

# Esame Software Engineering (AA 2022/23)

10 Febbraio 2023

*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 forno per pizza è:

$$\dot{x} = a \min(1, \max(0, u)) - b - d \quad (1)$$

(2)

dove:

1.  $u$  è l'input che rappresenta l'heater del forno;
2.  $a = 10$ : heating rate, cioè l'heater è ON ( $u = 1$ ).
3.  $b = 1$ : cooling rate, cioè l'heater è OFF ( $u = 0$ ).
4.  $d$  (disturbance) modella le perdite di calore per eventi non controllabili (ad esempio, apertura del portello del forno). Nello specifico, ogni  $T_d$  secondi (*sampling and holding*) il valore di  $d$  viene aggiornato scegliendo uniformemente a random un valore in  $[0, 2]$ . Quindi  $d$  è un input esogeni non controllabile per il controllore del forno.

Inizialmente abbiamo:

$$x(0) = 0 \quad (3)$$

(4)

Al fine di mantenere la temperatura del forno nel setpoint assegnato  $r$  si usa la seguente strategia di controllo.

Ogni  $T$  secondi (*sampling and holding*) il software di controllo calcola il valore di  $u$  come segue

$$x_1(t+1) = x(t) \quad (5)$$

$$u(t) = k_1(x(t) - r(t)) + \frac{k_2}{T}(x_1(t) - x(t)) \quad (6)$$

(7)

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

## Modello

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

1. Blocco **Plant** nel file **plant.mo** che modella il forno.
2. Blocco **Disturbance** nel file **dist.mo** che modella la perdita di calore.
3. Blocco **Controller** nel file **ctr.mo** che modella il sistema di controllo come descritto sopra.
4. Blocco **User** nel file **user.mo** che modella il conduttore del forno che sceglie  $r$ . Potete usare il valore costante  $r = 350$  (pizza napoletana, per la pizza romana deve essere  $r = 280$ ).
5. 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  $(t)$   $t$  come segue:

$$e(t) = (x(t) - r(t)) \quad (8)$$

(9)

## 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.

```
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:

< Valore medio dell'errore  $e$  > < Deviazione standard dell'errore  $e$  >

Si avranno quindi, a parte la prima riga di intestazione, 1 riga.

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.