

Modelowanie urządzenia hamującego samolot

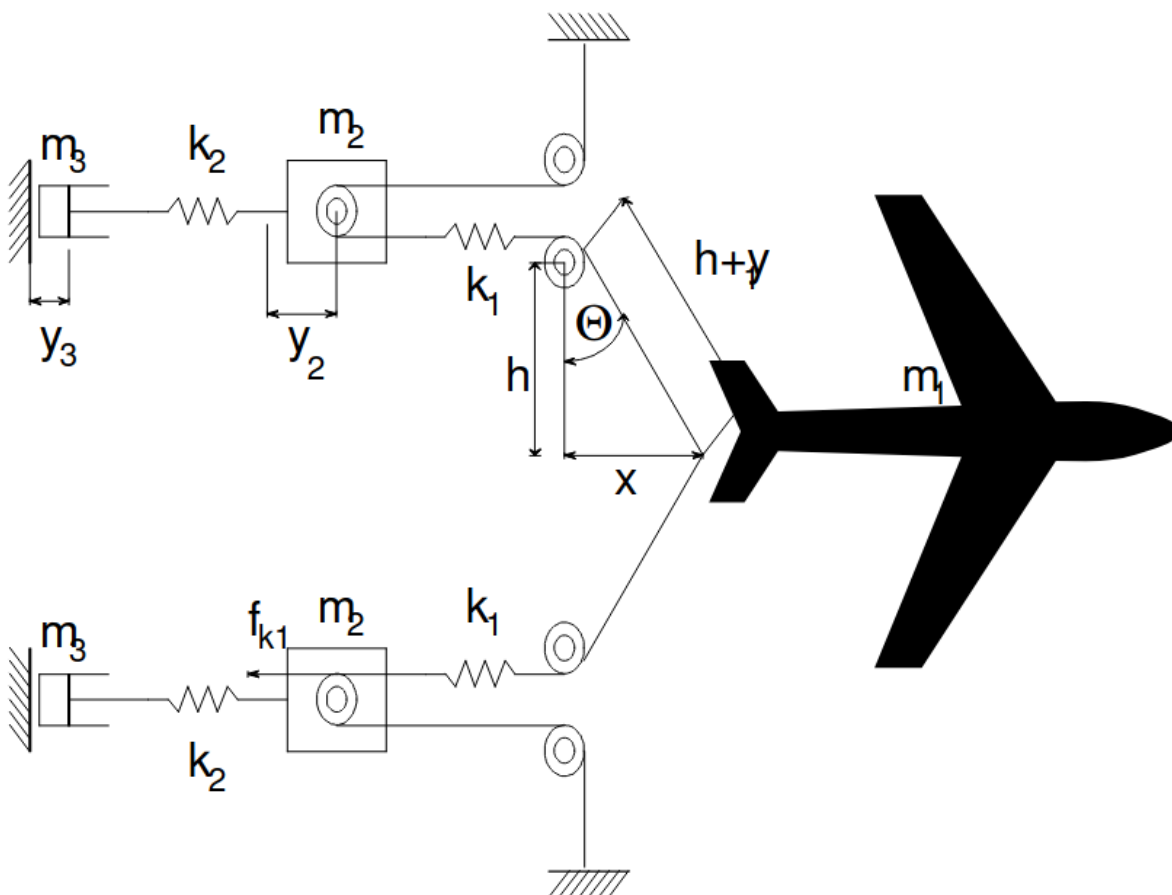
Mateusz Wójcik, 6.11.2024

Cel wiczenia:

Celem poniższego wiczenia jest implementacja modelu urządzenia hamującego samolot w środowisku SIMULINK, bazując na podanym w konspekcie modelu matematycznym. W czasie realizacji zadania wystąpi konieczność zapoznania się z zasadą działania nowego bloku pozwalającego interpolować funkcję na podstawie tabeli punktów.

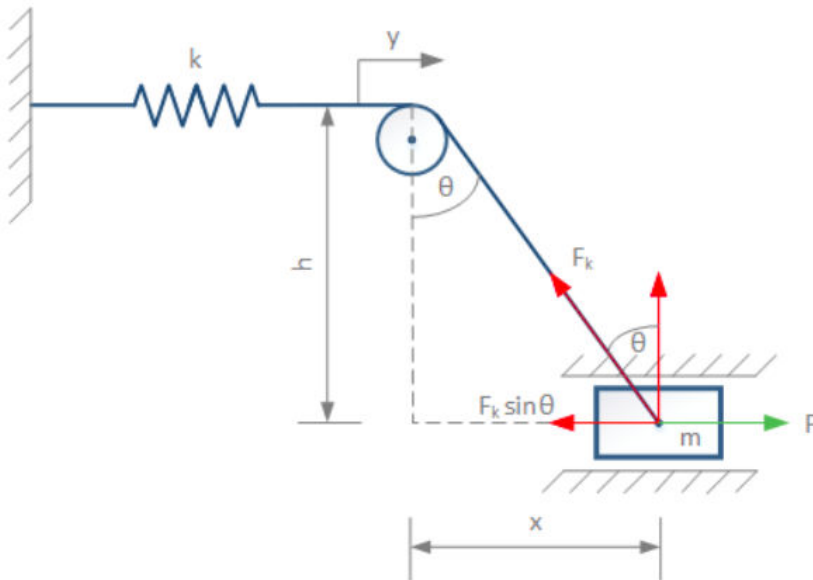
Część teoretyczna

Model hamownika jest reprezentowany przez następujący schemat:



Można zauważyć, że instalacja składa się z dwóch symetrycznych części. Można zatem skorzystać z dwóch podejść, albo założyć, że elementy są idealnie symetryczne i ostatecznie wyniki pomnożyć przez dwa, albo uwzględnić w swoim modelu możliwość odchyłek, tzn. różnic w współczynnikach sprężystości poszczególnych elementów, czy mas. W zaproponowanym rozwiązaniu skorzystano z drugiej opcji pozwalającej na badanie różnych przypadków, zmieniając jedynie zmienne zdefiniowane w tym skrypcie.

W celu ułatwienia analizy i tworzenia układu, modelowanie podzielono na trzy części, wraz z każdą zwikszczać dokładność odwzorowania problemu. Analizę rozpoczyna się od prostego modelu:



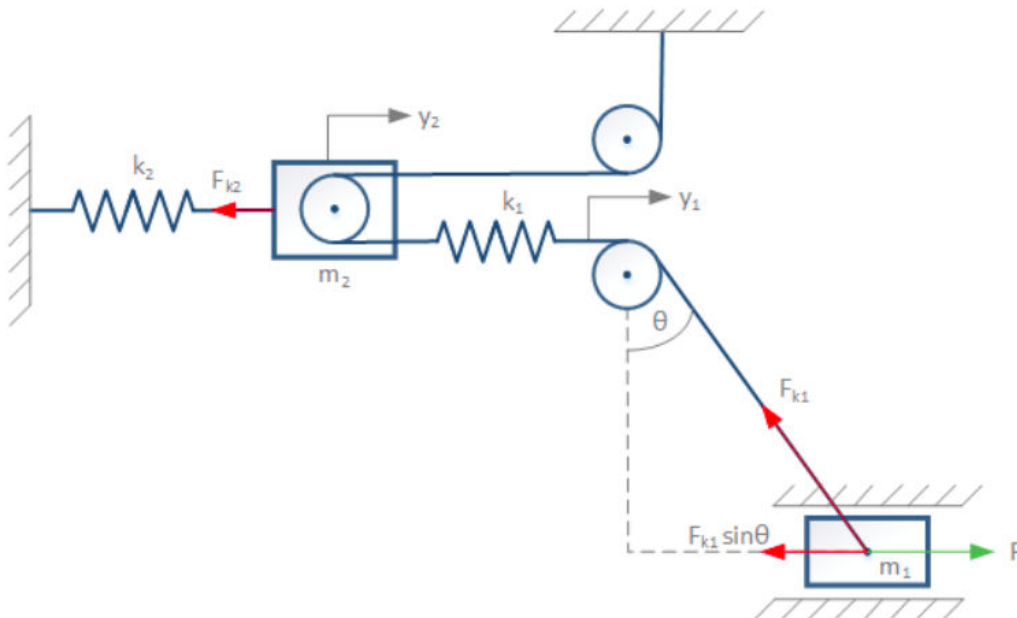
Model składa się z masy, która może poruszać się jedynie poziomo i połączonej za pomocą bloczka ze sprężyną. Dla powyższego układu jesteśmy w stanie napisać równanie:

$$m\ddot{x} = -F_k \sin \theta,$$

które po uwzględnieniu zależności trygonometrycznych można przekształcić do postaci:

$$m\ddot{x} = -k(\sqrt{h^2 + x^2} - h) \frac{x}{\sqrt{h^2 + x^2}},$$

W dalszej części skomplikowano układ rozważając model z bloczkiem przesuwalnym:



Na podstawie schematu zapisano równanie siły działającej na sprężynę k_1 :

$$F_{k1} = \begin{cases} k_1(y_1 - 2y_2), & y_1 \geq 2y_2 \\ 0, & y_1 < 2y_2 \end{cases}$$

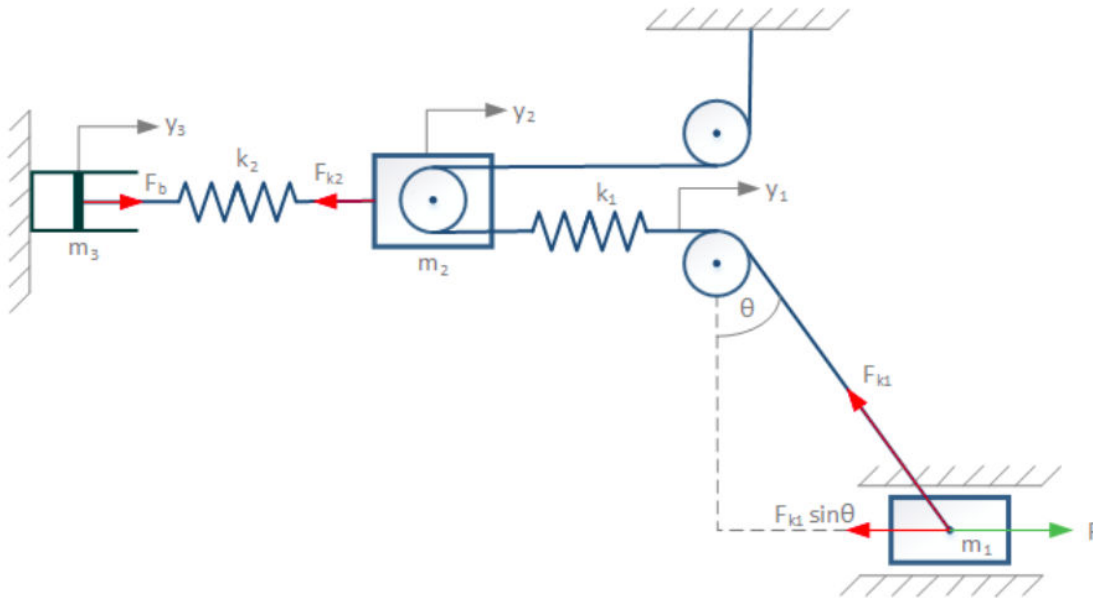
Zatem po modyfikacji wcześniejszego równania otrzymujemy:

$$m_1 \ddot{x} = -F_{k1} \frac{x}{\sqrt{l^2 + x^2}},$$

Jednak rozwijając ten model musimy jeszcze napisać równanie opisujące zmienną y_2 :

$$m_2 \ddot{y}_2 = -k_2 y_2 + 2F_{k1}.$$

Ostatnim brakującym elementem jest tłumik, który pozwoli na otrzymanie efektywnego hamowania, bez generacji przyspieszenia gwałtownością pilota. Ostateczny układ można przedstawić w postaci schematu poniżej:



W ostatecznym modelu równanie ruchu masy m_1 pozostaje bez zmian:

$$m_1 \ddot{x} = -F_{k1} \frac{x}{\sqrt{l^2 + x^2}},$$

gdzie:

$$F_{k1} = \begin{cases} k_1(y_1 - 2y_2), & y_1 \geq 2y_2 \\ 0, & y_1 < 2y_2 \end{cases}$$

Równanie ruchu masy m_2 przyjmuje za to postać:

$$m_2 \ddot{y}_2 = 2F_{k1} - F_{k2},$$

gdzie:

$$F_{k2} = \begin{cases} k_2(y_2 - y_3), & y_2 \geq y_3 \\ 0, & y_2 < y_3 \end{cases}$$

Aby opisać kolejny ruchomą masę dodaną do układu należy ułożyć jeszcze jedno równanie:

$$m_3 \ddot{y}_3 = F_{k2} - F_b,$$

gdzie F_b jest nieliniowa i wyraża się zależnością:

$$F_b = f(y_3) \dot{y}_3,$$

a funkcja $f(v_3)$ została zweryfikowana do wiadczalnie i przedstawiona w postaci danych tabelarycznych przedstawionych w konspekcie.

Implementacja w Simulinku

Posiadając wszelkie niezbędne zależności przystąpiono do tworzenia modelu w środowisku Simulink. Etap rozpoczął się od zdefiniowania stałych i zamiany jednostek na jednostki układu SI.

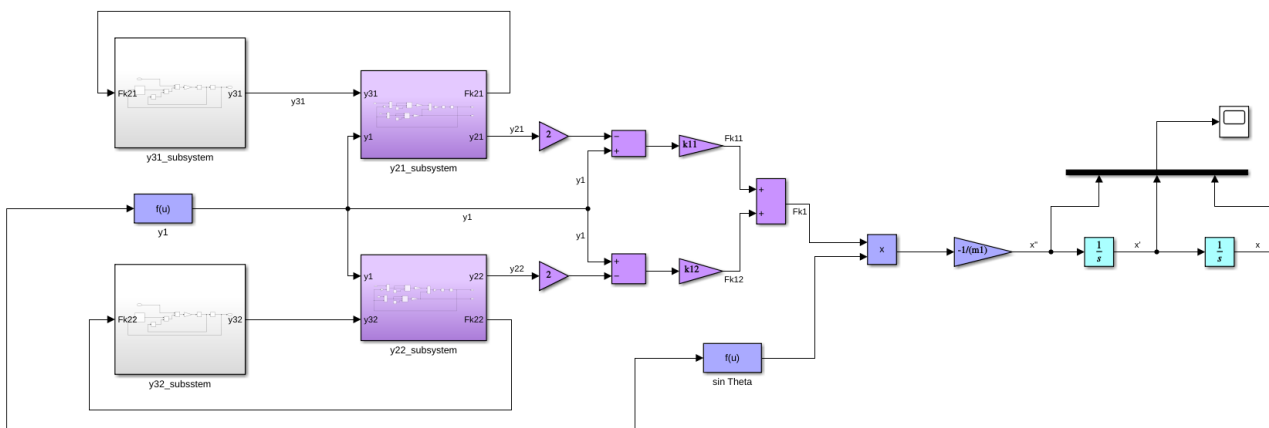
Definiowanie zmiennych

```
m1=14000;      %kg
m21=450.28;    %kg
m22=450.28;    %kg
m31=200;       %kg
m32=200;       %kg
K1=54.7;       %kN/m
K2=303.6;      %kN/m
h=42;          %m
k11 = K1 * 1000;
k12 = K1 * 1000;
k21 = K2 * 1000;
k22 = K2 * 1000;
```

Model w Simulinku

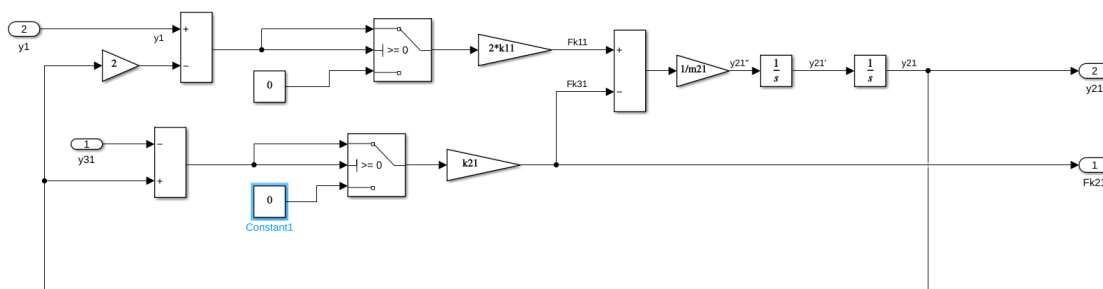
Tworząc model skorzystano z tworzenia subsystemów, w celu zmniejszenia komplikacji wizualnej modelu oraz umożliwienia łatwej duplikacji kodu zmieniając dane opisujące poszczególne strony hamownika. W celu odróżnienia od siebie poszczególnych sekcji hamownika użyto formatowania kolorami:

- jasnoniebieski - integratory opisujące zachowanie zmiennej x ,
- ciemnoniebieski - pierwsza część wyczerpania, elementy najprostszego modelu,
- fioletowy - części reprezentujący blok ruchomy,
- biały - podukład reprezentujący działanie tłumika.

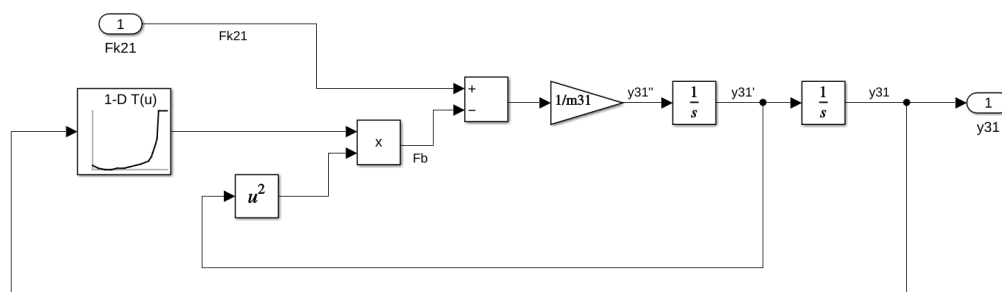


Podczas tworzenia modelu zastała potrzeba wykonywania różnych operacji matematycznych. W celu otrzymania mnożenia ze sobą odpowiednich sygnałów użyto oferowanego przez środowisko Simulink bločku **Product**. Oprócz tego dla bardziej złożonych funkcji, jak np. dla tej reprezentującej $\sin\theta$, skorzystano z bloku **Fcn**, który pozwala na zdefiniowanie wewnętrznej jego struktury, własnej funkcji matematycznej, nawet z uwzględnieniem zmiennych zdefiniowanych w skrypcie. Do załączania kolejnych funkcji F_{k1} i F_{k2} użyto natomiast bloku **Switch**.

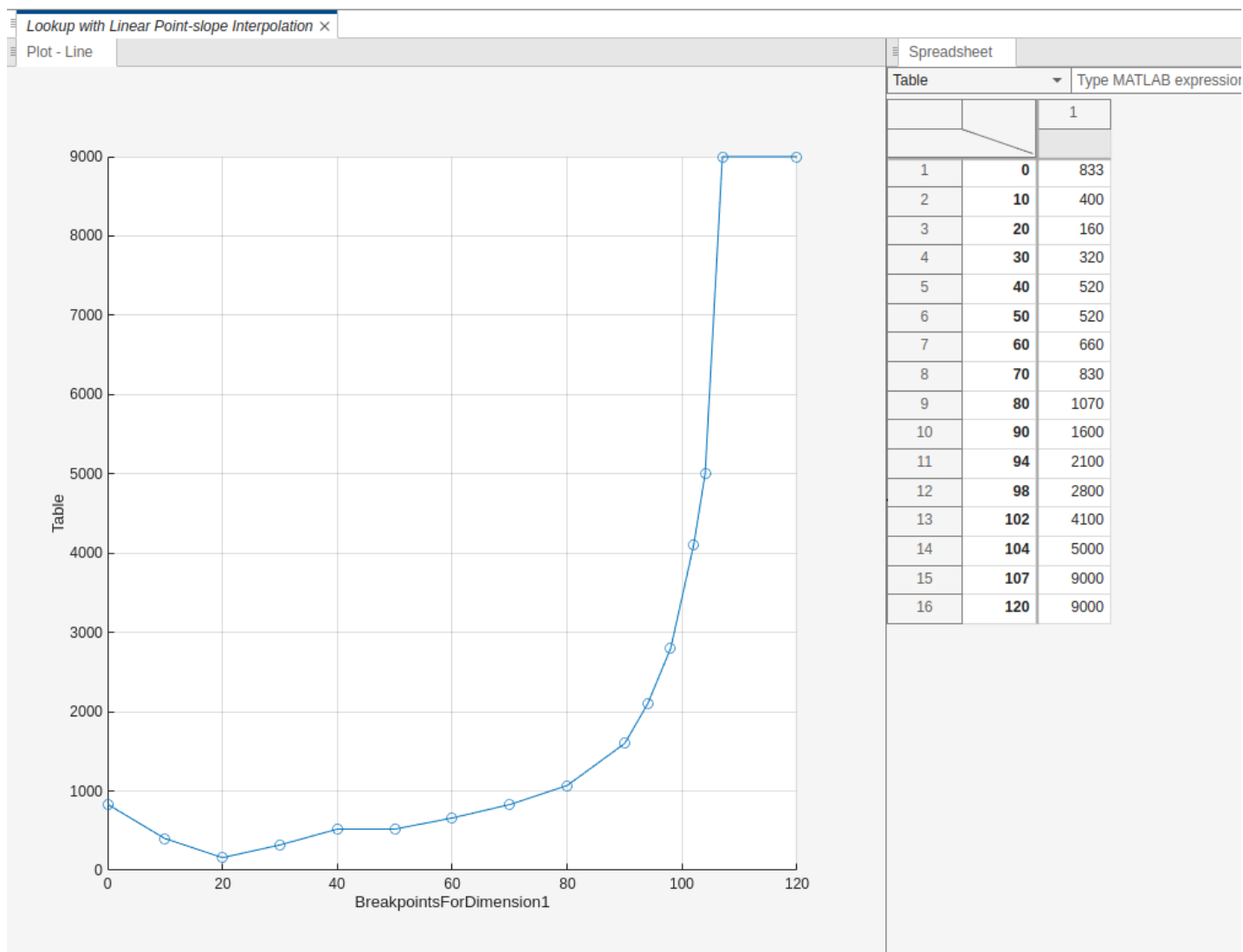
Poniżej przedstawiono strukturę y_21 subsystem. Blok y_22_subsystem jest analogicznie zbudowany, z tym jedynie różnicą, że drugi indeks w stałych jest zamieniony $1 \rightarrow 2$.



Struktura y_31_subsystem została przedstawiona poniżej. Jednym z kluczowych jej elementów jest blok **1-D Lookup Table**, który pozwala wygenerować na podstawie podanych punktów przebieg funkcji opisującej charakterystykę zastosowanego w urządzeniu tłumika. Następnie, podczas symulacji gdy podawana jest na wejście określona wartość sygnału to ten blok na jej podstawie i na podstawie wygenerowanego przebiegu funkcji wystawia na swoje wyjście odpowiedni odpowiedź.



Konfiguracja bloku 1-D Lookup Table, wraz z zdefiniowanymi punktami i wygenerowanym wykresem, została przedstawiona na rysunku poniżej:



Wszystkie omówione dotychczas elementy pozwalają na wykonanie odpowiednich obliczeń numerycznych w środowisku SIMULINK, pozwalając uzyskać wyniki dla symulowanego modelu.

Wyniki symulacji

Na podstawie zbudowanego modelu przeprowadzono symulację, której wyniki zobrazowano na poniższych wykresach:

```
figure("Position",[100 100 1000 1000])
subplot(311);
plot(out.tout, out.x{1}.Values.Data(:,3))
title("x")
xlabel("Czas [s]")
ylabel("Położenie [m]")
grid on

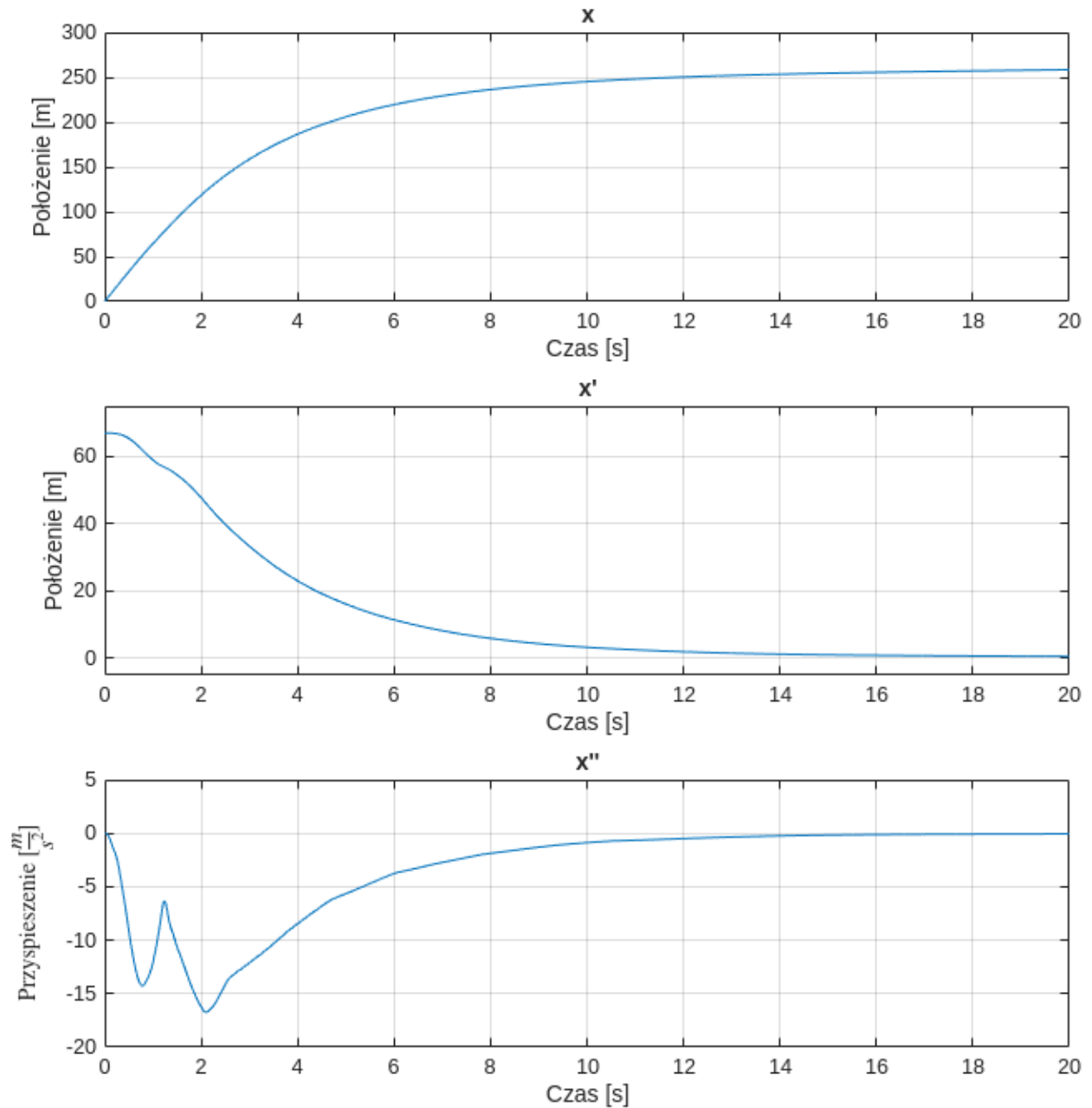
subplot(312);
plot(out.tout, out.x{1}.Values.Data(:,2))
title("x'")
xlabel("Czas [s]")
ylabel("Położenie [m]")
```

```

grid on
axis([0 20 -5 75])

subplot(313);
plot(out.tout, out.x{1}.Values.Data(:,1))
title("x'")
ylabel("Przyspieszenie  $[\frac{m}{s^2}]$ ", Interpreter="latex")
grid on
axis([0 20 -20 5])
xlabel("Czas [s]")

```



W wyniku symulacji otrzymano trzy wykresy, gdzie kolejno przedstawiono zależność :

1. Przemieszczenia samolotu w funkcji czasu,
2. Prędkość samolotu w funkcji czasu,
3. Działanie na samolot przyspieszenia w funkcji czasu.

Analizując powyższe charakterystyki zauważono, że odległość hamowania samolotu z wykorzystaniem samego hamownika (nigdzie nie uwzględnialiśmy siły pochodzącej od hamowania przez pilota) wynosi około 250 m. Jest to realny wynik, który można poprawić uwyślijmy dodatkowej siły. Mimo wszystko, należy zwrócić uwagę na wykres przedstawiający działanie na samolot przyspieszenie. Jego wartość nie przekracza $2g$, zatem hamownik zapewnia dla pilota równie przeciwnie, które jest w stanie po odpowiednim przeszkoleniu wytrzymać bez większych problemów. Można więc stwierdzić, że układ o podanych parametrach dobrze spełnia swoje zadanie.

Wnioski

Podczas zajęć zapoznało się z sposobem rozwijania złożonych modeli matematycznych. Szczególnie ważną umiejętnością jest tworzenie prostych modeli, a następnie systematycznie je udoskonalanie, dodając kolejne obiekty, pozwalające na lepsze odwzorowanie rzeczywistości. Jest to umiejętność, na którą zazwyczaj nie istnieje jeden idealny przepis, dlatego każdy przykład jest zawsze wartościowy, ponieważ pozwala rozbudowywać własną intuicję w tym obszarze. Szczególnie wartościowe jest, że temat zajęć dotyczy interesujących obiektów fizycznych, które mają zastosowanie w rzeczywistości, a nie tylko akademickich przykładów. Składa się to na to, że zadania wykonuje się z ciekawości i chęci przetestowania innych danych wejściowych, aby głębiej poznać dany temat.