Adaptive Feedback Linearization per sistemi non lineari in forma affine nell'input di controllo

Gianluca Rizzello, David Naso

Indice

Introduzione

In questo documento si tratterà la tecnica di Adaptive Feedback Linearization (AFL) per sistemi non lineari affini nell'ingresso di controllo. Tale tecnica verrà dapprima descritta in forma non adattativa, ovvero Feedback Linearization (FL), e successivamente ne verà introdotta la variante adattativa. Lo sviluppo della AFL verrà effettuato su tre classi di sistemi non lineari single-input single-output (SISO) tempo-continui in forma affine nell'ingresso di controllo, di complessità via via crescente, ovvero un sistema di ordine 1, un sistema di ordine n, ed un sistema di ordine n con grado relativo $r < n$.

Adaptive Feedback Linearization per sistemi non lineari di ordine 1 in forma affine

Si consideri il seguente sistema non lineare single-input single-output (SISO) del primo ordine in forma affine nell'input di controllo

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = x \end{cases}, \tag{1}$$

con

$$x \in \mathbb{R},\ u \in \mathbb{R},\ y \in \mathbb{R}, \tag{2}$$

$$f(x): \mathbb{R} \to \mathbb{R},\ g(x): \mathbb{R} \to \mathbb{R},\ g(x) \neq 0, \forall x \in \mathbb{R}. \tag{3}$$

Si assuma inoltre che le funzioni $f(x)$ e $g(x)$ siano smooth.

Si assuma che lo stato x sia misurabile. L'obiettivo é quello di progettare una legge di controllo $u(x)$ tale che l'uscita y insegua un riferimento $x^* = y^*$ desiderato. Si assuma, inoltre, che la derivata prima nel tempo di y^* esista e sia continua a tratti. Si definisce l'errore di tracking e come segue

$$e = y - y^* = x - x^*. \tag{4}$$

La legge di controllo u é scelta nella maniera seguente

$$u(x) = \frac{1}{g(x)}\left[-f(x) - \lambda e + \dot{x}^*\right], \tag{5}$$

dove $\lambda > 0$ é un parametro di progetto. Sostituendo (5) in (1), si ha la seguente dinamica in anello chiuso

$$\begin{cases} \dot{x} = -\lambda e + \dot{x}^* \\ y = x \end{cases}. \tag{6}$$

Utilizzando la (4), la (6) può essere riscritta in coordinate di errore come segue

$$\dot{e} = -\lambda e \,. \tag{7}$$

La dinamica dell'errore di tracking soddisferà quindi l'equazione (7), di conseguenza il suo andamento temporale sarà nella seguente forma

$$e(t) = e(0)\exp(-\lambda t) \tag{8}$$

e, se $\lambda > 0$, si avrà

$$\lim_{t\to\infty} e(t) = 0 \Rightarrow \lim_{t\to\infty} y(t) = y^*(t) \,. \tag{9}$$

La rapidità con cui y converge alla traiettoria desiderata y^* dipende dal valore di λ, che si comporta quindi come il polo che determina la convergenza a zero dell'errore.

Nel caso di riferimenti a gradino, é possibile annullare l'errore a regime in presenza di disturbi esterni mediante Feedback Linearization (FL) con azione integrale. Si definisce l'errore filtrato come

$$e_{PI}(t) = e(t) + k\int_0^t e(\tau)\,d\tau \,. \tag{10}$$

L'azione di controllo viene quindi ridefinita come segue

$$u(x) = \frac{1}{g(x)}\left[-f(x) - ke - \lambda e_{PI} + \dot{x}^*\right]. \tag{11}$$

Sostituendo la (11) nella (1), si ottiene

$$\dot{x} = -ke - \lambda e_{PI} + \dot{x}^* \,, \tag{12}$$

ovvero

$$\underbrace{\dot{x} - \dot{x}^*}_{\dot{e}} + ke + \lambda e_{PI} = 0 \,. \tag{13}$$

Infine, utilizzando la definizione in (10), la (13) può essere riscritta come segue

$$\dot{e}_{PI} + \lambda e_{PI} = 0 \,, \tag{14}$$

che assicura la convergenza a zero dell'errore filtrato e_{PI} se $\lambda > 0$.

In molti casi reali, le funzioni $f(x)$ e $g(x)$ non sono note con precisione, ma sono note solamente delle loro approssimazioni $\hat{f}(x)$ e $\hat{g}(x)$. Ovviamente, tanto meno accurate saranno tali approssimazioni, tanto meno precisa sarà la legge di controllo risultante. Uno dei possibili metodi per garantire buone prestazioni in assenza di conoscenza perfetta del modello consiste nell'apprendere le funzioni $f(x)$ e $g(x)$ durante il task di controllo, modificando gli approssimatori $\hat{f}(x)$ e $\hat{g}(x)$ in tempo reale in modo da massimizzare l'accuratezza dell'inseguimento della traiettoria desiderata. Questo paradigma rappresenta il principio su cui si basa la AFL.

Si supponga che le approssimazioni delle funzioni siano esprimibili come funzioni lineari nei parametri, ovvero

$$\hat{f}(x) = \varphi_f^{\,T}(x)\hat{\theta}_f \,, \tag{15}$$

$$\hat{g}(x) = \varphi_g^{\,T}(x)\hat{\theta}_g \,, \tag{16}$$

$$\varphi_f(x):\mathbb{R}\to\mathbb{R}^{n_f},\ \hat{\theta}_f\in\mathbb{R}^{n_f},\ \varphi_g(x):\mathbb{R}\to\mathbb{R}^{n_g},\ \hat{\theta}_g\in\mathbb{R}^{n_g}, \tag{17}$$

dove $\varphi_f(x)$ e $\varphi_g(x)$ sono regressori noti, mentre $\hat{\theta}_f$ e $\hat{\theta}_g$ sono parametri da stimare online. I numeri di regressori/parametri n_f e n_g sono parametri liberi di progetto. Tali funzioni possono rappresentare diversi tipi di approssimatori lineari nei parametri, come polinomi, reti neuronali a base radiale, ecc... Si assuma inoltre che, rispetto alla classe di modelli rappresentata da (15)-(16), le funzioni reali $f(x)$ e $g(x)$ ammettano le seguenti rappresentazioni

$$f(x)=\varphi_f^T(x)\theta_f^*+\varepsilon_f, \tag{18}$$

$$g(x)=\varphi_g^T(x)\theta_g^*+\varepsilon_g, \tag{19}$$

dove $\varphi_f^T(x)\theta_f^*$ e $\varphi_g^T(x)\theta_g^*$ rappresentano le migliori approssimazioni possibili di $f(x)$ e $g(x)$ nella famiglia di modelli considerata, in corrispondenza dei valori ottimali e costanti θ_f^* e θ_g^*, mentre ε_f e ε_g rappresentano gli errori residui dovuti al fatto che il sistema reale non appartiene esattamente alla classe di modelli selezionata. Al variare della classe dei modelli, ci aspettiamo che anche ε_f e ε_g varino di conseguenza, e in teoria i loro valori tendono a ridursi all'aumentare della varietà di modelli contenuti nella classe scelta. In generale, più risulta essere vasta la classe di modelli descritta dagli approssimatori (15)-(16), più ci aspettiamo che gli errori residui ε_f e ε_g siano piccoli.

Gli errori di approssimazione sono definiti come segue

$$\tilde{f}(x)=f(x)-\hat{f}(x)=\varphi_f^T(x)\tilde{\theta}_f+\varepsilon_f, \tag{20}$$

$$\tilde{g}(x)=g(x)-\hat{g}(x)=\varphi_g^T(x)\tilde{\theta}_g+\varepsilon_g, \tag{21}$$

con

$$\tilde{\theta}_f=\theta_f^*-\hat{\theta}_f, \tag{22}$$

$$\tilde{\theta}_g=\theta_g^*-\hat{\theta}_g. \tag{23}$$

Idealmente, la massima precisione possibile con gli approssimatori (15)-(16) si ha quando

$$\begin{cases}\tilde{\theta}_f=0\\ \tilde{\theta}_g=0\end{cases}\Rightarrow\begin{cases}\tilde{f}(x)=\varepsilon_f\\ \tilde{g}(x)=\varepsilon_g\end{cases}, \tag{24}$$

e quindi ε_f e ε_g rappresentano gli errori di approssimazione nell'ipotesi più favorevole. L'obiettivo della AFL é quello di adattare i parametri $\hat{\theta}_f$ e $\hat{\theta}_g$ in tempo reale in modo tale che la condizione (24) sia soddisfatta, facendo sì che la legge di controllo basata sulle approssimazioni (15)-(16) fornisca prestazioni quanto più simili possibili alla FL standard basata in caso di conoscenza ideale del modello.

La legge di controllo AFL si basa sulla seguente equazione

$$u(x)=\frac{1}{\hat{g}(x)}\left[-\hat{f}(x)-\lambda e+\dot{x}^*\right]. \tag{25}$$

con $\lambda > 0$ parametro di progetto. Si consideri nuovamente la (1), e si sommi e sottragga alla prima equazione la quantità $\hat{g}(x)u$

$$\begin{cases} \dot{x} = f(x) + g(x)u + \hat{g}(x)u - \hat{g}(x)u \\ y = x \end{cases}. \tag{26}$$

Sostituendo (25) nella (26) si ottiene

$$\begin{cases} \dot{x} = f(x) + g(x)u - \hat{f}(x) - \lambda e + \dot{x}^* - \hat{g}(x)u \\ y = x \end{cases}, \tag{27}$$

Che, usando le (20)-(21), può essere riscritta come segue

$$\begin{cases} \dot{x} = -\lambda e + \dot{x}^* + \tilde{f}(x) + \tilde{g}(x)u \\ y = x \end{cases}, \tag{28}$$

ovvero

$$\begin{cases} \dot{x} = -\lambda e + \dot{x}^* + \varphi_f^T(x)\tilde{\theta}_f + \varphi_g^T(x)\tilde{\theta}_g u + \varepsilon_f + \varepsilon_g u \\ y = x \end{cases}. \tag{29}$$

Definendo l'errore di tracking secondo la (4), la (29) diviene

$$\dot{e} = -\lambda e + \varphi_f^T(x)\tilde{\theta}_f + \varphi_g^T(x)\tilde{\theta}_g u + \varepsilon_f + \varepsilon_g u\,. \tag{30}$$

Si noti che, diversamente dalla (7) in cui il modello era noto con esattezza, in (30) si ha un sistema non autonomo. Di conseguenza, l'errore non evolve secondo la (8), e non si hanno garanzie di convergenza di quest'ultimo a zero, a causa del termine che dipende dall'errore di approssimazione. Si noti, tuttavia, che nel caso di approssimazione perfetta (30) diviene equivalente a (7), e di conseguenza il controllo diviene ideale.

A questo punto, occorre trovare delle leggi di adattamento per $\hat{\theta}_f$ e $\hat{\theta}_f$ che garantiscano che l'errore di tracking e sia quanto più piccolo possibile. A tal fine, si definisca la seguente funzione candidata di Lyapunov

$$V = \frac{1}{2}e^2 + \frac{1}{2\gamma_f}\tilde{\theta}_f^T\tilde{\theta}_f + \frac{1}{2\gamma_g}\tilde{\theta}_g^T\tilde{\theta}_g\,, \tag{31}$$

dove γ_f e γ_g sono parametri di progetto positivi. Si noti che $V\left(e,\tilde{\theta}_f,\tilde{\theta}_g\right)$ risulta essere definita positiva in (0,0,0). Occorre quindi progettare la legge di controllo u in modo che $\dot{V}\left(e,\tilde{\theta}_f,\tilde{\theta}_g\right)$ sia definita negativa in (0,0,0), assicurando quindi convergenza a zero sia dell'errore di tracking che dei parametri. La derivata di (31) vale

$$\dot{V} = e\dot{e} + \frac{1}{\gamma_f}\tilde{\theta}_f^T\dot{\tilde{\theta}}_f + \frac{1}{\gamma_g}\tilde{\theta}_g^T\dot{\tilde{\theta}}_g\,. \tag{32}$$

Tenendo conto delle (22)-(23), e considerando che i parametri ottimi θ_f^* e θ_g^* sono costanti, si ha

$$\dot{\tilde{\theta}}_f = -\dot{\hat{\theta}}_f\,, \tag{33}$$

$$\dot{\tilde{\theta}}_g = -\dot{\hat{\theta}}_g \,. \tag{34}$$

Sostituendo (30), (33)-(34) in (32) si ottiene

$$\dot{V} = -\lambda e^2 + e\varphi_f{}^T(x)\tilde{\theta}_f + e\varphi_g{}^T(x)\tilde{\theta}_g u + e\varepsilon_f + e\varepsilon_g u - \frac{1}{\gamma_f}\tilde{\theta}_f{}^T\dot{\hat{\theta}}_f - \frac{1}{\gamma_g}\tilde{\theta}_g{}^T\dot{\hat{\theta}}_g \,. \tag{35}$$

Riscriviamo quindi la (35) come segue

$$\dot{V} = -\lambda e^2 + \tilde{\theta}_f{}^T\left(\varphi_f(x)e - \frac{1}{\gamma_f}\dot{\hat{\theta}}_f\right) + \tilde{\theta}_g{}^T\left(\varphi_g(x)eu - \frac{1}{\gamma_g}\dot{\hat{\theta}}_g\right) + e\left(\varepsilon_f + \varepsilon_g u\right). \tag{36}$$

Definendo le leggi di adattamento dei parametri come segue

$$\dot{\hat{\theta}}_f = \gamma_f \varphi_f(x)e \,, \tag{37}$$

$$\dot{\hat{\theta}}_g = \gamma_g \varphi_g(x)eu \,, \tag{38}$$

é possibile riscrivere la (36) come

$$\dot{V} = -\lambda e^2 + e\left(\varepsilon_f + \varepsilon_g u\right). \tag{39}$$

La funzione (39) risulta essere semidefinita negativa rispetto ad e, se si può suporre di trascurare il termine $e(\varepsilon_f+\varepsilon_g u)$ (ovvero in caso di errori di stima ε_f e ε_g piccoli). Si noti che, anche in caso in cui gli errori di stima siano nulli, la (39) non risulterà mai definita negativa, poiché non dipende da $\tilde{\theta}_f$ e $\tilde{\theta}_f$. Di conseguenza, anche in caso di stimatori perfetti, l'AFL garantisce solamente che l'errore di tracking e vada a zero, ma non assicura che le stime ottime dei parametri convergano ai parametri reali del sistema. É tuttavia ragionevole aspettarsi che anche i parametri convergano ai loro valori reali, se viene soddisfatta l'ipotesi di persistente eccitazione. Inoltre, la (39) non convergerà mai a zero, ma tenderà a convergere in una regione nell'intorno dell'origine, tanto più piccola quanto più piccoli sono gli errori ε_f e ε_g, e quindi tanto più piccola quanto più sono accurati gli approssimatori utilizzati.

In definitiva, la legge di controllo AFL, ottenuta collezionando le equazioni (15), (16), (25) , (37), (38) é data da

$$\begin{cases} \dot{\hat{\theta}}_f = \gamma_f \varphi_f(x)e \\ \dot{\hat{\theta}}_g = \gamma_g \varphi_g(x)eu \\ u(x) = \dfrac{1}{\varphi_g{}^T(x)\hat{\theta}_g}\left[-\varphi_f{}^T(x)\hat{\theta}_f - \lambda e + \dot{x}^*\right] \end{cases} . \tag{40}$$

Le costanti γ_f e γ_g sono dette learning rates. Aumentare i valori dei learning rates aumentà la rapidità di adattamento dei parametri, favorendo quindi la convergenza a zero dell'errore. Valori troppo grandi dei learning rates, tuttavia, possono causare fenomeni indesiderati quali oscillazioni, instabilità e amplificazione del rumore di misura. Tali parametri vanno pertanto tarati a mano tenendo conto del compromesso tra questi effetti conflittuali. Si noti anche che entrambe le leggi di apprendimento date da (37) e (38) sono proporzionali all'errore di tracking

e, pertanto l'addestramento viene interrotto nel momento in cui il tracking risulti essere perfetto (e = 0).

La tecnica appena discussa é chiamata AFL indiretta, in quanto l'azione di controllo é determinata dalle stime di $f(x)$ e $g(x)$. Se $g(x)$ é una costante, si può applicare la AFL diretta, che si basa sulla parametrizzazione dell'ingresso di controllo come segue

$$u = \varphi_u^T(x)\tilde{\theta}_u \tag{41}$$

e sulla base di tale ingresso di controllo si ricavano $f(x)$ e $g(x)$. Tale tecnica, tuttavia, non sarà discussa in questa sede.

Uno dei problemi della legge di controllo (25) é causato dalla presenza di $\hat{g}(x)$ al denominatore. Se i parametri vengono adattati nel tempo in modo da rendere $\hat{g}(x)$ nullo, l'azione di controllo tenderebbe all'infinito, causando problemi di saturazione dell'attuatore. Una possibile soluzione consiste nel rendere gli elementi di $\varphi_g(x)$ e $\hat{\theta}_g$ tutti positivi o tutti negativi. Un problema analogo si verifica nel momento in cui anche $\hat{f}(x)$ tende all'infinito, ovvero quando i $\hat{\theta}_f$ tendono a crescere senza limite. Al fine di evitare ciò, é importante far sì che $\varphi_f(x)$ e $\hat{\theta}_f$ siano sempre limitati. Un metodo per assicurarsi che i coefficienti $\hat{\theta}_f$ e $\hat{\theta}_g$ non crescano o decrescano oltre una soglia di tolleranza prestabilita consiste nell'impiego di algoritmi di proiezione. Un possibile esempio di algoritmo di proiezione é mostrato qui di seguito

$$\begin{cases} if\ \ \hat{\theta}_f \in \left[\underline{\theta}_f,\ \overline{\theta}_f\right] \vee \left(\hat{\theta}_f = \underline{\theta}_f \wedge \dot{\hat{\theta}}_f > 0\right) \vee \left(\hat{\theta}_f = \overline{\theta}_f \wedge \dot{\hat{\theta}}_f < 0\right) & \rightarrow\ \dot{\hat{\theta}}_f = \gamma_f \varphi_f(x) e \\ else & \rightarrow\ \dot{\hat{\theta}}_f = 0 \end{cases}, \tag{42}$$

$$\begin{cases} if\ \ \hat{\theta}_g \in \left[\underline{\theta}_g,\ \overline{\theta}_g\right] \vee \left(\hat{\theta}_g = \underline{\theta}_g \wedge \dot{\hat{\theta}}_g > 0\right) \vee \left(\hat{\theta}_g = \overline{\theta}_g \wedge \dot{\hat{\theta}}_g < 0\right) & \rightarrow\ \dot{\hat{\theta}}_g = \gamma_g \varphi_g(x) e u \\ else & \rightarrow\ \dot{\hat{\theta}}_g = 0 \end{cases}, \tag{43}$$

che assicura che i parametri $\hat{\theta}_f$ e $\hat{\theta}_g$ rimangano vincolati negli intervalli scelti dal progettista.

Adaptive Feedback Linearization per sistemi non lineari di ordine *n* in forma affine

La tecnica precedentemente descritta permette di controllare sistemi non lineari di ordine 1 la cui dinamica non é nota con certezza, mediante apprendimento delle funzioni del sistema in tempo reale. In questa sezione, l'AFL verrà estesa a sistemi non lineari di ordine generico pari ad n.

Si consideri il seguente sistema non lineare single-input single-output (SISO) di ordine n in forma affine nell'input di controllo

$$\begin{cases} \dot{x}_1 = x_2 \\ \vdots \\ \dot{x}_{n-1} = x_n \\ \dot{x}_n = f(x) + g(x)u \\ y = x_1 \end{cases}, \tag{44}$$

con

$$x \in \mathbb{R}^n,\ u \in \mathbb{R},\ y \in \mathbb{R}, \tag{45}$$

$$f(x): \mathbb{R}^n \to \mathbb{R},\ g(x): \mathbb{R}^n \to \mathbb{R},\ g(x) \neq 0, \forall x \in \mathbb{R}^n. \tag{46}$$

Si assuma inoltre che le funzioni $f(x)$ e $g(x)$ siano smooth. Si noti inoltre che $f(x)$ e $g(x)$ sono funzioni scalari dell'ingresso vettoriale x. La classe di sistemi (44) generalizza la classica forma canonica di controllo per sistemi lineari, in quanto l'uscita y coincide con la prima variabile di stato x_1, e tutte le variabili di stato successive coincidono con le derivate temporali successive dell'uscita fino alla (n-1)-esima, ovvero

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix} \tag{47}$$

Si assuma che lo stato x sia completamente misurabile. L'obiettivo é quello di progettare una legge di controllo $u(x)$ tale che l'uscita y insegua un riferimento $x_1^* = y^*$ desiderato. Si assuma, inoltre, che la derivata n-esima nel tempo di y^* esista e sia continua a tratti. Si definisce l'errore di tracking sullo stato e come segue

$$e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} - \begin{bmatrix} x_1^* \\ x_2^* \\ \vdots \\ x_n^* \end{bmatrix} = x - x^*, \tag{48}$$

dove lo stato di riferimento x^* é dato, in accordo con la (47), da

$$\begin{bmatrix} x_1^* \\ x_2^* \\ \vdots \\ x_n^* \end{bmatrix} = \begin{bmatrix} y^* \\ \dot{y}^* \\ \vdots \\ y^{(n-1)*} \end{bmatrix}. \tag{49}$$

La u é scelta nella maniera seguente

$$u(x) = \frac{1}{g(x)}\left[-f(x) - Ke + \dot{x}_n^*\right], \tag{50}$$

dove K é un vettore riga n-dimensionale di guadagni da progettare

$$K = \begin{bmatrix} k_1 & \cdots & k_n \end{bmatrix}. \tag{51}$$

Sostituendo (50) in (44) si ha la seguente dinamica in anello chiuso

$$\begin{cases} \dot{x}_1 = x_2 \\ \vdots \\ \dot{x}_{n-1} = x_n \\ \dot{x}_n = -Ke + \dot{x}_n^{\,*} \\ y = x_1 \end{cases} . \tag{52}$$

Utilizzando la (48), la (52) può essere convertita in coordinate di errore come segue

$$\begin{cases} \dot{e}_1 = e_2 \\ \vdots \\ \dot{e}_{n-1} = e_n \\ \dot{e}_n = -Ke \end{cases} , \tag{53}$$

ovvero, in forma compatta,

$$\dot{e} = \Lambda e , \tag{54}$$

con

$$\Lambda = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_n \end{bmatrix} . \tag{55}$$

La dinamica dell'errore di tracking soddisferà l'equazione (54), e quindi il suo andamento temporale sarà dato da

$$e(t) = \exp(\Lambda t) e(0) . \tag{56}$$

L'evoluzione di $e(t)$ è quindi dettata dagli autovalori della matrice (55), che possono essere assegnati liberamente scegliendo in maniera opportuna i guadagni $k_1,...,k_n$. Essendo la matrice (55) in forma canonica di controllo, tali guadagni possono essere calcolati immediatamente, poiché il suo polinomio caratteristico é dato da

$$s^n + k_n s^{n-1} + \cdots + k_2 s + k_1 = 0 . \tag{57}$$

Se i guadagni vengono scelti in modo che il polinomio (57) abbia tutte le radici a parte reale negativa, allora

$$\lim_{t \to \infty} \left\| e(t) \right\| = 0 \Rightarrow \lim_{t \to \infty} y(t) = y^*(t) . \tag{58}$$

La rapidità con cui y converge alla traiettoria desiderata y^* dipende dalla radice di (57) avente la parte reale più piccola in valore assoluto, che funge quindi da polo dominante della risposta.

Nel caso in cui le funzioni $f(x)$ e $g(x)$ non siano note con precisione, é possibile rappresentarle con degli approssimatori lineari nei parametri $\hat{f}(x)$ e $\hat{g}(x)$ nella seguente forma

$$\hat{f}(x) = \varphi_f^{\,T}(x) \hat{\theta}_f , \tag{59}$$

$$\hat{g}(x) = \varphi_g^T(x)\hat{\theta}_g\,, \tag{60}$$

$$\varphi_f(x): \mathbb{R}^n \to \mathbb{R}^{n_f}\,,\ \hat{\theta}_f \in \mathbb{R}^{n_f}\,,\ \varphi_g(x): \mathbb{R}^n \to \mathbb{R}^{n_g}\,,\ \hat{\theta}_g \in \mathbb{R}^{n_g}\,, \tag{61}$$

dove $\varphi_f(x)$ e $\varphi_g(x)$ sono regressori noti, mentre $\hat{\theta}_f$ e $\hat{\theta}_f$ sono parametri da stimare online. I numeri di regressori/parametri n_f e n_g sono parametri liberi di progetto. Si assuma inoltre che, rispetto alla classe di modelli rappresentata da (59)-(60), le funzioni reali $f(x)$ e $g(x)$ ammettano le seguenti rappresentazioni

$$f(x) = \varphi_f^T(x)\theta_f^* + \varepsilon_f\,, \tag{62}$$

$$g(x) = \varphi_g^T(x)\theta_g^* + \varepsilon_g\,, \tag{63}$$

dove $\varphi_f^T(x)\theta_f^*$ e $\varphi_g^T(x)\theta_g^*$ costituiscono le migliori approssimazioni possibili di $f(x)$ e $g(x)$ nella famiglia di modelli considerata, in corrispondenza dei valori ottimali θ_f^* e θ_g^*, mentre ε_f e ε_g rappresentano gli errori residui dovuti al fatto che il sistema reale non appartiene alla classe di modelli selezionata.

Gli errori di approssimazione sono definiti, ancora una volta, come segue

$$\tilde{f}(x) = f(x) - \hat{f}(x) = \varphi_f^T(x)\tilde{\theta}_f + \varepsilon_f\,, \tag{64}$$

$$\tilde{g}(x) = g(x) - \hat{g}(x) = \varphi_g^T(x)\tilde{\theta}_g + \varepsilon_g\,, \tag{65}$$

con

$$\tilde{\theta}_f = \theta_f^* - \hat{\theta}_f\,, \tag{66}$$

$$\tilde{\theta}_g = \theta_g^* - \hat{\theta}_g\,. \tag{67}$$

A questo punto risulta necessario definire delle leggi di adattamento per $\hat{\theta}_f$ e $\hat{\theta}_g$ tali da rendere l'errore di tracking asintoticamente nullo.

La legge di controllo AFL per sistemi di ordine n é data dalla seguente

$$u(x) = \frac{1}{\hat{g}(x)}\left[-\hat{f}(x) - Ke + \dot{x}_n^*\right]. \tag{68}$$

con K vettore di guadagni da progettare. Si consideri nuovamente la (44), e si sommi e sottragga all'ultima equazione di stato la quantità $\hat{g}(x)u$

$$\begin{cases} \dot{x}_1 = x_2 \\ \vdots \\ \dot{x}_{n-1} = x_n \\ \dot{x}_n = f(x) + g(x)u + \hat{g}(x)u - \hat{g}(x)u \\ y = x_1 \end{cases}. \tag{69}$$

Sostituendo (68) nella (69), ed effettuando delle manipolazioni matematiche in maniera analoga al caso precedente, si ottiene

$$\begin{cases} \dot{x}_1 = x_2 \\ \vdots \\ \dot{x}_{n-1} = x_n \\ \dot{x}_n = -Ke + \tilde{f}(x) + \tilde{g}(x)u + \dot{x}_n^{\,*} \\ y = x_1 \end{cases}, \tag{70}$$

ovvero

$$\begin{cases} \dot{x}_1 = x_2 \\ \vdots \\ \dot{x}_{n-1} = x_n \\ \dot{x}_n = -Ke + \dot{x}_n^{\,*} + \varphi_f^{\,T}(x)\tilde{\theta}_f + \varphi_g^{\,T}(x)\tilde{\theta}_g u + \varepsilon_f + \varepsilon_g u \\ y = x_1 \end{cases}. \tag{71}$$

Convertendo la (71) in coordinate di errore secondo la (48) si ottiene

$$\begin{cases} \dot{e}_1 = e_2 \\ \vdots \\ \dot{e}_{n-1} = e_n \\ \dot{e}_n = -Ke + \varphi_f^{\,T}(x)\tilde{\theta}_f + \varphi_g^{\,T}(x)\tilde{\theta}_g u + \varepsilon_f + \varepsilon_g u \end{cases}. \tag{72}$$

È possibile riscrivere la (72) in forma compatta come segue

$$\dot{e} = \Lambda e + B\delta\,, \tag{73}$$

con

$$\Lambda = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_n \end{bmatrix}, \tag{74}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \tag{75}$$

$$\delta = \varphi_f^{\,T}(x)\tilde{\theta}_f + \varphi_g^{\,T}(x)\tilde{\theta}_g u + \varepsilon_f + \varepsilon_g u\,. \tag{76}$$

Diversamente dalla (54), la (76) presenta anche il termine esogeno δ che rende il sistema errore non autonomo. Occorre quindi trovare delle leggi di adattamento per $\hat{\theta}_f$ e $\hat{\theta}_f$ che garantiscano che l'errore di tracking e sia quanto più piccolo possibile.

A tal fine, si definisca la seguente funzione candidata di Lyapunov

$$V = \frac{1}{2} e^T P e + \frac{1}{2} \tilde{\theta}_f^{\,T} \Gamma_f^{-1} \tilde{\theta}_f + \frac{1}{2} \tilde{\theta}_g^{\,T} \Gamma_g^{-1} \tilde{\theta}_g \,, \tag{77}$$

dove P é una matrice definita positiva, mentre Γ_f e Γ_g sono matrici diagonali definite positive, da definire opportunamente. Si noti che $V\left(e, \tilde{\theta}_f, \tilde{\theta}_g\right)$ risulta essere definita positiva in (0,0,0). Occorre quindi progettare la legge di controllo u in modo che $\dot{V}\left(e, \tilde{\theta}_f, \tilde{\theta}_g\right)$ sia definita negativa in (0,0,0). La derivata di (77) vale

$$V = \frac{1}{2} \dot{e}^T P e + \frac{1}{2} e^T P \dot{e} + \tilde{\theta}_f^{\,T} \Gamma_f^{-1} \dot{\tilde{\theta}}_f + \tilde{\theta}_g^{\,T} \Gamma_g^{-1} \dot{\tilde{\theta}}_g \,. \tag{78}$$

Tenendo conto delle (66)-(67), e considerando che i parametri ottimi θ_f^* e θ_g^* sono costanti, si ha

$$\dot{\tilde{\theta}}_f = -\dot{\hat{\theta}}_f \,, \tag{79}$$

$$\dot{\tilde{\theta}}_g = -\dot{\hat{\theta}}_g \,. \tag{80}$$

Sostituendo (73), (76) (79), (80) in (78) si ottiene

$$\dot{V} = \frac{1}{2} e^T \left(P\Lambda + \Lambda^T P\right) e + \left(\tilde{\theta}_f^{\,T} \varphi_f(x) + \tilde{\theta}_g^{\,T} \varphi_g(x) u + \varepsilon_f + \varepsilon_g u\right) B^T P e - \tilde{\theta}_f^{\,T} \Gamma_f^{-1} \dot{\hat{\theta}}_f - \tilde{\theta}_g^{\,T} \Gamma_g^{-1} \dot{\hat{\theta}}_g \,. \tag{81}$$

Se la matrice Λ é tale da avere tutti gli autovalori a parte reale negativa, allora é possibile definire la patrice P in modo tale che

$$P\Lambda + \Lambda^T P = -Q \,, \tag{82}$$

con Q definita positiva (e quindi $-Q$ definita negativa). Una volta scelta una Q arbitraria e definita positiva, é possibile risalire alla matrice P della funzione di Lyapunov dalla risoluzione dell’equazione di Lyapunov (82). Tenendo conto della (82), é possibile riscrivere la (81) come segue

$$\dot{V} = -\frac{1}{2} e^T Q e + \tilde{\theta}_f^{\,T} \left(\varphi_f(x) B^T P e - \Gamma_f^{-1} \dot{\hat{\theta}}_f\right) + \tilde{\theta}_g^{\,T} \left(\tilde{\theta}_g^{\,T} \varphi_g(x) B^T P e u - \Gamma_g^{-1} \dot{\hat{\theta}}_g\right) + B^T P e \left(\varepsilon_f + \varepsilon_g u\right). \tag{83}$$

Definendo le leggi di adattamento dei parametri come segue

$$\dot{\hat{\theta}}_f = \Gamma_f \varphi_f(x) B^T P e \,, \tag{84}$$

$$\dot{\hat{\theta}}_g = \Gamma_g \varphi_g(x) B^T P e u \,, \tag{85}$$

é possibile riscrivere la (83) come

$$\dot{V} = -\frac{1}{2} e^T Q e + B^T P e \left(\varepsilon_f + \varepsilon_g u\right). \tag{86}$$

Ancora una volta, la funzione (86) risulta essere semidefinita negativa rispetto ad e, se si può suporre di trascurare il termine dipendente da ε_f e ε_g. Di conseguenza, le leggi di adattamento (84)-(85) garantiscono la convergenza a zero dell’errore di tracking. Il fatto che la (86) risulta, in ipotesi di poter trascurare ε_f e ε_g, solo semidefinita negativa, implica che gli errori di stima

$\tilde{\theta}_f$ e $\tilde{\theta}_g$ non convergano necessariamente a zero. Ancora una volta, é ragionevole aspettarsi che i parametri convergano ai loro valori reali, se viene soddisfatta l'ipotesi di persistente eccitazione. Se i termini ε_f e ε_g sono piccoli ma non nulli, la (86) non tenderà esattamente a zero, bensì ad una regione nell'intorno dell'origine, tanto più piccola quanto più piccoli sono gli errori ε_f e ε_g, e quindi tanto più piccola quanto più sono accurati gli approssimatori utilizzati.

In definitiva, la legge di controllo AFL per sistemi di ordine n, ottenuta collezionando le equazioni (68), (62),(63), (84), (85) é data da

$$\begin{cases} \dot{\hat{\theta}}_f = \Gamma_f \varphi_f(x) B^T Pe \\ \dot{\hat{\theta}}_g = \Gamma_g \tilde{\theta}_g^T \varphi_g(x) B^T Peu \\ u(x) = \dfrac{1}{\varphi_g^T(x)\hat{\theta}_g}\left[-\varphi_f^T(x)\hat{\theta}_f - Ke + \dot{x}_n^*\right] \end{cases}, \tag{87}$$

dove le componenti di K sono scelte in modo da rendere la (74) asintoticamente stabile, P é ottenuta risolvendo la (82) per una Q arbitraria definita positiva e per Λ ottenuta in seguito alla scelta di K, mentre le matrici diagonali Γ_f e Γ_g definiscono i learning rates, che nel caso generico possono essere settati in maniera diversa componente per componente, in modo da rendere la stima di alcuni parametri più veloce rispetto ad altri. Tutte le altre considerazioni già discusse per il caso di sistemi di ordine 1 continuano a valere anche nel caso generale con sistemi di ordine n.

Adaptive Feedback Linearization per sistemi non lineari di ordine *n* e grado relativo *r* < *n* in forma affine

Per concludere, si considera il caso di controllo AFL di sistemi non lineari di ordine n e grado relativo $r < n$. In particolare, si consideri la seguente classe di sistemi non lineari single-input single-output (SISO) di ordine n e grado relativo $r < n$ in forma affine nell'input di controllo, detta anche forma normale

$$\begin{cases} \dot{\xi}_1 = \xi_2 \\ \vdots \\ \dot{\xi}_{r-1} = \xi_r \\ \dot{\xi}_r = a(\xi,\eta) + b(\xi,\eta)u \\ \dot{\eta}_1 = s_1(\xi,\eta) \\ \vdots \\ \dot{\eta}_{n-r} = s_{n-r}(\xi,\eta) \\ y = \xi_1 \end{cases}, \tag{88}$$

con

$$x = \begin{bmatrix} \xi \\ \eta \end{bmatrix}, \tag{89}$$

$$x \in \mathbb{R}^n,\ \xi \in \mathbb{R}^r,\ \eta \in \mathbb{R}^{n-r},\ u \in \mathbb{R},\ y \in \mathbb{R}, \tag{90}$$

$$a(\xi,\eta): \mathbb{R}^r \times \mathbb{R}^{n-r} \to \mathbb{R},\ b(\xi,\eta): \mathbb{R}^r \times \mathbb{R}^{n-r} \to \mathbb{R},\ b(\xi,\eta) \neq 0, \forall (\xi,\eta) \in \mathbb{R}^r \times \mathbb{R}^{n-r}, \tag{91}$$

$$s_i(\xi,\eta): \mathbb{R}^r \times \mathbb{R}^{n-r} \to \mathbb{R},\ i = 1, \ldots, n-r. \tag{92}$$

Si assuma che valgano ipotesi di continuità e regolarità per le funzioni $a(\xi,\eta)$, $b(\xi,\eta)$, $s_i(\xi,\eta)$, $i = 1,...,n-r$. La classe di sistemi (88) permette di generalizzare il concetto di zeri di sistema, ben noti per sistemi lineari, al caso non lineare. L'uscita y coincide con la prima variabile di stato ξ_1, e tutte le variabili di stato successive coincidono con le derivate temporali successive dell'uscita fino alla (r-1)-esima, dove r é il grado relativo del sistema,

$$\xi = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_r \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(r-1)} \end{bmatrix}. \tag{93}$$

Le rimanenti n-r variabili di stato, invece, descrivono la dinamica interna del sistema. Tale dinamica valuta in corrispondenza di $\xi = 0$, cioé

$$\begin{cases} \dot{\eta}_1 = s_1(0,\eta) \\ \vdots \\ \dot{\eta}_{n-r} = s_{n-r}(0,\eta) \end{cases} \tag{94}$$

é detta dinamica zero, in quanto per sistemi lineari essa coincide con un sistema autonomo avente come autovalori gli zeri del sistema.

Si assuma che lo stato $x = [\xi^T\ \eta^T]^T$ sia completamente misurabile. L'obiettivo é quello di progettare una legge di controllo $u(\xi,\eta)$ tale che l'uscita y insegua un riferimento $\xi_1^* = y^*$ desiderato. Si assume, inoltre, che la derivata nel tempo r-esima di y^* esista e sia continua a tratti. Si definisce l'errore di tracking sulle prime r componenti dello stato e come segue

$$e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_r \end{bmatrix} = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_r \end{bmatrix} - \begin{bmatrix} \xi_1^* \\ \xi_2^* \\ \vdots \\ \xi_r^* \end{bmatrix} = \xi - \xi^*, \tag{95}$$

dove lo stato di riferimento ξ^* é dato, in accordo con la (93), da

$$\begin{bmatrix} \xi_1^* \\ \xi_2^* \\ \vdots \\ \xi_r^* \end{bmatrix} = \begin{bmatrix} y^* \\ \dot{y}^* \\ \vdots \\ y^{(r-1)*} \end{bmatrix} \tag{96}$$

La u é scelta nella maniera seguente

$$u(\xi,\eta)=\frac{1}{b(\xi,\eta)}\left[-a(\xi,\eta)-Ke+\dot{\xi}_r^{\,*}\right], \tag{97}$$

dove K é un vettore riga r-dimensionale di guadagni da progettare,

$$K=\left[k_1 \quad \cdots \quad k_r\right]. \tag{98}$$

Sostituendo (97) in (88) si ha la seguente dinamica in anello chiuso

$$\begin{cases} \dot{\xi}_1=\xi_2 \\ \vdots \\ \dot{\xi}_{r-1}=\xi_r \\ \dot{\xi}_r=-Ke+\dot{\xi}_r^{\,*} \\ \dot{\eta}_1=s_1(\xi,\eta) \\ \vdots \\ \dot{\eta}_{n-r}=s_{n-r}(\xi,\eta) \\ y=\xi_1 \end{cases}. \tag{99}$$

Utilizzando la (95), la (99) può essere riscritta come segue

$$\begin{cases} \dot{e}_1=e_2 \\ \vdots \\ \dot{e}_{n-1}=e_n \\ \dot{e}_n=-Ke \\ \dot{\eta}_1=s_1\left(\xi^*+e,\eta\right) \\ \vdots \\ \dot{\eta}_{n-r}=s_{n-r}\left(\xi^*+e,\eta\right) \end{cases}. \tag{100}$$

Si noti come la legge di controllo (97) permetta di disaccoppiare la dinamica zero dalla dinamica dell’errore di tracking, realizzando di fatto l’equivalente non lineare di una cancellazione polo-zero. Questo ovviamente può essere fatto con sicurezza solo nel momento in cui la dinamica zero (94) sia stabile. La dinamica dell’errore di tracking é quindi data, in forma compatta, da

$$\dot{e}=\Lambda e, \tag{101}$$

con

$$\Lambda=\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_r \end{bmatrix}. \tag{102}$$

Se i guadagni guadagni $k_1,\ldots,k_r$ sono scelti in modo da rendere tutti gli autovalori della (102) asintoticamente stabili, allora l’errore di tracking converge asintoticamente a zero. La dinamica zero, di contro, non converge necessariamente a zero. Tuttavia, se essa risulta stabile, allora si ha la certezza che essa rimanga comunque limitata.

Nel caso in cui le funzioni $a(\xi,\eta)$ e $b(\xi,\eta)$ non siano note con precisione é possibile rappresentarle con degli approssimatori lineari nei parametri $\hat{a}(\xi,\eta)$ e $\hat{b}(\xi,\eta)$, nella seguente forma

$$\hat{a}(\xi,\eta) = \varphi_a^T(\xi,\eta)\hat{\theta}_a, \tag{103}$$

$$\hat{b}(\xi,\eta) = \varphi_b^T(\xi,\eta)\hat{\theta}_b, \tag{104}$$

$$\varphi_a(\xi,\eta): \mathbb{R}^r \times \mathbb{R}^{n-r} \to \mathbb{R}^{n_a},\ \hat{\theta}_a \in \mathbb{R}^{n_a},\ \varphi_b(\xi,\eta): \mathbb{R}^r \times \mathbb{R}^{n-r} \to \mathbb{R}^{n_b},\ \hat{\theta}_b \in \mathbb{R}^{n_b}, \tag{105}$$

dove $\varphi_a(\xi,\eta)$ e $\varphi_b(\xi,\eta)$ sono regressori noti, mentre $\hat{\theta}_a$ e $\hat{\theta}_b$ sono parametri da stimare online. I numeri di regressori/parametri n_a e n_b sono parametri liberi di progetto. Si assuma inoltre che, rispetto alla classe di modelli rappresentata da (103)-(104), le funzioni reali $a(\xi,\eta)$ e $b(\xi,\eta)$ ammettano le seguenti rappresentazioni

$$a(\xi,\eta) = \varphi_a^T(\xi,\eta)\theta_a^* + \varepsilon_a, \tag{106}$$

$$b(\xi,\eta) = \varphi_b^T(x)\theta_b^* + \varepsilon_b. \tag{107}$$

Il significato dei simboli é analogo ai casi considerati in precedenza. Gli errori di approssimazione sono definiti, ancora una volta, come segue

$$\tilde{a}(\xi,\eta) = a(\xi,\eta) - \hat{a}(\xi,\eta) = \varphi_a^T(\xi,\eta)\tilde{\theta}_a + \varepsilon_a, \tag{108}$$

$$\tilde{b}(\xi,\eta) = b(\xi,\eta) - \hat{b}(\xi,\eta) = \varphi_b^T(\xi,\eta)\tilde{\theta}_b + \varepsilon_b, \tag{109}$$

con

$$\tilde{\theta}_a = \theta_a^* - \hat{\theta}_a, \tag{110}$$

$$\tilde{\theta}_b = \theta_b^* - \hat{\theta}_b. \tag{111}$$

A questo punto risulta necessario definire delle leggi di adattamento per $\hat{\theta}_a$ e $\hat{\theta}_b$ tali da rendere asintoticamente nullo l'errore di tracking.

La legge di controllo AFL é definita come segue

$$u(\xi,\eta) = \frac{1}{\hat{b}(\xi,\eta)}\left[-\hat{a}(\xi,\eta) - Ke + \dot{\xi}_r^*\right], \tag{112}$$

con K vettore di guadagni da progettare. Sostituendo la (112) in (88), usando le (108),(109) ed effettuando una manipolazione analoga al caso precedente, si ottiene

$$
\begin{cases}
\dot{\xi}_1 = \xi_2 \\
\vdots \\
\dot{\xi}_{r-1} = \xi_r \\
\dot{\xi}_r = -Ke + \dot{\xi}_r^{\,*} + \varphi_a^{\,T}\left(\xi,\eta\right)\tilde{\theta}_a + \varphi_b^{\,T}\left(\xi,\eta\right)\tilde{\theta}_b u + \varepsilon_a + \varepsilon_b u \\
\dot{\eta}_1 = s_1\left(\xi,\eta\right) \\
\vdots \\
\dot{\eta}_{n-r} = s_{n-r}\left(\xi,\eta\right) \\
y = \xi_1
\end{cases}
. \tag{113}
$$

Definendo l'errore di tracking analogamente a (95), la (113) diviene

$$
\begin{cases}
\dot{e}_1 = e_2 \\
\vdots \\
\dot{e}_{r-1} = e_r \\
\dot{e}_r = -Ke + \varphi_a^{\,T}\left(\xi,\eta\right)\tilde{\theta}_a + \varphi_b^{\,T}\left(\xi,\eta\right)\tilde{\theta}_b u + \varepsilon_a + \varepsilon_b u \\
\dot{\eta}_1 = s_1\left(\xi^* + e,\eta\right) \\
\vdots \\
\dot{\eta}_{n-r} = s_{n-r}\left(\xi^* + e,\eta\right)
\end{cases}
. \tag{114}
$$

Possiamo riscrivere le prime r equazioni della (114) in forma compatta come segue

$$
\dot{e} = \Lambda e + B\delta \,, \tag{115}
$$

con

$$
\Lambda = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_r \end{bmatrix}, \tag{116}
$$

$$
B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \tag{117}
$$

$$
\delta = \varphi_a^{\,T}\left(\xi,\eta\right)\tilde{\theta}_a + \varphi_b^{\,T}\left(\xi,\eta\right)\tilde{\theta}_b u + \varepsilon_a + \varepsilon_b u \,. \tag{118}
$$

Si noti come le equazioni (115)-(118) siano del tutto analoghe alle (73)-(76) ottenute nel caso di sistemi senza dinamica zero, eccezion fatta per l'ordine del sistema errore, che risulta pari ad r invece che n. Di conseguenza, é possibile applicare una trattazione pressoché analoga al caso precedente al fine trovare quindi delle leggi di adattamento per $\hat{\theta}_a$ e $\hat{\theta}_b$ che garantiscano che l'errore di tracking e sia quanto più piccolo possibile. Il comportamento degli stati η non é rilevante al fine del tracking dell'uscita, in quanto inosservabile rispetto all'uscita. É tuttavia necessario che tali stati rimangano limitati, poiché da tali valori dipende il calcolo della legge di controllo (112), che non deve crescere a dismisura in modo da evitare problemi di saturazione degll'attuatore. Tale condizione é comunque verificata in caso di sistemi a fase minima.

Si definisce la seguente funzione candidata di Lyapunov

$$V = \frac{1}{2} e^T P e + \frac{1}{2} \tilde{\theta}_a^T \Gamma_a^{-1} \tilde{\theta}_a + \frac{1}{2} \tilde{\theta}_b^T \Gamma_b^{-1} \tilde{\theta}_b , \tag{119}$$

dove P é una matrice definita positiva, mentre Γ_f e Γ_g sono matrici diagonali definite positive, da definire opportunamente. Si noti che $V\left(e, \tilde{\theta}_a, \tilde{\theta}_b\right)$ risulta essere definita positiva in (0,0,0). Tenendo conto delle (110)-(111), e considerando che i parametri ottimi θ_a^* e θ_b^* sono costanti, si ha

$$\dot{\tilde{\theta}}_a = -\dot{\hat{\theta}}_a , \tag{120}$$

$$\dot{\tilde{\theta}}_b = -\dot{\hat{\theta}}_b . \tag{121}$$

Derivando la (119) nel tempo e sostituendo le equazioni (115), (118), (120), (121) si ottiene

$$\dot{V} = \frac{1}{2} e^T \left(P\Lambda + \Lambda^T P \right) e + \left(\tilde{\theta}_a^T \varphi_a(\xi,\eta) + \tilde{\theta}_b^T \varphi_b(\xi,\eta) u + \varepsilon_a + \varepsilon_b u \right) B^T P e - \tilde{\theta}_a^T \Gamma_a^{-1} \dot{\hat{\theta}}_a - \tilde{\theta}_b^T \Gamma_b^{-1} \dot{\hat{\theta}}_b . \tag{122}$$

Se la matrice Λ é tale da avere tutti gli autovalori a parte reale negativa, allora é sempre possibile definire la patrice P in modo tale che

$$P\Lambda + \Lambda^T P = -Q , \tag{123}$$

con Q definita positiva (e quindi $-Q$ definita negativa). Una volta scelta una Q arbitraria e definita positiva, é possibile risalire alla matrice P della funzione di Lyapunov dalla risoluzione dell'equazione di Lyapunov (82). Definendo le leggi di adattamento dei parametri come segue

$$\dot{\hat{\theta}}_a = \Gamma_a \varphi_a(\xi,\eta) B^T P e , \tag{124}$$

$$\dot{\hat{\theta}}_b = \Gamma_b \tilde{\theta}_b^T \varphi_b(\xi,\eta) B^T P e u , \tag{125}$$

é possibile riscrivere la (122) come segue

$$\dot{V} = -\frac{1}{2} e^T Q e + B^T P e \left(\varepsilon_a + \varepsilon_b u \right). \tag{126}$$

Ancora una volta, la funzione (126) risulta essere semidefinita negativa rispetto ad e, se si può suporre di trascurare il termine dipendente da ε_f e ε_g.

In definitiva, la legge di controllo AFL per sistemi di ordine n con grado relativo $r < n$, ottenuta collezionando le equazioni (103), (104), (112), (124), (125) é data da

$$
\begin{cases}
\dot{\hat{\theta}}_a = \Gamma_a \varphi_a(\xi,\eta) B^T Pe \\
\dot{\hat{\theta}}_b = \Gamma_b \tilde{\theta}_b^{\,T} \varphi_b(\xi,\eta) B^T Peu \\
u(\xi,\eta) = \dfrac{1}{\varphi_b^{\,T}(\xi,\eta)\hat{\theta}_b}\left[-\varphi_a^{\,T}(\xi,\eta)\hat{\theta}_a - Ke + \dot{\xi}_n^{\,*}\right]
\end{cases}
\quad , \tag{127}
$$

dove le componenti di K sono scelte in modo da rendere la (74) asintoticamente stabile, P é ottenuta risolvendo la (123) per una Q arbitraria definita positiva e per Λ ottenuta in seguito alla scelta di K, mentre le matrici diagonali Γ_a e Γ_b definiscono i learning rates. Tutte le considerazioni già discusse nel caso precedente continuano a valere anche qui.