ИНБИКСТ МФТИ

Карпов В.Э., Сорокоумов П.С.

Указания по выполнению практических работ
по курсу “Управление в технических системах”

Работа 3. PID-регулятор

2019 г.

Цель работы
Освоить возможности scipy.signal для моделирования поведения реальной физической системы под управлением PID-регулятора.

Задачи

1. Получить общее понимание процесса моделирования реальных физических систем линейными дифференциальными уравнениями, ограничений и недостатков данного подхода.
2. Применить освоенные ранее средства анализа систем к блоку управления на основе PID-регулятора.

Порядок выполнения работы

1. Промоделируем работу системы управления, поддерживающей равновесие обратного маятника. Этот маятник представляет собой массивное тело, закреплённое на горизонтальной оси так, что его центр тяжести находится выше этой оси (рис. 1). Так как равновесие в верхней точке неустойчиво, то, чтобы маятник не перевернулся от малых случайных воздействий, необходимо удерживать его с помощью мотора, который прилагает к оси вращения момент силы. Входной сигнал мотора рассчитывается с помощью PID-регулятора, на вход которого подаётся текущее отклонение маятника от вертикали. Целью настройки является получение таких параметров PID-регулятора, которые позволят маятнику устойчиво приходить в вертикальное положение из любых позиций, отклоняющихся от вертикали не более чем на 10-15°.

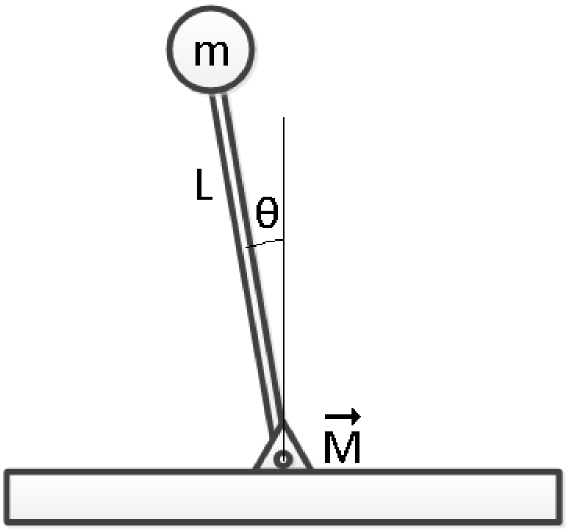

Рис. 1 Управляемая система

Пусть маятник состоит из груза массы m, закреплённого на невесомом жёстком стержне длины L без трения в оси. В этом случае на маятник действуют момент силы тяжести, равный $mgL\ sin\theta$, и момент силы мотора M. Сумма этих моментов придаёт маятнику угловое ускорение:

$$I\ddot{\theta} = mgL\ sin\theta + M, \qquad (1)$$

где I – момент инерции маятника относительно оси вращения. $I = mL^2$, так как стержень невесом, а груз можно считать материальной точкой. Тогда можно переписать формулу (1) как

$$mL^2\ddot{\theta} = mgL\ sin\theta + M \qquad (2)$$

Если считать, что угол отклонения θ мал, то $sin\theta \approx \theta$, что приводит к итоговому виду дифференциального уравнения управляемой системы:

$$(mL^2p^2 - mgL)\theta = M \qquad (3)$$

При этом входным сигналом является момент M, выходным – угол θ.

Примем следующие значения параметров системы: m = 100 г, L = 50см, тогда уравнение (3) после деления на коэффициент при p^2 примет вид

$$(p^2 - 19.52)\theta = 40M \qquad (3)$$

```
#!/usr/bin/env python3
import numpy as np
from scipy import signal
import matplotlib.pyplot as plt
dT = 0.1
TotalTime = 5
Tin = np.linspace(0, TotalTime, TotalTime/dT + 1)
m = 0.1
L = 0.5
g = 9.81
sys_tf = signal.TransferFunction([1], [m*L**2, 0, -m*g*L])
print(sys_tf)
TransferFunctionContinuous(array([40.]), array([1., 0., -19.62]), dt: None)
```

2. Чтобы задавать начальное состояние системы при моделировании, библиотека scipy.signal требует представить систему с помощью уравнений состояния. В простейших случаях можно выполнить преобразование вручную (выбрать в качестве параметров состояния, например, угловую скорость маятника и его отклонение, а затем привести уравнение (3) к нужному виду), но быстрее построить систему уравнений с помощью стандартной функции преобразования и затем проинтерпретировать переменные, то есть выяснить их физический смысл:

```
sys_tf = signal.tf2ss([1], [m*L**2, 0, -m*g*L])
print(sys_tf)
(array([[-0.  , 19.62], [ 1.  ,  0.  ]]),
 array([[1.], [0.]]),
 array([[0., 40.]]),
 array([[0.]]))
```

При печати представления системы в пространстве состояний выводятся последовательно матрицы A,B,C,D. Так как они выводятся по строкам, то видно, что система представлена в виде:

$$A = \begin{pmatrix} 0 & 19.62 \\ 1 & 0 \end{pmatrix}, B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, C = \begin{pmatrix} 0 & 40 \end{pmatrix}, D = 0$$

Иначе говоря, начальное дифференциальное уравнение равносильно системе:

$$\begin{cases} \dot{z_1} = 19.62z_2 + M \\ \dot{z_2} = z_1 \\ \theta = 40z_2 \end{cases}$$

Видно, что автоматически созданная переменная состояния $z_2 = \theta/40$ (т.е. пропорциональна углу отклонения), z1 – скорость её изменения (т.е. пропорциональна угловой скорости маятника).

Исходя из полученной информации можно промоделировать движение маятника из наперёд заданного состояния, т.е. по начальному отклонению и угловой скорости (рис. 2):

```
zero_input = np.zeros(Tin.shape)  # не будем прилагать момент
start_pos = 1 / 180.0*np.pi # начнём с позиции в 0.001°.
Tout,yout,xout = signal.lsim(sys_tf, zero_input, Tin, X0=[0, start_pos/40])  # начальная скорость нулевая, угол - start_pos
plt.plot(Tout, yout, 'b')
# покажем предельные допустимые отклонения красными линиями
limit = 15 / 180.0 * np.pi
plt.plot([0, TotalTime], [limit, limit], 'r')
plt.plot([0, TotalTime], [-limit, -limit], 'r')
plt.show()
```

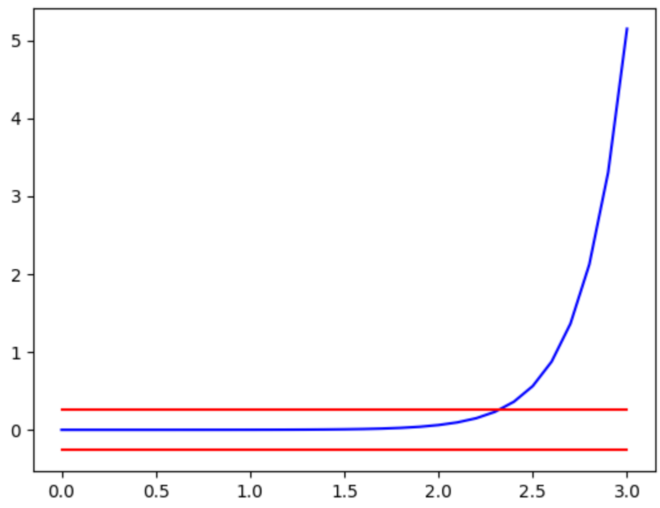

Рис. 2 Модель свободного движения маятника

Видно, что маятник отклоняется от начального, почти вертикального положения с возрастающей скоростью. При больших отклонениях (т.е. при выходе за пределы красных линий, соответствующих 15°) допущения модели не выполняются, и дальнейшее моделирование бессмысленно.

3. Добавим в систему PID-регулятор, на который поступает расхождение между требуемым (т.е. нулевым) и реальным отклонением маятника. Этот регулятор будет формировать входной сигнал для мотора, т.е. момент M (рис.3).

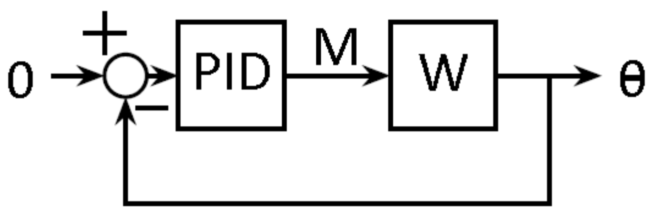

Рис. 3 Добавление PID-регулятора в систему

Так как передаточная функция PID-регулятора известна, то можно получить передаточную функцию итоговой замкнутой системы по стандартным формулам (в коде примера для этого используются стандартные функции перемножения многочленов библиотеки numpy).

```
P = 1
I = 0
D = 0
PID_num = [D, P, I]
PID_den = [1, 0]
interm_num = np.convolve([1], PID_num)
interm_den = np.convolve([m*L**2, 0, -m*g*L], PID_den)
total_sys = signal.tf2ss(interm_den, np.polyadd(interm_den, interm_num))
print(total_sys)
```

(array([[-0. , -20.38, -0.], [1. ,   0. ,   0.], [0. ,   1. ,   0.]]),

array([[1.], [0.], [0.]]),

array([[0., -40.,   0.]]),

array([[1.]]))

Видно, что теперь в состояние системы входят не две, а три переменные. После того, как они будут интерпретированы (аналогично тому, как ранее было сделано для z_1 и z_2), можно запустить моделирование для требуемых стартовых значений и по результатам сделать вывод о том, справился ли регулятор с поставленной задачей.

4. Ручная настройка работает не всегда, но иногда может быть полезна. Рекомендуется сначала выставить P-коэффициент так, чтобы порядок величины на выходе совпал с требуемым, потом – I, чтобы установившаяся величина переходного процесса совпала с требуемой, потом – D, чтобы сгладить переходный процесс.

5. Одним из распространённых методов ручной настройки PID-регулятора является метод Зиглера-Николса. Для настройки по этому методу сначала требуется приближённо определить минимальное значение P, при котором система теряет устойчивость и переходит в режим автоколебаний, при нулевых I и D. Когда это произойдёт (при некоем $P_{кр}$), следует оценить период этих автоколебаний ($T_{кр}$). Тогда можно попытаться использовать следующие коэффициенты регулятора: $P = 0.6P_{кр}$, $I=1.2P_{кр}/T_{кр}$, $D=3P_{кр}T_{кр}/40$.

Контрольные задания и вопросы

1. Какова передаточная функция PID-регулятора? Почему её компонент с коэффициентом D иногда заменяют на другое выражение, например DN/(1 + N/s), где N – большое число?
2. Если вход PID-регулятора очень быстро изменяется, может ли регулятор выдать непропорционально большой отклик? Если да, то какой компонент регулятора - P, I или D - определяет величину этого отклика?
3. Если контролируемая PID-регулятором система не может прийти в требуемое состояние из-за неисправности (механизм заклинило и т.п.), может ли PID-регулятор выдать непропорционально большой отклик? Если да, то какой компонент регулятора - P, I или D - определяет величину этого отклика?

4. Можно ли воздействовать на отрегулированную систему так, чтобы она потеряла устойчивость? Чтобы её выходной сигнал превысил некоторое заранее заданное значение?
5. Как интерпретировать полученное после добавления PID-регулятора состояние системы? Выпишите самостоятельно формулы для полученных переменных состояния. Каковы должны быть значения этих переменных для запуска маятника с заданным отклонением с нулевой начальной угловой скоростью?
6. Настройте регулятор для целевой системы. Удастся ли использовать для этого метод Зиглера-Николса?
7. Как можно получить значения моментов, подаваемые PID-регулятором на обратный маятник, по выходному сигналу системы $\theta(t)$?

Задания для самостоятельной работы

1. По согласованию с преподавателем выберите один из описанных ниже вариантов обратного маятника. Значения коэффициента a, если он имеется в задании, выберите самостоятельно:

   1. Маятник – материальная точка с m = a кг, L=10см, момент силы трения в оси пропорционален сумме угловой скорости вращения с коэффициентом 0.2 и угла отклонения с коэффициентом a;
   2. Маятник – материальная точка с m = 1 кг, L= a м, момент силы трения в оси пропорционален углу отклонения с коэффициентом 0.02;
   3. Маятник – тело с массой a кг, моментом инерции относительно оси 0.1 кг·м^2, центр его тяжести находится на расстоянии 10 см от оси, момент силы трения в оси пропорционален угловой скорости вращения с коэффициентом 0.02;
   4. Маятник – тело с массой 0.3 кг, моментом инерции относительно оси a кг·м^2, центр его тяжести находится на расстоянии 50 см от оси, момент силы трения в оси пропорционален угловой скорости вращения с коэффициентом 0.05;
   5. Маятник – тело с массой 0.4 кг, моментом инерции относительно оси 0.06 кг·м^2, центр его тяжести находится на расстоянии a м от оси, трения нет;
   6. Маятник – материальная точка с m = a кг, L=15 см, момент силы трения в оси пропорционален угловой скорости вращения с коэффициентом 0.002;
   7. Маятник – материальная точка с m = 0.5 кг, L=a м, момент силы трения в оси пропорционален углу отклонения с коэффициентом 2;
   8. Маятник – тело с массой 0.5 кг, моментом инерции относительно оси 0.009 кг·м^2, центр его тяжести находится на расстоянии a м от оси, трения нет;
   9. Маятник – тело с массой 1 кг, моментом инерции относительно оси a кг·м^2, центр его тяжести находится на расстоянии 5 см от оси, момент силы трения в оси пропорционален угловой скорости вращения с коэффициентом 1;
   10. Маятник – тело с массой 0.4 кг, моментом инерции относительно оси 0.06 кг·м^2, центр его тяжести находится на расстоянии a м от оси, момент силы трения в оси пропорционален сумме угловой скорости вращения с коэффициентом 0.2 и угла отклонения с коэффициентом a;

2. Вычислите дифференциальное уравнение регулируемой системы, её передаточную функцию. Всюду можно предполагать, что отклонения угла поворота маятника от нуля достаточно малы.

3. Представьте систему с помощью уравнения состояния. Проинтерпретируйте переменные состояния.

4. Промоделируйте свободное (т.е. без включения двигателя) движение маятника из малого ненулевого угла. Упадёт ли при этом маятник?

5. Добавьте в систему PID-регулятор. Проинтерпретируйте изменения в уравнении состояния.

6. Настройте PID-регулятор так, чтобы он возвращал маятник в положение равновесия за время меньше 10 с из всех допустимых отклонений ($\pm 15°$).