Laboratorio Matlab – Fondamenti di Automatica

Stabilizzazione di un pendolo inverso– Soluzioni

Innanzitutto occorre scrivere il sistema dinamico nella forma:

$$\dot{x}(t) = Ax(t) + bu(t)$$

$$y(t) = cx(t) + du(t)$$

Per scrivere il sistema dinamico in forma matriciale si può porre

$$x_1(t) = x(t)$$

$$x_2(t) = \dot{x}(t)$$

A questo punto si può scrivere:

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = \frac{1}{M_S}\left(-a_S x_2(t) - k_S x_1(t)\right)$$
$$y(t) = x_1(t)$$

Per studiare la stabilità del sistema, occorre conoscere gli autovalori[1] della matrice $A = \begin{vmatrix} 0 & 1 \\ -\frac{k_S}{M_S} & -\frac{a_S}{M_S} \end{vmatrix}$.

Il polinomio caratteristico del sistema è:

$$det(\lambda I - A) = det\begin{vmatrix} \lambda & -1 \\ \frac{k_S}{M_S} & \lambda + \frac{a_S}{M_S} \end{vmatrix} = \lambda^2 + \frac{a_S}{M_S}\lambda + \frac{k_S}{M_S}$$

da cui si possono calcolare gli autovalori: $\lambda_{1,2} = \frac{-a_S \pm \sqrt{a_S^2 - 4\,M_S\ k_S}}{2M_S}$.

Essendo sia *a_S* che *k_S* positivi, gli autovalori avranno sempre parte reale negativa. Pertanto il sistema è sempre asintoticamente stabile (come doveva essere rappresentando un sistema dissipativo).

Controlliamo numericamente che la soluzione analitica sia corretta. Assegniamo i valori dati alle variabili del problema

```
>> M_s = 250;                       % Kg
>> k_s = 5000;                      % N/m
>> a_s = 1000;                      % N*s/m
```

Ora scriviamo la matrice *A* e valutiamone gli autovalori:

```
>> A=[0 1;
-k_s/M_s -a_s/M_s];

>> autoval=eig(A)

     -2.0000 + 4.0000i
     -2.0000 - 4.0000i
```

[1] Essendo il sistema a tempo continuo del secondo ordine, l'asintotica stabilità è anche garantita dalle condizioni: $tr(A) = -\frac{a_S}{M_S} < 0$ e $det(A) = \frac{k_S}{M_S} > 0$

Quindi, essendo entrambi gli autovalori a parte reale negativa, il sistema risulta asintoticamente stabile.

Per considerare la salita di un passeggero nell'automobile, occorre modificare la seconda equazione di stato introducendo un termine che descrive la forza peso dello stesso, da ripartire per ruota (m rappresenta la massa del passeggero):

$$\dot{x}_2(t) = \frac{1}{M_S}\left(-a_S x_2(t) - k_S x_1(t) - \frac{1}{4} mg\right)$$ [2]

da cui

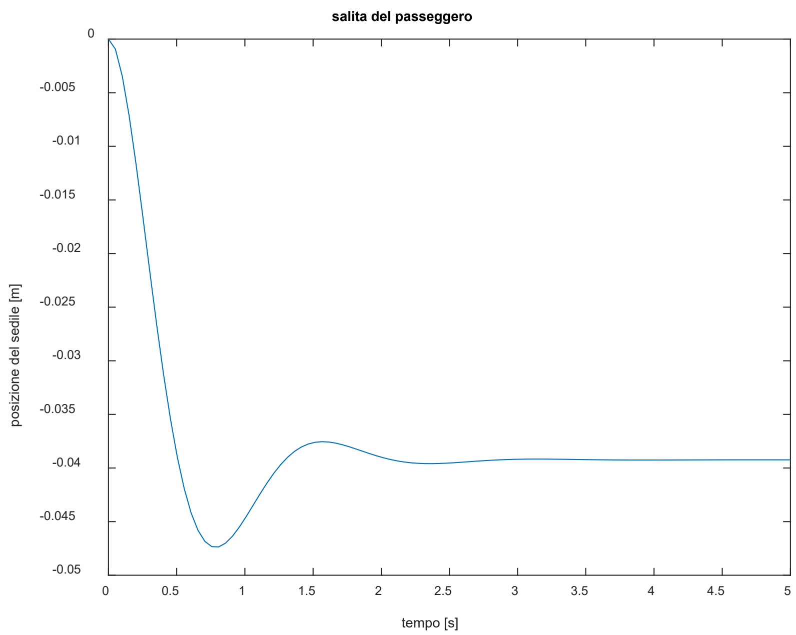
$$A = \begin{vmatrix} 0 & 1 \\ -\frac{k_S}{M_S} & -\frac{a_S}{M_S} \end{vmatrix} \qquad b = \begin{vmatrix} 0 \\ -\frac{mg}{4M_S} \end{vmatrix} \qquad c = |1 \quad 0| \qquad d = 0$$

Valutiamo dapprima il tempo di risposta del sistema (sarà il tempo necessario per andare a regime e ci darà informazione sulla durata della simulazione).

```
>> lambdaD=max(real(autoval));    % parte reale dell'autovalore dominante
>> TD=-1/lambdaD;                 % costante di tempo dominante
>> TR=5*TD                       % tempo di risosta
```

Simuliamo quindi la salita del passeggero, partendo dalla condizione iniziale nulla, per un orizzonte temporale pari a due volte il tempo di risposta del sistema:

```
>> m=80; g=9.81;
>> b=[0 -m*g/(4*M_s)]'; c=[1 0]; d=0;
>> sys_sedile=ss(A,b,c,d); [3]
>> T=linspace(0,2*TR,100); U=ones(size(T));
>> pos_sedile=lsim(sys_sedile,U,T); [4]
>> figure; plot(T,pos_sedile);
>> xlabel('tempo [s]'); ylabel('posizione del sedile [m]');
>> title('salita del passeggero');
```



Il nuovo equilibrio raggiunto ($\bar{x} = -A^{-1}b\bar{u}, \bar{y} = c\bar{x} + d\bar{u}$)[5] è pari a

[2] Posizione e velocità sono positive se dirette verso l'alto.

[3] Per sistemi a tempo discreto il comando è: `ss(A, b, c, d, 1)`.

[4] Per condizione iniziale non nulla il comando è: `lsim(sys, U, T, X0)` con `X0` vettore di stato iniziale.

```
>> Eq_pas=-inv(A)*b*1
>> Y_pas=c*Eq_pas+d*1
```

Per risolvere il secondo punto occorre riscrivere l'equazione del sistema.
Nell'equazione precedente, infatti, la variabile x descriveva la dinamica della massa (sedile + passeggero) quando l'auto era ferma. Per studiare i casi in cui l'auto si muove verticalmente a causa del manto stradale irregolare occorre riscrivere l'equazione tenendo conto che il moto del sedile sia relativo al moto del manto stradale, ovvero scrivendo:

$$M_S\ddot{x}_{sedile} = -k_S x_{rel} - a_S\dot{x}_{rel}$$

dove:

$$x_{rel} = x_{sedile} - x_{strada}$$

Quindi, l'equazione dinamica diventa:

$$M_S\ddot{x}_{sedile} + a_S\dot{x}_{sedile} + k_S x_{sedile} = k_S x_{strada} + a_S\dot{x}_{strada}$$

Considerando come ingresso la forma della strada si ottiene un sistema dinamico della forma

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = \frac{1}{M_S}\left(-a_S x_2(t) - k_S x_1(t)\right) + \frac{a_S\dot{x}_{strada} + k_S x_{strada}}{M_S}$$
$$y(t) = x_1(t)$$

dunque si scriverà l'ingresso come:

$$u = \begin{vmatrix} x_{strada} \\ \dot{x}_{strada} \end{vmatrix}$$

B sarà dunque una matrice della forma:

$$B = \begin{vmatrix} 0 & 0 \\ \frac{k_S}{M_S} & \frac{a_S}{M_S} \end{vmatrix}$$

Si può ora scrivere il nuovo sistema, aggiornando i valori delle matrici *B* e *d*:

```
>> B=[0 0;
k_s/M_s a_s/M_s];
>> c=[1 0];
>> d=[0 0];

>> sys_sedile_2=ss(A,B,c,d);
```

A questo punto, per simulare il sistema, occorre costruire correttamente l'ingresso e la sua derivata. La funzione `genera_rampa(vel)` modellizza una rampa a pendenza costante e di lunghezza (orizzontale) pari a 1 m e altezza 10 cm (*hrampa*). Chiamando *d* la lunghezza della rampa si può modellizzare la rampa (cioè l'altezza del manto stradale) in funzione della velocità di percorrenza del veicolo (*vcar*):

$$x_{strada}(t) = \begin{cases} 0 & t < t_0 \\ \frac{h_{rampa}}{d_{rampa}} v_{car}(t - t_0) & t_0 < t < t_1 \\ h_{rampa} & t > t_1 \end{cases}$$

dove il tempo di salita, dopo aver fissato arbitrariamente t_0, è così calcolato:

$$t_1 = t_0 + \frac{d_{rampa}}{v_{car}}$$

Inoltre

---

[5] A tempo discreto $\bar{x} = (I - A)^{-1}b\bar{u}$, dove *I* è la matrice identità di ordine pari all'ordine della matrice *A*, definita dal comando `eye(size(A))`.

$$\dot{x}_{strada}(t) = \begin{cases} 0 & t < t_0 \\ \dfrac{h_{rampa}}{d_{rampa}} v_{car} & t_0 < t < t_1 \\ 0 & t > t_1 \end{cases}$$

Dunque avendo tutti i dati è possibile generare il vettore di ingresso $u = \begin{vmatrix} x_{strada} \\ \dot{x}_{strada} \end{vmatrix}$.

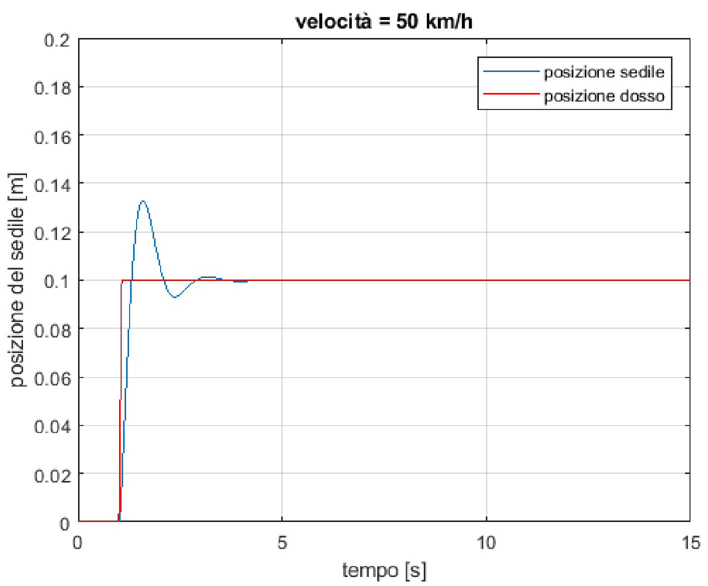
NOTA: le unità di misura da utilizzare affinché i conti siano corretti sono quelle del sistema internazionale (metri, secondi, m/s, etc...).

Si può pertanto risolvere l'ultimo punto nel modo seguente:

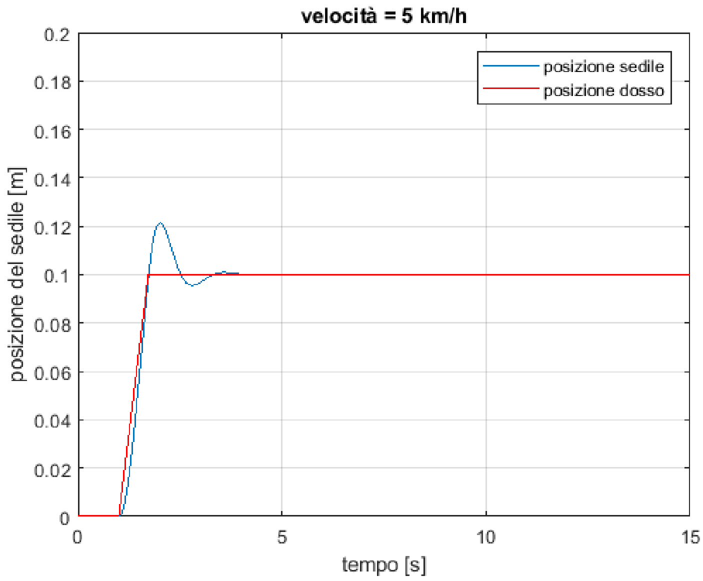
```
>> vel=50/3.6;                                    % m/s
>> [x_strada,x_dot_strada,t_strada]=genera_rampa(vel);
>>[pos_sedile_2,t_sim,st_sim]=lsim(sys_sedile_2,[x_strada;x_dot_strada],t
_strada);
```

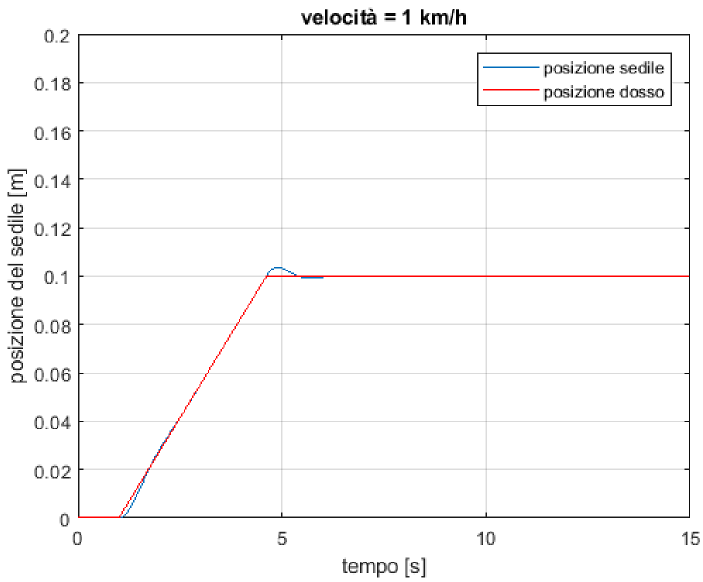
da cui, tracciando solo gli andamenti del profilo stradale, cioè l'altezza del dosso (`x_strada`) e della posizione del sedile (`pos_sedile_2`), si ottiene:

```
>> figure
>> plot( t_sim , pos_sedile_2 )
>> grid on
>> xlabel( 'tempo [s]' )
>> ylabel( 'posizione del sedile [m]' )
>> hold on
>> plot( t_sim,x_strada , 'r' )
>> legend( 'posizione sedile' , 'posizione rampa' );
>> title( sprintf( 'velocità = %i km/h',vel*3.6 ) )
>> axis([0 15 0 0.2])
```



Per simulare cosa succede ad altre velocità basta cambiare il valore di `vel`.





Dai grafici ottenuti si può notare che, diminuendo la velocità, si ha una minore elongazione del sedile rispetto alla posizione di regime.

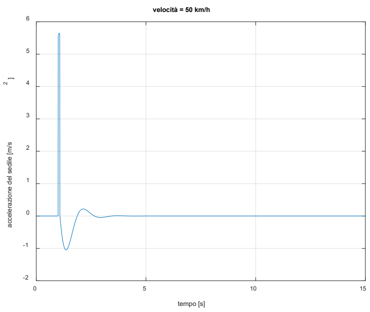
Supponendo infine che il comfort del passeggero dipenda dall'accelerazione verticale del sedile

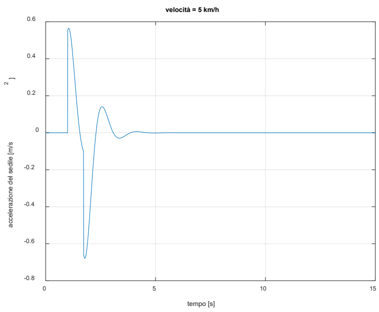
$$\ddot{x}_{sedile} = \frac{1}{M_S}(-a_S(\dot{x}_{sedile} - \dot{x}_{strada}) - k_S(x_{sedile} - x_{strada}))$$

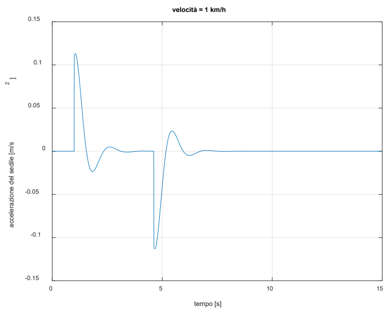
si ottengono i seguenti risultati:

```
>> vel=50/3.6;                              % m/s
>> [x_strada,x_dot_strada,t_strada]=genera_rampa(vel);
>>[pos_sedile_2,t_sim,st_sim]=lsim(sys_sedile_2,[x_strada;x_dot_strada],t
_strada);
>> acc_sedile=(-a_s*(st_sim(:,2)-x_dot_strada)-k_s*(st_sim(:,1)-
x_strada))/M_s;
>> figure; plot(t_sim,acc_sedile);
>> grid on;
>> xlabel('tempo [s]');
```

```
>> ylabel('accelerazione del sedile [m]');
>> title(sprintf( 'velocità = %i km/h',vel*3.6))
```







Il comfort, ovviamente, migliora riducendo la velocità di passaggio sulla rampa (le accelerazioni a cui è soggetto il passeggero diminuiscono).
Infatti, si ha:

velocità [m/s]	acceler. massima [m/s^2]
50	5.6
5	0.6
1	0.11