



Università Politecnica delle Marche

FACOLTÀ DI INGEGNERIA

Corso di Laurea Triennale in Ingegneria Informatica e dell'Automazione

TESI DI LAUREA TRIENNALE

**Studio e analisi di metodi di controllo
Data Driven per
sistemi non lineari a tempo discreto**

Study and analysis of Data Driven control methods
for
nonlinear discrete time systems

Candidato:

Giorgio Antonini

Matricola 1083747

Relatore:

Gianluca Ippoliti

Correlatore:

Giuseppe Orlando

Ringraziamenti

Un immenso grazie va alla mia famiglia che mi ha dato questa grande opportunità. Ringrazio il Professor Ippoliti per l'aiuto e la disponibilità in questi mesi. Un ringraziamento particolare va ai miei coinquilini, amici da sempre Marco e Simone, e insieme a loro mi piacerebbe ringraziare anche i tanti amici conosciuti a lezione. Vorrei cogliere l'occasione per ringraziare la mia fidanzata Laura, che mi ha aiutato e sostenuto fin dall'inizio in questo percorso di studi formativo non solo a livello accademico ma anche umano.

Indice

1	Introduzione	8
2	Teoria MBC	9
3	Teoria DDC.....	13
3.1	Necessarietà della teoria e dei metodi DDC	14
3.2	Model Free Adaptive Control	17
4	Tecniche di linearizzazione per sistemi non lineari a tempo discreto	18
4.1	Compact Form Dynamic Linearization.....	19
4.2	Partial Form Dynamic Linearization.....	24
5	Model Free Adaptive Control di sistemi SISO non lineari.....	27
5.1	Modello di dati CFDL basato su MFAC	29
5.1.1	Algoritmo di controllo	30
5.1.2	Schema di controllo	32
5.2	PFDL modello di dati basato su MFAC.....	35
5.2.1	Algoritmo di controllo	35
5.2.2	Schema di controllo	37
6	Applicazione e analisi	39
6.1	Simulazioni metodo CFDL-MFAC.....	39
6.1.1	Simulazione 1	40
6.1.2	Simulazione 2	43
6.2	Simulazioni metodo PFDL-MFAC	45
6.2.1	Simulazione 3	47
6.2.2	Simulazione 4	49
6.2.3	Simulazione 5	52
7	Conclusioni	56
	Lista delle variabili Simulink.....	57

Elenco delle figure

Figura 1: Architettura MBC.....	11
Figura 2: Architettura DDC	14
Figura 3: Oggetti controllati da DDC.....	15
Figura 4: Interpretazione geometrica di PPD	22
Figura 5: Algoritmo di controllo CFDL-MFAC.....	39
Figura 6: Schema Simulink CFDL-MFAC.....	40
Figura 7: Schema scambiatore di calore	40
Figura 8: Processo simulazione 1	41
Figura 9: Parametri simulazione 1	41
Figura 10: Uscita sistema simulazione 1	41
Figura 11: Stima di PPD simulazione 1	42
Figura 12: Errore tra uscita e riferimento simulazione 1	42
Figura 13: Uscita controllore simulazione 1	42
Figura 14: Processo 2 simulazione 2	43
Figura 16: Parametri simulazione 2	43
Figura 15: Matlab Function	43
Figura 17: Ingresso di riferimento del sistema simulazione 2	43
Figura 18: Stima di PPD simulazione 2.....	44
Figura 19: Errore tra uscita e riferimento simulazione 2	44
Figura 20: Uscita sistema simulazione 2	44
Figura 21: Algoritmo di controllo PFDL-MFAC	46
Figura 22: Schema PFDL-MFAC.....	46
Figura 23: Processo simulazione 3	47
Figura 24: Ingresso di riferimento simulazione 3	47
Figura 25: Parametri simulazione 3	47
Figura 26: Uscita sistema simulazione 3	48
Figura 27: Errore tra uscita e riferimento simulazione 3	48
Figura 28: Stima di PG simulazione 3	48
Figura 29: Ingresso di riferimento simulazione 4.....	49
Figura 30: Processo simulazione 4	49
Figura 31: Schema Simulink simulazione 4	50
Figura 32: Confronto uscite sistema MFAC (giallo)- PID (blu) simulazione 4	50
Figura 33: Confronto uscite controllore MFAC (giallo) - PID (blu) simulazione 4..	51
Figura 34: Stima di PG simulazione 4	51
Figura 35: Schema Simulink simulazione 5	52

Figura 36: Parametri simulazione 5	53
Figura 37: Ingresso di riferimento simulazione 5	53
Figura 38: Processo simulazione 5	53
Figura 39: Uscita CFDL-MFAC simulazione 5	54
Figura 40: Uscita PFDL-MFAC simulazione 5	54
Figura 41: Confronto uscite controllori CFDL (blu) - PFDL (giallo) simulazione 5	54
Figura 42: Confronto errore tra CFDL (blu) - PFDL (giallo) simulazione 5	55

1 Introduzione

La prima parte dell'elaborato si è basata su una analisi della teoria del controllo caratterizzata dallo studio del modello, la teoria MBC, la quale sebbene molto utilizzata, presenta dei limiti. Proprio per questo motivo viene quindi proposta come alternativa, la teoria DDC, affiancata da un nuovo efficace metodo di linearizzazione per sistemi non lineari. Nella parte centrale vengono quindi descritti due metodi di controllo adattativi per sistemi non lineari a tempo discreto completamente *data driven*.

Nella parte conclusiva vengono quindi proposti delle simulazioni, ricreate sul *Simulink* del *Matlab*, nelle quali i metodi verranno utilizzati per controllare sistemi di complessità via via crescente, messi a confronto tra loro e paragonati con altri metodi di controllo *data driven*.

2 Teoria MBC

L'introduzione del modello spazio di stato da parte di *Kalman* nel 1960 ha dato il via ai metodi e alla teoria del controllo moderno. La teoria del controllo moderno è stata fondata e sviluppata sulla base di un modello nominale o matematico del sistema da controllare noto con esattezza, da ciò chiamata teoria MBC, *Model Based Control*. Lo sviluppo della teoria MBC, includendo la teoria dei sistemi lineari, del controllo robusto, del controllo ottimo e via dicendo, ha trovato molti ambiti pratici in cui essere applicata con successo. D'altra parte, le dimensioni degli impianti industriali e aziendali sono aumentate, la produzione tecnologica e i processi sempre più complessi e le richieste sulla qualità dei prodotti sono sempre più alte. Tutto ciò ha portato grandi sfide agli studi teorici e alle applicazioni della MBC. In questo momento quasi la totalità dei metodi di controllo utilizzati per sistemi lineari e non lineari è comunque *model based*. Per i metodi MBC, il primo passo è costruire il modello matematico del processo, in seguito definire il controllore basato sul modello del processo ottenuto con la speranza che rappresenti il vero sistema e infine analizzare il sistema a ciclo chiuso in virtù del modello matematico. Ci sono due tipologie di metodi per modellare un processo: il metodo dei primi principi e il metodo di identificazione del sistema. Modellare un processo con il metodo dei primi principi significa stabilire le equazioni della dinamica del processo controllato secondo qualche legge fisica o chimica e determinare i parametri del modello attraverso una serie di esperimenti. Modellare un processo attraverso un sistema di identificazione significa sviluppare un modello del processo descritto da input e output, i quali seguono il vero sistema e possono

approssimare il vero sistema in termini di incertezza o errore, usando le misurazioni prese online o offline. È riconosciuto che utilizzando questi metodi si può ottenere solo una approssimazione del vero sistema, con degli errori dovuti alla complessità della struttura del sistema e dell'ambiente. In altre parole, le dinamiche non modellate e le altre incertezze esisteranno sempre nei processi modellati. La conseguenza è la possibilità di avere dei problemi proprio dovuti alla mancanza di accuratezza del modello matematico. Il corrispondente sistema a ciclo chiuso infatti potrebbe avere delle debolezze di robustezza e anche di sicurezza dovute anche a possibili disturbi esterni. Ciò significa che è difficile applicare metodi MBC per sintetizzare il controller per sistemi pratici e le performance e la sicurezza non sono sempre garantite quando è connesso al processo.

Per il design del sistema di controllo è naturale quindi mettere uno sforzo significativo fin da subito per ottenere un modello matematico più accurato possibile, includendo le incertezze, per poi definire il controller sul modello trovato per una qualche applicazione. Esistono comunque diversi problemi a livello teorico e pratico per delineare un modello perfetto del processo e del controller; primo tra tutti il problema delle parti non modellate e la robustezza che ne deriva; problemi che non possono essere risolti simultaneamente con la teoria MBC. Fino a oggi non esistono efficienti sistemi o metodi per identificare e produrre un modello accurato del processo per sistemi non lineari complessi che nel mondo reale sono molto diffusi. Spesso il modellare è più difficile dello stesso design del controllore. Se il processo ha dei parametri dipendenti dal tempo complica notevolmente il tentativo di modellare il processo. L'accuratezza del modello spesso comporta uno sforzo e un costo molto alto

e questo non rende di certo il controllore meno complesso. Un controllore di grado elevato comporta una scarsa robustezza e affidabilità oltre che ad enormi problemi di mantenimento e monitoraggio. In pratica spesso bisogna quindi affiancare una operazione di riduzione per ottenere sistemi più semplici. L'ultimo, ma non meno importante, è la necessità di un ingresso persistentemente eccitante per modellare, non solo non è semplice realizzare ma anche non è possibile produrre un modello accurato senza. La chiave del design e analisi del controllore MBC è nell'utilizzo dell'errore dinamico del sistema a ciclo chiuso, e il processo è incluso in ogni fase dell'applicazioni e design del sistema di controllo. L'architettura della teoria MBC è mostrata in *figura 1*.

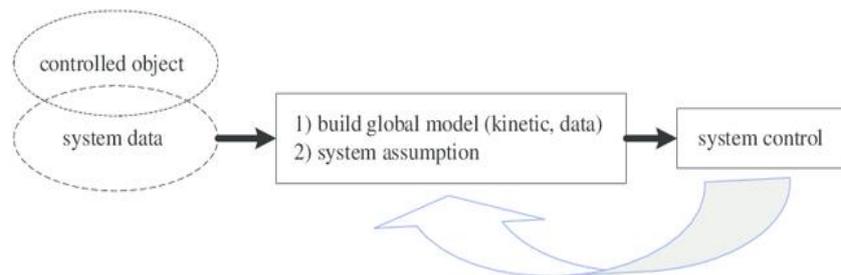


Figura 1: Architettura MBC

Dal momento che ci saranno sempre parti non modellate oltre a incertezze varie, il controllore basato sul modello non potrebbe funzionare nella pratica. Alle volte, nonostante l'accuratezza del modello, i risultati dell'analisi teorica come stabilità, convergenza, robustezza a ciclo chiuso non sono sempre valide se le assunzioni e approssimazioni non sono ragionevoli. Nel controllo adattativo si ipotizza che tutti i segnali rimarranno limitati nel tempo; ciononostante nella pratica con il tempo che diverge ad infinito il variare dei parametri e dinamiche non ben modellate potrebbero portare a un sistema a ciclo chiuso instabile. Per il design del controllore MBC,

l'accuratezza del modello, la correttezza delle assunzioni imposte sul modello del processo matematico determinano insieme le performance, affidabilità e sicurezza del sistema di controllo. Se il modello non è disponibile le assunzioni non sono soddisfatte e nessuna conclusione è ottenibile sul controllore. La MBC, come metodo di controllo, di conseguenza nasce e muore con il modello del sistema.

3 Teoria DDC

Con lo sviluppo della scienza e tecnologia molti processi industriali, ad esempio nella chimica, metallurgia, nell'elettronica nella logistica hanno subito profondi cambiamenti. L'aumentare delle dimensioni delle aziende, della produzione, delle attrezzature, gli stessi processi produttivi divenuti più complessi ha portato a considerare altre strade dalla tradizionale teoria MBC che diviene impraticabile in queste circostanze. Dall'altra parte si hanno processi che generano e immagazzinano una enorme quantità di dati, i quali contengono tutte informazioni sulle operazioni del processo e sulle attrezzature. In questo caso è di grande importanza l'utilizzo di questi dati acquisiti online o offline, direttamente per definire il controllore, monitorare, predire l'andamento del processo industriale quando non è possibile determinare un modello sufficientemente accurato. Fino a oggi esistono diversi metodi DDC, come il classico e noto controllore PID *Proporzionale Integrativo Derivativo*, MFAC *Model-Free Adaptive Control*, IFT *Iterative Feedback Tuning*, VRTF *Virtual Reference Feedback Tuning* e altri ancora. Nonostante questo, gli studi sulla teoria DDC sono ancora allo stato embrionale e attraggono l'attenzione della comunità scientifica.

Le definizioni di teoria DDC nella letteratura e rintracciabili facilmente su internet sono differenti:

- 1- *Data Driven Control* è una teoria e metodo di controllo nel quale il controllore è designato utilizzando solamente dati I/O presi online o offline del sistema di controllo, utilizzando la conoscenza dei dati processati senza l'utilizzo di informazioni, prese implicitamente o esplicitamente, dal modello matematico

del processo controllato, la cui stabilità, convergenza e robustezza possono essere garantite da una rigorosa analisi, sotto opportune condizioni.

2- *Data Driven Control* è la sintesi di un controllore utilizzando le misurazioni acquisite sul vero sistema da controllare, senza l'esplicito utilizzo di modelli, parametrici e non, del sistema da controllare. L'architettura DDC è mostrata in *figura 2*.

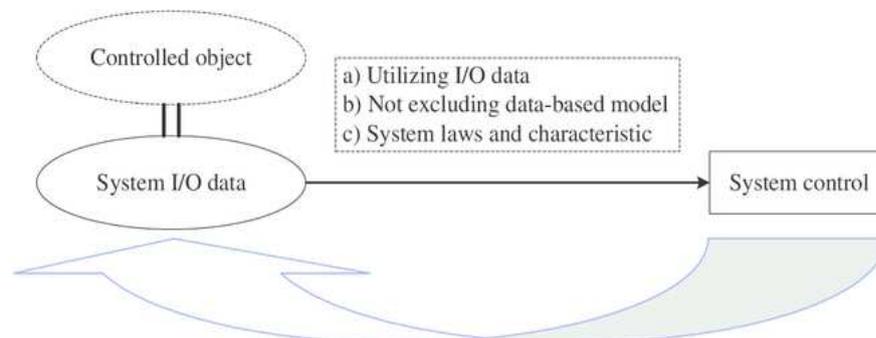


Figura 2: Architettura DDC

3.1 Necessarietà della teoria e dei metodi DDC

Il sistema di controllo consiste di due parti principali: il processo controllato e il controllore. Il processo da controllare può essere diviso in quattro classi differenti:

C1. Il modello è disponibile, ottenibile con metodi discussi sopra ed è sufficientemente accurato per realizzare un buon controllore.

C2. La definizione del modello o la sua identificazione è inaccurata o incerta.

C3. Il modello è identificabile ma complesso e con un grado elevato con forti non linearità.

C4. L'identificazione del processo non è possibile da realizzare.

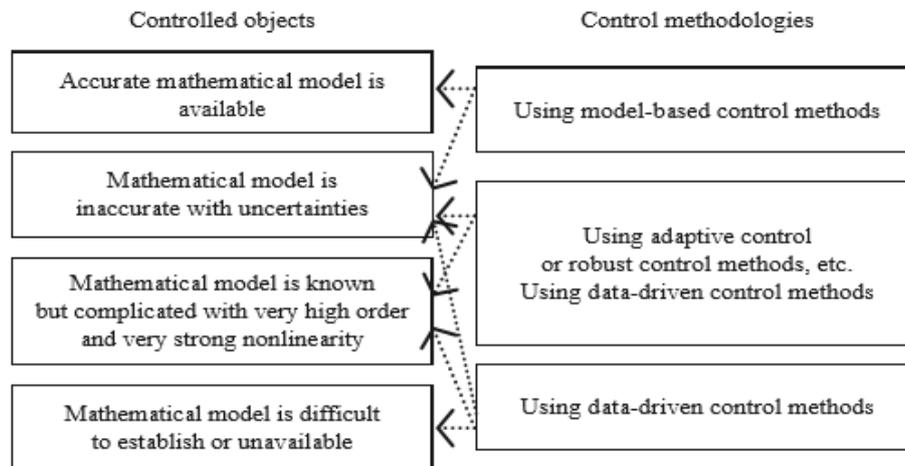


Figura 3: Oggetti controllati da DDC

Come mostrato in *figura 3* in modo semplice, solo la metà delle situazioni è descrivibile utilizzando la teoria MBC e non offre un buon modo per approcciarsi all'altra metà delle situazioni. I sistemi basati su dati I/O invece, possono essere sempre ottenuti senza contare la tipologia dell'oggetto controllato. Quindi è raccomandato l'utilizzo di metodi DDC mentre risulta essere una scelta inevitabile quando il modello non è disponibile o presenta incertezze. Una completa teoria di controllo dovrebbe coprire tutte le metodologie e situazioni per le tipologie di sistemi di controllo sopra menzionate. Si deduce come una teoria completa consiste in una parte della teoria MBC e parte della teoria DDC. Dal punto di vista teorico i metodi MBC spesso portano a mancanza di robustezza e non sono applicabili nel momento in cui il modello matematico del sistema non è disponibile e generalmente un ordine non lineare troppo elevato costringe a una successiva semplificazione e riduzione di ordine e robustezza del controllore. Dal punto di vista dell'applicazione, un sistema a basso costo è spesso la prima richiesta da soddisfare nella maggior parte delle situazioni e

spesso definire un modello matematico non è né semplice né economico. Gli strumenti matematici e le abilità professionali richieste per i metodi MBC mettono in difficoltà gli ingegneri non sentendosi a proprio agio per sistemi particolarmente complicati, aumentando il gap tra teoria e pratica. Storicamente lo sviluppo di un controllore segue alcune fasi: semplice regolazione del device e un controllore PID che funzioni senza modello matematico, classica teoria del controllo basata sulla funzione di trasferimento, la teoria MBC basata sul modello in spazio di stato, un controllo basato su una conoscenza approfondita del processo e in seguito un metodo DDC con lo scopo di rimuovere la dipendenza dal modello del processo.

È giusto far notare come i metodi DDC e i metodi MBC non sono mutuamente esclusivi, non possono essere rimpiazzati l'uno con l'altro: entrambi hanno i loro vantaggi e svantaggi. Il vantaggio di questa teoria sta nel fatto che non è più presente la dipendenza dal modello matematico del processo controllato evitando così i vari problemi causati da dinamiche non modellate e incertezze sulle parti modellate. Fino a ora sono state sviluppate più di dieci differenti tipologie di metodi DDC in letteratura a seconda della tipologia dei dati utilizzati si possono dividere in tre classi. La prima utilizza solo dati presi offline, la seconda che utilizza dati acquisiti online e una terza che utilizza entrambi.

3.2 Model Free Adaptive Control

L'approccio MFAC è stato proposto inizialmente nel 1994 da *Zhongsheng Hou* per una classe di sistemi non lineari a tempo discreto. Invece di identificare il modello non lineare del processo, viene costruito un modello equivalente linearizzato dinamico e virtuale ad ogni operazione del sistema a ciclo chiuso utilizzando una nuova tecnica dinamica di linearizzazione con un nuovo concetto chiamato *Pseudo Partial Derivative* PPD. Con l'aiuto del modello virtuale, un controllore adattativo è utilizzato per il sistema non lineare. Il fattore tempo variante PPD è stimato utilizzando semplicemente i dati del processo controllato. La tecnica di linearizzazione include il modello dai dati *Compact Form Dynamic Linearization* CFDL, *Partial-Form Dynamic Linearization* PFDL, *Full-Form Dynamic Linearization* FFDL. Confrontato con altri schemi di controllo adattativo tradizionali, il metodo MFAC presenta diversi meriti il che lo rende più adatto in molte applicazioni pratiche. In primo luogo, MFAC è un metodo puramente DDC dipendendo solamente dai dati ottenuti dal processo che per una classe di processi può essere definito in modo indipendente e in secondo luogo non ha bisogno di nessun segnale di test esterno rendendo l'implementazione a basso costo. Terzo punto a favore è che MFAC è facile da implementare con un piccolo sforzo computazionale e una forte robustezza e inoltre, sotto alcune assunzioni, viene garantita la convergenza monotona e la stabilità BIBO del modello basato su CFDL e anche PFDL.

4 Tecniche di linearizzazione per sistemi non lineari a tempo discreto

L'analisi e il design dei controllori per sistemi lineari a tempo discreto e tempo invarianti è stato portato avanti con successo e ha trovato numerose applicazioni nella pratica. Il modello lineare, tuttavia, è una approssimazione del sistema reale, il quale quasi sempre è non lineare. Proprio per questo risulta molto importante lo studio e controllo dei modelli non lineari anche se risulta molto più complesso. NARMAX, *Nonlinear Autoregressive Moving Average with the Exogenous Input*, è una descrizione generica di sistemi a tempo discreto non lineari non affini. Il controllo di tali sistemi non è semplice a causa delle non linearità intrinseche rispetto all'input di controllo. Approccio comune con questo tipo di sistemi è convertire l'originale NARMAX in un modello linearizzato che può essere studiato con maggiore facilità nel framework dei sistemi lineari. Ci sono diverse tipologie di metodi di linearizzazione come quella con feedback, quella di *Taylor*, o la linearizzazione basata sulla funzione ortogonale e altre ancora. Tutte queste tipologie di linearizzazioni hanno delle limitazioni, quella con feedback ha bisogno di un modello matematico molto accurato del sistema da controllare per essere efficace, la linearizzazione di *Taylor* spesso trascura i termini di ordine più elevato, offrendo solo una approssimazione del vero sistema e portando a difficoltà nell'analisi e nel design del controllore a causa di quelle omissioni. Questi metodi per linearizzare non sono orientati al design di un controllore; ad esempio il metodo basato sulla funzione ortogonale potrebbe condurre spesso a un modello con

molti parametri o un ordine molto elevato, complicando di conseguenza il controllore non rendendolo adatto all'applicazioni pratiche.

Un nuovo approccio per una linearizzazione dinamica, metodo per definire un controllore, è stato sviluppato per una classe di sistemi a tempo discreto non lineari basato sul nuovo concetto del PPD *Pseudo Partial Derivative*. Utilizzando questo nuovo metodo una serie di linearizzazioni dinamiche di modelli di dati nella forma di dati I/O sono proposti: i modelli di dati CFDL e quelli PFDL.

4.1 Compact Form Dynamic Linearization

Si consideri una classe di sistemi SISO non lineari a tempo discreto

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

Dove $u(k) \in \mathbb{R}$ e $y(k) \in \mathbb{R}$ sono l'input e l'output del controllore all'istante k rispettivamente n_y e n_u sono due interi positivi non noti e $(\dots): \mathbb{R}^{n_u+n_y+2} \rightarrow \mathbb{R}$ è una funzione non lineare non nota.

Prima che si possa definire il metodo CFDL vengono fatte delle assunzioni che devono essere verificate

Assunzione 1: La derivata parziale di $f(\dots)$ rispetto alla variabile $(n_y + 2)$ -esima è continua per tutti i k .

Assunzione 2: Il sistema (1) soddisfa la condizione generalizzata di *Lipschitz* per ogni k

$$|y(k_1 + 1) - y(k_2 + 1)| \leq b |u(k_1) - u(k_2)|$$

con $u(k_1) \neq u(k_2)$ per $k_1 \neq k_2$, $k_1, k_2 \geq 0$, dove

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

$i=1,2$ e b è una costante positiva.

Le assunzioni 1 e 2, sopra definite, per il processo sono ragionevoli, la prima è una tipica restrizione fatta per i sistemi non lineari nell'ambito del controllo mentre la seconda impone un limite superiore sulla velocità con cui varia l'uscita del sistema controllata dal suo ingresso.

Dal punto di vista energetico, la variazione di energia interna del sistema non può andare a infinito se la variazione di energia dell'input di controllo è finita. Molti sistemi reali soddisfano questo tipo di assunzioni in particolare sistemi di controllo di temperatura, pressione, livello di liquidi. Successivamente verrà utilizzata la notazione $\Delta y(k+1) = y(k+1) - y(k)$ per la variazione dell'uscita e $\Delta u(k) = u(k) - u(k-1)$ per la variazione del segnale di ingresso tra due istanti consecutivi.

Teorema 1: Considerando il sistema (1) che soddisfi le assunzioni 1 e 2. Se $|\Delta u(k)| \neq 0$ allora esiste un parametro tempo variante $\varphi_c(k) \in R$, chiamato *pseudo partial derivative* (PPD), Il sistema (1) può essere riscritto nel seguente CFDL modello

$$\Delta y(k+1) = \varphi_c(k) \Delta u(k) \tag{2}$$

Con $\varphi_c(k)$ limitato per ogni k .

Il parametro PPD è dipendente dal tempo anche se il sistema è di tipo LTI. Si nota come PPD sia legato ai valori dei segnali di ingresso e uscita sino all'istante presente. Per semplicità con la rappresentazione $\varphi_c(k)$ si denotano con gli indici tutti i valori fino all'istante k . Se il periodo di campionamento e $\Delta u(k)$ non sono troppo ampi, il $\varphi_c(k)$ può essere inteso come un parametro che varia lentamente nel tempo. Il suo comportamento numerico potrebbe essere semplice da stimare. In altre parole, anche se la struttura e il ritardo del parametro sono espliciti nel modello con i principi primi, che sono difficili da gestire nella teoria MBC, la variazione del comportamento numerico del PPD potrebbe non essere così sensibile a questi fattori. Infine, PPD è più che altro un concetto a livello matematico, l'esistenza del PPD è teoricamente garantita da una rigorosa analisi ma in generale non può essere formulata analiticamente. Dal momento che il valore del teorema di *Cauchy* non può essere esplicitamente mostrato in forma analitica anche per una semplice funzione non lineare, così PPD non può essere calcolato analiticamente. Per un semplice sistema non lineare $y(k+1) = f(u(k))$, PPD rappresenta la derivata della funzione non lineare $f(\cdot)$ in un certo punto tra $u(k+1)$ e $u(k)$. L'interpretazione geometrica più esplicativa è mostrata in *figura 4*. La limitatezza di PPD significa che la funzione non lineare non varia in modo brusco.

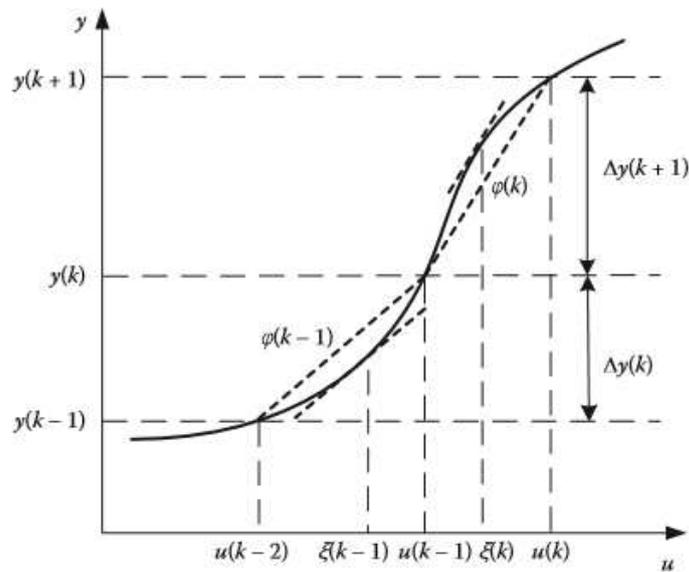


Figura 4: Interpretazione geometrica di PPD

Questa condizione è soddisfatta da molti sistemi nella pratica dal momento che le variabili input e output sono legate all'energia. La linearizzazione dinamica del modello dei dati dipende unicamente dai dati I/O del processo controllato a ciclo chiuso. Né informazioni strutturali né parametriche del modello del processo sono coinvolte, implicitamente o esplicitamente, nell'applicazione del metodo di linearizzazione proposto. È un metodo *data-driven*. Il risultato della linearizzazione del modello di dati (2) è abbastanza accurato ed è calcolato a ogni passo di campionamento. Questo metodo è orientato prettamente per il design di un controllore il che significa che non può essere utilizzato per altri fini.

Per un sistema lineare un criterio per stabilire la controllabilità di $x(k+1) = Ax(k) + Bu(k)$ è quello di calcolare il rango della matrice di controllabilità $[B \ AB \ \dots \ A^{n-1}B]$. dal momento che tutti gli stati nel passato e nel futuro possono essere completamente determinati dai coefficienti matriciali nel modello lineare, l'esistenza di ingressi

possibili può essere garantita dalla non singolarità della matrice di controllabilità e possiamo affermare che il test di controllabilità per sistema lineare è *a priori*. In un approccio *data-driven* le uniche informazioni disponibili sul sistema sono le misurazioni I/O fino all'istante presente, dal momento che gli stati e il modello in spazio di stato è sconosciuto, mentre i dati nel futuro non sono ancora disponibili. Per questi motivi, non è facile discutere la controllabilità del sistema non lineare originale o dell'equivalente linearizzato dal punto di vista della definizione classica di controllabilità in spazio di stato. Per questo motivo si introduce la “*controllabilità dell'uscita*”. Un sistema si dice essere controllabile dall'uscita per certi dati I/O all'istante, se l'uscita del sistema può essere guidata verso un valore specifico entro un tempo finito e una sequenza finita di ingressi. È evidente come il sistema (1) sia controllabile dall'uscita per uno specifico punto scelto se PPD $\varphi_c(k)$ non è né 0 né infinito per ogni k . La grandezza di $\varphi_c(k)$, che viene utilizzata per determinare se il sistema è controllabile dall'uscita, è stimata solamente utilizzando dati I/O presenti sino all'istante k . È da sottolineare come il PPD $\varphi_c(k)$ non può essere determinato analiticamente come per i sistemi lineari nella matrice della controllabilità, la sua limitatezza e diversità da zero di solito può essere verificata utilizzando i dati raccolti nel sistema a ciclo chiuso. Dal *Teorema 1* si può vedere come la limitatezza di PPD $\varphi_c(k)$ è garantita dalle condizioni generalizzate di *Lipschitz* fatte sul sistema. Il teorema richiede che $|\Delta u(k)| \neq 0$ sia soddisfatta per ogni k . Se effettivamente accadesse di $\Delta u(k) = 0$ una nuova linearizzazione, leggermente differente, può essere applicata, dopo essersi spostati di $\sigma_k \in Z$ istanti di tempo fino a $|u(k)| \neq u(k - \sigma_k)$.

Teorema 2: per sistema non lineari (1), soddisfatte l'Assunzione 1 e 2, se esiste un intero $k_0 \geq 1$ tale per cui

$$\Delta u(j) = 0 \text{ se } j = 1, \dots, k_0 - 1 \quad \text{o} \quad \Delta u(j) \neq 0 \text{ se } j = k_0$$

allora per ogni $k \geq k_0$ un intero finito σ_k può essere sempre trovato tale per cui:

$$\Delta u(k - j) = 0 \text{ se } j = 1, \dots, \sigma_k - 2 \quad \text{o} \quad \Delta u(k - j) \neq 0 \text{ se } j = \sigma_k - 1$$

Allo stesso tempo esiste un PPD $\varphi_c(k)$, e dunque il sistema (1) può essere trasformato nel seguente modello di dati CFDL:

$$y(k + 1) - y(k - \sigma_k + 1) = \varphi_c(k)(u(k) - u(k - \sigma_k))$$

Con $\varphi_c(k) \leq b$ per ogni istante k .

4.2 Partial Form Dynamic Linearization

Da quanto espresso prima si vede come il metodo CFDL trasformi un generale sistema non lineare a tempo discreto in un modello lineare, il quale ha un unico parametro scalare $\varphi_c(k)$. Tutti i possibili comportamenti complicati, come non linearità, parametri tempo varianti ecc. del sistema originale sono compressi e fusi in un unico scalare parametro tempo variante $\varphi_c(k)$. Proprio per questo il comportamento di PPD $\varphi_c(k)$ potrebbe essere troppo complicato per essere stimato. Il metodo CFDL considera semplicemente la relazione tra la variazione dell'uscita del sistema all'istante successivo e quello dell'ingresso all'istante presente. Infatti, la variazione dell'uscita del sistema all'istante successivo potrebbe essere legata al comportamento

dell'uscita a istanti precedenti oltre che alla variazione dell'input di controllo all'istante presente. Su questa considerazione viene esposto un nuovo metodo di linearizzazione chiamato *Partial Form Dynamic Linearization* PFDL, il quale considera l'influenza sull'uscita del sistema all'istante prossimo da parte di un ingresso di controllo di una lunghezza stabilita di istanti di tempo precedenti. Teoricamente il possibile comportamento anomalo del sistema originale dovrebbe essere meglio interpretato, corretto e addolcito dall'introduzione di un numero superiore di parametri nel metodo PFDL, chiaramente migliorando il semplice CFDL. Si noti $U_L(k) \in R^L$ come vettore di tutti i segnali di controllo nella finestra mobile di dimensioni $[k - L + 1, k]$, $U_L(k) = [u(k), \dots, u(k - L + 1)]^T$, con $U_L(k) = 0$ per $k \leq 0$. L'intero L è chiamato per l'input di controllo *Linearization Length Constant* LLC.

Assunzione 3: Le derivate parziali di $f(\dots)$ rispetto alle variabili dalle $(n_y + 2)$ -esime alla $(n_y + L + 1)$ e vale a dire $u(k), \dots, u(k-L+1)$, sono continue per tutti i k .

Assunzione 4: Il sistema (1) soddisfa la condizione generalizzata di Lipschitz per ogni k

$$|y(k_1 + 1) - y(k_2 + 1)| \leq b |U_L(k_1) - U_L(k_2)|$$

Per $U_L(k_1) \neq U_L(k_2)$ e con $k_1 \neq k_2$, $k_1, k_2 \geq 0$, dove $y(k_i + 1) = f(y(k_i), \dots, y(k_i - n_y), u(k_i), \dots, u(k_i - n_y))$ $i=1,2$, e b è una costante positiva.

Posto $\Delta U_L(k) = U_L(k) - U_L(k - 1)$ si propone il seguente con il seguente teorema il metodo PFDL sul sistema (1).

Teorema 3: Considerato il sistema non lineare (1) che soddisfi le *Assunzioni 3 e 4*. Per ogni L fissato, se $|\Delta U_L(k)| \neq 0$, allora esiste un vettore, tempo variante, $\varphi_{p,L}(k) \in R^L$, chiamato *Pseudo Gradient* PG, tale che il sistema (1) possa essere trasformato nel seguente modello di dati PFDL:

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

Con $\varphi_{p,L}(k) = [\varphi_1(k), \dots, \varphi_L(k)]^T$ per ogni istante k .

In particolare, da notare come il modello di dati PFDL diventi quello CFDL nel momento in cui $L=1$. Inoltre, le dimensioni di PG nel modello di dati PFDL aumentano comparato con il modello CFDL, la complessità delle dinamiche PG nel modello PFDL sono ridotte a quelle di PPD del modello CFDL. Conseguentemente, $\varphi_{p,L}(k)$ viene facilmente stimato con l'algoritmo utilizzando i dati input-output del sistema a ciclo chiuso quando PFDL viene utilizzato nei sistemi di controllo. Oltretutto si possono ottenere differenti modelli di dati PFDL selezionando un differente input di controllo $LLC L$. La flessibilità dell'approccio di linearizzazione dinamico nel descrivere il sistema non lineare originale può essere migliorata utilizzando differenti modelli PFDL, scegliendo una dimensione appropriata di PG e L .

5 Model Free Adaptive Control di sistemi SISO non lineari

Teorie e metodologie di controllo adattativo assumo la struttura del processo controllato come nota, mentre i parametri come non noti, tempo varianti o invariante che siano. Al giorno d'oggi lo studio e sviluppo del controllo adattativo per sistemi lineari è ben maturato e ha trovato applicazione in molti settori. Sta di fatto, che la non linearità è molto comune nei sistemi nel mondo reale come aeroplani, robot, processi industriali, sistemi del traffico ecc. La ricerca del controllo adattativo per sistemi non lineari ha attratto la ricerca negli ultimi anni. I metodi classici di progettazione del controllo adattativo per sistemi non lineari includono quello basato sulla linearizzazione del feedback, il metodo *backstepping*, metodo a modello multiplo, controllo predittivo adattativo e così via. Tutti questi metodi sono tecniche di progettazione di controllo basate su un modello, dal momento che è richiesto un modello accurato del processo del sistema controllato per definire e analizzare il controllore.

Quando la conoscenza di un modello del sistema non è disponibile o la dinamica ne presenta incertezze non trascurabili, i metodi citati sopra non sono applicabili. Come già detto in precedenza costruire un modello in spazio di stato di un impianto reale non è semplice e a volte dinamiche non modellate sono inevitabili. Le applicazioni di tali metodi su un modello con incertezze, sulla base di ipotesi matematiche aggiuntive, possono causare problemi e essere anche pericolose. Sebbene il controllo robusto sia stato ben approfondito, solo pochi metodi in realtà riescono a superare i problemi di

un controllo pratico ed essere usati con successo. Pertanto, è molto importante sviluppare questo metodo MFAC per sistemi non lineari, non solo in teoria ma anche per applicazioni pratiche.

MFAC è stato proposto per la prima volta nel 1994 per una classe di sistemi non lineari. Nel corso dei due decenni successivi c'è stato un miglioramento e sviluppo, è stato stabilito un quadro sistematico dei metodi MFAC. L'idea di base del progetto MFAC è implementata prima costruendo una dinamica equivalente di un modello di dati di una linearizzazione del sistema ad ogni operazione, in seguito stimando il valore di PPD del sistema online, utilizzando dati I/O del processo controllato e definendo un controllore in accordo con delle funzioni di costo un passo di campionamento avanti pesate in modo opportuno. Una serie di risultati di schemi e analisi è presentata, sulla base della linearizzazione dinamica dei modelli presentati nel capitolo precedente per una classe di sistemi SISO non lineari a tempo discreto. Tra i meriti di questo metodo, c'è in primo luogo il fatto che solo i dati misurati I/O del sistema controllato a ciclo chiuso, piuttosto che informazioni sul modello del sistema reale, sono necessari per la progettazione del controller. Le tradizionali parti non modellate non sono più un problema, non esistono in questo framework basato sui dati rendendo il metodo MFAC adatto per sistemi industriali. In secondo luogo, MFAC è un metodo di controllo a basso costo e carico computazionale e struttura semplice, poiché non è necessario un modello matematico, un segnale sperimentale o di test per tarare il processo. Infine, un controllo adattativo per un sistema non lineare con parametri e struttura che variano nel tempo può essere realizzato applicando proprio il metodo MFAC.

5.1 Modello di dati CFDL basato su MFAC

Consideriamo una classe di sistemi SISO a tempo discreto non lineari come segue:

$$y(k+1) = f\left(y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u)\right) \quad (3)$$

dove $u(k) \in \mathbb{R}$ e $y(k) \in \mathbb{R}$ sono l'input e l'output del controllore all'istante k rispettivamente n_y e n_u sono due interi positivi non noti e $f(\dots): \mathbb{R}^{n_u+n_y+2} \rightarrow \mathbb{R}$ è una funzione non lineare non nota.

Dal *Teorema 1* si può notare come il sistema soddisfi le *Assunzioni 1 e 2* con $|\Delta u(k)| \neq 0$ per tutti i k e quindi possiamo trasformarlo nel seguente CFDL modello:

$$\Delta y(k+1) = \varphi_c(k) \Delta u(k) \quad (4)$$

con $\varphi_c(k) \in \mathbb{R}$ il limite PPD del sistema (3).

L'equazione (4) è un equivalente virtuale dinamico e linearizzato del sistema non lineare. Si tratta di un modello lineare di dati variabile nel tempo, orientato alla progettazione del controller, con un parametro scalare e in una semplice forma incrementale. In questo senso, è totalmente diverso dagli altri modelli linearizzati derivati dai principi primi e da altri metodi di linearizzazione. Lo schema del sistema di controllo si baserà su quella linearizzazione virtuale del modello dato.

5.1.1 Algoritmo di controllo

Per un sistema a tempo discreto, l'algoritmo di controllo ottenuto dalla predizione di un passo avanti della funzione dell'errore potrebbe richiedere uno sforzo di controllo eccessivo il quale potrebbe danneggiare il sistema di controllo stesso, mentre l'algoritmo del controllore ottenuto pesando la funzione dell'errore anch'essa presa un passo avanti potrebbe portare a un errore di inseguimento costante. Quindi si usa la seguente funzione per definire l'algoritmo del controllore:

$$J(u(k)) = |y^*(k+1) - y(k+1)|^2 + \lambda |u(k) - u(k-1)|^2 \quad (5)$$

dove $\lambda > 0$ è un fattore che introduce delle restrizioni sul variare dell'ingresso di controllo, $y^*(k+1)$ è l'uscita di riferimento desiderata. Sostituendo il modello di dati CFDL (3) nella funzione di costo (4), differenziando rispetto a $u(k)$ la funzione (5) si ottiene la seguente legge di controllo:

$$u(k) = u(k-1) + \frac{\rho \varphi_c(k)}{\lambda + |\varphi_c(k)|^2} (y^*(k+1) - y(k)) \quad (6)$$

Dove il fattore $\rho \in (0,1]$ è aggiunto per rendere l'algoritmo di controllo più generale.

È ovvio come l'aggiunta del termine λ nell'algoritmo di controllo (6) è un fattore che penalizza il variare dell'input di controllo $\Delta u(k)$, per garantire un segnale di controllo levigato. λ è un importante parametro per il design del controllo MFAC, una scelta adeguata di λ può garantire stabilità e un segnale in uscita del sistema di controllo che segua il riferimento. L'algoritmo (6) può essere utilizzato esclusivamente se PPD è noto. L'accuratezza del valore PPD è difficile da ottenere dal momento che il modello

del sistema è ignoto e il parametro PPD è tempo variante come discusso in precedenza. Questo porta alla necessità di definire un certo parametro tempo variante che stimi PPD online attraverso I/O data del processo controllato. La situazione rende necessaria una nuova funzione che stimi PPD come segue:

$$J(\varphi_c(k)) = |y(k) - y(k-1) + \varphi_c(k)\Delta u(k-1)|^2 + \mu|\varphi_c(k) - \varphi_c(k-1)| \quad (7)$$

dove $\mu > 0$ è fattore di peso.

Minimizzando la funzione (7) rispetto a φ_c si ottiene il seguente algoritmo di stima di PPD:

$$\widehat{\varphi}_c(k) = \widehat{\varphi}_c(k-1) + \frac{\eta \Delta u(k-1)}{\mu + \Delta u(k-1)^2} (\Delta y(k) - \widehat{\varphi}_c(k-1)\Delta u(k-1)) \quad (8)$$

dove $\eta \in (0,2]$ è aggiunto per rendere l'algoritmo più generale e flessibile e dunque $\widehat{\varphi}_c(k)$ rappresenta una stima per φ_c . L'introduzione del termine μ al denominatore impedisce la divisione per 0 e determina un fattore che penalizza il variare del termine che stima PPD.

5.1.2 Schema di controllo

Integrando l'algoritmo di controllo (4) con la stima di PPD otteniamo il seguente schema CFDL - MFAC:

$$\widehat{\varphi}_c(k) = \widehat{\varphi}_c(k-1) + \frac{\eta \Delta u(k-1)}{\mu + \Delta u(k-1)^2} (\Delta y(k) - \widehat{\varphi}_c(k-1) \Delta u(k-1)) \quad (9)$$

$$\widehat{\varphi}_c(k) = \widehat{\varphi}_c(1) \quad \text{se } |\widehat{\varphi}_c(k)| \leq \varepsilon \quad \text{o } |\Delta u(k-1)| \leq \varepsilon \quad \text{o}$$

$$\text{sign}(\widehat{\varphi}_c(k)) \neq \text{sign}(\widehat{\varphi}_c(1)) \quad (10)$$

$$u(k) = u(k-1) + \frac{\rho \varphi_c(k)}{\lambda + |\varphi_c(k)|^2} (y^*(k+1) - y(k)) \quad (11)$$

dove $\lambda > 0$, $\mu > 0$, $\rho \in (0,1]$, $\eta \in (0,2]$, ε è una costante piccola positiva, $\widehat{\varphi}_c(1)$ è il valore iniziale di $\widehat{\varphi}_c(k)$.

La condizione inserita nell'algoritmo abilita il parametro di stima a seguire con dei buoni risultati di il parametro tempo variante. Dallo schema CFDL- MFAC possiamo vedere come lo schema MFAC utilizzi solamente dati I/O del sistema a ciclo chiuso senza l'utilizzo implicito o esplicito di alcuna informazione sul modello del sistema dinamico e per questa ragione lo si definisce “*model free*” adaptive control. Dal momento che PPD $\varphi_c(k)$ non è sensibile al variare dei parametri, alla struttura o al ritardo, lo schema CFDL-MFAC ha una forte capacità di adattamento e robustezza che sono difficili da ottenere nel framework del control design su base adattiva. Questo non significa che il metodo sia universale, non a caso è applicabile solo a una classe di sistemi non lineari a tempo discreto che seguono una serie di assunzioni.

Per studiare e analizzare la stabilità del sono fatte due assunzioni sul sistema come segue:

Assunzione 5: per una uscita desiderata limitata $y^*(k+1)$, esiste un ingresso di controllo limitato $u^*(k)$, tale per cui l'uscita del sistema guidata da $u^*(k)$ è uguale a $y^*(k+1)$.

Assunzione 6 il segno di PPD si ritiene immutato per ogni k e $\Delta u(k) \neq 0$, che $\varphi_c(k) > \varepsilon$ è soddisfatta, dove ε è una costante piccola positiva.

La prima assunzione implica che il problema di controllo è risolvibile, quindi il sistema risulta essere controllabile dall'uscita. Il significato fisico della seconda assunzione è che l'uscita non può diminuire se l'ingresso di controllo aumenta, il che potrebbe essere visto come una sorta di caratteristica lineare.

Teorema 4: Se il sistema non lineare (3), soddisfa le assunzioni 1, 2 precedenti e la 5 e 6, è controllato da uno schema CFDL-MFAC (9), (11) per risolvere il problema che è $y^*(k+1) = y^* = \text{const}$. Inoltre, esiste una costante $\lambda_{min} > 0$ tale per cui sono valide le seguenti proprietà per ogni $\lambda > \lambda_{min}$

- L'errore di tracciamento dell'uscita converge in modo monotono,

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

- Il sistema a ciclo chiuso è BIBO stabile, $y(k)$ e $u(k)$ sono limitati.

Il teorema è la prova che lo schema di controllo MFAC applicato a processi non lineari sconosciuti può garantire la stabilità e la convergenza dell'errore.

La differenza significativa tra il controllo adattativo basato su modello e lo schema CFDL-MFAC proposto si trova in tre aspetti principali:

- I processi da controllare sono diversi; nello schema MFAC la classe di sistemi non lineari è sconosciuta mentre il controllo adattativo tradizionale è basato su un modello il cui sistema è invariante nel tempo o varia lentamente con una struttura e ordine di sistemi noti.
- Le procedure di progettazione sono differenti. Per un approccio MFAC, ad ogni operazione viene creato un modello equivalente linearizzato di dati incrementale a ogni passo o operazione del sistema a ciclo chiuso con l'aiuto del concetto nuovo di PPD, il quale può essere stimato utilizzando solo i dati I/O del processo controllato, e quindi viene utilizzato per definire un algoritmo, un passo avanti, del controllore adattativo ponderato. Come conseguenza, nessuna informazione sulla struttura dell'impianto è necessaria per definire controllore e stimatore e non esistono così dinamiche non modellate. Per applicare il tradizionale metodo basato sul modello, il primo passo è modellare l'impianto utilizzando i principi primi o un approccio di identificazione del sistema, e quindi sviluppare un controllore che si basa sul modello matematico adattandosi a certi principi di equivalenza. Le performance del metodo MFAC non vengono influenzate dalla presenza di dinamiche non modellate.
- I metodi per analizzare la stabilità sono differenti; un approccio *data-driven* di mappatura della contrazione viene utilizzato per analizzare la stabilità del MFAC, mentre spesso viene utilizzato *Lyapunov* per il controllo adattativo classico.

5.2 PFDL modello di dati basato su MFAC

Come menzionato precedentemente, la dinamica del PPD $\varphi_c(k)$ nel modello di dati CFDL potrebbe essere troppo complicata quando il sistema non lineare è troppo complesso. Infatti, la variazione dell'uscita del sistema all'istante $k+1$ potrebbe essere sensibile alle variazioni dell'input di controllo all'interno di una certa finestra mobile di tempo all'istante k . Il controllore senza queste considerazioni potrebbe risultare instabile. Detto ciò, lo schema PFDL-MFAC è presentato, analizzandone la stabilità, sulla base del modello di dati PFDL. Confrontandolo con lo schema CFDL-MFAC e quello che si nota è la presenza di un numero maggiore di gradi di libertà e una maggiore flessibilità nel design del controllore. Dal *Teorema 4* il sistema non lineare (4) soddisfacendo le *Assunzioni 3, 4* con $|\Delta U_L(k)| \neq 0$ per ogni k , può essere trasformato nel seguente modello di dati PFDL:

$$\Delta y(k+1) = \varphi_{p,L}^T(k) \Delta U_L(k) \quad (11)$$

dove $\varphi_{p,L}(k) = [\varphi_1(k), \dots, \varphi_L(k)]^T \in R^L$ è lo sconosciuto ma limitato PG, $\Delta U_L(k) = [\Delta u(k), \dots, \Delta u(k-L+1)]^T$, e L è l'ingresso di controllo LLC.

5.2.1 Algoritmo di controllo

Si consideri la seguente funzione dell'ingresso di controllo:

$$J(u(k)) = |y^*(k+1) - y(k+1)|^2 + \lambda |u(k) - u(k-1)|^2 \quad (12)$$

Dove $\lambda > 0$ è un fattore di peso.

Sostituendo il modello di dati PFDL (11) nella funzione (12) e minimizzandola rispetto a $u(k)$ conduce al seguente algoritmo di controllo:

$$u(k) = u(k-1) + \frac{\rho \varphi_c(k)}{\lambda + |\varphi_c(k)|^2} (y^*(k+1) - y(k)) - \frac{\varphi_1(k) \sum_{i=2}^L \rho_i \varphi_i(k) \Delta u(k-i+1)}{\lambda + |\varphi_1(k)|^2} \quad (13)$$

Dove $\rho_i \in (0,1]$, $i = 1,2, \dots, L$ è aggiunto per dare maggiore flessibilità all'algoritmo. Analogamente a quanto fatto precedentemente la funzione di cost PG espressa come segue:

$$J(\varphi_{p,L}(k)) = |y(k) - y(k-1) - \varphi_{p,L}^T(k) \Delta U(k-1)|^2 + \mu |\varphi_{p,L}(k) - \varphi_{p,L}(k-1)|^2 \quad 14$$

Dove il parametro $\mu > 0$.

Minimizzando (14) rispetto $\varphi_{p,L}(k)$, seguendo le condizioni di ottimalità si ottiene:

$$\widehat{\varphi}_{p,L}(k) = \widehat{\varphi}_{p,L}(k-1) + \frac{\eta \Delta U_L(k-1) (y(k) - y(k-1) - \widehat{\varphi}_{p,L}^T(k-1) \Delta U_L(k-1))}{\mu + |\Delta U_L(k-1)|^2} \quad (15)$$

Dove $\eta \in (0,2]$ è un parametro aggiunto per una maggiore generalità, e $\widehat{\varphi}_{p,L}(k)$ è la stima del parametro non conosciuto PG $\varphi_{p,L}(k)$. Il parametro $\varphi_{p,L}(k)$ è un vettore tempo variante di dimensione L , al contrario dello scalare $\varphi_c(k)$ dello schema CFDL.

5.2.2 Schema di controllo

Dalla equazione (13) e (15), algoritmo di stima e del controllore, si può sintetizzare lo schema PFDL-MFAC come segue:

$$\widehat{\varphi}_{p,L}(k) = \widehat{\varphi}_{p,L}(k-1) + \frac{\eta \Delta U_L(k-1)(y(k) - y(k-1) - \widehat{\varphi}_{p,L}^T(k-1) \Delta U_L(k-1))}{\mu + |\Delta U_L(k-1)|^2} \quad (16)$$

$$\begin{aligned} \widehat{\varphi}_{p,L}(k) &= \widehat{\varphi}_{p,L}(1) \text{ se } |\widehat{\varphi}_{p,L}(k)| \leq \varepsilon \\ &\text{o } |\Delta U_L(k-1)| \leq \varepsilon \end{aligned} \quad (17)$$

$$\text{o } \text{sign}(\widehat{\varphi}_1(k)) \neq \text{sign}(\widehat{\varphi}_1(1))$$

$$\begin{aligned} u(k) &= u(k-1) + \frac{\rho \widehat{\varphi}_1(k)}{\lambda + |\widehat{\varphi}_1(k)|^2} (y^*(k+1) - y(k)) - \\ &\quad \left(\frac{\widehat{\varphi}_1(k) \sum_{i=2}^L \rho_i \widehat{\varphi}_i(k) \Delta u(k-i+1)}{\lambda + |\widehat{\varphi}_1(k)|^2} \right) \end{aligned} \quad (18)$$

Dove $\lambda > 0$, $\mu > 0$, $\eta \in (0,2]$, $\rho_i \in (0,1]$, $i = 1, 2, \dots, L$ e ε è una costante piccola e positiva e $\widehat{\varphi}_1(k)$ è il valore iniziale di $\widehat{\varphi}_{p,L}(k)$. Il meccanismo di reset è aggiunto per garantire l'esatta stima dei parametri, con una grande capacità di tracciare il suo andamento nel tempo.

C'è un vettore di dimensione L , chiamato $\widehat{\varphi}_{p,L}(k)$, che stima PG, che viene definito online nello schema PFDL-MFAC, mentre la sua lunghezza costante di linearizzazione LLC è anche modificabile. Varie simulazioni propongono di settare la dimensione di L tra 1 e la somma approssimativa dell'ordine del processo se n_u e n_y sono sconosciuti. Da un punto di vista pratico L dovrebbe essere settato come un intero pari a 1 per un sistema semplice e con una dimensione sempre maggiore per sistemi via, via più complessi. Inoltre, da notare come se $L=1$, lo schema PFDL-MFAC diviene lo schema

analizzato in precedenza CFDL-MAFC. Dunque, comparandoli, rispetto al CFDL si hanno più parametri su cui lavorare, aggiungendo flessibilità e più gradi di libertà con l'introduzione di fattori $\rho_1, \rho_2, \dots, \rho_L$.

Teorema 5: Se il sistema non lineare (4), soddisfa le assunzioni 1, 2, 5, e 6 fatte, controllato con lo schema PFDL-MFAC per far sì che $y^*(k+1) = y^* = \text{const}$ allora esiste una costante $\lambda_{min} > 0$ tale per cui sono valide le seguenti proprietà per ogni $\lambda > \lambda_{min}$

- L'errore di tracciamento dell'uscita converge in modo monotono,

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

- Il sistema a ciclo chiuso è BIBO stabile, $y(k)$ e $u(k)$ sono limitati.

6 Applicazione e analisi

In questo capitolo vengono esposti brevemente alcuni esempi, al fine di dimostrare l'efficacia dei metodi di controllo descritti nelle pagine precedenti.

6.1 Simulazioni metodo CFDL-MFAC

In questa sezione si esaminano simulazioni numeriche di due sistemi SISO a tempo discreto totalmente differenti. Entrambi i processi non lineari vengono controllati dallo stesso schema CFDL-MFAC. I modelli nei due esempi vengono utilizzati semplicemente per generare i dati I/O dell'impianto controllato, ma non per il progetto MFAC. Come premessa, viene proposto il modello CFDL-MFAC realizzato in *Matlab*, e *Simulink*, e lo schema di controllo, *figura 5* e *figura 6*.

```
function [phi_i,u_i,DeltaU_usc,DeltaU,yPPD_il]= fcn( e, y_i, y_il,yPPD_i, phi_i_11,u_i1,DeltaU1,epsistema, ro, mistima, etastima, lambda, phi11)

    Deltay=y_il-y_i;

    phi_i_1=phi_i_11;

    DeltaU=DeltaU1;

    phi_i=phi_i_1+ (etastima*DeltaU*(Deltay-phi_i_1*DeltaU)/(mistima+norm(DeltaU)^2));

    if (norm(phi_i)<=epsistema || norm(DeltaU)<=epsistema || (sign(phi_i(:,1))) ~= sign(phi11))
        phi_i(:,1)=phi11;
    end

    DeltayPPD=phi_i*DeltaU';

    yPPD_il = yPPD_i+DeltayPPD;

    u_i=u_i1+(ro*phi_i(:,1)*e/(lambda+phi_i(:,1)^2));

    DeltaU_1=u_i-u_i1;
    DeltaU_usc=DeltaU_1;
```

Figura 5: Algoritmo di controllo CFDL-MFAC

Nello schema *Simulink* viene utilizzata un blocco *Matlab Function* per realizzare il controllore e il processo mentre dei blocchi *Unit Delay* Z^{-1} per ottenere i passi precedenti di campionamento.

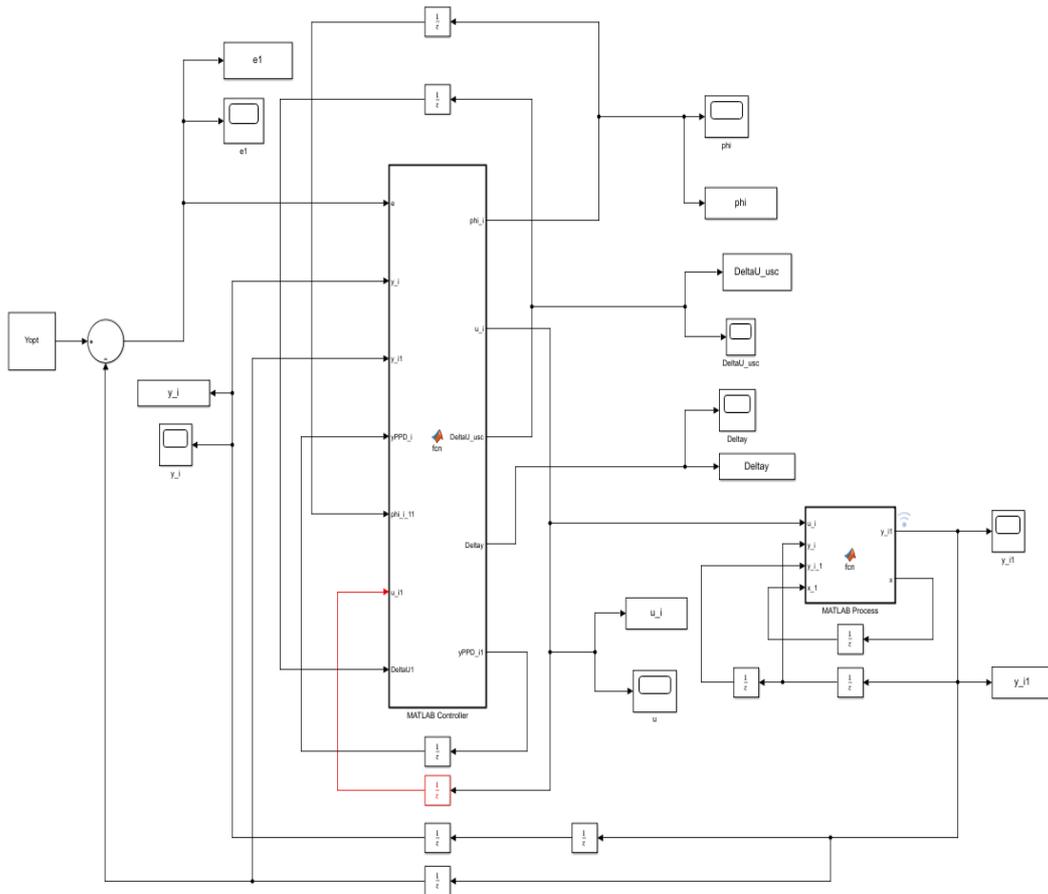


Figura 6: Schema Simulink CFDL-MFAC

6.1.1 Simulazione 1

Il primo è il controllo di uno scambiatore di cui si riporta non solo lo schema generale ma anche i parametri scelti per il controllo e relativi risultati

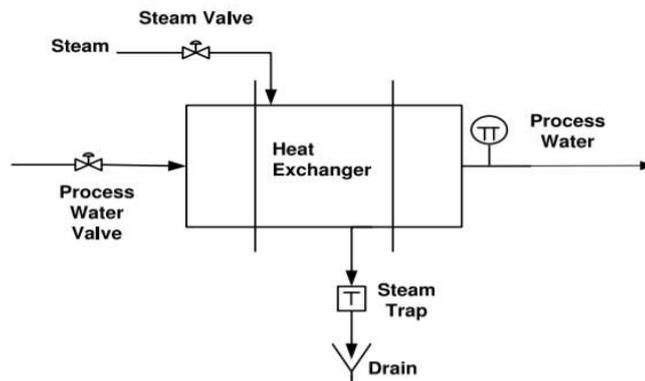


Figura 7: Schema scambiatore di calore

```
Param.m x MATLAB Process x +
function [y_i1,x] = fcn(u_i,y_i,y_i_1,x_1)

x = 1.5*u_i-1.5*u_i^2+0.5*u_i^3;
y_i1= 0.6*y_i-0.1*y_i_1+1.2*x-0.1*x_1;
```

Figura 8: Processo simulazione 1

Nella figura 8 viene riportata le dinamiche del processo dello scambiatore di calore, mentre, in figura 9, vengono riportati i parametri adottati per ottenere i risultati di seguito mostrati.

```
Param.m x +
1 - ro = 0.5;
2 - epsistema = 10^-5;
3 - etastima = 0.08;
4 - phill = 0.8;
5 - mistima = 1;
6 - lambda = 4;
7
8 - Tc = 0.1;
9 - Yopt=0.7;
10
```

Figura 9: Parametri simulazione 1

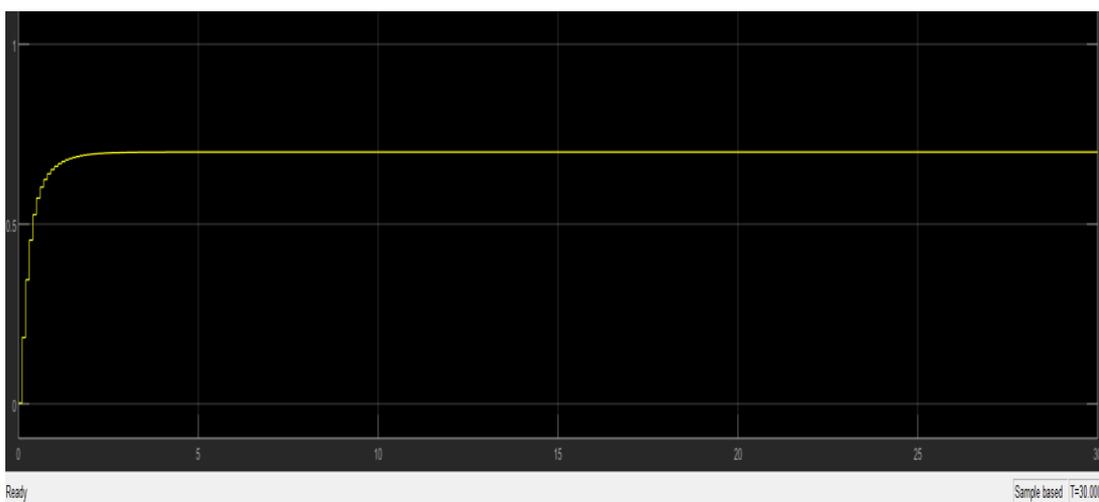


Figura 10: Uscita sistema simulazione 1



Figura 11: Stima di PPD simulazione 1

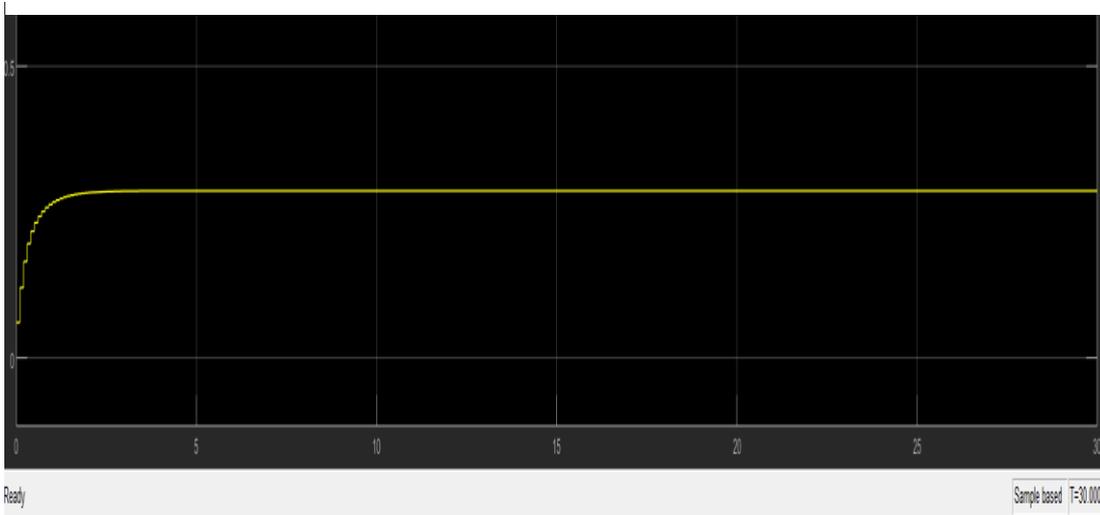


Figura 13: Uscita controllore simulazione 1



Figura 12: Errore tra uscita e riferimento simulazione 1

Da notare come la risposta sia rapida e l'errore, *figura 13*, vada a zero velocemente senza sovra elongazioni e rimanga tale per tutto il tempo della simulazione.

6.1.2 Simulazione 2

La seconda simulazione avviene sempre con lo schema proposto in *figura 6* e algoritmo CFDL-MFAC di *figura 5*. In questo caso il processo risulta essere più complesso e tempo variante. Del processo si riporta la dinamica in *figura 14*, e il blocco della *Matlab Function* utilizzato in *figura 15*.

```
function y_i1 = fcn(u_i,u_i_1,y_i,y_i_1,y_i_2,data)

k=data;
if k<=500 && k>=0
    y_i1= [(5*y_i*y_i_1)/(1+y_i^2+y_i_1^2+y_i_2^2)]+u_i+1.1*u_i_1;
else k > 500
    y_i1=[(2.5*y_i*y_i_1)/(1+y_i^2+y_i_1^2)]+1.2*u_i+1.4*u_i_1+0.7*sin(0.5*(y_i+y_i_1))*cos(0.5*(y_i+y_i_1));
end
```

Figura 14: Processo 2 simulazione 2

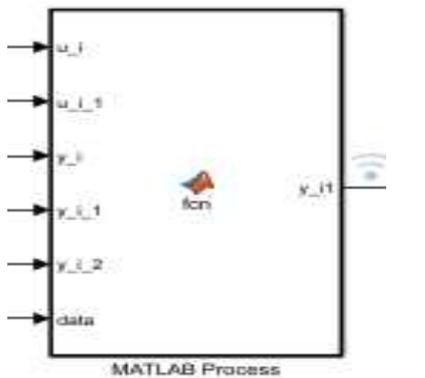


Figura 16: Matlab Function

```
ro = 0.6;
epsistema = 10^-5;
etastima = 1;
phi11 = 2;
mistima = 1;
lambda = 2;
|
Tc = 0.1;
```

Figura 15: Parametri simulazione 2

```
function Yopt = fcn(data)

k=data;
if k>=0 && k<=300
    Yopt=5*sin(3.1*k/50)+2*cos(3.1*k/100);
else k>= 300 && k<=700
    Yopt=5*(-1)^round(k/100);
end
```

Figura 17: Ingresso di riferimento del sistema simulazione 2

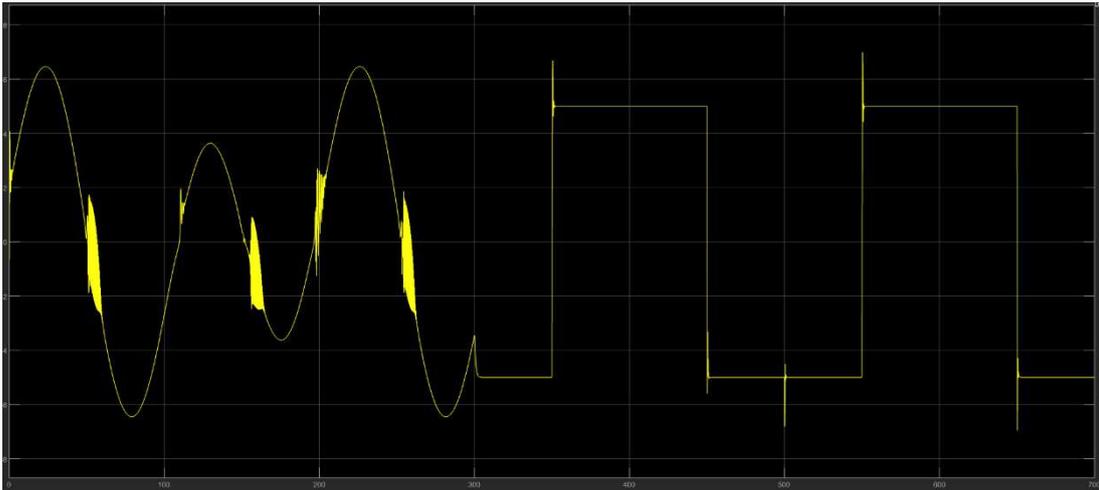


Figura 20: Uscita sistema simulazione 2

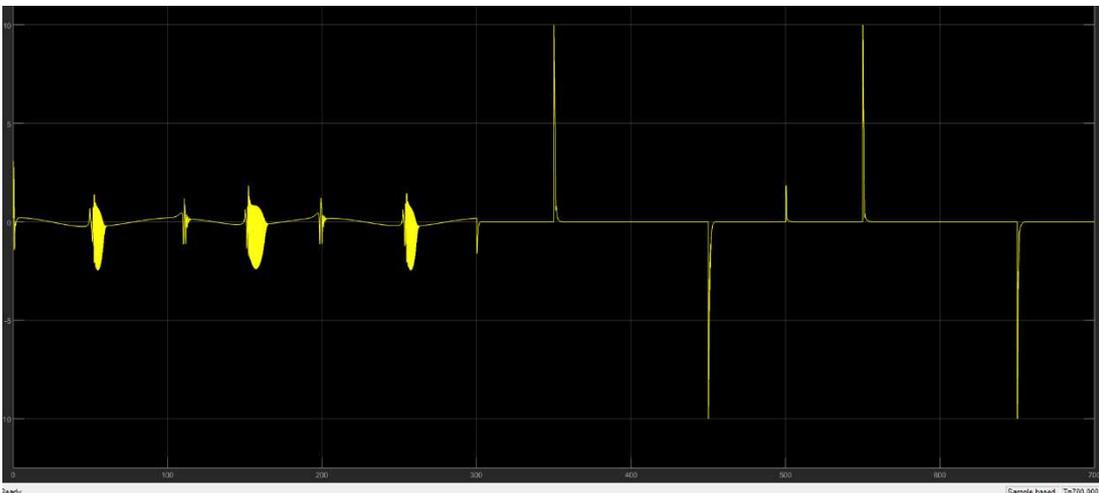


Figura 19: Errore tra uscita e riferimento simulazione 2

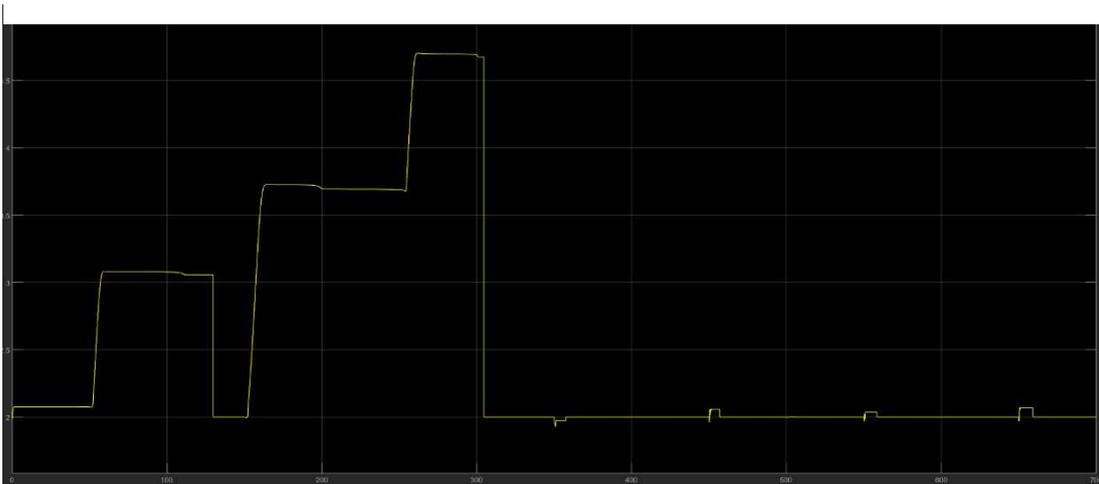


Figura 18: Stima di PPD simulazione 2

Analizzando i risultati della seconda simulazione, notiamo che l'errore non è sempre mantenuto a zero ma presenti alcuni picchi i quali comunque risultano essere limitati e di durata molto breve. Al contrario della prima simulazione si ricorda che non solo il processo è tempo variante ma lo è anche l'ingresso di riferimento, mostrato in *figura 17*. Risultati ottenuti sono un compromesso tra tempo di salita e sovra elongazione attraverso la gestione dei parametri e una selezione pesata dei vari fattori, *figura 15*. Una selezione differente del fattore λ porterebbe a risultati diversi. Una scelta di λ piccola comporta una risposta più pronta del sistema controllato e un tempo di salita minore ma allo stesso tempo una maggiore sovra elongazione. Per $\lambda = 0$ si ha spesso la possibilità di incorrere in instabilità del sistema a ciclo chiuso. Il comportamento del parametro PPD de sistema è relativamente semplice, è un parametro scalare che varia lentamente nel tempo. Infine, lo schema CFDL-MFAC ha una struttura semplice ed è molto semplice da implementare.

6.2 Simulazioni metodo PFDL-MFAC

In questa sezione vengono proposte le simulazioni per tre sistemi non lineari SISO a tempo discreto completamente differenti, controllati con lo stesso schema PFDL-MFAC. Rimanendo un metodo *Data Driven* i modelli negli esempi vengono utilizzati semplicemente per generare i dati I/O dell'impianto controllato, ma non per il progetto MFAC. Nella *figura 21* si propone il codice per l'algoritmo di controllo adottato, mentre in *figura 22* si mostra lo schema generale adottato.

```

function [phi_i,u_i,DeltaU_usc,DeltaY,yPPD_i1]= fcn( e, y_i, y_i1,yPPD_i, phi_i_11, phi_i_12,phi_i_13,u_i1,u_i2,u_i3,DeltaU1,

    Deltay=y_i1-y_i;

    phi_i_1=[phi_i_11,phi_i_12,phi_i_13];

    DeltaU=[DeltaU1,DeltaU2,DeltaU3];

    phi_i=phi_i_1+ (etastima*DeltaU*(Deltay-phi_i_1'*DeltaU)/(mistima+norm(DeltaU)^2));

    if (norm(phi_i)<=epsistima) || norm(DeltaU)<=epsistima || (sign(phi_i(:,1))) ~= sign(phi11)
        phi_i(:,1)=phi11;
        phi_i(:,2)=phi12;
        phi_i(:,3)=phi13;
    end

    DeltayPPD=phi_i*DeltaU';

    yPPD_i1 = yPPD_i+DeltayPPD;

    u_i=u_i1+(ro*phi_i(:,1)*e/(lambda+phi_i(:,1)^2)-phi_i(:,1)*(ro1*phi_i(:,2)*DeltaU(2)+ro2*phi_i(:,3)*DeltaU(3))/(lambda+

DeltaU_1=u_i-u_i1;
DeltaU_2=u_i1-u_i2;
DeltaU_3=u_i2-u_i3;
DeltaU_usc=[DeltaU_1,DeltaU_2,DeltaU_3];

```

Figura 21: Algoritmo di controllo PFDL-MFAC

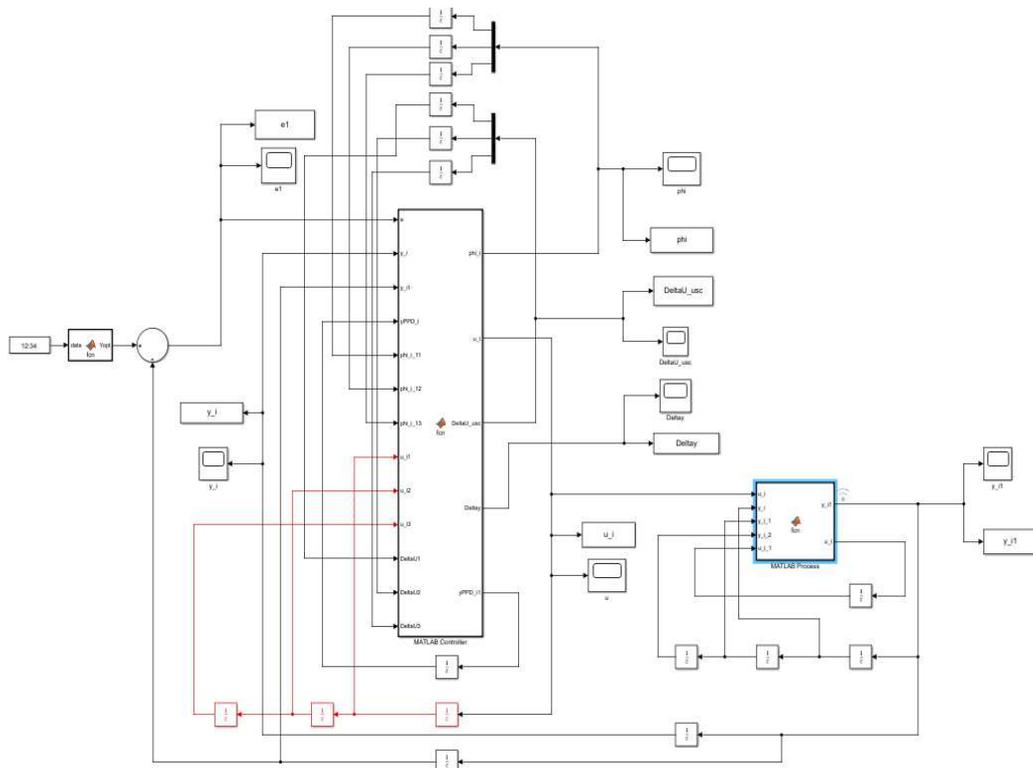


Figura 22: Schema PFDL-MFAC

Lo schema *Simulink*, come si può notare, è molto simile al precedente ma presenta come differenza fondamentale un maggior numero di blocchi *Unit Delay* Z^{-1} i quali

concatenati ci permettono di sfruttare passi di campionamento $k-2$, $k-3$ a seconda delle necessità.

6.2.1 Simulazione 3

Nonlinear system

$$y(k+1) = \frac{y(k)y(k-1)y(k-2)u(k-1)(y(k)-1) + (1+a(k))u(k)}{1 + y^2(k) + y^2(k-1) + y^2(k-2)}$$

```
function [y_i1,u_i] = fcn(u_i, y_i,y_i_1,y_i_2,u_i_1)

y_i1=(y_i*y_i_1*y_i_2*u_i_1*(y_i-1)+2)*u_i/(1+y_i^2+y_i_1^2+y_i_2^2);
```

Figura 23: Processo simulazione 3

```
function Yopt = fcn(data)
k=data;
Yopt=0.5*(-1).^round(k./50);

ro = 0.5;
ro1= 0.5;
ro2= 0.5;
epsistema = 10^-5;
etastima = 0.5;
phi11 = 1;
phi12 = 0;
phi13 = 0;
mistima = 1;
lambda = 0.01;
Tc = 0.1;
```

Figura 24: Ingresso di riferimento simulazione 3

Figura 25: Parametri simulazione 3

In figura 25 si riportano i parametri adottati, da notare la scelta del LLC pari a tre, necessario per sistemi più complessi come quelli analizzati nelle seguenti simulazioni. In figura 26 e figura 27 notiamo come l'errore mantenuto costantemente a zero abbia dei picchi, sebbene minori di uno, proprio in corrispondenza degli sbalzi dell'uscita la quale segue il segnale di ingresso, il quale dipendente dal tempo.

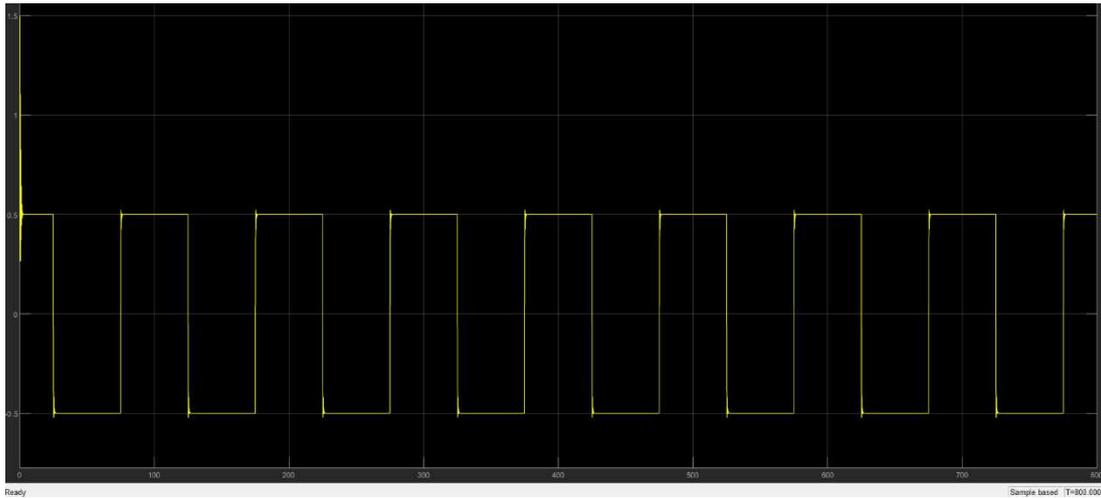


Figura 26: Uscita sistema simulazione 3

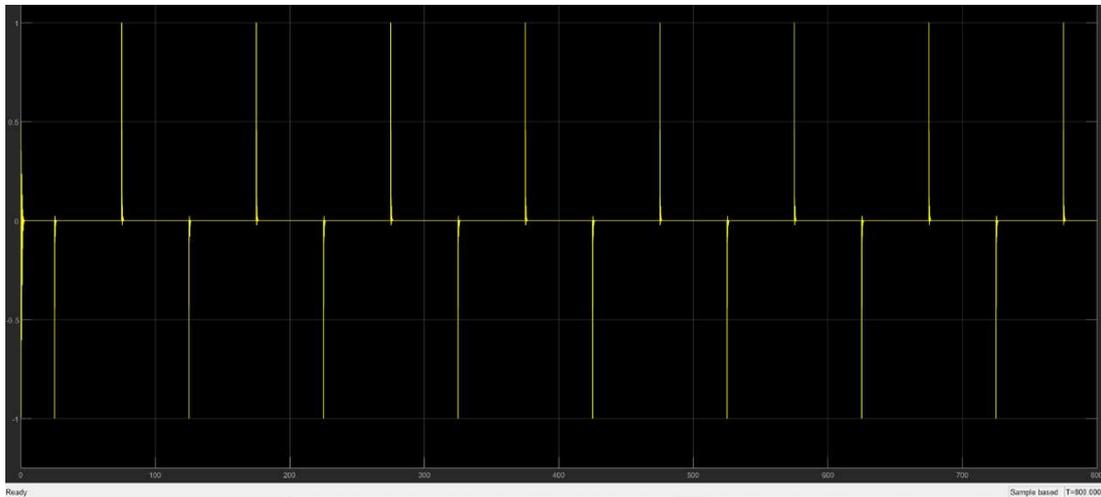


Figura 27: Errore tra uscita e riferimento simulazione 3

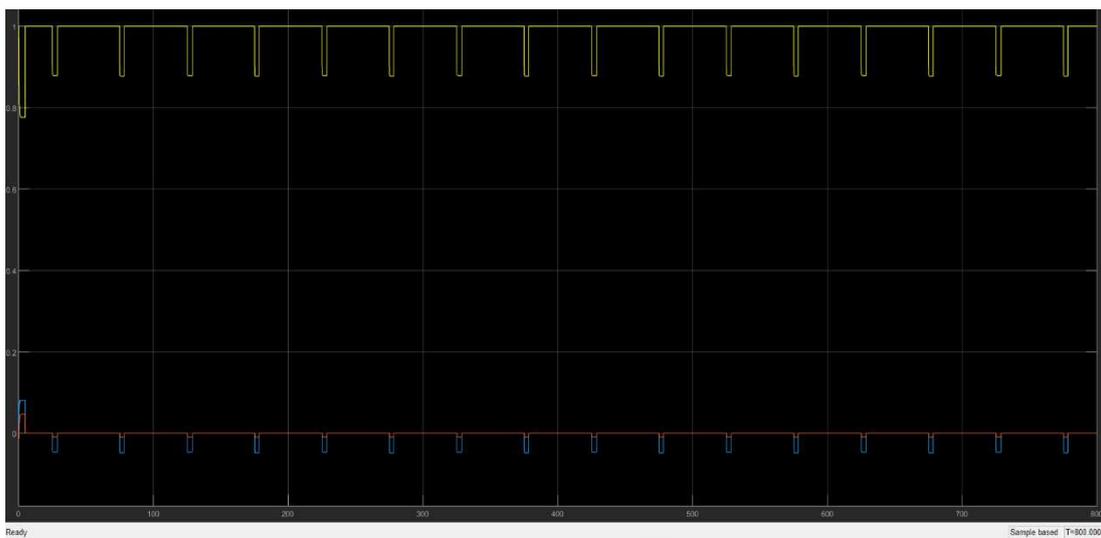


Figura 28: Stima di PG simulazione 3

6.2.2 Simulazione 4

Nella seconda simulazione proposta si è utilizzato il metodo PFDL-MFAC affianco a un classico controllore PID tarato con i seguenti valori $K_p = 0.15$, $K_d = 0$, $K_I = 0.5$. I parametri utilizzati anche in questa simulazione, nonostante il processo sia differente, sono quelli mostrati in *figura 25*.

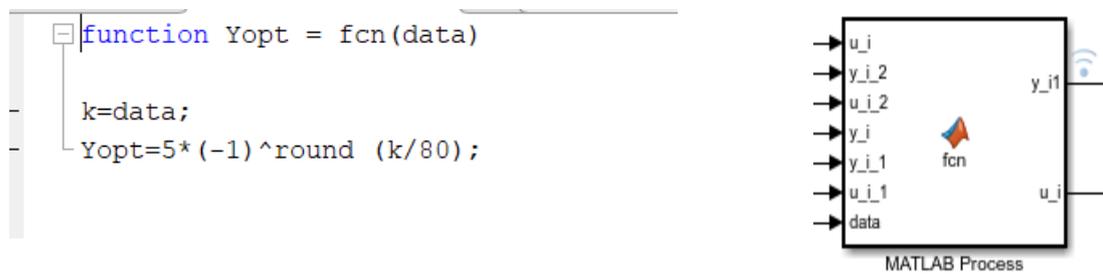


Figura 29: Ingresso di riferimento simulazione 4

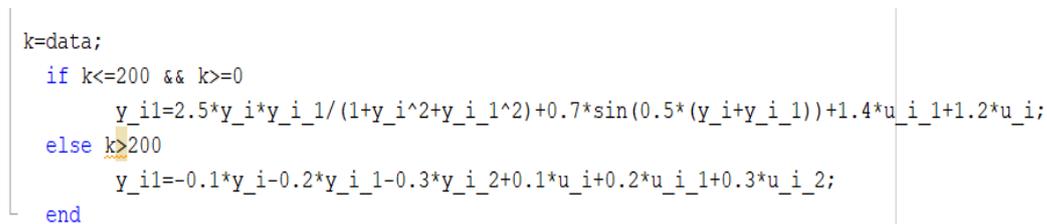


Figura 30: Processo simulazione 4

Nelle *figure 31* si mostra lo schema *Simulink* realizzato per confrontare le uscite date dalle due tipologie di controllori diverse.

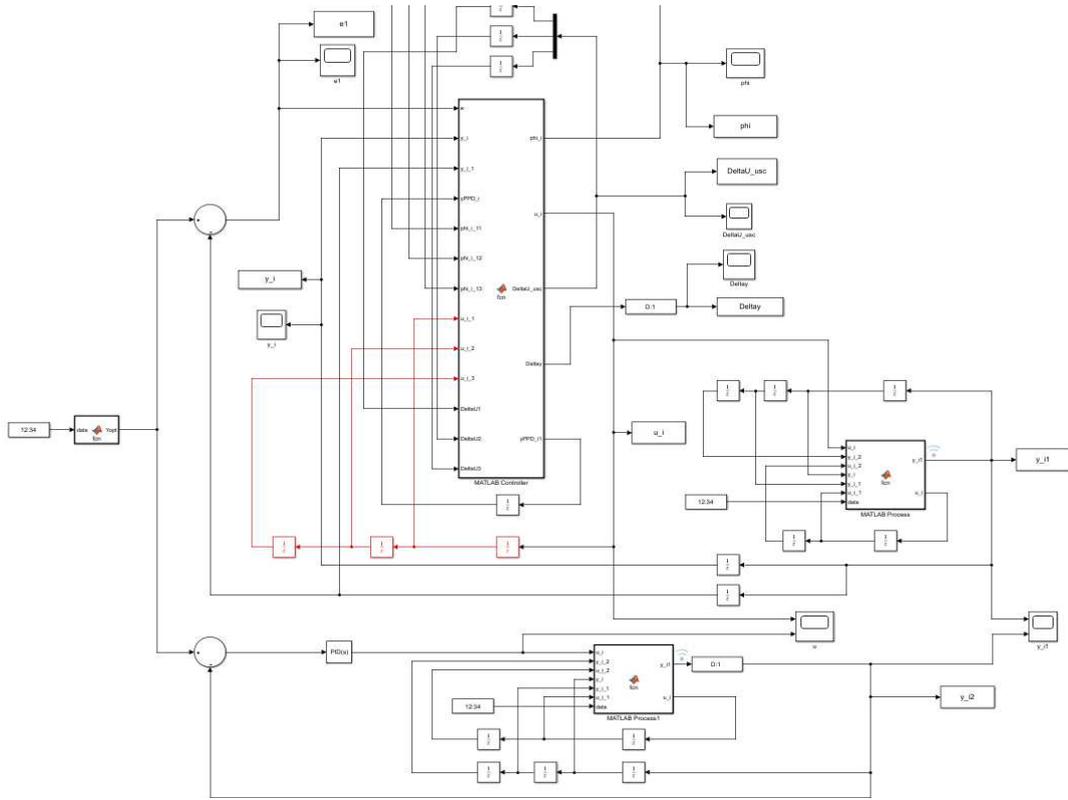


Figura 31: Schema Simulink simulazione 4

Dai risultati delle simulazioni, mostrati nella figura 32, figura 33 e figura 34, si nota come il PFDL-MFAC (giallo) ha un tempo di salita molto inferiore e riesce a definire una risposta adeguata in un tempo minore; lo stesso controllo ha un andamento più pronto e reattivo a quelle che sono le fluttuazioni dell'ingresso di riferimento su una dinamica che varia nel tempo.

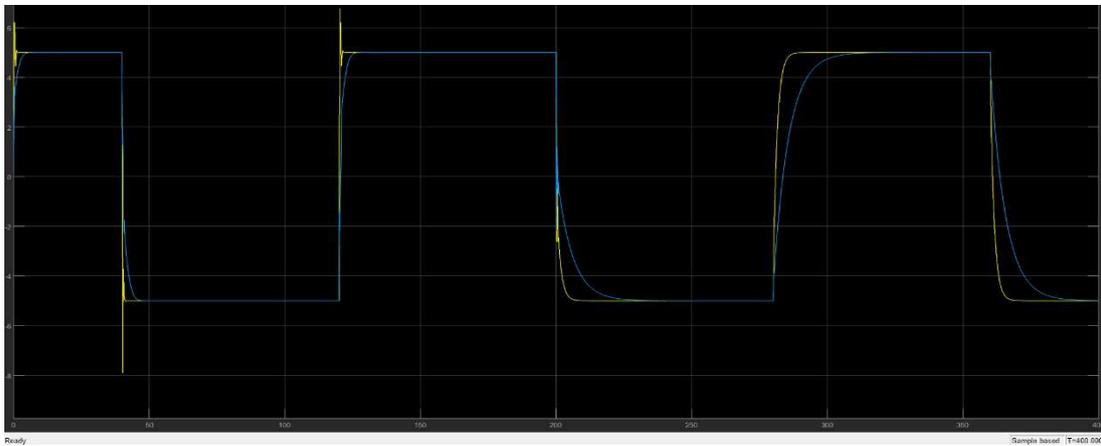


Figura 32: Confronto uscite sistema MFAC (giallo)- PID (blu) simulazione 4

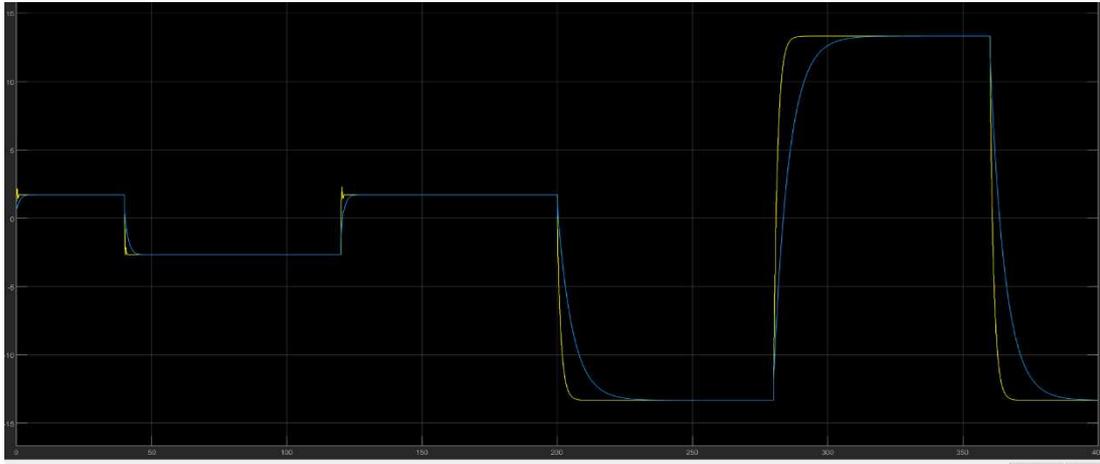


Figura 33: Confronto uscite controllore MFAC (giallo) - PID (blu) simulazione 4

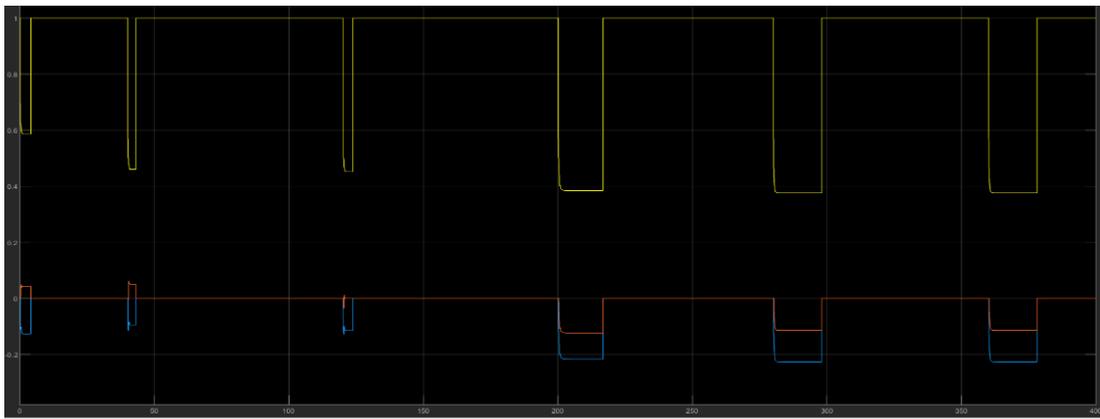


Figura 34: Stima di PG simulazione 4

Nonostante un processo che risulta essere variabile e un ingresso dipendente dal tempo, il metodo PFDL-MFAC riesce a garantire un'uscita uguale al riferimento, con un errore piccolo e di breve durata, sinonimo di un controllore che riesce ad adattarsi rapidamente al contrario del più semplicistico PID il quale, sebbene sia stato tarato al meglio dopo numerosi tentativi, non riesce a seguire il segnale di riferimento con la stessa rapidità.

6.2.3 Simulazione 5

In quest'ultima simulazione proposta si utilizzano entrambi i metodi proposti, CFDL e PFDL, su uno stesso processo per verificare le differenze che questi due metodi hanno su uscita, errore e uscita del controllore.

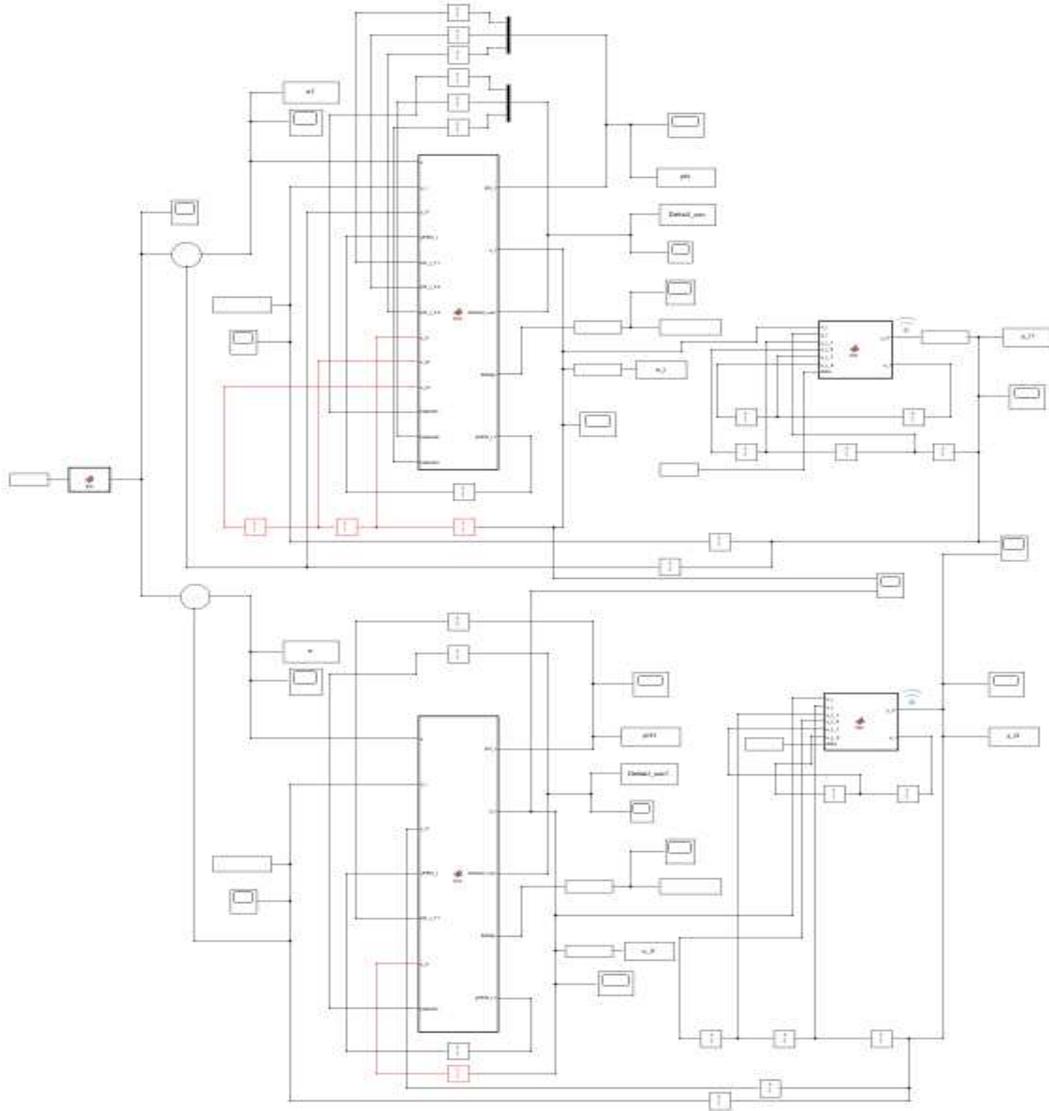


Figura 35: Schema Simulink simulazione 5

```

1 - ro = 0.5;
2 - ro1= 0.5;
3 - ro2= 0.5;
4 - epsistema = 10^-5;
5 - etastima = 0.5;
6 - phil1 = 1;
7 - phil2 = 0;
8 - phil3 = 0;
9 - mistima = 1;
10 - lambda =1;
11 - Tc=0.1;
12

```

Figura 36: Parametri simulazione 5

I parametri utilizzati sono gli stessi per entrambi i modelli di dati, semplicemente il CFDL non li utilizza tutti ma solo una parte.

```

1 - function Yopt = fcn(data)
2
3 - k=data;
4 - Yopt=5*sin(k/50)+2*cos(k/20);
5

```

Figura 37: Ingresso di riferimento simulazione 5

```

1 - function [y_i1,u_i] = fcn(u_i, y_i,y_i_1,y_i_2,u_i_1,u_i_2, data)
2
3 - k=data;
4 - if k>=0 && k<=400
5 - y_i1=((2.5*y_i*y_i_1)/(1+y_i^2+y_i_1^2))+1.2*u_i+0.09*u_i*u_i_1+1.6*u_i_2+0.7*sin(0.5*(y_i+y_i_1))*cos(0.5*(y_i+y_i_1));
6 - else k>400 && k<=800
7 - y_i1=((5*y_i*y_i_1)/(1+y_i^2+y_i_1^2+y_i_2^2))+u_i+1.1*u_i_1;
8 - end
9
10
11

```

Figura 38: Processo simulazione 5

Il processo da controllare non solo risulta tempo variante ma si presenta con una complessità maggiore rispetto a quelli analizzati nelle simulazioni precedenti. Con una dinamica del genere il metodo CFDL ha un comportamento peggiore rispetto al PFDL il quale riesce a tracciare meglio con l' algoritmo una migliore la stima.

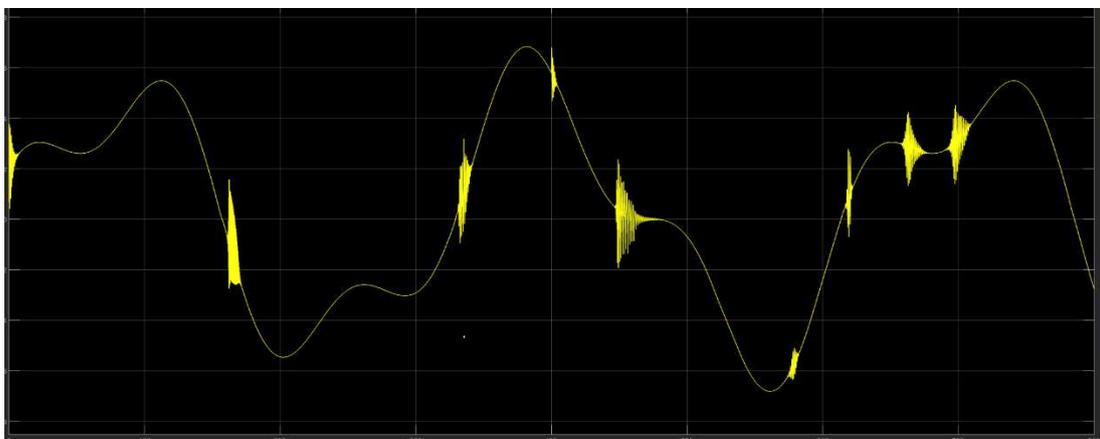


Figura 39: Uscita CFDL-MFAC simulazione 5

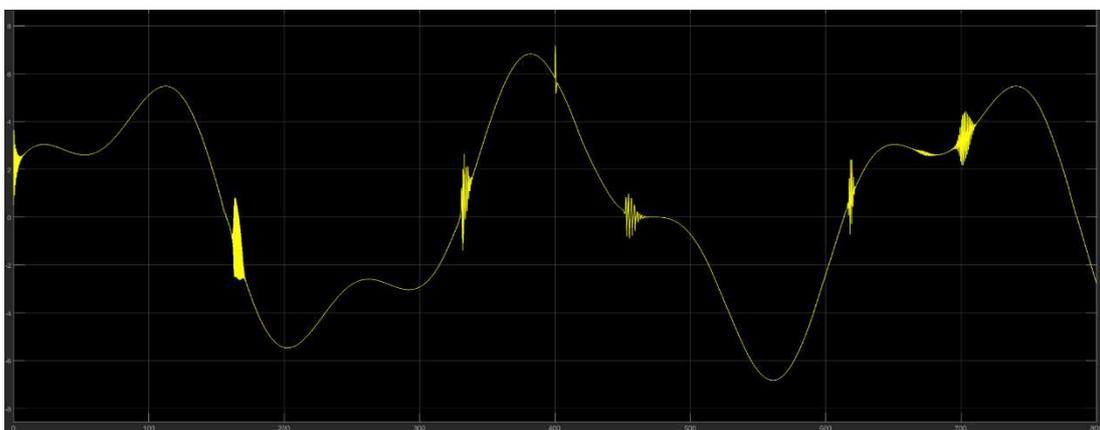


Figura 40: Uscita PFDL-MFAC simulazione 5

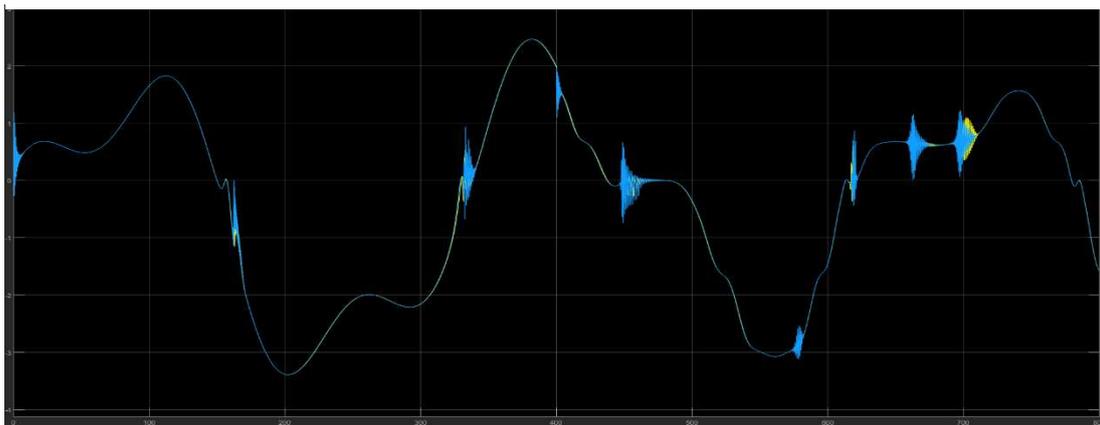


Figura 41: Confronto uscite controllori CFDL (blu) - PFDL (giallo) simulazione 5

Nella *figura 41* si può notare come nella sovrapposizione delle uscite dei controllori il segnale del PFDL sia completamente coperto da quello del CFDL, in particolare nei

momenti in cui sono presenti fluttuazioni, dovute a non linearità o al processo e all'ingresso che variano. Il metodo PFDL dunque si presenta in quei casi più pronto e efficace a riportare l'errore a zero. Proprio quanto appena detto viene mostrato in *figura 42* dove anche in questo caso, della sovrapposizione dell'errore, è visibile ancora una volta solo quello del CFDL in blu.

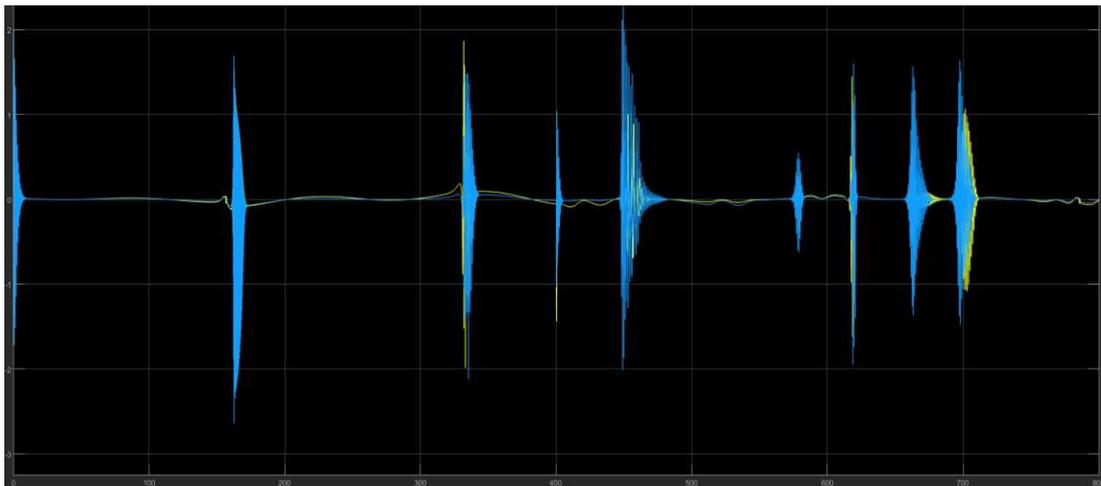


Figura 42: Confronto errore tra CFDL (blu) - PFDL (giallo) simulazione 5

7 Conclusioni

Sulla base dell'approccio di linearizzazione dinamica presentato nel *Capitolo 4*, sono stati presentati due tipi di schemi MFAC per un sistema SISO non lineare a tempo discreto. L'approccio MFAC proposto utilizza semplicemente dati I/O dell'impianto controllato e non sono coinvolte informazioni sul modello del processo. Oltre a risultare un metodo semplice da implementare si dimostra anche robusto in quanto riesce a affrontare in modo uniforme problemi di controllo per sistemi non lineari con variazioni nel tempo. Infatti, come si può vedere nelle simulazioni, in particolare del metodo PFDL, i parametri adottati sono gli stessi nonostante i processi controllati siano molto differenti. Infine, il compromesso tra il carico di calcolo e la semplicità del design del sistema di controllo su questi due metodi di controllo MFAC può essere ottenuto attraverso un'attenta selezione del metodo e il criterio vitale nella scelta del controller dipende sia dalle prestazioni del sistema di controllo sia dal costo che ci si potrebbe permettere per la realizzazione del controllore.

Lista delle variabili Simulink

Variabile	Simbolo	Variabile	Simbolo
$y^{*(k+1)}$	<i>Yopt</i>	$u(k+1)$	<i>u_i1</i>
$y(k+1)$	<i>y_i1</i>	$u(k)$	<i>u_i</i>
$y(k)$	<i>y_i</i>	$u(k-1)$	<i>u_i_1</i>
$y(k-1)$	<i>y_i_1</i>	$u(k-2)$	<i>u_i_2</i>
$y(k-2)$	<i>y_i_2</i>	$u(k-3)$	<i>u_i_3</i>
$\widehat{\varphi}_c(k)$	<i>phi_i</i>	ρ	<i>ro</i>
$\widehat{\varphi}_c(k-1)$	<i>phi_i_1</i>	ρ_1	<i>ro1</i>
$\widehat{\varphi}_c(k-2)$	<i>phi_i_2</i>	ρ_2	<i>ro2</i>
$\widehat{\varphi}_{p,1}(1)$	<i>phi11</i>	$\Delta u(k)$	<i>DeltaU</i>
$\widehat{\varphi}_{p,2}(1)$	<i>phi12</i>	$\Delta y(k)$	<i>Delta_y</i>
$\widehat{\varphi}_{p,3}(1)$	<i>phi13</i>	<i>Tempo</i>	<i>Tc</i>
η	<i>etastima</i>	λ	<i>lambda</i>
μ	<i>mistima</i>	ε	<i>epsistima</i>

Riferimenti bibliografici

- [1] Model Free Adaptive Control Zhongsheng Hou Shangtai Jin Theory and Applications
- [2] Data-Driven Adaptive Sliding Mode Control of Nonlinear Discrete-Time Systems With Drescribed Performance IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, VOL. 49, NO. 12, DECEMBER 2019
- [3] Simulazione 1 da: Data-Driven Adaptive Sliding Mode Control of Nonlinear Discrete-Time Systems With Drescribed Performance
- [4] Simulazione 2 da: esempio 4.2 pag. 88 Model Free Adaptive Control Zhongsheng Hou Shangtai Jin Theory and Applications
- [5] Simulazione 3 da: esempio 4.5 pag. 106 Model Free Adaptive Control Zhongsheng Hou Shangtai Jin Theory and Applications
- [6] Simulazione 4 da: esempio 4.4 pag. 104 Model Free Adaptive Control Zhongsheng Hou Shangtai Jin Theory and Applications
- [7] Simulazione 5 da: esempio 4.6 pag. 108 Model Free Adaptive Control Zhongsheng Hou Shangtai Jin Theory and Applications