Створення і перетворення лінійних стаціонарних моделей

Теоретичні відомості

LTI-моделі (лінійні стаціонарні моделі) можна створювати у виді SS-, TF-, ZPK-об'єктів. Для цього використовуються відповідно процедури-конструктори ss, tf, zpk.

SS-модель

Щоб створити SS-модель, необхідно, перш за все, привести диференціальне рівняння руху динамічної системи до вигляду Коші:

$$\frac{d\mathbf{x}}{dt} = \mathbf{Ax} + \mathbf{Bu}; \qquad (0.1)$$
$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du}.$$

Тут $\mathbf{u}$ – вектор вхідних змінних; $\mathbf{y}$ – вектор вихідних змінних; $\mathbf{x}$ – вектор змінних станів системи.

В даному випадку за вихідний вектор $\mathbf{y}$ приймемо:

$$\mathbf{y} = [y_1, y_2], \qquad (0.2)$$

а за вхідний вектор

$$\mathbf{u} = [u_1, u_2]. \qquad (0.3)$$

Припустимо, що

$$\mathbf{x} = [x_1, x_2], \qquad (0.4)$$

а значення вказаних матриць нехай задані наступними:

$$\mathbf{A} = \begin{bmatrix} -0.09 & 0.01 \\ 1 & 0 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 1 & -7 \\ 0 & -2 \end{bmatrix}; \mathbf{C} = \begin{bmatrix} 0 & 2 \\ 1 & -5 \end{bmatrix}; \mathbf{D} = \begin{bmatrix} -3 & 0 \\ 1 & 0 \end{bmatrix}. \qquad (0.5)$$

Введемо ці матриці в командному вікні MATLAB:

```
>>A=[-0.09   -0.01
    1         0];     % Матриця системи.

B=[ 1   -7
    0   -2];          % Матриця входу.

C=[ 0    2
    1   -5];          % Матриця виходу.

D=[-3    0
    1    0];          % Матриця обходу.
```

Тепер можна приступати до створення LTI-об'єкту під назвою GYROss, використовуючи модель в просторі станів:

```
>> GYROss=ss(A,B,C,D)

a =
             x1      x2
   x1   -0.09   -0.01
   x2       1       0
```

```
b =
       u1  u2
   x1   1  -7
   x2   0  -2


c =
       x1  x2
   y1   0   2
   y2   1  -5


d =
       u1  u2
   y1  -3   0
   y2   1   0

Continuous-timemodel.
```

TF-модель

Як бачимо модель сформована правильно. Можна розпочати деякі її перетворення. Спочатку знайдемо передавальні функції створеної системи. Очевидно, їх повинно бути чотири (в нашому випадку два входи і два виходи). Для цього використаємо функцію перетворення tf:

```
>>GYROtf=tf(GYROss)

Transferfunctionfrominput 1 tooutput...
-3 s^2 - 0.27 s + 1.97
 #1:  ----------------------
       s^2 + 0.09 s + 0.01

      s^2 + 1.09 s - 4.99
 #2:  -------------------
      s^2 + 0.09 s + 0.01

Transfer function from input 2 to output...
         -4 s - 14.36
 #1:  -------------------
      s^2 + 0.09 s + 0.01

          3 s + 35.92
 #2:  -------------------
      s^2 + 0.09 s + 0.01
```

ZPK-модель

Тепер перетворимо введену SS-модель в ZPK-модель, використовуючи для цього процедуру zpk:

```
>> GYROpk=zpk(GYROss)

Zero/pole/gain from input 1 to output...
      -3 (s+0.8566) (s-0.7666)
 #1:  ------------------------
       (s^2  + 0.09s + 0.01)

       (s+2.844) (s-1.754)
 #2:  ---------------------
      (s^2  + 0.09s + 0.01)

Zero/pole/gain from input 2 to output...
           -4 (s+3.59)
 #1:  ---------------------
      (s^2  + 0.09s + 0.01)

           3 (s+11.97)
```

```
#2:  --------------------
     (s^2  + 0.09s + 0.01)
```

Інформація про модель

Щоб отримати окремі характеристики (матриці та вектори, що описують простір станів, коефіцієнти чисельника і знаменника передавальної функції, тощо) створеної моделі, можна використовувати одну з наступних процедур: `tfdata` – для того, щоб отримати чисельник і знаменник передавальної функції системи, `ssdata`– значення матриць рівнянь простору станів, `zpkdata` – вектор значень полюсів і нулів системи. Наприклад:

```
>> [nom,den]=tfdata(GYROtf, 'v')
>> [A,B,C,D]=ssdata(GYROss)
>> [z,p,k]=zpkdata(GYROzp)
```

Процедура get дає можливість отримати повну характеристику моделі, включаючи імена входів і виходів, значення кроку дискретизації і т.д. Наприклад:

```
>>get(GYROss)
```

Про число виходів та входів системи можна взнати, якщо звернутися до процедури size:

```
>>size(GYROss)
```

Аналіз системи

Пакет ControlSystemToolbox надає широкий набір процедур, які здійснюють аналіз системи автоматичного управління з різних точок зору і визначають відгуки системи на зовнішні чинники як у часовій, так і в частотній областях.

Для знаходження часових відгуків системи на деякі зовнішні впливи передбачені такі функції:

- impulse – визначення відгуку системи на одиничний імпульсний вплив (рис.6.2);
- step – визначення реакції системи на одиничний скачок вхідного впливу (рис.6.1);
- initial – визначення власного руху системи при довільних початкових умовах;
- lsim – визначення реакції системи на вхідний вплив довільної форми, який задається вектором його значень в часі.

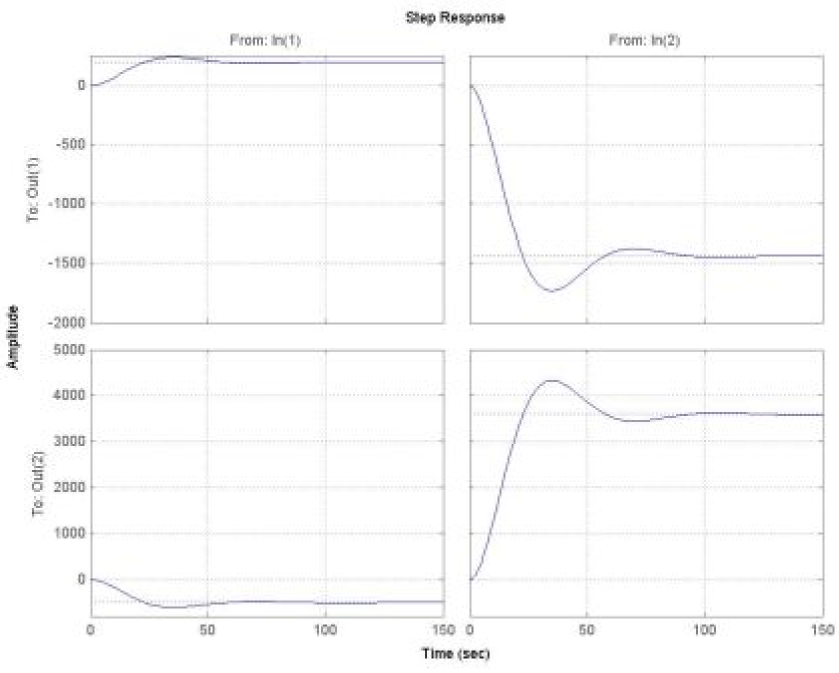

Рис. 0.1. Відгук системи на одиничний скачок вхідного сигналу `step(GYROss),grid`

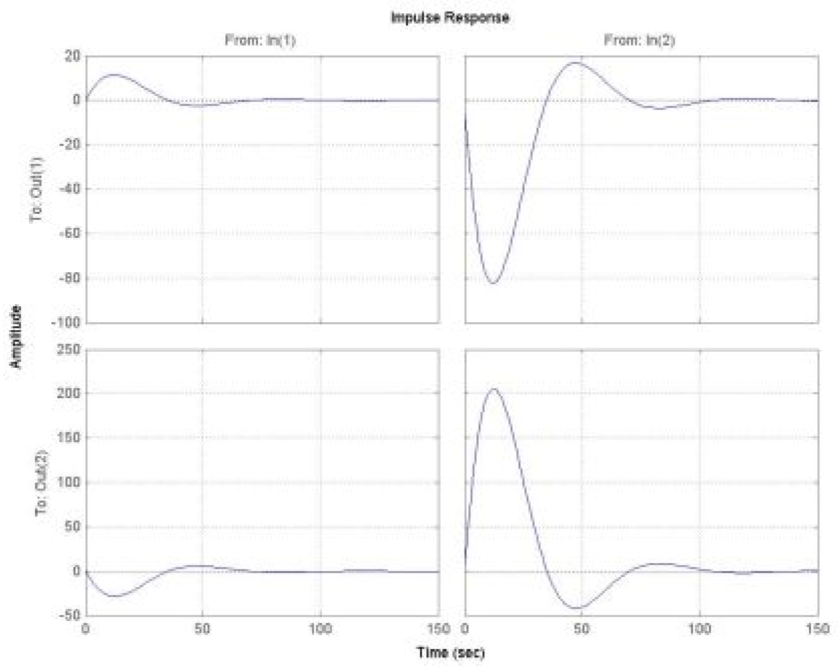

Рис. 0.2. Відгук системи на одиничний імпульсний вплив `impulse(GYROss),grid`

Наведена нижче група процедур відображає в частотній області реакцію системи на зовнішні гармонічні впливи. До таких процедур відносять наступні:

- bode – будує графіки амплітудно-частотної характеристики та фазово-частотної характеристики (діаграму Боде) вказаної системи.
- nyquist– будує в комплексній площині графік амплітудно-фазової характеристики системи в полярних координатах;
- nichols – будує карту Ніколса системи, тобто графік амплітудно-фазової характеристики розімкнутої системи в декартових координатах;
- sigma – будує графік залежності від частоти сингулярних значень системи (зазвичай співпадає з амплітудно-частотною характеристикою;
- margin – будує діаграму Бодеі вказує запаси по амплітуді і по фазі.

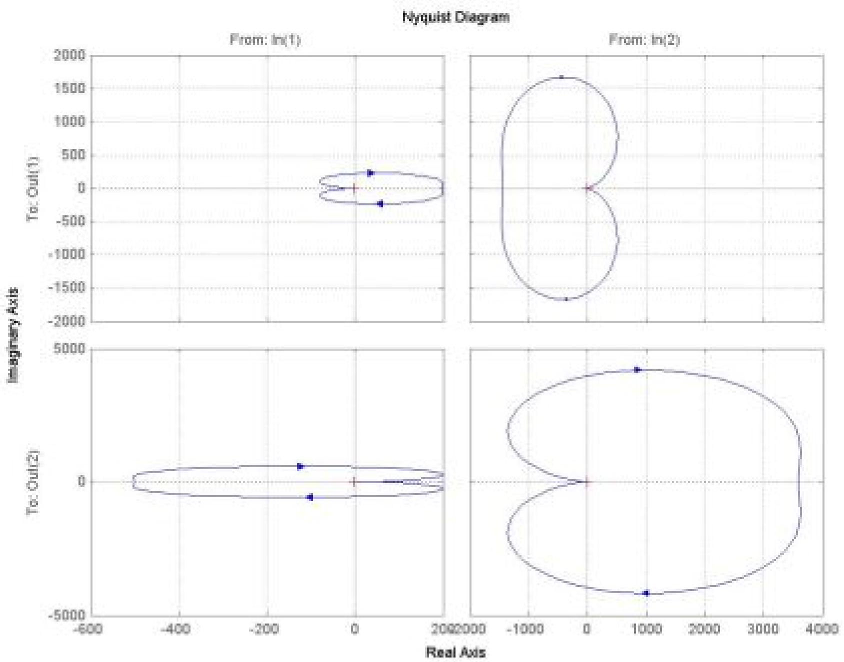

Рис. 0.3. Діаграма Найквіста системи `nyquist(GYROss),grid`

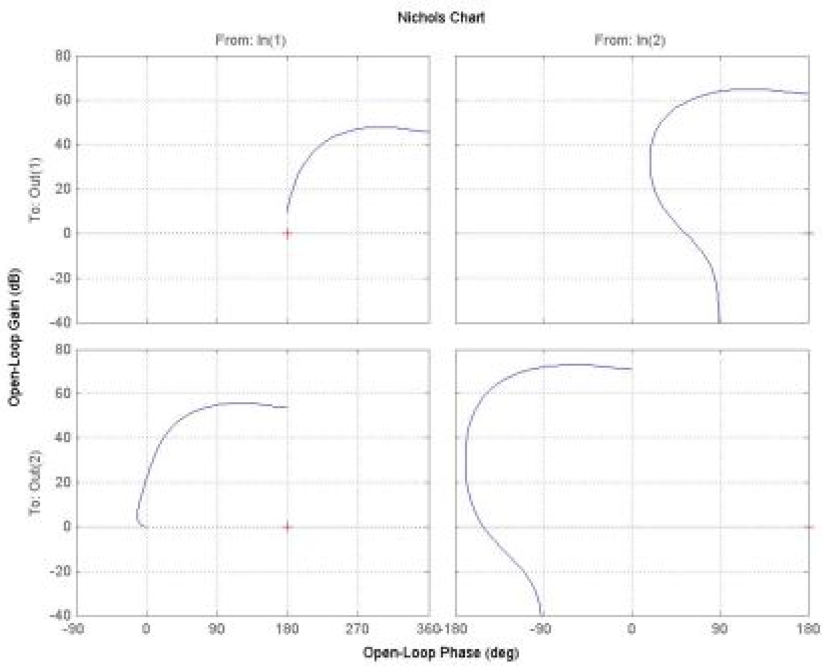

Рис. 0.4. Карта Ніколса розімкнутої системи `nichols(GYROss),grid`

LTIViewer

Після вводу в командному вікні MATLAB команди ltiviewна екрані появиться вікно LTIView. В цьому вікні можна «будувати» в інтерактивному режимі практично всі вищеперечислені графіки, причому одночасно для декількох систем.

Роботу з LTIView слід починати з завантаження в його середовище тих LTI-об'єктів , які необхідно аналізувати. Для цього необхідно скористатися командою File => Import. В результаті на екрані появиться діалогове вікно ImportSystemData(Імпорт даних системи).

Після завантаження LTI-об'єктів, необхідно визначити кількість і вид графіків, які виводяться в вікно LTIView. Для цього необхідно виконати команду Edit => Plot Configuration. На екрані появиться діалогове вікно PlotConfiguration (конфігурації графіків) (рис. 6.5).

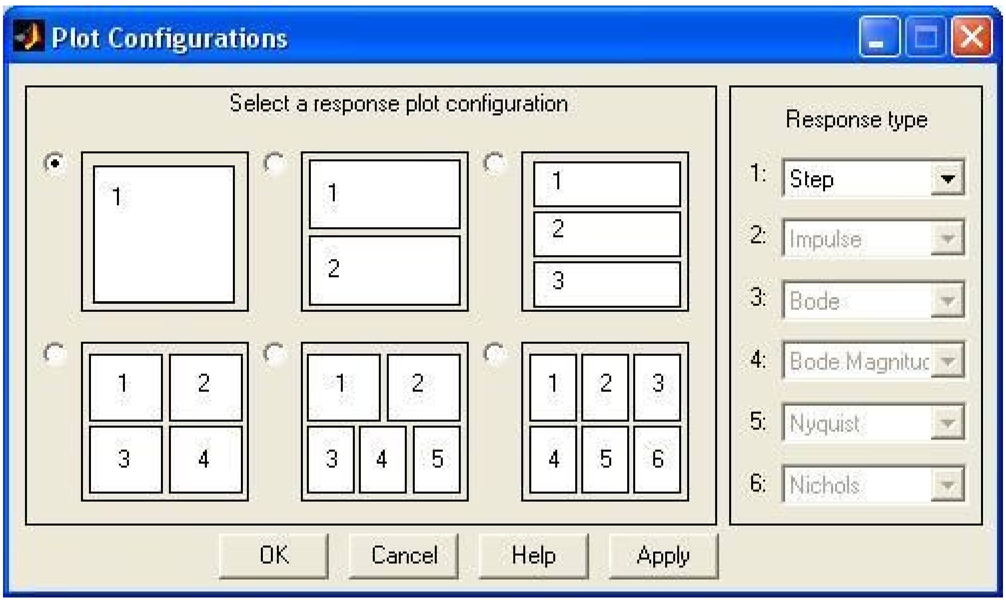

Рис. 0.5. Діалогове вікно PlotConfiguration

Завдання

Відповідно до заданого варіанту створити та проаналізувати SS-модель, що відповідає диференціальному рівнянню руху динамічної системи. При $g = 9.8$ та $m = 5$.

Варіант 1.

$$mgy''(x) - \frac{m}{2}y'(x) - \frac{g}{m^2+2}y(x) = 8.$$

Варіант 2.

$$mgy''(x) + y'(x) - \frac{m}{g}y(x) = \frac{m+1}{g}.$$

Варіант 3.

$$mgy''(x) + gy'(x) - \frac{6}{2g^2+1}y(x) = 6m + 0.5.$$

Варіант 4.

$$mgy''(x) - \frac{1}{m}y'(x) = -\frac{2}{mg}.$$

Варіант 5.

$$mgy''(x) + y'(x) - \frac{m}{\cos^2 g}y(x) = \frac{1}{\cos^2 mg}.$$

Варіант 6.

$$mgy''(x) - 2my'(x) - 2gy(x) = -3xe^{mg}.$$

Варіант 7.

$$mgy''(x) + 2gy'(x) - \frac{4}{m}y(x) = 2.$$

Варіант 8.

$$mgy''(x) - \frac{6m}{3g^2-0.5}y'(x) - \frac{1}{mg}y(x) = 0.5 - m^2.$$

Варіант 9.

$$mgy''(x) + \frac{1}{m}y'(x) = \frac{1}{g}.$$

Варіант 10.

$$mgy''(x) + 2my'(x) - y(x) = 2(m^2+1)\cos g.$$

Варіант 11.

$$mgy''(x) + y'(x) - 2y(x) = 3e^{m}.$$

Варіант 12.

$$mgy''(x) - 2\operatorname{tg} m\, y'(x) = m + g.$$

Варіант 13.

$$mgy''(x) - y'(x) - gy(x) = -2e^{m}.$$

Варіант 14.

$$mgy''(x) + 2my'(x) - y(x) = 2(g^{2} + 1)\sin m.$$

Варіант 15.

$$mgy''(x) - m^{2}y'(x) - \frac{2}{g^{2}}y(x) = m - g.$$