

POLITECNICO DI TORINO

**Corso di Laurea
in Ingegneria Aerospaziale**

Tesi di Laurea

**Modelli di Ordine Ridotto con
applicazione al controllo di sistemi
fluidodinamici.**



Relatrice:
Prof.ssa Stefania Scarsoglio

Candidato:
Enrico Foglia

ANNO ACCADEMICO 2020/2021

Indice

1	Introduzione	2
2	Richiami di teoria del controllo	4
2.1	Teoria del controllo ad anello chiuso	4
2.1.1	Controllo ottimo	7
2.1.2	Controllo predittivo basato su modello	9
3	Modelli di ordine ridotto	11
4	Proper Orthogonal Decomposition	14
4.1	POD	14
4.2	BPOD	16
4.2.1	Balanced Truncation	17
4.2.2	BPOD	17
4.2.3	Output Projection	19
5	Dynamic Mode Decomposition	20
5.1	DMD	20
5.1.1	Analisi di Koopman	22
5.2	DMDc	24
5.2.1	B nota	24
5.2.2	B non nota	25
6	Applicazioni	27
6.1	Flusso intorno a una piastra piana	27
6.1.1	BPOD per sistemi instabili	28
6.1.2	Risultati	30
6.2	Ali a geometria variabile	33
6.2.1	Algebraic DMD with control e modelli ridotti parametrici	34
6.2.2	Risultati	35
7	Conclusioni	39

Capitolo 1

Introduzione

L'evoluzione temporale di un fluido, finché è valida l'ipotesi del continuo, è totalmente caratterizzata dalle equazioni di Navier-Stokes, un insieme di tre equazioni differenziali alle derivate parziali che generalizzano le equazioni di Euler nel caso di fluidi compressibili e viscosi. Il modello, per quanto non si sappia ancora se rappresenti un problema ben posto in tre dimensioni [7], è in grado di dare soluzioni di grande accuratezza, e viene applicato in svariati ambiti della scienza e dell'ingegneria. È, insomma, un modello di grande successo, preciso e studiato in maniera molto profonda, ma non privo di limitazioni. Quella che si analizzerà in questo breve lavoro, deriva dall'altissima dimensionalità delle soluzioni di interesse pratico, che limita fortemente l'uso delle simulazioni più precise in ambiti dove la rapidità di calcolo è un fattore determinante, come il controllo o l'ottimizzazione.

Supponendo che la soluzione studiata sia $\mathbf{x} \in \mathbb{R}^n$, non è raro che n sia un numero molto grande, ovvero che \mathbf{x} sia un vettore con milioni di elementi. Solo calcolare l'evoluzione di tale stato può richiedere tempi di calcolo lunghissimi anche alle infrastrutture informatiche più evolute, usare modelli così complessi in algoritmi di ottimizzazione o per un controllatore in tempo reale è semplicemente fuori dalla portata anche del calcolatore più potente.

La soluzione che verrà qui studiata prevede di proiettare la dinamica del sistema in uno spazio di dimensione $r \ll n$, abbastanza piccola da comportare una drastica riduzione dei tempi di calcolo. Questa non è un'idea nuova, giacché la fisica matematica già da secoli sfrutta funzioni speciali per approssimare la soluzione di equazioni differenziali complesse. Lo sviluppo importante di queste tecniche negli ultimi anni deriva dalla sempre maggiore disponibilità di dati, da esperimenti o simulazioni, che possono aiutare a individuare tali funzioni in modo che siano ottimali per rappresentare lo specifico problema in esame. Lo studio di tali tecniche basate sui dati è lo scopo di questo lavoro.

Le sezioni sono organizzate nella maniera seguente: vengono innanzitutto dati alcuni richiami della teoria del controllo ad anello chiuso, in quanto

questa rappresenta la principale applicazione studiata in seguito per i modelli di ordine ridotto (capitolo 2.1); nel capitolo 3 viene data un'idea generale del funzionamento dei modelli di ordine ridotto e delle proiezioni di Galerkin e Petrov-Galerkin; i capitoli 4.1 e 5.1 studiano in maniera più approfondita alcuni algoritmi per ottenere delle basi per il sottospazio di ordine ridotto a partire dall'analisi dati. In particolare il capitolo 4.1 tratta la Proper Orthogonal Decomposition (POD) e la sua variante Balanced POD (sezione 4.2), mentre il capitolo 5.1 analizza la Dynamic Mode Decomposition (DMD), la sua variante applicata a sistemi controllati DMD with control (sezione 5.2) e la correlazione che si può tracciare con l'analisi di Koopman (5.1.1); nel capitolo 6 vengono analizzate due applicazioni per controllori in anello chiuso basati su varianti specializzate della BPOD (6.1) e della DMDc (6.2).

Capitolo 2

Richiami di teoria del controllo

2.1 Teoria del controllo ad anello chiuso

Definiamo un sistema dinamico che prende un vettore \mathbf{u} come input e restituisce una risposta \mathbf{y} nella seguente maniera:

$$\frac{d}{dt}\mathbf{x} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \quad (2.1)$$

$$\mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u}) \quad (2.2)$$

Dove \mathbf{u} è generalmente legato agli attuatori e la funzione \mathbf{g} dipende dal genere di sensori posti nel sistema. Schematicamente il sistema, detto ad anello aperto, è rappresentabile come in figura (2.1)

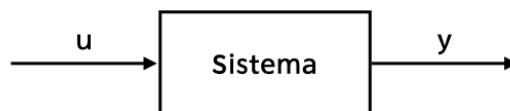


Figura 2.1: Schema a blocchi di un sistema ad anello aperto

La teoria del controllo moderna si basa il più delle volte su modelli lineari del sistema da controllare, di modo che le equazioni precedenti possono essere

scritte come:

$$\frac{d}{dt}\mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (2.3)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (2.4)$$

Delle quattro matrici, è la \mathbf{A} quella che rappresenta il sistema nel suo stato non controllato e ne definisce la stabilità. È possibile dimostrare che il sistema è stabile (asintoticamente) se e solo se tutti gli autovalori di \mathbf{A} hanno parte reale negativa [9]. Cercare di controllare un sistema in anello aperto richiederebbe una conoscenza quasi perfetta della dinamica, il che è praticamente impossibile. I controllori progettati per lavorare in anello aperto soffrono di una generale mancanza di robustezza, ovvero non sono in genere in grado di adattarsi ad errori nel modello o degli stessi attuatori [4].

La soluzione solitamente adottata per superare questa difficoltà consiste nell'utilizzo delle informazioni dell'output del sistema per controllare gli attuatori. Il sistema in generale può essere rappresentato come in figura (2.2).

Nello schema il segnale r rappresenta una referenza, per esempio uno stato che

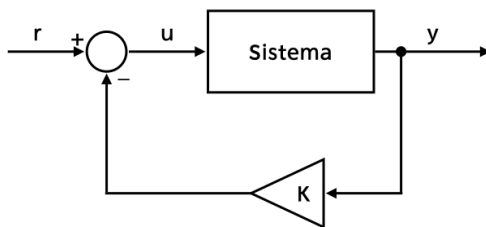


Figura 2.2: Schema a blocchi di un sistema controllato in anello chiuso.

si vorrebbe il sistema seguisse nella sua evoluzione, e la matrice \mathbf{K} permette di regolare la retroazione per ottenere il comportamento desiderato degli attuatori.

Supponendo, per il momento, di essere in grado di conoscere interamente lo stato del sistema, ovvero che $\mathbf{y} = \mathbf{x}$, l'equazione (2.3) diventerebbe:

$$\frac{d}{dt}\mathbf{x} = (\mathbf{A} - \mathbf{BK})\mathbf{x} + \mathbf{B}r \quad (2.5)$$

Dalla (2.5) si può evincere che, tramite una scelta accurata della matrice \mathbf{K} si possa sostanzialmente modificare la stabilità del sistema, tramite gli autovalori della matrice $\mathbf{A} - \mathbf{BK}$. Esistono tuttavia sistemi che resistono a questo tipo di approccio (detti *non controllabili*).

Un sistema si dice *controllabile* se è possibile virarne lo stato verso l'origine in un tempo finito, attraverso l'uso di una serie di azioni degli attuatori che siano continue a tratti. Più rigorosamente, uno stato è controllabile se dato un tempo

iniziale t_0 esiste un tempo finito t_f e una legge di controllo \mathbf{u} continua a tratti definita sull'intervallo $[t_0, t_f]$ tali che:

$$\mathbf{x}(t_f) = e^{\mathbf{A}(t_f-t_0)}\mathbf{x} + \int_{t_0}^{t_f} e^{\mathbf{A}(t_f-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau = 0 \quad (2.6)$$

Un sistema è controllabile se ogni suo stato è controllabile. È possibile determinare la controllabilità di un sistema a partire dalle matrici \mathbf{A} e \mathbf{B} definendo il cosiddetto gramiano di controllabilità:

$$\mathbf{W}_c := \int_0^\infty e^{\mathbf{A}\tau}\mathbf{B}\mathbf{B}^T e^{\mathbf{A}^T\tau}d\tau \quad (2.7)$$

Un sistema è controllabile se e solo se il gramiano \mathbf{W}_c (che è una matrice delle stesse dimensioni dello stato \mathbf{x}), è non singolare.

Alternativamente, per modelli lineari tempo invarianti, è possibile dare un criterio di controllabilità sulla base della cosiddetta matrice di controllabilità, definita come [3]:

$$\mathbf{C} = [\mathbf{B} \quad \mathbf{A}\mathbf{B} \quad \mathbf{A}^2\mathbf{B} \quad \dots \quad \mathbf{A}^{n-1}\mathbf{B}] \quad (2.8)$$

Un sistema è controllabile se la matrice \mathbf{C} ha rango massimo, ovvero $\text{rank}(\mathbf{C}) = n$.

Conoscere nella sua interezza lo stato di un sistema è un'impresa spesso impossibile, in quanto si è limitati dalla natura e dal numero dei sensori che possono essere impiegati. Conoscendo il segnale di uscita \mathbf{y} è possibile costruire un'approssimazione dello stato completo $\hat{\mathbf{x}}$, utilizzando un modello a blocchi del sistema simile a quello usato per approssimare il modello reale (figura 2.3).

La matrice \mathbf{L} compie una funzione simile alla matrice \mathbf{K} del controllore, in quanto permette di definire l'accuratezza con cui l'uscita del sistema approssimato (detto *osservatore*) segue quella del sistema reale. L'equazione a cui risponde lo stato ricostruito risulta dunque:

$$\frac{d}{dt}\hat{\mathbf{x}} = (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}} + \mathbf{L}\mathbf{y} \quad (2.9)$$

Risulta quindi possibile, in genere, costruire in questo modo un'approssimazione dello stato del sistema che ne segua in maniera fedele il comportamento reale. Quest'approssimazione può essere usata per controllare il sistema reale come si farebbe conoscendone lo stato completo.

Affinché tutto questo sia possibile è necessario che il sistema sia *osservabile*. Uno stato \mathbf{x}_0 si dice osservabile se la conoscenza di $\mathbf{u}(t)$ e $\mathbf{y}(t)$ per un tempo finito, è sufficiente a ricostruire \mathbf{x}_0 . Ancora una volta, volendo essere più rigorosi, si può dire che uno stato \mathbf{x}_0 non è osservabile se la risposta dell'equazione (2.3) con $\mathbf{u} = 0$ e stato iniziale $\mathbf{x}(t_0) = \mathbf{x}_0$ è $\mathbf{y} \equiv 0, \forall t > t_0$. Un sistema è detto osservabile se ogni suo stato è osservabile. Come nel caso della controllabilità è possibile determinare l'osservabilità di un sistema a partire dalle matrici di

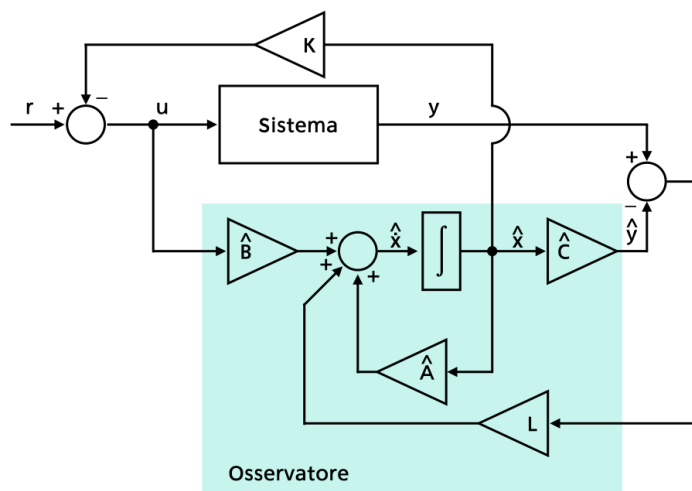


Figura 2.3: Sistema in anello chiuso, controllato utilizzando lo stato ricostruito \hat{x} come retroazione.

coefficienti, definendo il gramiano di osservabilità:

$$W_o := \int_0^{\infty} e^{A^T \tau} C^T C e^{A \tau} d\tau \quad (2.10)$$

Si può dimostrare che un sistema è osservabile se e solo se il gramiano W_o è non singolare.

Come per il caso della controllabilità è possibile, per sistemi lineari tempo invarianti, definire la matrice di osservabilità [3]:

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \quad (2.11)$$

Anche in questo caso il sistema risulta controllabile se il rango di questa matrice è massimo, ossia $\text{rank}(\mathcal{O}) = n$.

2.1.1 Controllo ottimo

Finora non si è detto niente di come sia possibile scegliere in maniera efficiente le matrici K ed L . È possibile selezionare gli autovalori delle matrici $A - BK$ e $\hat{A} - L\hat{C}$, nell'intento di soddisfare "a mano" le richieste del sistema attraverso

un processo di prova ed errore. Per sistemi single-input-single-output è possibile calcolare le matrici attraverso la formula di Ackerman [9], per sistemi più complessi esistono altri algoritmi, che soddisfano condizioni aggiuntive (come ad esempio il comando `place` di MatLab, che massimizza la robustezza contro l'incertezza sui valori di \mathbf{A} e \mathbf{B} [14]).

È possibile tuttavia scegliere tali matrici in modo tale che minimizzino funzioni di costo, trasformando il problema dalla selezione degli autovalori a uno di ottimizzazione. Nel caso del controllatore si può definire tale funzione come:

$$J = \frac{1}{2} \int_{t_0}^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) d\tau \quad (2.12)$$

Dove le matrici \mathbf{Q} , simmetrica e semidefinita positiva, e \mathbf{R} , simmetrica e definita positiva, contengono pesi associati alle singole componenti del vettore di stato e del vettore di input. Il problema di individuare una matrice \mathbf{K} che minimizzi J è detto controllatore lineare quadratico (LQR). Tale matrice viene data come:

$$\mathbf{K} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \quad (2.13)$$

dove \mathbf{P} è soluzione dell'equazione algebrica di Riccati:

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} + \mathbf{Q} = 0 \quad (2.14)$$

Per poter analizzare il problema dell'osservatore ottimo (detto filtro di Kalman, o estimatore lineare quadratico LQE) è necessario modificare le (2.3) e (2.4) affinché contengano l'effetto dell'incertezza del modello e del rumore legato alle misurazioni:

$$\frac{d}{dt} \mathbf{x} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u} + \mathbf{G} \mathbf{w} \quad (2.15)$$

$$\mathbf{y} = \mathbf{C} \mathbf{x} + \mathbf{v} \quad (2.16)$$

dove \mathbf{G} determina come si ripartisce il rumore \mathbf{w} sulle variabili di stato. I rumori si considerano gaussiani non correlati, con varianza conosciuta, ovvero:

$$E[\mathbf{w}] = E[\mathbf{v}] = 0 \quad (2.17)$$

$$E[\mathbf{w}^T \mathbf{w}] = \mathbf{Q}, \quad E[\mathbf{v}^T \mathbf{v}] = \mathbf{R} \quad (2.18)$$

dove, quindi, \mathbf{Q} ed \mathbf{R} sono le matrici di covarianza del rumore del sistema e dei sensori, rispettivamente. L'osservatore ottimo è scelto come quello che minimizza la covarianza dell'errore si estimazione, ovvero che minimizza:

$$\mathbf{P} = E[\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}] \quad (2.19)$$

dove $\boldsymbol{\varepsilon} = \mathbf{x} - \hat{\mathbf{x}}$. Il guadagno \mathbf{L} che risolve il problema è dato da:

$$\mathbf{L} = \mathbf{P} \mathbf{C}^T \mathbf{R}^{-1} \quad (2.20)$$

dove \mathbf{P} è soluzione dell'equazione di Riccati:

$$\mathbf{A} \mathbf{P} + \mathbf{P} \mathbf{A}^T + \mathbf{G} \mathbf{Q} \mathbf{B}^T - \mathbf{P} \mathbf{C}^T \mathbf{R}^{-1} \mathbf{C} \mathbf{P} = 0 \quad (2.21)$$

Un controllatore che implementi una retroazione LQR e un estimatore LQE è detto un controllatore lineare quadratico gaussiano (LQG)

2.1.2 Controllo predittivo basato su modello

Invece che ottimizzare l'azione di controllo per un orizzonte temporale infinito è possibile farlo in un intervallo di tempo finito, ripetendo l'ottimizzazione varie volte man mano che il sistema evolve. Questo approccio è detto controllo predittivo basato su modello (Model Predictive Control, MPC)[11]. Il sistema viene generalmente impostato con un tempo discretizzato:

$$\mathbf{x}(k+1) = \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \mathbf{u}(k) \quad (2.22)$$

dove con $\mathbf{x}(k)$ si intende $\mathbf{x}(k\Delta t)$, con [3]:

$$\mathbf{A}_d = e^{\mathbf{A}\Delta t} \quad (2.23)$$

$$\mathbf{B}_d = \int_0^{\Delta t} e^{\mathbf{A}\tau} \mathbf{B} d\tau \quad (2.24)$$

L'azione di controllo al passo di tempo k viene calcolata attraverso il seguente problema di ottimizzazione:

$$J_{(p,m)}(\mathbf{x}_0) = \min_{\mathbf{u}(\cdot)} \sum_{i=0}^{p-1} \mathbf{x}^T(i) \mathbf{Q} \mathbf{x}(i) + \sum_{i=0}^{m-1} \left(\mathbf{u}^T(i) \mathbf{R}_u \mathbf{u}(i) + \Delta \mathbf{u}^T(i) \mathbf{R}_{\Delta u} \Delta \mathbf{u}(i) \right) \quad (2.25)$$

soggetto a un insieme di restrizioni lineari

$$\mathbf{E} \mathbf{x} + \mathbf{F} \mathbf{u} \leq \boldsymbol{\psi} \quad (2.26)$$

dove $p \geq m$ è la lunghezza dell'orizzonte di predizione (o orizzonte dell'output) e m è la lunghezza dell'orizzonte di controllo (o orizzonte dell'input). Se $\mathbf{u}^*(i|\mathbf{x}(k))$, $i = 0, \dots, m-1$ è l'azione di controllo che minimizza $J_{(p,m)}(\mathbf{x}(k))$, allora è possibile calcolare lo stato al passo $k+1$ come

$$\mathbf{x}(k+1) = \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \mathbf{u}^*(0|\mathbf{x}(k)) \quad (2.27)$$

Il nuovo stato viene quindi usato per calcolare la successiva azione di controllo (si noti che \mathbf{u}^* per $i = 1, \dots, m-1$ non vengono utilizzati).

In generale non esistono soluzioni analitiche per il problema con restrizioni. Lo svantaggio di dover usare un algoritmo numerico per ottenere una soluzione è compensato dal vantaggio di poter imporre delle zone non accessibili per lo stato e l'azione del controllo, con indubbi vantaggi pratici [5]. È inoltre importante ricordare che il problema in esame cade nella categoria della programmazione quadratica, ampiamente studiata in letteratura [13].

Si noti inoltre che, in assenza di vincoli e considerando un orizzonte temporale infinito, il problema posto nelle (2.25) (2.26) è equivalente al problema del controllatore ottimo LQR studiato nella precedente sezione.

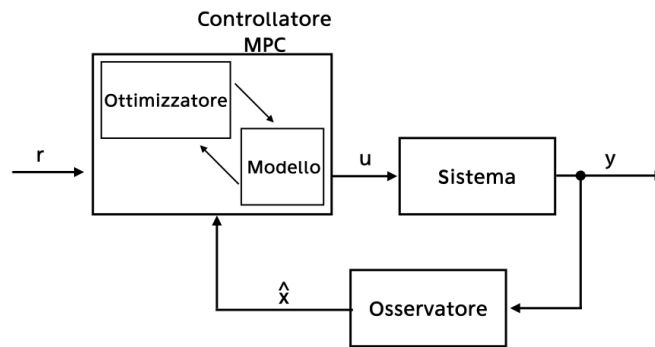


Figura 2.4: Schema a blocchi di un controllatore MPC con un osservatore per la ricostruzione dello stato \hat{x} a partire dall'output. È qui possibile apprezzare in maniera immediata perché l'MPC rientri nei controllatori ad anello chiuso.

Capitolo 3

Modelli di ordine ridotto

Si supponga di voler analizzare un sistema dinamico che evolve secondo l'equazione:

$$\frac{d}{dt} \mathbf{x} = \mathbf{f}(\mathbf{x}, t) \quad (3.1)$$

dove \mathbf{x} potrebbe essere un vettore che contiene le componenti della velocità ad ogni nodo di una griglia di calcolo e \mathbf{f} , in generale non lineare, potrebbe essere una discretizzazione delle equazioni di Navier-Stokes. Per problemi complessi di interesse ingegneristico il vettore \mathbf{x} potrebbe avere un numero di componenti n nell'ordine dei milioni. Risulta dunque di grande interesse cercare un sottospazio S con base $\{\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \dots, \boldsymbol{\psi}_r\}$, con $r \ll n$, dove sia possibile scrivere:

$$\mathbf{x}(t) \approx \sum_{i=1}^r a_i(t) \boldsymbol{\psi}_i \quad (3.2)$$

o, in notazione matriciale:

$$\mathbf{x} \approx \boldsymbol{\Psi} \mathbf{a}(t) \quad (3.3)$$

con

$$\boldsymbol{\Psi} = \begin{bmatrix} | & | & & | \\ \boldsymbol{\psi}_1 & \boldsymbol{\psi}_2 & \dots & \boldsymbol{\psi}_r \\ | & | & & | \end{bmatrix}, \quad \mathbf{a} = [a_1, a_2, \dots, a_r]^T$$

In generale non è possibile garantire che \mathbf{f} appartenga al medesimo sottospazio S . Si opta quindi per proiettarvi la dinamica tramite una matrice di proiezione \mathbf{P} .

Una matrice si dice di proiezione se

$$\mathbf{P}^2 = \mathbf{P} \quad (3.4)$$

questo comporta che [6]:

- gli unici autovalori possibili di \mathbf{P} sono 0 e 1;

- \mathbf{P} è diagonalizzabile;
- detto r il rango di \mathbf{P} , allora esiste una base \mathbf{V} tale che:

$$\mathbf{P} = \mathbf{V} \begin{bmatrix} \mathbf{I}_r & \mathbf{0}_{r \times n-r} \\ \mathbf{0}_{n-r \times r} & \mathbf{0}_{n-r \times n-r} \end{bmatrix} \mathbf{V}^{-1} \quad (3.5)$$

Scomponendo in due parti \mathbf{V} , in modo che $\mathbf{V} = [\mathbf{V}_1 \quad \mathbf{V}_2]$, con $\mathbf{V}_1 \in \mathbb{R}^{n,r}$, $\mathbf{V}_2 \in \mathbb{R}^{n,n-r}$, si ha che, $\forall \mathbf{x} \in \mathbb{R}^n$:

$$\begin{aligned} \mathbf{P}\mathbf{x} &\in \text{range}(\mathbf{V}_1) = \text{range}(\mathbf{P}) := S \\ \mathbf{x} - \mathbf{P}\mathbf{x} &\in \text{range}(\mathbf{V}_2) = \text{Ker}(\mathbf{P}) := \tilde{S} \end{aligned} \quad (3.6)$$

ovvero \mathbf{P} definisce una proiezione su S parallela ad \tilde{S} , e inoltre $S \oplus \tilde{S} = \mathbb{R}^n$.

Se $\tilde{S} \perp S$ allora, detta $\Psi \in \mathbb{R}^{n,r}$ una matrice ortogonale tale che $S = \text{range}(\Psi)$, la proiezione ortogonale su S è data da:

$$\mathbf{P}\mathbf{x} = \Psi\Psi^* \mathbf{x} \quad (3.7)$$

il che garantisce che il vettore risultante sia il più prossimo vettore di S nel senso della norma euclidea. Il simbolo $*$ denota la matrice trasposta coniugata, in modo che la teoria si possa applicare anche nel caso si tratti con dati nel campo dei numeri complessi.

La proiezione non deve per forza essere ortogonale. Nel caso generale in cui $\tilde{S} \not\perp S$ si prenda una matrice Φ tale che $\tilde{S}^\perp = \text{range}(\Phi)$, dove \tilde{S}^\perp è lo spazio ortogonale a \tilde{S} . La proiezione di \mathbf{x} su S parallelamente a \tilde{S} è data da:

$$\mathbf{P}\mathbf{x} = \Psi(\Phi^*\Psi)^{-1}\Phi^* \mathbf{x} \quad (3.8)$$

dove, se i vettori delle due basi sono mutuamente ortonormali, tali cioè che $\langle \psi_i, \phi_j \rangle = \delta_{ij}$ (dove δ_{ij} è la delta di Kronecker e $\langle \cdot, \cdot \rangle$ rappresenta il prodotto interno canonico su \mathbb{R}^n), allora $\mathbf{P} = \Psi\Phi^*$.

Richiamando la (3.1) e la scomposizione (3.3), l'equazione della dinamica nella base ridotta può essere scritta come:

$$\Psi \frac{d}{dt} \mathbf{a} = \mathbf{f}(\Psi \mathbf{a}, t) + \mathbf{r}(t) \quad (3.9)$$

dove si è fatto uso del fatto che Ψ non dipende dal tempo. Il residuo $\mathbf{r}(t)$ deriva dal fatto che la scomposizione non è esatta. Scegliendo Ψ e Φ in modo tale che $\Phi^* \mathbf{r}(t) = 0$ si giunge alla forma:

$$\frac{d}{dt} \mathbf{a} = (\Phi^* \Psi)^{-1} \Phi^* \mathbf{f}(\Psi \mathbf{a}, t) \quad (3.10)$$

ovvero un sistema di r equazioni differenziali invece che n . Questa procedura è detta proiezione di Petrov-Galerkin.

Nel caso che $\Psi = \Phi$, e che questa sia una matrice ortogonale, allora il modello ridotto è detto proiezione di Galerkin e l'equazione precedente si riduce a:

$$\frac{d}{dt}\mathbf{a} = \Psi^* \mathbf{f}(\Psi \mathbf{a}, t) \quad (3.11)$$

In generale, essendo S di dimensioni molto ridotte rispetto a \mathbb{R}^n il sistema risulta molto più rapido da risolvere. È poi immediato, risolta la dinamica di \mathbf{a} , calcolare nuovamente \mathbf{x} tramite Ψ .

Capitolo 4

Proper Orthogonal Decomposition

Finora non si è detto niente su come scegliere in maniera efficace la base $\{\psi\}$ su cui proiettare la dinamica. Una scelta spesso impiegata è rappresentata dall'espansione di Fourier, in quanto fornisce un'interpretazione immediata basata sulla rappresentazione del sistema nel dominio della frequenza ed esistono algoritmi molto efficienti per calcolarne i coefficienti [3]. Le funzioni di base, tuttavia, non rappresentano in maniera ottima il sistema, e rappresentare in maniera accurata una particolare soluzione può richiederne un numero elevato, vanificando di fatto lo scopo per il quale si sta cambiando base.

Per ottenere una riduzione della dimensione del problema l'obiettivo è individuare un insieme di funzioni che rappresentino in maniera abbastanza efficiente il sistema in esame da richiederne una manciata per caratterizzare la maggior parte della dinamica. Tale ricerca può essere effettuata a partire da una collezione di dati sull'evoluzione dinamica del sistema da cui poter estrapolare le informazioni necessarie, sia essa una simulazione ad alta fedeltà o una frutto di esperimenti.

4.1 POD

Il più semplice degli algoritmi per l'estrapolazione di una base a partire da un insieme di dati è la Proper Orthogonal Decomposition (POD), in quanto si fonda direttamente sulla scomposizione in valori singolari di una matrice che contenga i dati in esame. Tale matrice è costruita come:

$$\mathbf{X} = \begin{bmatrix} | & | & \dots & | \\ \mathbf{x}(t_1) & \mathbf{x}(t_2) & \dots & \mathbf{x}(t_m) \\ | & | & & | \end{bmatrix} \quad (4.1)$$

dove i vettori $\mathbf{x}(t_k)$ sono fermo immagini (snapshots) della simulazione completa.

Per ogni matrice appartenente a $\mathbb{R}^{n,m}$ esiste una scomposizione unica, detta scomposizione in valori singolari (SVD) tale che:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^* \quad (4.2)$$

dove $\mathbf{\Sigma} \in \mathbb{R}^{n,m}$ ha elementi non negativi sulla diagonale e le matrici $\mathbf{U} \in \mathbb{R}^{n,n}$ e $\mathbf{V} \in \mathbb{R}^{m,m}$ sono matrici unitarie, la cui colonne sono dette vettori singolari sinistri e destri rispettivamente. Gli elementi di $\mathbf{\Sigma}$ sono i cosiddetti valori singolari e vengono in genere ordinati in ordine decrescente.

È possibile tracciare una connessione tra la SVD e la POD attraverso la procedura proposta in [21].

Definendo la matrice

$$\mathbf{B} = \mathbf{\Sigma}\mathbf{V}^* \quad (4.3)$$

risulta

$$\mathbf{X} = \mathbf{U}\mathbf{B} \quad (4.4)$$

Data la struttura della matrice \mathbf{X} è possibile dare la scomposizione delle sue colonne \mathbf{x}_i nella base formata dalle colonne di \mathbf{U} usando i termini di \mathbf{B} come coefficienti. Sfruttando l'ortogonalità di \mathbf{U} si può scrivere:

$$\begin{aligned} \mathbf{x}_i &= \sum_{j=1}^n B_{j,i} \mathbf{u}_j = \sum_{j=1}^n (\mathbf{\Sigma}\mathbf{V}^*)_{j,i} \mathbf{u}_j = \sum_{j=1}^n (\mathbf{U}^* \mathbf{U} \mathbf{\Sigma}\mathbf{V}^*)_{j,i} \mathbf{u}_j = \\ &= \sum_{j=1}^n (\mathbf{U}^* \mathbf{X})_{j,i} \mathbf{u}_j = \sum_{j=1}^n \langle \mathbf{u}_j, \mathbf{x}_i \rangle \mathbf{u}_j \end{aligned} \quad (4.5)$$

Overo gli elementi di \mathbf{B} rappresentano la proiezione del vettore \mathbf{x}_i sui vettori della base $\{\mathbf{u}\}$.

Volendo ora trovare un singolo vettore che rappresenti al meglio tutti i \mathbf{x}_i contemporaneamente è necessario risolvere

$$\max_{\mathbf{u} \in \mathbb{R}^n} \sum_{i=1}^m \langle \mathbf{x}_i, \mathbf{u} \rangle, \text{ soggetto a } \|\mathbf{u}\|^2 = 1 \quad (4.6)$$

Con $\|\mathbf{u}\|^2 = \langle \mathbf{u}, \mathbf{u} \rangle$. La risoluzione di tale problema di ottimizzazione porta alla risoluzione del problema agli autovalori:

$$\mathbf{X}\mathbf{X}^* \mathbf{u} = \lambda \mathbf{u} \quad (4.7)$$

dove λ è il moltiplicatore di Langrange usato per implementare la restrizione. È possibile dimostrare che il primo vettore singolare sinistro \mathbf{u}_1 soddisfa la (4.7) e, dato l'ordinamento decrescente dei valori singolari, è soluzione di (4.6). Inoltre

$$\sum_{i=1}^m \langle \mathbf{x}_i, \mathbf{u}_1 \rangle = \lambda_1 = \sigma_1^2 \quad (4.8)$$

dove σ_1 è il primo valore singolare di \mathbf{X} .

Volendo ora trovare un secondo vettore che rappresenti in maniera ottima l'insieme di dati a disposizione, tale che sia ortogonale \mathbf{u}_1 , il problema si presenta in maniera molto simile:

$$\max_{\mathbf{u} \in \mathbb{R}^n} \sum_{i=1}^m \langle \mathbf{x}_i, \mathbf{u} \rangle, \text{ soggetto a } \|\mathbf{u}\|^2 = 1, \langle \mathbf{u}, \mathbf{u}_1 \rangle = 0 \quad (4.9)$$

Data l'ortogonalità delle colonne di \mathbf{U} segue che il vettore cercato è il secondo vettore singolare sinistro di \mathbf{X} , inoltre

$$\sum_{i=1}^m \langle \mathbf{x}_i, \mathbf{u}_2 \rangle = \sigma_2^2 \quad (4.10)$$

Il ragionamento può essere ripetuto fino all'ultimo vettore singolare non associato ad un valore singolare nullo.

La generalizzazione di quanto detto è data dal teorema di Eckart-Young, il quale stabilisce che, troncando la decomposizione al rango r , ossia approssimando \mathbf{X} come:

$$\mathbf{X}_r = \mathbf{U}_r \mathbf{\Sigma}_r \mathbf{V}_r^* \quad (4.11)$$

con $\mathbf{\Sigma}_r \in \mathbb{R}^{r,r}$ contenente i primi r valori singolari, e \mathbf{U}_r e \mathbf{V}_r i rispettivi vettori singolari, allora

$$\|\mathbf{X} - \mathbf{X}_r\|_F \leq \|\mathbf{X} - \mathbf{B}\|_F, \forall \mathbf{B} \in \mathbb{R}^{n,m} \text{ di rango } k \quad (4.12)$$

dove $\|\cdot\|_F$ è la norma di Frobenius, data da:

$$\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^m X_{ij}^2} \quad (4.13)$$

La base ridotta formata dalle prime r colonne di \mathbf{U} rappresenta quindi la migliore approssimazione possibile dei dati a disposizione nel senso della norma di Frobenius. La base fornita dalla POD può essere interpretata come quella che cattura in maniera ottima l'energia del sistema analizzato. È in questo pregio della POD che si incontra anche il principale difetto quando il sistema analizzato è non lineare o in cui ci sia una forte correlazione fra la dinamica su scale diverse del sistema: modi di bassa energia, quindi troncati nella POD, possono in certi sistemi (si pensi a flussi turbolenti o onde sonore) avere un grande impatto sulla dinamica di larga scala [18].

4.2 BPOD

Un metodo per ovviare alle limitazioni della POD è quello di troncare quei modi che non sono controllabili o osservabili. Questo consente di applicare la teoria del controllo ad anello chiuso (sec. 2.1) a solo quegli stati che hanno un grande impatto sulla dinamica e che si prestano ad essere controllati.

4.2.1 Balanced Truncation

Il metodo sviluppato nell'ambito della teoria del controllo è detto *Balanced Truncation* [17]. Il metodo si basa sul notare che i gramiani di controllabilità e osservabilità definiti in (2.7) e (2.10) si trasformano in seguito ad un cambio di coordinate $\mathbf{x} = \mathbf{T}\mathbf{z}$ come:

$$\mathbf{W}_c \mapsto \mathbf{T}^{-1}\mathbf{W}_c(\mathbf{T}^{-1})^* \quad (4.14)$$

$$\mathbf{W}_o \mapsto \mathbf{T}^T\mathbf{W}_o\mathbf{T} \quad (4.15)$$

Risulta possibile scegliere tale trasformazione in modo che i gramiani trasformati siano diagonali e uguali (da qui il termine *balanced*), ovvero:

$$\mathbf{T}^{-1}\mathbf{W}_c(\mathbf{T}^{-1})^* = \mathbf{T}^T\mathbf{W}_o\mathbf{T} = \mathbf{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_n) \quad (4.16)$$

dove gli elementi di $\mathbf{\Sigma}$ sono detti valori singolari di Hankel. La trasformazione \mathbf{T} cercata può essere calcolata come:

$$\mathbf{W}_c\mathbf{W}_o\mathbf{T} = \mathbf{T}\mathbf{\Sigma}^2 \quad (4.17)$$

ovvero come un problema agli autovalori.

Chiamata $\mathbf{T}^{-1} = \mathbf{S}$ per brevità, la dinamica (2.3), (2.4) diventa (con $\mathbf{D} = 0$)

$$\frac{d}{dt}\mathbf{z} = \mathbf{S}\mathbf{A}\mathbf{T}\mathbf{z} + \mathbf{S}\mathbf{B} \quad (4.18)$$

$$\mathbf{y} = \mathbf{C}\mathbf{T}\mathbf{z} \quad (4.19)$$

Di questo sistema è possibile troncare tutti gli elementi z_i con $i > r$, mantenendo quindi i modi che risultano contemporaneamente più controllabili ed osservabili. Questo metodo dà ottimi risultati nel produrre un modello di ordine ridotto per il controllo [18], ma per sistemi con \mathbf{x} molto grandi il calcolo dei gramiani e degli autovettori di $\mathbf{W}_c\mathbf{W}_o$ è intrattabile.

4.2.2 BPOD

Per superare i limiti al costo computazionale associato ai gramiani, è possibile approssimarli attraverso la risposta all'impulso del sistema e del suo aggiunto [17]. Seguendo la procedura proposta in [3] si studia l'evoluzione del sistema in tempo discreto:

$$\text{diretto} \quad \mathbf{x}(k+1) = \mathbf{A}_d\mathbf{x}(k) + \mathbf{B}_d\mathbf{u}(k) \quad (4.20)$$

$$\text{aggiunto} \quad \mathbf{x}(k+1) = \mathbf{A}_d^*\mathbf{x}(k) + \mathbf{C}^*\mathbf{y}(k) \quad (4.21)$$

Si noti che, mentre i dati del sistema diretto possono essere presi da esperimenti, il sistema aggiunto non è fisico e deve quindi essere simulato. Esistono formulazioni alternative che non fanno uso del sistema aggiunto [3], [18].

Scrivendo \mathbf{B} come $\mathbf{B} = [\mathbf{b}_1 \dots \mathbf{b}_p]$ la risposta ad un impulso unitario in uno degli input, ovvero la risposta a $u_i = \delta(t)$, è data da:

$$\mathbf{x}(k) = \mathbf{A}_d^k\mathbf{b}_i \quad (4.22)$$

organizzando i dati per un numero m_c di passi temporali in una matrice si ottiene:

$$\mathbf{C}_d = [\mathbf{B}_d \mathbf{A}_d \mathbf{B}_d \dots \mathbf{A}_d^{m_c-1} \mathbf{B}_d] \quad (4.23)$$

Analogamente per il sistema aggiunto, per un numero m_o di passi, si ottiene:

$$\mathbf{O}_d = \begin{bmatrix} \mathbf{C} \\ \mathbf{C} \mathbf{A}_d \\ \vdots \\ \mathbf{C} \mathbf{A}_d^{m_o-1} \end{bmatrix} \quad (4.24)$$

Si noti che le matrici delle risposte impulsive in tempo discreto \mathbf{C}_d e \mathbf{O}_d sono strettamente legate alle matrici di controllabilità e osservabilità definite in precedenza.

I gramiani di controllabilità e osservabilità possono essere approssimati mediante gramiani empirici dati da:

$$\mathbf{W}_c \approx \mathbf{W}_c^e = \mathbf{C}_d \mathbf{C}_d^* \quad (4.25)$$

$$\mathbf{W}_o \approx \mathbf{W}_o^e = \mathbf{O}_d^* \mathbf{O}_d \quad (4.26)$$

$$(4.27)$$

Si definisce quindi la matrice di Hankel generalizzata:

$$\mathbf{H} = \mathbf{O}_d \mathbf{C}_d \quad (4.28)$$

Che viene poi fattorizzata usando la SVD:

$$\mathbf{H} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^* = [\mathbf{U}_r \ \mathbf{U}_t] \begin{bmatrix} \mathbf{\Sigma}_r & 0 \\ 0 & \mathbf{\Sigma}_t \end{bmatrix} \begin{bmatrix} \mathbf{V}_r^* \\ \mathbf{V}_t^* \end{bmatrix} \approx \mathbf{U}_r \mathbf{\Sigma}_r \mathbf{V}_r^* \quad (4.29)$$

Per ottenere il modello di ordine $r \ll n$ si definiscono le trasformazioni:

$$\text{modi diretti: } \mathbf{\Psi} = \mathbf{C}_d \mathbf{V}_r \mathbf{\Sigma}_r^{-1/2} \quad (4.30)$$

$$\text{modi aggiunti: } \mathbf{\Phi} = \mathbf{O}_d^* \mathbf{U}_r \mathbf{\Sigma}_r^{-1/2} \quad (4.31)$$

Dove i modi diretti $\mathbf{\Psi} \in \mathbb{R}^{n,r}$ e aggiunti $\mathbf{\Phi} \in \mathbb{R}^{n,r}$ non mutuamente ortonormali, ovvero: $\mathbf{\Phi}^* \mathbf{\Psi} = \mathbf{I}_{r,r}$ (dove $\mathbf{I}_{r,r}$ è la matrice identità in $\mathbb{R}^{r,r}$). Questi modi bilanciano i gramiani empirici di controllabilità e osservabilità [17], e rappresentano quindi un'approssimazione delle trasformazioni \mathbf{T} ed \mathbf{S} definite nella precedente sezione. In particolare $\mathbf{\Psi}$ approssima le prime r colonne di \mathbf{T} , mentre $\mathbf{\Phi}^*$ le prime r righe di $\mathbf{\Phi}$. Le matrici che definiscono la dinamica di ordine ridotto sono quindi:

$$\tilde{\mathbf{A}} = \mathbf{\Phi}^* \mathbf{A}_d \mathbf{\Psi} \quad (4.32)$$

$$\tilde{\mathbf{B}} = \mathbf{\Phi}^* \mathbf{B}_d \quad (4.33)$$

$$\tilde{\mathbf{C}} = \mathbf{\Phi}^* \mathbf{C}_d \quad (4.34)$$

4.2.3 Output Projection

É importante notare che, se il numero di output del sistema è molto grande (per esempio se $\mathbf{C}_d = \mathbf{I}$) il numero di simulazioni del sistema aggiunto diventa proibitivo. Risulta quindi necessario trovare un modo di ridurlo in maniera ottimale. A tale scopo si può porre [17]:

$$\mathbf{y} = \mathbf{\Psi}_d \mathbf{\Psi}_d^* \mathbf{C} \mathbf{x} \quad (4.35)$$

dove $\mathbf{\Psi}_d$ sono i primi r modi POD del sistema diretto. In questo modo si sono proiettati gli output ortogonalmente in un sottospazio di dimensione ridotta che cattura in maniera ottimale l'energia del sistema. Siccome il gramiano di osservabilità in tempo continuo sarebbe (tenendo conto che $\mathbf{\Psi}_d^* \mathbf{\Psi}_d = \mathbf{I}$):

$$\mathbf{W}_o := \int_0^\infty e^{\mathbf{A}^* \tau} \mathbf{C}^* \mathbf{\Psi}_d \mathbf{\Psi}_d^* \mathbf{C} e^{\mathbf{A} \tau} d\tau \quad (4.36)$$

può essere approssimato attraverso r simulazioni del sistema:

$$\mathbf{x}_{k+1} = \mathbf{A}_d^* \mathbf{x}_k + \mathbf{C}_d^* \mathbf{\Psi}_d \mathbf{y}_k \quad (4.37)$$

Capitolo 5

Dynamic Mode Decomposition

Un secondo algoritmo che ha riscosso un certo successo in ambito fluidodinamico è la Dynamic Mode Decomposition, o DMD. Quest'algoritmo, oltre a permettere di calcolare una base di ordine ridotto ha il vantaggio di essere basato unicamente sui dati raccolti come output da un sistema, del quale non si conosce analiticamente la dinamica. Quest'algoritmo ha dunque il vantaggio sopra la POD (per quanto, come si vedrà nelle sezioni seguenti, la DMD può fare uso della POD per ottenere un sistema di ordine ridotto rispetto ai dati raccolti) di essere in grado di "scoprire" una legge della dinamica lineare e di basso rango che rappresenti al meglio i dati a disposizione. Inoltre la DMD è collegata con la teoria dell'operatore di Koopman [19] [16], un approccio allo studio dei sistemi dinamici che consente di studiare un sistema lineare di dimensione infinita invece di uno a dimensione finita ma non lineare.

5.1 DMD

Dato un sistema dinamico in tempo discreto

$$\mathbf{x}(k+1) = \mathbf{F}(\mathbf{x}(k)) \quad (5.1)$$

in generale non lineare, anche nel caso in cui \mathbf{F} non sia nota esplicitamente, la dynamic mode decomposition ha l'obiettivo di approssimarne la dinamica con un modello lineare tempo invariante:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) \quad (5.2)$$

attraverso uno studio dei dati prodotti dal modello in esperimenti o simulazioni. Seguendo la procedura presentata in [10], assumendo di avere accesso all'intero

stato \mathbf{x} del sistema in m intervalli di tempo, si costruiscono le matrici:

$$\mathbf{X} = \begin{bmatrix} | & | & \dots & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_{m-1} \\ | & | & & | \end{bmatrix} \quad (5.3)$$

$$\mathbf{X}' = \begin{bmatrix} | & | & \dots & | \\ \mathbf{x}_2 & \mathbf{x}_3 & \dots & \mathbf{x}_m \\ | & | & & | \end{bmatrix} \quad (5.4)$$

$$(5.5)$$

dove $\mathbf{x}_k = \mathbf{x}(k\Delta t)$. Siccome la matrice \mathbf{X}' non è altro che l'evoluzione di \mathbf{X} in un passo di tempo si cerca una matrice tale che:

$$\mathbf{X}' = \mathbf{A}\mathbf{X} \quad (5.6)$$

Tale matrice può essere calcolata come

$$\mathbf{A} \approx \mathbf{X}'\mathbf{X}^\dagger \quad (5.7)$$

dove \mathbf{X}^\dagger è la pseudoinversa di Moore-Penrose, in modo da minimizzare:

$$\|\mathbf{X}' - \mathbf{A}\mathbf{X}\|_F \quad (5.8)$$

Per sistemi molto grandi, dove il numero n di elementi dello stato \mathbf{x}_k è nell'ordine di 10^6 , \mathbf{A} avrebbe 10^{12} elementi, il che, oltre che essere poco pratico, potrebbe comportare l'impossibilità di stivare l'intera matrice in memoria. Risulta quindi necessario proiettare la dinamica in un sottospazio a dimensione ridotta prima di calcolare effettivamente \mathbf{A} .

Si noti inoltre l'utilità di calcolare la decomposizione spettrale di \mathbf{A} : se questa infatti può essere diagonalizzata dal cambio di coordinate $\mathbf{x} = \mathbf{\Phi}\mathbf{z}$, tale che:

$$\mathbf{\Phi}^{-1}\mathbf{A}\mathbf{\Phi} = \mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_n) \quad (5.9)$$

dove $\mathbf{\Lambda}$ contiene gli autovalori di \mathbf{A} e le colonne di $\mathbf{\Phi}$ sono gli autovettori, allora l'evoluzione temporale in tempo discreto è data da:

$$\mathbf{x}_k = \mathbf{\Phi}\mathbf{\Lambda}^k\mathbf{z}_0 \quad (5.10)$$

$$\mathbf{\Lambda}^k = \text{diag}(\lambda_1^k, \dots, \lambda_n^k) \quad (5.11)$$

$$\mathbf{z}_0 = \mathbf{\Phi}^\dagger\mathbf{x}(0) \quad (5.12)$$

L'algoritmo completo può quindi essere dato come:

1. Si approssima la matrice \mathbf{X} con una matrice di rango r dato come nella POD da:

$$\mathbf{X} \approx \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^* \quad (5.13)$$

2. La matrice \mathbf{A} è data tramite la pseudoinversa come:

$$\mathbf{A} \approx \mathbf{X}'\mathbf{V}_r\boldsymbol{\Sigma}_r^{-1}\mathbf{U}_r^* \quad (5.14)$$

proiettando \mathbf{A} sui modi della POD, ovvero $\tilde{\mathbf{A}} = \mathbf{U}_r^*\mathbf{A}\mathbf{U}_r$ si ottiene:

$$\tilde{\mathbf{A}} = \mathbf{U}_r^*\mathbf{X}'\mathbf{V}_r\boldsymbol{\Sigma}_r^{-1} \quad (5.15)$$

dove $\tilde{\mathbf{A}}$ definisce la dinamica sulle coordinate ridotte della POD:

$$\mathbf{z}_{k+1} = \tilde{\mathbf{A}}\mathbf{z}_k \quad (5.16)$$

con $\mathbf{x}_k = \mathbf{U}\mathbf{z}_k$

3. Si calcola la decomposizione spettrale di $\tilde{\mathbf{A}}$:

$$\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\boldsymbol{\Lambda} \quad (5.17)$$

4. La decomposizione di \mathbf{A} è data quindi da:

$$\mathbf{A}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda} \quad (5.18)$$

$$\boldsymbol{\Phi} = \mathbf{X}'\mathbf{V}_r\boldsymbol{\Sigma}_r^{-1}\mathbf{W} \quad (5.19)$$

le colonne di $\boldsymbol{\Phi}$ sono i modi della DMD.

È possibile ricavare l'evoluzione in tempo continuo ponendo $\omega_k = \ln(\lambda_k)/\Delta t$:

$$\mathbf{x}(t) \approx \boldsymbol{\Phi}\mathbf{z}_0 e^{\boldsymbol{\Omega}t} \quad (5.20)$$

con $\boldsymbol{\Omega} = \text{diag}(\omega_1, \dots, \omega_r)$.

5.1.1 Analisi di Koopman

Uno dei motivi per i quali la DMD suscita tanto interesse è la connessione che si può tracciare con l'analisi di Koopman.

Le potenzialità della teoria di Koopman derivano dalla possibilità di trasformare un sistema dinamico non lineare in uno spazio di dimensione finita in uno governato da equazioni lineari, al prezzo di passare ad uno spazio di Hilbert di dimensione infinita[19].

Dato il sistema dinamico non lineare in tempo discreto:

$$\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k), \quad \text{con } \mathbf{x}_k \in \mathbb{R}^n \quad (5.21)$$

La teoria è più generale e ammette che $\mathbf{x}_k \in \mathcal{M}$, dove \mathcal{M} è una varietà differenziabile. Data la natura numerica dei dati considerati in questo lavoro ci si limiterà a $\mathcal{M} = \mathbb{R}^n$.

Definiamo l'insieme delle funzioni di \mathbf{x}_k $g : \mathbb{R}^n \rightarrow \mathbb{R}$ tali che formino uno spazio di Hilbert \mathcal{H} (e.g. l'insieme delle funzioni a quadrato integrabili), dette *osservabili*. Si definisce l'operatore di Koopman $\mathcal{K} : \mathcal{H} \rightarrow \mathcal{H}$ come:

$$\mathcal{K}g(\mathbf{x}_k) := g(\mathbf{F}(\mathbf{x}_k)) = g(\mathbf{x}_{k+1}) \quad (5.22)$$

Siccome l'operatore \mathcal{K} è lineare è possibile darle la decomposizione spettrale come:

$$\mathcal{K}\phi_i = \lambda_i\phi_i, \quad i = 1, 2, \dots, \infty \quad (5.23)$$

Definendo l'osservabile $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, supponendo che $\mathbf{g} \in \text{span}(\phi)$, si può scrivere:

$$\mathbf{g}(\mathbf{x}_k) = \begin{bmatrix} g_1(\mathbf{x}_k) \\ g_2(\mathbf{x}_k) \\ \vdots \\ g_m(\mathbf{x}_k) \end{bmatrix} = \sum_{i=1}^{\infty} \phi_i(\mathbf{x}_k) \mathbf{v}_i \quad (5.24)$$

dove \mathbf{v}_i sono detti modi di Koopman. È possibile ora dare una nuova formulazione dell'azione di \mathcal{K} come:

$$\mathcal{K}\mathbf{g}(\mathbf{x}_k) = \mathcal{K} \sum_{i=1}^{\infty} \phi_i(\mathbf{x}_k) \mathbf{v}_i = \sum_{i=1}^{\infty} \lambda_i \phi_i(\mathbf{x}_k) \mathbf{v}_i \quad (5.25)$$

Si nota che l'osservabile al tempo k può essere scritto come:

$$\mathbf{g}(\mathbf{x}_k) = \sum_{i=1}^{\infty} \mathcal{K}^k \phi_i(\mathbf{x}_0) \mathbf{v}_i = \sum_{i=1}^{\infty} \lambda_i^k \phi_i(\mathbf{x}_0) \mathbf{v}_i \quad (5.26)$$

da cui si deduce che l'evoluzione temporale viene data dagli autovalori dell'operatore di Koopman.

La DMD può essere vista come una tecnica di approssimazione dei modi di Koopman [19] [16]. Dette infatti:

$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k), \quad \mathbf{z}_k = \mathbf{g}(\mathbf{x}_{k+1}) \quad (5.27)$$

delle misurazioni di un sistema che evolve secondo la (5.21), si possono costruire le matrici:

$$\mathbf{Y} = \begin{bmatrix} | & | & \dots & | \\ \mathbf{y}_1 & \mathbf{y}_2 & \dots & \mathbf{y}_m \\ | & | & \dots & | \end{bmatrix}, \quad \mathbf{Z} = \begin{bmatrix} | & | & \dots & | \\ \mathbf{z}_1 & \mathbf{z}_2 & \dots & \mathbf{z}_m \\ | & | & \dots & | \end{bmatrix} \quad (5.28)$$

Si noti che in generale le misurazioni non devono per forza essere ordinate, fintantoché \mathbf{y}_k e \mathbf{z}_k sono nella medesima posizione.

Calcolando, come prima, $\mathbf{A} = \mathbf{Z}\mathbf{Y}^\dagger$, si ha che:

$$\mathbf{z}_k \approx \mathbf{A}\mathbf{y}_k \Rightarrow \mathbf{g}(\mathbf{x}_{k+1}) \approx \mathbf{A}\mathbf{g}(\mathbf{x}_k) \quad (5.29)$$

Data la decomposizione di \mathbf{A} :

$$\mathbf{A}\mathbf{v}_i = \lambda_i \mathbf{v}_i \quad (5.30)$$

se questa possiede un set completo di autovettori, allora è possibile proiettarvi \mathbf{y}_k :

$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k) = \sum_{i=1}^n c_{ik} \mathbf{v}_i = \sum_{i=1}^n \langle \mathbf{y}_k, \mathbf{w}_i \rangle \mathbf{v}_i \quad (5.31)$$

dove \mathbf{w}_i sono gli autovalori della matrice trasposta coniugata di \mathbf{A} , tali che $\langle \mathbf{v}_i, \mathbf{w}_j \rangle = \delta_{ij}$. Si ottiene quindi che:

$$\begin{aligned} \mathbf{z}_k = \mathbf{g}(\mathbf{F}(\mathbf{x}_k)) &\approx \mathbf{A}\mathbf{y}_k = & (5.32) \\ &= \mathbf{A} \sum_{i=1}^n c_{ik} \mathbf{v}_i = \\ &= \sum_{i=1}^n \lambda_i c_{ik} \mathbf{v}_i \end{aligned}$$

Se l'uguaglianza vale in maniera esatta, ovvero la dinamica è lineare, gli autovettori di \mathbf{A} e i modi di Koopman coincidono, il che si vede immediatamente confrontando la (5.32) con la (5.25).

5.2 DMDc

La DMD come è stata formulata nella precedente sezione non permette di identificare l'azione di un controllatore sul sistema. Si è fatto implicitamente uso dell'ipotesi che le matrici di dati potessero essere raccolte osservando la dinamica non controllata, i.e. con $\mathbf{u}(t) = 0$, e che l'effetto di eventuali attuatori sul sistema fosse poi noto nel caso il modello dovesse essere usato per il controllo. Si noti infatti che se i dati provenissero da un sistema controllato, dove quindi $\mathbf{f} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ l'algoritmo presentato precedentemente produrrebbe una matrice \mathbf{A} incorretta[15].

La procedura può essere modificata come segue per tenere in conto di tali input. Si supponga di avere un sistema dinamico controllato, in tempo discreto, tale che:

$$\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{u}_k) \quad (5.33)$$

Supponendo di avere accesso allo stato completo e al vettore di input per ogni passo di tempo, si costruiscono le matrici \mathbf{X}' e \mathbf{X} come visto in precedenza nelle (5.4) e (5.5). Si costruisce inoltre la matrice degli input:

$$\mathbf{\Upsilon} = \begin{bmatrix} | & | & \dots & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_{m-1} \\ | & | & & | \end{bmatrix} \quad (5.34)$$

Si vorranno quindi trovare le matrici \mathbf{A} e \mathbf{B} tali che:

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{\Upsilon} \quad (5.35)$$

5.2.1 B nota

Nel caso ideale in cui la matrice \mathbf{B} fosse nota, la procedura non è poi molto differente rispetto alla DMD.

Si ha infatti che:

$$\mathbf{A}\mathbf{X} = \mathbf{X}' - \mathbf{B}\mathbf{\Upsilon} \quad (5.36)$$

$$\Rightarrow \mathbf{A} \approx (\mathbf{X}' - \mathbf{B}\mathbf{\Upsilon})\mathbf{X}^\dagger = (\mathbf{X}' - \mathbf{B}\mathbf{\Upsilon})\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^* \quad (5.37)$$

Dove, per quanto si stia intendendo che la SVD venga troncata al rango r , il pedice è stato omissso per snellire la notazione.

Siccome il calcolo in questa forma può risultare intrattabile per problemi di grande dimensione, si opta per proiettare lo stato in una base calcolata tramite la POD, ovvero si opera il cambio di base $\mathbf{x}_k = \mathbf{U}\mathbf{z}_k$, da cui:

$$\mathbf{z}_{k+1} = \mathbf{U}^*\mathbf{A}\mathbf{U}\mathbf{z}_k + \mathbf{U}^*\mathbf{B}\mathbf{u}_k = \quad (5.38)$$

$$= \mathbf{U}^*(\mathbf{X}' - \mathbf{B}\mathbf{\Upsilon})\mathbf{V}\mathbf{\Sigma}^{-1} + \mathbf{U}^*\mathbf{B}\mathbf{u}_k = \tilde{\mathbf{A}}\mathbf{z}_k + \mathbf{U}^*\mathbf{B}\mathbf{u}_k \quad (5.39)$$

ancora una volta, data la decomposizione della matrice ridotta $\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda}$, i modi della DMDc sono dati da:

$$\mathbf{\Phi} = (\mathbf{X}' - \mathbf{B}\mathbf{\Upsilon})\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{W} \quad (5.40)$$

5.2.2 B non nota

Nel caso più generale la matrice \mathbf{B} non è nota e deve essere calcolata con \mathbf{A} . In questo caso la dipendenza di \mathbf{X}' con \mathbf{X} e $\mathbf{\Upsilon}$ può essere scritta come.

$$\mathbf{X}' \approx \mathbf{G}\mathbf{\Omega} \quad (5.41)$$

$$\text{con } \mathbf{G} = [\mathbf{A} \ \mathbf{B}], \quad \mathbf{\Omega} = \begin{bmatrix} \mathbf{X} \\ \mathbf{\Upsilon} \end{bmatrix} \quad (5.42)$$

dunque

$$\mathbf{G} = \mathbf{X}'\mathbf{\Omega}^\dagger \quad (5.43)$$

$$[\mathbf{A} \ \mathbf{B}] = \mathbf{X}' \begin{bmatrix} \mathbf{X} \\ \mathbf{\Upsilon} \end{bmatrix}^\dagger \quad (5.44)$$

e la DMDc sarà, similarmnte alla DMD, la decomposizione in autovalori e autovettori di \mathbf{G} . Ancora una volta l'inversa di Moore-Penrose è approssimata tramite una SVD troncata al rango p :

$$\mathbf{G} \approx \mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^* \quad (5.45)$$

Come si può facilmente intuire dalla figura 5.1 per poter ottenere \mathbf{A} e \mathbf{B} è necessario dividere \mathbf{U}^* in due parti, $\mathbf{U}_1^* \in \mathbb{R}^{p,n}$ e $\mathbf{U}_2^* \in \mathbb{R}^{p,l}$, ovvero:

$$[\mathbf{A} \ \mathbf{B}] = [\mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}_1^* \ \mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}_2^*] \quad (5.46)$$

Le matrici calcolate in questo modo hanno una dimensione eccessiva per essere efficacemente usate nella progettazione di un controllatore, dunque si opta

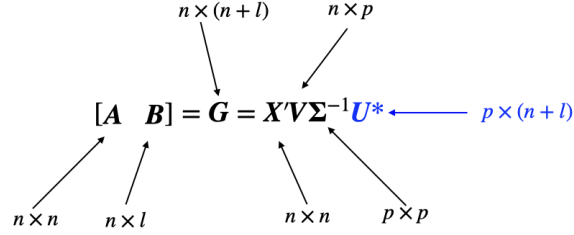


Figura 5.1: Schema delle dimensioni delle matrici coinvolte nella DMDc, prima della proiezione su una base di ordine ridotto.

per proiettare la dinamica su una base di ordine ridotto $\mathbf{x} = \mathbf{Tz}$. A differenza di quanto fatto finora non è possibile usare direttamente la matrice \mathbf{U} già calcolata, perché questa, essendo parte della decomposizione di $\mathbf{\Omega}$, definisce una base dello spazio degli *input*. La scelta dovrà dunque ricadere sulla decomposizione della matrice che definisce lo spazio degli *output* se si vuole rappresentare in maniera ottimale la variabile di output \mathbf{x} . Si usa quindi $\mathbf{X}' \approx \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^*$, dove la SVD è troncata al rango r , con in generale $r < p$. La dinamica di ordine ridotto è data quindi da:

$$\mathbf{z}_{k+1} = \tilde{\mathbf{A}}\mathbf{z}_k + \tilde{\mathbf{B}}\mathbf{u}_k \quad (5.47)$$

$$\text{dove } \tilde{\mathbf{A}} = \hat{\mathbf{U}}^*\mathbf{X}'\mathbf{V}\Sigma^{-1}\mathbf{U}_1^*\hat{\mathbf{U}}, \quad \tilde{\mathbf{B}} = \hat{\mathbf{U}}^*\mathbf{X}'\mathbf{V}\Sigma^{-1}\mathbf{U}_2^* \quad (5.48)$$

Una volta calcolati gli autovettori di $\tilde{\mathbf{A}}$ tramite l'equazione $\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda}$ gli autovettori di \mathbf{A} sono dati dalle colonne di:

$$\mathbf{\Phi} = \mathbf{X}'\mathbf{V}\Sigma^{-1}\mathbf{U}_1^*\hat{\mathbf{U}}\mathbf{W} \quad (5.49)$$

Capitolo 6

Applicazioni

Avere a disposizione modelli di ordine ridotto è di fondamentale importanza in quegli ambiti dove la rapidità di calcolo è un elemento determinante. Sarebbe molto più costoso avviare un algoritmo di ottimizzazione senza un adeguata riduzione della dimensionalità del sistema, e risulterebbe impossibile usare le equazioni complete di Navier-Stokes per un controllatore in tempo reale. Un controllatore MPC, che risolve un problema di ottimizzazione online rappresenta forse il campo dove la scelta di un modello adeguato risulta totalmente indispensabile. Tecniche come la DMD possono essere addirittura usate per dare un'equazione matematica lineare a sistemi dinamici che evolvono seguendo leggi non note a priori.

In questo capitolo si porteranno esempi di controllori costruiti utilizzando un modello di ordine ridotto del sistema, ottenuto attraverso POD o DMD.

6.1 Flusso intorno a una piastra piana

In questa sezione si studierà l'applicazione di un modello di ordine ridotto ottenuto attraverso la BPOD alla stabilizzazione di un flusso bidimensionale intorno ad una piastra piana a basso numero di Reynolds ($Re = 100$) e alti angoli di attacco, tramite un controllore ad anello chiuso.

La motivazione per tale studio, portata avanti da Ahuja e Rowley [1], deriva dall'interesse per lo sviluppo di velivoli miniaturizzati (micro air vehicles, o MAVs). Lo studio del volo di uccelli e insetti ha evidenziato come vortici localizzati al bordo d'attacco possano aiutare a stabilizzare il flusso sull'ala, generando una portanza maggiore [2]: progettare controllori in grado di replicare questa funzione porterebbe a MAVs con maggiore portanza e una manovrabilità migliore in presenza di raffiche. In particolare nello studio viene progettato un controllore ad anello chiuso in grado di sopprimere in maniera efficace la separazione di vortici dovuta all'alto angolo di incidenza della corrente in arrivo.

6.1.1 BPOD per sistemi instabili

Siccome la dimensione dello stato \mathbf{x} dato dalla discretizzazione delle equazioni di Navier-Stokes è di dimensione eccessiva per poter ottenere un controllore efficace, si opta per proiettare la dinamica in uno spazio di dimensione ridotta tramite la BPOD. Sorge tuttavia il problema del trattamento dei modi instabili, in quanto la risposta agli impulsi non sarebbe limitata, il che determina l'impossibilità di calcolare i gramiani empirici come nella sezione 4.2. Si noti inoltre che i gramiani definiti in (2.7) e (2.10) non sono ben definiti quando \mathbf{A} ha autovalori con parte reale positiva, in quanto gli integrali tendono a infinito. Seguendo l'approccio dato in [20], si definiscono i gramiani generalizzati nel dominio della frequenza:

$$\mathbf{W}_c = \frac{1}{2\pi} \int_{-\infty}^{\infty} (i\omega\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \mathbf{B}^* (-i\omega\mathbf{I} - \mathbf{A}^*)^{-1} d\omega \quad (6.1)$$

$$\mathbf{W}_o = \frac{1}{2\pi} \int_{-\infty}^{\infty} (-i\omega\mathbf{I} - \mathbf{A}^*)^{-1} \mathbf{C}^* \mathbf{C} (i\omega\mathbf{I} - \mathbf{A})^{-1} d\omega \quad (6.2)$$

i quali sono identici ai gramiani già definiti per sistemi stabili, ma risultano ben definiti anche per quelli instabili.

Dato il cambio di coordinate $\mathbf{x} = \mathbf{T}\tilde{\mathbf{x}}$ tale che la dinamica stabile e instabile risulti disaccoppiata, si ottiene il sistema nello spazio degli stati:

$$\frac{d}{dt} \tilde{\mathbf{x}} = \frac{d}{dt} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{x}_s \end{bmatrix} = \begin{bmatrix} \mathbf{A}_i & 0 \\ 0 & \mathbf{A}_s \end{bmatrix} \tilde{\mathbf{x}} + \begin{bmatrix} \mathbf{B}_i \\ \mathbf{B}_s \end{bmatrix} \mathbf{u} \quad (6.3)$$

$$\mathbf{y} = \begin{bmatrix} \mathbf{C}_i & \mathbf{C}_s \end{bmatrix} \tilde{\mathbf{x}} \quad (6.4)$$

detti \mathbf{W}_c^i , \mathbf{W}_o^i , \mathbf{W}_c^s , \mathbf{W}_o^s i gramiani di controllabilità e osservabilità dei sottosistemi stabile e instabile rispettivamente, questi sono legati ai gramiani del sistema completo da:

$$\mathbf{W}_c = \mathbf{T} \begin{bmatrix} \mathbf{W}_c^i & 0 \\ 0 & \mathbf{W}_c^s \end{bmatrix} \mathbf{T}^* \quad (6.5)$$

$$\mathbf{W}_o = (\mathbf{T}^{-1})^* \begin{bmatrix} \mathbf{W}_o^i & 0 \\ 0 & \mathbf{W}_o^s \end{bmatrix} \mathbf{T}^{-1} \quad (6.6)$$

Dove * indica la matrice trasposta coniugata.

Per ottenere un sistema di ordine ridotto che tenga in conto i modi instabili, si suppone che questi siano in numero ridotto, ovvero, detto n_i il numero di autovalori instabili, $n_i = \mathcal{O}(10)$. In questo caso è possibile considerare questi modi in maniera non approssimata senza incorrere in un eccessivo costo computazionale. Supponendo inoltre di poter calcolare le basi Ψ_i e Φ_i degli autospazi instabili destro e sinistro, scalate in modo che $\Phi_i^* \Psi_i = \mathbf{I}_{n_i}$, si definisce la trasformazione

$$\mathbf{P} = \mathbf{I} - \Psi_i \Phi_i^* \quad (6.7)$$

che proietta lo stato \mathbf{x} sul sottospazio stabile della matrice \mathbf{A} . Detto quindi $\mathbf{x}_s = \mathbf{P}\mathbf{x}$, la dinamica è data da:

$$\frac{d}{dt}\mathbf{x}_s = \mathbf{A}\mathbf{x}_s + \mathbf{P}\mathbf{B}\mathbf{u} \quad (6.8)$$

$$\mathbf{y} = \mathbf{C}\mathbf{P}\mathbf{x}_s \quad (6.9)$$

mentre quella del sistema aggiunto (definito allo stesso modo di (4.21)):

$$\frac{d}{dt}\mathbf{z}_s = \mathbf{A}^*\mathbf{z}_s + \mathbf{P}^*\mathbf{C}^*\mathbf{y}_s \quad (6.10)$$

$$\mathbf{w}_s = \mathbf{B}^*\mathbf{P}^*\mathbf{z}_s \quad (6.11)$$

Siccome questi sistemi sono stabili, è possibile generare le matrici delle risposte agli impulsi \mathbf{C}_d e \mathbf{O}_d in maniera simile a quelle ricavate in precedenza. Si noti che nella sezione 4.2 il problema si era posto in tempo discreto, mentre qui in tempo continuo. Passare da uno all'altro è molto semplice, in Matlab sono sufficienti due righe di codice:

```

1   discr_sys = ss(A,B,C,0,st); % st = sample time
2   cont_sys = d2c(discr_sys);
    
```

Procedendo quindi come in precedenza è possibile calcolare una base per il modello di ordine ridotto r a partire dalla SVD di $\mathbf{O}_d\mathbf{C}_d$, ottenendo i modi diretti Ψ_s e i modi aggiunti Φ_s . Lo stato viene quindi espresso come:

$$\mathbf{x} = \Psi_i\mathbf{a}_i + \Psi_s\mathbf{a}_s \quad (6.12)$$

ed evolve secondo:

$$\begin{aligned} \frac{d}{dt}\mathbf{a} &= \frac{d}{dt} \begin{bmatrix} \mathbf{a}_i \\ \mathbf{a}_s \end{bmatrix} = \begin{bmatrix} \Phi_i^* \mathbf{A} \Psi_i & 0 \\ 0 & \Phi_s^* \mathbf{A} \Psi_s \end{bmatrix} \mathbf{a} + \begin{bmatrix} \Phi_i^* \\ \Phi_s^* \end{bmatrix} \mathbf{B}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}[\Psi_i\mathbf{a}_i + \Psi_s\mathbf{a}_s] \end{aligned} \quad (6.13)$$

Nel caso il numero di output del sistema renda intrattabile il computo delle risposte agli impulsi del sistema aggiunto è possibile operare similmente alla sezione 4.2.3 proiettando la parte stabile \mathbf{y} in un sottospazio dato dai primi r modi della POD del sistema diretto.

Detto infatti $\mathbf{x} = \mathbf{x}_i + \mathbf{x}_s$ dove $\mathbf{x}_s = \mathbf{P}\mathbf{x}$ e $\mathbf{x}_i = (\mathbf{I} - \mathbf{P})\mathbf{x}$, l'output può essere visto come $\mathbf{y} = \mathbf{y}_i + \mathbf{y}_s = \mathbf{C}(\mathbf{x}_i + \mathbf{x}_s)$. Chiamando Θ la matrice che contiene come colonne i primi r modi POD del sistema diretto, \mathbf{y}_s può essere proiettato ortogonalmente sul suo spazio della colonne, dando:

$$\mathbf{y} = \mathbf{C}\mathbf{x}_i + \Theta\Theta^*\mathbf{C}\mathbf{x}_s \quad (6.14)$$

che, una volta proiettato anche \mathbf{x} sul sottospazio di ordine ridotto, da come output:

$$\mathbf{y} = \mathbf{C}\Psi_i\mathbf{a}_i + \Theta\Theta^*\mathbf{C}\Psi_s\mathbf{a}_s \quad (6.15)$$

6.1.2 Risultati

L'obiettivo dello studio è stabilizzare il flusso all'aumentare dell'angolo di attacco tramite un controllatore che aziona un attuatore situato al bordo d'attacco della piastra. Per fare ciò, supponendo per semplicità che lo stato sia completamente accessibile, si opta per utilizzare un regolatore LQR, ovvero scegliendo il guadagno di retroazione \mathbf{K} che minimizzi

$$J = \frac{1}{2} \int_{t_0}^{\infty} (\mathbf{x}^* \mathbf{Q} \mathbf{x} + \mathbf{u}^* \mathbf{R} \mathbf{u}) d\tau \quad (6.16)$$

con $\mathbf{Q} = \mathbf{C}^* \mathbf{C}$ per catturare al meglio l'energia del sistema, ed $\mathbf{R} = c \mathbf{I}$ con $c = 10^5$ per evitare che il controllatore sia troppo aggressivo.

Viene innanzitutto calcolata la linearizzazione del sistema intorno allo stato stazionario. Si nota che l'angolo di attacco α si comporta da parametro di controllo e lo stato stazionario diventa instabile per $\alpha \approx 27^\circ$ (figura 6.1)

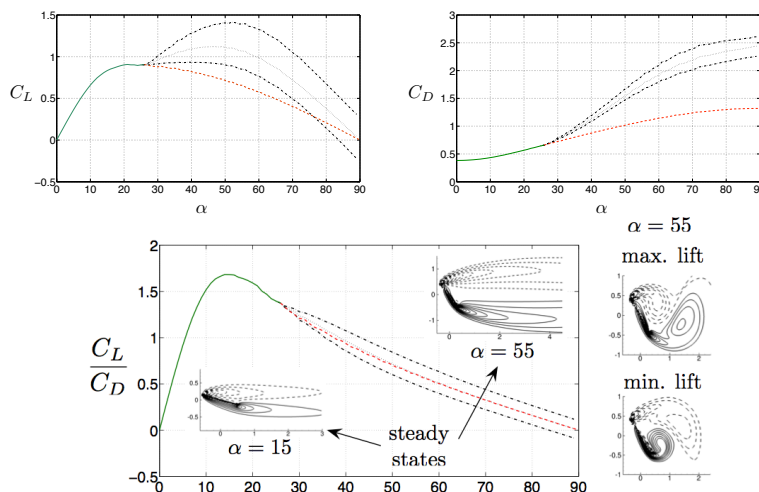


Figura 6.1: Visualizzazione dei coefficienti di sustentazione C_L e attrito C_D e dell'efficienza aerodinamica C_L/C_D al variare di α . (—): stato stazionario stabile. (---): stato stazionario instabile. (-.-.-): massimo e minimo durante il distacco di vortici. (-.-.-.-): valore medio durante il distacco di vortici. Sono inoltre mostrati alcuni profili di vorticità (valori negativi con linea tratteggiata) [1].

Dalla figura 6.1 si può chiaramente vedere come la generazione di vortici produca un significativo aumento della resistenza aerodinamica e una forte oscillazione nella portanza, cosa che rende desiderabile sopprimere questo stato. Siccome il sistema va incontro a una biforcazione di Hopf si genera una coppia di

autovalori instabili complessi coniugati, ovvero il sottospazio instabile ha dimensione 2. Inizializzando la simulazione con un leggero rumore è possibile eccitare i modi instabili, il che porta, dopo un tempo sufficientemente lungo, il sistema vicino al sottospazio instabile, in quanto i modi stabili decadono. Questo consente di calcolare gli autovettori destri Ψ_i . Analogamente, eccitando il sistema aggiunto, è possibile ottenere gli autovettori sinistri Φ_i .

Conoscendo tali vettori è possibile costruire la proiezione sul sottospazio stabile \mathbf{P} , e ottenere quindi un modello di ordine ridotto dei modi stabili. La matrice \mathcal{C}_d delle risposte impulsive è calcolata dal sistema:

$$\frac{d}{dt}\mathbf{x}_s = \mathbf{P}\mathbf{A}\mathbf{x}_s + \mathbf{P}\mathbf{B}\mathbf{u} \quad (6.17)$$

siccome il sottospazio stabile è un sottospazio invariante per la dinamica linearizzata la risposta agli impulsi di (6.9) e di (6.17) è teoricamente identica, ma errori numerici nel calcolo di \mathbf{P} porterebbero (6.9) a essere instabile nel lungo periodo in quanto la dinamica non è completamente ristretta al sottospazio stabile.

La matrice \mathcal{O}_d viene poi calcolata dalla risposta del sistema aggiunto:

$$\frac{d}{dt}\mathbf{z}_s = \mathbf{P}^*\mathbf{A}^*\mathbf{z}_s + \mathbf{P}^*\mathbf{C}^*\mathbf{y}_s \quad (6.18)$$

dove \mathbf{y}_s è la proiezione su 20 modi POD. È ora possibile calcolare il sistema di ordine ridotto. Si ha che i valori diagonali dei gramiani sono simili per un numero di modi ≤ 20 . Per controllare l'accuratezza del modello viene confrontata la risposta di (6.14) proiettata nel sottospazio stabile, ovvero con $\mathbf{a}_i = 0$, ad un input di tipo impulsivo. I risultati, mostrati in figura 6.2, mostrano come già con 10 modi il sistema ridotto riesca a riprodurre in maniera fedele la dinamica completa per il primo output, mentre l'undicesimo sia rappresentato in maniera accurata solo con 20 modi.

Per il disegno del controllatore si mantengono 12 modi (2 instabili e 10 stabili). L'output si approssima con una output projection con 4 modi. Il modello di ordine ridotto viene calcolato sottraendo lo stato stazionario \mathbf{x}_0 allo stato \mathbf{x} prima di incominciare l'algoritmo.

La figura 6.3 mostra che il modello di ordine ridotto cattura bene la dinamica del modello linearizzato. Il controllatore è in grado di smorzare in maniera adeguata le oscillazioni dovuti ad un impulso nell'attuatore.

Viene quindi effettuato il confronto con il modello non lineare completo. In figura 6.4 viene mostrato il comportamento del coefficiente di sostentazione. A partire da uno stato stazionario instabile piccoli errori numerici sono sufficienti a instaurare il regime di distacco periodico di vortici, dove C_L ha delle forti oscillazioni. Il controllatore, attivato per $t = 170, 180, 210$, è in grado di smorzare tali oscillazioni e riportare il sistema allo stato iniziale, anche a partire da uno stato che vi si discosta molto.

Si osservi che, per quanto la capacità del controllatore di smorzare le perturbazioni si estenda ben oltre la configurazione stazionaria intorno alla quale

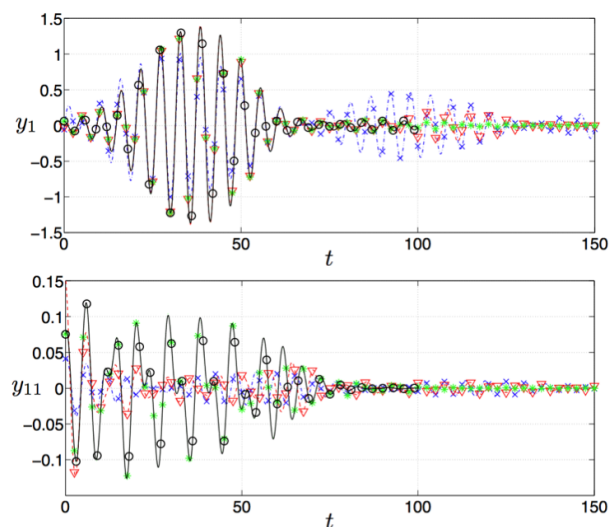


Figura 6.2: Confronto del primo e undicesimo output del modello completo (—) con quelli di modelli BPOD con 4 modi (---, ×), 10 modi (---, ▽) e 20 modi (....., *). I risultati sono ottenuti con una output projection su 20 modi [17].

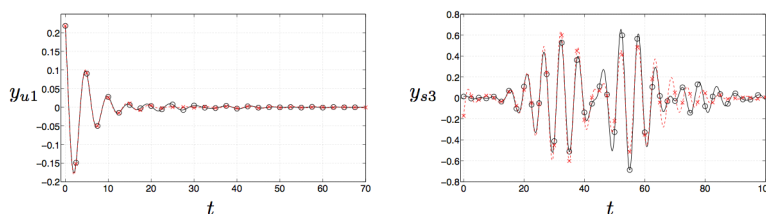


Figura 6.3: Comparazione del comportamento di un controllatore LQR, basato sul modello completo linearizzato (—, ○) e su un modello ridotto con 12 modi (---, ×). Condizione iniziale ottenuta a partire dalla risposta ad un impulso nell'attuatore [17].

è stato calcolato il modello di ordine ridotto, la predizione dell'output peggiori notevolmente dopo poche unità di tempo (figura 6.5). Se ne può dedurre che sia necessario, affinché il controllatore possa operare in maniera efficace, che la dinamica venga catturata precisamente solo in una breve finestra temporale in seguito all'accensione.

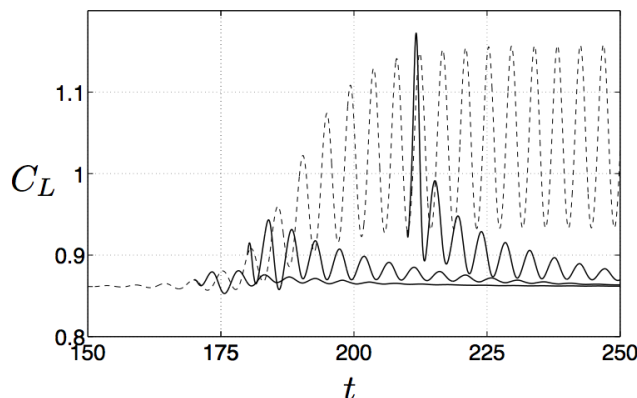


Figura 6.4: (---): modello completo non lineare non controllato. (—): comportamento del modello completo non lineare dopo l'accensione del controllatore ($t = 170, 180, 210$) [17].

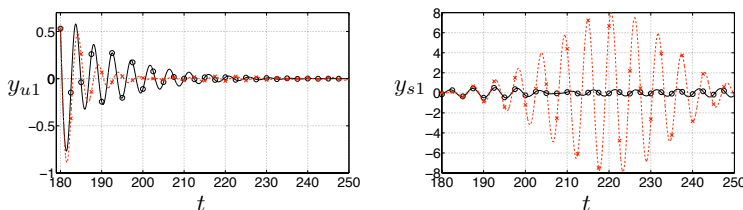


Figura 6.5: Output nel primo modo instabile e stabile per il modello completo (—, \circ) e ridotto (---, \times). La condizione iniziale è data dallo stato non controllato per $t = 180$ [17].

6.2 Ali a geometria variabile

Uno dei possibili passi avanti dell'industria dell'aviazione potrebbe essere rappresentato dall'introduzione di aeromobili dotati di ali in grado di cambiare forma in risposta alle condizioni esterne. Se si pensa al volo degli uccelli si intuisce immediatamente l'enorme potenzialità in termini di efficienza di simili tecnologie. Uno dei problemi dei quali soffrono aerei dotati di tale assetto è rappresentato dall'intrinseca flessibilità di tali strutture, e quindi dei possibili effetti aeroelastici indesiderati generati dalla loro interazione con l'aria. Nell'articolo di Fonzi et. al. [8] viene studiato un controllatore MPC in grado di mantenere le ali in un assetto ottimale sfruttando un modello aeroelastico calcolato con una variante della DMDC sviluppata per poter operare con equazioni differenziali algebriche. Viene inoltre discussa una tecnica di interpolazione di modelli diversi ottenuti per condizioni di volo differenti, in modo da poter esten-

dere la capacità predettiva del modello a una vasta gamma di condizioni di volo.

Nello studio viene utilizzato un modello Nastran della struttura, dalla quale vengono estratti i principali 8 modi di deformazione, ai quali vengono aggiunti 2 modi addizionali che caratterizzano il modo in cui le ali si trasformano per permettere il controllo del velivolo (il vettore contenente questi modi sarà denominato $\boldsymbol{\nu}$). A questi vengono aggiunti, come ulteriori componenti del vettore di stato non ridotto, l'intensità delle doppiette usate per calcolare l'intensità delle forze aerodinamiche nell'ambito della teoria linearizzata delle ali ad allungamento finito (vettore $\boldsymbol{\mu}$), il che da il vettore:

$$\boldsymbol{x} = [\boldsymbol{\nu} \quad \boldsymbol{\mu}]^T \quad (6.19)$$

il vettore di input è costruito come segue:

$$\boldsymbol{u} = [\alpha \quad p \quad q \quad r \quad F_1 \quad \dots \quad F_k]^T \quad (6.20)$$

dove α è l'angolo d'attacco, p , q ed r sono le velocità angolari di rollio, beccheggio e imbardata, le F_i sono le forze degli attuatori, normalizzate.

6.2.1 Algebraic DMD with control e modelli ridotti parametrici

Siccome la forma generalizzata delle equazioni che governano i fenomeni aeroelastici può essere scritta nella forma:

$$\boldsymbol{f}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}) = \boldsymbol{q}(\boldsymbol{x}_k, \boldsymbol{u}_k) \quad (6.21)$$

è necessario che l'algoritmo DMDc tenga conto di questa struttura, ovvero si cerca un modello nella forma:

$$\boldsymbol{z}_{k+1} = \tilde{\boldsymbol{A}}\boldsymbol{z}_k + \tilde{\boldsymbol{B}}\boldsymbol{u}_k + \tilde{\boldsymbol{F}}\boldsymbol{u}_{k+1} \quad (6.22)$$

L'algoritmo DMDc modificato prende il nome di algebraic Dynamic Mode Decomposition with control (aDMDc). A differenza dell'algoritmo presentato nella sezione 5.2, qui si contruisce una seconda matrice di fermo-immagine per poter modellizzare l'evoluzione temporale degli input:

$$\boldsymbol{\Upsilon}' = \begin{bmatrix} | & | & \dots & | \\ \boldsymbol{u}_2 & \boldsymbol{u}_3 & \dots & \boldsymbol{u}_m \\ | & | & & | \end{bmatrix} \quad (6.23)$$

e le matrici \boldsymbol{G} e $\boldsymbol{\Omega}$ vengono modificate come segue:

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{B} \\ \boldsymbol{F} \end{bmatrix}, \quad \boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{\Upsilon} \\ \boldsymbol{\Upsilon}' \end{bmatrix} \quad (6.24)$$

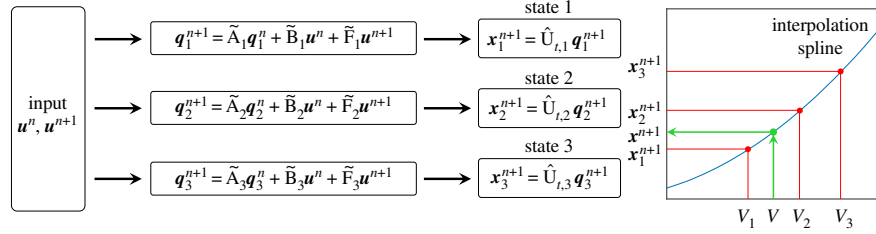


Figura 6.6: Schema grafico del processo di interpolazione di ROM ottenuti a differenti velocità di volo [8].

e, procedendo come già fatto in precedenza, da come risultato le matrici:

$$\tilde{\mathbf{A}} = \hat{\mathbf{U}}^* \mathbf{X}' \mathbf{V} \boldsymbol{\Sigma}^{-1} \mathbf{U}_1^* \hat{\mathbf{U}} \quad (6.25)$$

$$\tilde{\mathbf{B}} = \hat{\mathbf{U}}^* \mathbf{X}' \mathbf{V} \boldsymbol{\Sigma}^{-1} \mathbf{U}_2^* \quad (6.26)$$

$$\tilde{\mathbf{F}} = \hat{\mathbf{U}}^* \mathbf{X}' \mathbf{V} \boldsymbol{\Sigma}^{-1} \mathbf{U}_3^* \quad (6.27)$$

Modelli diversi vengono generati per un range di velocità fra i 35 e gli 80 ms^{-1} . I vari ROM vengono poi interpolati con un metodo spline, a partire dalla variabile di ordine ridotto \mathbf{z} . Data l'elevata dimensionalità del sistema, interpolare lo stato completo porterebbe a modelli in generale instabili.

6.2.2 Risultati

Per generare i dati su cui allenare il modello si utilizzano simulazioni del sistema completo eccitato con impulsi casuali nelle variabili di controllo come input. I test vengono effettuati confrontando la risposta del modello completo e del modello generato con la aDMDC a un input sinusoidale e di tipo impulsivo. I risultati per la singola velocità di 50 ms^{-1} e per la versione interpolata sono mostrati nelle figure 6.7 e 6.8.

Si può notare come il ROM segua in maniera quasi ottimale il modello completo, con una riduzione del tempo di calcolo di 2894 per il modello a una singola velocità e di 1338 per il modello parametrico.

L'obiettivo rimane quello di controllare tale velivolo in modo che possa compiere la missione mostrata nella figura. A tale scopo il modello di ordine ridotto viene impiegato all'interno di un controllore MPC, sfruttando, per semplicità, l'algoritmo interior point `fmincon` codificato in Matlab per la routine di ottimizzazione. Il controllore viene testato in tre scenari: alleggerimento dei carichi causati da una raffica di vento, seguimento di un coefficiente di sustentazione e infine una possibile missione che comprenda sia il seguimento che l'alleggerimento dei carichi. In tutti e tre i casi si considera disponibile per la retroazione l'intero stato attuale, eliminando quindi la necessità di disegnare un osservatore

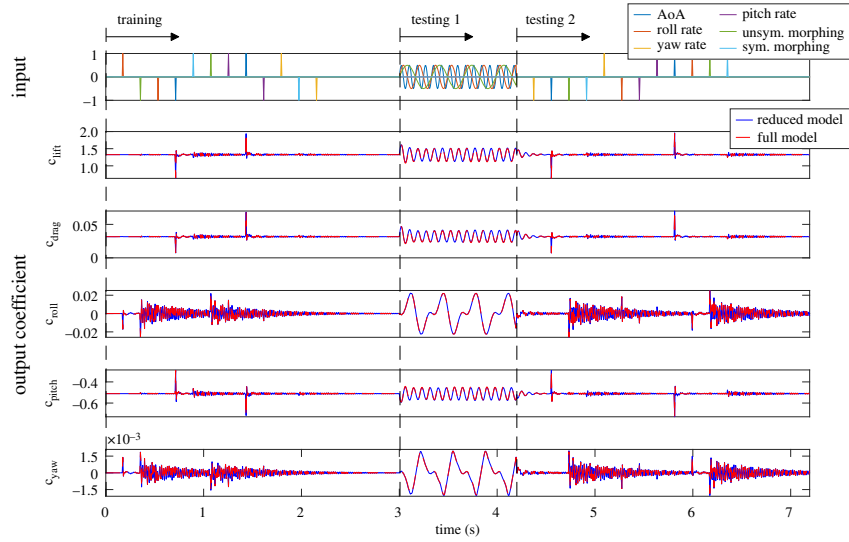


Figura 6.7: Comparazione fra la risposta del modello di ordine ridotto (in blu) e il modello completo (in rosso) alla velocità di 50 ms^{-1} [8].

adeguato.

Si considera un passo di tempo del sistema $\Delta t_{sys} = 0.006 \text{ s}$ e per il controllore $\Delta t_{MPC} = 0.018 \text{ s}$. L'azione degli attuatori è limitata a $u \in [-1, 1]$, $\Delta u \in [-0.18, 0.18]$, e gli orizzonti di controllo e predizione sono posti uguali e pari a 10. Siccome il rischio maggiore per la struttura è rappresentato da una deformazione dell'ala nel primo modo di deformazione, questo valore è usato per calcolare l'errore nei vari casi. Il valore di tale parametro può essere stimato attraverso dei sensori di deformazione posti alla radice dell'ala, ma viene considerato completamente disponibile ai fini di questo studio. I risultati sono mostrati nelle figure 6.9, 6.10 e 6.11.

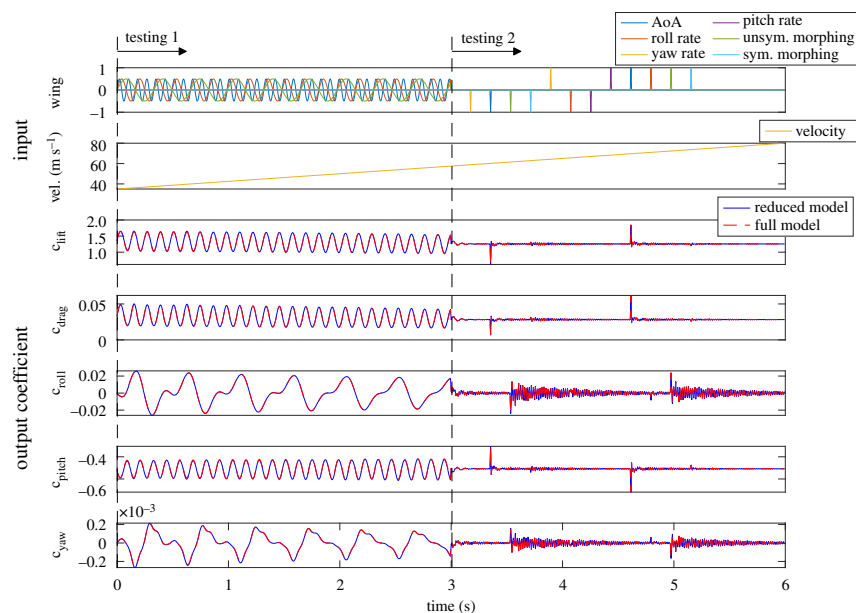


Figura 6.8: Comparazione fra la risposta del modello di ordine ridotto (in blu) e il modello completo (in rosso) in un range di velocità $35\text{-}80\text{ ms}^{-1}$ [8].

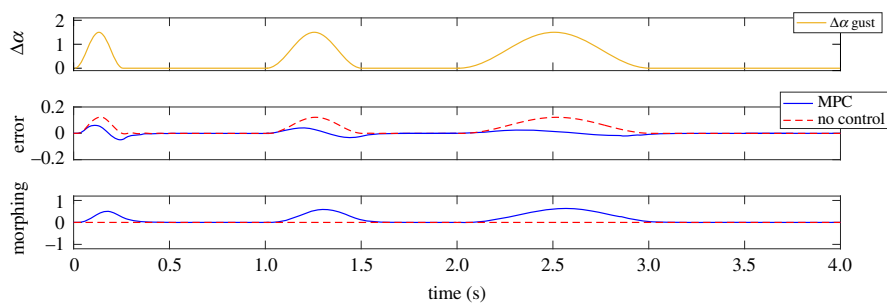


Figura 6.9: Risultati per l'alleviamento dei carichi dovuti a raffiche con $\Delta\alpha = 1.5^\circ$ e $t_r = [0.25\text{ s}, 0.5\text{ s}, 1\text{ s}]$, volando a velocità costante $V = 60\text{ ms}^{-1}$. Confronto fra il modello non controllato e controllato, con matrici dei pesi $Q = 10$, $R_u = 0$, $R_{\Delta u} = 1$ [8].

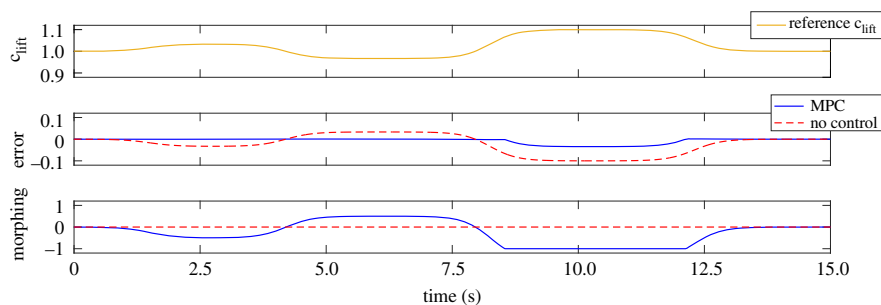


Figura 6.10: Risultati nel seguimento di un coefficiente di sustentazione dato. Pesì: $Q = 1$, $R_u = 0$, $R_{\Delta u} = 10,000$ [8].

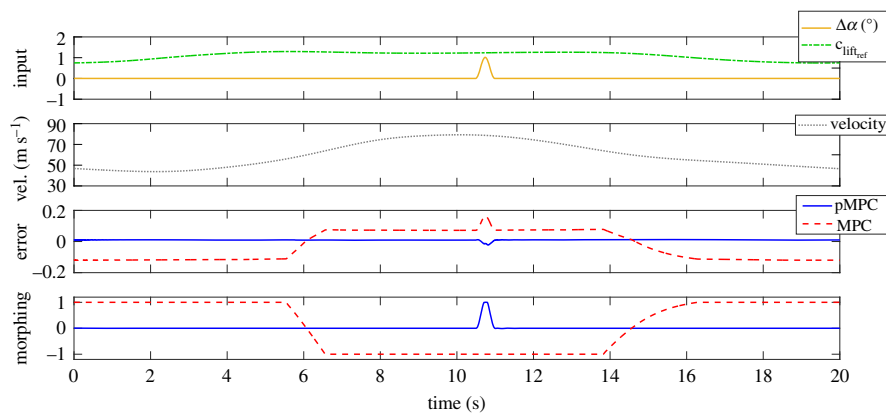


Figura 6.11: Comparazione del modello parametrico interpolato (pMPC) e modello non interpolato (MPC) in una missione con seguimento e alleggerimento dei carichi. $V \in [40\text{ms}^{-1}, 80\text{ms}^{-1}]$, raffica $\Delta\alpha = 1^\circ$, $t_r = 0.5$ s [8].

Capitolo 7

Conclusioni

In questo lavoro si sono studiate alcune delle tecniche di riduzione della dimensionalità più usate in ambito fluidodinamico. L'enfasi posta sul problema del controllo in tempo reale mostra in maniera chiara la necessità e l'efficacia di tali tecniche.

Il grande interesse che algoritmi come la POD o la DMD stanno suscitando deriva dalla sempre maggiore disponibilità di biblioteche di dati presi da esperimenti e simulazioni che consentono di costruire e validare modelli efficaci, superando in capacità predittiva e velocità di calcolo tecniche meno specifiche come l'analisi di Fourier. Questa infatti, non basandosi sulle informazioni disponibili sul sistema in esame, non garantisce di dare una base ottimale per rappresentarne la dinamica. Sebbene sia stata studiata in gran dettaglio e possa essere usata per ridurre le dimensionalità di insiemi di dati di grandi dimensioni (come nella compressione delle immagini [3]), in contesti come quelli analizzati in queste pagine non fornisce un guadagno in termini di costo computazionale tale da giustificare l'utilizzo al posto di tecniche basate sui dati.

POD e DMD sono tecniche basilari che si prestano a numerose estensioni per adattarsi a specifici casi di studio, ma non sono le uniche utilizzate e studiate in letteratura. L'algoritmo ERA (Eigensystem Realization Algorithm), ad esempio, può essere visto come un'estensione della BPOD che non richiede la simulazione del sistema aggiunto, ed è quindi utilizzabile anche per dati derivanti da contesti sperimentali [18]. Risultano particolarmente promettenti anche modelli calcolati attraverso tecniche di machine learning [4]. Questi ultimi modelli hanno il grande pregio di essere spesso molto precisi e di dare una rappresentazione del sistema in esame molto accurata se questo non si discosta eccessivamente dall'insieme di dati usati per l'allenamento, ma non sempre risultano facilmente interpretabili. Alcuni studi recenti puntano a combinare il machine learning con altre tecniche per superare questa limitazione, ad esempio affiancandoci modelli POD, che hanno il pregio di essere facilmente interpretabili e di rango basso [12].

Bibliografia

- [1] S. AHUJA e C. W. ROWLEY. «Feedback control of unstable steady states of flow past a flat plate using reduced-order estimators». In: *Journal of Fluid Mechanics* 645 (2010), pp. 447–478. DOI: 10.1017/S0022112009992655.
- [2] J. M. Birch e M. H. Dickinson. «Spanwise flow and the attachment of the leading-edge vortex on insect wings». In: *Nature* 412.6848 (2001), pp. 729–733. ISSN: 00280836. DOI: 10.1038/35089071.
- [3] Steven L. Brunton e Nathan Kuts. *Data driven science and engineering*. 2017.
- [4] Steven L. Brunton e Bernd R. Noack. «Closed-Loop Turbulence Control: Progress and Challenges». In: *Applied Mechanics Reviews* 67.5 (ago. 2015). 050801. ISSN: 0003-6900. DOI: 10.1115/1.4031175. eprint: https://asmedigitalcollection.asme.org/appliedmechanicsreviews/article-pdf/67/5/050801/6074501/amr_067_05_050801.pdf. URL: <https://doi.org/10.1115/1.4031175>.
- [5] Eduardo F. Camacho e Carlos Bordons. «CONTROL PREDICTIVO: PASADO, PRESENTE Y FUTURO». In: *Revista Iberoamericana de Automática e Informática Industrial* (2010).
- [6] Charbel Farhat. *AA216/CME345: Model Reduction. Projection-based Model Order Reduction*.
- [7] Charles L. Fefferman. *Existence and smoothness of the Navier-Stokes equation*. URL: <http://www.claymath.org/millennium-problems/navier%E2%80%93stokes-equation>. (consultato il 17/06/2021).
- [8] N. Fonzi, S. L. Brunton e U. Fasel. «Data-driven nonlinear aeroelastic models of morphing wings for control». In: *Proc. R. Soc. A* (2020). URL: <http://dx.doi.org/10.1098/rspa.2020.0079>.
- [9] Robert L. Williams II e Douglas A. Lawrence. *Linear State-Space Control Systems*. A cura di John Wiley & Sons. ISBN: 978-0-471-73555-7.
- [10] Nathan Kuts et al. *Dynamic Mode Decomposition. Data-Driven Modeling of Complex Systems*. A cura di SIAM. 2016.
- [11] «Model predictive control: past, present and future». In: *Computers and Chemical Engineering* (1999).

-
- [12] Rem Sophia Mouradi et al. «Physically interpretable machine learning algorithm on multidimensional non-linear fields». In: *Journal of Computational Physics* 428 (2021), p. 110074. ISSN: 10902716. DOI: 10.1016/j.jcp.2020.110074. arXiv: 2005.13912. URL: <https://doi.org/10.1016/j.jcp.2020.110074>.
- [13] Jorge Nocedal e Stephen J. Wright. *Numerical Optimization*. A cura di Springer. Cap. 16.
- [14] *Place, pole placement design*. URL: https://www.mathworks.com/help/control/ref/place.html?s_tid=doc_ta#bq1no8b-1. (consultato il 08/06/2021).
- [15] Joshua L. Proctor, Steven L. Brunton e J. Nathan Kutz. «Dynamic Mode Decomposition with Control». In: *SIAM J. on Applied Dynamical Systems* 15 (2016), pp. 142–161. DOI: 10.1137/15M1013857.
- [16] Joshua L. Proctor, Steven L. Brunton e Nathan J. Kutz. «Generalizing Koopman Theory to Allow for Inputs and Control». In: *SIAM J. Applied Dynamical Systems* (2018).
- [17] C. W. Rowley. «Model reduction for fluids, using balanced proper orthogonal decomposition». In: *Int. J. on Bifurcation and Chaos*. (2005).
- [18] Clarence W. Rowley e Scott T.M. Dawson. «Model Reduction for Flow Analysis and Control». In: *Annual Review of Fluid Mechanics* 49.1 (2017), pp. 387–417. DOI: 10.1146/annurev-fluid-010816-060042. eprint: <https://doi.org/10.1146/annurev-fluid-010816-060042>. URL: <https://doi.org/10.1146/annurev-fluid-010816-060042>.
- [19] Clarence W. Rowley et al. «Spectral analysis of nonlinear flows». In: *J. Fluid Mech.* (2009).
- [20] Gregory Salomon, Kemin Zhou e Eva Wu. «A new balanced realization and model reduction method for unstable systems». In: *IFAC Proceedings Volumes* 32.2 (1999), pp. 1696–1701. ISSN: 14746670. DOI: 10.1016/S1474-6670(17)56288-6.
- [21] S. Volkwein. *Model Reduction Using Proper Orthogonal Decomposition*. 2011.