

Esame Software Engineering (AA 2021/22)

4 Novembre 2022

Enrico Tronci

*Computer Science Department, Sapienza University of Rome
Via Salaria 113 - 00198 Roma - Italy*

`tronci@di.uniroma1.it`

`http://mclab.di.uniroma1.it`

Esercizio 5 (15 punti)

L'unità di tempo per questo esercizio è il secondo.

Indicando come la solito con \dot{x} la derivata rispetto al tempo ($\frac{dx}{dt}$) della variabile x , un modello semplificato della dinamica di un drone è (g è l'accelerazione di gravità):

$$\dot{x}_1 = v_1 \tag{1}$$

$$\dot{x}_2 = v_2 \tag{2}$$

$$\dot{x}_3 = v_3 \tag{3}$$

$$\dot{v}_1 = u_1 + w_1(t) \tag{4}$$

$$\dot{v}_2 = u_2 + w_2(t) \tag{5}$$

$$\dot{v}_3 = u_3 - g \tag{6}$$

dove, per ogni valore di t , $w_1(t)$ e $w_2(t)$ sono variabili aleatorie a valori reali uniformemente distribuite in $[-1, 1]$ e tutte tra loro statisticamente indipendenti. Queste variabili modellano l'effetto del vento.

Inizialmente abbiamo:

$$x_1(0) = 0 \tag{7}$$

$$x_2(0) = 0 \tag{8}$$

$$x_3(0) = 0 \tag{9}$$

$$v_1(0) = 0 \tag{10}$$

$$v_2(0) = 0 \tag{11}$$

$$v_3(0) = 0 \tag{12}$$

Al fine di mantenere la posizione del drone nel setpoint assegnato $r = (r_1, r_2, r_3)$ si usa la seguente strategia di controllo.

Ogni T secondi (*sampling and holding*) il software di controllo calcola il valore di $u = (u_1, u_2, u_3)$ come segue

$$u_1 = k_1(x_1 - r_1) + k_2v_1 \quad (13)$$

$$u_2 = k_1(x_2 - r_2) + k_2v_2 \quad (14)$$

$$u_3 = g + k_1(x_3 - r_3) + k_2v_3 \quad (15)$$

Parametri del Modello

Il modello conterrà i seguenti parametri:

1. **HORIZON**, contenente l'orizzonte di simulazione, cioè il tempo simulato. Potete usare **HORIZON** = 1000.

Per le costanti si usino i seguenti valori:

1. $T = 0.001$
2. $p = -1$
3. $k_1 = -p^2$
4. $k_2 = 2p$

Modello

Se sviluppi un modello Modelica consistente di almeno i seguenti blocchi:

1. Blocco **Plant** nel file **plant.mo** che modella il drone.
2. Blocco **Controller** nel file **ctr.mo** che modella il sistema di controllo come descritto sopra.
3. Blocco **User** nel file **user.mo** che modella il conduttore del drone che sceglie $r = (r_1, r_2, r_3)$. Potete usare il valore costante $r = (10, 20, 30)$.
4. Blocco **Monitor** nel file **monitor.mo** che calcola, per ogni componente, l'errore rispetto alla posizione desiderata.

Nello specifico, ogni T secondi il monitor calcola il l'errore $e_i(t)$ t per la componente i ($i = 1, 2, 3$) come segue:

$$e_i(t) = (x_i(t) - r_i(t)) \quad (16)$$

$$(17)$$

Output della simulazione

Si usi l'istruzione Modelica **terminate** per terminare la simulazione quando la variabile Modelica **time** ha un valore maggiore del parametro **HORIZON**.

Alla terminazione si stampino nel file **outputs.txt** i valori medi e le deviazioni standard per tutti le componenti.

```
ComponenteIndex AvgErr StdDevErr (ID = aaa, MyMagicNumber = bbb,  
HORIZON = ccc, time = ddd)
```

dove:

- **aaa** è il valore del parametro **ID**,
- **bbb** è il valore del parametro **MyMagicNumber**,
- **ccc** è il valore del parametro Modelica **HORIZON**,
- **ddd** è il valore della variabile Modelica **time** quando la simulazione viene terminata dal comando **terminate**.

Le altre righe hanno il seguente formato:

```
<Indice  $i$  della componente> <Valore medio dell'errore  $e_i$  per la componente  
 $i$ > < Deviazione standard dell'errore  $e_i$  per la componente  $i$  >
```

Si avranno quindi, a parte la prima riga di intestazione, 3 righe, una per ogni componente.

Si usi un orizzonte di simulazione molto grande. In particolare si verifichi che l'orizzonte di simulazione sia maggiore del valore del **time** quando la simulazione viene terminata dal comando **terminate**. Se questo non è verificato il modello è sbagliato. Questo valore di **time** è visibile su **stdout**.

NOTA

Si vedano le istruzioni ed in particolare la sezione *NOTA BENE* delle istruzioni.