

Лабораторная работа 5

Моделирование поступательного движения вращающегося диска с двумя массами закрепленными на нем упругими элементами

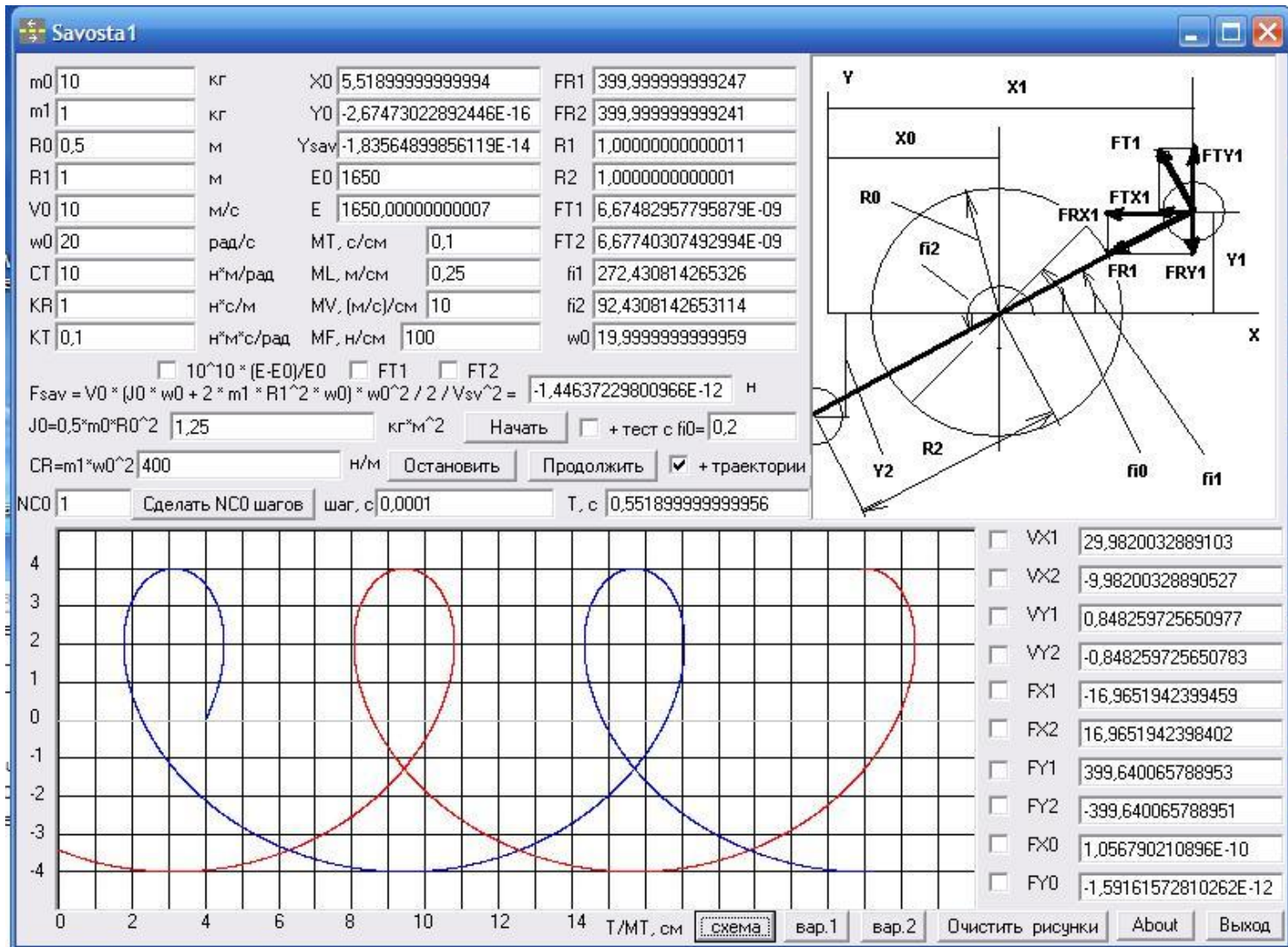
Цель работы: смоделировать поступательное движение вращающегося диска с двумя массами закрепленными на нем упругими элементами

Теоретическая часть

Основное назначение программы Savosta продемонстрировать, как из реальных сил инерции при вращательном движении возникает фиктивная центробежная сила, которая может быть использована как аналог сил инерции поступательного движения при решении простейших задач с вращающимися телами, а побочное назначение показать, что при любых скоростях движущегося поступательно и вращающегося диска не возникает поперечного дисбаланса центробежных сил, который бы приводил, как пишет в своей статье Савоста, к возникновению, так называемой, поперечной силы.

<http://scicommunity.ru/index.php/materials/nauchnye-trudy/112-savosta-v-s-o-zakone-poperechnoj-sily-dejstvuyushchej-na-vrashchayushchiesya-tela-pri-ikh-postupatelnom-dvizhenii-v-svobodnom-prostranstve-vakuume>

Конкретно в программе моделируется движение в плоскости $X-Y$ с начальной скоростью V_0 вдоль оси X диска радиусом R_0 с массой m_0 и моментом инерции J_0 , и который вращается вокруг оси Z с начальной угловой скоростью ω_0 . К этому диску с противоположных сторон прикреплены с помощью упругих элементов две массы m_1 и m_2 ($m_2 = m_1$), которые в начале моделирования находятся на расстояниях R_1 и R_2 от центра диска ($R_2 = R_1$) вдоль оси X , т.е. их ординаты в начале моделирования равны нулю. Упругие элементы имеют одинаковую радиальную жесткость CR и тангенциальную CT и позволяют перемещаться грузам вдоль радиусов и закручиваться относительно диска.



И, если Вы нажмете кнопку <схема>, то увидите на правом рисунке, как это показано на скриншоте, схему расчета сил действующих на массы m_1 и m_2 . Естественно, точно такие же силы, но с противоположным знаком будут действовать на массу m_0 . При этом, зная (вычислив) координаты всех масс и определив расстояния R_1 и R_2 , радиальные силы мы будем определять по формуле

$$FR = CR \cdot R + KR \cdot VR$$

где VR это скорость изменения радиуса (деформации пружины), а KR это коэффициент жидкостного трения. В принципе, наличие последнего члена в этой формуле не обязательно и в самой конструкции этот демпфер может отсутствовать, но для более стабильного решения дифференциальных уравнений численными методами я всегда добавляю в уравнения силы демпфирования. Но, если они Вам так не нравятся, то можете задать в исходных данных $KR=0$, и никаких сил демпфирования не будет.

Аналогично, зная угол поворота φ_0 места крепления упругого элемента СТ на диске и угловое положение масс m_1 и m_2 , т.е. углы φ_1 и φ_2 , находим углы закрутки упругих элементов $D\varphi_1$ и $D\varphi_2$, а потом по формуле

$$F_T = (C_T * D\varphi + K_R * V_T) / R$$

находим тангенциальную силу, действующую на массы m_1 и m_2 , где аналогично K_T это коэффициент жидкостного трения в упругом элементе, а V_T это угловая скорость закрутки упругого элемента. При этом силы F_{T1} и F_{T2} будут не только (с обратным знаком) действовать на массу m_0 , но и создавать крутящий момент (с обратным знаком), который будет замедлять вращение диска.

Теперь раскладываем эти силы F_R и F_T по осям X и Y и находим, используя 2-ой закон Ньютона, ускорения масс m_0 , m_1 и m_2 по осям X и Y , а, зная крутящий момент, приложенный к диску, находим его угловое ускорение. Таким образом, вращательное движение масс m_1 и m_2 вокруг диска с массой m_0 моделируется только как поступательное движение масс m_1 и m_2 по осям X и Y , т.е. тут не может быть никаких центробежных сил, а возникают только силы инерции, которые пытаются сохранить прямолинейное движение, но при этом увеличивают расстояние между массами m_1 и m_2 и массой m_0 , что приводит к увеличению радиальной силы растягивающейся пружины и эта сила изменяет направление движения масс m_1 и m_2 , заставляя их не улетать далеко от массы m_0 , а в конечном итоге двигаться вокруг нее на одном и том же расстоянии.

А вот в окошке $\langle Y_{sav} \rangle$ у нас будет выводиться ордината диска, которая должна у него быть на данный момент времени, если бы на него действовала сила F_{sav} . Как не трудно убедиться, сравнив эту ординату Y_{sav} с получающимся значением Y_0 , никакой боковой силы $S_{авосты}$ не существует, т.е. центробежная сила, действующая на 1-ый и 2-ой грузы, когда они расположены перпендикулярно скорости движения, будет одна и та же. Это легко объясняется, если мы посмотрим на траектории движения этих масс в этих положениях (смотрите скриншот программы), из которых видно, что у левого груза радиус кривизны будет меньше, а у правого больше, что при разных линейных скоростях дает одинаковое значение фиктивной центробежной силы.

Описание работы с программой

Основное назначение программы Savosta продемонстрировать, как из реальных сил инерции при вращательном движении возникает фиктивная центробежная сила, а побочное назначение показать, что при любых скоростях движущегося поступательно и вращающегося диска не возникает поперечного дисбаланса центробежных сил, который бы приводил, как пишет в своей статье Савоста, к возникновению, так называемой, поперечной силы.

Конкретно в программе моделируется движение в плоскости $X Y$ с начальной скоростью V_0 вдоль оси X диска радиусом R_0 с массой m_0 и моментом инерции J_0 , и который вращается вокруг оси Z с начальной угловой скоростью ω_0 . К этому диску с противоположных сторон прикреплены с помощью упругих элементов две массы m_1 и m_2 ($m_2 = m_1$), которые в начале моделирования находятся на расстояниях R_1 и R_2 от центра диска ($R_2 = R_1$) вдоль оси X , т.е. их ординаты в начале моделирования равны нулю. Упругие элементы имеют одинаковую радиальную жесткость C_R и тангенциальную C_T и позволяют перемещаться грузам вдоль радиусов и закручиваться относительно диска.

Если Вы нажмете кнопку <схема>, то увидите на правом рисунке схему расчета сил действующих на массы m_1 и m_2 . Естественно, точно такие же силы, но с противоположным знаком будут действовать на массу m_0 . При этом, зная координаты всех масс и определив расстояния R_1 и R_2 , радиальные силы мы будем определять по формуле

$$F_R = C_R * R + K_R * V_R$$

где V_R это скорость изменения радиуса (деформации пружины), а K_R это коэффициент жидкостного трения. В принципе, наличие последнего члена в этой формуле не обязательно и в самой конструкции этот демпфер может отсутствовать, но для более стабильного решения дифференциальных уравнений численными методами я всегда добавляю в уравнения силы демпфирования. Но, если они Вам так не нравятся, то можете задать в исходных данных $K_R=0$, и никаких сил демпфирования не будет.

Аналогично, зная угол поворота φ_0 места крепления упругого элемента СТ на диске и угловое положение масс m_1 и m_2 , т.е. углы φ_1 и φ_2 , находим углы закрутки упругих элементов $D\varphi_1$ и $D\varphi_2$, а потом по формуле

$$F_T = (C_T * D\varphi + K_R * V_T) / R$$

находим тангенциальную силу, действующую на массы m_1 и m_2 , где аналогично K_T это коэффициент жидкостного трения в упругом элементе, а V_T это угловая скорость закрутки упругого элемента. При этом силы F_{T1} и F_{T2} будут не только (с обратным знаком) действовать на массу m_0 , но и создавать крутящий момент (с обратным знаком), который будет замедлять вращение диска.

Теперь раскладываем эти силы F_R и F_T по осям X и Y и находим, используя 2-ой закон Ньютона, ускорения масс m_0 , m_1 и m_2 по осям X и Y , а, зная крутящий момент, приложенный к диску, находим его угловое ускорение. Данная система дифференциальных уравнений решается в программе численными методами с заданным в окошке <шаг, с> шагом решения методом Рунге-Кутты и через заданное Вами в окошке < N_{C0} > число шагов решения данные выводятся на экран. Кого интересуют подробности, могут ознакомиться с исходным кодом программы, а сейчас я просто коротко опишу порядок работы с программой.

После загрузки программы, Вы видите слева вверху исходные данные по рассматриваемой конструкции диска с двумя массами, которые загружаются по умолчанию или после нажатия на кнопку <вар.1>. А, если Вы нажмете на кнопку <вар.2>, то загрузятся данные, которые позволяют получить заметное значение поперечной силы Савосты, которая по приведенной на форме формуле будет вычислена после начала работы программы и будет выведена в соответствующее окошко. Точно также по приведенным формулам будут вычисляться момент инерции диска J_0 и радиальная жесткость упругого элемента C_R .

До того как Вы нажмете на кнопку <Начать>, Вы можете изменить прямо в окошках все начальные данные, а если Вы остановили программу (нажали на кнопку <Остановить>) и потом хотите продолжить выполнение (нажать на кнопку <Продолжить>) или сделать только NC0 шагов решения (нажать на кнопку <Сделать NC0 шагов>), то можете изменить только некоторые из них. Конкретно это шаг решения и NC0, а также масштабы вывода рассчитанных данных на рисунки. Здесь MT это масштаб времени для развертки на нижнем рисунке, ML это масштаб вывода изображения конструкции на правом рисунке и координат X и Y для масс, когда отмечен чекбокс <+ траектории>, а MV и MF это масштабы вывода на нижний рисунок скоростей и сил, которые Вы видите в окошках справа внизу формы в том случае, если Вы отметите чекбокс у нужной Вам силы или скорости.

А в процессе работы программы Вы всегда будете видеть в соответствующих окошках текущие координаты и скорости элементов конструкции и действующие на них силы, а также в окошке <T, с> будет выводиться смоделированное время. А вот в окошке <Ysav> будет выводиться ордината диска, которая должна у него быть на данный момент времени, если бы на него действовала сила Fsav. Как не трудно убедиться, сравнив эту ординату Ysav с получающимся значением Y0, никакой боковой силы Савосты не существует, т.е. центробежная сила, действующая на 1-ый и 2-ой грузы, когда они расположены перпендикулярно скорости движения, будет одна и та же. Это легко объясняется, если мы посмотрим на траектории движения этих масс в этих положениях, из которых видно, что у левого груза радиус кривизны будет меньше, а у правого больше, что при разных линейных скоростях дает одинаковое значение фиктивной центробежной силы.

А для того, чтобы убедиться, что все рассмотренные процессы именно моделируются, а не являются чисто кинематическим расчетом, перед началом вычислений с данными варианта 1, отметьте чекбокс <+ тест с $\dot{\varphi}_0 \neq 0$ > и Вы увидите, что грузы будут закручиваться относительно нейтрального положения на диске то в одну, то в другую стороны, а при наличие жидкостного трения эти колебания со временем прекратятся.

При этом, учитывая то, что данная конструкция с данными варианта 2 очень требовательна к точности решения, а используемая мною двойная точность вычислений в языке Visual Basic 6, на котором написана эта программа, составляет только 15 цифр из которых несколько последних могут вообще собирать мусор (посмотрите на значение Т при расчете с данными варианта 1, где на каждом шаге решения к Т просто прибавляется 1 шаг), я решил для контроля точности решения вывести в окошках еще и полную энергию системы (E0- в начале вычислительного эксперимента и E- текущее значение).

И, если Вы отметите чекбокс $<10^{10} * (E-E0)/E0>$, то ошибка решения по энергии системы будет выводиться и графически на нижнем рисунке. Данный контроль нужен для того, чтобы правильно подобрать тангенциальную жесткость и коэффициенты жидкостного трения для конструкции с другими массами и размерами. И особенно это относится к шагу решения. Чем более массивные элементы конструкции и меньше их скорости, тем может быть больше шаг решения. Кстати, если программа будет очень медленно считать с заданным Вами очень маленьким шагом решения, то задайте побольше значение NC0, чтобы компьютер тратил меньше времени на работу с графикой (вывод данных в окошки и на рисунки).

А, что касается наглядного представления откуда берутся ошибки при малой разрядности компьютера или используемого типа данных, отметьте чекбоксы <FT1> или <FT2> и Вы увидите, что в начале, когда был маленький угол поворота диска, эти силы выводятся как сплошная линия, затем она становится все шире и шире, а потом и вообще разделяется на две широких полосы. Связано это с тем, что происходит не точный перевод полного угла поворота диска в угол от 0 до $2 * \pi$, когда из общего угла поворота вычитается целое значение поворотов с углом $2 * \pi$ и находится остаток. По этому, если у Вас при моделировании будет не устойчивое решение системы дифференциальных уравнений методом Рунге-Кутты, то это объясняется либо не правильным выбором шага решения или коэффициентов демпфирования или просто не достаточной разрядностью компьютера или типа используемых данных, т.к. сам метод Рунге-Кутты зарекомендовал себя как один из самых стабильных методов.

Практическая часть

Прилагается готовая программа **Savosta1.exe**.

Savosta1.exe – моделируется поступательное движение вращающегося диска с двумя массами закрепленными на нем упругими элементами.

исполняемый файл Savosta1.exe.zip – (Windows 95-98-XP-Vista-7, 36 kb)

исходные файлы Savosta1.vbp.zip - (Windows 95-98-XP-Vista-7, Visual Basic 6.0, 20 kb)

Задачи:

1. Смоделировать поступательное движение вращающегося диска с двумя массами закрепленными на нем упругими элементами, рассказать про функционал программы и пояснить параметры;
2. Доказать, что при любых скоростях движущегося поступательно и вращающегося диска не возникает поперечного дисбаланса центробежных сил, который бы приводил к возникновению, так называемой, поперечной силы.

Оформить отчет (шапка, цель, теорчасть, пракчасть, вывод).

ВНИМАНИЕ, если не работают программы - здесь программы, написанные в основном на языке программирования VisualBasic 6.0, которые требуют для своей работы библиотеку MSVBVM60.DLL. По этому, если у Вас на компьютере установлена операционная система Windows Vista или Windows 7, то эта библиотека у Вас уже есть, а если установлена Windows XP, то у Вас в ней быстрее всего стоит библиотека MSVBVM50.DLL, т.е. библиотека для предыдущей версии языка VisualBasic. Если это так (посмотрите в папке C: \ WINDOWS \ SYSTEM32), то Вам надо будет скачать библиотеку MSVBVM60.DLL, а затем поместить ее в папку C: \ WINDOWS \ SYSTEM32. Потом можете скачивать только заархивированные исполняемые файлы нужных Вам программ, например, Solsys4exe.zip или Solsys5exe.rar, распаковывать их и запускать как и все остальные программы. Если программы на других языках программирования, то особенности их работы будут оговорены отдельно. Таким образом, если у Вас стоят Windows Vista, Windows 7 или Windows XP следующий абзац можете не читать.

А вот если у Вас установлена Windows 98 или Windows 95, а также если у Вас стоит Internet Explorer версии ниже 5.0, то Вам быстрее всего придется установить на свой компьютер не только библиотеку MSVBVM60.DLL, но и некоторые другие библиотеки, например, OLEAUT32.DLL, OLEPRO32.DLL, STDOLE2.TLB, ASYCFILT.DLL, COMCAT.DLL, CTL3D32.DLL или MSSTDFMT.DLL. Для этого надо скачать и файл 7dll.zip, разархивировать и поместить эти файлы в папку C: \ WINDOWS \ SYSTEM. О том, что у Вас нет какой то библиотеки компьютер Вам сообщит сам, если программа при запуске не найдет ее в нужном месте. Все эти библиотеки (если нужно) Вы устанавливаете только один раз, а затем производите установку только нужных Вам программ.

P.S. Если у Вас установлена 64-х битная Windows 7, то некоторые программы будут требовать некоторые 32-х битные библиотеки, которых нет в 64-х битной версии Windows 7. В таких случаях необходимые файлы библиотек будут распространяться вместе с исполняемыми файлами программ. Чтобы программы работали, эти библиотеки не обязательно регистрировать. Просто оставьте их в той папке, где находится исполняемый файл программы.

[библиотека msvbvm60dll.zip](#) (695 kb)

[прочие библиотеки 7dll.zip](#) (447 kb)

=====

Если всё равно не запускается, то попробуйте так:

Копируем файл MSSTDFMT.DLL в

C:\Windows\system\

C:\Windows\

Далее, Пуск->cmd(запускаем от имени администратора)->

ВВОДИМ

rem regsvr32 MSSTDFMT.DLL /i /s

или

regsvr32 MSSTDFMT.DLL /s

Запускаем **Программа.exe**.