



**UNIVERSITA' POLITECNICA DELLE MARCHE**

---

**FACOLTA' DI INGEGNERIA**

**Corso di Laurea Triennale in Ingegneria Informatica e dell'Automazione**

**Confronto tra tecniche di controllo "Data Driven" e tecniche  
di controllo classiche**

**A Comparison between "Data Driven" and classical control  
techniques**

Candidato:

**Eleonora Brasili**

Matricola:

**1083599**

Relatore:

**Giuseppe Orlando**

Correlatore:

**Gianluca Ippoliti**

**Anno Accademico 2020/2021**



## **Sommario**

Il presente elaborato intende analizzare e mettere a confronto gli aspetti e le proprietà di due tipologie di tecniche di controllo ben distinte seppur complementari: le tecniche di controllo cosiddette “Data-Driven” e le tecniche di controllo “Model-Based” o classiche.

Dopo una panoramica sulle caratteristiche principali di ognuno dei due approcci, sono stati approfonditi e poi messi a confronto in simulazione due metodi di controllo, Robust Adaptive Control (RAC) e Model-Free Adaptive Control (MFAC) appartenenti rispettivamente alle classi Model-Based e Data-Driven.

Lo studio in ambiente SIMULINK (di Matlab) dell’esempio proposto, ha portato a delle considerazioni interessanti sulle prestazioni di tali metodi.



# Indice

## Introduzione

1. Model-based-control (MBC).....	9
1.1. Generalità.....	9
1.2. Model-based adaptive control.....	10
1.2.1 Fully model-based adaptive control.....	10
1.2.2 Learning-based adaptive control.....	11
1.3 Robust Adaptive Control.....	13
1.3.1 Progetto del sistema di controllo.....	15
1.3.2 Analisi della Stabilità.....	17
2. Data-driven-control (DDC).....	19
2.1 Generalità.....	19
2.2 Model-free adaptive control (MFAC).....	20
2.2.1 Full-form dinamic linearization (FFDL).....	20
2.3 Progetto del sistema di controllo FFDL-based MFAC.....	23
2.4 Analisi della Stabilità di FFDL-MFAC.....	26
3. MBC e DDC a confronto.....	27
3.1 Similitudini e differenze.....	27
3.2 Esempio applicativo.....	28
3.3 Simulazione.....	35
3.4 Conclusioni.....	37
4. Appendice.....	38
<b>A-</b> Controllo PID.....	38
<b>B-</b> Sistemi Hammerstein.....	40
<b>C-</b> Tabelle.....	41
C.1-Tabella 1: Simboli MFAC (matlab).....	41
C.2-Tabella 2: Simboli Robust Adaptive Control (matlab).....	42



## Introduzione

La moderna teoria di controllo o Model-based control (MBC), deriva dal modello dello spazio degli stati introdotto da Kalman nel 1960. Negli anni tale teoria di controllo si è evoluta, sviluppandosi in vari ambiti. I suoi rami principali, identificazione del sistema, il controllo adattativo, il controllo robusto, etc. sono stati ampiamente utilizzati nei processi industriali, in campo aerospaziale e in altre applicazioni.

Tuttavia, la teoria MBC contiene ancora molti argomenti ostici sia da una prospettiva teorica che pratica. Per sistemi materialmente complessi, la struttura del modello è spesso difficile da determinare e i parametri sono difficili da identificare, il che rende la progettazione e le applicazioni degli approcci MBC discutibili. Proprio da qui parte lo studio degli approcci data-driven.

Può un controllore essere progettato direttamente dai dati di input-output (I/O), sia online che offline, bypassando lo step della modellazione?

Gli approcci di controllo data-driven si concentrano sulla progettazione del controllore semplicemente utilizzando i dati di misurazione in ingresso e in uscita di un sistema. Poiché questi approcci non richiedono un modello del sistema, nel contesto di progettazione scompaiono le ipotesi teoriche sulla dinamica del sistema, la questione della modellazione e con essa anche il problema delle dinamiche non modellate.

Per la progettazione di un controllore di questo tipo non vengono utilizzati né il modello fisico esplicito né la teoria della stabilità di Lyapunov. È stato dimostrato infatti che la tecnica di controllo secondo Lyapunov, un massiccio strumento matematico nella teoria del controllo, non agisce in maniera efficace con i sistemi non lineari a tempo discreto. Allo stesso tempo, la stabilità, la convergenza e la robustezza sono comunque garantite da una rigorosa analisi teorica basata su delle opportune assunzioni.





# 1. Model-based-control (MBC)

## 1.1 Generalità

La comune procedura per l'applicazione della moderna teoria di controllo si compone di quattro fasi:

- 1- Modellare il sistema;
- 2- Progettare il controllore in base al modello;
- 3- Analizzare le proprietà del sistema di controllo in base a delle assunzioni fatte sul modello;
- 4- Applicare il metodo di controllo progettato.

Dal momento in cui la progettazione del controllore, l'analisi di stabilità, la valutazione delle prestazioni, ecc., sono tutti basati sulla struttura, sulla dinamica e sulle ipotesi imposte al modello del sistema, la moderna teoria di controllo viene anche chiamata teoria basata sul modello (Model Based Control Theory).

Il controllore è quindi progettato sulla base del modello del sistema, presupponendo che il modello rappresenti a tutti gli effetti la dinamica reale del sistema e che i metodi di modellazione siano del tutto affidabili.

Possiamo scomporre le tecniche di controllo Model-Based in due sottoclassi, vale a dire:

1. Controllo *totalmente* basato sul modello o controllo adattativo classico.
2. Controllo adattativo *parzialmente* basato sul modello o basato sull'apprendimento.

Nel primo caso, sia il controllore che i filtri adattativi sono basati sul modello. Al contrario, nella seconda sottoclasse, il controllore si basa sul modello del sistema, ma i filtri adattativi sono data-driven (basati sui dati), ad esempio basati su algoritmi di *machine learning*. Ciascuna di queste classi può essere ulteriormente scomposta in sottoclassi. Infatti, quando si parla di controllo adattativo basato (totalmente/parzialmente) sul modello, possiamo identificare le seguenti sottoclassi: controllo adattativo diretto e adattativo indiretto.

- Il controllo adattativo *diretto* si basa sull'idea di progettare un controllore che si adatti alle incertezze del modello, senza dover identificare esplicitamente i veri valori di queste incertezze.
- Dall'altra parte, il controllo adattativo *indiretto* mira a stimare i valori reali delle incertezze e quindi utilizza le stime di incertezza e il loro modello associato per progettare il controllore.

Ulteriori sottoclassi possono essere identificate in base alla natura matematica del modello. Ad esempio, lineare, non lineare, continuo, discreto etc.

## 1.2 Model-based adaptive control

Per Model-based adaptive controllers, intendiamo controllori progettati a partire dal modello del sistema.

### 1.2.1 Fully model-based adaptive control

In questa sottoclasse includiamo tutti i metodi che sono completamente basati sul modello del sistema, ovvero sia il controllore che i filtri adattativi si basano sul modello del sistema. Questi metodi possono essere ulteriormente classificati in termini di approccio utilizzato per compensare le incertezze del modello (diretto o indiretto), e anche in base alla natura del modello e delle equazioni del controllore.

Potremmo classificare come controllo adattativo lineare diretto il controllo di commutazione adattativo a più modelli, in cui un set di controllori lineari robusti è progettato per gestire una gamma di incertezze del modello e un supervisore viene utilizzato per passare da un controllore ad un altro, mantenendo le prestazioni e la stabilità.

Un problema importante nel controllo adattativo è quello che riguarda la reiezione di un disturbo sconosciuto, spesso chiamata “regolazione adattativa”. Infatti, in questi casi, il modello del sistema è noto o identificato, ma il modello del disturbo è sconosciuto o incerto. Nei metodi adattativi diretti, i filtri adattativi vengono utilizzati per regolare il controllore (per esempio, tramite osservatori) con l’obiettivo di generare una reiezione di un disturbo sconosciuto, senza stimare esplicitamente il modello.

Tra i metodi adattativi indiretti di reiezione di disturbo, possiamo includere numerosi algoritmi che utilizzano il “Principio del modello interno” per modellare disturbi sconosciuti e quindi utilizzare filtri adattativi per stimare i coefficienti del modello interno.

Includiamo anche in questa sezione il cosiddetto ‘Controllo adattativo composito (o combinato)’. In questo approccio, l’idea è di combinare i vantaggi

del controllo adattativo sia diretto che indiretto. È stato dimostrato che fornisce un comportamento transitorio migliore rispetto ai controllori adattativi MRAC diretti o indiretti.

Alcuni dei concetti menzionati precedentemente sono stati estesi anche al caso più impegnativo dei sistemi non lineari. Naturalmente, per poter ottenere un'analisi utile dei controllori non lineari, è stato necessario formulare alcune ipotesi sul tipo di non linearità.

A conclusione di questo paragrafo si forniscono alcuni esempi di strategie di controllo che possono essere classificate sotto il paradigma Model-based: Linear quadratic regulator (LQR)-based adaptive control, Model predictive control-based adaptive control, Nonquadratic Lyapunov function-based MRAC, Sliding mode-based adaptive control, Robust adaptive control.

## 1.2.2 Learning-based adaptive control

Per Learning-based adaptive controllers (controllori basati sull'apprendimento), si intendono controllori che sono in parte basati sul modello del sistema e in parte basati su un algoritmo di apprendimento 'data-driven'. La parte 'data-driven' viene utilizzata per integrare la parte basata sul modello e compensare le informazioni incerte o mancanti nel modello.

Recentemente, diversi studi sono stati dedicati a questo ramo del controllo adattativo. Uno dei motivi alla base di questo crescente interesse è che il campo dell'apprendimento guidato dai dati ha raggiunto una certa maturità, il che ha portato a una buona analisi e comprensione delle principali proprietà degli algoritmi di questa tipologia. Infatti, l'idea su cui si basa la combinazione di un controllore basato su modello con un algoritmo di apprendimento basato sui dati è interessante. Tramite questa combinazione, si potrebbe trarre vantaggio dalla progettazione model-based, con le sue garanzie di stabilità, e aggiungervi i vantaggi dell'apprendimento data-driven, con la sua rapida convergenza e robustezza alle incertezze.

Ad esempio, nel progetto di controllo basato su reti neurali (NNs), l'idea è di scrivere il modello del sistema come combinazione di una parte nota e di una parte sconosciuta (nota anche come parte di disturbo). Il NN viene quindi utilizzato per approssimare la parte sconosciuta del modello. Infine, un

controllore, basato sia sulla parte nota che sulla stima della parte ignota, realizza le regolazioni desiderate.

Il campo di controllo adattativo completamente basato su modelli ha una lunga storia di analisi teoriche e, come tale, è considerato molto maturo in termini di garanzie teoriche. Tuttavia, i risultati ottenuti sono piuttosto limitati ad alcuni tipi di modelli e le restanti estensioni a modelli più generali sono molto impegnative.

Con lo sviluppo della scienza dell'informazione e delle varie tecnologie, attrezzature e processi produttivi stanno diventando sempre più complessi. Perciò, dal momento in cui il modello matematico esatto non è disponibile, il tradizionale approccio MBC diventa impossibile da applicare.

Nella fase di modellazione, quella che si sta attuando non è altro che un'approssimazione del sistema originale. Ciò genererà inevitabilmente delle dinamiche non modellate. Il controllore risultante, quindi, porterebbe a prestazioni insoddisfacenti o ad un comportamento instabile.

Anche se fosse disponibile un modello accurato del sistema, esso risulterebbe molto complesso, con una forte non linearità e con parametri variabili nel tempo. Pertanto, il controllore risultante sarebbe altrettanto complesso e di conseguenza difficile da progettare, utilizzare e mantenere. Costruire un modello accurato di ordine elevato, puntando a buone prestazioni, per poi dover attuare una riduzione dell'ordine o una semplificazione per abbassare la complessità del controllore, sembra paradossale.

Tuttavia, viene generata e memorizzata una quantità enorme di dati di processo contenente preziose informazioni. L'acquisizione, l'archiviazione e l'elaborazione dei dati sono diventate molto più facili e una grande quantità di dati potrebbe essere analizzata ed esplorata grazie alle avanzate tecnologie hardware e software.

Allora la questione principale è la seguente: come usare queste enormi quantità di dati, acquisiti sia on-line che off-line, per progettare direttamente il sistema di controllo?

## 1.3 Robust Adaptive Control

Di seguito si propone l'analisi di un approccio di controllo adattativo robusto per sistemi non lineari Hammerstein SISO a tempo discreto. Si rimanda ad Appendice **B** per un approfondimento sui sistemi di tipo Hammerstein.



Fig.1: Sistema Hammerstein SISO a tempo discreto.

Denotiamo il segnale di ingresso, il segnale intermedio e il segnale di uscita rispettivamente con  $u(k)$ ,  $v(k)$  e  $y(k)$ .

Il sottosistema non lineare è approssimato con una combinazione lineare di un insieme finito di funzioni note insieme a una dinamica non modellata  $\varepsilon(k)$ .

$$v(k) = \sum_{i=1}^p c_i f_i(u(k)) + \varepsilon(k) \quad (1)$$

Dove  $[f_1(u(k)), f_2(u(k)), \dots, f_p(u(k))] = [u(k), u^2(k), \dots, u^p(k)]$

Il modello di equazioni alle differenze del sottosistema lineare è descritto come segue:

$$[A(z^{-1}) + \Delta A(z^{-1})] y(k) = [B(z^{-1}) + \Delta B(z^{-1})] v(k) + \xi(k) \quad (2)$$

Con  $A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}$ ,

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m},$$

$$\Delta A(z^{-1}) = a_{n+1} z^{-(n+1)} + a_{n+2} z^{-(n+2)} + \dots + a_{n+\Delta n} z^{-(n+\Delta n)},$$

$$\Delta B(z^{-1}) = b_{m+1} z^{-(m+1)} + b_{m+2} z^{-(m+2)} + \dots + b_{m+\Delta m} z^{-(m+\Delta m)}.$$

$[A(z^{-1}) + \Delta A(z^{-1})]$  e  $[B(z^{-1}) + \Delta B(z^{-1})]$  sono i polinomi del sistema reale;  $A(z^{-1})$  e  $B(z^{-1})$  sono i polinomi del sistema parametrico;  $\xi(k)$  rappresenta il disturbo.

Da (1) e (2) il sistema può essere riscritto:

$$y(k) = \varphi^T(k)\theta + \zeta(k) \quad (3)$$

dove il vettore dei parametri  $\theta$  ed il vettore regressivo  $\varphi$  sono:

$$\theta = [a_1, a_2, \dots, a_n, b_1c_1, b_1c_2, \dots, b_1c_p, \dots, b_m c_p]^T \quad (4)$$

$$\varphi(k) = [-y(k-1), -y(k-2), \dots, -y(k-n), u(k-1), \\ u^2(k-1), \dots, u^p(k-1), \dots, u^p(k-m)]^T \quad (5)$$

ed il termine non lineare  $\zeta(k)$  è l'errore del modello dovuto ai termini  $\varepsilon(k)$ ,  $\xi(k)$ ,  $\Delta A(z^{-1})$  e  $\Delta B(z^{-1})$ .

$$\zeta(k) = [B(z^{-1}) + \Delta B(z^{-1})] \varepsilon(k) + \Delta B(z^{-1}) \sum_{i=1}^p c_i f_i(u(k)) - \\ \Delta A(z^{-1})y(k) + \xi(k). \quad (6)$$

L'obiettivo è determinare un ingresso di controllo  $u(k)$  tale che l'uscita di sistema  $y(k)$  segua asintoticamente il riferimento  $y^*(k)$ .

**Ipotesi 1:** Gli ordini  $n$ ,  $m$  e  $p$  sono noti a priori.

**Ipotesi 2:**  $[B(z^{-1}) + \Delta B(z^{-1})]$  è a fase minima.

**Ipotesi 3:**  $\zeta(k)$  deve soddisfare la seguente condizione

$$|\zeta(k)| \leq [B(z^{-1}) + \Delta B(z^{-1})] \varepsilon(k) \quad (7a)$$

$$+ \left| \Delta B(z^{-1}) \sum_{i=1}^p c_i f_i(u(k)) \right| + |\Delta A(z^{-1})y(k)| + |\xi(k)| \leq |\bar{\zeta}(k)|$$

$$|\bar{\zeta}(k)| \leq \mu m(k) + M \quad (7b)$$

$$m(k) \leq \sigma m(k-1) + \|\varphi(k)\| + m_0 \quad (7c)$$

dove  $\mu \geq 0$ ,  $\sigma \in [0,1)$ ,  $m_0 \geq 0$ ,  $M \geq 0$  sono delle costanti scelte dall'utente.

### 1.3.1 Progetto del sistema di controllo

Per prima cosa, definiamo

$$\tilde{m}(k) = \sigma \tilde{m}(k-1) + \|\varphi(k)\| + m_0 \quad (8a)$$

$$E(k) = \mu \tilde{m}(k) + M \quad (8b)$$

Il vettore  $\hat{\theta}(k)$  è definito come la stima del vettore dei parametri  $\theta$

$$\hat{\theta}(k) = [\hat{a}_1, \hat{a}_2, \dots, \hat{a}_n, \hat{b}_1 \hat{c}_1, \hat{b}_1 \hat{c}_2, \dots, \hat{b}_1 \hat{c}_p, \dots, \hat{b}_m \hat{c}_p]^T = \\ [\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_n, \hat{\theta}_{n+1}, \hat{\theta}_{n+2}, \dots, \hat{\theta}_{n+2}, \dots, \hat{\theta}_{n+mp}]^T \quad (9)$$

ed è aggiornato tramite l'algoritmo ricorsivo dei 'minimi quadrati':

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{\alpha(k)P(k-1)\varphi(k)e(k)}{1+\varphi^T(k)P(k-1)\varphi(k)} \quad (10)$$

$$P(k) = \frac{P(k-1) - \alpha(k)P(k-1)\varphi(k)\varphi^T(k)P(k-1)}{1+\varphi^T(k)P(k-1)\varphi(k)} \quad (11)$$

$$e(k) = y(k) - \varphi^T(k)\hat{\theta}(k-1) \quad (12)$$

$$\alpha(k) = \begin{cases} 0, & \text{if } |e(k)| \leq 2E(k) \\ \varpi, & \text{altrimenti } (\varpi \in [\chi, 3(1-\chi)/4], 0 \leq \chi \leq 3/7) \end{cases} \quad (13)$$

dove  $e(k)$  è l'errore di stima;  $\alpha(k)$  è un fattore pesato non negativo;

$[\varphi(0), \hat{\theta}(0), P(0)]$  sono le condizioni iniziali dell'algoritmo e  $P(0) = \rho I$  con  $\rho$  numero positivo elevato.

Al fine di ottenere prestazioni migliori, bisogna compensare le dinamiche non modellate  $\zeta(k)$ , ovvero introducendo una stima  $\hat{\zeta}(k)$  di  $\zeta(k)$  come feedback.

$$\hat{\zeta}(k) = y(k) - \varphi^T(k)\hat{\theta}(k) \quad (14)$$

Riassumendo, l'output previsto è definito

$$\hat{y}(k + 1) = \varphi^T(k + 1)\hat{\theta}(k) + \hat{\zeta}(k) \quad (15)$$

La legge di controllo si ottiene come segue:

$$y^*(k + 1) = \hat{y}(k + 1) = \varphi^T(k + 1)\hat{\theta}(k) + \hat{\zeta}(k) \quad (16)$$

L'ingresso di controllo  $u(k)$  è implicito nella (16) e può essere ricavato risolvendo la seguente equazione non lineare

$$\begin{aligned} y^*(k + 1) + \sum_{i=1}^n \hat{\theta}_i y(k + 1 - i) \\ - \sum_{l=2}^m \sum_{r=1}^p \hat{\theta}_{n+p(l-1)+r} u^r(k + 1 - l) - \hat{\zeta}(k) \\ = \hat{\theta}_{n+1} u(k) + \hat{\theta}_{n+2} u^2(k) + \dots + \hat{\theta}_{n+p} u^p(k) \end{aligned} \quad (17)$$

L'ingresso di controllo  $u(k)$  nell'equazione di cui sopra può essere trovato tramite metodi numerici.

Ovviamente può esistere più di un ingresso  $u(k)$  che produca la stessa

$y^*(k + 1)$ . Tuttavia, per la maggior parte dei processi industriali, in normali condizioni di lavoro, le caratteristiche operative sono soggette a funzioni monotone localmente ed esiste un unico input per l'output desiderato.

In sintesi, l'algoritmo ricorsivo si compone dei seguenti passi:

1. Raccogliere i dati disponibili  $\{u(k), y(k)\}$  e costruire il vettore regressivo  $\varphi(k)$ .
2. Calcolare la stima dei parametri  $\hat{\theta}(k)$  e la stima delle dinamiche non modellate  $\hat{\zeta}(k)$ ;
3. Calcolare il segnale di controllo  $u(k)$  risolvendo la funzione non lineare (17) attraverso metodi numerici;
4. Applicare l'ingresso di controllo corrente  $u(k)$  al sistema. Porre  $k = k + 1$  e ritornare al Passo 1.



### 1.3.2 Analisi della Stabilità

In [8] si dimostrano i seguenti risultati.

Innanzitutto, definiamo  $\tilde{\theta}(k) = \hat{\theta}(k) - \theta$ .

**Lemma 1:** Per il sistema (3) con lo stimatore ricorsivo (10)-(13), segue che

$$V(k) - V(k - 1) \leq 0, \text{ dove } V(k) = \tilde{\theta}^T(k) P^{-1}(k) \tilde{\theta}(k), \quad (18)$$

$$\lim_{k \rightarrow \infty} \frac{a(k)e^2(k)}{1 + \varphi^T(k)P(k-1)\varphi(k)} = 0, \quad (19)$$

$$\lim_{k \rightarrow \infty} \|\hat{\theta}(k) - \hat{\theta}(k - 1)\| = 0 \quad (20)$$

**Definizione 1:**  $l_e^\infty$  denota lo spazio delle sequenze che sono limitate in un tempo finito, ossia

$$l_e^\infty = \{X : N \rightarrow R / \sup_{c \geq j} |x(j)| < \infty, \text{ per ogni } c \in N\}.$$

**Lemma 2:** Per il sistema (3), se  $u^P(k)$  e  $v(k)$  appartengono a  $l_e^\infty$ , allora  $u^P(k)$  e  $v(k)$  crescono con la stessa velocità.

**Lemma 3:** Se:

1. Esistono delle costanti positive  $C_1, C_2, C_3$  tali che:

$$\begin{aligned} Z(k) \leq & C_1 + C_2 \max_{0 \leq \tau \leq k} L_1(\tau) \\ & + C_3 \max_{0 \leq \tau \leq k} \{L_2(\tau) + L_3(\tau)Z(\tau - 1)\} \end{aligned} \quad (21)$$

dove  $\{Z(k)\}, \{L_1(\tau)\}, \{L_2(\tau)\}$  e  $\{L_3(\tau)\}$  sono sequenze reali non negative

2.  $\lim_{k \rightarrow \infty} L_3(k) = 0$ ,

allora esistono delle costanti positive  $C_4, C_5, C_6$  tali che

$$Z(k) \leq C_4 + C_5 \max_{0 \leq \tau \leq k} \{L_1(\tau)\} + C_6 \max_{0 \leq \tau \leq k} \{L_2(\tau)\}. \quad (22)$$

**Lemma 4:** Per il sistema (3) con il regolatore adattivo proposto, esistono costanti positive  $C_7, C_8, C_9$  tale che valga la condizione di limitatezza lineare:

$$\|\varphi(k)\| \leq C_7 + C_8 \max_{0 \leq \tau \leq k} |\Delta e(\tau)| + \mu C_9 \max_{0 \leq \tau \leq k} |\tilde{m}(\tau)| \quad (23)$$

Dove  $\Delta = 1 - z^{-1}$ .

**Teorema 1:** Sia il sistema (1)-(2) tale che soddisfi le Ipotesi 1-3. Per il sistema ad anello chiuso (3) con lo stimatore ricorsivo (9)-(14) e la legge di controllo (17), si possono ottenere i seguenti risultati.

1. Tutti i segnali nel sistema ad anello chiuso sono uniformemente limitati, cioè

$$\limsup_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k [u^2(i) + v^2(i) + y^2(i)] < \infty \quad (24)$$

2. L'errore di tracciamento dell'output è tale che

$$\lim_{N \rightarrow \infty} \left| \frac{1}{N} \sum_{k=1}^N (y(k) - y^*(k)) \right| = 0 \quad (25)$$

3. Inoltre, per un disturbo costante  $\xi(k)$  (non necessariamente uguale a zero) e riferimento fisso  $y^*(k)$ , segue che

$$\lim_{k \rightarrow \infty} |y(k) - y^*(k)| = 0. \quad (26)$$

## 2. Data-driven-control (DDC)

### 2.1 Generalità

DDC indica la teoria e il metodo di controllo in cui il controllore è progettato utilizzando direttamente i dati I/O ottenuti monitorando il sistema controllato o dalla conoscenza dei dati di elaborazione. In altre parole, è il metodo utilizzato qualora non si disponga del modello matematico del sistema. Poiché la progettazione del controllore si basa appunto solo sull'utilizzo di dati di I/O misurati ad anello chiuso, i problemi irrisolvibili nei tradizionali sistemi di controllo adattativi basati su modelli (dinamiche non modellate, robustezza, la condizione di eccitazione persistente e il controllo ad anello chiuso, ecc.), possono essere evitati in quanto tutte le informazioni sull'evoluzione dello stato del sistema sono già incluse nei dati misurati.

Ad oggi, esistono diverse tipologie di metodi DDC: il controllo PID, con i suoi numerosi criteri di regolazione dei parametri, può essere considerato il primo dei metodi DDC.

I vari metodi DDC possono essere divisi in due classi:

1. Metodi DDC con struttura di controllo fissa predefinita.
2. Metodi DDC senza una struttura di controllo predefinita.

Qui di seguito si analizzano brevemente le caratteristiche di queste due categorie.

1. Per questa prima classe, il problema di progettazione del sistema di controllo è stato trasformato nell'identificazione dei parametri del controllore poiché si presume che la struttura del controllore sia nota a priori. Tuttavia, per questi metodi, la struttura del controllore non è considerata in fase di progettazione e la stabilità viene solitamente valutata mediante simulazioni o verifiche sperimentali.

Dei metodi tipici appartenenti a questa classe sono il controllo PID, virtual reference feedback tuning (VRFT), iterative feedback tuning (IFT), correlation-based tuning (CbT), unfalsified control (UC) ecc.

La struttura del controllore PID è fissata come semplice combinazione lineare di passato, presente e futuro dell'errore di tracciamento.

VRFT, IFT, UC e CbT di solito presuppongono che il controllore abbia una struttura del tipo PID o simile. Gli approcci per la regolazione dei parametri del controllore sono vari: in 'batch', come PID, VRFT, IFT e UC, o iterativi, come IFT e CbT.

2. Esempi tipici per la seconda categoria sono ‘Simultaneous perturbation stochastic approximation (SPSA) based model-free control’, ‘Nonlinear inversion control (NIC) based DDC’, e ‘Model-free adaptive control’(MFAC). Il vantaggio principale di questo tipo di metodi è il quadro teorico su cui è basata la progettazione del sistema di controllo, compresi il disegno della struttura, l'algoritmo di regolazione dei parametri e l'analisi della stabilità.

Di seguito ci concentreremo sull'analisi del metodo MFAC e delle sue applicazioni.

## 2.2 Model-free-adaptive control (MFAC)

Il metodo ‘Model-free adaptive control’ è stato proposto per la prima volta per una classe di sistemi non lineari a tempo discreto (Hou and Huang 1997). Alla base dell'MFAC c'è il metodo di ‘Linearizzazione dinamica’(DL): invece di identificare un modello non lineare del sistema, viene costruito dinamicamente un modello equivalente lungo i punti operativi del sistema ad anello sfruttando il nuovo concetto di ‘Derivata pseudo-parziale’(PPD). Il controllore ottimale è progettato sulla base di una stima della PPD, la quale viene calcolata esclusivamente mediante i dati di input e di output del sistema controllato.

Esistono tre tipi di modelli per la linearizzazione dinamica:

- 1- Compact form dynamic linearization (CFDL)
- 2- Partial form dynamic linearization (PFDL)
- 3- Full form dynamic linearization (FFDL)

Nel seguente paragrafo si analizzerà la tecnica di linearizzazione del tipo FFDL.

### 2.2.1 Full-form dynamic linearization (FFDL)

Si consideri una classe di sistemi non lineari a tempo discreto non affini sconosciuti (SISO)

$$y(k + 1) = f \left( y(k), y(k - 1), \dots, y(k - n_y), u(k), u(k - 1), \dots, u(k - n_u) \right) \quad (27)$$

Dove  $u(k) \in R$  e  $y(k) \in R$  sono rispettivamente l'input e l'output del sistema all'istante  $k$ ,  $n_y, n_u \in Z_+$  sono gli ordini sconosciuti dell'output e dell'input, e  $f(\cdot): R^{n_u+n_y+2} \rightarrow R$  è una funzione non lineare sconosciuta.

*Ipotesi 1:* Le derivate parziali di  $f(\cdot)$  rispetto a tutte le variabili sono continue.

*Ipotesi 2:* il sistema (27) soddisfa la seguente condizione di Lipschitz generalizzata

$$|y(k_1 + 1) - y(k_2 + 1)| \leq b \|H_{Ly,Lu}(k_1) - H_{Ly,Lu}(k_2)\|$$

dove  $H_{Ly,Lu}(k) = [y(k), \dots, y(k - Ly + 1), u(k), \dots, u(k - Lu + 1)]^T \in R^{Ly+Lu}$  vettore contenente tutti i segnali di input relativi ad una finestra temporale di ampiezza  $[k - Lu + 1, k]$  e tutti i segnali di output relativi ad una finestra temporale di ampiezza  $[k - Ly + 1, k]$ ,  $b$  costante positiva.

Denotando con  $\Delta H_{Ly,Lu}(k) = H_{Ly,Lu}(k) - H_{Ly,Lu}(k - 1)$

**Teorema 1:** Si consideri il sistema non lineare (1) che soddisfi le Assunzioni 1 and 2, per ogni  $1 \leq Ly \leq n_y$  e  $1 \leq Lu \leq n_u$  fissati,  $\|\Delta H_{Ly,Lu}(k)\| \neq 0$ . Allora esisterà un vettore tempo variante  $\varphi_{f,Ly,Lu}(k) \in R^{Ly+Lu}$ , chiamato pseudo-gradiente (PG), tale che il sistema (27) può essere trasformato nel seguente modello FFDL:

$$\Delta y(k + 1) = \varphi_{f,Ly,Lu}^T(k) \Delta H_{Ly,Lu}(k) \quad (28)$$

con  $\|\varphi_{f,Ly,Lu}(k)\| \leq b$  per ogni istante  $k$ , dove

$$\varphi_{f,Ly,Lu}(k) = [\varphi_1(k), \dots, \varphi_{Ly}(k), \varphi_{Ly+1}(k), \dots, \varphi_{Ly+Lu}(k)]^T.$$

Dalla dimostrazione del Teorema 1 possiamo formulare le seguenti osservazioni:

- 1- Il modello dei dati di linearizzazione dinamica non omette, in termini matematici, alcuna informazione del sistema non lineare originale. Quindi esso è esternamente equivalente al sistema non lineare e non è coinvolta alcuna dinamica non modellata. I tradizionali metodi di linearizzazione sono, nella migliore delle ipotesi, un'approssimazione del sistema non lineare. Ad esempio, la linearizzazione di Taylor ignora i termini di ordine elevato e quindi porta a un modello approssimativo. Si può provare, anche in campo pratico e sperimentale, che l'output del modello di dati DL può descrivere quasi accuratamente il sistema originale, tranne che per intervalli di tempo in cui avvengono bruschi cambiamenti del punto di funzionamento.
- 2- Il vettore dello pseudo-gradiente  $\varphi_{f,Ly,Lu}(k)$  (PG) in questo modello di dati non è né esplicito né sensibile alle variazioni nel tempo di parametri, struttura, ritardo e ordini del sistema controllato. Questa caratteristica consentirà di rendere gli approcci di controllo MFAC, progettati utilizzando questo modello di dati di linearizzazione dinamica, capaci di affrontare i problemi di controllo riguardanti sistemi a struttura tempo-variante.
- 3- Se poniamo il valore dello pseudo-ordine dell'uscita  $Ly = 0$  scompariranno le componenti relative alle uscite passate del sistema nel vettore  $H_{Ly,Lu}(k)$ . In questo caso possiamo considerare inclusi nel modello FFDL i seguenti due casi speciali:

- Se  $Ly = 0$  e  $Lu = L$ , il modello FFDL diventa il seguente modello PFDL:

$$y(k + 1) = y(k) + \varphi_{p,L}^T(k) \Delta u(k)$$

Dove  $\varphi_{p,L}(k) = [\varphi_1(k), \dots, \varphi_L(k)]^T \in R^L$  è il PG limitato e la costante  $L$  è chiamata *costante di livello di linearizzazione dell'ingresso di controllo*.  $\Delta u(k) = [\Delta u(k), \dots, \Delta u(k - L + 1)]^T$ .

- Se  $Ly = 0$  e  $Lu = 1$ , il modello FFDL diventa il seguente modello CFDL:

$$y(k + 1) = y(k) + \varphi_c(k) \Delta u(k)$$

Dove  $\Delta u(k) = u(k) - u(k - 1)$  e  $\varphi_c(k) \in R$  è chiamata derivata pseudo-parziale (PPD) del sistema controllato all'istante  $k$ .

- 4- Questi tre modelli di linearizzazione sono modelli tempo-varianti incrementali dinamici e, a differenza del modello nello spazio degli stati nel MBC, non hanno uno specifico significato fisico e descrivono solamente la relazione esistente tra i dati di I/O misurati. Sono quindi dei modelli virtuali atti alla sola progettazione del controllore.

### 2.3 Progetto del sistema di controllo FFDL-based MFAC

Si consideri la seguente funzione di costo dell'input di controllo:

$$J(u(k)) = |y_d(k+1) - y(k+1)|^2 + \lambda |u(k) - u(k-1)|^2 \quad (29)$$

dove  $\lambda > 0$  è il fattore di peso e  $y_d(k+1)$  è l'uscita desiderata.

Considerando l'equazione (28) e (29) e minimizzando la funzione (29) rispetto all'ingresso  $u(k)$  si ottiene il seguente controllore:

$$\Delta u(k) = \begin{cases} \frac{\varphi_{Ly+1}(k) [\rho_{Ly+1}(y_d(k+1) - y(k)) - \sum_{i=1}^{Ly} \rho_i \varphi_i(k) \Delta y(k-i+1)]}{\lambda + |\varphi_{Ly+1}(k)|^2} \\ - \frac{\varphi_{Ly+1}(k) \sum_{i=1}^{Ly+Lu} \rho_i \varphi_i(k) \Delta u(k+Ly-i+1)}{\lambda + |\varphi_{Ly+1}(k)|^2}, Lu \geq 2 \\ \frac{\varphi_{Ly+1}(k) [\rho_{Ly+1}(y_d(k+1) - y(k)) - \sum_{i=1}^{Ly} \rho_i \varphi_i(k) \Delta y(k-i+1)]}{\lambda + |\varphi_{Ly+1}(k)|^2}, Lu = 1 \end{cases} \quad (30)$$

Dove il fattore di crescita  $\rho_i \in (0, 1], i = 1, \dots, Ly + Lu$  è stato aggiunto per rendere l'algoritmo più flessibile.

Per implementare il controllore è necessario implementare un algoritmo di stima dei parametri tempo-varianti del PG.

Si consideri ora la funzione di costo:

$$J(\varphi_{f,Ly,Lu}(k)) = |y(k) - y(k-1) - \varphi_{f,Ly,Lu}(k) \Delta H_{Ly,Lu}(k-1)|^2 + \mu \|\varphi_{f,Ly,Lu}(k) - \hat{\varphi}_{f,Ly,Lu}(k-1)\|^2 \quad (31)$$

Dove  $\mu > 0$  è un fattore di peso.  $\hat{\varphi}_{f,Ly,Lu}(\cdot) \in \mathbb{R}^{Ly+Lu}$  è il valore stimato di  $\varphi_{f,Ly,Lu}(\cdot)$ . Minimizzando la funzione di costo (31) rispetto a  $\varphi_{f,Ly,Lu}(k)$  e utilizzando il lemma sull'inversione delle matrici ottengo:

$$\hat{\varphi}_{f,Ly,Lu}(k) = \hat{\varphi}_{f,Ly,Lu}(k-1) + \frac{\eta \Delta H_{Ly,Lu}(k-1)(y(k) - y(k-1))}{\mu + \|\Delta H_{Ly,Lu}(k-1)\|^2} - \frac{\eta \Delta H_{Ly,Lu}(k-1) \hat{\varphi}_{f,Ly,Lu}^T(k-1) \Delta H_{Ly,Lu}(k-1)}{\mu + \|\Delta H_{Ly,Lu}(k-1)\|^2}$$

$$\hat{\varphi}_{f,Ly,Lu}(k) = \hat{\varphi}_{f,Ly,Lu}(1)$$

$$\text{se } \|\hat{\varphi}_{f,Ly,Lu}(k)\| \leq \varepsilon$$

$$\text{O } \|\Delta H_{Ly,Lu}(k-1)\| \leq \varepsilon$$

$$\text{O } \text{sign}(\hat{\varphi}_{f,Ly,Lu}(k)) \neq \text{sign}(\hat{\varphi}_{f,Ly,Lu}(1))$$

dove  $\eta \in (0, 2]$ ,  $\varepsilon$  è una costante arbitrariamente piccola,

$\hat{\varphi}_{f,Ly,Lu}(k) := [\hat{\varphi}_1(k), \dots, \hat{\varphi}_{Ly}(k), \hat{\varphi}_{Ly+1}(k), \dots, \hat{\varphi}_{Ly+Lu}(k)]^T$ , e  $\hat{\varphi}_{f,Ly,Lu}(1)$  è il valore iniziale della stima di  $\varphi_{f,Ly,Lu}(k)$ .

Una volta ottenuta la stima del vettore PG il controllore può essere riscritto così



$$\Delta u(k) = \begin{cases} - \sum_{i=1}^{Ly+Lu} \rho_i \hat{\varphi}_i(k) \Delta u(k+Ly-i+1), & Lu \geq 2 \\ \hat{\xi}_{Ly+1}(k) \left[ \rho_{Ly+1} (y_d(k+1) - y(k)) - \sum_{i=1}^{Ly} \rho_i \hat{\varphi}_i(k) \Delta y(k-i+1) \right], & Lu = 1 \end{cases}$$

dove  $\hat{\xi}_{Ly+1}(k) = \hat{\varphi}_{Ly+1}(k) / (\lambda + |\hat{\varphi}_{Ly+1}(k)|^2) \in \mathbb{R}$

Di seguito alcune osservazioni sulla scelta dei parametri.

- 1- La scelta di pseudo-ordini  $Lu$  e  $Ly$  piccoli farà diminuire l'onerosità dei calcoli ma farà incrementare notevolmente la complessità dell'andamento delle componenti del PG e di conseguenza sarà più complicato costruire una stima di queste ultime. Al contrario assegnando valori più grandi a  $Lu$  e  $Ly$  il sistema acquisirà la capacità di gestire sempre meglio il processo controllato, avendo a disposizione una lunga memoria dell'andamento dell'uscita controllata. Generalmente all'inizio si scelgono  $Lu$  e  $Ly$  piccoli, se la performance risulta non soddisfacente in termini di convergenza, allora si aumentano  $Lu$  e  $Ly$ . Per quando riguarda gli altri parametri,  $\lambda$  gioca un ruolo importante nell'analisi della stabilità. Infatti, nelle applicazioni pratiche, prima si pone  $\lambda$  in modo da garantire la stabilità dell'uscita e poi si assegnano i valori  $\eta, \mu, \rho_i$  ( $i = 1, \dots, Ly + Lu$ ) e i valori iniziali per migliorare ulteriormente le prestazioni.
- 2- Possiamo notare che ponendo  $Ly = 2$  e  $Lu = 1$  e  $y_d = const$  il controllore diventa un controllore di tipo PID.

## 2.4 Analisi della Stabilità di FFDL-MFAC

*Ipotesi 3:* Assumiamo che il sistema (27) sia controllabile, ovvero, esiste un ingresso di controllo limitato  $u^*(k) \in R$  tale che l'uscita del sistema controllato da  $u^*(k)$  converga all'uscita limitata  $y_d(k)$ .

*Ipotesi 4:* Si assume che sia noto e invariato il segno di  $\varphi_{Ly+1}(k)$ , cioè

$$\varphi_{Ly+1}(k) > \varepsilon > 0 \text{ (oppure } \varphi_{Ly+1}(k) < -\varepsilon < 0)$$

Dove  $\varepsilon$  è una costante piccola positiva.

Il significato di quest'ultima assunzione è che all'aumentare dell'ingresso di controllo  $\Delta u(k)$ , l'uscita del sistema non potrà diminuire, o viceversa.

**Teorema 2:** Se il sistema (27) soddisfa le Ipotesi 1- 4 ed è controllato dallo schema FFDL-MFAC con uscita desiderata  $y_d(k) = y_d = \text{const}$ , allora esiste in  $\lambda_{min}$  tale che, quando  $\lambda > \lambda_{min}$ , il sistema di controllo a ciclo chiuso garantisce:

- 1- L'errore nell'uscita del sistema converge a zero asintoticamente, ossia,

$$\lim_{k \rightarrow \infty} |y_d - y(k+1)| = 0$$

- 2- Il sistema a ciclo chiuso è BIBO stabile, ossia,  $\{y(k)\}$  e  $\{u(k)\}$  sono limitati per ogni  $k$ ;
- 3- Il sistema a ciclo chiuso è stabile internamente.

Dal punto di vista dell'analisi della stabilità, l'analisi del tradizionale controllo adattativo è basata sulla teoria della stabilità di Lyapunov. Il controllo adattativo riesce a raggiungere una convergenza asintotica. Invece, l'analisi della stabilità dello schema di controllo MFAC è basata sul metodo della 'mappatura delle contrazioni' ed MFAC riesce a raggiungere la convergenza monotona.

### 3. MBC e DDC a confronto

#### 3.1 Similitudini e differenze tra gli approcci MBC e DDC

Gli approcci MBC e DDC sono due parti distinte della teoria di controllo. Il loro obiettivo fondamentale è lo stesso: progettare un controllore in grado di guidare il sistema controllato verso un certo comportamento prefissato in modo da soddisfare delle specifiche ben precise.

Ogni metodo di controllo, a prescindere dalla sua natura, ha i propri vantaggi e svantaggi. I metodi di tipo MBC sono i migliori quando abbiamo a disposizione un modello matematico affidabile ed accurato del sistema in oggetto, e con esso un insieme di metodi sistematici di analisi e di progettazione. Ma, in situazioni di incertezza o quando viene meno la possibilità di ottenere un modello affidabile del sistema, i metodi migliori da applicare sono i metodi DDC.

È doveroso evidenziare che la rimozione o la sostituzione completa dei metodi MBC può comportare un enorme spreco di risorse; è molto vantaggioso sviluppare un controller *complementare*, incorporando metodi DDC e metodi MBC. Hou e Jin hanno proposto un sistema di controllo in cui sono combinati il metodo tradizionale MBC (ad esempio, il controllo adattativo) e MFAC. Se il loop di feedback interno esistente funziona bene, il loop esterno (MFAC) è inattivo e non influenza in alcun modo il sistema di controllo originale. Solo quando si rileva un errore di tracciamento, il loop esterno sarà attivo per migliorare le prestazioni del sistema.

Dal momento in cui il controllore è progettato solo a partire dai dati I/O, bisogna riconoscere ai metodi DDC alcuni meriti:

- 1- Grazie al modello di linearizzazione dinamica, in MFAC, la struttura del controllore è determinata solamente mediante i dati I/O on-line.
- 2- Non sono più coinvolte dinamiche non modellate e il concetto di robustezza perde di significato.
- 3- L'analisi della stabilità e della convergenza dell'approccio DDC non dipende dall'accuratezza del modello. A differenza dell'approccio MBC, il quale usa le

informazioni sulla struttura del sistema, MFAC, secondo una serie di risultati teorici, richiede soltanto un limite superiore per la derivata pseudo-parziale.

- 4- Il presupposto principale di MFAC è che il sistema controllato soddisfi la condizione di Lipschitz generalizzata, il cui significato fisico è molto chiaro: a variazioni finite dell'ingresso non possono corrispondere variazioni infinite dell'uscita del sistema.
- 5- La condizione di eccitazione persistente, che è condizione necessaria al tradizionale metodo di controllo adattativo, non è più richiesta per gli schemi MFAC.

Come visto nei paragrafi precedenti, i risultati ottenuti dalla strategia MBC sono piuttosto limitati ad alcuni tipi di modelli. Alcuni 'rilassamenti' di queste restrizioni sono stati ottenuti con il controllo adattativo parzialmente model-based, in cui solo una parte del modello è necessaria per progettare il controllore.

I metodi completamente basati sui dati, invece, apprendono le migliori politiche di controllo attraverso l'interazione diretta con il sistema, senza alcuna conoscenza preliminare del modello del sistema e ciò consente una grande flessibilità. Tuttavia, tale vantaggio si ottiene a partire da numerose misurazioni e una raccolta di dati approfondita. Ne consegue l'esigenza di un'elevata potenza di calcolo. Infatti, tali metodi, non utilizzando nessuna conoscenza sulla fisica del sistema, hanno bisogno di esplorare uno spazio d'azione più ampio per trovare la strategia ottimale.

## 3.2 Esempio applicativo

Per provare l'efficacia pratica dell'approccio analizzato, si propone l'applicazione su di un modello di Hammerstein non lineare del secondo ordine con successiva simulazione comparata con la performance di un controllore PID (Appendice A) e di un controllore adattativo robusto (Robust Adaptive Controller). Il tutto è stato svolto in ambiente Simulink (MATLAB). Si rimanda ad Appendice C per le tabelle dei simboli utilizzati negli algoritmi.

Il modello del sistema è descritto dalla seguente equazione:

$$y(k+1) = 0.6y(k) - 0.1y(k-1) + 1.8u(k) - 1.8u^2(k) + 0.6u^3(k) - 0.15u(k-1) + 0.15u^2(k-1) - 0.05u^3(k-1) \\ := f(y(k), y(k-1), u(k), u(k-1)).$$

È importante sottolineare che tale modello è utilizzato solo per generare i dati di I/O.

Il segnale di riferimento è dato da:

$$yd(k+1) = \begin{cases} 0.5, & 0 \leq k < 100 \\ 1.0, & 100 \leq k < 200 \\ 2.0, & 200 \leq k < 300 \\ 1.5, & 300 \leq k < 400 \end{cases}$$

I condizioni iniziali per MFAC sono

$$y(0) = y(1) = 0$$

$$u(0) = u(1) = 0.3$$

$$\hat{\varphi}_{f, Ly, Lu}(1) = [0.4; 0.4]$$

$$Ly = Lu = 1$$

Le condizioni iniziali per RAC sono:

$$\hat{\theta}(0) = [0.031; 0.031; 0.031; 0.031]$$

$$P(0) = 10^{-5}I$$

La scelta dei parametri (vedi Fig.2, Fig.3 e Fig.4), per tutti gli algoritmi, deriva da una numerosa serie di tentativi operati sul sistema fino al raggiungimento di un risultato abbastanza soddisfacente in termini di stabilità.

In particolare, per i parametri del controllore PID (Fig.4) si è scelto di azzerare l'azione derivativa in quanto, dalle prove eseguite, è risultato un

comportamento fortemente instabile, anche per piccolissimi valori del guadagno derivativo.

Per la scelta dei parametri del RAC (Fig.3), i coefficienti  $\mu$  e  $\sigma$  sono principalmente correlati alle dinamiche non modellate  $\varepsilon(t)$  e alla discrepanza tra la struttura del modello reale e quello stimato  $\Delta A(z^{-1})$  e  $\Delta B(z^{-1})$ . Essi possono svolgere un ruolo importante se il modello parametrico non è sovrapponibile al modello reale; altrimenti possono essere semplicemente scelti uguali a zero. In generale, per garantire che  $E(t)$  cambi in maniera fluida, si suggerisce che siano

$\mu \in [0,0.2]$ ,  $\sigma \in [0,0.2]$ . Il coefficiente  $m0$  è solo un valore iniziale. Per sistemi sconosciuti può essere scelto uguale a zero; altrimenti, potrebbe essere calcolato in base allo stato iniziale.

```
lambda = 1;
mu = 0.4;
etha = 1;
rho = 1;
epsi= 10^-5;
phil=0.4;
phi2=0.4;
```

Fig.2: Scelta parametri di MFAC.

```
sigma = 0;
m0 = 0;
mi = 0;
M = 0.1;
m = 1;
n = 1;
p = 3;
```

Fig.3: Scelta parametri di Robust Adaptive Control.

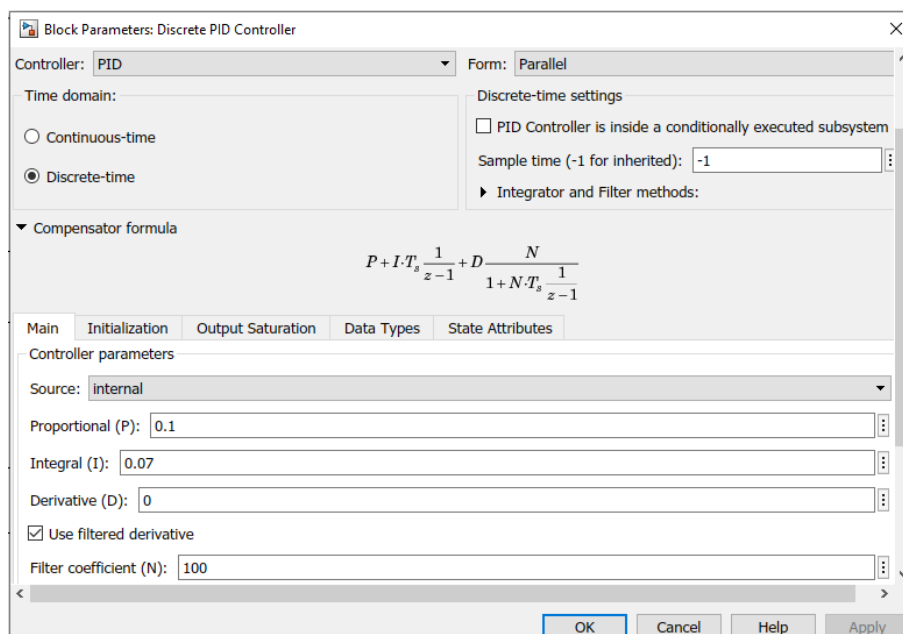


Fig.4: Settaggio parametri PID.

Il diagramma a blocchi in Simulink si presenta come illustrato in Fig.5.

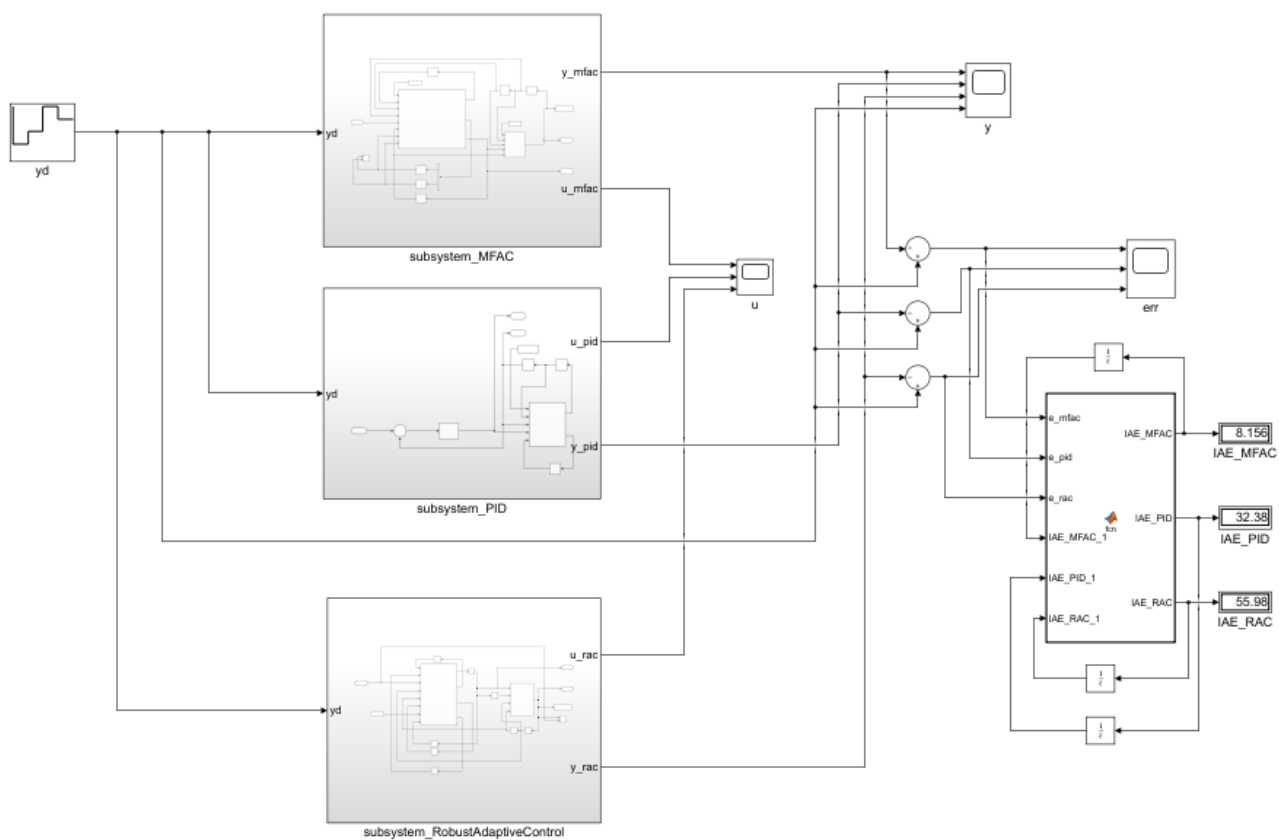


Fig.5: Diagramma a blocchi del sistema.

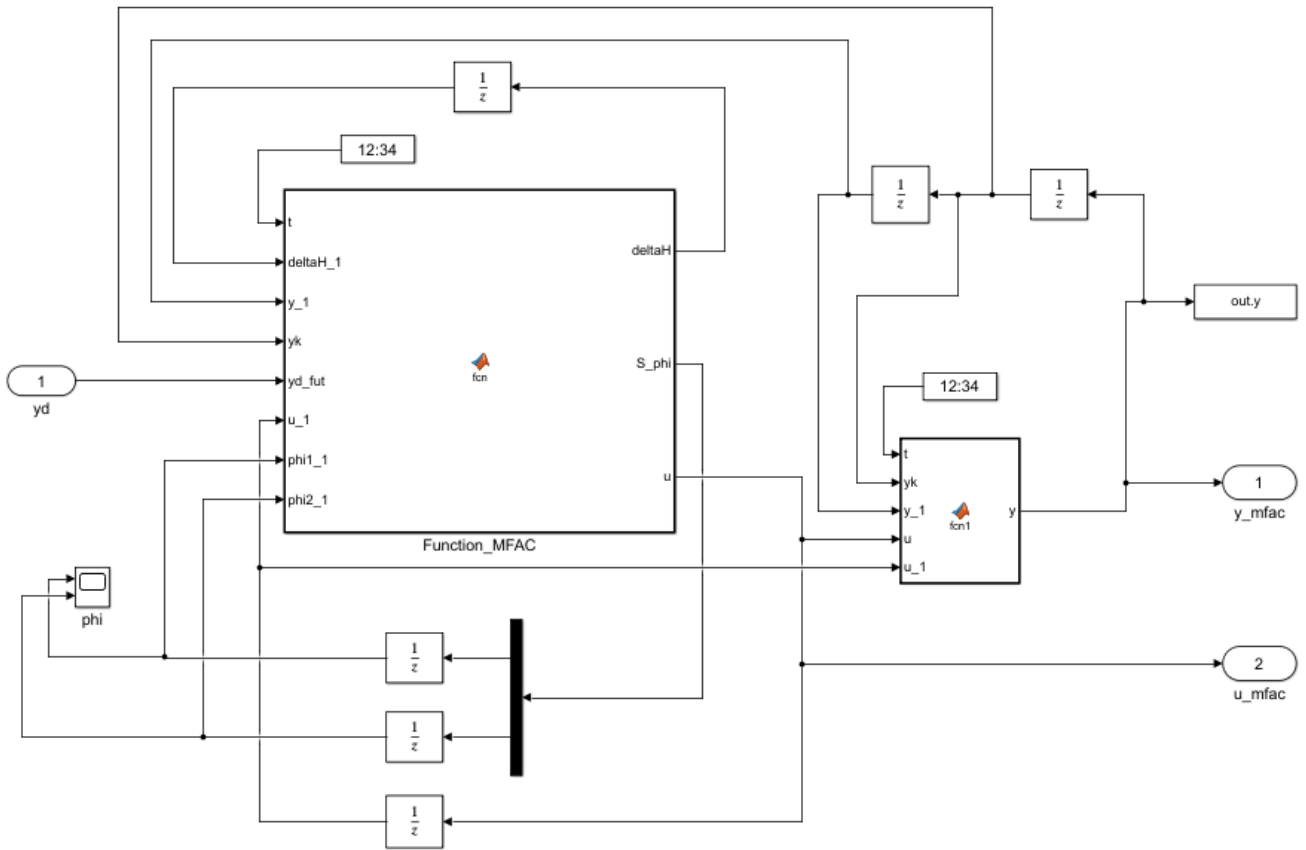


Fig.6: subsystem\_MFAC

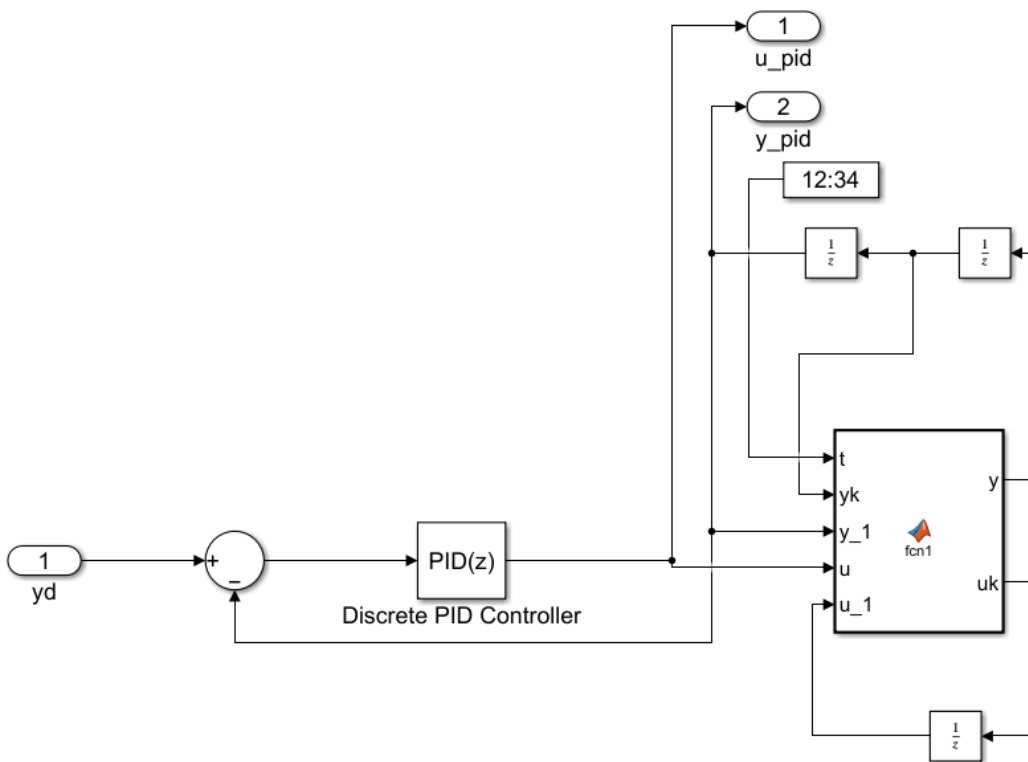


Fig.7: subsystem\_PID



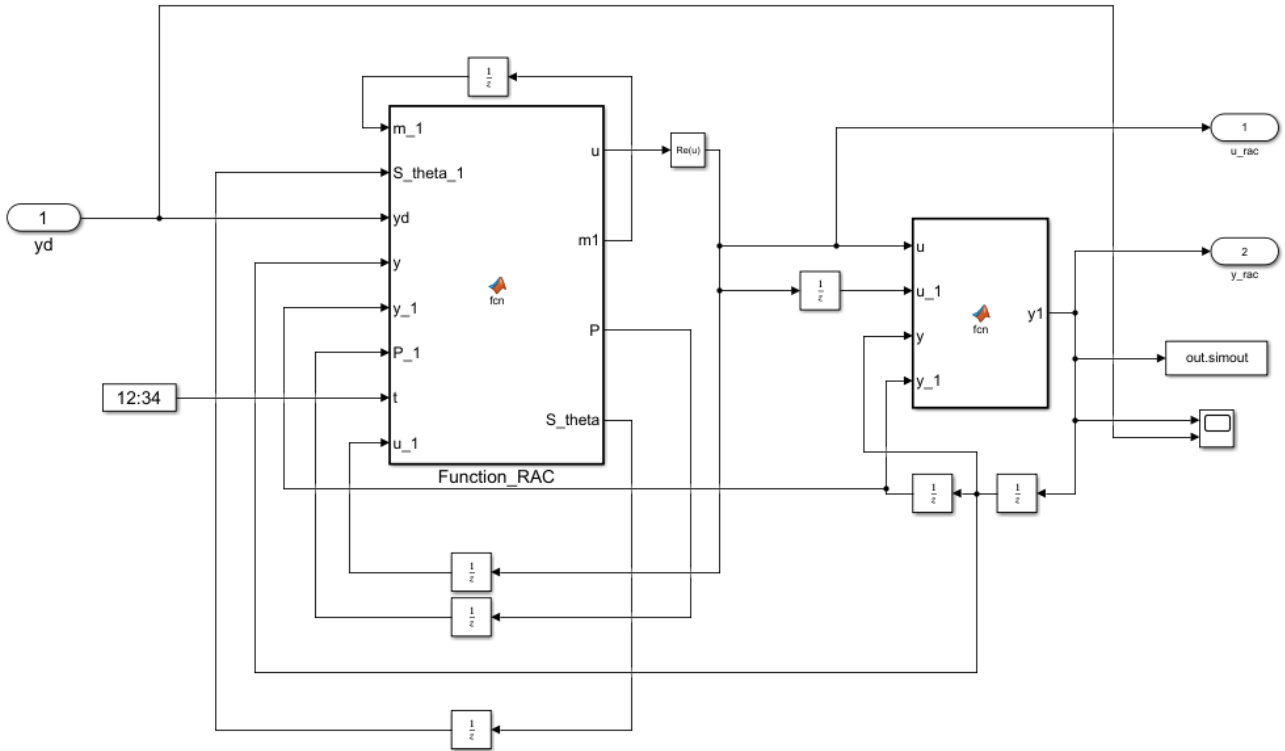


Fig.8: subsystem\_RobustAdaptiveControl

```

1  function [deltaH,S_phi,u] = fcn(t,deltaH_1,y_1,yk,yd_fut,u_1,phil_1,phi2_1,lambda, mu, etha, rho, phil, phi2, epsi)
2
3
4
5  -   deltaY = yk-y_1;
6  -   S_phi_1 = [phil_1;phi2_1];
7  -   S_phi = S_phi_1+(etha*deltaH_1)*(deltaY)/(mu+norm(deltaH_1)^2)-((etha*deltaH_1)*S_phi_1.*deltaH_1)/(mu+norm(deltaH_1)^2);
8  -   if norm(S_phi)<=epsi || norm(deltaH_1)<=epsi || sign(S_phi(1))~= sign(phil)
9  -       S_phi(1) = phil;
10 -      S_phi(2) = phi2;
11 -   end
12
13 -   Xi = (S_phi(2))/(lambda + abs(S_phi(2)^2));
14
15 -   if t<=2
16 -       u=0.3;
17 -   else
18 -       u = u_1+ Xi*(rho*(yd_fut-yk)-(rho*S_phi(1)*deltaY));
19 -   end
20
21
22 -   H = [yk;u];
23 -   H_1= [y_1;u_1];
24 -   deltaH = H - H_1;

```

Fig.9: subsystem\_MFAC > Function\_MFAC (Algoritmo FFDL-MFAC)

```

1  function [u,m1,P,S_theta] = fcn(m_1,S_theta_1,yd,y,y_1,P_1,t,u_1, sigma, m0, mi, M, n, p, m)
2  -
3  -     P_1=10^5*eye(4);
4  -     S_theta_1 = [0.031;0.031;0.031;0.031];
5  - end
6  -
7  -     phi = [-y_1;u_1;u_1^2;u_1^3];
8  -     m1 = sigma*m_1+norm(phi)+m0;
9  -     E = mi*m1+M;
10 -     e = y-phi.'*S_theta_1;
11 -     if abs(e)<= E
12 -         a = 0;
13 -     else
14 -         a = 0.5;
15 -     end
16 -     P = P_1 - (a*P_1*(phi*phi.)*P_1)/(1+phi.'*P_1*phi);
17 -
18 -     S_theta= S_theta_1+ (a*P_1*phi*e)/(1+phi.'*P_1*phi);
19 -     S_zeta= y-phi.'*S_theta;
20 -
21 -     S1 = S_theta(1)*y;
22 -     S2=0;
23 -     S3=0;
24 -     for l = 2:m
25 -         for r = 1:p
26 -             S2 = S2 + S_theta((n+p(l-1))+r)*u_1^(r);
27 -         end
28 -         S3 = S3 + S2;
29 -     end
30 -     poly = [S_theta(n+3) S_theta(n+2) S_theta(n+1) -(yd + S1 - S3 - S_zeta)];
31 -     c = roots(poly);
32 -     u = c(3);

```

Fig.10: subsystem\_RobustAdaptiveControl > Function\_RAC (Algoritmo Robust Adaptive Control).

### 3.3 Simulazione

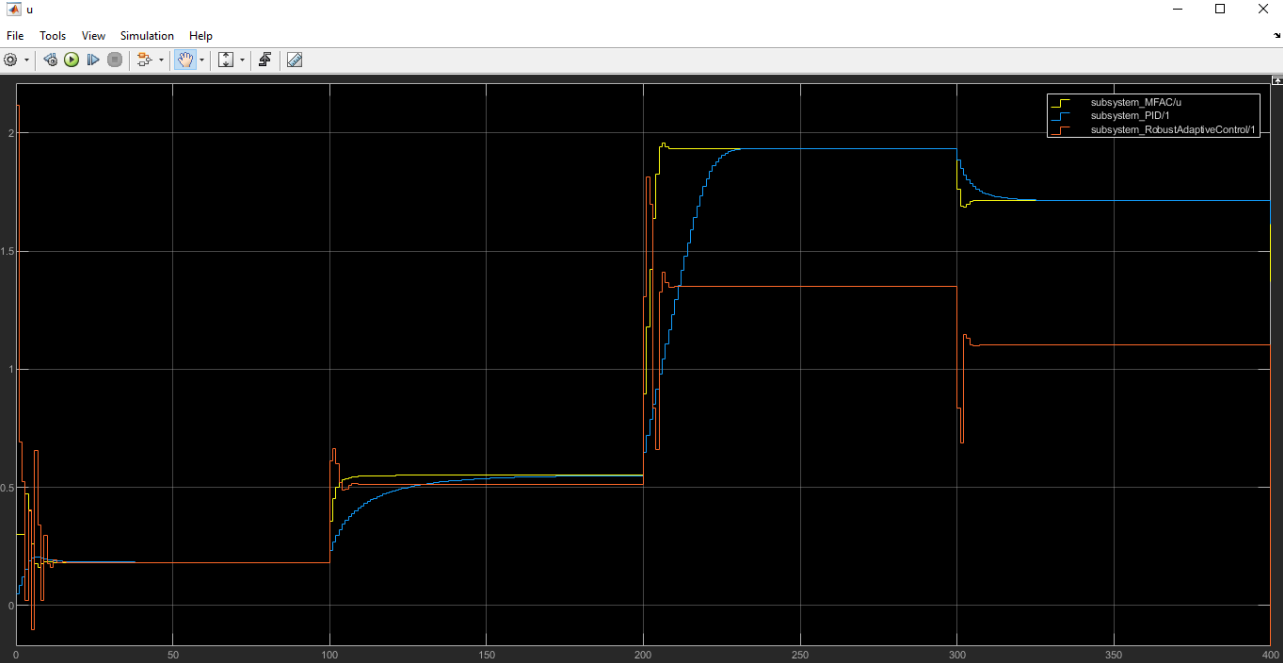


Fig.11: Ingressi di controllo di FFDL-MFAC (giallo), PID (blu), Robust Adaptive Control (rosso).

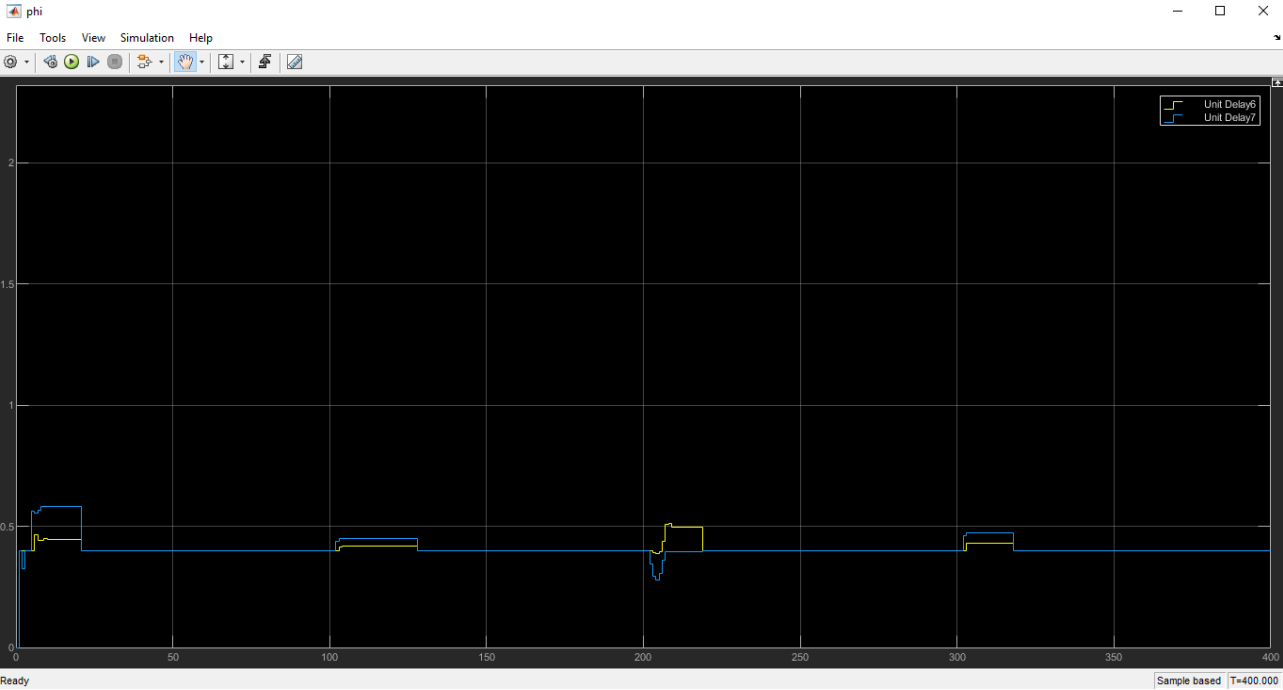


Fig.12: Stime valori delle PPD (MFAC)

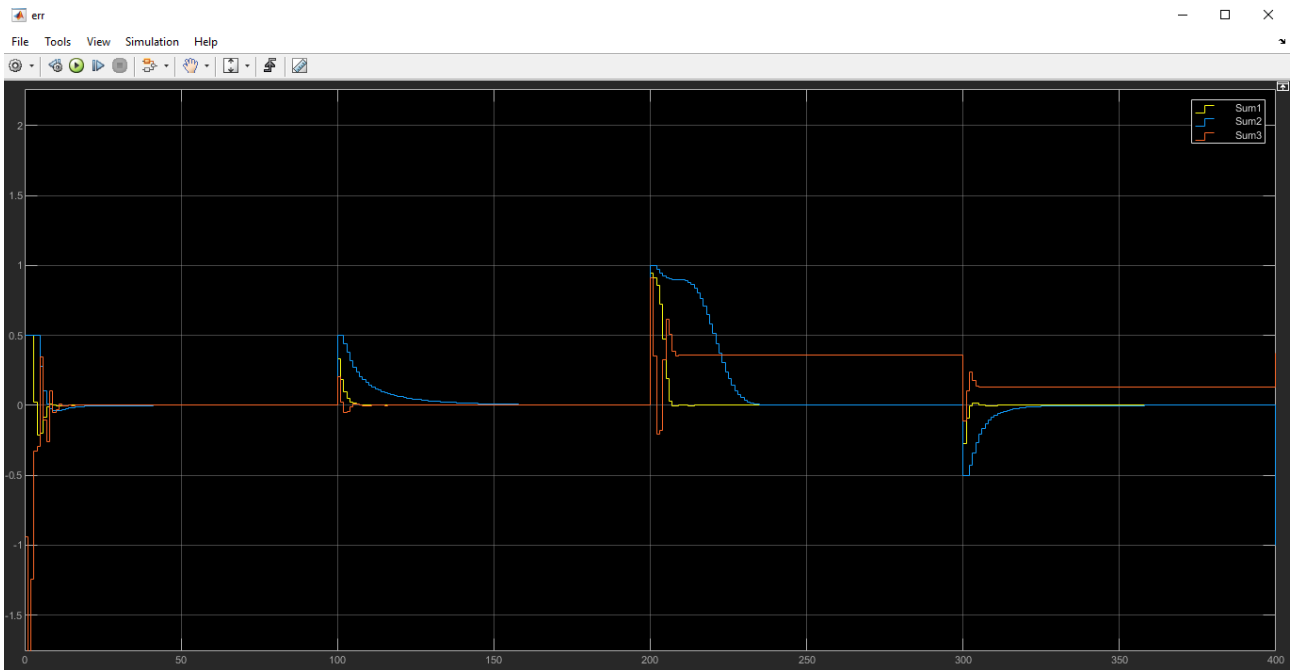


Fig.13: Errore FFDL-MFAC (giallo), PID (blu), Robust Adaptive Control (rosso).

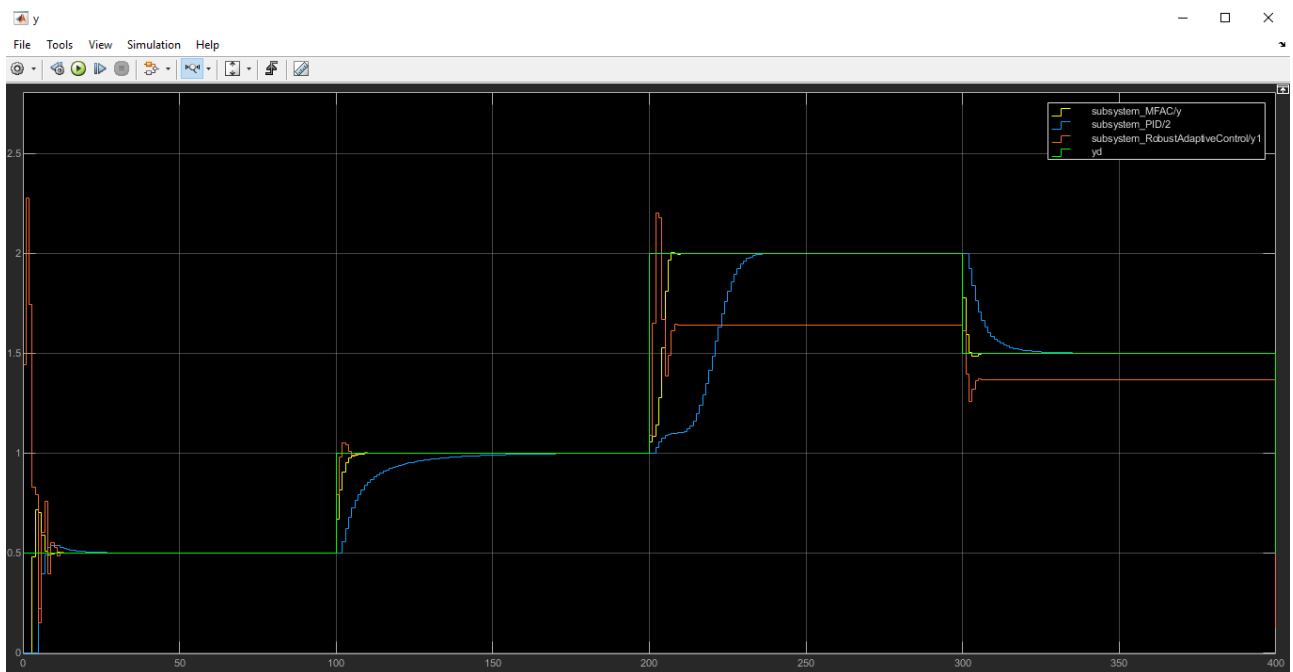


Fig.14: Uscita desiderata  $y_d$  (verde), Uscita FFDL-MFAC (giallo), Uscita PID (blu), Uscita RAC (rosso).

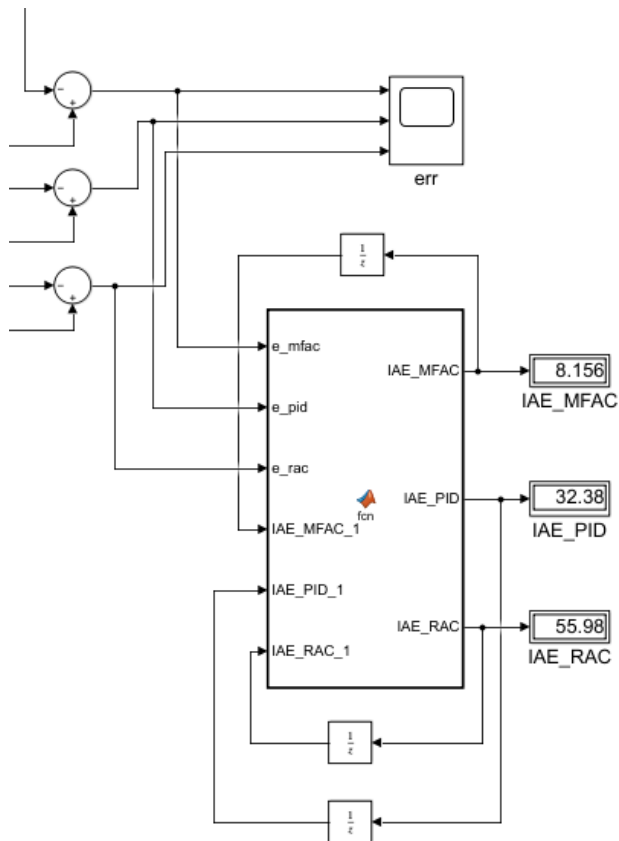


Fig.15: Zoom IN schema di controllo. Valutazione indici IAE (Integral of the Absolute value of the Error).

### 3.4 Conclusioni

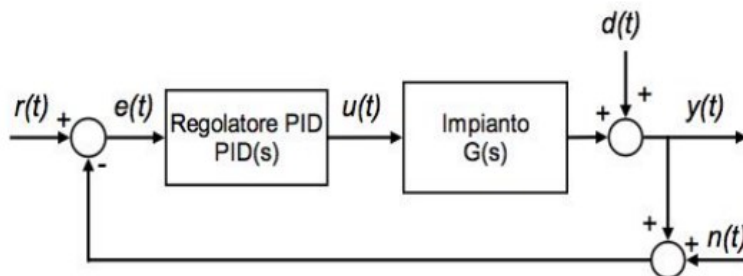
I risultati della simulazione dimostrano in modo evidente la qualità del controllo MFAC. Si può vedere che i metodi DDC (MFAC e PID) sono entrambi efficaci sebbene le uscite (Fig.14) presentino dei picchi in corrispondenza di rapidi transitori. In particolare, l'uscita di MFAC converge al valore desiderato molto più rapidamente delle uscite degli altri due metodi. Al contrario, in questo caso, le prestazioni di controllo di RAC non sono soddisfacenti con un pessimo transitorio ed elevati valori di scostamento dal segnale di riferimento.

Possiamo vedere, inoltre, che gli ingressi di controllo dell'MFAC e PID (Fig.11) hanno un andamento più fluido mentre l'input di controllo di RAC presenta brusche variazioni nel momento in cui viene cambiato il punto di lavoro. Questo perché nella progettazione del controllore di tipo MFAC è aggiunto un fattore di penalità, il quale può limitare la velocità di variazione dell'ingresso; mentre nel controllore adattativo, il cambiamento nell'uscita desiderata si riflette direttamente sull'ingresso di controllo.

Come riportato in Fig.15, è stato calcolato l'indice prestazionale integrale IAE dei tre metodi, grazie al quale si può avere una visione generale sull'andamento della risposta lungo tutto l'arco temporale interessato dal processo. Tale indice, infatti, quantifica lo scostamento dal riferimento nel corso del transitorio.

## 4. Appendice

### A – Controllo PID



- $r(t)$ : riferimento
- $e(t)$ : errore
- $u(t)$ : uscita del controllore
- $y(t)$ : grandezza da controllare
- $d(t)$ : disturbo additivo di uscita
- $n(t)$ : rumore di misura

In un regolatore PID la variabile di uscita  $u(t)$  viene generata in base al contributo di tre termini. Il primo è proporzionale all'errore calcolato tra il riferimento  $r(t)$  e la variabile da controllare  $y(t)$ . Il secondo è proporzionale all'integrale dell'errore  $e(t)$ , quindi dipende dal valor medio dell'errore. Il terzo è proporzionale alla derivata dell'errore, cioè risente della velocità di variazione dell'errore.

La legge di controllo può essere scritta come:

$$u(t) = K_P e(t) + K_I \int_0^t e(t) d\tau + K_D \frac{de(t)}{dt}$$

Dove:

$K_P$  = guadagno proporzionale,  $K_I$  = guadagno integrale,  $K_D$  = guadagno derivativo.

### **Azione Proporzionale**

Con la sola azione proporzionale si produce una differenza tra il valore desiderato e quello effettivamente ottenuto. Tale differenza può essere ridotta aumentando il guadagno proporzionale del controllore. Un controllore esclusivamente proporzionale è utilizzato di solito solo in processi asintoticamente stabili. Infatti, un aumento del guadagno proporzionale potrebbe portare ad un aumento delle oscillazioni a seguito di rapidi transitori.

### **Azione Integrale**

L'azione integrale può essere vista come un dispositivo per l'azzeramento dell'errore a regime introdotto dall'azione proporzionale. I controllori PI permettono di ottenere una maggiore precisione senza peggiorare la stabilità del sistema.

### **Azione Derivativa**

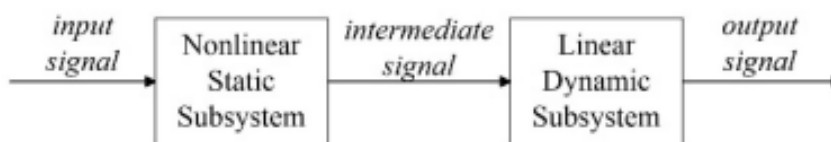
Tale azione è utile al miglioramento della stabilità a ciclo chiuso. Spesso però i controllori non hanno bisogno di tale azione in quanto presenta l'inconveniente di amplificare i segnali con contenuto armonico a frequenze elevate. Ciò potrebbe portare ad un'instabilità nel sistema rendendo pessima la qualità del controllo.

Grazie a procedure di taratura basate sull'esperienza è possibile calcolare i guadagni solo mediante semplici prove sperimentali da effettuare sul sistema stesso.

## B- Sistemi Hammerstein

Per associare un modello matematico ad un sistema fisico, bisogna prima scegliere un modello (modellazione) che sia conforme al sistema. Quando non ci sono informazioni sulla struttura del sistema, è necessario un modello generale che si ritiene si adatti bene al sistema sulla base dell'esperienza precedente. In generale, quando possibile, si cerca di utilizzare un modello lineare, perché di più semplice gestione dal punto di vista del controllo. Sfortunatamente, la maggior parte dei sistemi fisici ha un comportamento non lineare. Gestire sistemi dinamici non lineari generici è un compito arduo, pertanto, di solito ci si limita a considerare solo una certa classe di sistemi non lineari. Una categoria è quella dei sistemi a blocchi, i quali consistono in una sequenza in cascata di blocchi statici non lineari e blocchi dinamici lineari. È stato dimostrato che modelli di questo tipo forniscono una buona approssimazione dei sistemi fisici non lineari incontrati nella pratica ingegneristica.

Un sistema dinamico non lineare a blocchi è detto di tipo Hammerstein quando è costituito da un blocco non lineare statico seguito da un blocco lineare dinamico.



La difficoltà che si incontra lavorando con questo tipo di sistemi è che solo i segnali di ingresso e di uscita sono misurabili, mentre il segnale intermedio non è direttamente disponibile.

Non è quindi possibile stimare i parametri dei due blocchi direttamente a partire dai loro ingressi e uscite. Le strutture non lineari sono generalmente considerate note per semplificare i problemi di identificazione; una struttura ampiamente utilizzata è la struttura polinomiale, che ha il vantaggio di avere i coefficienti lineari e di poter approssimare bene un'ampia classe di funzioni (purché venga utilizzato un polinomio di grado sufficientemente alto).



## C – Tabelle

### C.1-Tabella 1

Tabella 1	
$\lambda$	lambda
$\mu$	mu
$\eta$	etha
$\rho$	rho
$\varepsilon$	epsi
$\hat{\varphi}_{f,1,1}(\mathbf{1})$	[phi1;phi2]
$\Delta H_{Ly,Lu}(\mathbf{k})$	deltaH
$\Delta H_{Ly,Lu}(\mathbf{k} - \mathbf{1})$	deltaH_1
$H_{Ly,Lu}(\mathbf{k})$	H
$H_{Ly,Lu}(\mathbf{k} - \mathbf{1})$	H_1
$\Delta y$	deltaY
$\hat{\varphi}_{f,Ly,Lu}(\mathbf{k})$	S_phi
$\hat{\varphi}_{f,Ly,Lu}(\mathbf{k} - \mathbf{1})$	S_phi_1
$\hat{\xi}_{Ly+1}(\mathbf{k})$	xi
$u(\mathbf{k})$	u
$u(\mathbf{k} - \mathbf{1})$	u_1
$y(\mathbf{k})$	yk
$y(\mathbf{k} - \mathbf{1})$	y_1
$yd(\mathbf{k} + \mathbf{1})$	yd_fut
$\hat{\varphi}_{f,1,1}(\mathbf{k} - \mathbf{1})$	[phi1_1; phi2_1]
$k$	t

Tabella 1: Simboli MFAC (matlab)

## C.2-Tabella 2

<b>Tabella 2</b>	
$\tilde{m}(k)$	m1
$\tilde{m}(k - 1)$	m_1
$\hat{\theta}(k)$	s_theta
$\hat{\theta}_i(k)$	s_theta(i)
$\hat{\theta}(k - 1)$	s_theta_1
$\hat{\zeta}(k)$	S_zeta
$\varphi(k)$	phi
$y^*(k + 1)$	yd
$y(k)$	y
$E(k)$	E
$e(k)$	e
$a(k)$	a
$y(k - 1)$	y_1
$P(t - 1)$	P_1
$u(k - 1)$	u_1
$\sigma$	sigma
$m0$	m0
$\mu$	mi
$M$	M
$n$	n
$m$	m
$p$	p

Tabella 2: Simboli Robust Adaptive Control (matlab)

## Bibliografia e Linkografia

- [1] Hou, Zhongsheng, Ronghu Chi, and Huijun Gao. "An Overview of Dynamic-Linearization-Based Data-Driven Control and Applications." *IEEE Transactions on Industrial Electronics* 5.64 (2017): 4076-4090.
- [2] Z. Hou and S. Xiong, "On Model-Free Adaptive Control and Its Stability Analysis," in *IEEE Transactions on Automatic Control*, vol. 64, no. 11, pp. 4555-4569, Nov. 2019.
- [3] Hou, Zhong-Sheng, and Zhuo Wang. "From model-based control to data-driven control: Survey, classification and perspective." *Information Sciences* 235 (2013): 3-35.
- [4] Hou, Zhongsheng, and Yuanming Zhu. "Model based control and MFAC, which is better in simulation?" *IFAC Proceedings Volumes* 46.13 (2013): 82-87.
- [5] Hou, Zhongsheng, and Shangtai Jin. "A novel data-driven control approach for a class of discrete-time nonlinear systems." *IEEE Transactions on Control Systems Technology* 19.6 (2010): 1549-1558.
- [6] S. Xiong and Z. Hou, "Model-Free Adaptive Control for Unknown MIMO Nonaffine Nonlinear Discrete-Time Systems With Experimental Validation," in *IEEE Transactions on Neural Networks and Learning Systems*.
- [7] Benosman M. "Model-based vs data-driven adaptive control: An overview." *Int J Adapt Control Signal Process.* 2018;32:753–776.
- [8] Yuan, Ping, Bi Zhang, and Zhi-Zhong Mao. "A robust adaptive controller for Hammerstein nonlinear systems." *International Journal of Control, Automation and Systems* 15.2 (2017): 656-667.
- [9] [http://www.unife.it/ing/lm.meccanica/insegnamenti/dinamica-controllo-diagnosi-di-sistemi-b/materiale-didattico/Controllori\\_PID.pdf](http://www.unife.it/ing/lm.meccanica/insegnamenti/dinamica-controllo-diagnosi-di-sistemi-b/materiale-didattico/Controllori_PID.pdf)
- [10] [https://etd.adm.unipi.it/theses/available/etd-04052005-114205/unrestricted/003\\_CAP1.PDF](https://etd.adm.unipi.it/theses/available/etd-04052005-114205/unrestricted/003_CAP1.PDF)