



Начинаем работать в программном комплексе «Универсальный механизм»

Начинаем работать

2010

Это руководство поможет вам освоить работу с программным комплексом «Универсальный механизм». Руководство посвящено созданию и исследованию ряда простых механических систем. Уроки даны по принципу «от простого к сложному». Предполагается, что вы будете изучать уроки последовательно, как они представлены в данном руководстве. Некоторые разделы руководства посвящены изучению особенностей отдельных модулей программы, о чем будет сказано в начале таких разделов.

Начинаем работать в программном комплексе «Универсальный механизм»

Это руководство поможет вам освоить работу с программным комплексом «Универсальный механизм». Руководство основано на создании и исследовании ряда простых механических систем. Уроки даны по принципу «от простого к сложному». Предполагается, что вы будете изучать уроки последовательно, как они представлены в данном руководстве. Некоторые разделы руководства посвящены изучению особенностей отдельных модулей программы, о чем будет сказано в начале таких разделов.

Моделирование в ПК «Универсальный механизм» таких механических систем, как железнодорожные экипажи, автомобили, имеет, безусловно, свои особенности, однако основные принципы описания моделей остаются универсальными. Эти принципы и изложены в данном руководстве.

Последняя редакция данного руководства доступна в Интернете по адресу: http://www.umlab.ru/download/60/rus/g_s_um.pdf.

Контактная информация

Самая последняя версия программы и руководства пользователя доступны по адресу http://www.umlab.ru/download_rus.htm.

Сообщения об ошибках в программе и документации, ваши замечания и предложения шлите по адресу um@umlab.ru.

Почтовый адрес:

Россия, 241035, Брянск,
бульв. 50-летия Октября, 7
Брянский государственный технический университет,
Лаборатория «Вычислительная механика»
проф. Погорелову Дмитрию Юрьевичу

Телефон/факс: +7 4832 568637.

Оглавление

НАЧИНАЕМ РАБОТАТЬ В ПРОГРАММНОМ КОМПЛЕКСЕ «УНИВЕРСАЛЬНЫЙ МЕХАНИЗМ»	2
1. МАТЕМАТИЧЕСКИЙ МАЯТНИК	5
1.1. Что мы изучим.....	5
1.2. Создание схемы объекта.....	6
1.3. Создание модели	7
1.3.1. Запуск UM Input и создание нового объекта моделирования.....	7
1.3.2. Основные элементы интерфейса пользователя	7
1.3.3. Создание графических образов	9
1.3.3.1. Образ сцены	9
1.3.3.2. Образ маятника	15
1.3.4. Создание тел	16
1.3.5. Создание шарниров	17
1.3.6. Сохранение введенных данных.....	18
1.3.7. Подготовка к моделированию: синтез уравнений движения.....	19
1.3.7.1. Численно-итерационный метод	20
1.3.7.2. Символьный метод	21
1.3.8. Переход к моделированию движения.....	22
1.4. Моделирование движения маятника.....	23
1.5. Многозвенный маятник	32
1.5.1. Доработка модели однозвенного маятника	32
1.5.2. Использование метода подсистем.....	34
1.5.2.1. Структура составного объекта	34
1.5.2.2. Подготовка модели подсистемы	36
1.5.2.1. Описание составного объекта	39
1.5.2.1.1. Создание графического образа сцены	39
1.5.2.1.2. Добавление подсистем	41
1.5.2.1.3. Создание шарниров	43
1.5.2.2. Моделирование движения	45
1.5.2.3. Использование внешних связей	50
2. ГРУЗ НА ПРУЖИНЕ	58
2.1. Что мы изучим.....	58
2.2. Описание модели	58
2.3. Создание модели	59
2.3.1. Запуск UM Input и создание нового объекта моделирования.....	59
2.3.2. Создание графических образов	59
2.3.3. Создание тел	64
2.3.4. Создание шарниров	66
2.3.5. Описание сил	69
2.3.6. Визуализация пружины и демпфера	70
2.3.7. Дополнительные параметры	72

2.3.8.	Подготовка к моделированию.....	73
2.4.	Моделирование движения.....	74
2.4.1.	Свободные колебания.....	74
2.4.2.	Статистический анализ.....	80
2.4.3.	Линейный анализ.....	81
2.4.4.	Вынужденные колебания	83
3.	ДАЛЬНЕЙШЕЕ ИЗУЧЕНИЕ ПРОГРАММЫ	85

1. Математический маятник¹

1.1. Что мы изучим

В этом уроке мы научимся создавать новую модель, добавлять в нее тела и шарниры, проводить синтез и компиляцию уравнений движения, запускать моделирование динамики моделируемого объекта, а также строить графики различных величин. В целом урок призван дать вам общее представление о последовательности моделирования механических систем в ПК «Универсальный механизм».

В конце данного раздела мы создадим модель маятника, которая будет включать одно твердое тело – собственно маятник, один вращательный шарнир, и, кроме того, графический образ неподвижных частей – стойки. После описания модели мы пройдем этапы синтеза и компиляции уравнений движения, а затем перейдем непосредственно к моделированию движения нашей модели. Готовую модель вы можете найти в каталоге `{um_root}\samples\tutorial\rus\pendulum` или скачать по адресу:

<http://www.umlab.ru/download/60/rus/pendulum.zip>

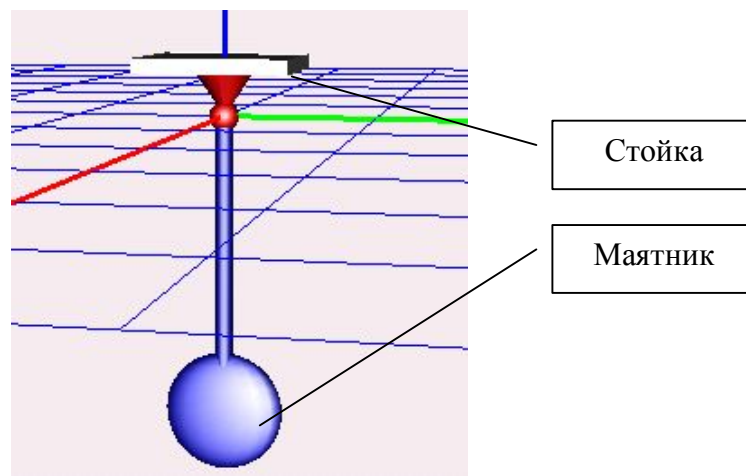
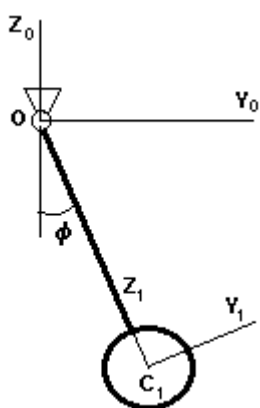


Рис. 1.1. Модель маятника

Далее мы создадим модель многозвенного трехзвенного маятника двумя способами: путем доработки модели однозвенного маятника и с использованием метода подсистем. При этом мы получим навыки создания составных моделей с использованием метода подсистем и рассмотрим особенности работы с системами с замкнутыми кинематическими цепями.

¹ **Математический маятник** — механическая система, состоящая из материальной точки, подвешенной на невесомой нерастяжимой нити или на невесомом стержне в поле тяжести.

1.2. Создание схемы объекта



До начала моделирования маятника с использованием УМ мы рекомендуем нарисовать его схему на бумаге, наподобие той, что представлена слева. Как вы видите, мы нарисовали простой маятник и выбрали две системы координат (СК): базовую (инерциальную) систему координат $Ox_0y_0z_0$ (СК0) и систему координат, связанную с телом (СК1) с началом в центре масс тела, оси совпадают с главными осями инерции (в данном случае они являются осями симметрии маятника). Базовая система координат всегда присутствует в любом объекте и, как правило, связывается с Землей. Относительно этой системы координат описывается положение элементов объекта.

1.3. Создание модели

1.3.1. Запуск UM Input и создание нового объекта моделирования

Для запуска программы описания моделей UM Input:

1. Выберите кнопку **Пуск**.
2. Выберите **Программы/Universal Mechanism 5.0/UM Input**.

Для создания новой модели:

3. Выберите пункт меню **Файл/Новый объект**.

Появится окно конструктора для описания новой модели, см. рис. 1.2.

1.3.2. Основные элементы интерфейса пользователя

До того, как продолжить описание модели, ознакомимся с основными элементами окна конструктора, см. рис. 1.2.

Список элементов модели в левой верхней части конструктора используется для быстрого доступа к отдельному элементу по его имени.

Анимационное окно в центре конструктора отображает визуально конструируемый объект. В окне изображена базовая система координат, для осей используется цветовая идентификация RGB: **красный (Red)** – ось X, **зеленый (Green)** – ось Y, **синий (Blue)** – ось Z. Повороты камеры, приближение/удаление объекта, изменение типа графики (каркасная/объемная) выполняются с помощью кнопок в верхней части окна. Через контекстное меню можно настроить перспективу, задать опорную сетку и так далее.

Инспектор в правой части является основным инструментом создания элементов и в каждый момент времени отображает параметры активного элемента.

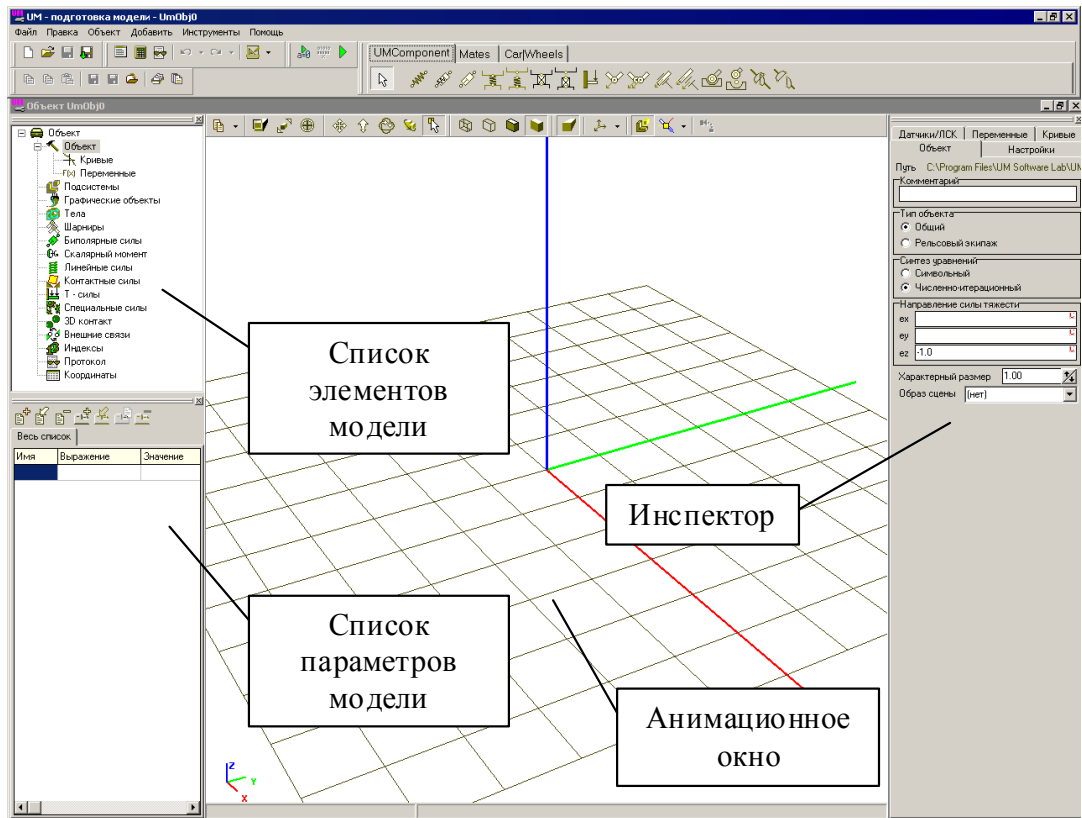


Рис. 1.2. Окно конструктора


1.3.3. Создание графических образов

Мы рекомендуем начинать описание любой механической системы с создания набора *Графических объектов (ГО)* – образов элементов конструкции (тел, силовых элементов, окружающей среды).

1.3.3.1. Образ сцены

Создание нового графического объекта - образа сцены

Сценой назовем рисунок, соответствующий неподвижным элементам объекта. Для сцены (создание которой вообще-то не обязательно) создается отдельный ГО. В случае маятника сцене будет принадлежать изображение неподвижного шарнира (стойки), к которому прикрепляется маятник. Для создания соответствующего образа выполните следующие действия.

1. Перейдите на ветку **Графические объекты** в списке элементов модели.
2. Для добавления нового графического объекта щелкните на кнопке  в верхней части инспектора, см. рис. 1.3.

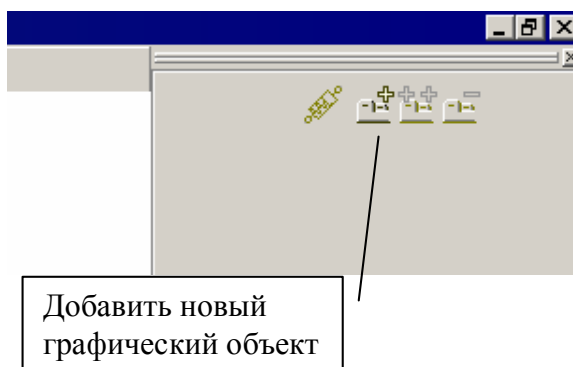


Рис. 1.3. Добавление нового элемента

Замечание. Таким же образом происходит добавление новых элементов других типов.

Переименование графического объекта

По умолчанию, вновь созданному графическому объекту присваивается стандартное имя **GO1**. Переименуем графический объект в **Стойка**.

3. Перейдите в поле имени графического объекта и введите **Стойка** и нажмите *Enter*, см. рис. 1.4.

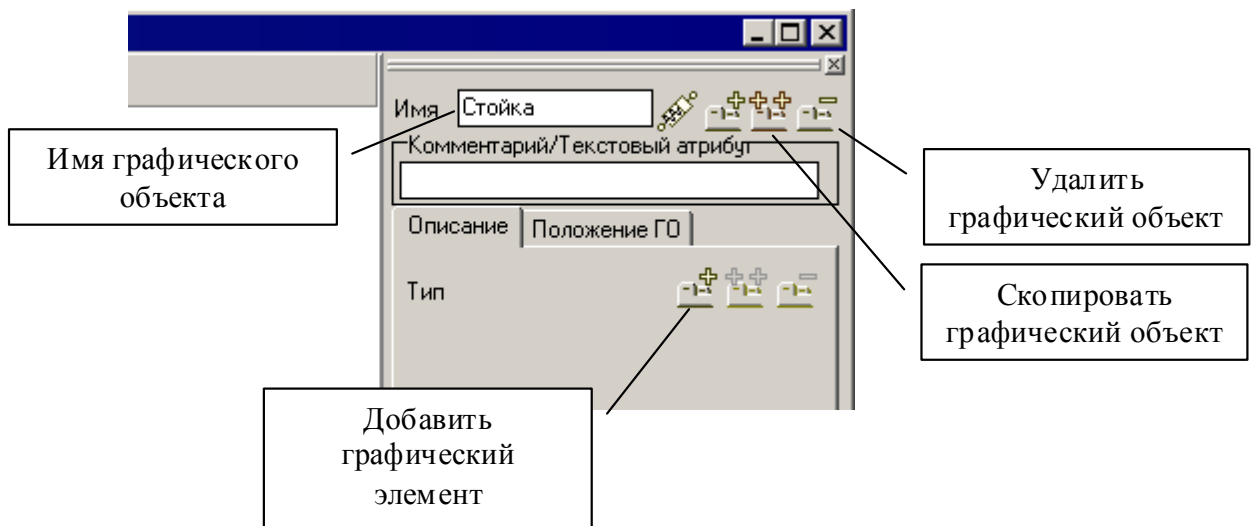


Рис. 1.4. Переименование графического объекта

Создание графических элементов

Каждый *графический объект* может состоять из произвольного числа *графических элементов* (примитивов, ГЭ), что позволяет создавать весьма сложные образы. В нашем случае создадим три элемента - шар, конус и параллелепипед, которые вместе и образуют образ стойки.

Создание нового графического элемента: шар

4. Нажмите кнопку **Добавить графический элемент**, см. рис. 1.4.

Образовалась новая закладка **GE1**, см. рис. 1.5.

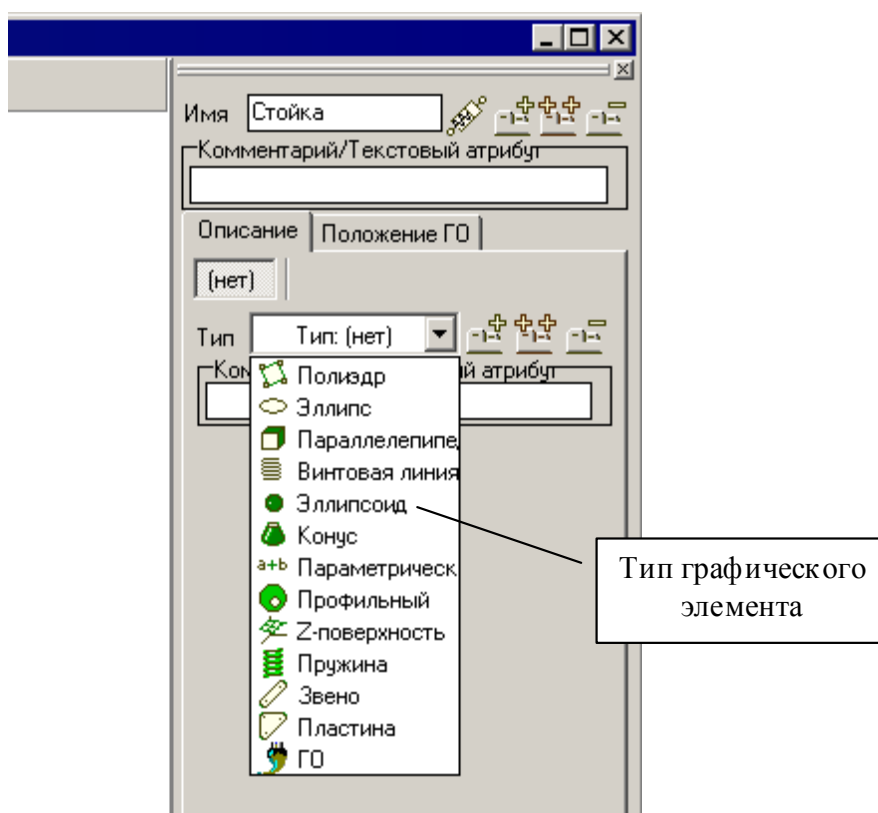


Рис. 1.5. Тип графического элемента

5. Выберите тип нового *графического элемента* – **Эллипсоид**.
6. На закладке **Параметры** задайте полуоси эллипсоида **$a = b = c = 0.05$** .
7. Перейдите на закладку **Цвета** и выберите красный цвет в качестве диффузного.

Создание нового графического элемента: конус

8. Добавьте новый *графический элемент* и выберите его тип – **Конус**.

Замечание. Не путайте создание нового *графического объекта* и создание новых *графических элементов* (примитивов) в рамках одного *графического объекта*. В данном примере мы создаем один *графический объект* – **Стойка**, который будет включать три *графических элемента*: шар, конус и параллелепипед.

9. На закладке **Параметры** установите **$R2 = 0.1$; $R1 = 0$; $h = 0.15$** .

10. Перейдите на закладку **Цвета** и выберите красный цвет в качестве диффузного.

Создание нового графического элемента: параллелепипед

11. Добавьте новый графический элемент и выберите его тип – **Параллелепипед**.
12. На закладке Параметры установите **A = 0.5; B = 0.5; C = 0.05**.
13. Перейдите на закладку **Положение ГЭ**. В поле **Z**, группы **Сдвиг** установите **0.15**, см. рис. 1.6.

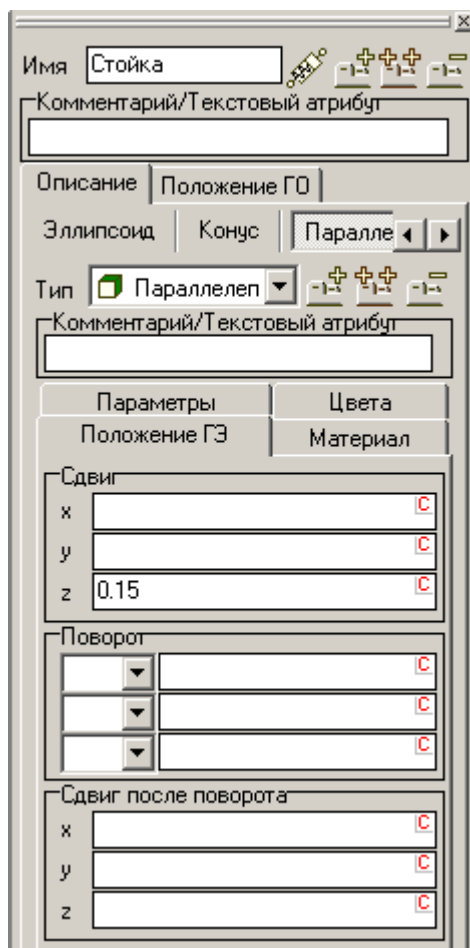


Рис. 1.6. Положение графического элемента

Назначение графического объекта образу сцены

14. Перейдите на ветку **Объект** в списке элементов модели.

15. В поле **Образ сцены** выберите **Стойка**, см. рис. 1.7.

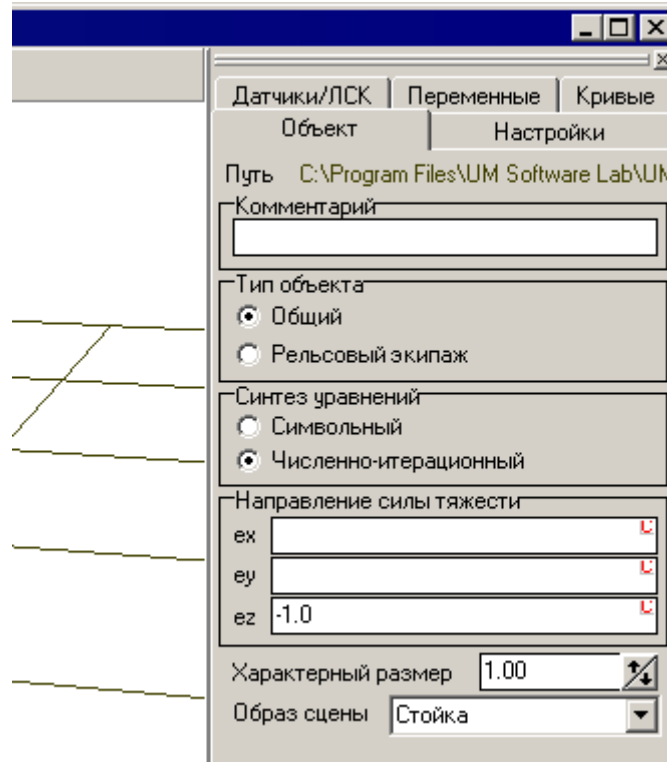


Рис. 1.7. Выбор образа сцены

1.3.3.2. Образ маятника

1. Вернитесь на ветку **Графические объекты** в списке элементов модели.
2. Создайте новый графический объект.
3. Переименуйте второй графический объект в **Маятник**.

Замечание. Обратите внимание, что после ввода любых данных следует нажимать клавишу **Enter**, чтобы введенная информация тут же обрабатывалась программой.

Образ маятника будем изображать двумя графическими элементами: эллипсоидом и конусом.

4. Добавьте новый графический элемент **Эллипсоид** с параметрами **$a = 0.05$; $b = 0.2$; $c = 0.2$** . Выберите синий цвет в качестве диффузного.
5. Добавьте новый графический элемент **Конус** с параметрами **$R2 = 0.03$; $R1 = 0.03$; $h = 1$** . Выберите синий цвет в качестве диффузного.

Графический образ маятника создан.

1.3.4. Создание тел

Маятник как механическая система образован единственным телом.

1. Для его описания выберите ветку **Тела** в списке элементов модели.
2. Создайте новое тело (так же как новый графический объект).
3. Переименуйте новое тело в **Маятник**
4. Выберите графический образ **Маятник** из списка доступных графических образов.
5. В поле **Масса** поставьте **1** (кг).

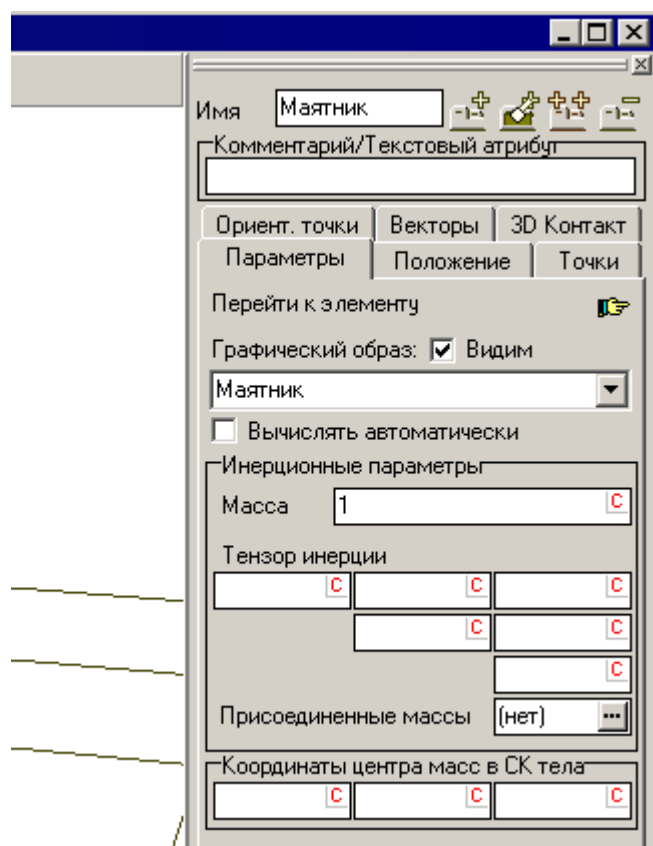


Рис. 1.8. Параметры маятника

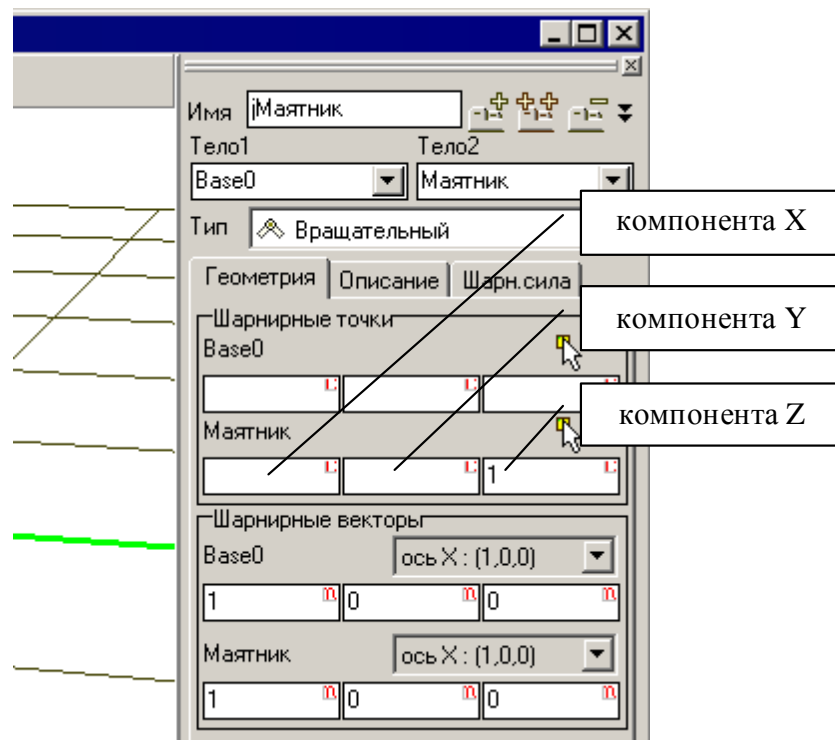
1.3.5. Создание шарниров

Маятник и стойку связывает вращательный шарнир. Для его создания выполните следующие операции.

1. В дереве элементов модели выберите **Тела/Маятник**.
2. В инспекторе нажмите кнопку **Перейти к элементу**
3. В появившемся списке выбрать раскрывающийся пункт меню **Создать шарнир**, затем выберите **Вращательный**.

После этого автоматически будет создан вращательный шарнир с именем **jМаятник**, связывающий базу (Base0, СК0) и маятник. Шарнирные точки и шарнирные векторы определяют положение оси вращения относительно каждого тела.

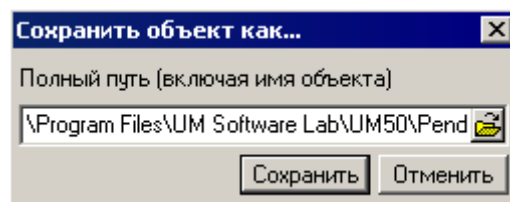
4. В поле **Шарнирные точки/Маятник** задайте компоненте **Z** значение **1**. Таким образом, мы заставим маятник вращаться вокруг своей верхней точки.



1.3.6. Сохранение введенных данных

После того, как мы описали тело и шарнир, модель маятника описана полностью. Самое время сохранить все введенные данные. По умолчанию модель называлась **UmObj0**, дадим модели новое имя – **Pend**.

1. Выберите пункт меню **Файл/Сохранить как...**
2. В поле полный путь введите *{Путь к модели}***Pend**, так, как показано на рисунке ниже.



1.3.7. Подготовка к моделированию: синтез уравнений движения

Программный комплекс «Универсальный механизм» состоит из двух независимых программ: *программы описания моделей* (UM Input) и *программы моделирования движения* (UM Simulation). Прежде чем перейти к моделированию динамики маятника, следует рассмотреть проблему синтеза уравнений движения.

«Универсальный механизм» поддерживает два способа синтеза уравнений движения: *символьный* и *численно-итерационный*. Рассмотрим их более подробно.

Символьный синтез уравнений движения

Символьный синтез уравнений движения предполагает вывод уравнений на одном из поддерживаемых языков программирования (в настоящей версии программы поддерживаются *C* и *Pascal*). Далее эти синтезированные файлы будут откомпилированы некоторым заданным внешним компилятором, что на выходе даст динамически загружаемую библиотеку *UMTask.dll*, которая будет подключаться к *программе моделирования движения*.

Синтез исходных текстов и запуск внешнего компилятора выполняется из *программы описания моделей*.

Численно-итерационный синтез уравнений движения

Численно-итерационный метод предполагает вывод уравнений движения численно на каждом шаге численного метода интегрирования уравнений движения непосредственно в *программе моделирования движения*.

Сравнение методов

Рассмотрим преимущества и недостатки обоих методов синтеза уравнений движения.

С точки зрения скорости численного моделирования *символьный* метод эффективнее на 10-30% для относительно сложных моделей с числом степеней свободы более 10-20. Для моделей с числом степеней свободы менее 10 разницы в скорости численного моделирования двумя методами практически нет. Эта разница связана с тем, что на этапе символьного вывода уравнений

движения получившиеся выражения оптимизируются с точки зрения быстродействия. Так, например, выполняется вынос общих множителей за скобки и удаление операций умножения и сложения с нулем. Тогда как при численно-итерационном способе синтеза уравнений движения такая оптимизация невозможна.

С другой стороны, символьный метод синтеза уравнений движения требует наличия на компьютере пользователя внешнего компилятора. «Универсальный механизм» поддерживает в качестве внешних компиляторов Borland Delphi, Borland C++ Builder, Microsoft Visual C++. Эти компиляторы не поставляются вместе с «Универсальным механизмом» и должны быть лицензированы отдельно.

Вместе с тем, численно-итерационный метод не требует явного этапа синтеза и компиляции (а также предварительной настройки компиляторов) и таким образом проще в использовании.

Рекомендации

Начинающим пользователям рекомендуется выбирать численно-итерационный метод синтеза уравнений, как более простой в использовании. Символьный метод может быть рекомендован более опытным пользователям, исследующим сложные модели.

Далее в уроках мы будем придерживаться только численно-итерационного метода синтеза уравнений.

В разделах 1.3.7.1 и 1.3.7.2 описана последовательность подготовки к моделированию при использовании, соответственно, численно-итерационного и символьного методов синтеза уравнений движения.

1.3.7.1. Численно-итерационный метод

В списке элементов модели в левой верхней части окна конструктора объектов выберите **Объект**. В **Инспекторе моделирования** в поле **Синтез уравнений** выберите **Численно-итерационный**.

Сохраните модель еще раз для сохранения метода синтеза уравнений. Теперь модель готова для использования в программе моделирования движения.

Перейдите к разделу «1.3.8. Переход к моделированию движения», стр. 22.

1.3.7.2. Символьный метод

При символьном методе генерации уравнений одним из результатов работы с программой описания моделей является динамическая библиотека *UmTask.dll*, которая получается после синтеза уравнений движения на одном из доступных алгоритмических языков и компиляции этих уравнений при помощи внешнего компилятора. Эта динамическая библиотека всегда находится в каталоге задачи. После ее создания можно переходить к моделированию движения.

Чтобы установить символьный способ синтеза уравнений движения в поле **Синтез уравнений** выберите **Символьный**.

Далее мы должны выполнить синтез уравнений движения, компиляцию уравнений движения и, в случае успешного завершения этих операций, перейти к исследованию динамики нашей модели в программе моделирования движения UM Simulation.

Настройка путей к внешнему компилятору

1. Выберите пункт меню **Инструменты/Настройки**.

«Универсальный механизм» поддерживает в качестве внешних компиляторов Borland Delphi, Borland C++ Builder, Microsoft Visual C++. Дальнейшие действия зависят от того, какой из вышеперечисленных компиляторов установлен у вас на компьютере:

Delphi

2. Перейдите на закладку **Пути/Delphi**.
3. Нажмите кнопку **Найти Delphi**.

Borland C++ Builder, Microsoft Visual C++

2. Перейдите на закладку **Пути/C++**.
3. Нажмите кнопку **Найти Visual C** или **Найти Borland C++ Builder** в зависимости от того, какой конкретно компилятор C установлен на вашем компьютере.

В случае успешного обнаружения поддерживаемых компиляторов автоматически инициализируются поля путей поиска компиляторов и библиотек.

Существует возможность использования компиляторов, установленных на ПК, входящих в локальную сеть. При включении флажка **Сеть** появляется поле выбора ПК локальной сети, на котором предполагается найти компилятор. Если компилятор установлен на выбранном ПК и у текущего пользователя есть доступ к файлам компилятора, то его можно выбрать для компиляции.

Если не удастся автоматически определить компилятор (например, установлена версия компилятора выше последней, включенной во внутренний список доступных компиляторов УМ), то пути к компилятору и библиотекам следует указать вручную.

Синтез и компиляция уравнений движения

4. Выберите пункт меню **Объект/Синтезировать** уравнения.

Если объект описан корректно, то появится окно параметров синтезатора уравнений, иначе, автоматически открывается **Протокол** с сообщениями об обнаруженных ошибках описания модели.

5. В появившемся окне параметров синтезатора уравнений установите галочку в поле **Компилировать уравнения**.

6. Нажмите кнопку **Синтезировать**.

7. Нажмите кнопку **Выйти**.

При успешном завершении синтеза и компиляции уравнений движения пользователю будет подано сообщение: «**Компиляция успешно завершена. Задача готова для моделирования**». Задача готова для загрузки в программу моделирования динамики.

1.3.8. Переход к моделированию движения

1. Выберите пункт меню **Объект/Моделирование**.

Это приведет к запуску программы моделирования движения с открытием в ней текущей модели.

1.4. Моделирование движения маятника

Мы находимся в программе моделирования движения. Сейчас мы откроем новое анимационное окно, отклоним маятник на один радиан от вертикального положения и запустим моделирование движения.

Создание нового анимационного окна

1. Выберите пункт меню **Инструменты/Анимационное окно**.

Появится новое анимационное окно с изображением загруженной модели. Познакомимся с тем, как управлять положением камеры в анимационном окне.

Поворот

Наведите курсор мыши на анимационное окно так, чтобы он принял вид, который показан на рисунке справа. Нажмите левую кнопку мыши и, перемещая мышь, вращайте модель в анимационном окне. Пункт **Стиль поворота** всплывающего меню анимационного окна дает возможность выбора стиля поворота. **X-, Y- и Z-стили** удобно использовать в моделях, где объект(ы) вытянут вдоль соответствующей оси. Стиль **По сфере** идентичен стилям поворота в большинстве САД-систем.



Перемещение

Наведите курсор мыши на анимационное окно так, чтобы он принял вид «Поворот», и нажмите кнопку **Ctrl** на клавиатуре и курсор мыши изменит свой вид на тот, который показан на рисунке справа. Нажмите левую кнопку мыши и, перемещая мышь, перемещайте модель в анимационном окне.

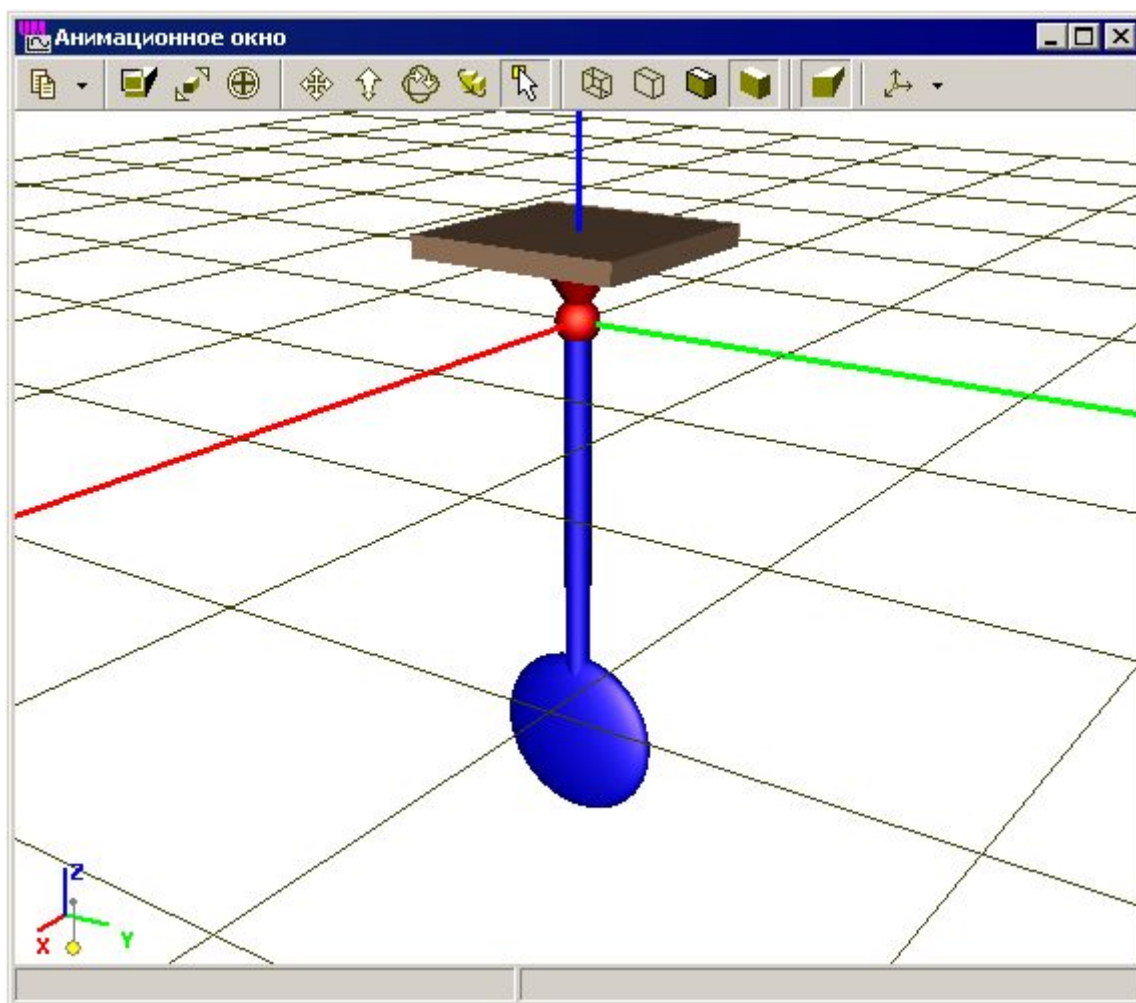


Приближение/Удаление

Наведите курсор мыши на анимационное окно так, чтобы он принял вид «Поворот», и нажмите кнопку **Shift** на клавиатуре. Приближайте/удаляйте модель, нажав левую кнопку мыши и перемещая мышь. Кроме того, для изменения масштаба можно использовать колесо мыши.



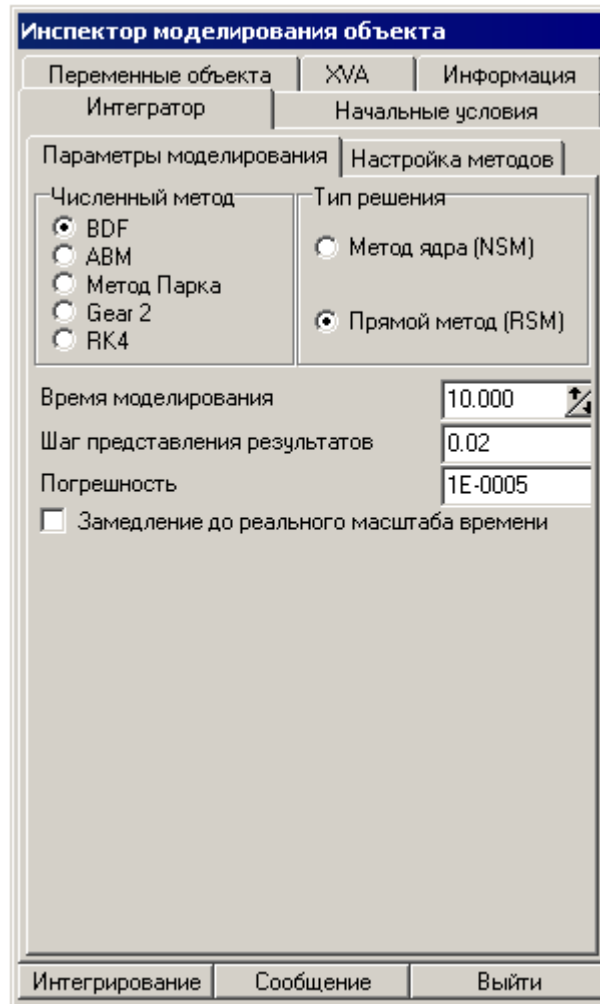
Немного поупражнявшись с работой в анимационном окне, вы можете получить что-то похожее на то, что показано на рисунке ниже.



Переход в режим моделирования

1. Выберите пункт меню **Анализ/Моделирование**.

Появится окно инспектора моделирования объекта, см. рис. ниже.



Задание начальных условий

Чтобы наш маятник закачался его нужно предварительно отклонить. Для этого воспользуемся диалогом задания начальных условий.

2. Перейдите на закладку **Начальные условия** | **Координаты Инспектора моделирования объекта**.

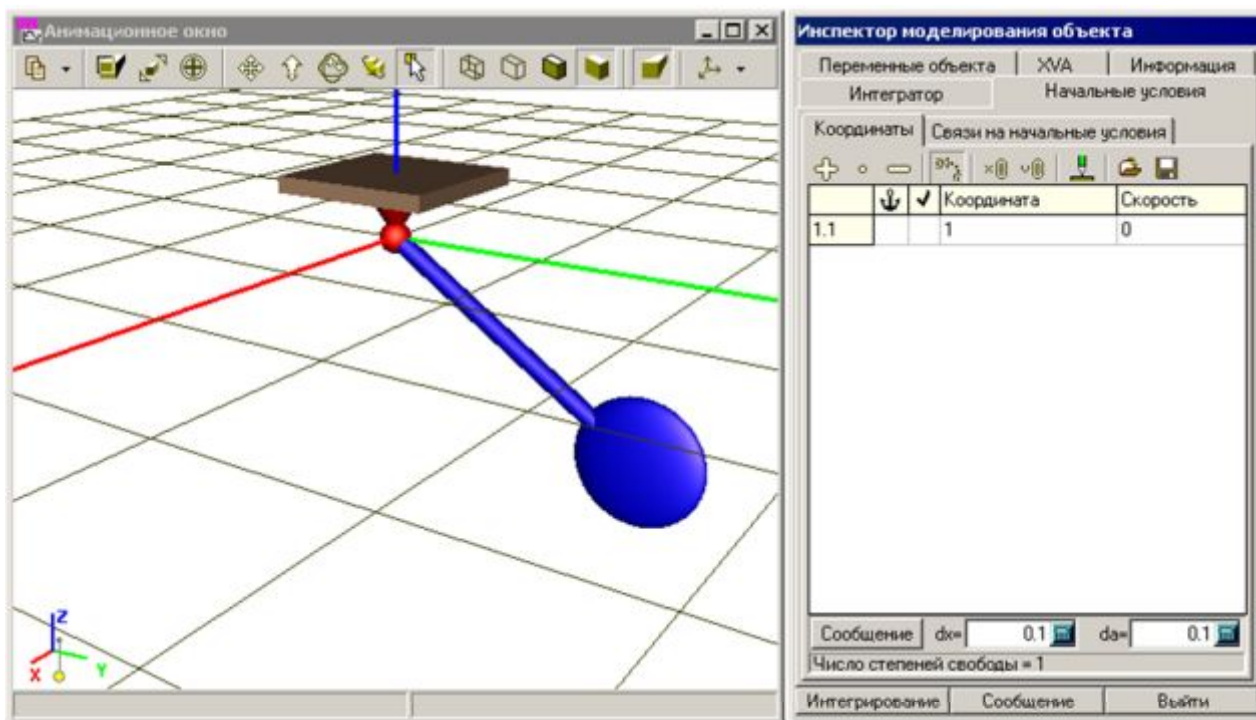
Сейчас вы видите список всех координат системы. Для нашего маятника это одна координата в шарнире **jМаятник**.

3. Введите в поле **Координата** значение **1**. Нажмите клавишу **Enter**.

В анимационном окне маятник повернется на 1 радиан¹.

Замечание. Не забывайте нажимать клавишу **Enter**, чтобы вводимые данные тут же отражались программой.

Помните также, что в программе используется система **СИ**. Размерность угловых величин – радианы.



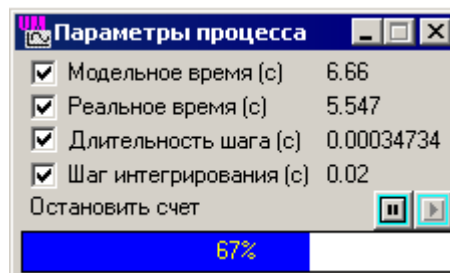
¹ Единица измерения угловых величин в системе СИ – радианы.

Моделирование движения маятника

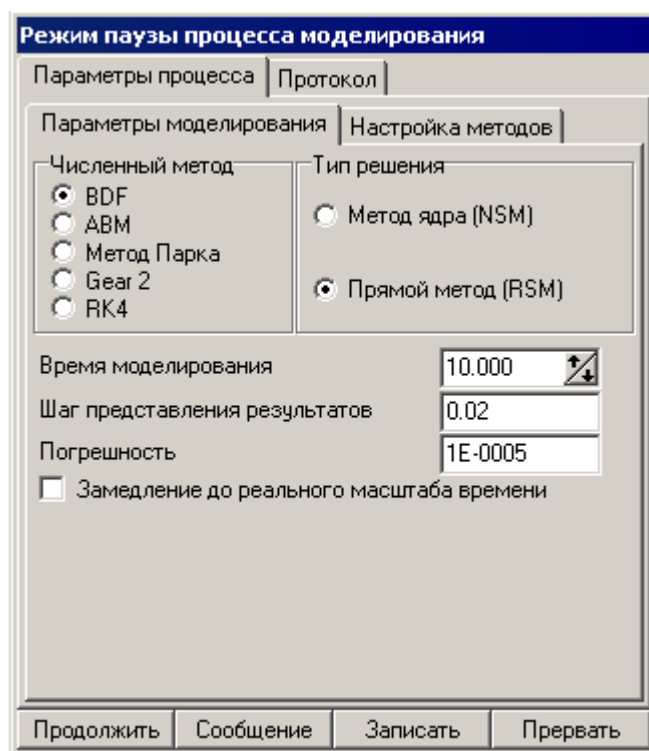
После того, как мы отклонили маятник, все готово к моделированию. Для начала просто запустим процесс моделирования на **10** секунд (установка по умолчанию).

4. Нажмите кнопку **Интегрирование** в окне инспектора моделирования.

В правом нижнем углу экрана появится окно отображения текущих параметров процесса интегрирования.



После окончания появится окно **Инспектора режима паузы процесса моделирования**. В нем вы можете увеличить время моделирования, изменить параметры численного метода.



5. Нажмите кнопку **Прервать**. Появится окно **инспектора моделирования объекта**.

Построение графиков

В процессе моделирования вы можете строить графики различных величин. Для этого сначала откроем новое графическое окно, затем в мастере переменных создадим новую переменную – проекцию центра масс маятника на ось Y. Перетащим мышкой созданную переменную из мастера переменных в графическое окно. После этого в процессе моделирования график переменной будет строиться в графическом окне.

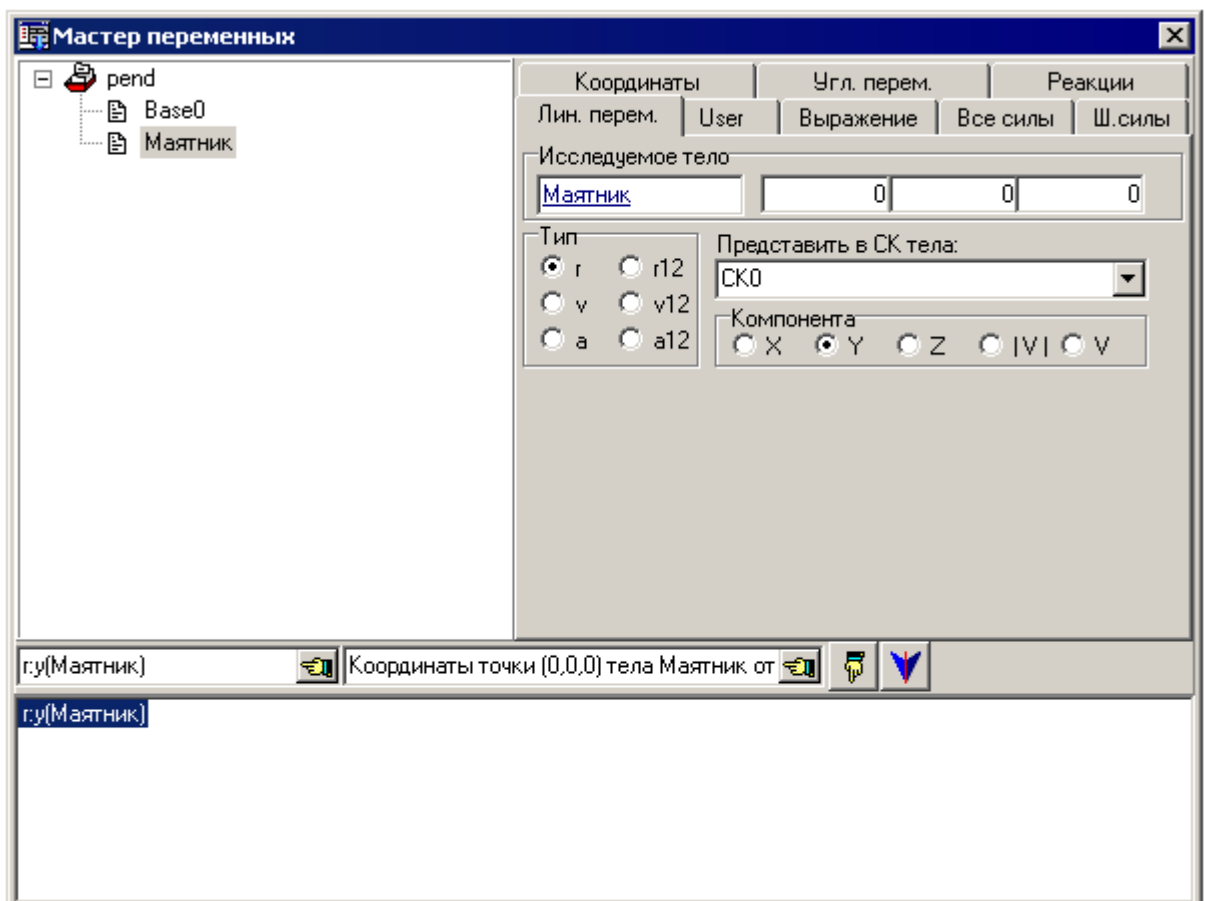
Итак, создадим новое графическое окно.

6. Выберите пункт меню **Инструменты/Графическое окно**.


Далее откройте окно **Мастера переменных**.

7. Выберите пункт меню **Инструменты/Мастер переменных**.

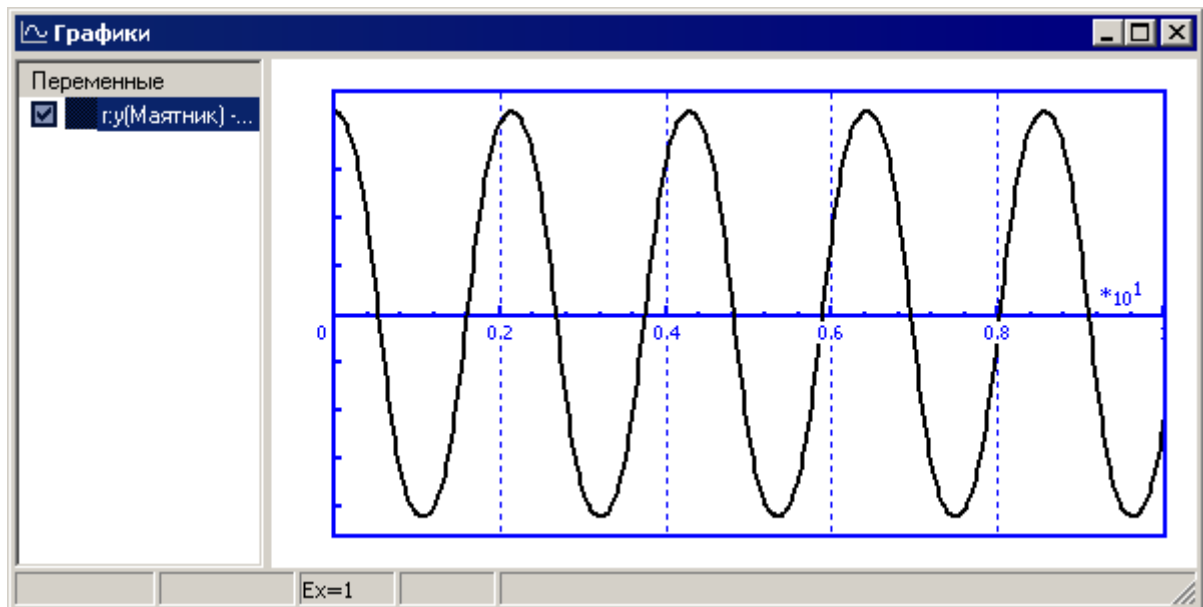
Мастер переменных – специальный инструмент для создания переменных, которые затем можно построить в графическом окне, отображать в анимационном окне (если это вектор), сохранить в списке переменных.



Давайте построим график перемещения центра масс маятника в проекции на ось Y.


8. Перейдите на закладку **Лин. перем.** (Линейные перемещения).
9. В разделе **Компоненты** выберите **Y**.
10. Поместите описанную переменную в контейнер, нажав кнопку . В контейнере переменных в нижней части **Мастера переменных** появится переменная **г:у(Маятник)**.
11. Выделите эту переменную и мышкой перетащите ее в графическое окно.
12. Перейдите к **инспектору моделирования** объекта и нажмите кнопку **Интегрирование**.

В графическом окне построится график переменной, см. рис. ниже.



Построение векторов и траекторий

В процессе моделирования вы можете строить векторы различных величин в анимационном окне. Для анимации вектора скорости центра масс маятника, сначала надо создать для него переменную в **мастере переменных**.

- 13.Перейдите в окно **мастера переменных** и выберите закладку **Лин. перем.**
- 14.В разделе **Тип** выберите v (скорость).
- 15.В разделе **Компонента** укажите V (вектор).
- 16.Поместите описанную переменную в контейнер, нажав кнопку .
- 17.Выделите новую переменную в контейнере и мышкой перетащите ее в анимационное окно.

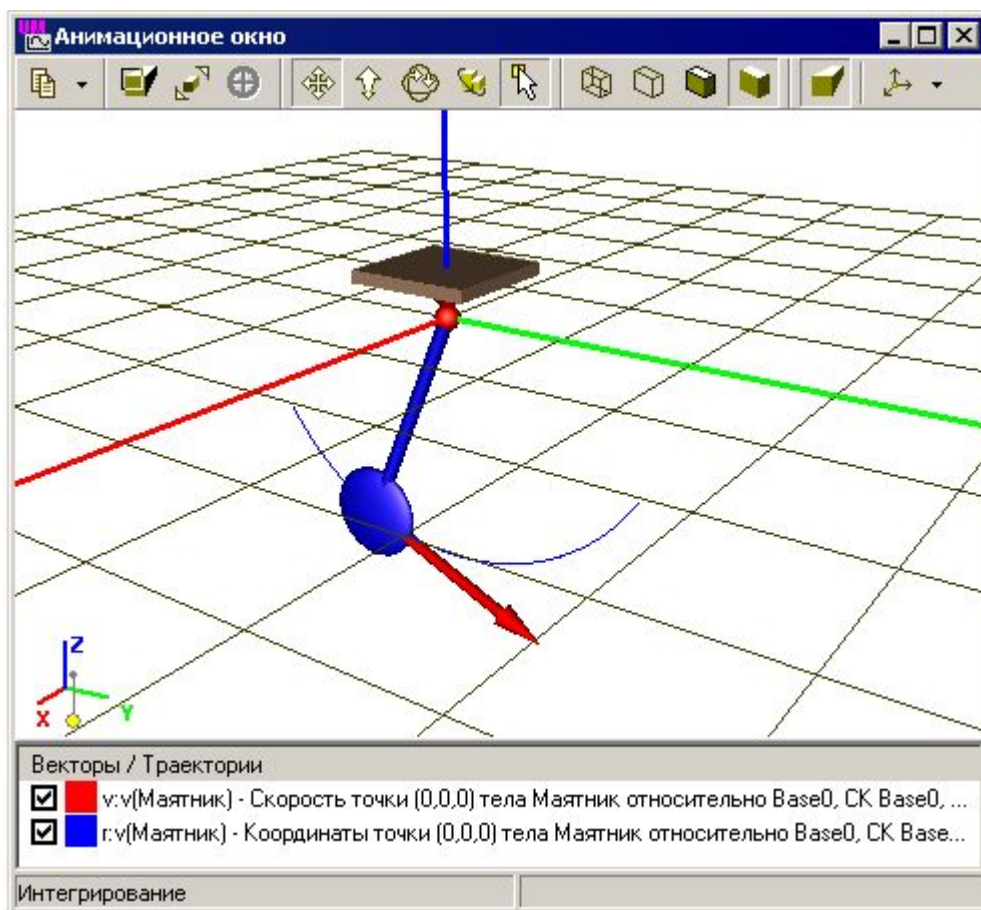
По умолчанию список переменных в анимационном окне не показывается. Для управления его положением используйте пункт **Положение списка векторов** контекстного меню в анимационном окне.

- 18.Вызовите контекстное меню анимационного окна и выберите пункт меню **Положение списка векторов/Снизу**.

Кроме вектора скорости построим также траекторию маятника в анимационном окне.

- 19.Повторите действия по созданию вектора скорости, только в разделе **Тип** укажите r (радиус-вектор). Перетащите переменную в анимационное окно.
- 20.Запустите интегрирование.

На этот раз в анимационном окне вы дополнительно видите вектор скорости и траекторию. Для изменения масштаба вектора скорости в анимационном окне используйте пункт контекстного меню **Масштаб векторов**. Двойной щелчок мыши на элементе списка векторов (или на образе вектора в режиме паузы) позволит вам изменить цвет вектора и траектории, а для траектории – также изменить число точек ("длину" траектории).

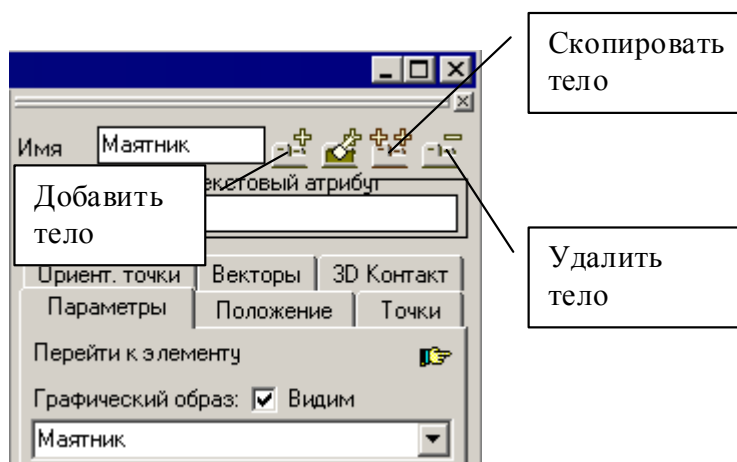


1.5. Многозвенный маятник

Превратить объект в многозвенный маятник, то есть цепочку маятников, можно двумя способами. Первый из способов заключается в доработке текущей модели путем копирования ранее созданных элементов; второй – в создании новой модели с использованием метода подсистем.

1.5.1. Доработка модели однозвенного маятника

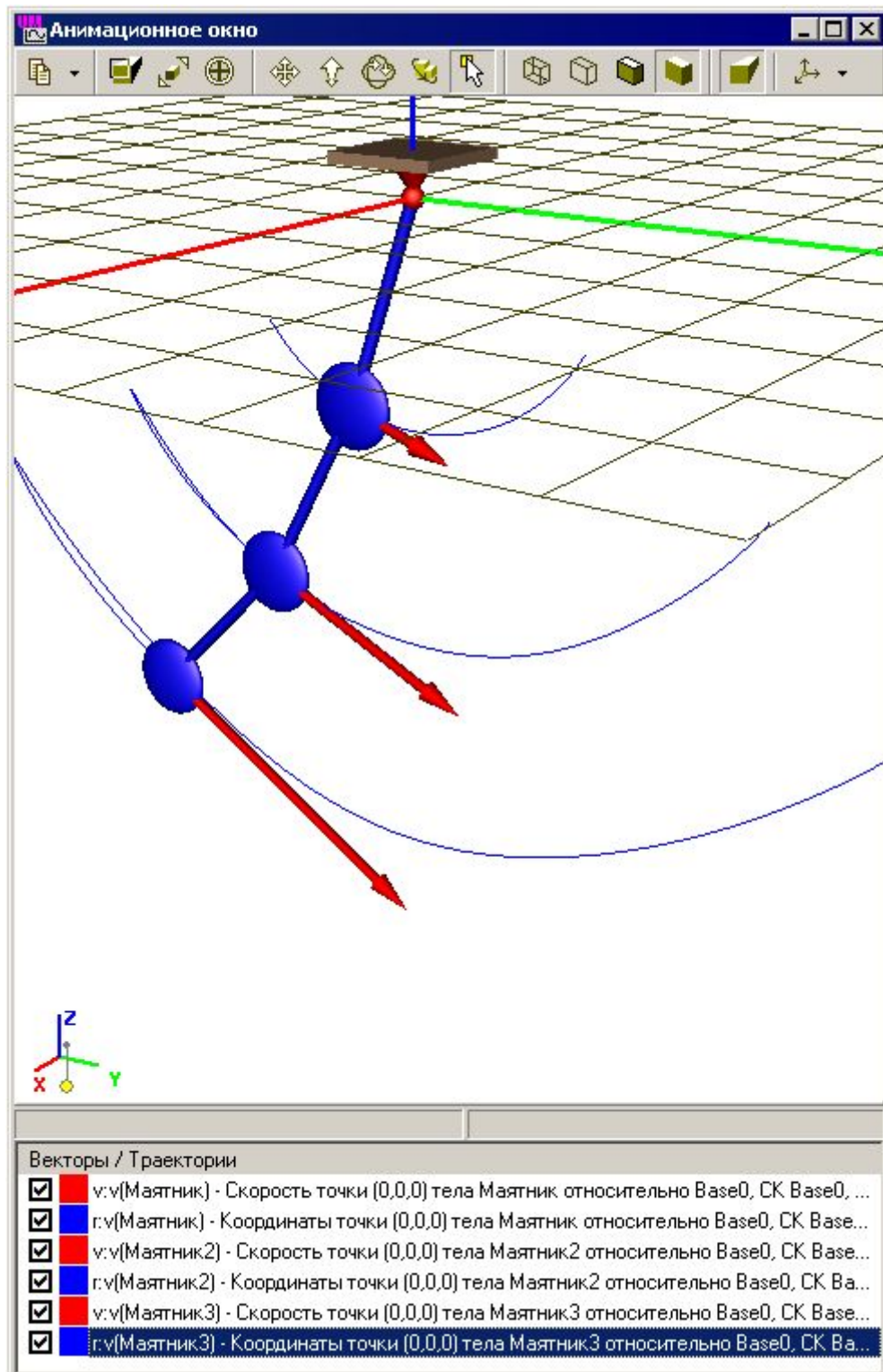
1. Закройте программу моделирования и вернитесь в программу ввода данных.
2. Перейдите на ветку **Тела** и скопируйте маятник два раза.



3. Переименуйте тело **Маятник** в **Маятник1**, а добавленные тела – в **Маятник2** и **Маятник3**.
4. Перейдите на ветку **Шарниры** и скопируйте единственный существующий шарнир два раза.
5. Измените тела, соединяемые шарнирами: **Маятник1** и **Маятник2** для второго шарнира и **Маятник2** и **Маятник3** для третьего.
6. Сохраните модель с именем **ThreeLinkPend** (пункт меню **Файл/Сохранить как...**).
7. Выберите численно-итерационный метод синтеза уравнений движения (см. п. 1.3.7.1) и синтезируйте их (пункт меню **Объект/Синтезировать уравнения...**).
8. Запустите программу моделирования (пункт меню **Объект/Моделирование...**).
9. Откройте анимационное окно.
10. Перейдите в режим моделирования, пункт меню **Анализ/Моделирование**.

11. Задайте маятникам начальное положение – на закладке **Начальные условия** инспектора моделирования объекта задайте значение **0.3** для каждой из трех шарнирных координат.

12. Запустите моделирование кнопкой **Интегрирование**.



1.5.2. Использование метода подсистем

Метод подсистем позволяет создавать *составные* модели механической системы из набора стандартных и разработанных пользователем подсистем – моделей отдельных частей систем.

Для создания модели девятизвенного маятника трижды добавим ранее созданную модель трехзвенного маятника (ТЗМ – ТрехЗвенныйМаятник) в качестве подсистемы.

В этом примере мы проиллюстрируем возможности создания *составных* моделей с использованием *включенных* подсистем.

1.5.2.1. Структура составного объекта

Прежде чем приступить к созданию составной модели, определимся с ее структурой и особенностями описания.

Внешние и включенные подсистемы

В ПК «Универсальный Механизм» реализовано два механизма добавления подсистем, созданных пользователем. Добавление *внешней* подсистемы создает ссылку на ранее созданную модель, добавление объекта как *включенной* подсистемы фактически означает включение его в составную модель со всей структурой (телами, шарнирами, силовыми элементами, графическими образами и так далее) и параметрами.

Для создания модели многозвенного маятника на базе *кинематически тождественных* подсистем (трехзвенного маятника) наиболее эффективным способом является использование *внешних* подсистем. В то же время, использование *включенных* подсистем является более распространенным при создании составных моделей, поэтому в нашем примере мы будем использовать *включенные* подсистемы.

Описание взаимодействия между подсистемами

Существует два подхода к описанию составного объекта. Первый подход предполагает описание шарниров и силовых элементов как элементов составного объекта, соединяющих тела, входящие в различные подсистемы.

Альтернативой данному подходу является использование при создании подсистем внешних тел, обозначающихся через специальное тело **External**. В этом случае при описании составного объекта необходимо после добавления подсистемы описать *Внешние связи* – выбрать для каждого внешнего шарнира и силового элемента подсистемы точку связи, принадлежащую телу со-

ставного объекта или другой подсистемы. Установить какую либо связь, значит указать, что за тело принять в качестве **External** в рассматриваемом шарнире или силовом элементе и какую точку этого тела выбрать в качестве характерной для рассматриваемого элемента.

При создании модели девятизвенного маятника сначала соединим подсистемы (модели трехзвенных маятников) с помощью шарниров – элементов составной модели (схема А – слева), а затем сделаем то же самое с помощью внешних связей (схема Б – справа).

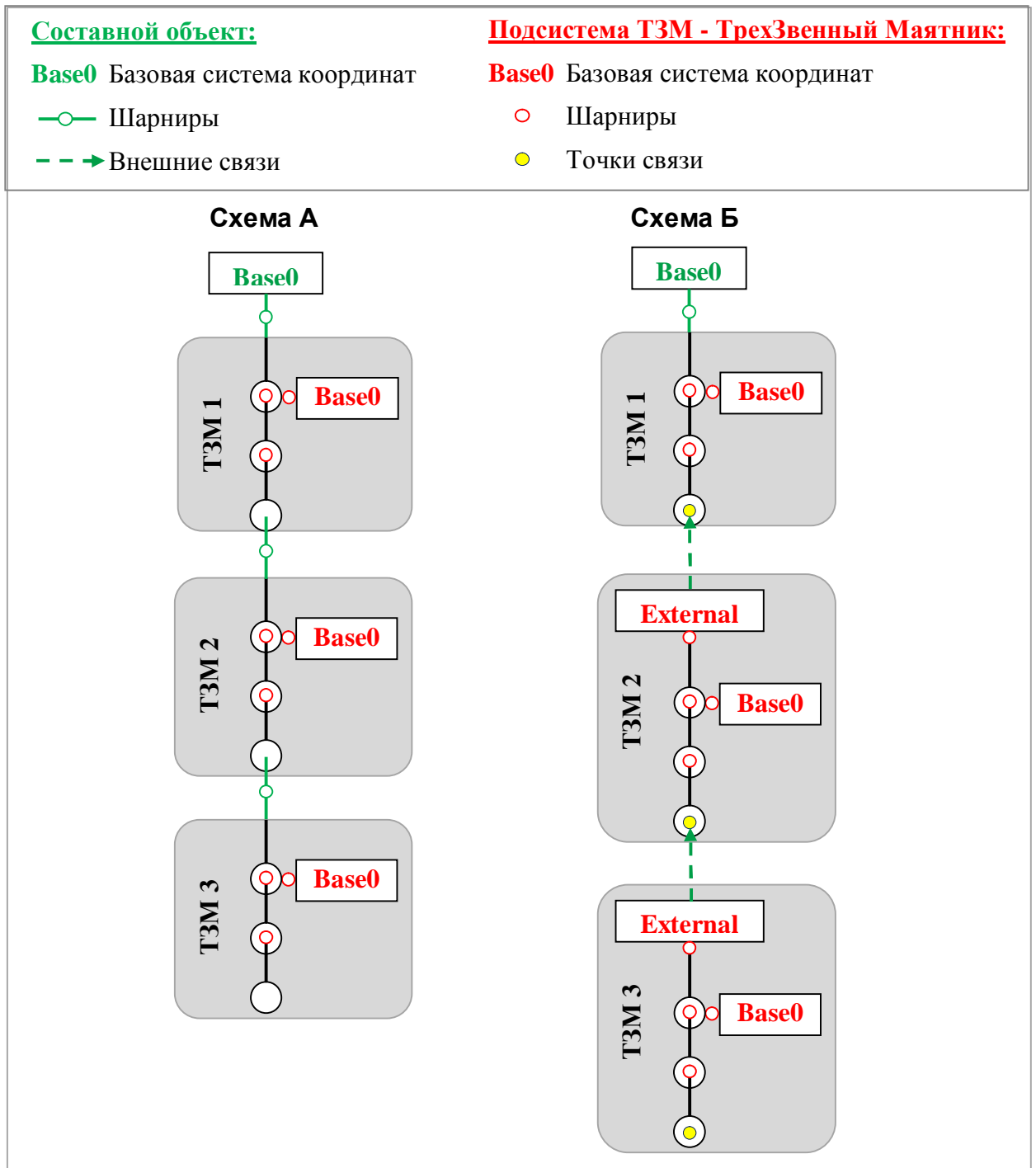


Рис. 1.9. Структура составной модели девятизвенного маятника

1.5.2.2. Подготовка модели подсистемы

Для использования модели трехзвенного маятника в качестве подсистемы составного объекта необходимо внести в нее ряд изменений.

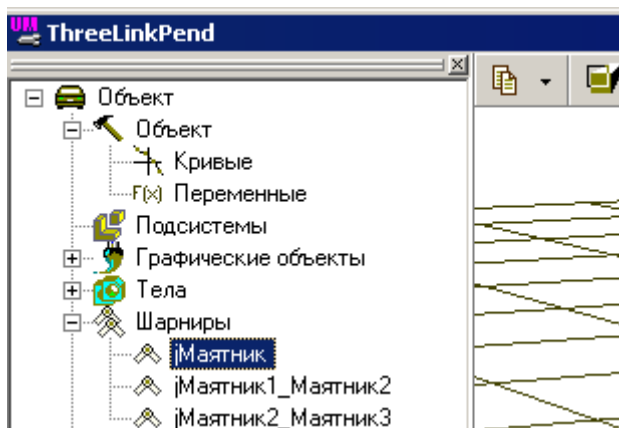
Изменение кинематической схемы

Поскольку вращательный шарнир, описывающий подвес девятизвенный маятника, мы будем задавать в составной модели, а шарниры, последовательно соединяющие подсистемы **ТЗМ1**, **ТЗМ2** и **ТЗМ2**, **ТЗМ3**, будут описываться либо в составном объекте (рис. 1.9, схема А), либо с использованием внешних связей (рис. 1.9, схема Б), то шарнир-подвес (**jМаятник**) в подсистеме не требуется.

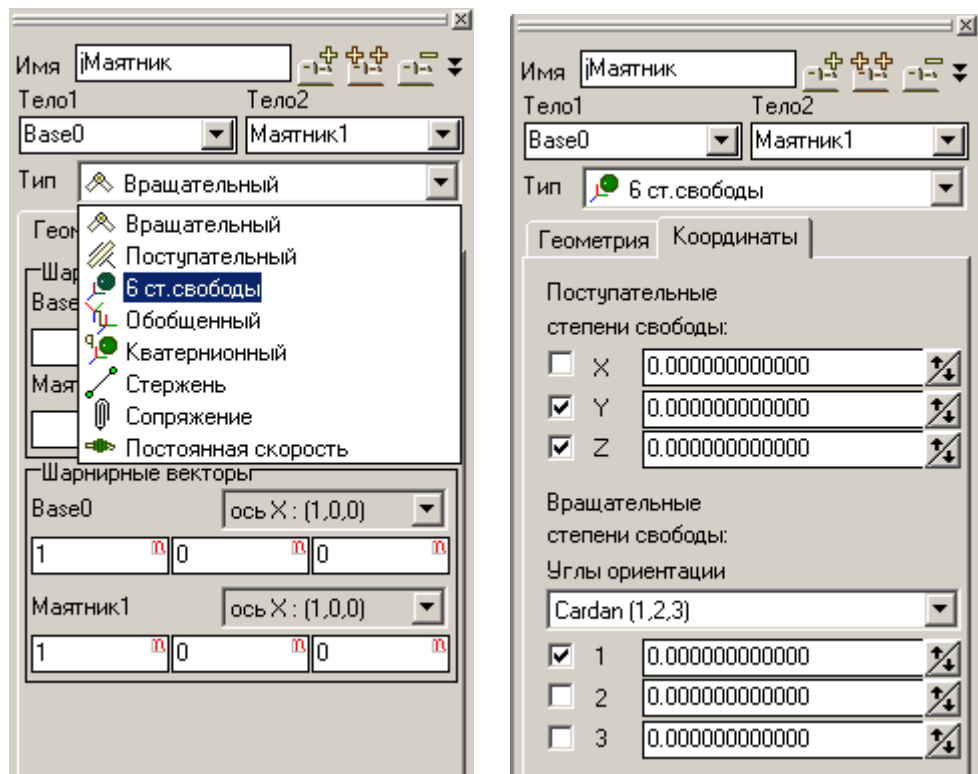
Для подсистемы должно выполняться условие *связности*. Это означает, что любое тело подсистемы должно быть связано с базовым телом (**Base0**) шарниром, либо цепочкой тел, попарно соединенных шарнирами. При этом кинематическая модель подсистемы должна учитывать характер движения составной системы. Тела многозвенного маятника совершают плоское движение, поэтому, в модели подсистемы они должны иметь возможность перемещаться относительно **Base0** в плоскости качания.

Для выполнения этих условий заменим вращательный шарнир-подвес (**jМаятник**) в модели **ThreeLinkPend** на шарнир с тремя степенями свободы, позволяющий системе выполнять движение в плоскости качания маятника.

1. Закройте программу моделирования и вернитесь в программу ввода данных (с объектом **ThreeLinkPend**).
2. Перейдите на ветку **Шарниры** и выделите шарнир **jМаятник**, описывающий вращательную степень свободы тела **Маятник1** относительно **Base0**.



3. В инспекторе измените **Тип шарнира** на **Шарнир с 6 степенями свободы**. При этом возникнет предупреждение о возможной потере ранее введенных данных. Подтвердите изменение типа.



4. Отключите **3** степени свободы как показано на рисунке справа.


Теперь тело **Маятник1**, а также соединенные с ним тела **Маятник2** и **Маятник3**, могут совершать плоское движение относительно **Base0**.

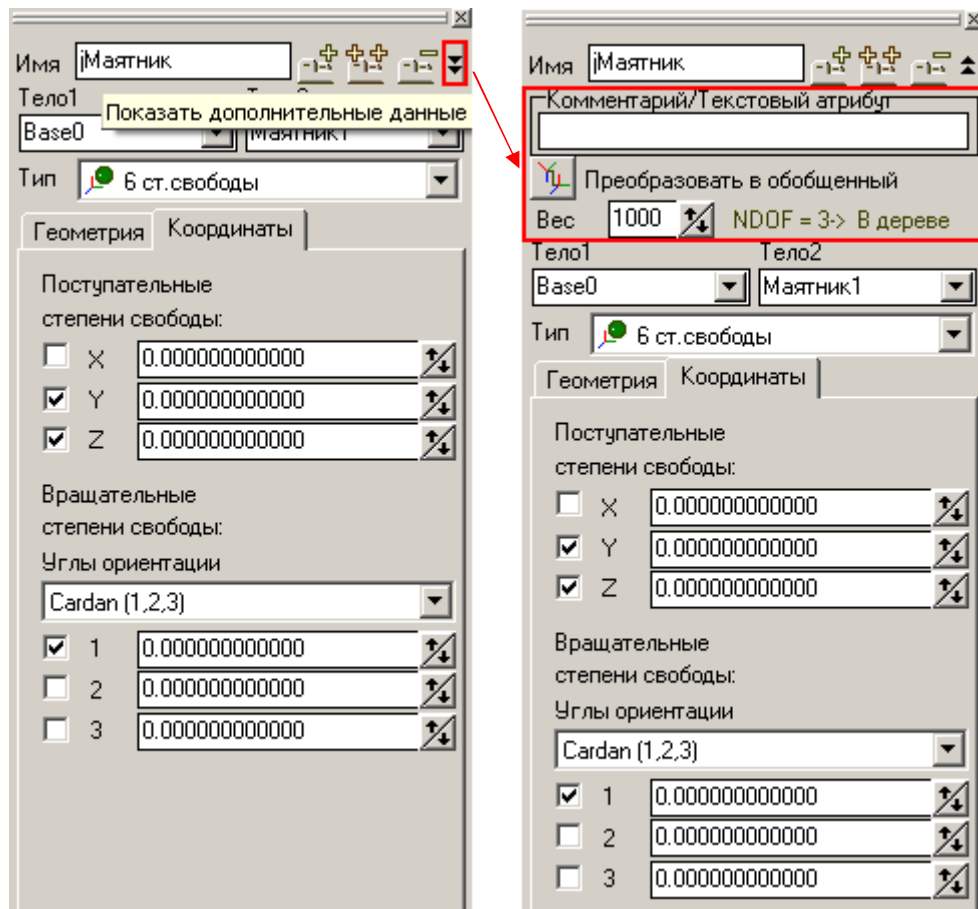
Выбор разрезаемых шарниров

После добавления подсистем и описания связывающих их шарниров составная модель будет иметь несколько замкнутых кинематических цепей. Другими словами, количество координат, соответствующее набору введенных шарниров, будет превышать число степеней свободы системы.

В подобных случаях часть шарниров условно разрезается. Автоматический выбор разрезаемых шарниров является неоднозначным и поэтому имеется возможность их наилучшего, оптимального выбора пользователем.

Для описания многозвенного маятника удобно использовать координаты во вращательных шарнирах, соединяющих отдельные звенья. Поэтому, мы зададим **Веса шарниров** таким образом, чтобы при синтезе уравнений движения были разрезаны шарниры, соединяющие тела подсистем с базовым телом.

5. В инспекторе нажмите кнопку , чтобы отобразить панель дополнительной информации.



6. В поле **Вес** задайте значение **1000**. При синтезе уравнений движения при прочих равных условиях разрезаются шарниры с большими весами.
7. Сохраните модель под именем **ThreeLinkPend_Subs** (пункт меню **Файл/Сохранить как**).

Доработка модели трехзвенного маятника для использования в качестве подсистемы составного объекта по схеме А (рис. 1.9, стр. 35) завершена.

Теперь мы можем приступить к созданию составной модели.

1.5.2.1. Описание составного объекта

Создание нового объекта

1. Не закрывая объект **ThreeLinkPend_Subs**, создайте новый объект (пункт меню **Файл/Новый объект**) и сохраните его под именем **NineLinkPend**.

Далее, в программе ввода данных мы будем работать с двумя объектами (**ThreeLinkPend_Subs** и **NineLinkPend**), переключаясь между ними по необходимости.

1.5.2.1.1. Создание графического образа сцены

Подготовим образ сцены для девятизвенного маятника. Для этого перенесем ранее созданный графический образ подвеса **Стойка** из модели **ThreeLinkPend_Subs** в модель **NineLinkPend**.

Поскольку оба объекта открыты, мы будем использовать копирование через буфер обмена. Также можно использовать сохранение и последующее чтение элемента объекта из файла. Эти приемы можно использовать для копирования любых элементов объекта.

Выбор активного объекта/Переключение между окнами

Для переключения между объектами, открытыми в программе ввода данных, а также окнами вспомогательных инструментов используется инструмент **Список окон** (пункт меню **Инструменты/Список окон** или *Alt+O*).

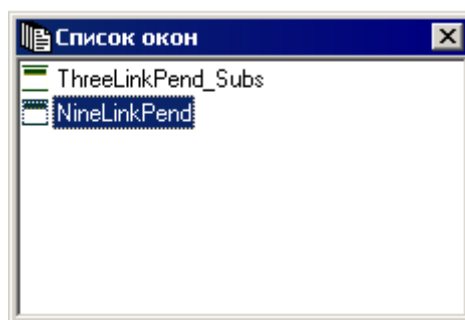


Рис.1.10. Переключение между открытыми объектами в программе ввода данных

Активность объекта также можно установить, активировав окно его описания стандартными средствами операционной системы.

Копирование/Вставка элементов объекта

1. Сделайте активным объект **ThreeLinkPend_Subs** (см. рис.1.10, стр. 39).
2. Перейдите на ветку **Графические объекты** и выберите объект **Стойка**.
3. Скопируйте графический объект **Стойка** в буфер обмена (пункт меню **Правка/Копировать в буфер обмена**).

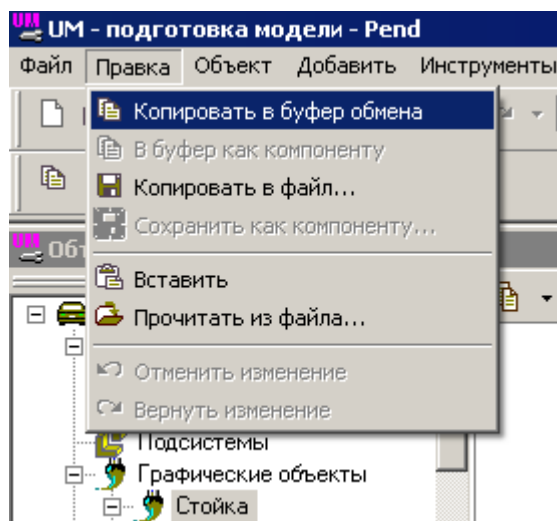
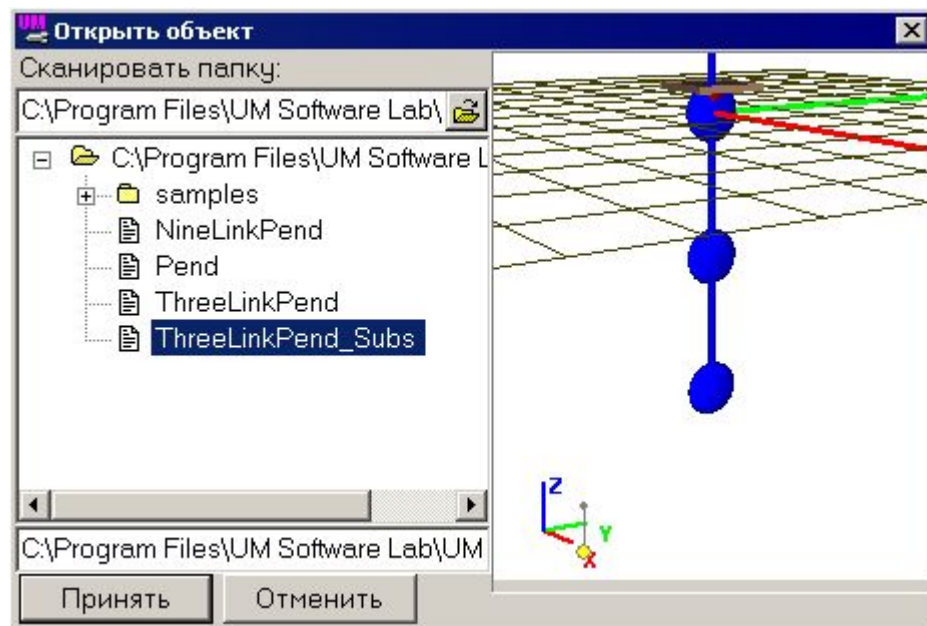
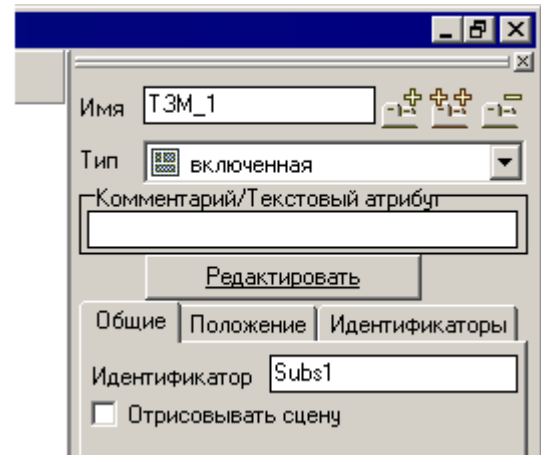
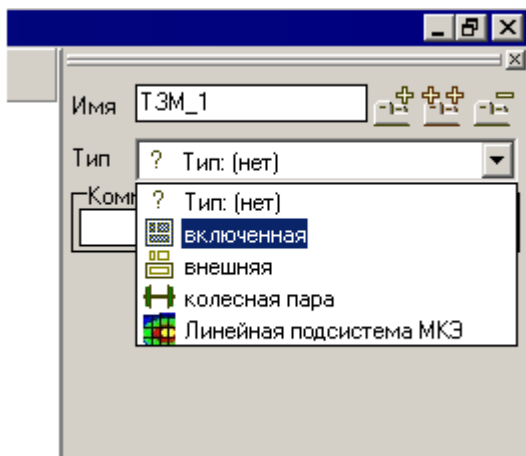


Рис.1.11. Копирование элементов объекта через буфер обмена в программе ввода данных

4. Сделайте активным объект **NineLinkPend** (см. рис.1.10, стр. 39).
5. Вставьте графический объект **Стойка** из буфера обмена (пункт меню **Правка/Вставить**). Объект отобразится на ветке **Графические объекты** и в **Анимационном окне**.
6. Перейдите на ветку **Объект** в списке элементов модели и в поле **Образ сцены** выберите **Стойка**.

1.5.2.1.2. Добавление подсистем

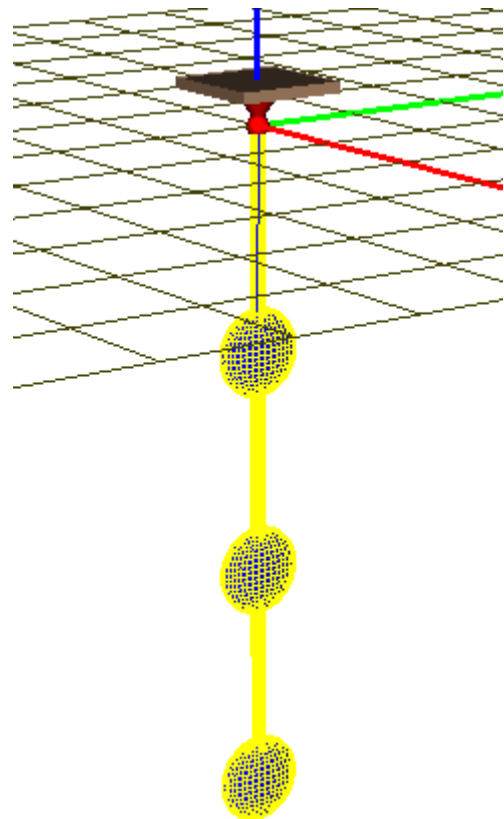
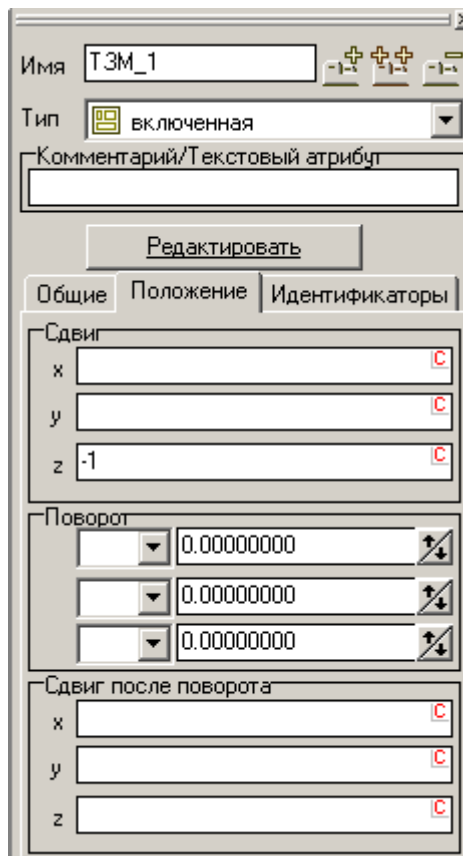
1. Сделайте активным объект **NineLinkPend** (см.рис.1.10, стр. 39).
2. Перейдите на ветку **Подсистемы** в списке элементов модели, добавьте новый элемент (так же как новый графический объект) и переименуйте его в **ТЗМ_1**.
3. В инспекторе выберите **Тип подсистемы – Включенная**. Откроется диалог выбора модели, которую вы бы хотели добавить как включенную подсистему. Нажмите **F5**, чтобы обновить список файлов в открывшейся директории. Выберите объект **ThreeLinkPend_Subs**.



Задание положения подсистемы

Положение добавленной подсистемы определяется наложенными на ее тела кинематическими связями. Тем не менее, для наглядности отображения объекта в программе ввода есть возможность настроить положение добавленной подсистемы.

4. В инспекторе выберите закладку **Положение** и задайте **Сдвиг по Z** на -1, как показано на рисунке.



5. Дважды скопируйте подсистему **ТЗМ_1**.
6. Переименуйте добавленные подсистемы в **ТЗМ_2**, **ТЗМ_3**.
7. Задайте **Сдвиг по Z** для **ТЗМ_2** равным **-4**, а для **ТЗМ_3** равным **-7**.

Добавление подсистем закончено. Перейдем к описанию шарниров, связывающих отдельные подсистемы в многозвенный маятник.

1.5.2.1.3. Создание шарниров

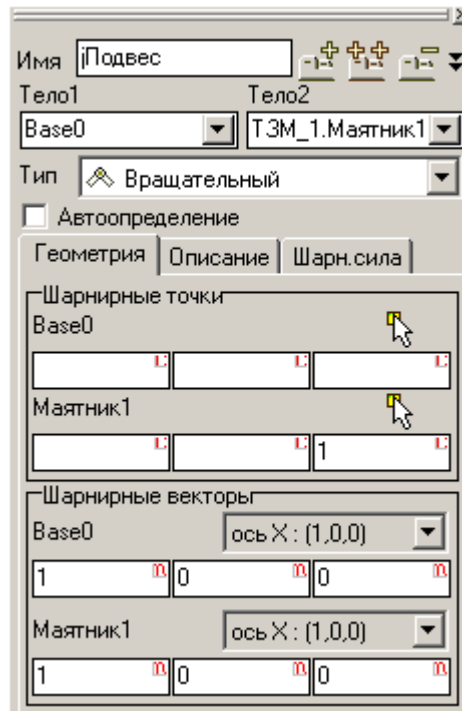
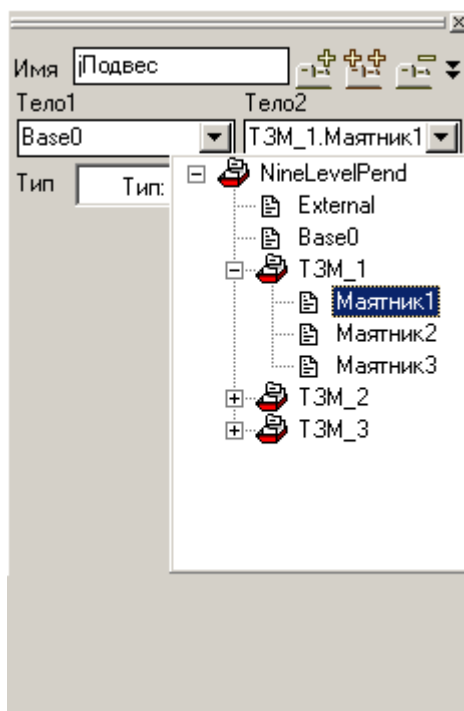
Модель **NineLinkPend** включает три подсистемы. Каждая подсистема состоит из трех маятников, последовательно соединенных между собой вращательными шарнирами, и способно выполнять плоское движение. Для создания девятизвенного маятника необходимо задать вращательные шарниры, описывающие подвес и соединения подсистем.

1. В дереве элементов модели перейдите на ветвь **Шарниры**.

Создание шарнира подвеса

Маятник и стойку связывает вращательный шарнир. Для его создания выполним следующие операции.

2. Создайте новый шарнир и переименуйте его в **jПодвес**.
3. Выберите в качестве **Тела1 – Base0**, а в качестве **Тела2 – ТЗМ_1.Маятник1**.



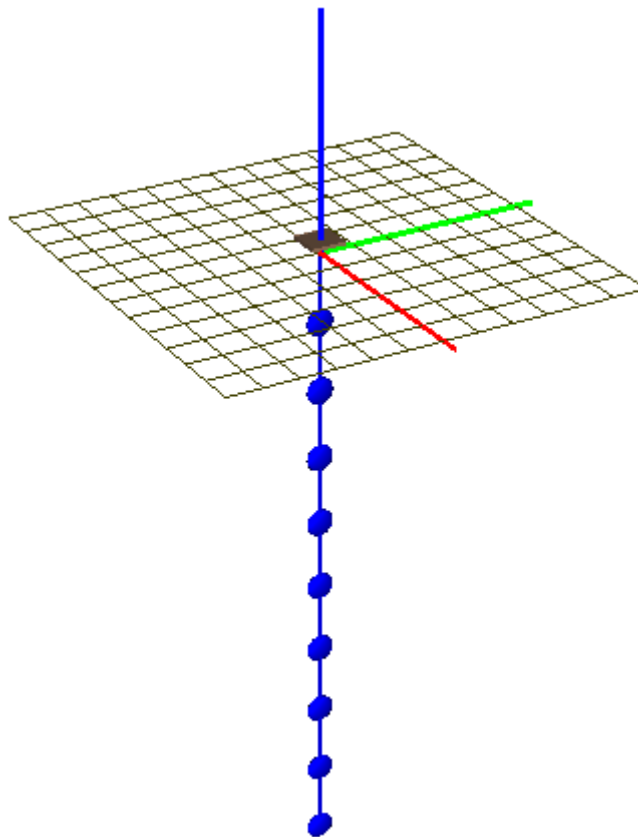
4. Задайте **Тип** шарнира – **Вращательный**.
5. В поле **Шарнирные точки/Маятник1** задайте компоненте **Z** значение **1**.

Создание шарнира между телами подсистем ТЗМ_1 и ТЗМ_2

6. Скопируйте шарнир **jПодвес** и переименуйте его в **jТЗМ_12**.
7. Выберите в качестве **Тела1** – **ТЗМ_1.Маятник3**, а в качестве **Тела2** – **ТЗМ_2.Маятник1**.

Создание шарнира между телами подсистем ТЗМ_2 и ТЗМ_3

8. Скопируйте шарнир **jТЗМ_12** и переименуйте его в **jТЗМ_23**.
 9. Выберите в качестве **Тела1** – **ТЗМ_2.Маятник3**, а в качестве **Тела2** – **ТЗМ_3.Маятник1**.
10. Сохраните модель **NineLinkPend** (пункт меню **Файл/Сохранить**).



Описание составной модели в соответствии со схемой А (см. рис. 1.9, стр. 35) завершено.

Проведем моделирование движения системы и сохраним его результаты.

1.5.2.2. Моделирование движения

Подготовка процесса моделирования

1. Выберите численно-итерационный метод синтеза уравнений (см. п. 1.3.7.1).
2. Запустите программу моделирования.
3. Перейдите в режим моделирования, пункт меню **Анализ/Моделирование**.
4. Перейдите на закладку **Начальные условия | Координаты** инспектора моделирования и задайте начальное значение **0.1** для всех координат.

Обратите внимание на наличие в таблице строк, помеченных знаком **✗**. Эти строки соответствуют координатам в разрезанных шарнирах (по три степени свободы шарнира **jМаятник1** в каждой из подсистем). Начальные значения для этих координат не задаются, а вычисляются автоматически при изменении начальных значений прочих координат.

Инспектор моделирования объекта

Переменные объекта | XVA | Информация | Инструменты

Интегратор | Начальные условия

Координаты | Связи на начальные условия

ninelinkpend.

			Координата	Скорость	Записать значения координат в файл
1.1			0.1	0	jBase0_Маятник1 1a
1.2			0.1	0	jТЗМ_12 1a
1.3			0.1	0	jТЗМ_23 1a
1.10	✗		0.09983341664	0	ТЗМ_1.jМаятник 1c(cut)
1.11	✗		-0.9950041652	0	ТЗМ_1.jМаятник 2c(cut)
1.12	✗		0.1	0	ТЗМ_1.jМаятник 3a(cut)
1.4			0.1	0	ТЗМ_1.jМаятник1_Маятник2 1a
1.5			0.1	0	ТЗМ_1.jМаятник2_Маятник3 1a
1.13	✗		0.98344129641	0	ТЗМ_2.jМаятник 1c(cut)
1.14	✗		-3.8514682262	0	ТЗМ_2.jМаятник 2c(cut)
1.15	✗		0.4	0	ТЗМ_2.jМаятник 3a(cut)
1.6			0.1	0	ТЗМ_2.jМаятник1_Маятник2 1a
1.7			0.1	0	ТЗМ_2.jМаятник2_Маятник3 1a
1.16	✗		2.67172699565	0	ТЗМ_3.jМаятник 1c(cut)
1.17	✗		-6.3192285903	0	ТЗМ_3.jМаятник 2c(cut)
1.18	✗		0.7	0	ТЗМ_3.jМаятник 3a(cut)
1.8			0.1	0	ТЗМ_3.jМаятник1_Маятник2 1a
1.9			0.1	0	ТЗМ_3.jМаятник2_Маятник3 1a


Сообщение dx= 0.1 da= 0.1

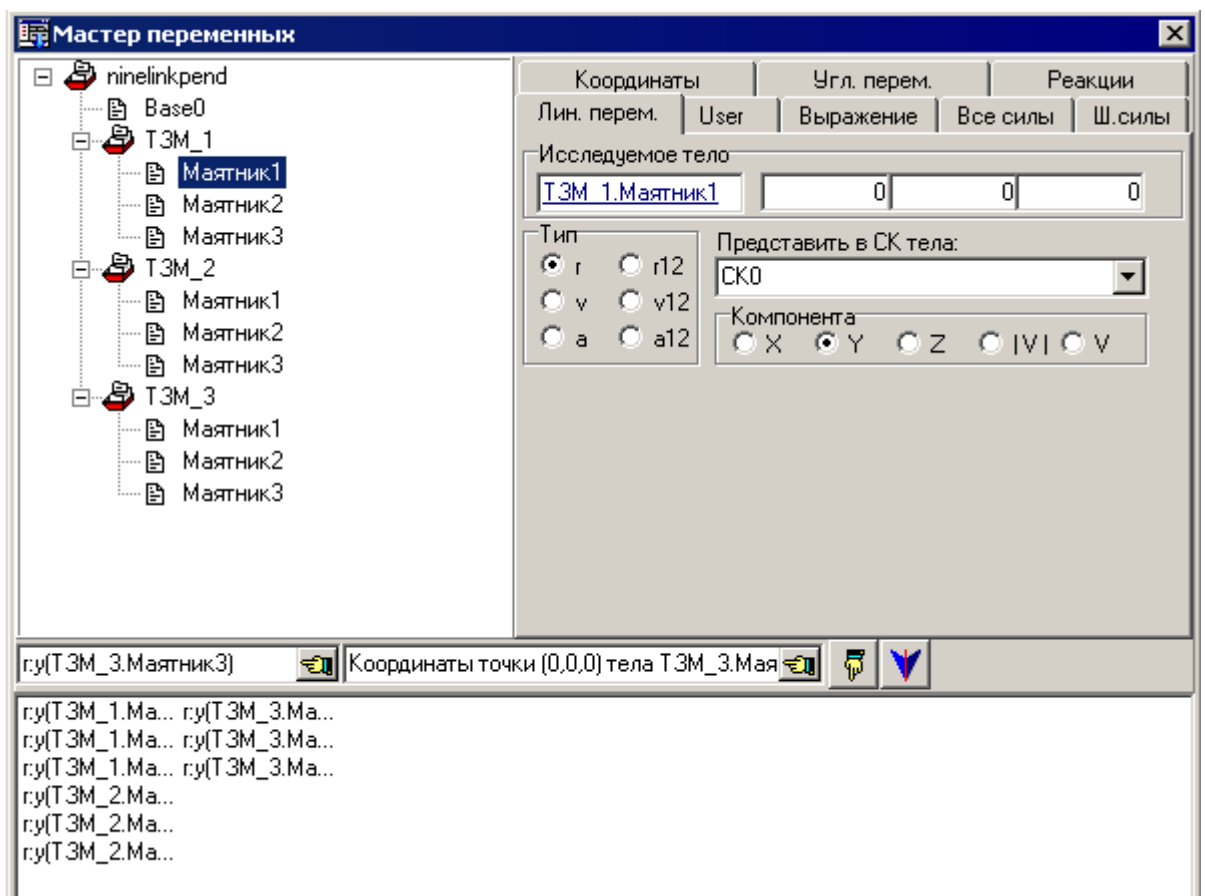
Число степеней свободы = 9

Интегрирование | Сообщение | Выйти

Создание переменных

В **Мастере переменных** создадим переменные – перемещения точек с координатами $(0, 0, 0)$ тел **Маятник1**, **Маятник2**, **Маятник3** подсистем **ТЗМ_1**, **ТЗМ_2** и **ТЗМ_3** в проекции на ось **Y**.

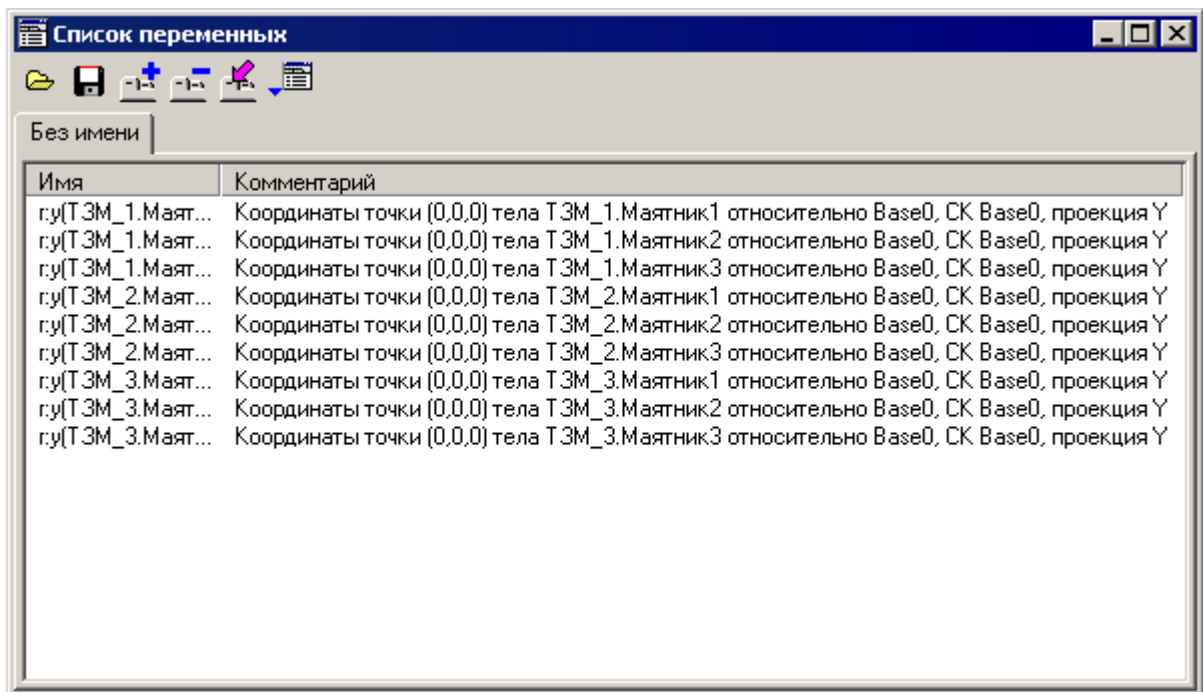
5. Выберите пункт меню **Инструменты/Мастер переменных**.
6. В **Мастере переменных** перейдите на закладку **Лин. перем.** (Линейные перемещения).
7. В дереве элементов модели выберите тело **Маятник1** подсистемы **ТЗМ_1**.
8. В разделе **Компонента** выберите **Y**.
9. Поместите описанную переменную в контейнер, нажав кнопку . В контейнере переменных в нижней части **Мастера переменных** появится переменная **r:y(ТЗМ_1.Маятник1)**.
10. Добавьте аналогичные переменные для остальных восьми тел объекта. После добавления окно **Мастера переменных** будет иметь следующий вид.




Работа с переменными объекта

Сохраним созданные переменные для последующего использования. Для этого воспользуемся инструментом **Список переменных**.

11. Создайте новый список переменных (пункт меню **Инструменты | Список переменных**).
12. Выделите все переменные в контейнере **Мастера переменных** и мышкой перетащите его в окно **Список переменных**.



13. Сохраните список в файл **ninelinkpend.var** с помощью кнопки  на верхней панели окна (по умолчанию будет выбрана директория объекта **NineLinkPend**).

Построение графиков


14. Создайте новое графическое окно (пункт меню **Инструменты | Графическое окно**).
15. Выделите переменные в окне **Список переменных** и перетащите в графическое окно.

Сохранение результатов моделирования

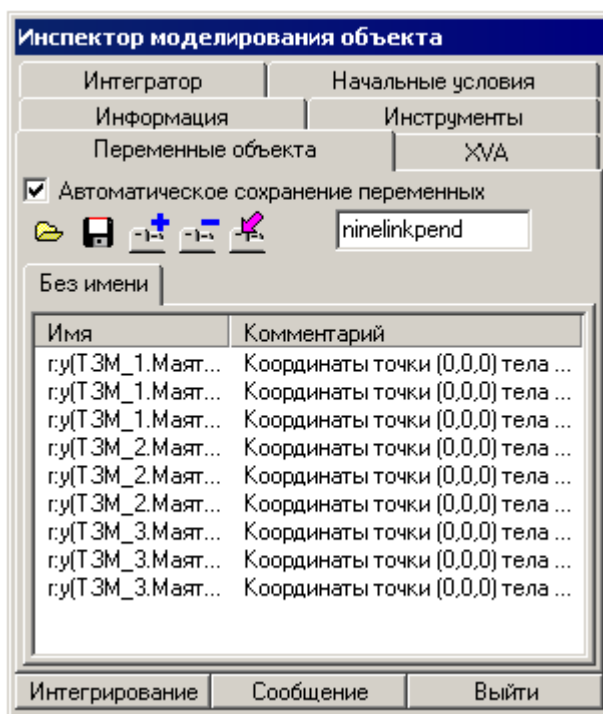
Результаты моделирования – истории изменения во времени созданных переменных – могут быть сохранены в файл рассчитанных переменных с расширением **tgr**. Альтернативой является сохранение графиков переменных в текстовый файл.

В нашем примере мы будем использовать оба способа для сравнения результатов моделирования движения девятизвенного маятника, созданного по схеме с шарнирами, описанными в составном объекте (рис. 1.9, схема А, стр. 35), и по схеме с внешними связями (рис. 1.9, схема Б, стр. 35).

Сохранение результатов в файл рассчитанных переменных

16. Перейдите к **Инспектору моделирования объекта** и откройте закладку **Переменные объекта**.
17. Загрузите  список переменных из файла **ninelinkpend.var** либо выделите переменные в окне **Список переменных** и перетащите его на белое поле окна.
18. Установите галочку в поле **Автоматическое сохранение переменных**.

Теперь в процессе моделирования переменные из списка будут автоматически сохраняться в файл с именем **ninelinkpend.tgr** (по умолчанию будет выбрана директория объекта **NineLinkPend**).



Моделирование движения

Запустим процесс моделирования на **30** секунд.

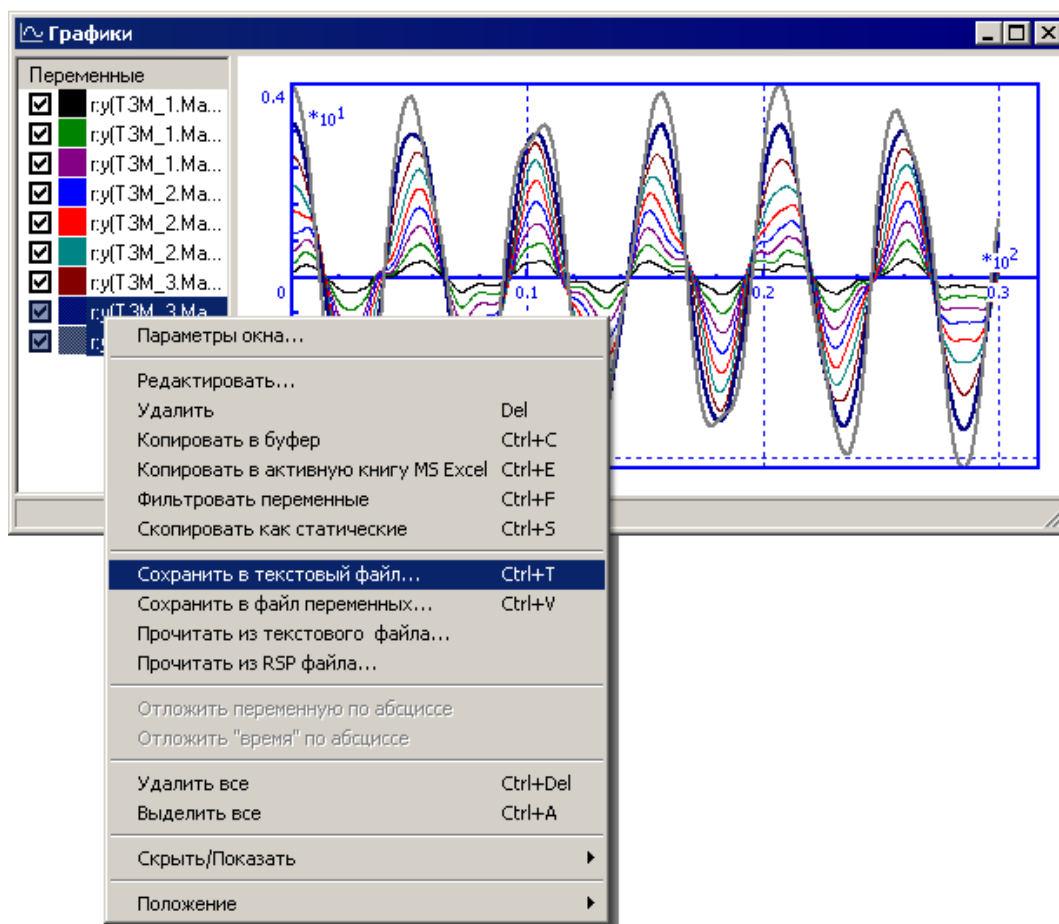
19. Установите значение **30** в поле **Время интегрирования** на закладке **Интегратор | Параметры моделирования** в **Инспекторе моделирования объекта**.

20. Запустите моделирование (кнопка **Интегрирование** в окне **Инспектора моделирования объекта**). Для ускорения процесса интегрирования можно свернуть анимационное окно.

В графическом окне построятся графики переменных, см. рис. ниже.

Копирование графиков переменных в текстовый файл

21. В списке **Переменные** графического окна наведите курсор на две последние переменные и щелкните правой клавишей мыши. В открывшемся контекстном меню выберите **Сохранить в текстовый файл...** Сохраните данные в файл **result.txt** (по умолчанию будет выбрана директория объекта **NineLinkPend**).



Сохраненные графики можно загружать в графические окна как *статические* переменные – переменные, которые не будут пересчитываться в процессе интегрирования.

1.5.2.3. Использование внешних связей

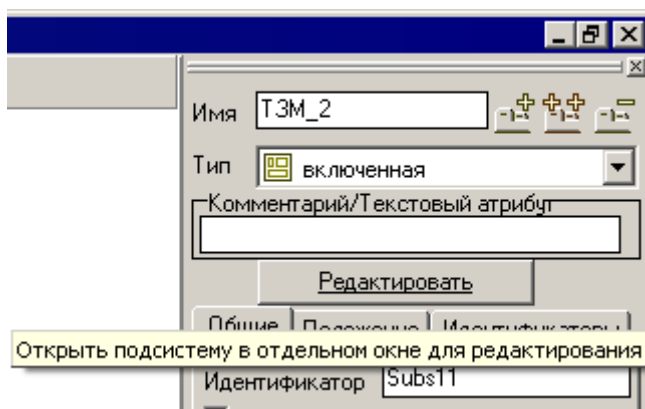
При создании модели девятизвенного маятника мы описывали шарниры, соединяющие тела различных подсистем, с помощью шарниров, относящихся к составному объекту (рис. 1.9, схема А, стр. 35).

Теперь мы воспользуемся альтернативным способом и опишем кинематическую схему составного объекта с помощью внешних связей (рис. 1.9, схема Б, стр. 35). Для этого нам потребуется внести ряд изменений в описание составного объекта **NineLinkPend**, а также его подсистем.

Затем мы выполним моделирование движения отредактированной модели и сравним результаты с полученными ранее.

Редактирование включенных подсистем

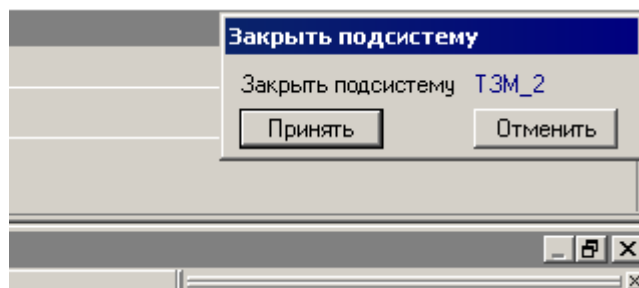
1. Закройте программу моделирования и вернитесь в программу ввода данных.
2. Перейдите на ветку **Подсистемы** и выберите элемент **ТЗМ_2**.



3. В инспекторе нажмите кнопку **Редактировать**.

В **Анимационном окне** откроется объект **ТЗМ_2**. Мы можем редактировать его как обычный объект. Во время работы с включенной подсистемой описание составного объекта недоступно.

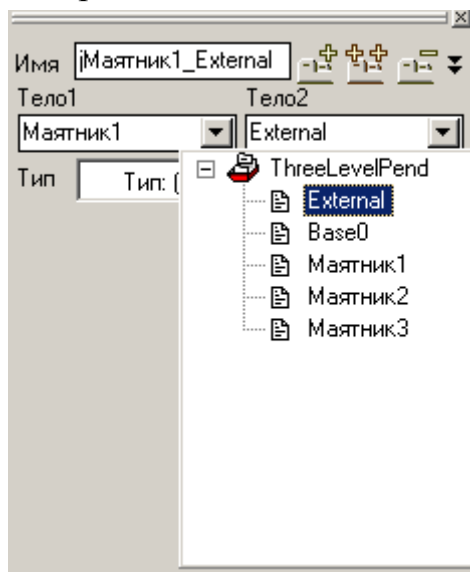
Обратите внимание на окно **Закреть подсистему** в правом верхнем углу экрана. Это окно используется для перехода к составному объекту с сохранением или без сохранения внесенных изменений.



Добавление в подсистемы шарниров с внешним телом External

Добавим в подсистему ТЗМ_2 шарнир, связывающие тела **Маятник1** со специальным телом **External**.

4. Перейдите на ветку **Шарниры** и добавьте новый шарнир.
5. Задайте **Маятник1** в качестве первого тела, и **External** в качестве второго, как показано на рисунке.
6. Задайте **Тип** шарнира – **Вращательный**.
7. Задайте значение **1** для координаты **Z** тела **Маятник1**.




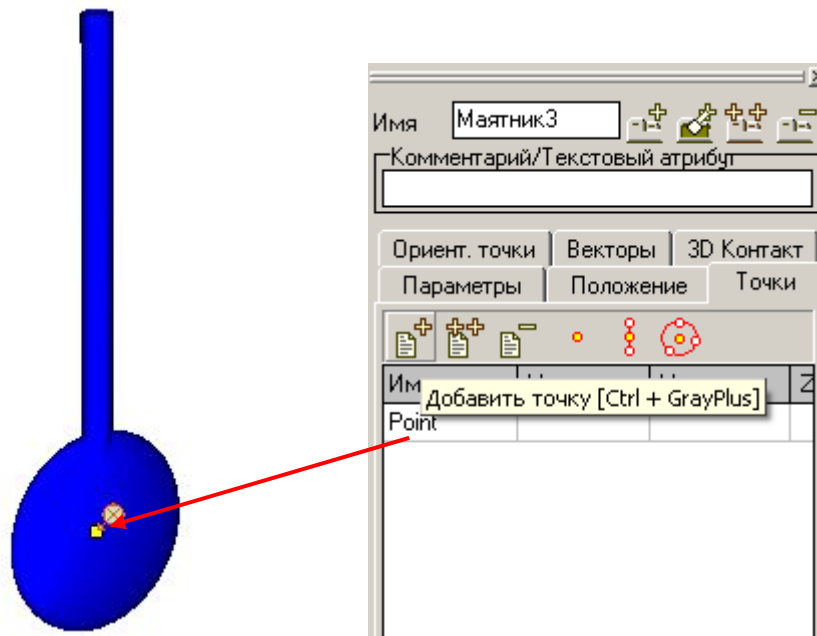
8. Скопируйте добавленный шарнир **jМаятник1_External** в буфер обмена (пункт меню **Правка/Копировать в буфер обмена**, см. рис.1.11 стр.40).
9. Нажмите **Принять** в окне **Закреть подсистему**. Объект **ТЗМ_2** будет закрыт, модель **NineLinkPend** станет активной.
10. Запустите режим редактирования подсистемы **ТЗМ_3** (выберите подсистему **ТЗМ_3** и нажмите **Редактировать**).
11. Вставьте скопированный шарнир из буфера обмена (пункт меню **Правка/Вставить**). Новый элемент отобразится на ветке **Шарниры**.
12. Выберите добавленный шарнир и укажите в поле **Тело1** инспектора тело **Маятник1**.
13. Нажмите **Принять** в окне **Закреть подсистему**. Объект **ТЗМ_3** будет закрыт, активным станет модель **NineLinkPend**.

Добавление в подсистемы шарниров с телом **External** завершено.

Добавление точек связи

Точки связи необходимы для описания **Внешних связей** между подсистемами. Для описания шарниров между телом **Маятник1** подсистемы **ТЗМ_2** и телом **Маятник3** подсистемы **ТЗМ_1**, а также телом **Маятник1** подсистемы **ТЗМ_3** и телом **Маятник3** подсистемы **ТЗМ_2**, необходимо для тел **Маятник3** создать точки связи, соответствующие положению шарниров в системе координат тела **Маятник3**.

14. Запустите режим редактирования подсистемы **ТЗМ_1** (выберите подсистему **ТЗМ_1** и нажмите **Редактировать**).
15. Перейдите на ветку **Тела** и выделите тело **Маятник3**.
16. В инспекторе перейдите на закладку **Точки** и добавьте новую точку (кнопка ). По умолчанию будет добавлена точка связи с нулевыми координатами относительно системы координат тела. Координаты точки отображаются в таблице (пустые поля соответствуют нулевым значениям), а ее положение выделяется желтым маркером на графическом образе тела. Добавленная точка совпадает с осью вращения присоединяемого маятника.



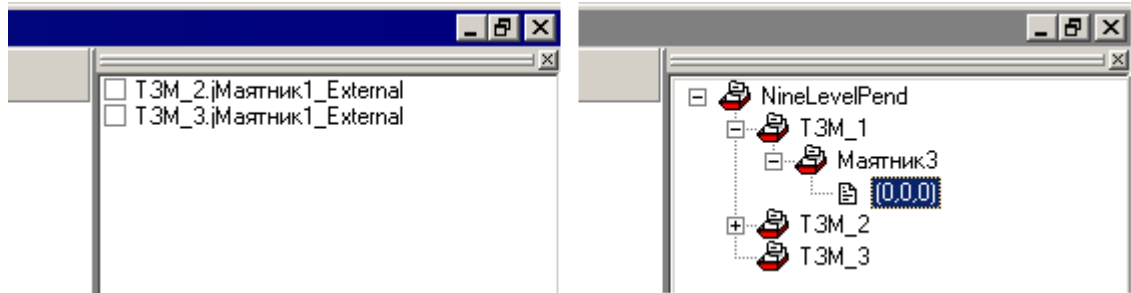
17. Нажмите **Принять** в окне **Заккрыть подсистему**.
18. Добавьте точку связи для тела **Маятник3** подсистемы **ТЗМ_2**.
19. Нажмите **Принять** в окне **Заккрыть подсистему**. Объект **ТЗМ_2** будет закрыт, активным станет модель **NineLinkPend**.

Редактирование подсистем завершено.

Задание внешних связей

20. Перейдите на ветку **Внешние связи** дерева элементов модели **NineLinkPend**.

В инспекторе отобразится список, включающий все шарниры и силовые элементы в объекте и его подсистемах, для которых в качестве второго тела указано тело **External**. В нашем случае это шарниры **jМаятник1_External** подсистем **ТЗМ_2** и **ТЗМ_3**.



21. Двойным щелчком мыши установите метку слева от элемента **ТЗМ_2.jМаятник1_External**. При этом появится дерево тел составного объекта, в котором нижний уровень это уровень точек связи тел.

22. Выберите точку **(0, 0, 0)** тела **Маятник3** подсистемы **ТЗМ_1**.

23. Двойным щелчком мыши установите метку слева от элемента **ТЗМ_3.jМаятник1_External**.

24. Выберите точку **(0, 0, 0)** тела **Маятник3** подсистемы **ТЗМ_2**.

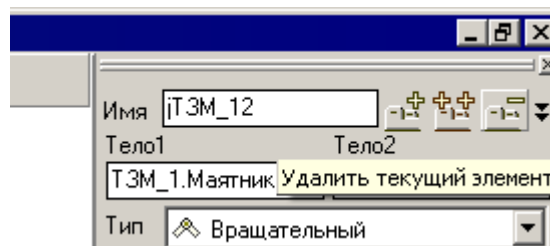
Описание внешних связей завершено.

Удаление избыточных шарниров

Добавленные ранее шарниры составного объекта **jТЗМ_12** и **jТЗМ_23** дублируют шарниры, созданные с использованием внешних связей. Удалим их.

25. Перейдите на ветку **Шарниры** дерева элементов модели **NineLinkPend**.

26. Удалите шарниры **jТЗМ_12** и **jТЗМ_23**.

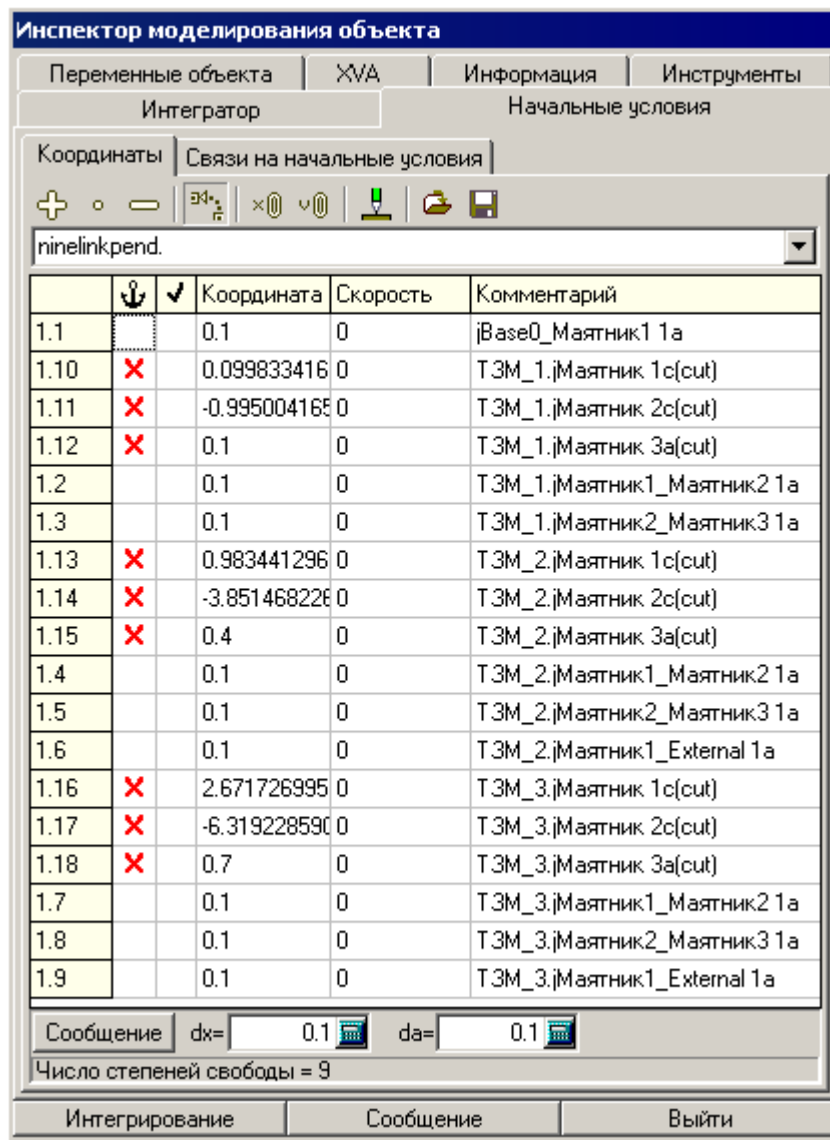


Сравнение результатов

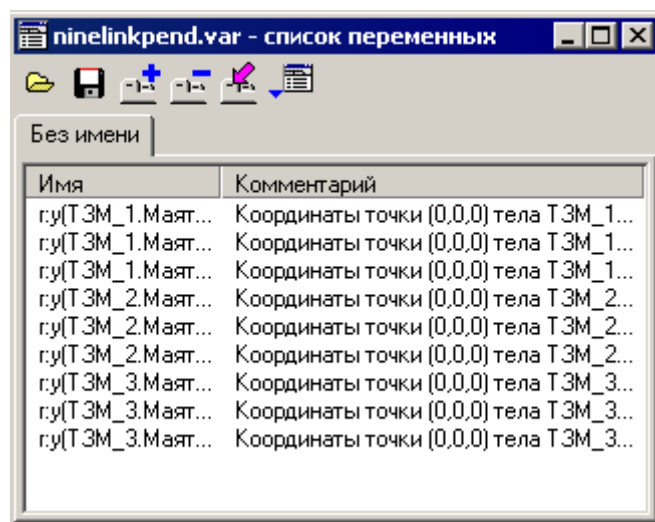
Сравним результаты моделирования движения многозвенного маятника, полученные с использованием отредактированной модели с ранее полученными результатами.

Для этого построим графики перемещения центра масс двух последних в цепочке маятников в проекции на ось Y и сравним их с данными из файла рассчитанных переменных **ninelinkpend.tgr**, а также графиками, сохраненными ранее в текстовом файле **results.txt**.

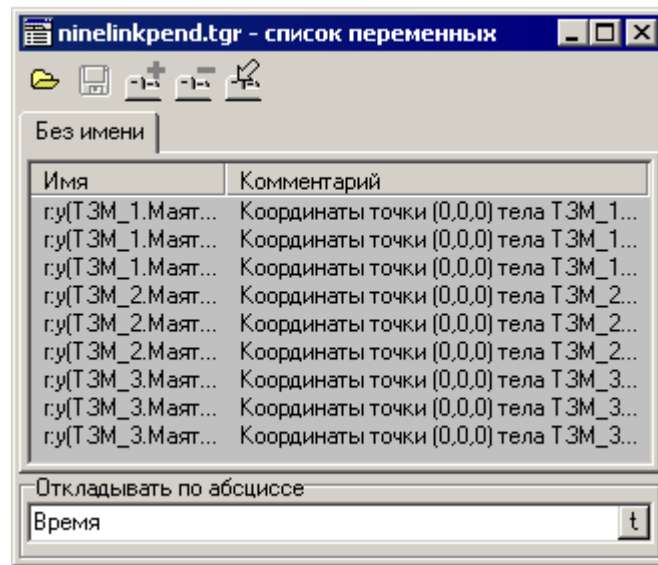
27. Запустите программу моделирования. Если включена опция автосохранения **Конфигурация** (см. закладка **Автосохранение** диалогового окна **Опции интегратора**, вызываемого с помощью пункта меню **Инструменты | Настройки...**), то откроются ранее созданные анимационное и графическое окно с переменными.
28. Перейдите в режим моделирования, пункт меню **Анализ/Моделирование**.
29. Перейдите на закладку **Начальные условия | Координаты**. Если опция автосохранения **Начальные условия** включена, то загрузятся ранее введенные начальные условия, если нет, то задайте значение **0.1** для всех неразрезанных шарниров, как показано на рисунке.



- 30.Создайте еще одно графическое окно (пункт меню **Инструменты** | **Графическое окно**).
- 31.Создайте новый список переменных (пункт меню **Инструменты** | **Список переменных**) и откройте в нем файл переменных **ninelinkpend.var**.



32. Перетащите две последние переменные из окна **Список переменных** в новое графическое окно. Графики, соответствующие добавленным переменным, будут строиться в процессе интегрирования.
33. С помощью пункта меню **Инструменты | Список рассчитанных переменных** откройте список ранее рассчитанных переменных **ninelinkpend.tgr**.



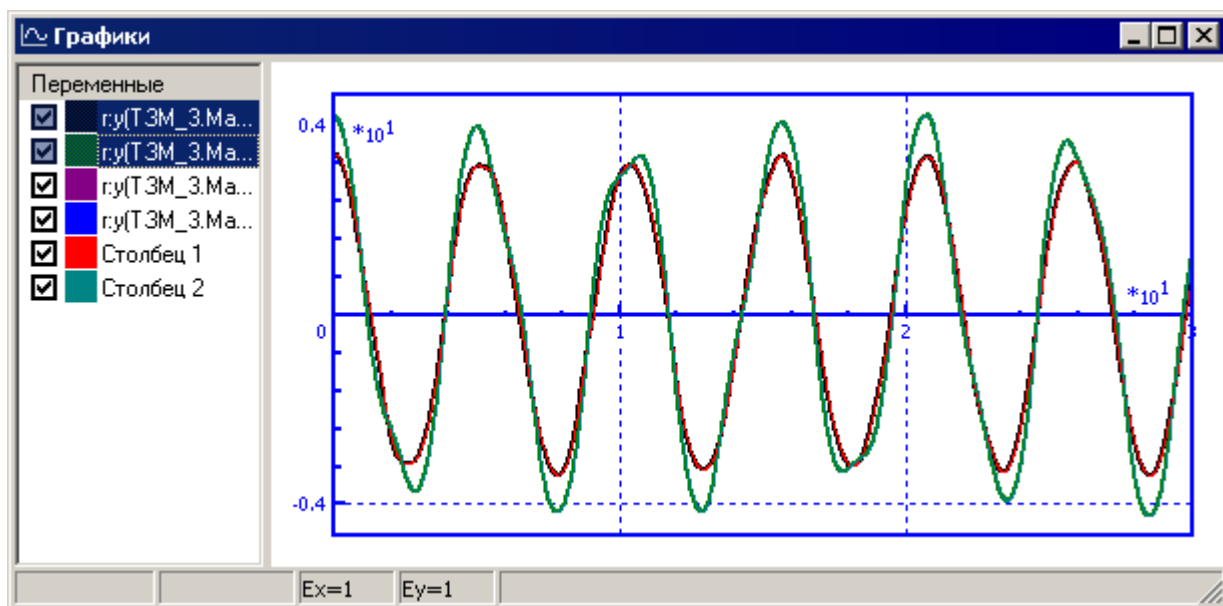
34. Перетащите две последние переменные из окна **Список переменных** в графическое окно. Добавленные *статические* переменные сразу отобразятся в виде графиков.
35. Запустите моделирование (окно **Инспектор моделирования**, кнопка **Интегрирование**) и дождитесь его завершения.
36. Наведите курсор на поле **Переменные** графического окна и щелкните правой клавишей мыши. В открывшемся контекстном меню выберите **Прочитать из текстового файла ...** и загрузите данные из файла **results.txt**.

В графическом окне теперь отображаются три пары графиков, соответствующих перемещениям центров тяжести двух последних в цепочке маятников.

Как видно, рассчитанные в процессе интегрирования отредактированной модели графики (первые два графика) полностью совпадают с ранее полученными результатами.

При выделении переменных в контейнере графического окна их графики отрисовываются линиями двойной толщины. Выделяя попеременно каждую

из трех пар переменных можно убедиться в том, что их графики полностью совпадают.



2. Груз на пружине

2.1. Что мы изучим

В этом уроке мы научимся добавлять силы, задавать движение тел как функцию времени, познакомимся с понятием параметризации модели. Изучим применение *линейного анализа* для нахождения положения равновесия, частот и форм колебаний. А также изучим спектральный состав выходного сигнала с помощью окна *Статистика*.

2.2. Описание модели

В этом уроке мы рассмотрим пример моделирования прямолинейных свободных и вынужденных колебаний тела с учетом демпфирования. В конце данного урока мы создадим модель, представленную на рис. 2.1. Модель включает два твердых тела *Подвес* и *Груз*, два поступательных шарнира и упруго-диссипативную силу между телами.

Готовую модель вы можете найти в каталоге `{um_root}\samples\tutorial\oscillator` или скачать по адресу:

<http://www.umlab.ru/download/60/oscillator.zip>

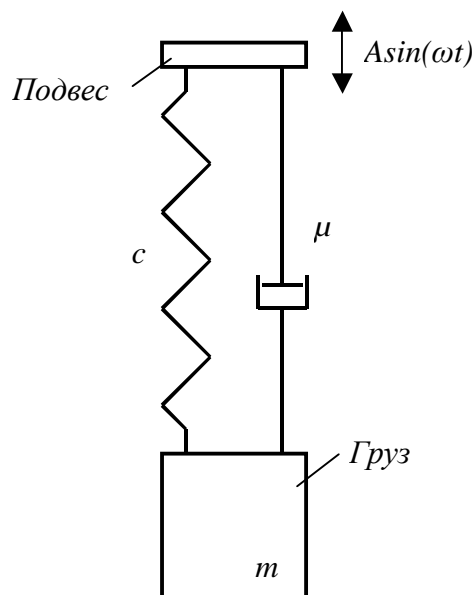


Рис. 2.1. Расчетная схема

2.3. Создание модели

2.3.1. Запуск UM Input и создание нового объекта моделирования

Для запуска программы описания моделей **UM Input**:

1. Выберите кнопку **Пуск**.
2. Выберите **Программы/Universal Mechanism 5.0/UM Input**.

Для создания новой модели:

3. Выберите пункт меню **Файл/Новый объект**.

Выберем численно-итерационный метод синтеза уравнений движения.

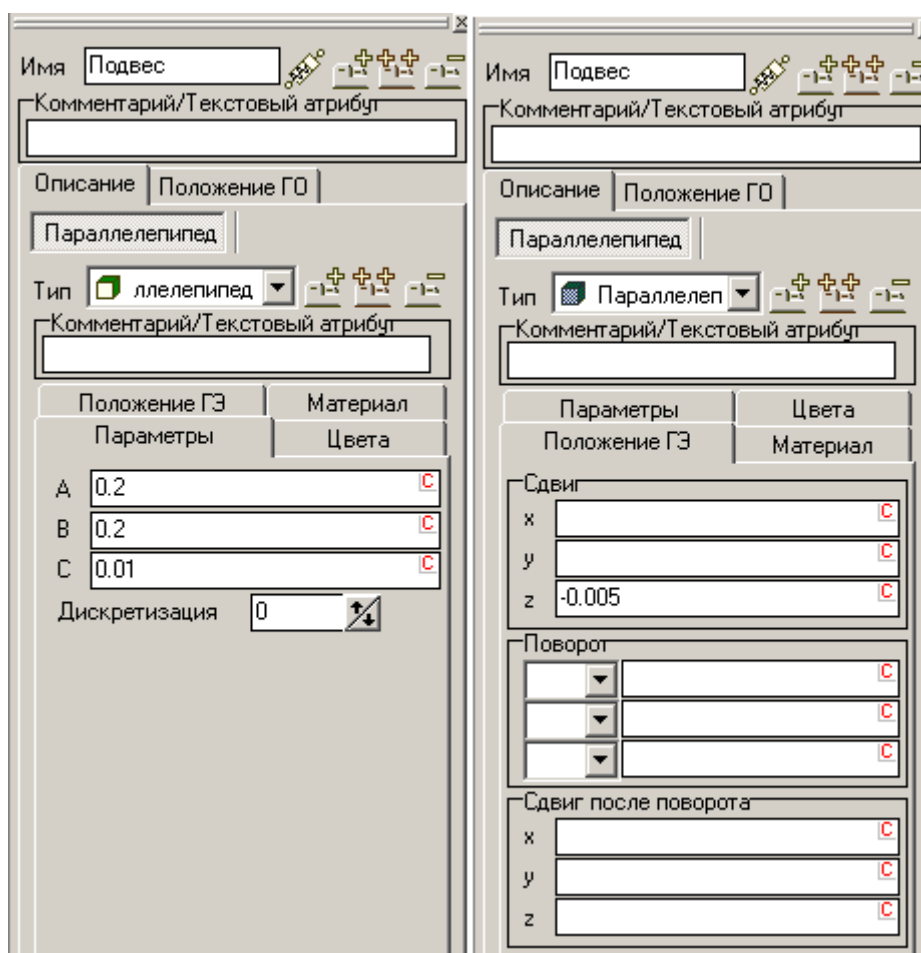
4. В списке элементов модели выберите поле **Объект**. Далее, в инспекторе данных в поле **Синтез уравнений** выберите **Численно-итерационный**.

2.3.2. Создание графических образов

Подвес

В качестве точки подвеса пружины и демпфера создадим тонкий прямоугольник.

1. Создайте новый графический объект.
2. Назначьте ему имя – **Подвес**.
3. В графический объект добавьте новый графический элемент - **Параллелепипед**.
4. Назначьте параметры **Параллелепипеда**, как показано на рисунке ниже.
5. Перейдите на закладку **Цвета** и назначьте **синий** в качестве **диффузного и отраженного** цветов.

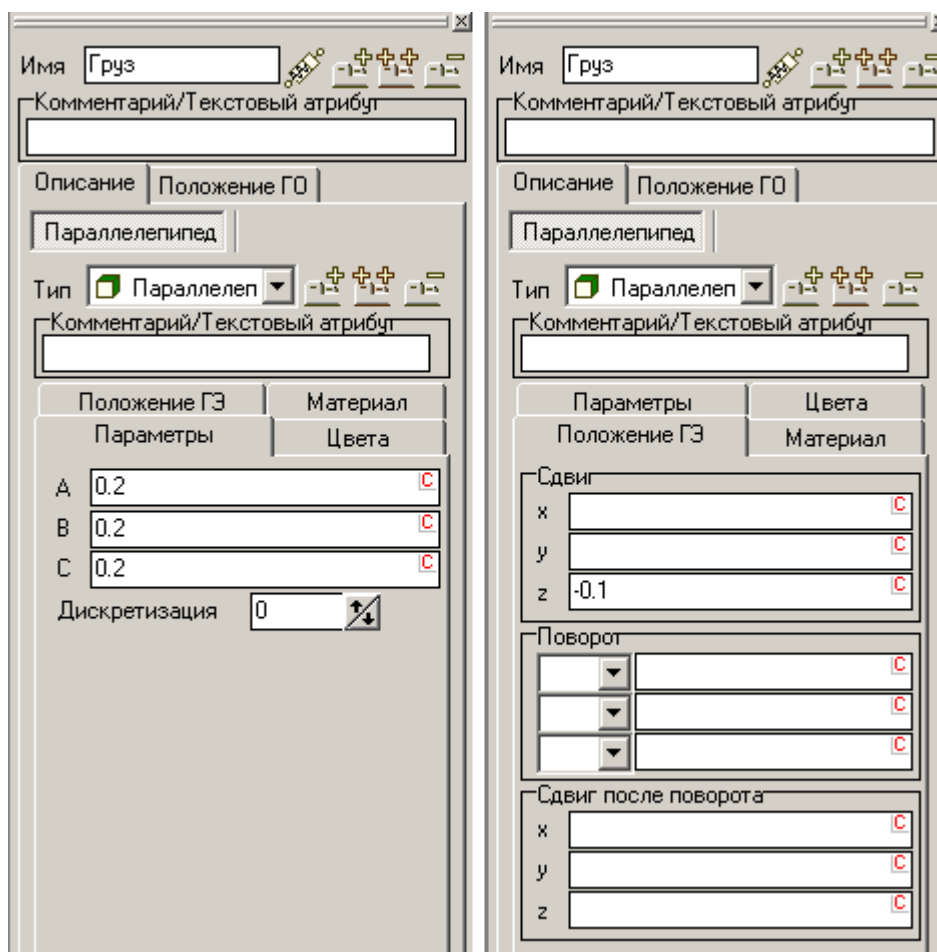


Груз

Груз на пружине изобразим как куб со стороной **0.2** м.

1. Создайте новый графический объект.
2. Назначьте ему имя – **Груз**.
3. В графический объект добавьте новый графический элемент - **Параллелепипед**.
4. Назначьте параметры **Параллелепипеда**, как показано на рисунке ниже.

Перейдите на закладку **Цвета** и назначьте **красный** в качестве **диффузного** и **отраженного** цветов.



Пружина

Создадим графический образ пружины.

1. Создайте новый графический объект.
2. Назначьте ему имя – **Пружина**.
3. В графический объект добавьте новый графический элемент – **Пружина**.
4. Назначьте параметры **Пружины**, как показано на рисунке ниже.
5. Перейдите на закладку **Цвета** и назначьте **желтый** в качестве **диффузного** и **красный** в качестве **отраженного** цветов.

Имя: Пружина

Комментарий/Текстовый атрибут

Описание: Пружина

Положение ГО

Тип: Пружина

Комментарий/Текстовый атрибут

Положение ГЗ: Левая Правая

Материал

Цвета

Радиус: 0.07

Высота: 1

d прутка: 0.02

Число витков: 5

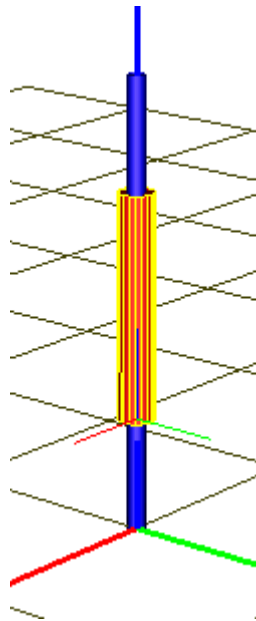
Дискретизация витка: 20

Дискретизация прутка: 10

Демпфер

Создадим графический образ демпфера.

1. Создайте новый графический объект.
2. Назначьте ему имя – **Демпфер**.
3. В графический объект добавьте новый графический элемент – **Конус**.
4. Назначьте **Конусу** следующие параметры:
 $R2 = 0.02$;
 $R1 = 0.02$;
 $h = 1$.
5. Выберите синий цвет в качестве диффузного.
6. Добавьте еще один **конус** с параметрами: $R2 = 0.04$; $R1 = 0.04$; $h = 0.5$. На закладке **Положение ГЭ** в поле **Сдвиг/z** установите **0.25**. Установите для графического элемента красный диффузный цвет.

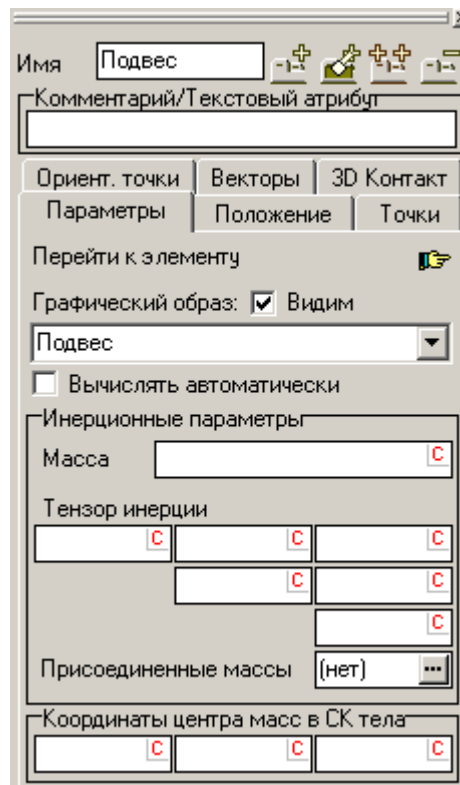


2.3.3. Создание тел

Подвес

Создадим новое твердое тело – подвес.

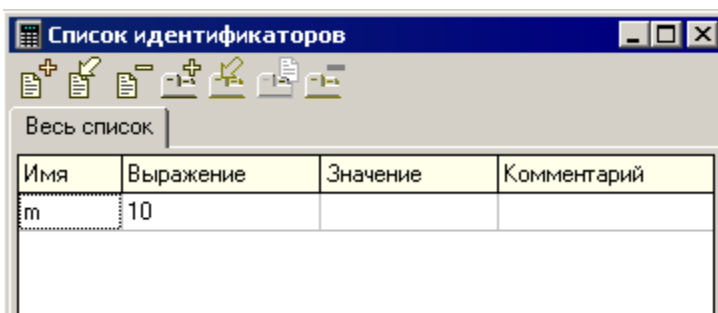
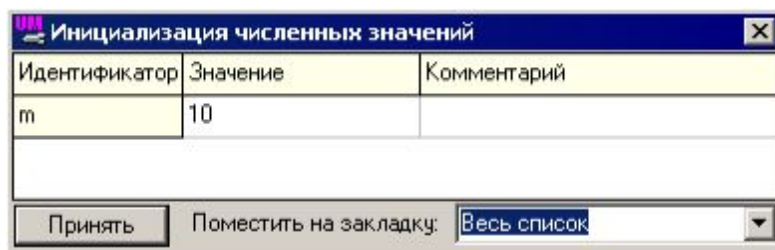
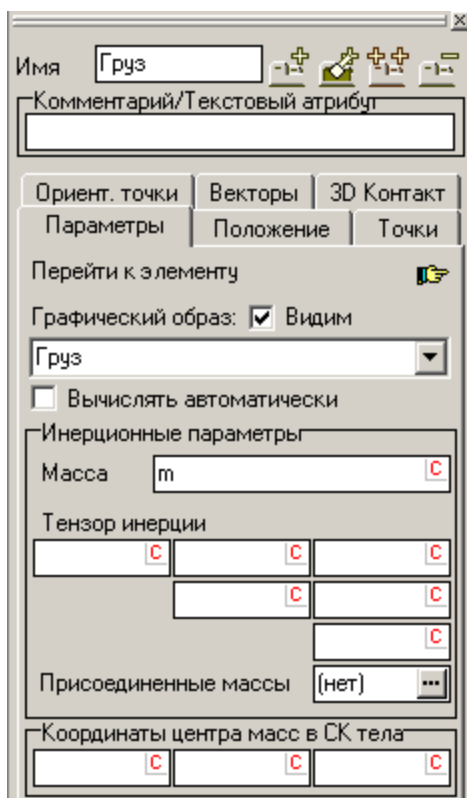
1. Добавьте новое твердое тело.
2. Переименуйте его в **Подвес**.
3. Выберите графический образ для тела. В списке **Графический образ** выберите **Подвес**.



Груз

Создадим новое твердое тело – груз. Массу груза вместо численного значения массу зададим параметром (идентификатором) **m**. Параметризация модели позволяет исследовать динамику систем при различных значениях параметров, не синтезируя уравнения движения заново каждый раз.

1. Добавьте новое твердое тело.
2. Переименуйте его в **Груз**.
3. Выберите графический образ для тела. В списке **Графический образ** выберите **Груз**.
4. В поле **Масса** введите **m** и нажмите **Enter**. Появится диалоговое окно **Инициализация численных значений**.
5. В поле **Значение** введите **10**. Нажмите **Enter**.
6. В списке параметров модели, слева в окне конструктора, появится первый параметр модели – **m**.



2.3.4. Создание шарниров

Шарнир для подвеса

В нашей задаче подвес совершает гармонические вертикальные колебания с амплитудой A . Сейчас мы добавим в модель поступательный шарнир между базой и подвесом, а координату зададим функцией времени.

1. Выберите тело **Подвес**.
2. Нажмите кнопку **Перейти к элементу** и выберите пункт **Создать шарнир**.
3. В появившемся списке типов шарниров выберите **Поступательный**, см. рис. 2.2, слева. Будет создан новый шарнир указанного типа.
4. Установите параметры шарнира, как показано на рис. 2.2, справа.

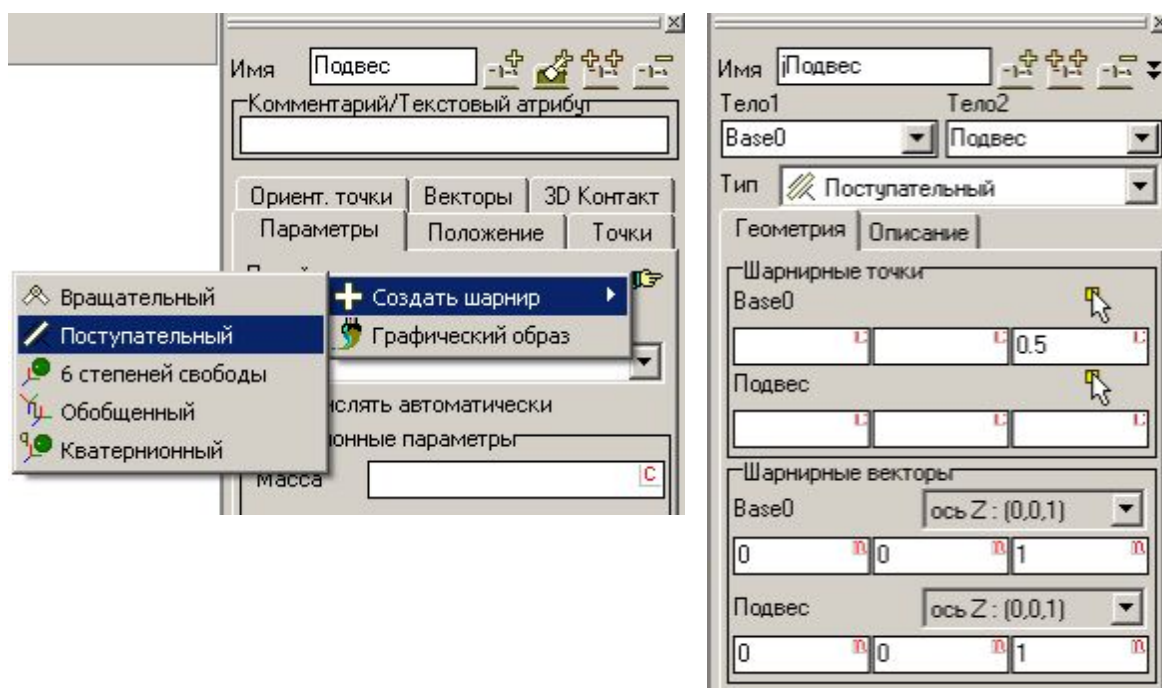


Рис. 2.2. Добавление поступательного шарнира

5. Перейдите на закладку **Описание**.
6. Включите флажок **Заданная функция времени**.
7. В поле **Тип задания функции** выберите **Выражение** и в качестве выражения введите $a \cdot \sin(\omega \cdot t)$, см. рис. 2.3.

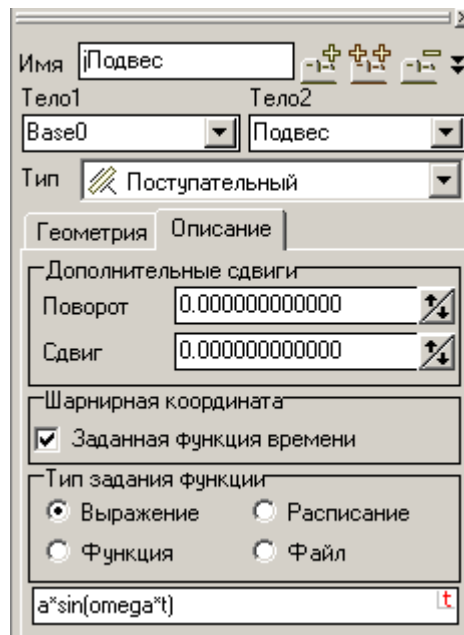
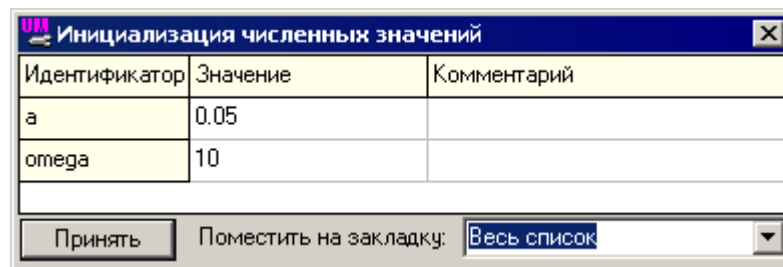


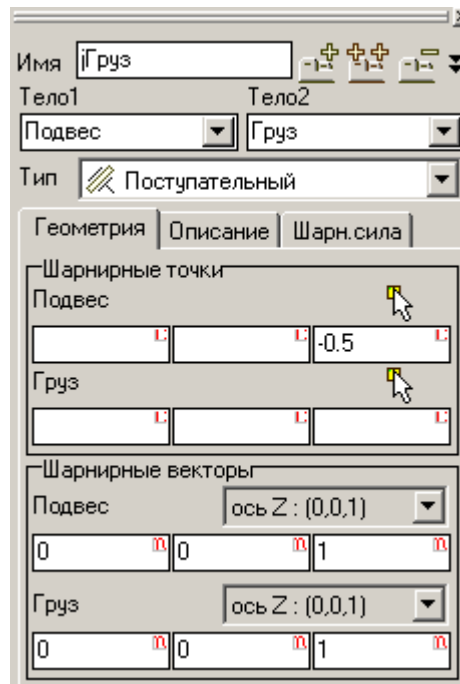
Рис. 2.3. Координата в шарнире как функция времени

8. В появившемся окне инициализации численных значений введите $a = 0.05$ и $\omega = 10$.



Шарнир для груза

1. Выберите тело **Груз**.
2. Нажмите кнопку **Перейти к элементу** и выберите пункт **Создать шарнир**.
3. В появившемся списке типов шарниров выберите **Поступательный**.
4. В качестве первого тела для шарнира выберите **Подвес** вместо **Base0**.
5. Остальные параметры шарнира установите, как показано на рисунке ниже.



2.3.5. Описание сил

Введем упруго-диссипативную силу, действующую между подвесом и грузом. Жесткость пружины обозначим как **c**, коэффициент диссипации как **mu**. Длину нерастянутой пружины обозначим **l0**.

1. Выберите шарнир **jГруз**.
2. Перейдите на закладку **Шарн. сила** (Шарнирная сила).
3. В поле **Шарнирная сила** выберите **Линейный**.

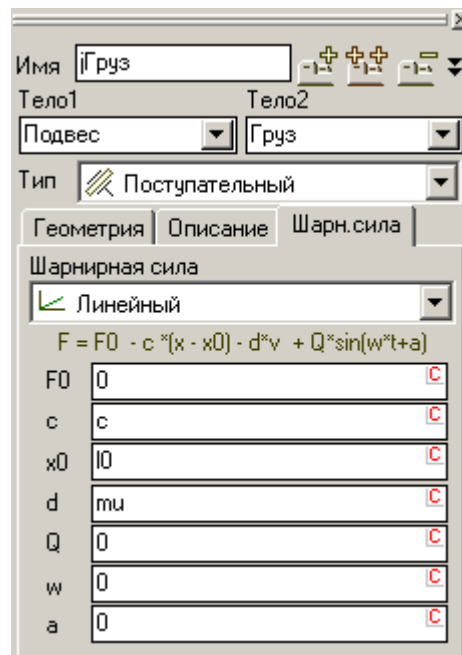


Рис. 2.4. Упругая и диссипативная шарнирные силы

4. В поле **c** (коэффициент жесткости) введите **c**, в поле **x0** введите **l0**, а в поле **d** (коэффициент диссипации) введите **mu**. При инициализации численных значений положите **c = 250**, **l0 = 0.4**, **mu = 5**.

2.3.6. Визуализация пружины и демпфера

Сейчас наш объект полностью описан с точки зрения механики. Однако внешний вид модели оставляет желать лучшего – пружина и демпфер не видны, см. рис. 2.5, слева. Чтобы визуализировать пружину и демпфер, мы введем в модель биполярные силы, которым назначим графический образ демпфера и пружины. Значения же сил положим равными нулю. Введенные таким образом силы никак не отразятся на динамике системы, однако позволят нам визуализировать пружину и демпфер, см. рис. 2.5, справа.

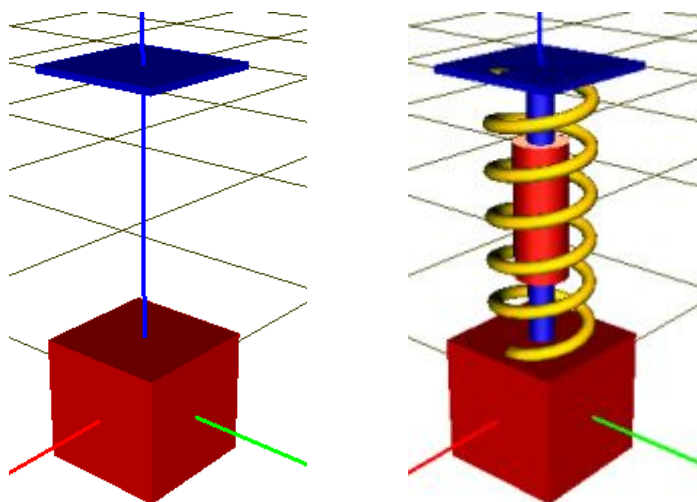


Рис. 2.5. Визуализация сил

Замечание. Упругую и диссипативную силы, описанные как шарнирные, можно было бы описать непосредственно как биполярные. В этом случае не пришлось бы вводить фиктивные силы для изображения пружины и демпфера. Однако дело в том, что рассматриваемый нами идеальный случай¹ позволяет смоделировать ситуацию, в которой длина пружины и демпфера будет равна нулю (тело поднимется на такую высоту, что точки прикрепления силовых элементов окажутся на одном уровне). В этом случае происходит вырождение биполярных элементов – при нулевой длине биполярной силы невозможно определить ее направление. Шарнирные силы не имеют такого вырождения. Именно поэтому мы и использовали их для моделирования данной механической системы.

¹ Силовая характеристика реальной пружины – нелинейная. При смыкании витков пружины ее жесткость резко возрастает.

1. Итак, перейдите на элемент **Биполярные силы** в списке элементов модели.
2. Добавьте две биполярные силы. Параметры сил укажите в соответствии с рис. 2.6.

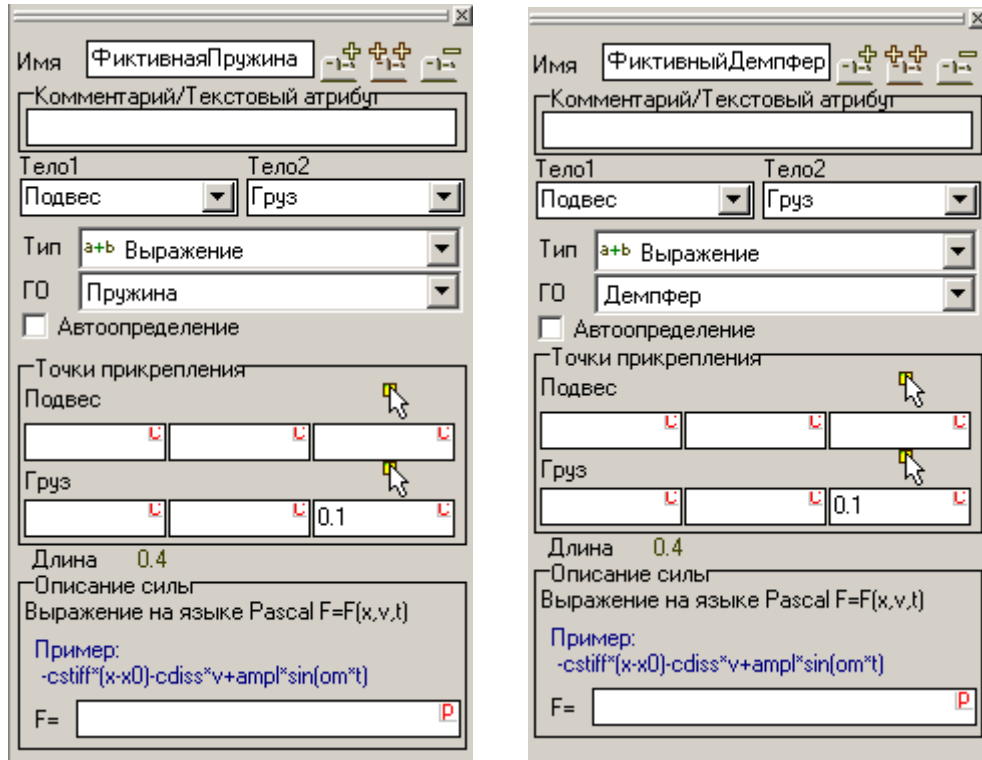


Рис. 2.6. Фиктивные биполярные силы

2.3.7. Дополнительные параметры

Возможности параметризации в УМ позволяют выразить одни параметры модели через другие. Покажем это на примере нашей модели. Введем в модель два новых параметра – определим точные значения собственной частоты колебаний груза и величину критического коэффициента демпфирования¹.

Собственную частоту колебаний груза определим по следующей формуле:

$$k = \sqrt{\frac{c}{m}}, \text{ где}$$

k – собственная частота колебаний, рад/с;


c – коэффициент жесткости пружины, Н/м;

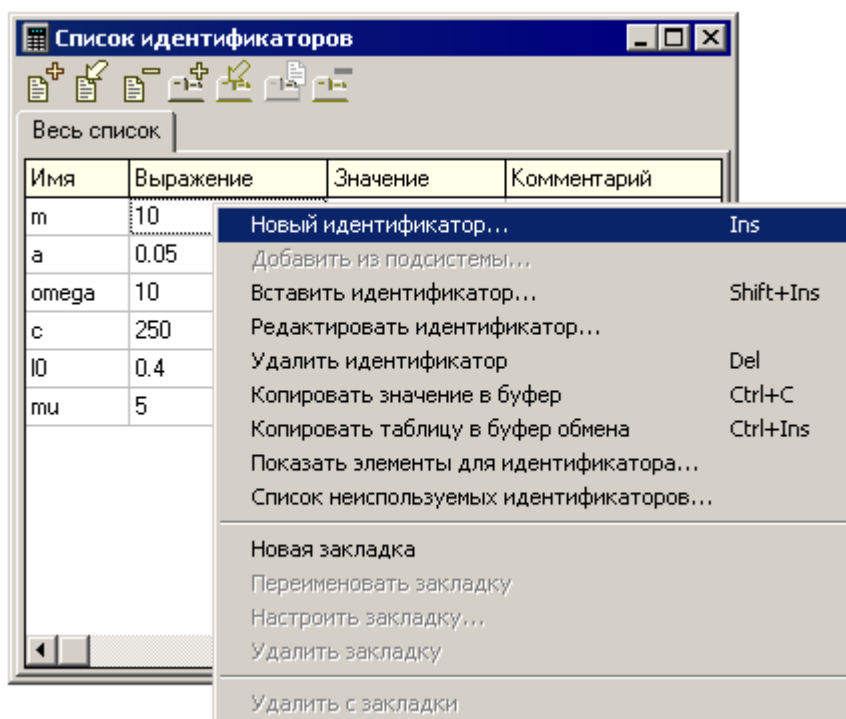
m – масса груза, кг.

Величина критического коэффициента демпфирования определяется следующим образом:

$$\mu^* = 2\sqrt{cm}, \text{ где}$$

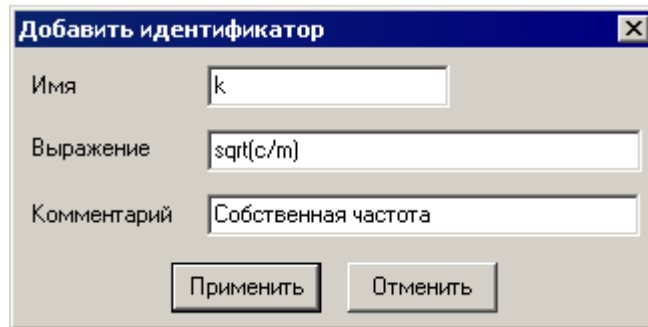
μ^* – критическое значение коэффициента демпфирования, Нс/м.

- Итак, добавим два новых идентификатора (параметра) к модели. В списке идентификаторов нажмите кнопку  или выберите пункт **Новый идентификатор** в контекстном меню.



¹ Существует такое значение коэффициента диссипации, называемое критическим, при котором движение груза уже больше не является периодическим и асимптотически стремится к равновесному положению.

2. В появившемся окне **Добавить идентификатор** введите данные, как показано на рисунке ниже (*sqrt* – обозначение квадратного корня).



3. Добавьте еще один идентификатор $\mu_{star} = 2*\sqrt{c*m}$ с комментарием «Критический коэффициент демпфирования».

2.3.8. Подготовка к моделированию

1. Сохраните модель с именем **Oscillator** (пункт меню **Файл/Сохранить как**).

Переходим в программу моделирования динамики объектов.

2. Выберите пункт меню **Объект/Моделирование**, или просто нажмите клавишу *Ctrl+M*.

Автоматически будет запущена программа моделирования **UM Simulation**, в которой откроется текущий объект.





2.4. Моделирование движения

Рассмотрим некоторые частные случаи колебаний: свободные затухающие колебания и вынужденные колебания без учета сил сопротивления.

2.4.1. Свободные колебания

Свободные колебания с учетом сил сопротивления

Откроем новое анимационное окно, а также графическое окно, в котором будем строить график вертикальной координаты груза.

1. Откройте новое анимационное окно (пункт меню **Инструменты/Анимационное окно** или кнопка ).
2. Откройте новое графическое окно (**Инструменты/Графическое окно** или кнопка ).
3. Откройте мастер переменных (**Инструменты/Мастер переменных** или кнопка ).
4. Перейдите на закладку **Лин. перем.** (линейные переменные), в списке тел выберите **Груз**, в поле **Тип** выберите **r** (координата), в поле **Компонента** выберите **Z**. Кнопкой  поместите описанную переменную в контейнер переменных и перетащите созданную переменную в графическое окно. Закройте мастер переменных.
5. Выберите пункт меню **Анализ/Моделирование**. Появится **Инспектор моделирования объекта**.
6. Разместите окна на рабочем столе программы моделирования так, как вам кажется это удобным, например, как показано на рис. 2.7.

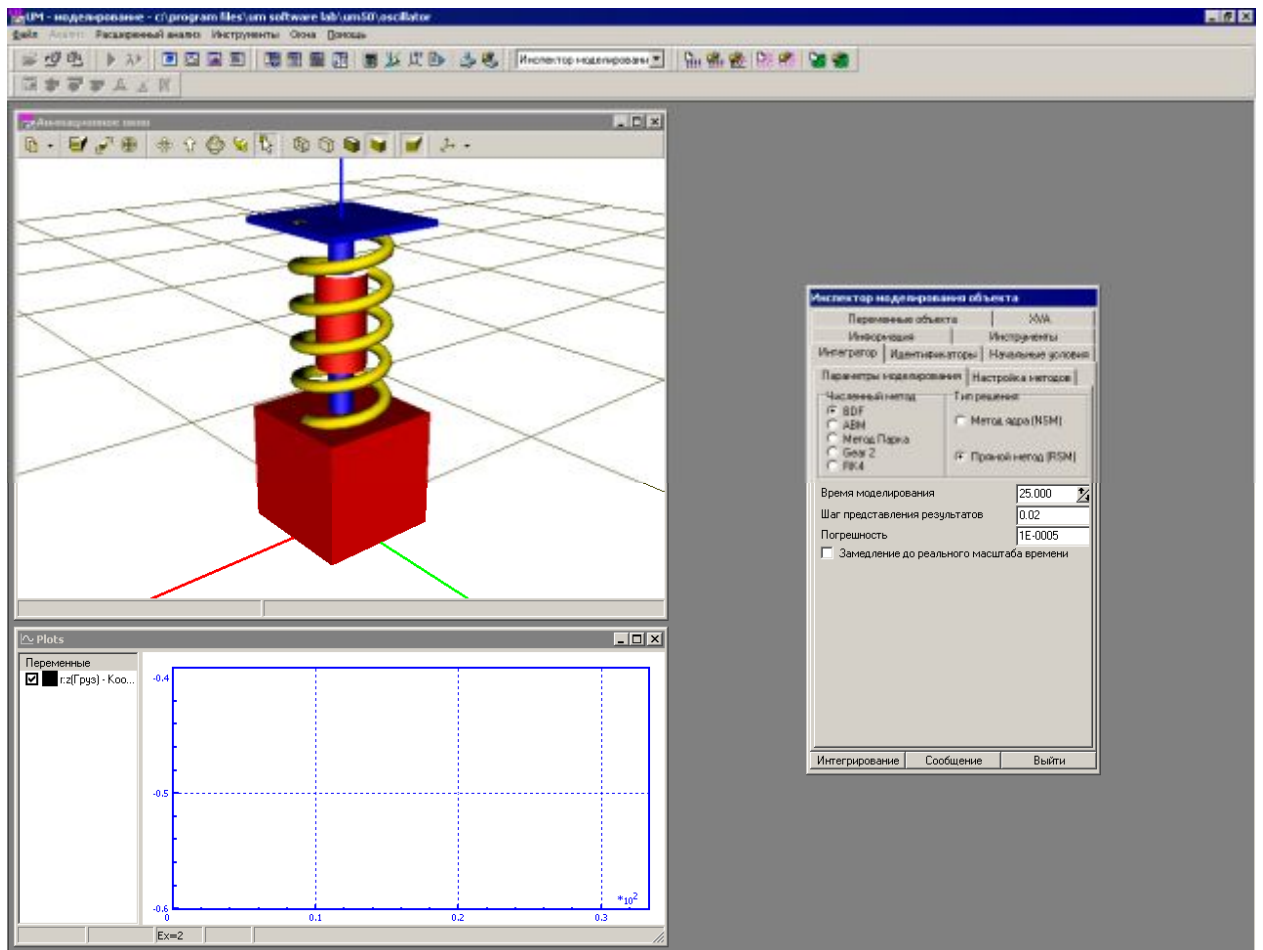


Рис. 2.7. Рабочий стол программы моделирования

7. Перейдите в **Инспектор моделирования объекта** и выберите закладку **Идентификаторы**.
8. Значение параметра **a** установите равным нулю и нажмите клавишу *Enter*. Таким образом, мы установим нулевую амплитуду колебаний подвеса, другими словами сделаем его неподвижным для исследования свободных колебаний.

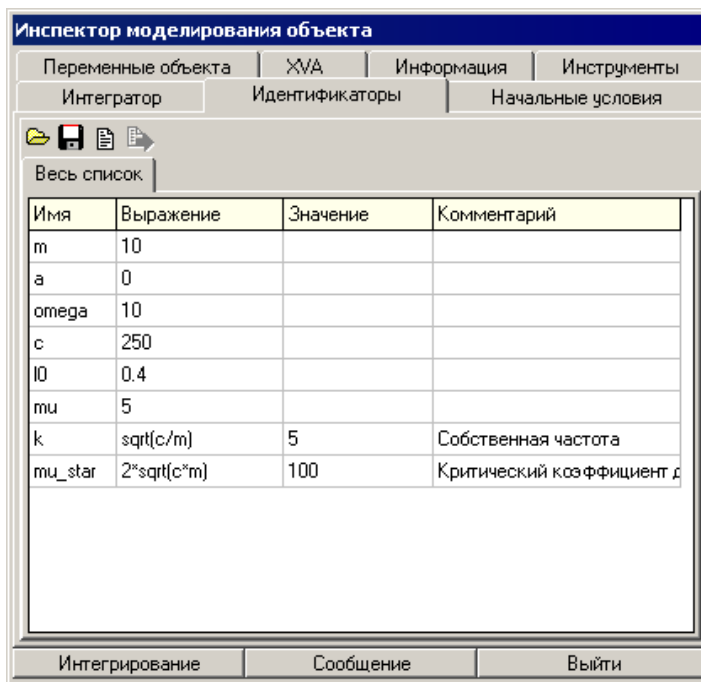


Рис. 2.8. Установка параметров модели

9. Перейдите на закладку **Начальные условия**. В поле **Координата/1.1** введите **0.1**. Дело в том, что положение системы при нулевых координатах довольно близко к положению равновесия и если не изменить начальные условия, то амплитуда колебаний будет невелика.
10. Вернитесь на закладку **Интегратор**. В поле **Время моделирования** введите **25**.
11. Запустите расчет динамики системы кнопкой **Интегрирование**.

Запустится процесс численного решения уравнений динамики в течение 25 с. В анимационном окне будет показана динамика системы, а в графическом – построен график вертикальной координаты груза в зависимости от времени.

В графическом окне нажмите кнопку **100%** на выпадающей панели быстрого доступа или выберите пункт **Показать все** контекстного меню, см. рис. 2.9. График переменной впишется в размеры окна.

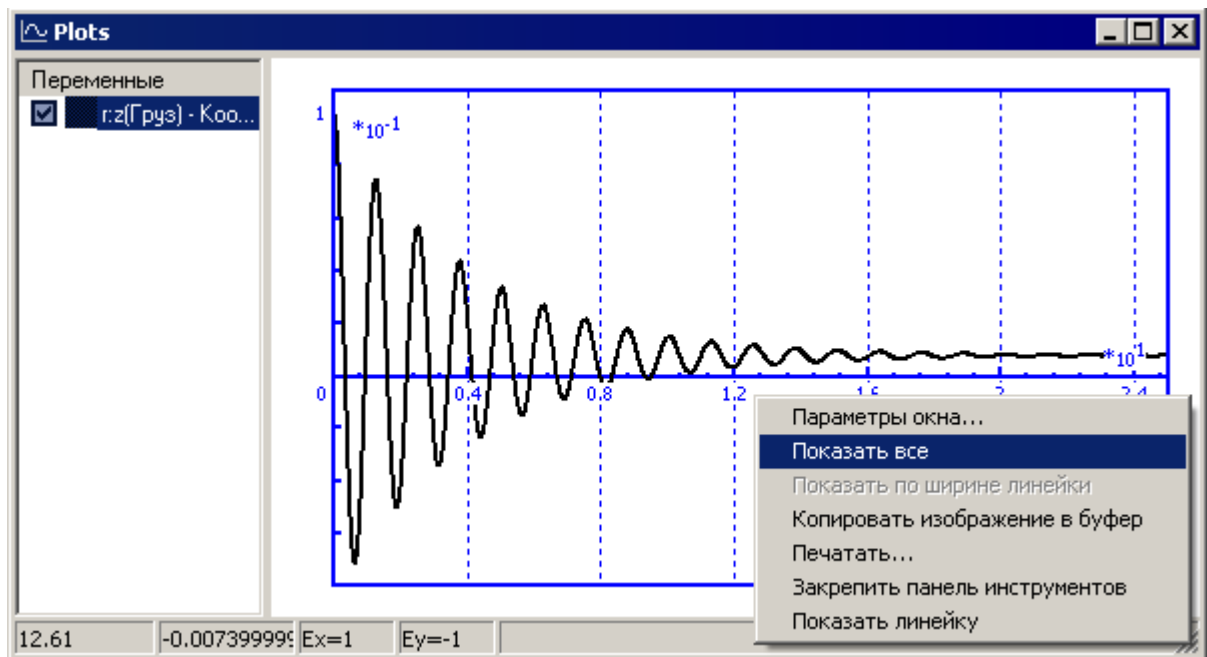


Рис. 2.9. Графическое окно после первого численного эксперимента

Свободные колебания без учета сил сопротивления

Сейчас мы отключим демпфирование и сравним графики перемещений груза. При нулевом коэффициенте диссипации мы должны получить график незатухающих колебаний.

1. Перейдите в графическое окно. Выделите переменную **r:z(Brick)** в списке переменных слева и нажмите правую кнопку мыши. Появится контекстное меню. Выберите пункт **Скопировать как статические**. Выделенная переменная будет скопирована.
2. Перейдите в окно **инспектора режима паузы** и нажмите кнопку **Прервать**. Появится **инспектор моделирования объекта**.

Замечание. Переменная **r:z(Brick)**, которую мы перетаскили в графическое окно, будет пересчитываться для каждого численного эксперимента – так называемая *динамическая* переменная. Чтобы сравнить графики одной и той же величины для экспериментов с различными значениями параметров необходимо скопировать интересующую вас переменную как *статическую*.

3. В **инспекторе моделирования объекта** перейдите на закладку **Идентификаторы**.

4. Установите значение $\mu = 0$ и нажмите **Enter**. Таким образом, мы установили нулевой коэффициент диссипации, то есть отключили демпфер.
5. Нажмите кнопку **Интегрирование**.

В течение нескольких секунд завершится моделирование свободных незатухающих колебаний. Графическое окно после двух численных экспериментов показано на рис. 2.10.

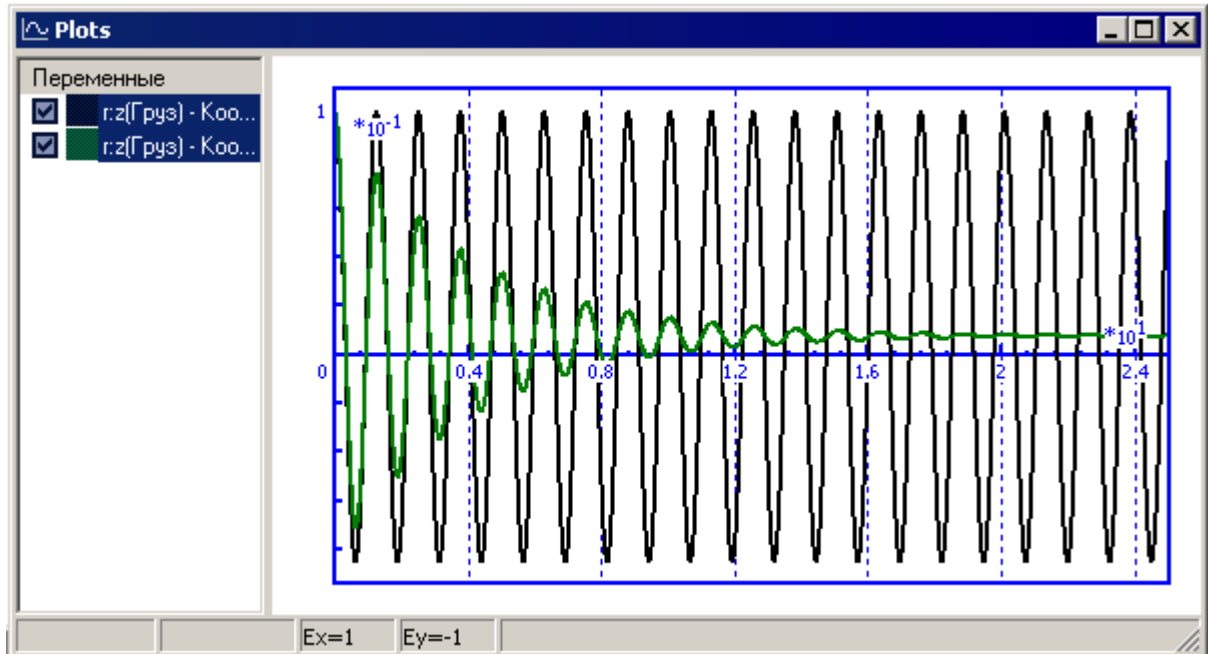


Рис. 2.10. Графическое окно после двух численных экспериментов

Свободные колебания: случай критического коэффициента диссипации

Как мы выяснили критическим значением коэффициента диссипации является $\mu = 100$ Нс/м. Проанализируем характер движения груза в этом случае.

1. Перейдите в графическое окно. Выделите первую переменную **r:z(Brick)** в списке переменных слева и выберите пункт контекстного меню **Скопировать как статические**.
2. Перейдите в окно инспектора режима паузы и нажмите кнопку **Прервать**.
3. В появившемся **инспекторе моделирования объекта** установите $\mu = 100$.
4. Запустите моделирование.

Как мы и предполагали – движение груза при критической величине коэффициента диссипации носит аperiodический характер, см. рис. 2.11.

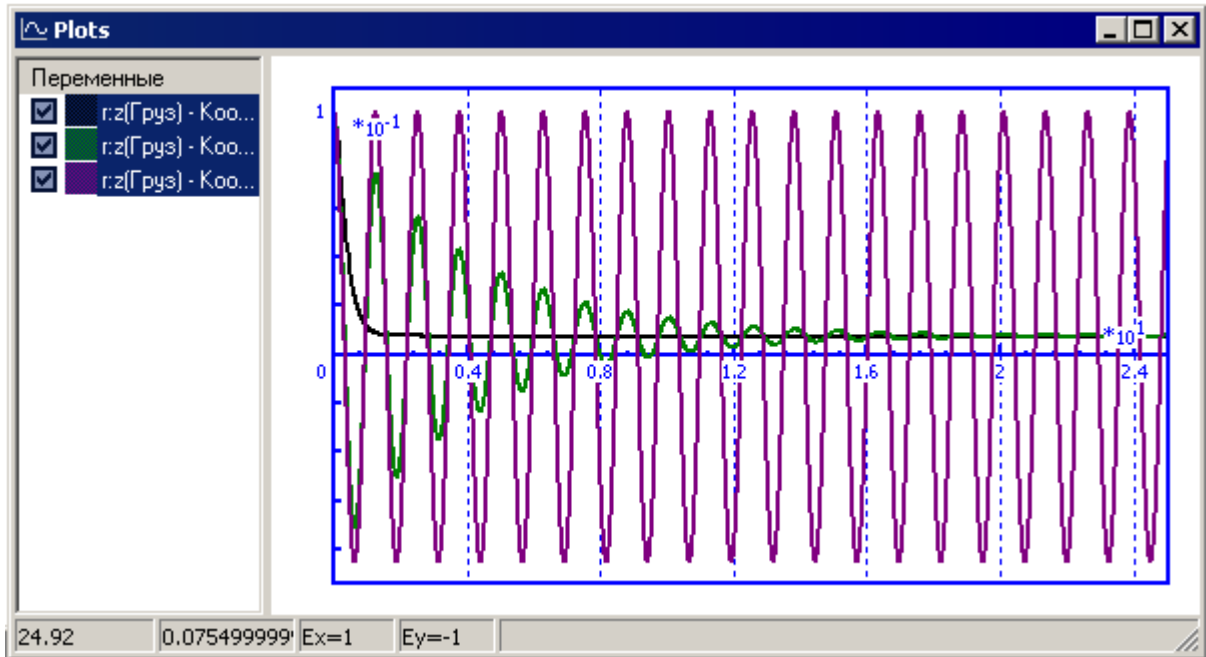


Рис. 2.11. Графическое окно после трех численных экспериментов

5. Проведите численные эксперименты для других коэффициентов диссипации. При этом не забывайте копировать переменные как статические.
6. Если вы меняли величину коэффициента диссипации, то установите ее обратно в значение $\mu = 100$ Нс/м.

2.4.2. Статистический анализ

Познакомимся с некоторыми дополнительными инструментами обработки результатов моделирования.

1. Откройте окно статистического анализа, пункт меню **Инструменты/Статистика**. Появится окно статистической обработки результатов моделирования.
2. Перетащите переменную, соответствующую незатухающим колебаниям, из графического окна в окно статистики.
3. Перейдите в окно статистики и выберите закладку **Спектральная плотность мощности**.

Как видно из графика спектральной плотности мощности процесса в решении присутствует только одна частота. Определим эту частоту по графику – приблизительно 0,78 Гц (см. значение абсциссы в левом нижнем углу окна, рис. 2.12), что составляет $0,78 * 2\pi = 4,9$ рад/с. Точное значение собственной круговой частоты составляет 5 рад/с.

Замечание. Для более точного определения частоты используйте изменение масштаба графика при помощи правая кнопки мыши и его положения при помощи левая кнопка мыши.

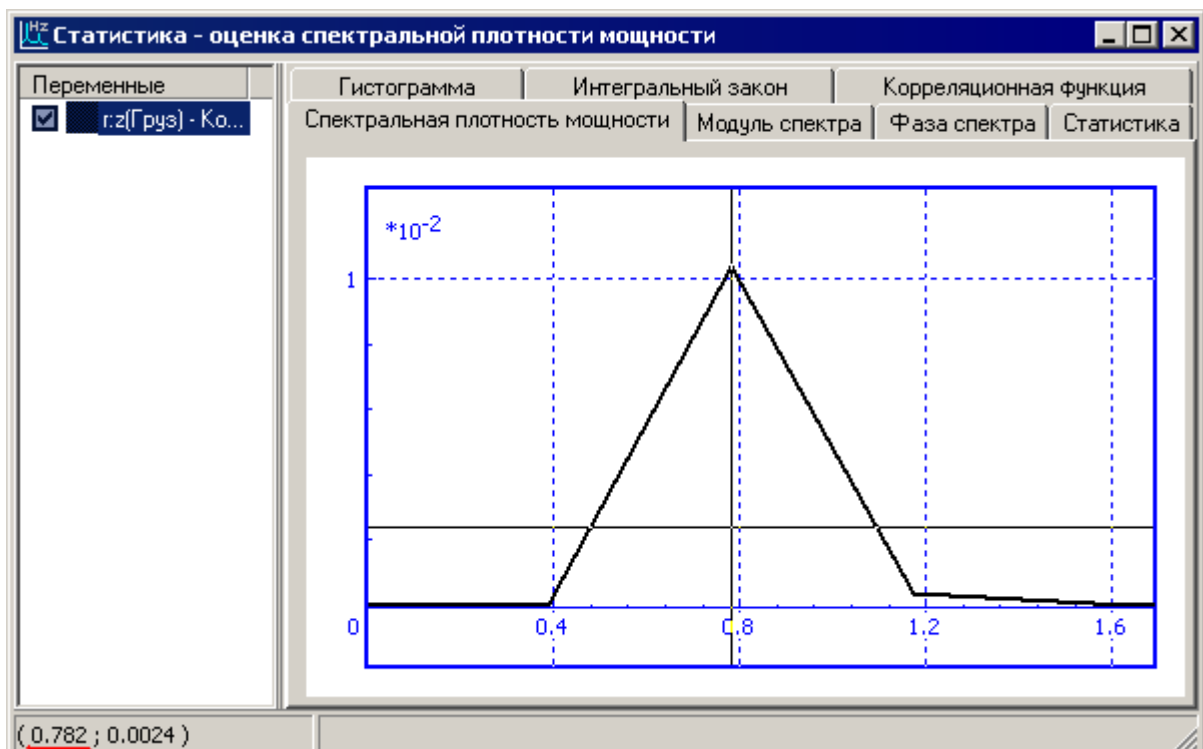


Рис. 2.12. Спектральная плотность мощности незатухающих колебаний

2.4.3. Линейный анализ

Рассмотрим пример использования *линейного анализа*. С помощью этого инструмента мы найдем положение равновесия системы, ее собственные частоты и формы колебаний, определим степень задемпфированности каждой (в нашем случае единственной) формы.

Итак, в начале закроем окна **Инспектора паузы** и **Инспектора моделирования объекта**.


1. Перейдите в окно **Инспектора паузы** и нажмите кнопку **Прервать**.
2. В окне **Инспектора моделирования объекта** нажмите **Выйти**.

Откроем окно линейного анализа

3. Выберите пункт меню **Анализ/Линейный анализ**.

Найдем положение равновесия системы

4. Перейдите на закладку **Равновесие** и нажмите кнопку **Выполнить расчет**. Появится сообщение: «Положение равновесия успешно рассчитано». Груз в анимационном окне займет положение равновесия.

Замечание. Найденные координаты системы, соответствующие положению равновесия можно сохранить в файл. Для этого используйте кнопку  на закладке **Начальные условия**. Сохраненные начальные условия вы можете затем открыть в **Инспекторе моделирования объекта**, и, таким образом, запустить моделирование динамики из положения равновесия.

Найдем собственные частоты и формы колебаний

5. Перейдите на закладку **Частоты**. В списке слева вы увидите все собственные частоты системы. Найденная собственная круговая частота равна 0,795775 Гц, что соответствует частоте 5,0000 рад/с.
6. Для анимации формы колебаний нажмите кнопку **Показать**. Выставьте приемлемую **Амплитуду** и **Скорость** анимации. Нажмите кнопку **Стоп** для остановки показа собственных форм.

Устойчивость движения

Найдем корни характеристического уравнения системы линеаризованных уравнений движения и по характеру корней будем судить об устойчивости модели.

7. В поле **Рассчитать** выберите **Корни характ.** (характеристического) **уравнения**. Как видим реальные части (в общем случае комплексных) корней меньше нуля. Таким образом, можно сделать вывод, что система устойчива.

Доля демпфирования от критического

8. Определим величину коэффициента демпфирования от критического. В контекстном меню на списки корней выберите **Частота+доля демпфирования**. Как видим, случаю $\mu = 100$ соответствует нулевая частота (апериодическое движение), а доля демпфирования от критического составляет 100%.

Замечание. Расчет доли демпфирования от критического позволяет оценить степень задемпфированности каждой собственной формы и, таким образом, при необходимости уточнить параметры гасителей колебаний и/или места их прикрепления к элементам конструкции.

9. Вы можете поменять значение коэффициента диссипации на закладке **Идентификаторы** и посмотреть, как меняются корни и доля демпфирования от критического.

10. Закройте окно линейного анализа.

2.4.4. Вынужденные колебания

Рассмотрим моделирование вынужденных колебаний без учета сил сопротивления.

1. Удалите все статические переменные из графического окна. Оставьте только первую (динамическую) переменную.
2. Запустите **Инспектор моделирования объекта**. Выберите пункт меню **Анализ/Моделирование**.
3. Перейдите на закладку идентификаторы. Установите следующие значения параметров: $\mathbf{a = 0.05}$, $\mathbf{\omega = 8}$, $\mathbf{\mu = 0}$.
4. Запустите моделирование. Теперь вы видите, что точка подвеса пружины также совершает колебания. График положения груза в проекции на вертикальную ось представлен на рис. 2.13.

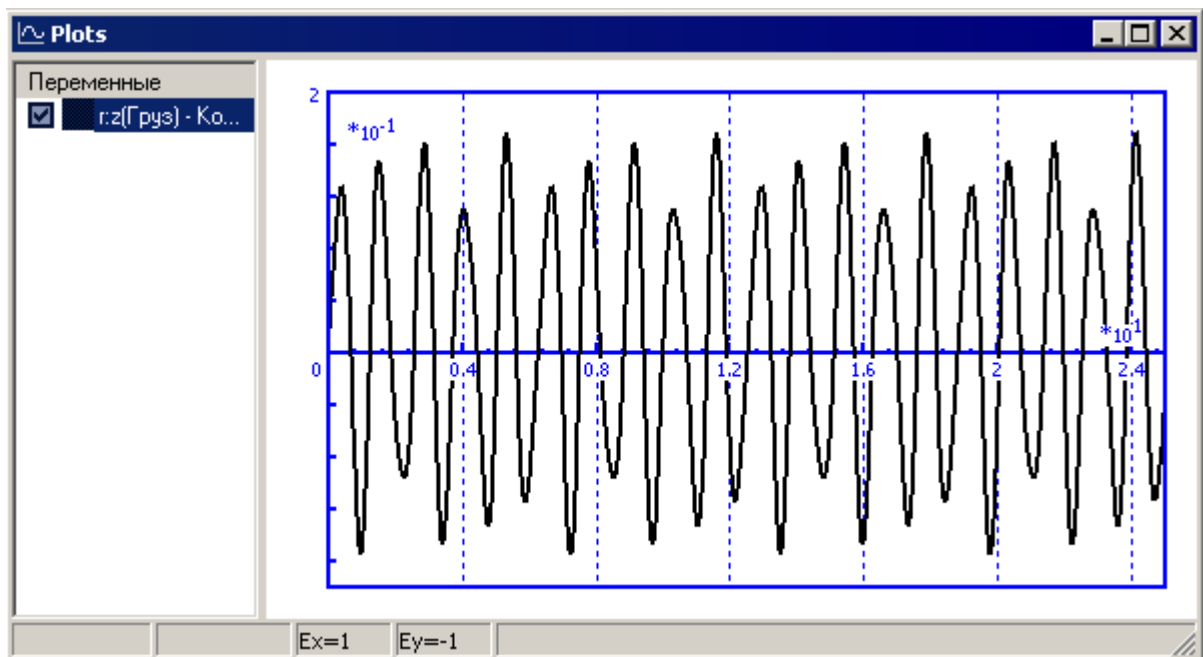


Рис. 2.13. Вынужденные колебания: частота точки подвеса пружины 8 рад/с

Резонанс

В заключение рассмотрим случай, когда частота возбуждения равна собственной частоте колебаний – резонанс.

1. В окне **Режима паузы** нажмите кнопку **Прервать**.
2. В **Инспекторе моделирования объекта** установите $\omega = 5$.
3. Запустите моделирование. Как и ожидалось, в резонансном случае амплитуда колебаний возрастает с течением времени, см. рис. 2.14.

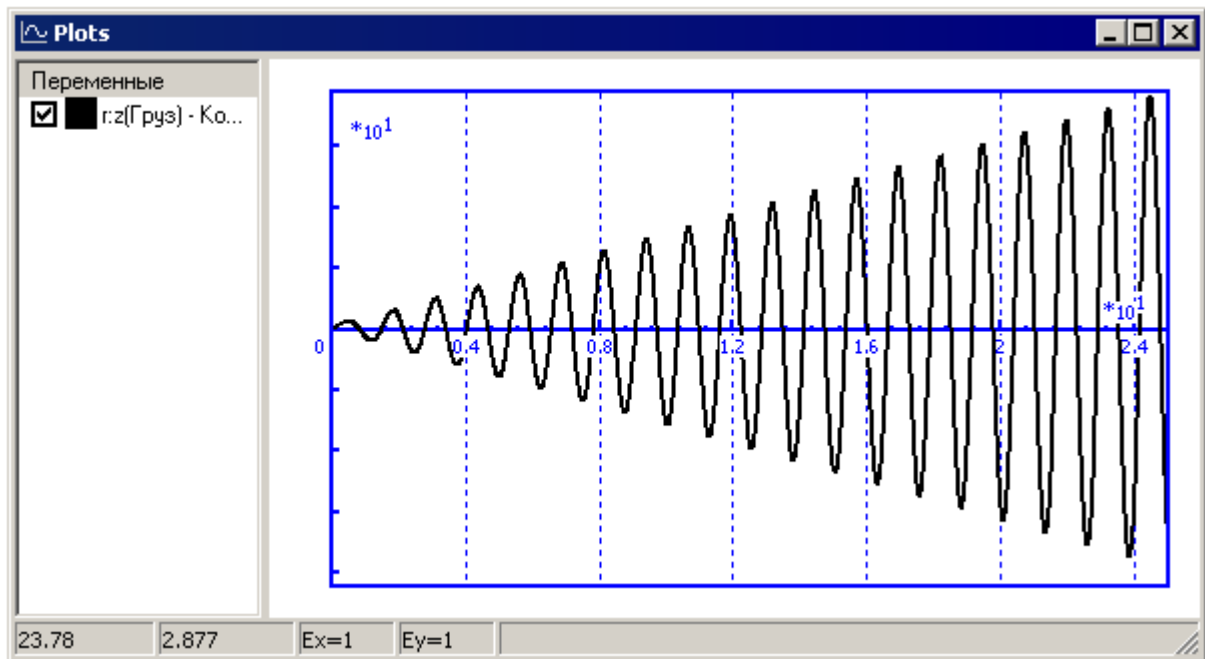


Рис. 2.14. Вынужденные колебания: резонанс

3. Дальнейшее изучение программы

Вы познакомились с двумя примерами создания и моделирования механических систем: маятника и груза на пружине. И прошли через все основные этапы создания и анализа динамики моделей.

Серия глав *Начинаем работать* включает также следующие главы, посвященные введению в использование отдельных модулей программы.

- [Начинаем работать: моделирование автомобилей](#)
- [Начинаем работать: железнодорожные экипажи](#)
- [Начинаем работать: модуль моделирования упругих тел](#)
- [Начинаем работать: интерфейс с Matlab/Simulink](#)
- [Начинаем работать: модуль многовариантных расчетов](#)
- [Начинаем работать: расчет долговечности.](#)

Библиотека примеров для изучения

Руководство пользователя ПК «Универсальный механизм» включает главу **07_UM_Simulation_Examples.pdf**, которая посвящена рассмотрению простых примеров, на которых изучается описание графических элементов, шарниров и силовых элементов различных типов. Изучение этих примеров позволит вам глубже ознакомиться с элементной базой программы и подходами к моделированию тех или иных механических систем. Библиотека моделей находится в каталоге `{um_root}\samples\library`.

Файл **07_UM_Simulation_Examples.pdf** вы можете найти в каталоге `{um_root}\manual` или в интернете по адресу:

http://www.umlab.ru/download/60/rus/07_um_simulation_examples.pdf