосе с помощью реактивных двигателей, для которых предусмотрено задание соответствующих параметров.

Математические модели реактивных (ракетных) двигателей на основе заводских параметров (максимальная сила тяги, расход топлива, скорость истечения газов из сопла и т.д.) и параметров, поступающих из системы управления, позволяют вычислить развиваемую двигателем силу тяги в непрерывном и импульсном режимах работы. С помощью таких двигателей осуществляется управление движением космических летательных аппаратов в рамках решения задач их ориентирования, стабилизации, коррекции орбиты, стыковки и посадки.

В процессе движения объекты могут сталкиваться друг с другом. Для виртуальных объектов первым этапом имитации столкновения является определение факта пересечения этих объектов [31] (определение коллизий). Затем вычисляются возникающие силы и моменты, моделируется расталкивание объектов [32] (разрешение коллизии). Так как определить пересечение объектов сложной формы в масштабе реального времени достаточно сложно, то оно проверяется для набора аппроксимирующих контейнеров (геометрических примитивов), окружающих объекты. В качестве контейнеров вы-

ступают сферы, прямоугольные параллелепипеды, цилиндры, капсулы и др. В рассматриваемой подсистеме разработаны эффективные алгоритмы нахождения информации (точек контакта, глубин проникновения и нормали расталкивания) о пересечении аппроксимирующих контейнеров. Полученные данные далее используются на стадии разрешения коллизий, в которой применяются математические модели контакта, удара и трения тел, формулируемые в терминах скоростей и импульсов.

В подсистеме моделируются также динамические объекты специальных типов. К ним относятся, например, системы частиц [33], состоящие из большого числа мелких элементов, каждый из которых порождается в определенный момент времени, движется под действием действующих на него сил и уничтожается через определенный промежуток времени. С помощью таких систем можно моделировать реактивные струи, дождь, снег и т.д. Расчет движения каждой частицы выполняется в независимом потоке на многоядерном графическом процессоре (GPU) с поддержкой архитектуры CUDA. Параллельное выполнение потоков на нескольких тысячах ядер GPU обеспечивает обработку миллионов частиц в масштабе реального времени. Также в подсистеме динамики в масштабе реального времени моделируются одномерные упругие объекты [34] (тросы, веревки, кабели и т.д.). Тросы считаются нерастяжимыми и подверженными деформациям изгиба и кручения. При моделировании задействованы параметры, в которые входит плотность троса, его длина, модуль Юнга, модуль сдвига и коэффициент затухания при кручении троса. Кроме параметров, трос задается ломаной линией, вершины которой задают положения соответствующих точек троса, а звенья – ориентацию соответствующих участков троса. С применением этой модели нерастяжимость троса реализуется с помощью метода множителей Лагранжа, что приводит к системе линейных уравнений с матрицей трехдиагонального вида, решаемой численным методом прогонки. В свою очередь изгиб и кручение троса моделируются на основе метода конечных элементов, применяемого для интегрирования потенциальной энергии деформации троса.

Динамика пружины с демпфером [35] реализуется с помощью ограничений специального типа (мягких ограничений), содержащих коэффициенты жесткости и демпфирования пружины. Вывод таких ограничений основан на применении неявной схемы Эйлера для интегрирования дифференциальных уравнений движения тел, соединенных пружиной. С помощью предлагаемого подхода моделируются рессоры подвесок колес, захватные устройства с пружиной, автоматически закрывающиеся двери и т.д. Для получения управляемыми объектами и оператором информации о внешней среде в системах СВО используются виртуальные датчики.

В данной подсистеме смоделировано несколько различных типов датчиков, описание которых представлено в таблице.



Таблица

## Моделируемые в СВО датчики

Тип датчика	Определяемые параметры
Датчик положения	Координаты центра объекта в мировой СК $x, y, z$
Датчик ориентации	Углы Эйлера ориентации объекта относительно осей мировой СК
Акселерометр	Вектор ускорения объекта в локальной СК
Датчик угловой скорости	Угловые скорости поворота объекта в локальной СК $\omega_x, \omega_y, \omega_z$
Датчик силомоментного ощущения	Сила и момент, приложенные к датчику от объекта внешней среды
Датчик касания	Факт столкновения рабочего органа объекта с другими объектами виртуального окружения
Дальномер	Расстояние от датчика до ближайшего виртуального объекта вдоль собственной линии визира датчика
Датчик интенсивности излучения	Уровень интенсивности измеряемого излучения в месте положения датчика

Показания активных датчиков на каждом шаге моделирования передаются на средства отображения или в подсистему управления для формирования управляющих сигналов в СВО. Рассмотренная подсистема динамики позволяет реалистично моделировать в масштабе реального времени сложные динамические поведения и взаимодействия объектов систем виртуального окружения.

### Подсистема визуализации

Подсистема визуализации осуществляет высококачественный рендеринг трехмерной сцены в СВО. Рендеринг выполняется в масштабе реального времени (т.е. один кадр формируется не более 40 мсек, а с учетом работы остальных подсистем для этого выделяется не более 10 мсек) и обеспечивает поддержку стереорежима, многопортового и многоэкранного отображения. Масштаб реального времени достигается за счет использования аппаратно-программных средств для параллельных вычислений на GPU, включая шейдеры. Применение последних позволяет заменять части стандартного графического конвейера на программируемые блоки. В данной подсистеме используются вершинный, тесселяционный, геометрический и фрагментный шейдеры. С их помощью обработка графических примитивов производится с высокой степенью параллелизма, одновременно на тысячах ядер графического процессора. Это позволяет быстро визуализировать сцены из нескольких миллионов полигонов и реализовать попиксельный расчет освещенности объектов в рамках сложных моделей освещения (например, Блинна, Кука-Торренса) с поддержкой нескольких типов источников света (всенаправленные, прожекторы, «фары»).

На рисунке 3а показан пример освещения фарой космического модуля «Поиск» с моделированием светового пятна. Подсистема также обеспечивает моделирование рассеянного освещения виртуальных объектов. Такой тип освещения характерен, например, для открытых пространств в условиях плотной облачности. Данная возможность реализована с применением алгоритмов, основанных на базе технологии SSAO (*Screen Space Ambient Occlusion*) [36].

На рисунке 3б представлен пример визуализации виртуальной модели космического модуля с применением технологии PBR.

В СВО для космических тренажеров часто встречаются сцены в расширенном диапазоне яркостей (HDR). Для визуализации таких сцен предусмотрен механизм нелинейной тональной компрессии светового диапазона в стандартный диапазон средств отображения.

Для повышения реалистичности виртуальных объектов в данной подсистеме используются материалы, включающие текстуры различных типов (отражения, прозрачности, нормалей, микрорельефа, цвета и яркости блика и др.) в любых сочетаниях, а также физически корректный рендеринг (PBR), основанный на материалах с текстурными картами альбеда, шероховатости и металлических свойств, которые определяют физические параметры этих материалов [37].

Реализована также поддержка мультитекстурования, при котором несколько материалов смешиваются на одной поверхности объекта с использованием текстурных масок [38]. Поддержка планарных, параболических и каскадных карт теней позволяет формировать жесткие и мягкие тени от виртуальных объектов. С помощью генерируемых в реальном времени кубических карт среды можно создавать в сцене отражающие объекты произвольной формы.

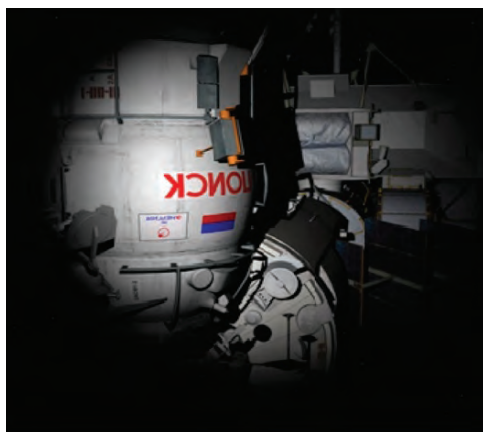


Рис. 3а. Освещение фарой модуля «Поиск»



Рис. 3б. Модель космического модуля с применением технологии PBR

Многопортовый режим отображения заключается в одновременном выводе на экран изображений из нескольких различных точек наблюдения (виртуальных камер) с целью более точного понимания положения динамического управляемого объекта в виртуальном окружении. С помощью многопортового режима можно создавать виртуальные модели специальных приборов, например, модель многоканального оптического прибора ВСК-4. На рисунке 4 показана модель многоканального оптического прибора ВСК-4. Многоэкранный режим отображения предполагает вывод визуализируемой виртуальной сцены одновременно на несколько экранов, что позволяет реализовывать различные панорамные изображения.

В рассматриваемой подсистеме визуализации реализована также ключевая анимация объектов. Ее идея заключается в том, что задаются положения и ориентации объекта в нескольких (ключевых) кадрах, а между ними подсистема автоматически вычисляет положения и ориентации по одному из выбранных пользователем контроллеров. В настоящее время поддерживаются контроллеры линейной интерполяции, кубические сплайны и сплайны Безье.

Отдельной составляющей подсистемы визуализации является моделирование в виртуальных камерах помех и бликов, часто встречающихся в оптических средствах наблюдения, а также эффекта расфокусировки. На рисунке 4 показан также пример моделирования оптических эффектов: расфокусировки, засветки и пересветки средства наблюдения.

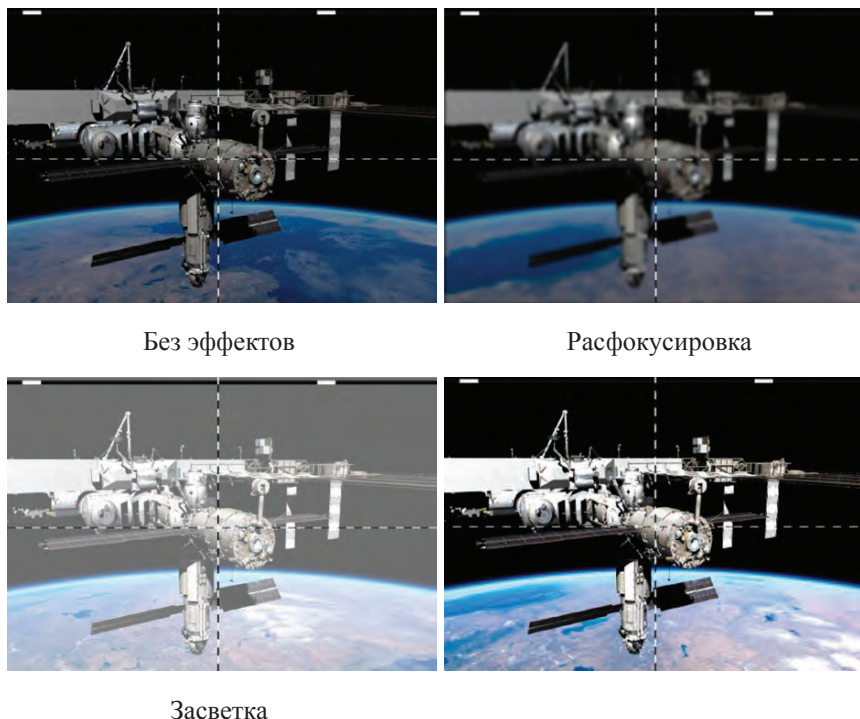


Рис. 4. Моделирование оптических эффектов в СВО

Создание СВО предполагает реализацию стереорежима, системы трекинга, динамики взаимодействия оператора с виртуальными объектами и т.д.

В данной подсистеме обеспечивается возможность погружения пользователя в виртуальную среду посредством стереовизуализации с использованием VR-гарнитуры Oculus Rift CV1, поляризационных, анаглифических и затворных стереочков, мониторов с чересстрочной разверткой и автостереоскопических мониторов Philips 3D-Solutions.

Во всех методах реализации стерео для космических тренажеров встает вопрос о степени адекватности виртуального окружения реальной обстановке.

Виртуальные объекты, которые кажутся наблюдателю расположенными на определенном расстоянии, должны казаться ему таких же размеров, как соответствующие реальные объекты, расположенные на таком же расстоянии.

Это обеспечивается заданием правильного соотношения между виртуальной и реальной системами координат, а также правильным заданием параметров виртуальных камер, образующих стереопару. Если параметры камер заданы неверно, наблюдатель все равно будет видеть стерео (т.е. объемное изображение), так как стерео создается в мозге человека. Однако это стерео будет неправильным, что приводит к нарушению видимых размеров объектов и расстояний между ними, а это может быть критичным во многих задачах, использующих систему визуализации. Для правильного задания параметров виртуальных камер необходимо знать положение и ориентацию наблюдателя. Как правило, данная информация обеспечивается системами трекинга. Рассмотренные возможности подсистемы визуализации обеспечивают высококачественный рендеринг виртуальных сцен в масштабе реального времени, достаточный для построения имитационно-тренажерных комплексов управления сложными динамическими системами.

### **Подсистема визуализации Земли**

Подсистема визуализации Земли (ПВЗ) работает совместно с основной подсистемой визуализации и предназначена для формирования в виртуальном окружении комплекса VIRSIM трехмерной модели Земли, которую можно использовать в космических видеотренажерах (в задачах стыковки, внекорабельной деятельности, мониторинга земной поверхности и др.). Неотъемлемой составляющей такой модели являются сверхбольшие текстуры ( $10^5 \times 10^5$  пикселей и выше), созданные на основе детальных спутниковых снимков Земли. Сверхбольшие текстуры не помещаются целиком в память видеокарты, поэтому для модели Земли не применима стандартная процедура подготовки 3D-моделей в системах типа 3ds Max, описанная в разделе 2. Вместо этого ПВЗ реализует собственную процедуру, основанную на построении для каждого кадра только видимой наблюдателю части модели Земли, что требует существенно меньше текстурных данных. Процесс построения и визуализации модели Земли выполняется в масштабе реального времени.

Адекватность формируемой модели Земли достигается решением следующих задач:

1) *Моделирование широкого комплекса визуальных свойств подстилающей поверхности (ПП) Земли* (рис. 5а), включающего:

- разрешение изображения ПП 100–150 м/пиксел и выше;
- глобальный географический охват (трассы всех суточных витков МКС);
- независимый слой с подвижным облачным покровом (имеется несколько вариантов с повышенной облачностью над морями, океанами и пустынными участками);
- возможность подключения комплектов изображений ПП для различных времен года (основным является комплект, соответствующий летнему периоду);
- наличие шельфовых зон, плавная стыковка берегов и водных поверхностей;
- отражение солнечного света от водных поверхностей;
- огни ночных городов на теневой стороне Земли (рис. 5б).



Рис. 5а. Моделирование ПП Земли в дневных условиях



Рис. 5б. Моделирование огней ночных городов Земли

Моделируемые визуальные свойства ПП соответствуют требованиям по наблюдению Земли невооруженным глазом с борта МКС [39], а также обеспечивают формирование адекватного изображения Земли в средствах наблюдения пилотируемых космических кораблей на этапах полета от 100 км и выше. Ключевой особенностью ПВЗ является визуализация ПП с детализацией, наиболее близкой к разрешению средства наблюдения (экрана) [40]. Это позволяет синтезировать качественные изображения земной поверхности, т.е. без выраженного эффекта «ложной детализации» (муара), который возникает в случае, если детализация изображения превышает разрешение средства наблюдения.

2) *Моделирование геометрических особенностей (рельефа) земной поверхности*, демаскируемых при визуальном-приборном наблюдении Земли. ПВЗ осуществляет моделирование рельефа Земли с шагом по высоте до 35 метров на основе эллипсоида вращения, определенного в действующем международном стандарте WGS-84 (World Geodetic System 1984) [41]. При

моделировании наблюдения Земли невооруженным глазом демаскирование рельефа осуществляется за счет формирования динамических протяженных теней от гор и возвышенностей. При использовании увеличивающих космических средств наблюдения, в том числе фото- и телекамер, рельеф Земли визуально самопроявляется на синтезируемых изображениях [42].

3) *Физически корректное моделирование динамической светотеневой обстановки Земли.* ПВЗ осуществляет расчет адекватной светотеневой обстановки для произвольного взаимного расположения наблюдателя, Солнца и Земли на основе уравнения переноса излучения через атмосферный слой, состоящий из молекул воздуха и аэрозольных частиц [43]. При этом учитываются такие явления, как поглощение, многократное рассеяние света в атмосфере, а также переотражение света от земной поверхности. ПВЗ обеспечивает моделирование следующих ключевых составляющих светотеневой обстановки Земли (рис. 6а и 6б):

- зона терминатора с областью сумерек (когда Солнце за горизонтом);
- изменение плотности и цвета атмосферы на поверхности Земли и горизонте;
- рассветное и закатное освещение Земли;
- изменение спектра освещения модели МКС при восходе и закате;
- мягкие неоднородные тени от облачного покрова;
- плотные протяженные тени от гор и др.



Рис. 6а. Моделирование зоны терминатора, сумерек

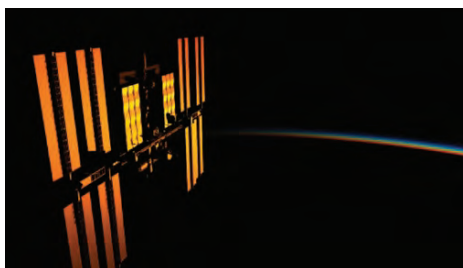


Рис. 6б. Моделирование закатного освещения МКС

Расчет светотеневой обстановки выполняется в расширенном диапазоне яркостей, что позволяет визуализировать сложные оттенки рассветного и закатного освещения, а также корректно учитывать вклад освещенности Земли при одновременном использовании естественного и искусственного освещения, например, при подсветке фарой стыковочного узла, находящегося в тени станции, в дневное время суток.

4) *Моделирование сеанса орбитального наблюдения Земли.* Кроме визуальной составляющей в ПВЗ также реализовано моделирование видимого в космическом средстве наблюдения движения земной поверхности («бега местности») вдоль подспутниковой трассы МКС. Расчет видимых на экране точек модели Земли осуществляется на основе классического уравнения

Кеплера (для эллиптической орбиты) с использованием разработанной модифицированной вычислительной схемы [44], которая обеспечивает:

- возможность задания произвольного участка суточной трассы МКС (поддерживаются все суточные витки МКС);
- постоянство заданного направления «бега местности» в средстве наблюдения;
- отсутствие погрешностей позиционирования, влияющих на качество синтезируемых изображений.

Также в ПВЗ реализован интерфейс оперативного управления сеансом орбитального наблюдения модели Земли, включающий в себя панель орбитального наблюдателя (рис. 7) и управление скоростью течения модельного времени с помощью горячих клавиш.

Панель можно вызвать в процессе визуализации, изменить в ней значения параметров (орбиты МКС, модели Земли, участка трассы и др.) и, применив их, немедленно видеть соответствующие изменения в визуализации Земли. Горячие клавиши позволяют ускорить, замедлить, остановить, обратить модельное время, восстановить реальную скорость течения времени, переместиться в начало или конец заданного участка трассы.

Рассмотренная подсистема визуализации Земли обеспечивает построение и визуализацию в масштабе реального времени высококачественной виртуальной модели Земли, применимой для решения широкого круга актуальных и перспективных задач космических имитационно-тренажерных комплексов.



Рис. 7. Вид панели орбитального наблюдателя

## Подсистема управления

Подсистема управления используется для воздействия на виртуальные объекты в процессе моделирования. В созданной СВО такими объектами являются колесные и гусеничные роботы, космические летательные аппараты, манипуляторы, камеры, источники освещения и т.д. Управление ими может осуществляться различными способами с помощью реальных устройств или их моделей. Примерами таких устройств являются пульты для дистанционного управления объектами [45].

В разработанной СВО управление объектами осуществляется с использованием функциональных схем [46]. Функциональная схема представляет собой графическое представление алгоритма управления виртуальными объектами в виде блоков, соединенных друг с другом. На входы схемы поступают управляющие команды от виртуальных или реальных пультов, сигналы от реальных устройств (например, значения углов датчиков в экзоскелете), показания виртуальных датчиков и т.д. На выходах функциональной схемы формируются управляющие сигналы, подаваемые на исполнительные устройства (электрические и реактивные двигатели, приводы ведущих колес, камеры, источники освещения и т.д.) виртуальных объектов. В рамках СВО была предложена технология создания и расчета функциональных схем. Для этого был разработан графический редактор, поддерживающий обширную библиотеку блоков (арифметических, логических, тригонометрических, цифровых, операции из теории автоматического управления, голосового управления и т.д.), из которых можно создавать различные по сложности функциональные схемы.

Расчет функциональной схемы осуществляется с помощью модуля вычисления управляющих сигналов, реализованного в виде динамической библиотеки. Этот модуль в режиме реального времени вычисляет значения на выходах функциональных схем по значениям их входов.

Командный режим управления в СВО реализуется с помощью виртуальных пультов [47]. Для этого разработан специальный графический редактор, в котором виртуальный пульт создается из заданного набора визуальных элементов (кнопок, джойстиков, регуляторов и т.д.). На рисунке 8а показан пример схемы в редакторе, предназначенной для управления виртуальным колесным роботом, который содержит камеру и две фары. На рисунке 8б показан пример виртуального пульта, созданного в этом редакторе и предназначенного для управления манипулятором «Эра».

В рассматриваемой СВО реализованы следующие режимы управления:

- командный режим с применением виртуальных пультов;
- копирующий режим с помощью экзоскелета, Кинекта и других систем трекинга;
- супервизорный режим на основе голосового и жестового интерфейсов;
- полуавтоматический режим с помощью джойстика и расчета обратной кинематики;



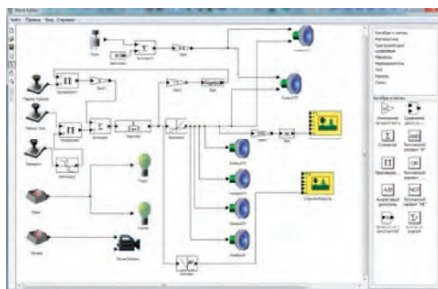


Рис. 8а. Схема управления колесным роботом в редакторе функциональных схем



Рис. 8б. Виртуальный пульт управления транспортным манипулятором

- программный режим путем записи и воспроизведения скриптов;
- адаптивный режим на основе показаний виртуальных датчиков.

Командный режим управления подходит для изменения состояния объектов, а также для управления манипуляторами и роботами с малым количеством степеней свободы. Управление объектами в командном режиме осуществляется с помощью кнопок и переключателей виртуального пульта, которые отвечают за выполнение конкретных операций.

Копирующий режим заключается в том, что движения человека-оператора с помощью специальных устройств передаются в СВО, а виртуальные объекты повторяют (копируют) их. В данной СВО такой подход может применяться для управления антропоморфным роботом с помощью экзоскелета и беспроводных систем трекинга. Экзоскелет представляет собой костюм, надеваемый на тело человека. Движения человека определяются по датчикам, установленным на экзоскелете, и передаются в СВО. Альтернативой экзоскелету служат беспроводные системы трекинга. Они могут быть встроены непосредственно в VR-гарнитуру или базироваться на оптических элементах, электромагнитных датчиках, устройстве Microsoft Kinect и др. Для взаимодействия оператора с объектами СВО также необходимы системы трекинга.

На рисунке 9а показан пример управления моделью торсового антропоморфного робота в копирующем режиме с применением системы Microsoft Kinect, которая определяет опорные точки торса и рук оператора. На рисунке 9б показана виртуальная модель марсохода, в которой для управления буром и солнечными панелями применяется программный режим.

Супервизорный режим управления заключается в том, что человек-оператор задает специальным образом команду для выполнения виртуальным объектом заданного набора действий. Реализация супервизорного режима в СВО осуществляется с помощью голосового и/или жестового интерфейса. При голосовом интерфейсе оператор задает команду голосом, затем эта команда распознается и выполняется. Жестовый интерфейс заключается в том, что команда задается с помощью распознаваемой позы или последовательности положений рук человека. Отличие супервизорного режима состоит



Рис. 9а. Управление антропоморфным роботом в копирующем режиме

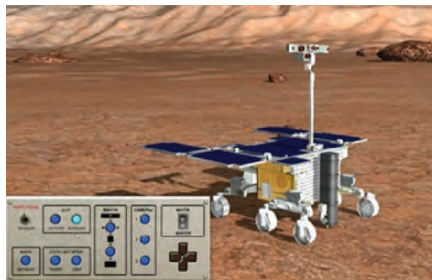


Рис. 9б. Управление марсоходом в программном режиме

в том, что оператор может быть освобожден от манипуляций с оконечными устройствами управления объектами, то есть применяется дистанционный вид интерфейса «человек–робот».

Полуавтоматический режим в СВО был применен для управления манипуляционными роботами [48]. Его идея состоит в том, что оператор с помощью джойстика виртуального или реального пульта задает лишь движение рабочего органа робота, в то время как формирование согласованных движений в приводах робота осуществляется автоматически. Реализация полуавтоматического управления роботом осуществляется путем расчета инверсной кинематики методом покоординатного спуска. В этом методе итерационно выполняются повороты в шарнирах робота, при которых достигается максимум целевой функции, сформированной относительно невязки между текущими и желаемыми координатами рабочего органа робота. Полуавтоматическое управление манипуляционными роботами предпочтительнее командного, так как требует меньше усилий для выставления рабочего органа робота в требуемое положение.

Программный режим заключается в том, что управление объектами осуществляется по заранее составленной программе. Для его реализации в СВО был применен подход на основе записи и воспроизведения скрипта. Сначала осуществляется запись скрипта для выполнения конкретной операции. Один из вариантов формирования скрипта заключается в записи данных при выполнении операции оператором. Данные хранятся в виде лог-файлов, в которых фиксируются все промежуточные положения, ориентации, параметры и состояния виртуальных объектов. Управление объектом в программном режиме реализуется путем запуска подготовленного скрипта по команде из системы управления. Программный режим хорошо подходит для типовых и повторяющихся операций, выполняемых в детерминированной среде.

Адаптивный режим состоит в том, что управление виртуальными объектами формируется в зависимости от изменений параметров объекта или внешних возмущений, которые действуют на объект. В СВО такой тип управления реализуется с помощью виртуальных датчиков сенсорного очувствления (положения, ориентации, скорости, силы и т.д.). Также рассматри-

ваемый тип управления позволяет реализовать поддержание требуемого значения управляемого параметра с помощью ПИД (пропорционально-интегрально-дифференцирующего) регулятора. Адаптивный режим подходит для решения такого типа задач, как выполнение антропоморфным роботом операции поворота вентиля панели управления космического модуля, когда движение манипулятора антропоморфного робота осуществляется по траектории, которая формируется на основе целевых координат положения кисти руки робота и его пальцев, а взаимодействие руки с вентиляем – по значениям сил и моментов, которые робот оказывает на вентиль.

## Заключение

В свете активно разрабатываемых перспективных проектов освоения Луны разработка СВО с расширенными возможностями моделирования, выполняемых в миссии операций, представляется своевременной, и, более того, отработанность основных модулей и зрелость методологии позволяют практически незамедлительно перейти к постановке лабораторных и полунатурных исследований и испытаний в интересах моделирования новых видов деятельности и специфических условий, влияющих на взаимодействие в системе «человек–техника».

*Публикация выполнена в рамках государственного задания по проведению фундаментальных научных исследований (ГП 14) по теме (проекту) «34.9. Системы виртуального окружения: технологии, методы и алгоритмы математического моделирования и визуализации» (0065-2019-0012).*

## ЛИТЕРАТУРА

- [1] NASA's Eyes. URL: [https://en.wikipedia.org/wiki/NASA%27s\\_Eyes](https://en.wikipedia.org/wiki/NASA%27s_Eyes) (accessed 24.01.2020).
- [2] Unity. URL: <https://unity.com/> (accessed 24.01.2020).
- [3] Engineering DOUG Graphics for Exploration (EDGE). URL: <https://software.nasa.gov/featuredsoftware/edge> (accessed 24.01.2020).
- [4] UNIGINE. URL: <https://unigine.com/> (accessed 24.01.2020).
- [5] Google Earth. URL: <https://earth.google.com/web> (accessed 24.01.2020).
- [6] NASA World Wind. URL: <http://worldwind.arc.nasa.gov> (accessed 24.01.2020).
- [7] Celestia. URL: <https://celestia.space> (accessed 24.01.2020).
- [8] Orbiter. URL: <http://orbit.medphys.ucl.ac.uk> (accessed 24.01.2020).
- [9] Outerra. // URL: <http://www.outerra.com/> (accessed 24.01.2020).
- [10] PhysX Nvidia. URL: <https://www.nvidia.ru/object/nvidia-physx-ru.html> (accessed 24.01.2020).
- [11] Bullet. URL: <https://pybullet.org/wordpress/> (accessed 24.01.2020).
- [12] Open Dynamics Engine. URL: <http://www.ode.org/> (accessed 24.01.2020).
- [13] Havok. URL: <https://www.havok.com/> (accessed 24.01.2020).
- [14] Adams. The Multibody Dynamics Simulation Solution. URL: <http://www.mscsoftware.com/product/adams> (accessed 24.01.2020).

- [15] Simpack. Multibody Simulation Software. URL: <http://www.simpack.com/> (accessed 24.01.2020).
- [16] Универсальный механизм. URL: <http://www.umlab.ru> (accessed 24.01.2020).
- [17] Euler (Эйлер). URL: <http://www.euler.ru/> (accessed 24.01.2020).
- [18] Simulink. URL: <http://matlab.ru/products/simulink> (accessed 24.01.2020).
- [19] Scicos: Block diagram modeler/simulator. URL: <http://www.scicos.org> (accessed 24.01.2020).
- [20] Dymola и проектирование систем. URL: <https://www.3ds.com/ru/produkty-i-uslugi/catia/produkty/dymola/> (accessed 24.01.2020).
- [21] LabView. URL: <http://www.ni.com/labview> (accessed 24.01.2020).
- [22] Carpin S., Lewis M., Wang J., Balakirsky S. and Scrapper C. USARSim: a robot simulator for research and education. In Proceedings of the IEEE conference on robotics and automation (ICRA). – 2007. – pp. 1400–1405.
- [23] Koenig N. and Howard A. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS). – 2004. – pp. 2149–2154.
- [24] Rohmer E., Singh S.P.N. and Freese M. V-REP: A versatile and scalable robot simulation framework. In International Conference on Intelligent Robot and Systems. – 2013. – pp. 1321–1326.
- [25] Michel O. Webots: Professional mobile robot simulation. International Journal of Advanced Robotics Systems. – 2004. – Vol. 1. – No 1.
- [26] Pinciroli C., Trianni V., O'Grady R. and others. ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems. Swarm Intelligence. – 2012. – No 6(4). – pp. 271–295.
- [27] Масалкин А.И., Торгашев М.А. Опыт использования систем имитации визуальной обстановки в тренажерах пилотируемых космических аппаратов // Пилотируемые полеты в космос. – 2015. – № 2. – С. 36–42.
- [28] Erleben K. Stable, robust and versatile multibody dynamics animation, PhD thesis, Dept. of Computer Science, Univ. of Copenhagen, 2004.
- [29] Catto E. Iterative dynamics with temporal coherence // In Game Developer Conference. – 2005. – pp. 1–24.
- [30] Mikhaylyuk M.V., Strashnov E.V. and Timokhin P.Yu. Algorithms of multibody dynamics simulation using articulated-body method, Mathematica Montisnigri, Vol. XXXIX. – 2017. – pp. 133–145.
- [31] Михайлюк М.В., Трушин А.М. Алгоритмы определения коллизий сфер на GPU // Программная инженерия. – 2017. – Том 8. – № 8. – С. 354–358.
- [32] Страшнов Е.В., Тимохин П.Ю. Блочный подход для разрешения коллизий виртуальных объектов // Вестник кибернетики. – 2018. – № 3. – С. 91–101.
- [33] Мальцев А.В. Реализация системы частиц в реальном времени на GPU. Программные продукты и системы // Программные продукты и системы. – 2014. – № 4. – С. 57–62.
- [34] Страшнов Е.В., Торгашев М.А. Моделирование динамики одномерных упругих объектов в системах виртуального окружения // Вестник кибернетики. – 2018. – № 3. – С. 41–50.
- [35] Страшнов Е.В., Торгашев М.А., Тимохин П.Ю. Моделирование пружин в системах виртуального окружения с помощью метода мягких ограничений // Информационные технологии и вычислительные системы. – 2017. – № 3. – С. 70–78.

- [36] Методы распределенной имитации непрямого затенения виртуальной среды на GPU в реальном времени // Информационные технологии и вычислительные системы. – 2019. – Т. 9. – № 3. – С. 16–24.
- [37] Мальцев А.В., Страшнов Е.В. Методы визуализации 3D-объектов с учетом физических свойств их материалов // Виртуальное моделирование, прототипирование и промышленный дизайн. Материалы V Международной научно-практической конференции. – 2018. – Вып. 5. – Т. 1. – С. 616–619.
- [38] Мальцев А.В., Тимохин П.Ю., Трушин А.М. Методы распределенной визуализации виртуальных объектов с использованием мультитекстурирования на основе смешивания материалов // Вестник кибернетики. – 2018. – № 1. – С. 110–115.
- [39] Митин А.И., Брагин В.И. Пути повышения адекватности моделирования визуальных условий мониторинга земной поверхности на тренажере служебного модуля российского сегмента Международной космической станции // Пилотируемые полеты в космос. – 2014. – № 3. – С. 60–70.
- [40] Тимохин П.Ю. Система визуализации текстурированных моделей планет для тренировок проведения космических экспериментов // Программные продукты и системы. – 2015. – № 4. – С. 99–104.
- [41] Тимохин П.Ю., Михайлюк М.В., Мальцев А.В. Построение на GPU в масштабе реального времени адаптивной модели рельефа Земли на основе эллипсоида вращения // International Journal of Open Information Technologies. – 2019. – Т. 7. – № 10. – С. 22–35.
- [42] Тимохин П.Ю., Михайлюк М.В. Метод сжатия разрядности карт высот на основе критерия визуальной значимости // Труды НИИСИ РАН. – 2017. – Т. 7. – № 1. – С. 30–35.
- [43] Тимохин П.Ю. Моделирование освещения Земли в реальном времени с учетом атмосферы // Труды НИИСИ РАН. – 2014. – Т. 4. – № 2. – С. 78–83.
- [44] Тимохин П.Ю. Моделирование видимого движения Земли вдоль участков суточной трассы МКС в космических видеотренажерах // Труды НИИСИ РАН. – 2019. – Т. 9. – № 6. – С. 111–117.
- [45] Михайлюк М.В., Крючков Б.И., Усов В.М. Виды интерфейса для дистанционного взаимодействия космонавтов с автономными мобильными роботами при внекорабельной деятельности на лунной поверхности. // Пилотируемые полеты в космос. – 2017. – № 4(25). – С. 41–53.
- [46] Михайлюк М.В., Торгашев М.А. Визуальный редактор и модуль расчета функциональных схем для имитационно-тренажерных комплексов // Программные продукты и системы. – 2014. – № 4. – С. 10–15.
- [47] Михайлюк М.В., Омельченко Д.В., Страшнов Е.В. Командный и супервизорный режимы управления виртуальными роботами // Вестник кибернетики. – 2016. – № 4. – С. 79–84.
- [48] Страшнов Е.В., Михайлюк М.В. Моделирование полуавтоматического режима управления манипуляционными роботами в системах виртуального окружения // Вестник кибернетики. – 2017. – № 4. – С. 191–198.
- [49] Страшнов Е.В., Михайлюк М.В. Методы силового управления манипуляционными роботами в системах виртуального окружения // International Journal of Open Information Technologies. – 2019. – Том 7. – № 9. – С. 39–45.

## REFERENCES

- [1] NASA's Eyes. URL: [https://en.wikipedia.org/wiki/NASA%27s\\_Eyes](https://en.wikipedia.org/wiki/NASA%27s_Eyes) (accessed 24.01.2020).
- [2] Unity. URL: <https://unity.com/> (accessed 24.01.2020).
- [3] Engineering DOUG Graphics for Exploration (EDGE). URL: <https://software.nasa.gov/featuredsoftware/edge> (accessed 24.01.2020).
- [4] UNIGINE. URL: <https://unigine.com/> (accessed 24.01.2020).
- [5] Google Earth. URL: <https://earth.google.com/web> (accessed 24.01.2020).
- [6] NASA World Wind. URL: <http://worldwind.arc.nasa.gov> (accessed 24.01.2020).
- [7] Celestia. URL: <https://celestia.space> (accessed 24.01.2020).
- [8] Orbiter. URL: <http://orbit.medphys.ucl.ac.uk> (accessed 24.01.2020).
- [9] Outerra. // URL: <http://www.outerra.com/> (accessed 24.01.2020).
- [10] PhysX Nvidia. URL: <https://www.nvidia.ru/object/nvidia-physx-ru.html> (accessed 24.01.2020).
- [11] Bullet. URL: <https://pybullet.org/wordpress/> (accessed 24.01.2020).
- [12] Open Dynamics Engine. URL: <http://www.ode.org/> (accessed 24.01.2020).
- [13] Havok. URL: <https://www.havok.com/> (accessed 24.01.2020).
- [14] Adams. The Multibody Dynamics Simulation Solution. URL: <http://www.mscsoftware.com/product/adams> (accessed 24.01.2020).
- [15] Simpack. Multibody Simulation Software. URL: <http://www.simpack.com/> (accessed 24.01.2020).
- [16] Multi-purpose mechanism URL: <http://www.umlabor.ru> (accessed 24.01.2020).
- [17] Euler. URL: <http://www.euler.ru/> (accessed 24.01.2020).
- [18] Simulink. URL: <http://matlab.ru/products/simulink> (accessed 24.01.2020).
- [19] Scicos: Block diagram modeler/simulator. URL: <http://www.scicos.org> (accessed 24.01.2020).
- [20] Dymola and systems design. URL: <https://www.3ds.com/ru/produkty-i-uslugi/catia/produkty/dymola/> (accessed 24.01.2020).
- [21] LabView. URL: <http://www.ni.com/labview> (accessed 24.01.2020).
- [22] Carpin S., Lewis M., Wang J., Balakirsky S. and Scrapper C. USARSim: a robot simulator for research and education. In Proceedings of the IEEE Conference on Robotics and Automation (ICRA). – 2007. – pp. 1400–1405.
- [23] Koenig N. and Howard A. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS). – 2004. – pp. 2149–2154.
- [24] Rohmer E., Singh S.P.N. and Freese M. V-REP: A versatile and scalable robot simulation framework. In International Conference on Intelligent Robot and Systems.– 2013. – pp. 1321–1326.
- [25] Michel O. Webots: Professional mobile robot simulation. International Journal of Advanced Robotics Systems. – 2004. – Vol. 1. – No 1.
- [26] Pinciroli C., Trianni V., O'Grady R. and others. ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems. Swarm Intelligence. – 2012. – No 6(4). – pp. 271–295.
- [27] Masalkin A.I., Torgashev M.A. Experience of the use of visual environment modeling systems in simulators of manned space vehicles. // Scientific Journal *Manned Space-flight*, Star City. – 2015. – No 2. – pp. 36–42.

- [28] Erleben K. Stable, robust and versatile multibody dynamics animation, PhD thesis, Dept. of Computer Science, Univ. of Copenhagen, 2004.
- [29] Catto E. Iterative dynamics with temporal coherence // In Game Developer Conference. – 2005. – pp. 1–24.
- [30] Mikhaylyuk M.V., Strashnov E.V., Timokhin P.Yu. Algorithms of multibody dynamics simulation using articulated-body method, *Mathematica Montisnigri*. – 2017. – Vol. XXXIX. – pp. 133–145.
- [31] Mikhaylyuk M.V., Trushin A.M. Spheres collision detection algorithms on GPU // *Journal “Software Engineering”*. – 2017. – Vol. 8. – No 8. – pp. 354–358.
- [32] Strashnov E.V., Timokhin P.Yu. Modular approach for resolving collisions of virtual objects // *Proceedings in Cybernetics*. – 2018. – No 3. – pp. 91–101.
- [33] Mal'tsev A.V. Real-time particle system realization on GPU // *Programmnye produkty i sistemy [Software & Systems]*. – 2014. – No 4. – pp. 57–62.
- [34] Strashnov E.V., Torgashev M.A. Modeling the dynamics of monodimensional elastic objects in virtual environment systems // *Proceedings in Cybernetics [Vestnik kibernetiki]*. – 2018. – No 3. – pp. 41–50.
- [35] Strashnov E.V., Torgashev M.A., Timokhin P.Yu. Simulation of springs with soft constraints method in virtual environment systems // “*Informacionnye tekhnologii I I vichslitel'nye sistemy*” [Journal of Information Technologies and Computing Systems]. – 2017. – No 3. – pp. 70–78.
- [36] Methods for real-time distributed imitation of indirect shading in virtual environment on GPU // “*Informacionnye tekhnologii I I vichslitel'nye sistemy*” [Journal of Information Technologies and Computing Systems]. – 2019. – Vol. 9. – No 3. – pp. 16–24.
- [37] Maltsev A.V., Strashnov E.V. Visualization methods for 3D-objects with reference to physical properties of their materials // *Virtual modeling, prototyping and industrial design. Proceedings of the V International Scientific and Practical Conference 2018*. – Issue. 5. – V. 1. – pp. 616–619.
- [38] Maltsev A.V., Timokhin P.Yu., Trushin A.M. Methods for distributed visualization of virtual objects using multitexturing based on material blending // *Proceedings in Cybernetics*. – 2018. – No 1. – pp. 110–115.
- [39] Mitin A.I., Bragin V.I. Ways of improving the adequacy of modeling visual conditions of the earth surface monitoring on the service module simulator of the ISS RS // *Scientific Journal Manned Spaceflight*. – 2014. – No 3. – pp. 60–70.
- [40] Timokhin P.Yu. Textured planet model visualization system for space experiments simulation // *Programmnye produkty i sistemy [Software & Systems]*. – 2015. – No 4. – pp. 99–104.
- [41] Timokhin P.Yu., Mikhaylyuk M.V., Maltsev A.V. Real-time construction on the GPU of adaptive terrestrial relief model based on the spheroid // *International Journal of Open Information Technologies*. – 2019. – Vol. 7. – No 10. – pp. 22–35.
- [42] Timokhin P.Yu., Mikhaylyuk M.V. The method of reducing the resolution of height-maps based on the criterion of visual significance // *Proceedings of NIISI RAS*. – 2017. – V. 7. – No 1. – pp. 30–35.
- [43] Timokhin P.Yu. Simulation of the Earth's illumination in real time against the background of the atmosphere // *Proceedings of NIISI RAS*. – 2014. – V. 4. – No 2. – pp. 78–83.
- [44] Timokhin P.Yu. Modeling the apparent motion of the Earth along the sections of the ISS daily route in space video simulators // *Proceedings of NIISI RAS*. – 2019. – Vol. 9. – No 6. – pp. 111–117.

- [45] Mikhaylyuk M.V., Kryuchkov B.I., Usov V.M. Options of interfaces for the remote interaction of cosmonauts with autonomous mobile robots during extravehicular activity on the lunar surface // *Scientific Journal Manned Spaceflight*. – 2017. – Vol. 25. – No 4. – pp. 41–53.
- [46] Mikhailyuk M.V., Torgashev M.A. The visual editor and calculation module of block diagrams for simulation and training complexes // *Programmnye produkty i sistemy [Software & Systems]*. – 2014. – No 4. – pp. 10–15.
- [47] Mikhailyuk M.V., Omel'chenko D.V., Strashnov E.V. Command and supervisory modes for virtual robot control // *Proceedings in Cybernetics*. – 2016. – No 4. – pp. 79–84.
- [48] Strashnov E.V., Mikhailyuk M.V. Simulation of semi-automatic control of manipulator robots // *Proceedings in Cybernetics*. – 2017. – No 4. – pp. 191–198.
- [49] Strashnov E.V., Mikhailyuk M.V. Methods for robotic manipulator force control in virtual environment systems // *International Journal of Open Information Technologies*. – 2019. – Vol. 7. – No 9. – pp. 39–45.