



KATEDRA MECHANIKI STOSOWANEJ
Wydział Mechaniczny
POLITECHNIKA LUBELSKA

INSTRUKCJA DO ĆWICZENIA NR 5

PRZEDMIOT	Modelowanie w biomechanice
TEMAT	<i>Metody analizy układów nieliniowych: przebiegi czasowe, portrety fazowe, FFT. Wpływ warunków początkowych i wielkości kroku całkowania.</i>

CEL ĆWICZENIA

Wykonanie schematu blokowego nieliniowego modelu w programie Matlab. Zapoznanie się z podstawowymi właściwościami układów nieliniowych.

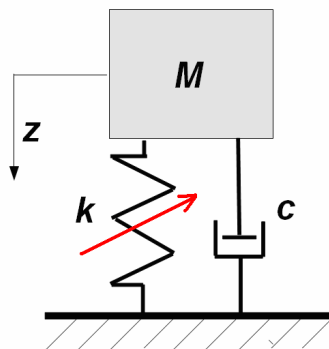
PODSTAWY TEORETYCZNE

Testowany dotychczas (ćwiczenia 1, 2, 3) liniowy model układu biomechanicznego o jednym stopniu swobody opisano równaniem różniczkowym:

$$M\ddot{z} + kz + c\dot{z} - Mg = 0 \quad (1)$$

Modele liniowe są powszechnie używane w modelowaniu systemów, ale zakres ich stosowalności jest ograniczony. Wynika to z faktu, że obiekty rzeczywiste z natury mają charakter nieliniowy. Nieliniowości mogą być spowodowane różnymi przyczynami, np. zmianą sztywności wraz ze wzrostem odkształcenia elementu podatnego. Mięśnie w organizmie ludzkim nie mogą wydłużać się w nieskończoność. Zależność pomiędzy obciążeniem a wydłużeniem mięśni jest proporcjonalna tylko w pewnym zakresie. Poza nim stałe przyrosty obciążenia wywołują stopniowo zmniejszające się wydłużenie mięśni. W wykonywanym ćwiczeniu do opisu nieliniowych właściwości mięśni kończyn dolnych wykorzystano zmodyfikowany model:

$$M\ddot{z} + kz + \alpha kz^3 + c\dot{z} - Mg = 0 \quad (2)$$



Rys.1. Nieliniowy model o jednym stopniu swobody układu biomechanicznego.

MATERIAŁY CHRO

```

1 - M=80; % masa czlowieka
2 - k=9600; % wspolczynnik sztywnosci
3 - c=100; % wspolczynnik tlumienia
4 - a=20; % wspolczynnik nieliniowosci
5 - t=[0,3]; % definicja czasu startu i konca symulacji
6 - options=simset('AbsTol',1e-6, 'RelTol', 1e-6, 'Maxstep', 0.001); % okreslenie opcji
7 - figure % otwiera nowe okno graficzne
8 - hold on
9 - h=0.25; % minimalna wysokosc przeszkody
10 - v=sqrt(2*9.81*h); % warunek poczatkowy
11 - [T,Y]=sim('nmodel',t, options); % uruchomienie symulacji w Simulink
12 - plot(T,Y(:,1),'r') % wykreslenie przebiegu czasowego, czerwona seria
13 - h=0.5; % posrednia wysokosc przeszkody
14 - v=sqrt(2*9.81*h); % warunek poczatkowy
15 - [T,Y]=sim('nmodel',t, options); % uruchomienie symulacji w Simulink
16 - plot(T,Y(:,1),'b') % wykreslenie przebiegu czasowego, niebieska seria
17 - h=0.75; % posrednia wysokosc przeszkody
18 - v=sqrt(2*9.81*h); % warunek poczatkowy
19 - [T,Y]=sim('nmodel',t, options); % uruchomienie symulacji w Simulink
20 - plot(T,Y(:,1),'g') % wykreslenie przebiegu czasowego, zielona seria
21 - h=1; % maksymalna wysokosc przeszkody
22 - v=sqrt(2*9.81*h); % warunek poczatkowy
23 - [T,Y]=sim('nmodel',t, options); % uruchomienie symulacji w Simulink
24 - plot(T,Y(:,1),'k') % wykreslenie przebiegu czasowego, czarna seria
25
26 - saveas(gcf, 'rysunek1.tif') % zapisujemy rysunek

```

Na podstawie otrzymanych rysunków określić maksymalną wartość z_{max} oraz okres drgań $T_{Drgań}$ w zależności od warunków początkowych (wysokości przeszkody h).

2. **Wpływ wielkości kroku całkowania.** Podczas symulacji numerycznych wykorzystywane mogą być algorytmy obliczeniowe ze stałą lub zmienną długością kroku całkowania. Jedną z metod całkowania ze stałym krokiem jest ode4. Charakteryzuje się ona tym, że dokładność i czas symulacji zależą od długości zastosowanego kroku całkowania. Im jest on dłuższy to precyzja obliczeń jest gorsza, natomiast czas trwania procesu obliczeniowego jest krótszy. Istnieje graniczna wartość kroku całkowania Δt_{max} , taka że zastosowanie mniejszych długości kroku nie będzie powodowało pogorszenia precyzji obliczeń. W realizowanym ćwiczeniu wykonać symulacje numeryczne na podstawie zaproponowanego skryptu. Analizę sporządzić dla układu liniowego $a=0$ oraz nieliniowego $a=20$. Zmieniając wielkość kroku całkowania $kk1$ znaleźć maksymalną jego wartość Δt_{max} , aby wyświetlane portrety fazowe były identyczne.

```

1 - M=80; % masa czlowieka
2 - k=9600; % wspolczynnik sztywnosci
3 - c=100; % wspolczynnik tlumienia
4 - a=0; % wspolczynnik nieliniowosci
5 - t=[0,2]; % definicja czasu startu i konca symulacji
6 - h=1; % maksymalna wysokosc przeszkody
7 - v=sqrt(2*9.81*h); % warunek poczatkowy
8 - figure % otwiera nowe okno graficzne
9 - hold on
10 - kk1=0.1; % pierwsza wartosc kroku calkowania
11 - options=simset('Solver','ode4', 'Fixedstep', kk1); % okreslenie opcji
12 - [T,Y]=sim('nmodel',t, options); % uruchomienie symulacji w Simulink
13 - plot(Y(:,1),Y(:,2),'b') % wykreslenie portretu fazowego, niebieska seria
14
15 - kk2=0.00001; % druga wartosc kroku calkowania
16 - options=simset('Solver','ode4', 'Fixedstep', kk2); % okreslenie opcji
17 - [T,Y]=sim('nmodel',t, options); % uruchomienie symulacji w Simulink
18 - plot(Y(:,1),Y(:,2),'k') % wykreslenie portretu fazowego, czarna seria
19
20 - saveas(gcf, 'rysunek1.tif') % zapisujemy rysunek

```

3. **Analiza sygnału w dziedzinie częstotliwości.** Sygnały można analizować w dziedzinie czasu (przebiegi czasowe, portrety fazowe) lub częstotliwości. Jedną z metod analiz częstotliwościowych jest szybka transformata Fouriera (FFT). Jest ona podstawowym narzędziem w analizie harmonicznego sygnału. Prezentowany kod wykonuje jednocześnie analizę porównawczą sygnałów modeli liniowego i nieliniowego.

```

1 - M=80; % masa człowieka
2 - k=9600; % współczynnik sztywności
3 - c=100; % współczynnik tłumienia
4 - t=[0,8]; % definicja czasu startu i końca symulacji
5 - h=1; % wysokość przeszkody
6 - v=sqrt(2*9.81*h); % warunek początkowy
7 - dt=0.0001; % wartość kroku całkowania
8 - options=simset('Solver','ode4', 'Fixedstep', dt); % określenie opcji
9 - figure % otwiera nowe okno graficzne
10 - hold on
11 - A=[0, 20];
12 - for i=1:2 % pętla sterująca
13 - a=A(i); % współczynnik nieliniowości
14 - [T,Y]=sim('nmodel',t, options); % uruchomienie symulacji w Simulink
15 - n=length(T); % długość wektora czasu
16 - fw=fft(Y(:,1)); % szybka transformata Fouriera
17 - fw(1)=[];
18 - nn=length(fw);
19 - frequency=inv(dt)*(0:n/2-1)/n; % os częstotliwości
20 - amplitude=2*abs(fw(1:nn/2))./nn;
21 - plot(frequency,amplitude, 'linewidth',i) % wykreślenie fft
22 - xlim([0 5])
23 - end
24
25 - saveas(gcf, 'rysunekf1.tif') % zapisujemy rysunek

```

Na podstawie otrzymanych rysunków dla różnych wysokości przeszkody h określić odległość „pików” (maksimów charakterystyk) Δf .

MATERIAŁY CHRONIONE

OPRACOWANIE WYNIKÓW

Po przeprowadzeniu symulacji numerycznych należy uzupełnić tabelę pomiarową.

I etap					
Nr próby	h (m)	Model liniowy		Model nieliniowy	
		z_{max} (m)	$T_{Drgań}$ (s)	z_{max} (m)	$T_{Drgań}$ (s)
1					
2					
3					
4					
II etap					
Model liniowy	Δt_{max} (s)		Model nieliniowy	Δt_{max} (s)	
III etap					
Nr próby	1	2	3	4	
h (m)					
Δf (Hz)					

SPRAWOZDANIE

Sprawozdanie z ćwiczenia powinno zawierać:

1. Tabelkę identyfikacyjną.
2. Cel ćwiczenia.
3. Schemat blokowy modelu układu w programie Matlab.
4. Tabelę pomiarów i wyników.
5. Obliczenia i wykresy $z_{max}(h)$, $T_{Drgań}(h)$, $\Delta f(h)$.
6. Wnioski.

Uwaga.

Studenci przystępujący do odrobienia ćwiczenia laboratoryjnego powinni posiadać odręcznie przygotowany protokół, który musi zawierać tabelę pomiarów i wyników.