

Ю. Б. Колесов, Ю. Б. Сениченков

МОДЕЛИРОВАНИЕ СИСТЕМ

ПРАКТИКУМ ПО КОМПЬЮТЕРНОМУ МОДЕЛИРОВАНИЮ

Технология разработки моделей сложных динамических систем в среде визуального моделирования *MvStudio*

Типовые вычислительные эксперименты

Учебные задания по компьютерному моделированию

Численная библиотека среды

Компакт-диск с моделями и демонстрационной версией среды

+ CD



УЧЕБНОЕ ПОСОБИЕ



Ю. Б. Колесов
Ю. Б. Сениченков

МОДЕЛИРОВАНИЕ СИСТЕМ

ПРАКТИКУМ ПО КОМПЬЮТЕРНОМУ МОДЕЛИРОВАНИЮ

Рекомендовано Учебно-методическим объединением по университетскому политехническому образованию в качестве учебного пособия для студентов высших учебных заведений, обучающихся по направлению подготовки 220100 – «Системный анализ и управление»

Санкт-Петербург
«БХВ-Петербург»
2007

УДК 681.3.06
ББК 32.973.26
К60

Колесов, Ю. Б.

К60 Моделирование систем. Практикум по компьютерному моделированию / Ю. Б. Колесов, Ю. Б. Сениченков. — СПб.: БХВ-Петербург, 2007. — 352 с.: ил. + CD-ROM

ISBN 5-94157-580-7

Учебное пособие содержит: описание технологии разработки моделей сложных динамических систем и типовых вычислительных экспериментов, набор учебных заданий и описание численной библиотеки MvStudium. На прилагаемом к книге компакт-диске помещены все рассматриваемые в этом томе модели и демонстрационная версия среды визуального моделирования MvStudium. Книга является последней из трех, объединенных общим названием МОДЕЛИРОВАНИЕ СИСТЕМ.

Для студентов вычислительных специальностей технических вузов

УДК 681.3.06
ББК 32.973.26

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Людмила Еремеевская</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Анна Кузьмина</i>
Компьютерная верстка	<i>Натальи Смирновой</i>
Корректор	<i>Наталья Першакова</i>
Дизайн серии	<i>Игоря Цырульникова</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Рецензенты:

Евгений Г. Б., д.т.н., профессор кафедры «Компьютерные системы автоматизации производств» МГТУ им. Баумана;

Ивановский Р. И., д.т.н., профессор кафедры «Распределенные вычисления и компьютерные сети» СПбГПУ.

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 30.11.06.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 28,38.

Тираж 1500 экз. Заказ №

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"

199034, Санкт-Петербург, 9 линия, 12

ISBN 5-94157-580-7

© Колесов Ю. Б., Сениченков Ю. Б., 2007
© Оформление, издательство "БХВ-Петербург", 2007

Оглавление

ПРЕДИСЛОВИЕ	1
ГЛАВА 1. ТИПОВЫЕ МОДЕЛИ: РАЗРАБОТКА И ОТЛАДКА.....	3
Создание и отладка модели непрерывной системы	3
Создание и отладка модели гибридной системы.....	31
Создание и отладка структурной модели	55
ГЛАВА 2. ТИПОВЫЕ ВЫЧИСЛИТЕЛЬНЫЕ ЭКСПЕРИМЕНТЫ	93
Реализация определенного сценария поведения	93
Сравнение вариантов.....	96
Параллельная композиция вариантов.....	97
Последовательная композиция вариантов	99
Получение параметрической зависимости.....	100
Параметрическая оптимизация	105
Сравнение результатов моделирования с внешними данными	109
Проведение статистических испытаний.....	115
ГЛАВА 3. УЧЕБНЫЕ ЗАДАНИЯ	121
Динамические системы	121
Задание 1	121
Задание 2.....	123
Задание 3.....	124
Задание 4.....	126
Задание 5.....	132
Гибридные автоматы.....	134
Задание 6.....	134
Задание 7.....	136

Задание 8	138
Задание 9	139
Компонентные модели	141
Задание 10	141
Модели общего вида	144
Задание 11	144
Задание 12	146
ГЛАВА 4. ЧИСЛЕННАЯ БИБЛИОТЕКА MvSTUDIUM	165
Матричные вычисления	166
Использование MvStudium как матричного калькулятора	166
Задача 1	172
Задача 2	184
Задача 3	195
Решение нелинейных уравнений.....	201
Почему так трудно решать нелинейные уравнения?	201
Задача 4	214
Задача 5	220
Задача 6	220
Задача 7	223
Задача 8	224
Решение обыкновенных дифференциальных уравнений	229
Простейшие методы и их свойства	232
Жесткие задачи	243
Колебательные задачи	259
Странные аттракторы. Отображение Пуанкаре	275
Уравнения, не приведенные к форме Коши	284
Решение алгебро-дифференциальных уравнений	289
ПРИЛОЖЕНИЕ 1. ТЕСТОВЫЕ МАТРИЦЫ С ИЗВЕСТНОЙ ОБРАТНОЙ	299
ПРИЛОЖЕНИЕ 2. БИБЛИОТЕКА ПАКЕТА MvSTUDIUM.....	309
Организация библиотеки	310
Структура решателей.....	312

Автоматические решатели	312
Решатели для задач конкретного типа	312
Решатели для отладки	313
Системы нелинейных алгебраических уравнений	314
Программы для решения нелинейных уравнений	315
Программа-автомат	318
Дифференциальные уравнения	320
Программы для решения нежестких уравнений	322
Программы для решения жестких уравнений	324
Программа-автомат	326
Отладочные программы	328
Алгебро-дифференциальные уравнения	329
Явный способ	330
Неявный способ	331
Программа-автомат для решения алгебро-дифференциальных уравнений	333
ПРИЛОЖЕНИЕ 3. ОПИСАНИЕ КОМПАКТ-ДИСКА	335
ЛИТЕРАТУРА	337
Дополнительная литература по главам	337
К главе 1	337
К главам 3 и 4	338

Предисловие

Вы держите в руках третью книгу по моделированию систем — "Практикум по компьютерному моделированию".

Она является завершающей в серии книг "Моделирование систем. Динамические и гибридные системы" [5], "Моделирование систем. Объектно-ориентированный подход" [6] и демонстрирует, как использовать объектно-ориентированный подход при моделировании сложных динамических систем, а также может служить руководством по моделированию сложных динамических систем в графической среде MvStudium.

Глава 1 посвящена вопросам технологии создания и отладки непрерывных, гибридных и многокомпонентных моделей систем. В ней читатель найдет последовательно усложняющиеся модели движения снаряда, полета неуправляемой и управляемой ракеты. Рассматриваемые модели, конечно же, являются упрощенными, однако описанный процесс их разработки отражает существенные особенности, характерные для моделирования технических систем на ранних этапах проектирования.

Глава 2 посвящена систематизации типичных для проектирования сложных динамических систем вычислительных экспериментов. В ней строятся типовые схемы проведения экспериментов и воплощаются в графической среде MvStudium. Сравнивая между собой примеры из книги "Моделирование систем. Объектно-ориентированный подход" [6], и примеры, приведенные в этой главе, можно сравнить между собой различные реализации и оценить достоинства объектно-ориентированного подхода.

Глава 3 содержит учебные задания, которые можно использовать при изучении курсов "Основы моделирования", "Компьютерные вычисления", "Моделирование механических систем". Задачи классифицированы с точки зрения используемых пользовательских интерфейсов пакета MvStudium.

Глава 4 посвящена использованию MvStudium как специализированной численной библиотеки. Встроенная библиотека для решения алгебро-дифференциальных уравнений и достаточно мощный алгоритмический язык позволяют организовать двухэтапное обучение многим разделам современного численного анализа. Изобразительные возможности среды позволяют формулировать задачи на "естественном" математическом языке и сразу же видеть графики численных решений. При необходимости более углубленно-

го изучения численные алгоритмы могут быть реализованы студентом самостоятельно на языке MVL, а затем можно сравнить полученные результаты с результатами работы встроенных алгоритмов.

Приложение 1 — это набор тестовых матриц, обратные для которых можно построить в явном виде. Тестовые матрицы используются в заданиях из главы 3.

В *приложении 2* приводятся таблицы кодов завершения программных реализаций численных методов визуальной среды моделирования MvStudium и рекомендации пользователям, что следует делать в том или ином случае аварийного завершения.

Приложение 3 — это описание компакт-диска, прилагаемого к книге. Все примеры выполнены в визуальной среде моделирования MvStudium, демонстрационная версия которой находится на компакт-диске. На этом же диске находятся также файлы всех рассматриваемых в книге примеров. Мы надеемся, что эти материалы понадобятся вам при чтении книги, обучении студентов и научной работе. Дополнительную информацию по пакету MvStudium вы можете найти на сайте **www.mvstudium.com**.

Учебное пособие соответствует содержанию федеральной дисциплины ОПД.Ф.10 "Системное моделирование" государственных образовательных стандартов по направлению подготовки бакалавров 220100 "Системный анализ и управление".

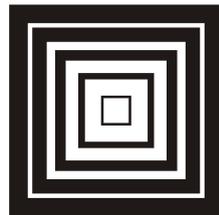
Мы признательны:

- заведующему кафедрой "Баллистика и аэродинамика" МГТУ им. Н. Э. Баумана д.т.н., профессору Лысенко Л. Н., прочитавшему рукопись первой главы и сделавшему ряд ценных замечаний;
- профессорам Евгеневу Г. В. и Ивановскому Р. И., рекомендовавшим книгу к изданию;
- коллективу издательства "БХВ-Петербург", всемерно помогавшему нам при подготовке рукописи к изданию.

Эта книга не могла состояться и без поддержки декана факультета технической кибернетики СПбГПУ профессора Черноруцкого И. Г.

И, конечно же, особая благодарность членам наших семей, еще не потерявшим надежды видеть авторов не только за компьютером днем и ночью.

ГЛАВА 1



Типовые модели: разработка и отладка

В этой главе на примере нескольких последовательно усложняющихся моделей демонстрируется технология разработки и отладки непрерывных, дискретных, гибридных и других моделей. Все обсуждаемые модели выполнены в визуальной среде объектно-ориентированного моделирования MvStadium. На прилагаемом к данной книге компакт-диске находятся демонстрационная версия этого пакета и соответствующие файлы для всех примеров. В связи с этим материал данной главы содержит некоторый минимальный объем инструкций по работе с пакетом моделирования MvStadium.

Создание и отладка модели непрерывной системы

В качестве примера рассмотрим процесс создания и отладки модели движения снаряда, выстреливаемого с начальной скоростью V_0 под углом θ_0 к горизонту, которая уже рассматривалась в качестве примера в [5, 6]. Здесь мы достаточно подробно изложим технологию создания и отладки набора последовательно усложняющихся моделей. Мы начнем с простейшей модели движения снаряда для схемы плоской аппроксимации поверхности Земли с постоянным по величине и направлению полем тяготения в безвоздушном пространстве. Далее мы будем вносить в эту модель доработки, учитывая различные факторы:

- сопротивление воздуха с постоянной плотностью;
- изменение плотности воздуха с высотой;

- изменение аэродинамических характеристик в зависимости от скорости;
- изменение силы тяжести с высотой;
- кривизна Земли.

Создадим новый проект (пакет с точки зрения UML) с именем **Снаряд**. Для этого в интегрированной среде MvStudium выполним команду главного меню **Проект | Новый** или нажмем кнопку  на инструментальной панели. В появившемся диалоге (рис. 1.1) вводим название папки и имя проекта, а также выбираем вид модели — в данном случае **Непрерывный элементарный объект**.

После нажатия кнопки **Создать** будет создана папка ...\\Модели\\Снаряд и в ней появится файл **Снаряд.mvb**, содержащий базу данных проекта. Для удобства мы будем каждую модель размещать в отдельной папке, с помощью команды **Проект | Сохранить как** вы можете в дальнейшем сохранять в той же папке различные модификации данного проекта (например, **Снаряд_1.mvb** и т. д.). В дальнейшем в этой же папке автоматически будут созданы еще два файла — **Снаряд.ini** и **Снаряд_em.ini**, в которых сохраняются соответственно установки проекта и установки визуальной выполняемой модели.

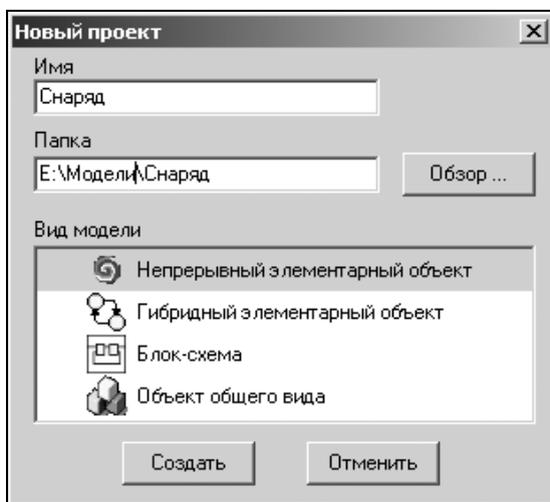


Рис. 1.1. Создание новой модели

На рис. 1.2 показано начальное состояние редактора моделей (главное окно интегрированной среды MvStudium). Для данного простейшего случая он

включает только одно максимизированное окно редактора класса `Model`, который, как показано в томе 2 данной книги, и содержит описание модели для вычислительного эксперимента.

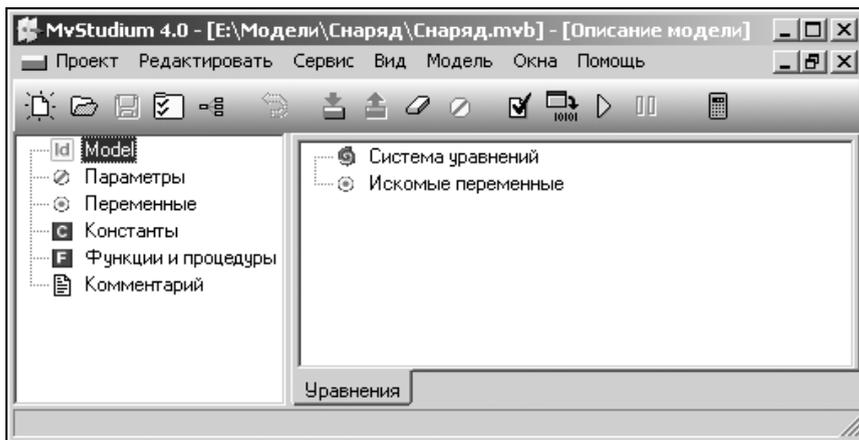


Рис. 1.2. Начальное состояние редактора моделей для модели непрерывной элементарной системы

В общем случае главное окно интегрированной среды включает в себя менеджер проекта в левой части окна и многооконную область редакторов классов (рис. 1.3). Соотношение между ними можно изменить с помощью перемещаемого разделителя (сплиттера) на правой границе менеджера проекта. Менеджер проекта является визуальным представлением элементов проекта (классов, констант, процедур и функций), а также импортируемых пакетов, в виде дерева. В новом проекте всегда присутствует класс `Model` и импортируется пакет стандартных классов `SysLib`. Вы можете скрыть менеджер проекта с помощью кнопки в его правом верхнем углу или команды меню **Вид | Менеджер проекта**. Вы можете отобразить менеджер проекта с помощью кнопки  на инструментальной панели или команды меню **Вид | Менеджер проекта**.

Теперь нужно задать переменные и систему уравнений, определяющую поведение непрерывной модели. Переменные можно определить предварительно, а можно просто использовать их в системе уравнений, и редактор уравнений, встретив неопределенную переменную, будет автоматически воспринимать ее как переменную типа `double` с нулевым начальным значением. Выберем второй путь.

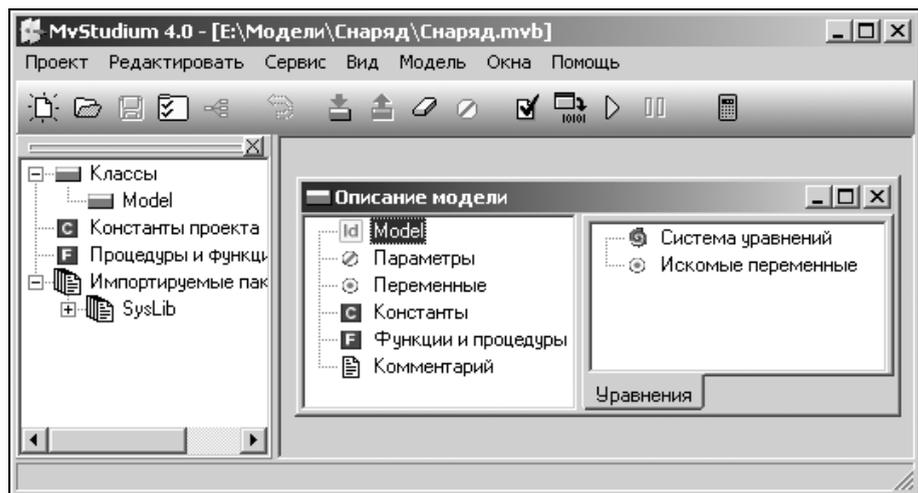


Рис. 1.3. Общий вид редактора моделей

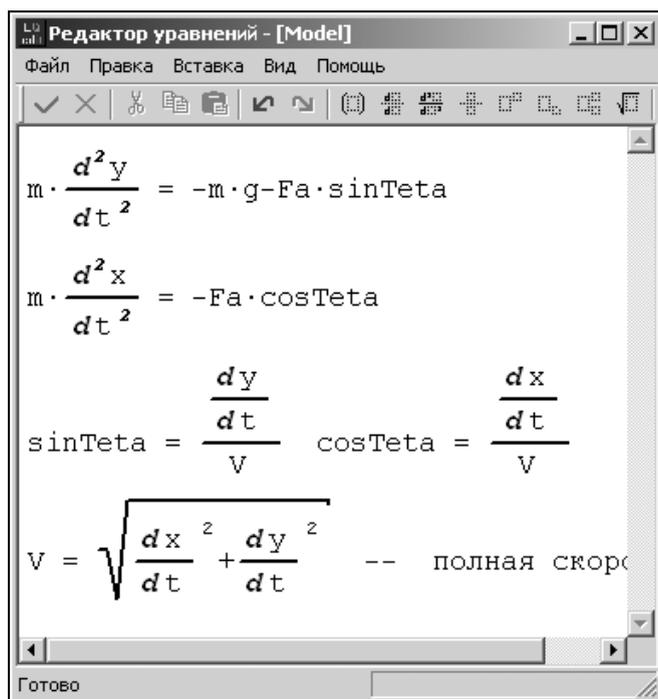


Рис. 1.4. Простейшая система уравнений движения снаряда

Для начала зададим простейшую систему уравнений, описывающую движение снаряда в постоянном по величине и направлению поле тяготения с нулевым значением (пока) силы аэродинамического сопротивления F_a (рис. 1.4). Уравнения можно пояснять комментариями.

Редактор уравнений попросит подтвердить необходимость определения новых переменных, при подтверждении в соответствующем диалоге вводим нужные начальные значения переменных и комментарии. После ввода системы уравнений мы получим соответствующую модель, показанную на рис. 1.5. Для модели элементарной непрерывной системы описание ее поведения содержит только одну вкладку — **Уравнения**. Обратите внимание на то, что для решения дифференциальных уравнений второго порядка необходимы начальные значения первых производных искомых переменных.

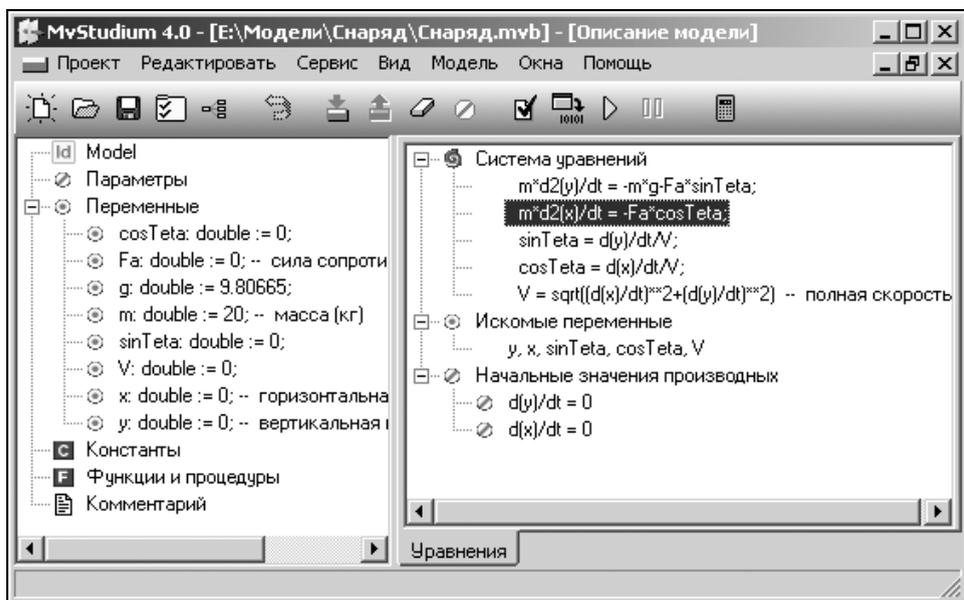


Рис. 1.5. Модель, соответствующая простейшей системе уравнений движения снаряда

Однако получившаяся модель лишена всякого смысла, т. к. при нулевых начальных значениях в ней отсутствует всякая динамика. Введем параметры модели, соответствующие начальной скорости V_0 и начальному углу наклона

траектории θ_0 , и зададим соответствующие им начальные значения первых производных (рис. 1.6).

Редактировать уравнения, начальные значения производных, переменные и параметры, а также комментарий к модели можно с помощью команды **Изменить...** (см. рис. 1.6) всплывающего меню, либо с помощью двойного щелчка мышью на изображении соответствующего узла визуального дерева в левой части редактора модели. Добавить или удалить переменные или параметры можно с помощью команд **Добавить...** или **Удалить** всплывающего меню (см. рис. 1.6).

Теперь целесообразно остановиться и провести первый вычислительный эксперимент с простейшей целью проверки правильности нашей модели.

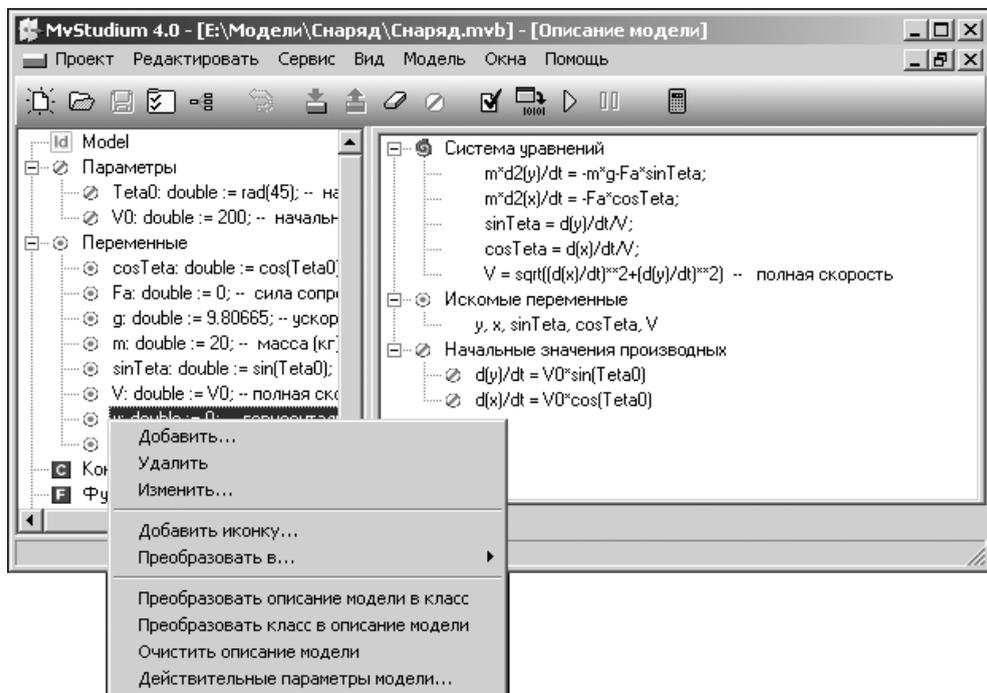


Рис. 1.6. Модель движения снаряда с ненулевыми начальными условиями

С помощью команды **Модель | Запустить** главного меню или кнопки  на инструментальной панели создадим и запустим визуальную модель. Пакет

MvStudium относится к категории "компилирующих": для всех элементов модели генерируется "код" на промежуточном объектно-ориентированном языке программирования, который затем компилируется в машинный код и связывается с исполняющей системой MvStudium с помощью штатного компилятора командной строки этого языка. В данной версии пакета в качестве промежуточного языка используется Borland Object Pascal (Delphi). Вся работа с промежуточным кодом проводится в локальной папке ...\\Tmp (в данном примере в ...:\\Модели\\Снаряд\\Tmp), в ней же формируется файл model.exe. Обычно это занимает несколько секунд. При выходе из интегрированной среды все содержимое этой папки автоматически уничтожается. Для того чтобы сохранить модель как независимую от среды разработки выполняемую программу, сначала создайте визуальную модель с помощью команды **Модель | Создать** или кнопки  на инструментальной панели, а затем выполните команду **Модель | Сохранить как** главного меню. Если папка проекта недоступна для записи (например, проект находится на компакт-диске), то все временные файлы и файл выполняемой модели формируются в рабочей папке Windows (например, C:\\Temp). После создания визуальная модель запускается параллельно с интегрированной средой. В любой момент вы можете прервать выполнение модели из интегрированной среды (например, в случае ее "зависания") с помощью команды **Модель | Стоп** или кнопки  на инструментальной панели.

На рис. 1.7 показано начальное состояние главного окна визуальной модели снаряда при первом запуске. Модель находится в точке останова. Для непрерывной модели по умолчанию открываются:

- окно переменных (включая используемые производные);
- окно временной диаграммы для первых четырех искомым переменных.

Не следует забывать, что исходная система уравнений (см. рис. 1.4) автоматически преобразуется к эквивалентной форме, пригодной для численного решения. Для данной модели эквивалентная система имеет следующий вид (перед двоеточием указана переменная, определяемая из данного уравнения):

$$y : d(y)/dt = y' ;$$

$$y' : d(y')/dt = y'' ;$$

$$x : d(x)/dt = x' ;$$

$$x' : d(x')/dt = x'' ;$$

$$y'' : m*y'' = -m*g - F_{ay} ;$$

$$x'' : m*x'' = -F_{ax} ;$$

Вы можете посмотреть численно решаемую систему уравнений, выполнив команду **Показать представление уравнений для численного решения** всплывающего меню на вкладке **Уравнения**.

В нижней части окна указывается информация о типе решаемой в настоящий момент глобальной системы уравнений и ее составе. В данном случае это система дифференциально-алгебраических уравнений (DAE), в которой 4 дифференциальных уравнения и 2 алгебраических.

ЗАМЕЧАНИЕ

Когда мы станем говорить об элементах описания моделей, то будем вынуждены иногда отступать от обычных обозначений из-за ограниченных возможностей текстового представления. Например, вместо общепринятого обозначения производной по времени \dot{x} мы будем использовать обозначение x' . Знак операции умножения всегда будет указан явно, и F_{ax} означает идентификатор переменной, а не произведение $F \cdot a \cdot x$.

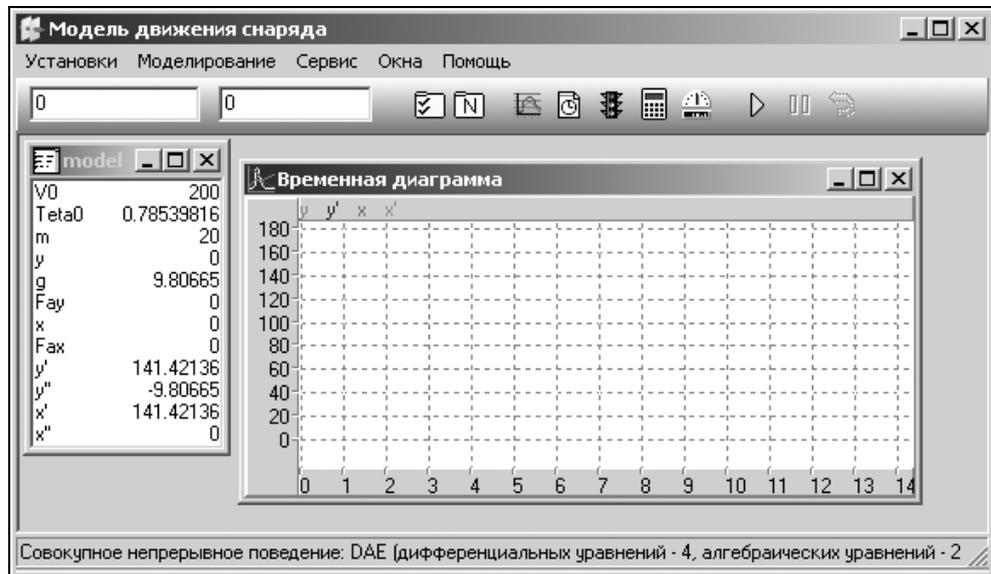


Рис. 1.7. Начальное состояние главного окна визуальной модели снаряда

В заголовок главного окна визуальной модели выводится первая строка комментария для класса Model (если комментарий отсутствует, то выводится стандартный заголовок **Визуальная модель MvStadium 4**). В левой части инструментальной панели показаны непрерывная и дискретная составляющие модельного времени.

Значения переменных в окне переменных могут отличаться от начальных значений, указанных в описании модели. Дело в том, что в точках дискретных событий (для непрерывной модели это только точки останова) решается алгебраическая составляющая полной системы уравнений (совокупность алгебраических уравнений и формул), и в окнах показываются согласованные значения переменных. Если переменная является искомой, то ее начальное значение является всего лишь начальным приближением для численного метода.

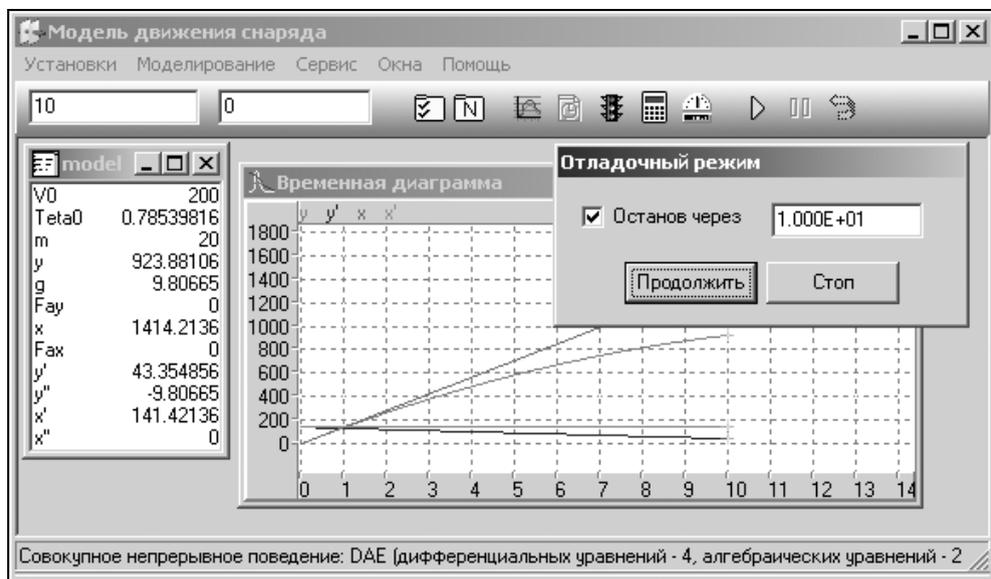


Рис. 1.8. Отладочный останов модели

Запустим продвижение модельного времени. Для этого нужно выполнить команду **Моделирование | Пуск** главного меню или нажать кнопку  на инструментальной панели. Система уравнений модели начнет численно решаться. По умолчанию в опциях численных методов (изменяются с помощью

команды **Установки** | **Численные методы** или кнопки  выбраны автоматические методы и погрешности 10^{-6} . Эти установки подходят для большинства не очень сложных моделей. Вы увидите, как начнут изменяться значение непрерывной составляющей модельного времени и значения в окне переменных, а также строиться графики на временной диаграмме. По умолчанию в опциях модели (изменяются с помощью команды **Установки** | **Модель** или кнопки  на инструментальной панели) принято соотношение модельного и реального времени "так быстро, как возможно". Для простых уравнений это может быть очень быстро, так что вы ничего не успеете заметить. Поэтому при первом запуске визуальной модели автоматически устанавливается отладочный режим с остановом через 10 единиц непрерывной составляющей модельного времени (см. рис. 1.8).

В нашем случае будет полезнее использовать фазовую диаграмму и отладочный останов по условию. Отменим отладочный останов по времени и остановим эксперимент. В окне диаграммы выполним команду **Настройка** всплывающего меню. В появившемся диалоге настройки (рис. 1.9) щелкнем мышью в столбце **X** строки переменной x . Появившийся "плюс" означает, что именно эта переменная будет откладываться по оси абсцисс графика, а не модельное время. Таким образом, мы получаем фазовую диаграмму, на которой изображаются графики зависимостей переменных от выбранной переменной — в данном случае от горизонтальной координаты x (если "плюс" убрать, диаграмма вновь станет временной).

С помощью кнопки  на инструментальной панели или команды **Сервис** | **Условия останова** главного меню откроем диалог **Условия останова**, выберем вкладку **По предикату**, с помощью команды **Добавить** всплывающего меню перейдем в диалог редактирования логического предиката и зададим отношение, определяющее момент падения снаряда (рис. 1.10). В этом отношении задано $y \leq 0$, т. к. определение момента выполнения точного условия $y=0$ вызывает большие трудности.

С помощью кнопки  на инструментальной панели или команды **Моделирование** | **Рестарт** главного меню вернем модель в начальное состояние и снова запустим. Теперь сработает отладочный останов в точке падения, и мы получим графики, показанные на рис. 1.11 (график с утолщенной линией соответствует собственно траектории движения снаряда $y(x)$). Характер изменения переменных вполне правдоподобен. Запомним горизонтальную координату точки падения — 4079 м.

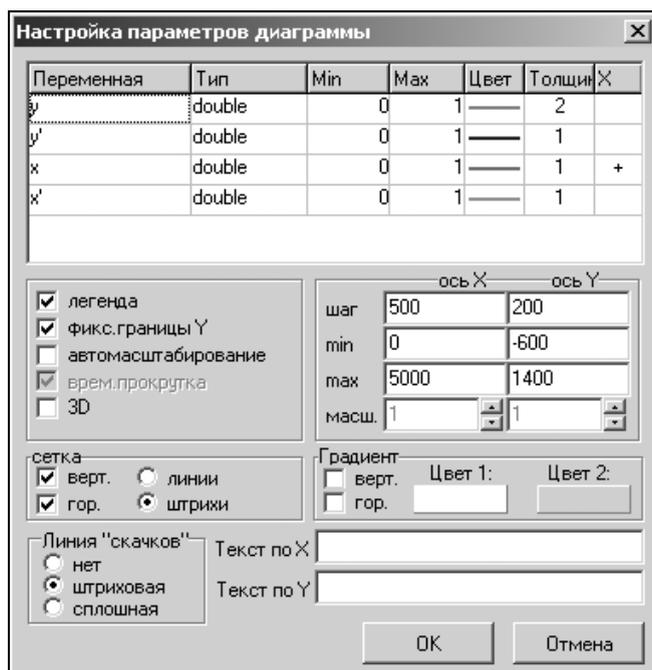


Рис. 1.9. Диалог настройки диаграммы

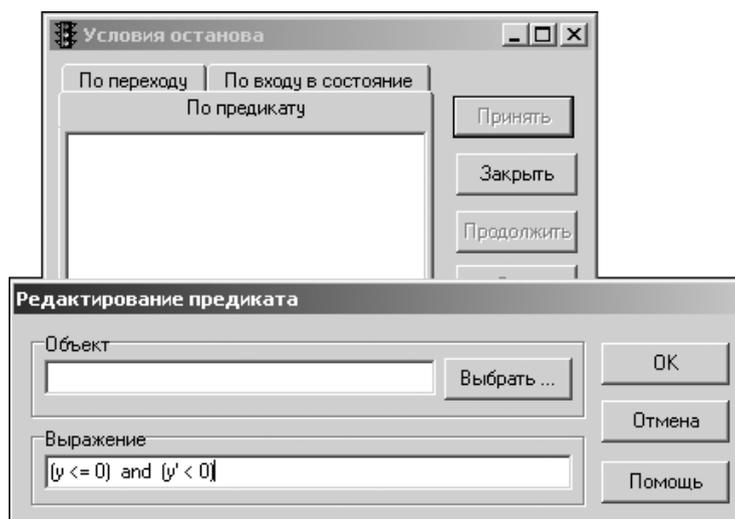


Рис. 1.10. Задание отладочного останова в момент падения снаряда

При отладке динамических моделей часто бывает полезно проверить выполнение различных законов сохранения. В нашем случае сумма кинетической и потенциальной энергии должна быть величиной постоянной при $F_a = 0$ или уменьшаться при $F_a \neq 0$.

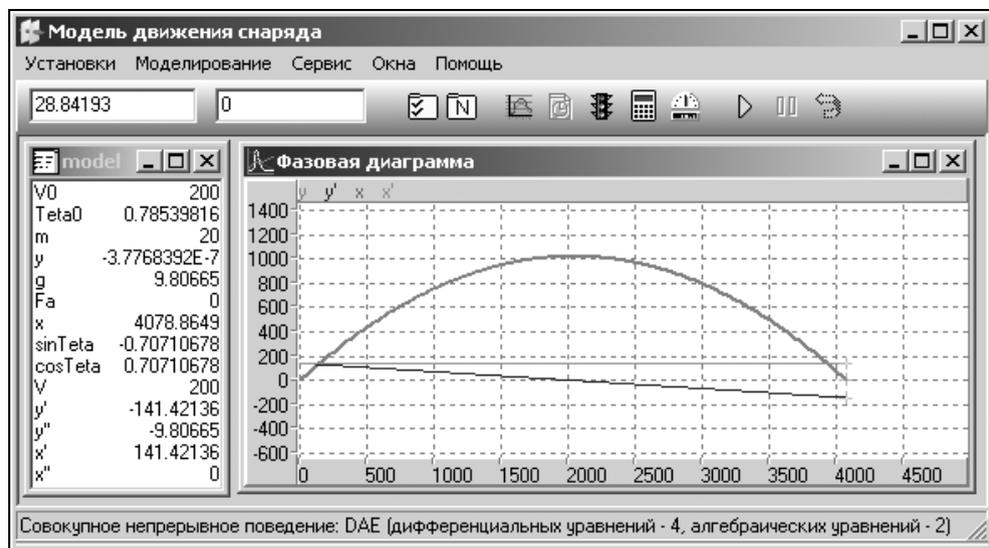


Рис. 1.11. Состояние модели снаряда в момент падения

Для таких целей очень удобно использовать виртуальные переменные. Виртуальной переменной не существует в описании модели, она определяется в визуальной модели вместе с формулой, по которой вычисляется ее значение. Виртуальные переменные можно использовать для построения графиков, анимации, условий останова и т. п. наравне с обычными переменными модели. С помощью команды **Создать виртуальную переменную** всплывающего меню в окне переменных создадим виртуальные переменные $W = \frac{m \cdot V^2}{2}$ (рис. 1.12), $Q = m \cdot g \cdot y$ и $E = W + Q$, соответствующие кинетической, потенциальной и полной энергии снаряда.

С помощью кнопки  на инструментальной панели или команды **Окна | Новая диаграмма** главного меню создадим новое окно диаграммы (по умолчанию это будет временная диаграмма, т. е. по оси абсцисс будут откла-

дываться значения модельного времени). Методом drag-and-drop перетащим переменные w , Q и E из окна переменных на окно диаграммы. Запустим модель и получим график, показанный на рис. 1.13. График вполне соответствует нашим представлениям о моделируемом физическом процессе.

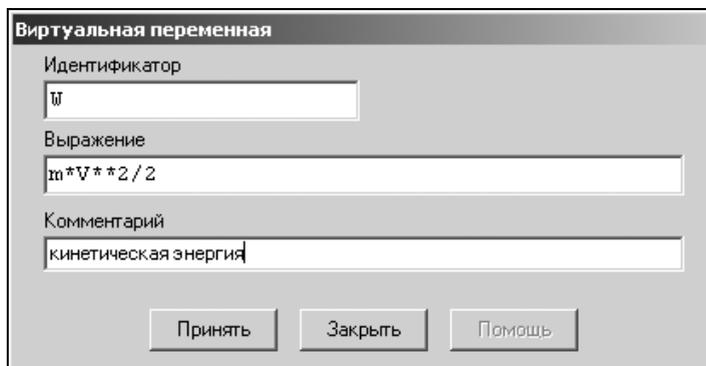


Рис. 1.12. Определение виртуальной переменной

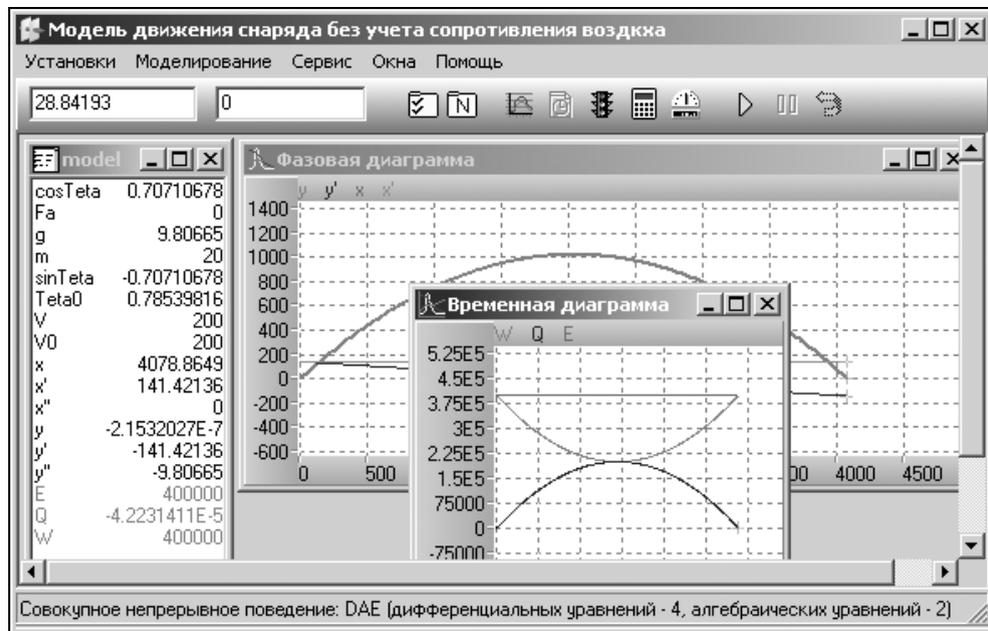


Рис. 1.13. Зависимость энергии от времени

Остается проверить правильность учета сопротивления воздуха. Для простейшей проверки вернем модель в начальное состояние и зададим ненулевое значение переменной F_a (F_a). Активный вычислительный эксперимент предполагает возможность интерактивного вмешательства. Вы можете в любой точке эксперимента изменить значение любой переменной. Для этого нужно выполнить команду **Изменить** всплывающего меню или дважды щелкнуть мышью на нужной строке в окне переменных и затем ввести новое значение в появившемся диалоге (это значение может быть выражением, содержащим другие переменные). После этого вычисляются новые значения зависимых переменных. В случае, если вы изменяете значение параметра, то сначала заново производится вычисление начальных значений переменных, а затем вычисление согласованных значений. Зададим, например, $F_a = 100$. Результаты вычислений показаны на рис. 1.14 и выглядят вполне правдоподобно. Теперь можно перейти к усложнению модели (состояние модели для этой точки разработки сохранено в файле `Снаряд_1.mvb`).

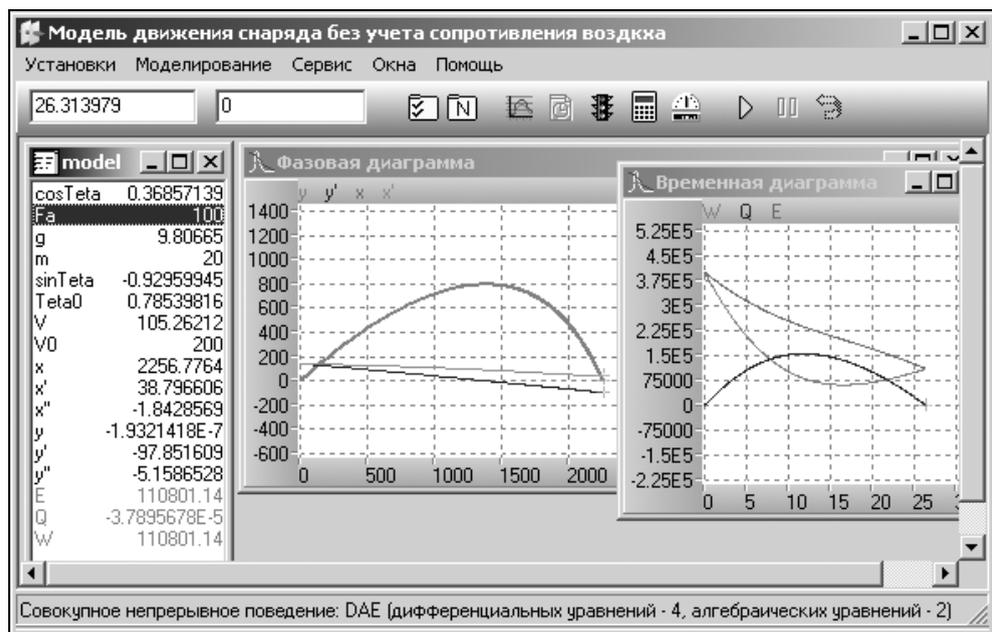


Рис. 1.14. Движение снаряда при фиксированной силе сопротивления воздуха

Попробуем учесть сопротивление воздуха. Используем зависимость [7]

$$F_a = C_X \cdot S_M \cdot \frac{\rho \cdot V^2}{2}.$$

Здесь S_M — это геометрическая характеристика движущегося объекта, называемая площадью "миделевого сечения". Эту величину сделаем параметром нашей модели и положим $S_M = 0.04 \text{ м}^2$.

C_X — это безразмерный коэффициент силы аэродинамического лобового сопротивления, зависящий от геометрической формы движущегося объекта (строго говоря, здесь мы используем коэффициент сопротивления C_{X_a} , заданный в скоростной системе координат, будем полагать $C_X = C_{X_a}$). Этот коэффициент не является постоянным и зависит от режима обтекания, характеризующегося так называемым "числом Маха" (отношение скорости движения к скорости звука). Поэтому мы введем соответствующую переменную и присвоим ей значение 0.29.

ρ — это плотность воздуха. Пока мы пренебрежем ее зависимостью от высоты и введем соответствующую переменную с начальным значением $\rho = 1.225 \text{ кг/м}^3$ (плотность воздуха на уровне моря при температуре $15 \text{ }^\circ\text{C}$).

Добавляем в систему уравнений формулу для F_a и получаем дальность точки падения 2063 м. Это почти в два раза меньше дальности, полученной без учета сопротивления воздуха, значит, предыдущая модель была совсем неадекватной. Состояние модели в данной точке разработки сохранено в файле `Снаряд_2.mvb`.

Дальнейшее совершенствование нашей модели связано с учетом изменений параметров воздуха в зависимости от высоты. Без этого невозможно также учесть изменение коэффициента C_X , поскольку скорость звука также меняется с высотой.

Используются различные модели атмосферы — "стандартные атмосферы". Все они предусматривают выделение диапазонов высоты ("слоев"), в которых действуют различные зависимости. Для наших примеров достаточно использовать первые три слоя [7]:

- 0 — 11 000 м;
- 11 000 — 25 000 м;
- 25 000 — 46 000 м.

Для первого слоя:

$$T = T_0 + k_1 \times H,$$

$$\rho = \rho_0 \cdot e^{-\left(1 + \frac{G_0}{k_1 \times R_{уд}}\right) \cdot \ln \frac{T}{T_0}},$$

где

$$T_0 = 288.15 \text{ }^\circ\text{K}, \quad \rho_0 = 1.225 \text{ кг/м}^3, \quad R_{уд} = 287.039 \text{ м}^2/\text{град} \cdot \text{с}, \quad G_0 = 9.80665 \text{ м/с}^2,$$

$$k_1 = -651\,122 \cdot 10^{-8} \text{ град/м}.$$

Для второго слоя:

$$T = T_1,$$

$$\rho = \rho_1 \cdot e^{-\frac{G_0}{R_{уд}} T (H - H_1)}.$$

Для третьего слоя:

$$T = T_2 + k_3 \times (H - H_2),$$

$$\rho = \rho_2 \cdot e^{-\left(1 + \frac{G_0}{k_3 \times R_{уд}}\right) \cdot \ln \frac{T}{T_2}},$$

$$k_3 = 276\,098 \cdot 10^{-8}.$$

Во всех этих зависимостях под величиной H понимается:

□ высота над уровнем моря $H = h$, если не учитывается изменение ускорения силы тяжести с высотой;

□ так называемая "геопотенциальная высота" $H = \frac{R_3 \cdot h}{R_3 + h}$, где

$R_3 = 6\,371\,210 \text{ м}$ — средний радиус Земли, если такое изменение учитывается.

Эти зависимости можно отразить в модели с помощью алгоритмических функций. В данном случае будет вполне достаточно использовать внутренний алгоритмический язык пакета MvStudium. Например, зависимость плотности от высоты может быть задана так:

```
function Rho_H(in H: double) return double
  T: double := T_H(H); -- температура
```

```
begin
  if H<=11000 then
    -- слой 1
    return Rho0*exp(-(1+G0/k1*Rуд)*ln(T/T0));
  elseif H<=25000 then
    -- слой 2
    return Rho1*exp(-G0*(H-H1)/Руд*T);
  elseif H<=46000 then
    -- слой 3
    return Rho2*exp(-(1+G0/k3*Rуд)*ln(T/T2));
  else
    -- за границей определения модели
    return NaN;
  end if;
end Rho_H;
```

Обратите внимание, что для высот ниже уровня моря неявно распространяется действие зависимостей для слоя 1, а при выходе за верхнюю границу слоя 3 возвращается значение NaN ("Not A Number" — предопределенная функция, возвращающая специальное вещественное значение). Существует и второе специальное вещественное значение — Infinity, соответствующее очень большой величине, выходящей за границы разрядной сетки компьютера. В окнах переменных выполняемой модели такие значения переменных станут обозначаться соответствующими идентификаторами, а на графиках не будет изображаться ничего.

Заметим, однако, что зависимости параметров атмосферы от высоты имеют изломы, т. е. разрывы производной. Скорее всего, в данном случае это не вызовет каких-либо проблем у численных методов, однако быть уверенными в этом мы не можем. Поэтому подобные зависимости полезно реализовать как гибридный автомат. Таким образом, при более детальном рассмотрении даже чисто непрерывных, на первый взгляд, моделей в них часто появляются зависимости с разрывами, которые приводят к необходимости создания гибридных моделей.

Далее, ясно, что модель атмосферы никак особенно не связана с нашей моделью снаряда и может пригодиться и для других моделей, например, модели ракеты. Поэтому имеет смысл создать отдельный пакет **Земля**, содержащий специфические константы, функции и классы, и импортировать его в специальных моделях.

Для этого снова создадим новый проект в папке ...\\Модели\\Земля, выбрав вид модели **Объект общего вида**. В менеджере проекта в левой части окна добавим в проект с помощью команды **Добавить...** всплывающего меню нужные константы и функции (в частности, радиус Земли и функцию вычисления геопотенциальной высоты). Теперь добавим новый класс — **СтандартнаяАтмосфера** (рис. 1.15).

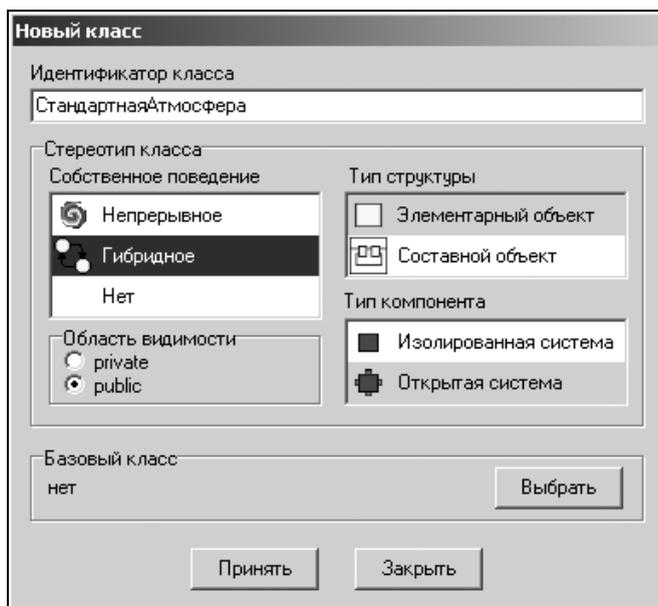


Рис. 1.15. Добавление класса **СтандартнаяАтмосфера**

При создании нового класса указывается начальный стереотип этого класса, включающий тип собственного поведения (в нашем случае **Гибридное**), тип структуры (**Элементарный объект**), тип компонента (в нашем случае это **Открытая система**, поскольку в противном случае этим компонентом не смогут пользоваться другие модели) и область видимости — только в данном пакете (**private**) или в других пакетах, импортирующих данный (**public**). В дальнейшем стереотип класса можно изменить с помощью команды **Преобразовать в ...** всплывающего меню.

Перейдем в окно редактора класса **СтандартнаяАтмосфера**. Оно содержит визуальное дерево атрибутов и методов в левой части и редактор поведения в правой части окна. С помощью команды **Добавить иконку** всплывающе-

го меню визуального дерева присвоим классу подходящий значок, который будет изображаться на всех экземплярах этого класса. С помощью команды **Добавить...** всплывающего меню добавим один вход (высота), три выхода (температура, плотность и скорость звука) и необходимые константы (рис. 1.16).

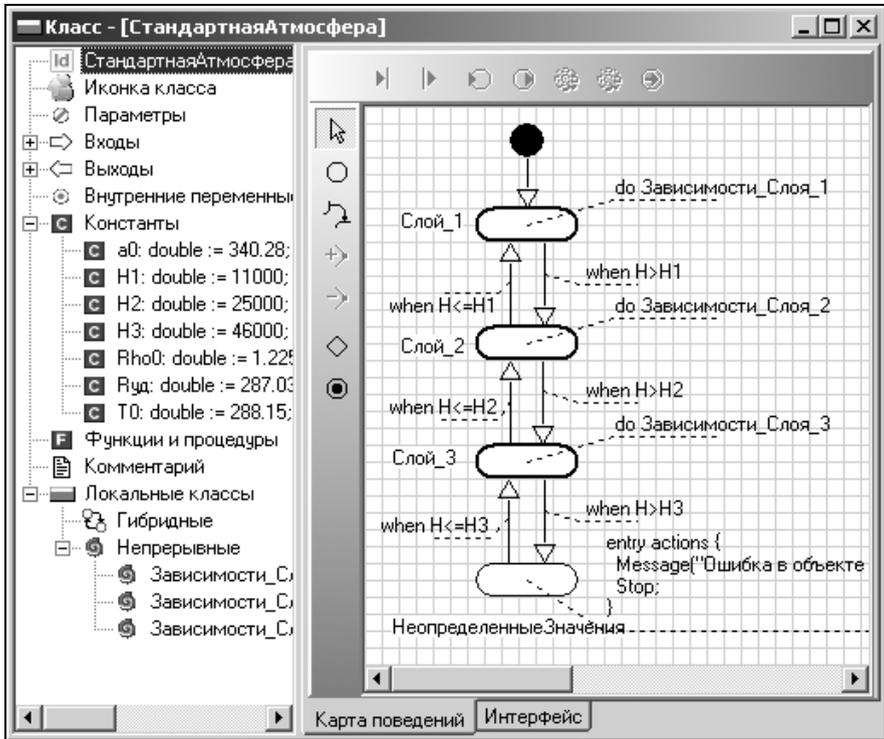


Рис. 1.16. Окно редактора класса `СтандартнаяАтмосфера`

Перейдем теперь на вкладку **Карта поведений**. В редакторе карты поведений имеются две инструментальные панели. Панель слева предназначена для добавления новых элементов: состояний, переходов и опорных точек на переходе. Верхняя панель служит для редактирования свойств выделенного элемента (условий и действий перехода, входных/выходных действий и локальной деятельности в состоянии). Карта поведений нового гибридного класса при первом запуске содержит начальное состояние, обычное состояние `s1` и безусловный переход между ними. Состояние `s1` переименуем в

Слой_1 и расположим состояния вертикально, переместив с помощью указателя мыши. Положение всех элементов выравнивается по клеткам сетки. Нам будут нужны также состояния Слой_2 и Слой_3, а также состояние НеопределенныеЗначения, соответствующее выходу за границы определения модели атмосферы. Для добавления нового обычного состояния нажмем кнопку  на левой инструментальной панели и нажмем левую кнопку мыши в точке, соответствующей верхнему левому углу нового состояния. Для добавления нового перехода нажмем кнопку  на левой инструментальной панели, нажмем кнопку мыши на изображении исходного состояния перехода, не отпуская ее, переместим курсор мыши на изображение конечного состояния перехода и отпустим кнопку. Переход можно также добавлять "по частям", отпустив кнопку на свободном поле, а затем использовать конечную точку как исходное состояние.

Введем теперь условия срабатывания переходов. Для этого выделим нужный переход и нажмем кнопку  на верхней инструментальной панели. В появившемся диалоге введем условие срабатывания (рис. 1.17).

Условие срабатывания перехода

Запускающее событие Логическое выражение

Идентификатор сигнала

when Идентификатор сигнала

Вещественное выражение

after

нет

Охраняющее условие Логическое выражение

else

Рис. 1.17. Условие срабатывания перехода

Далее выделяем состояние НеопределенныеЗначения и с помощью кнопки  верхней инструментальной панели вводим мгновенную последовательность входных действий: при попадании в это состояние выводится сообщение об ошибке (выход за область определения), и модель останавливается.

Теперь нужно задать зависимости температуры, плотности воздуха и скорости звука от высоты для трех слоев. Для этого нужно создать три локальных непрерывных класса, экземпляры которых будут приписаны в качестве деятельности соответствующим состояниям. Выделим, например, состояние Слой_1 и выполним команду **Создать непрерывную деятельность** всплывающего меню. В классе СтандартнаяАтмосфера появится локальный непрерывный класс с именем Деятельность_в_Слой_1 по умолчанию. Переименуем его в Зависимости_Слоя_1. Локальный класс отличается тем, что в нем видимы все элементы описания класса, в который он вложен. В данном случае в определении класса Зависимости_Слоя_1 видимы все константы и переменные класса СтандартнаяАтмосфера. Кроме того, в определении локального класса могут быть введены собственные константы, переменные и функции. Экземпляры локального класса могут использоваться только как деятельности в состояниях карты поведений охватывающего класса. В данном случае введем собственную константу k1, необходимую только для задания системы уравнений класса Зависимости_Слоя_1 (рис. 1.18).

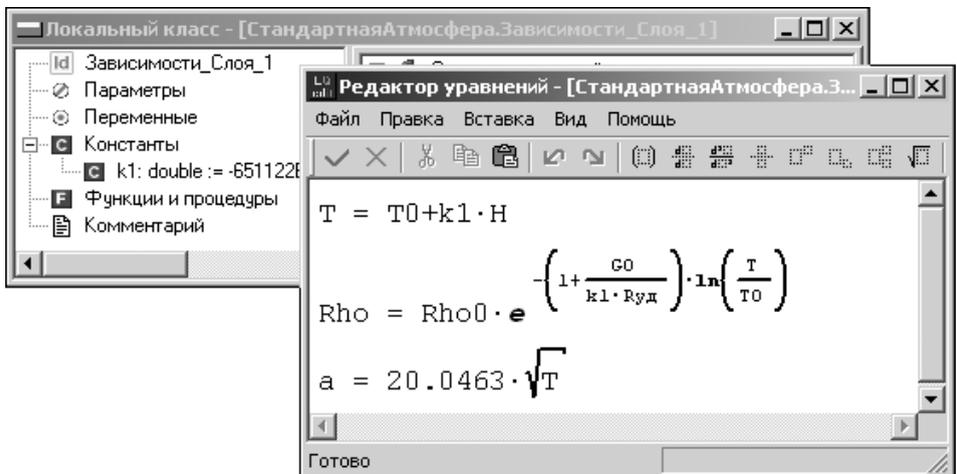


Рис. 1.18. Локальный класс Зависимости_Слоя_1

Аналогично введем определения локальных классов Зависимости_Слоя_2 и Зависимости_Слоя_3, экземпляры которых соответственно будут являться деятельностью в состояниях Слоя_2 и Слоя_3.

Теперь класс СтандартнаяАтмосфера определен, и нужно проверить его работоспособность. Откроем вкладку **Структура** класса Model, схватим мышью узел класса СтандартнаяАтмосфера в дереве классов менеджера проекта, перетащим его в окно редактора класса Model и бросим на вкладке **Структура**. В локальной структуре класса Model появится объект — экземпляр класса СтандартнаяАтмосфера, который назовем Атмосфера (рис. 1.19). Однако этого недостаточно. Если мы просто запустим модель в таком виде, то в ней не будет ничего происходить, т. к. модель атмосферы явно не зависит от времени. Для испытаний необходимо изменять значение высоты. Поэтому в дереве **Импортируемые пакеты** менеджера проекта развернем узел SysLib, выберем стандартный класс CRampGenerator (генератор равномерно нарастающего сигнала) и методом drag-and-drop бросим его на вкладке **Структура** класса Model. Появившийся экземпляр этого класса назовем RG (см. рис. 1.19).

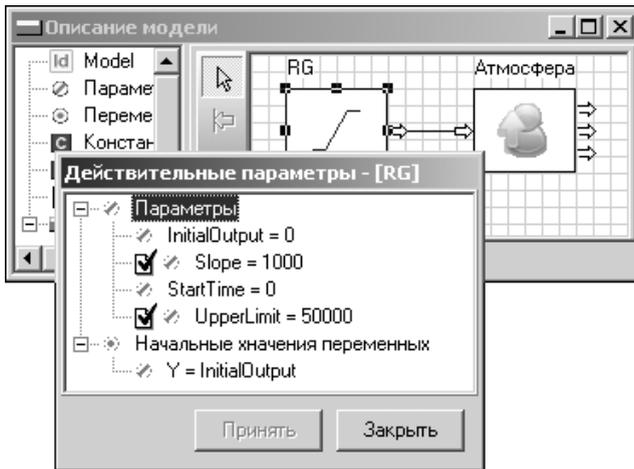


Рис. 1.19. Испытание модели атмосферы

С помощью команды **Действительные параметры** всплывающего меню изменим значения параметров объекта RG: нарастание высоты со скоростью 1000 м/с и верхний предел 50 000 м (см. рис. 1.19). Соединим его выход RG.Y с входом Атмосфера.Н. Для этого нажмем кнопку  на левой инструментальной панели редактора структуры. Редактор переходит в режим добавле-