Identificazione termodinamica di un edificio con posizionamento ottimo di reti di sensori wireless

Bonollo Federico, Cattai Daniele, Verga Marco

***Abstract*— Una delle applicazioni più' comuni per reti di sensori wireless e' quella del monitoraggio di quantità di interesse quali temperature, livelli di sostanze biochimiche, etc. Spesso questi sensori sono disposti in modo casuale e in soprannumero in termini di qualità di stima del campo di interesse. Nel caso di un edificio e al monitoraggio della temperatura interna, in linea di principio, sarebbe possibile ottenere delle stime di temperatura con una accuratezza elevata tramite l'utilizzo di un modello dinamico di stato in cui solamente poche misure sono necessarie per ricostruirlo. In questa trattazione si studiano due particolari tecniche per la riduzione del numero di sensori: PCR e PLS, rispettivamente *Principal Component Regression* e *Partial Least Squares*. Queste due metodologie di selezione vengono inserite in un contesto di identificazione a sottospazi che permette di ricavare un modello dell'edificio in questione. Alla loro applicazione su dati rilevati sperimentalmente, vengono affiancate delle simulazioni effettuate utilizzando il software *DesignBuilder* con lo scopo di fornire diversi set di esperimenti su qui creare e testare il modello, oltre che avere un qualitativo confronto con le tecniche sopracitate**

***Index Terms*— identificazione a sottospazi, PCA, PCR, PLS, DesignBuilder.**

I. Introduzione

NEGLI ultimi anni l'interesse verso sistemi abitativi e industriali ecocompatibili é andato via via crescendo in relazione alla stessa crescita della sensibilitá ambientale. Il motore principale di questo crescente interesse é il massiccio utilizzo di risorse energetiche non rinnovabili e del conseguente impatto ambientale che a livello globale si é creato. É nata dunque l'esigenza di coordinare le ricerche in più ambiti per ottenere strategie di utilizzo energetico valide e nel rispetto della salvaguardia dell'ambiente. Purtroppo a livello domestico si ha una scarsa efficienza nell'utilizzo delle risorse energetiche in particolare, il riscaldamento e il condizionamento, che nelle abitazioni risentono spesso dell'efficienza strutturale dell'edificio.

In tale contesto un'efficace identificazione del modello termodinamico si rivela utile per un mirato controllo dei parametri fisici. Ciò permette inoltre di testare e classificare il rendimento termodinamico dell'edificio.

Questa trattazione si focalizza nel determinare attraverso misure reali e simulative, un modello per l'evoluzione della temperatura nello spazio e nel tempo dell'edificio. Conoscendo tale modello si vuole verificare e fornire un'idea qualitativa sul posizionamento ottimo dei sensori per un futuro controllo o classificazione dell'edificio, applicando i risultati ottenuti anche ad altri contesti data la grande versatilità delle reti di sensori wireless. Per la tematica della classificazione, a livello europeo esiste giá una normativa in materia la 2002/91/EC per l'introduzione della classe energetica anche al settore edilizio. A tale scopo giá diverse aziende e imprese di costruzioni si sono rivolte alla ricerca di metodologie e software per la determinazione di tale classe, e una di queste viene descritta e sfruttata in questo progetto, ossia $DesignBuilder$. Il progetto cercherà inoltre di migliorare e comparare i risultati ottenuti nel lavoro [13]. In tale lavoro la selezione di ingressi/uscite significativi, é stata effettuata utilizzando algoritmi di tipo Greedy Search, e Genetic Search. Mentre con l'uso delle tecniche di selezione PCR e PLS, é possibile constatare un effettivo miglioramento nella bontá e consistenza del modello identificato, permettendo ampie conclusioni. Per quanto riguarda l'identificazione vera e propria sono state utilizzate tecniche a sottospazi del tipo: $N4SID$, $PBSID$ e una versione ottimizzata e gentilmente fornita dal prof. A.Chiuso, denominata $PBSIDopt$. La teoria a sottospazi é stata ampiamente sviluppata in letteratura e vi sono algoritmi, tra cui quelli sopracitati, che forniscono risultati robusti. Si é studiata l'identificazione a sottospazi in maniera approfondita quanto bastava per poter intervenire sul problema, descrivendone alcuni dettagli e basi teoriche avanzate.

Per quel che riguarda la selezione di ingressi ed uscite, in letteratura esiste una folta teoria ma poche sono le soluzioni algoritmiche implementate in maniera robusta. Si è cercato quindi di sviluppare un sistema di selezione basato su PCR e uno utilizzante una versione dinamica di PLS. Per la verifica della bontà di selezione e del relativo modello identificato si sono sviluppati diversi criteri di validazione.

II. I Simulatori

A. Introduzione

A seguito dell'introduzione della certificazione energetica degli edifici da parte della Comunitá Europea allo scopo di ridurne i consumi e l'inquinamento, molti software sono stati sviluppati per calcolarne l'*EPB*, ossia *"l'energy performance of buildings"*. Essi considerano nel calcolo svariati parametri in condizioni statiche o dinamiche, il primo dei quali é la posizione geografica. Nel Nord d'Europa il clima é freddo durante gran parte dell'anno, per cui l'EPB dipende essenzialmente dall'energia del calore disperso. Di conseguenza la progettazione degli edifici richiede un cospicuo isolamento termico e soluzioni adatte alla massima cattura dell'energia solare per il relativo sfruttamento. Viceversa nei paesi molto piú a sud, dove il clima é caldo e umido, le soluzioni tecniche richieste al fine di massimizzare la dispersione del calore sono:

- ventilazione passiva,

- progetti di climatizzazione e sfruttamento dell'inerzia termica.

Al centro di queste due filosofie di progettazione, si inserisce l'area mediterranea caratterizzata da andamenti climatici intermedi, tali da rendere necessaria una maggiore flessibilitá delle soluzioni tecniche, tradotta nella realtá in una diversificazione dell'architettura storica di questi paesaggi.

Lo sviluppo delle tecnologie costruttive agli inizi del 20°secolo non prendevano in considerazione il carattere energetico degli edifici. Solo a partire dagl'anni '50 sono stati introdotti i primi accorgimenti nell'ambito, per esempio:

- riduzione dell'inerzia termica coibentando gli edifici,
- riduzione dello spessore dei muri.

Queste prime soluzioni, peró, si sono dimostrate subito insufficienti con l'avvento della crisi economica del 1973 e con l'adozione di nuove politiche climatiche. Durante gli anni '80 l'Europa a tale proposito ha prodotto una serie di standard costruttivi fino alla recente EN 832. Nello stesso periodo la societá ha subito un forte sviluppo economico, e una seguente modifica della percezione del comfort, sia abitativo che lavorativo.
Purtroppo questo sviluppo nel benessere abitativo unito ad altri fattori, ha causato l'incremento dei consumi di energia nel settore edilizio, specialmente nell'area mediterranea dove si utilizza molta energia sia per riscaldare durante l'inverno, sia per il raffreddamento d'estate. La Comunitá Europea stima che piú del 40% del consumo di energia sia attribuibile agli edifici. Per una riduzione e una maggiore razionalizzazione delle risorse, sono state emanate leggi e normative all'interno del direttivo 91/2002 EPBD *"Energy Performance of Buildings Directive"*. Le normative tecniche collegate al EPBD sono state sviluppate dal C.E.N.(*"European Committee for Standardization"*) e vanno sotto il nome di CEN-Umbrella. L'EPBD introduce l'obbligo della certificazione energetica degli edifici, il cui approccio é molto simile alle etichette di classe energetica delle apparecchiature elettriche. La CEN-Umbrella include una revisione della EN 832, della EN 13790, introducendo nuovi paramteri di valutazione:

- l'energia richiesta durante l'inverno e l'estate per: riscaldamento, raffreddamento e la ventilazione.
- la produzione di acqua calda ad uso domestico (DHW-"'Domestic Hot Water"').
- consumi energia elettrica.

In tale contesto i software e i modelli per il calcolo dell'EPB ricoprono un valore importante ai fini di una corretta identificazione della classe, potendo contribuire anche alla ricerca della migliore soluzione per incrementare l'efficienza energetica. Teoricamente si dovrebbe garantire una forte corrispondenza tra i dati simulati e quelli reali, per permettere un altrettanto miglioramento nella fase progettuale e migliorandone il controllo della stessa efficienza anche da parte dell'utente. Evidentemente questo si scontra con la richiesta di software con capacitá globali, ossia corrette per diverse aree climatiche e caratteristiche costruttive, e robusti all'aumentare della complessitá dei fattori e delle variabili in gioco che non sono comparabili e standardizzabili.

B. Modelli di calcolo e riferimenti standard del simulatore

Il metodo di calcolo utilizzato dal software DesignBuilder, e adottato nelle simulazioni in questo progetto, segue i riferimenti CEN con gli standard: prEN 13790, prEN 15603 e relative normative sugli impianti. Si tenga presente che DesignB. non corrisponde perfettamente allo standard CEN-Umbrella per la mancanza di aggiornamenti su vai coefficienti e nell'utilizzo fonti di energia rinnovabili.

1) Descrizione calcoli: Il software per l'energia termica usa la formula dettata dagli standard europei.

$$Q_{H,nd} = Q_{L,H} - \eta_{g,H} Q_{G,H} \quad (1)$$

Dove $Q_{H,nd}$ é l'energia richiesta per il riscaldamento dell'edificio (in MJ), $Q_{L,H}$ é la quantitá di calore totale trasferito e $Q_{G,H}$ é il calore totale fornito dalle risorse energetiche, infine $\eta_{g,H}$ é il fattore di guadagno di utilizzazione adimensionale. Per il raffreddamento la formula diviene:

$$Q_{C,nd} = Q_{G,C} - \eta_{l,C} Q_{L,C} \quad (2)$$

Dove $Q_{C,nd}$ é l'energia richiesta per il raffrescamento dell'edificio (in MJ), $Q_{G,C}$ é la quantitá di raffreddamento totale trasferito e $Q_{L,C}$ é il calore totale fornito dalle risorse energetiche, infine $\eta_{g,H}$ é il fattore di guadagno di utilizzazione adimensionale. Nelle equazioni 1, 2 le lettere H e C fanno riferimento rispettivamente alle modalitá di riscaldamento e climatizzazione, mentre i rispettivi addendi per entrambe le equazioni si scompongono in:

$$Q_L = Q_{tr} + Q_{ve} \quad (3)$$

$$Q_G = Q_{int} + Q_{sol} \quad (4)$$

Dove il calore totale trasferito Q_L é composto da Q_{tr} calore trasferito per trasmissione, dipendente dal valore della trasmittanza per ogni componente e Q_{ve} per la ventilazione. In analogia le totali fonti di calore Q_G per ogni funzionamento, sono espresse come la somma di tutte le fonti di calore interne sul periodo Q_{int} e la somma delle fonti solari sempre sul periodo Q_{sol}, quest'ultimo dipendente dai dati meteorologici. La Q_{int} é ottenuta con metodi stocastici, non comparabile con i reali consumi. Questo valore include tutta l'energia derivante da cucine, luce, consumi elettrici e generali dell'unitá.

$$Q_{int} = \sum_k Q_{int,k} + \sum_l (1 - b_l)\phi_{int,u,l} \quad (5)$$

Con $Q_{int,k}$ il rate orario di flusso di calore dalla k-esima fonte di calore interna (computer, luci artificiali, apparecchi domestici,.. sia nel caso reale che da tabelle normalizzate), $\phi_{int,u,l}$ rate orario di flusso di calore dalla l-esima fonte di calore interna nello spazio incondizionato W, infine con $b_l(W)$ il relativo fattore di riduzione (ISO 13789). Il calore disperso e la fornitura dello stesso, sono calcolate in relazione con le variazioni della temperatura esterna e con l'irraggiamento

medio durante tutto l'anno, basato sui dati meteorologici. La valutazione per il calcolo dei consumi DHW sfruttano la formula:

$$Q_{h,w} = V_w \rho C n (t_w - t_0) \quad (6)$$

dove V é il volume di DHW consumato, ρ la densitá dell'acqua, C il calore specifico dell'acqua (4186 J/Kg K), t_W la temperatura dell'acqua in uscita e t_0 quella in ingresso. Si tenga presente che l'uso di tutte queste standardizzazioni (DHW, luce, energia per cucinare, consumo elettrico e ventilazione naturale) rappresentano un punto molto critico per la modelizzazione e il calcolo. Questo perché sono innanzitutto ideali, estremamente variabili, e non sono completamente valide per la normativa italiana (Appendice II). DHW (*Domestic hot water*) consumata dipende da molteplici fattori: destinazione d'uso, numero di utenti, numero di utenze finali. Le altre udm si riassumono in guadagni di energia interna ottenuti dai computer, apparecchi domestici, usi del gas per la cucina standardizzati con il parametro W/m^2. Una nota va' ai valori dei fattori di ventilazione naturale che differiscono anch'essi dalla normativa italiana.

C. DesignBuilder

Il mondo dell'edilizia stà rapidamente evolvendo verso un'industrializzazione sia della fase costruttiva dei componenti che della costruzione degli edifici nei cantieri. La disponibilitá di nuovi strumenti software per la gestione ottimale di questa spinta evolutiva ha in effetti migliorato la qualitá complessiva delle costruzioni e ridotto le tempistiche nei cantieri. Non tutto peró é ancora stato ottimizzato anche perché, nel mondo delle costruzioni, operano diverse discipline non sempre fra loro omogenee. La differenza maggiore di impostazione riguarda il mondo degli impianti, la cui importanza, alla luce dei costi energetici, stà crescendo in modo esponenziale. Tanto che la qualitá energetica di un edificio sará l'elemento discriminante, a breve, nelle compravendite di immobili. In questa integrazione si inserisce il software DesignBuilder suggerito per le simulazioni dell'edificio di questo progetto. DesignBuilder nasce da un progetto europeo e si propone come uno strumento di grande qualitá per analisi energetiche serie e attendibili di edifici semplici o estremamente complessi. É un software che unisce alla rapiditá e alla facilitá di rappresentazione tridimensionale dell'edificio, l'analisi energetica del sistema Edificio/Impianto che si basa sul motore di simulazione energetica "'Energyplus"' [1]. Questo software fornisce una serie di dati di rendimento ambientali come: consumo annuo di energia, temperature massime estive e potenze dei componenti *HVAC (Heating Ventilation and Air Conditioning)*. Il software puó essere utilizzato efficacemente in qualunque fase del processo di disegno dalla fase di concetto dove si puó voler provare svariate tipologie di strutture in ogni singola stanza, oppure attraverso tutta la fase di realizzazione. L'utilizzo di "'Template"' e dati di "'Default"' permettono di guidare l'utente attraverso il processo di inserimento dei dati, delle analisi e delle varianti in corso d'opera. I modelli di dati (Template) permettono all'utente di caricare geometrie, strutture e impianti di edifici tipici, ottenendo immediatamente calcoli attendibili per valutazioni di massima. Successivamente si puó passare a fasi di dettaglio, modificando opportunamente i singoli dati, per definire al meglio i valori di potenza e di consumi dello specifico progetto. É inoltre possibile aggiungere i propri modelli se si lavora spesso su tipologie di edifici simili. L'interazione con la vista tridimensionale renderizzata permette all'utente di capire immediatamente la differenza tra architettura e altri dati pratici come il surriscaldamento di porzioni non climatizzate dell'edificio. É dunque adatto all'utilizzo di architetti, ingegneri edili, termotecnici e studenti.

[1]EnergyPlus é il motore di calcolo energetico sviluppato dal ministero degli Stati Uniti(DOE) a seguito della prima crisi energetica del 1974. Nato alla Berkely University continua lo sviluppo anche in organizzazioni scientifiche e tecniche come la ASHARE che ne valida le procedure di calcolo tramite un apposito protocollo internazionale

III. L'IMPOSTAZIONE DELL'ESPERIMENTO

Tenendo in considerazione il lavoro svolto lo scorso anno [13], e con l'obiettivo di comparare i risultati ottenuti si sono utilizzati come dati sperimentali quelli raccolti in [13]. Si ritiene utile tuttavia fornire indicazioni su come impostare al meglio la fase di test e raccolta dati sul campo, in modo da permettere una successiva fase di identificazione che sia la piú efficace possibile. É possibile definire in maniera adatta le modalitá e il set di dati da raccogliere nell'esperimento principale [6].
Le fasi preliminari si possono distinguere in:

- **Tempo invarianza** Molti metodi di identificazione assumono modelli tempo-invarianti e gli esperimenti/misure riescono a determinare tale proprietá. Peró é possibile che vi sia una valutazione errata. Infatti delle dinamiche lente appaiono come dinamiche tempo varianti se consideriamo una scala dei tempi troppo corta. É necessario qualora si voglia identificare delle dinamiche veloci, eliminare quelle lente assimilandole a trend.
- **Linearitá** In Matlab, il toolbox di identificazione contiene routine per identificare solo sistemi dinamici. La linearitá puó essere verificata attraverso l'analisi delle risposte del sistema a differenti ampiezze degli ingressi, con la simmetria della risposta ad un'onda quadra o derivando proprietá spettrali.
  Essendo molti sistemi non lineari, é comunque possibile, sotto certe ipotesi, ricavarsi la linearizzazione del modello.
- **Transitori** L'analisi dei transitori attraverso risposte all'impulso o al gradino, fornisce valutazioni su constanti di tempo, ritardi, guadagni stazionari.
- **Risposta in frequenza** Sono metodologie relegate alla stima non parametrica, in comune con l'analisi delle correlazioni. Esse forniscono la stima della funzione di trasferimento del sistema. In tale contesto il diagramma di Bode si rivela utile per le informazioni sulla locazione di poli e zeri, guadagni e ritardi.

Per quanto riguarda la parte principale dell'esperimento, si sa' che l'identificazione fornisce accurati modelli alle frequenze dove il segnale di ingresso contiene maggiore energia.

Questo si traduce in una sufficiente eccitazione dell'ingresso a tali frequenze. Allora é naturale cercare di concentrare le frequenze del segnale di ingresso in modo tale da rendere piccolo l'errore di stima dove ció si rende piú necessario. Una scelta comune é l'ingresso PRBS *pseudo-random binary sequence* che ha come caratteristica di avere molta energia su uno spettro ampio. Se questa scelta non fosse possibile, come in questo caso di ingresso esterno, é auspicabile avere un segnale con un buon rapporto segnale/rumore e una condizione di persistente eccitazione sufficiente ad una buona identificazione.

IV. L'IDENTIFICAZIONE

Per quanto riguarda l'identificazione di modelli a variabili di stato, il procedimento ideale che si basa sulla fattorizzazione della matrice di Hankel, raramente porta a risultati soddisfacenti nelle applicazioni con dati reali. Questo tipo di identificazione comporta generalmente grossolani errori di stima relazionati a:

- non è possibile disporre di dati esatti, conseguenza di disturbi nelle misure (errori di misura),
- inevitabili arrotondamenti nelle elaborazioni numeriche (approssimazioni numeriche),
- il modello lineare a dimensioni finite ipotizzato per spiegare i dati, si dimostra inadatto a descrivere la realtà (semplificazioni di modello).

Questa serie di cause comporta un aumento di rango della matrice di Hankel, e con esso una diretta sovrastima del modello a fronte di un complessità reale ben minore ([5] p.132]). Questo cruciale problema trova un adeguato rimedio nella scomposizione matriciale di Beltrami, nota come scomposizione ai valori singolari (SVD). Si scompone la matrice di Hankel H in tre matrici reali: U (s x s con s le righe di H), V (rxr con r colonne di H) e S matrice diagonale con medesime dimensioni di H. Si tenga presente che la nozione di valore singolare non và confusa con quella di autovalore di una matrice, poichè i primi sono tutti reali non negativi, possono essere associati a matrici con dimensione qualsiasi e sono di facile e robusto computo. Un'altra caratteristica significativa della scomposizione ai valori singolari é che la determinazione del rango coincide con il numero di autovalori singolari non nulli della matrice S. É possibile capire cosí l'ordine del sistema o l'ordine della realizzazione minima. Tenendo poi presente che i campioni della risposta impulsiva sono affetti da rumore, gli stessi valori singolari si presentano in soprannumero ai fini identificativi, per cui in tale contesto vengono classificati in due famiglie: valori singolari dominanti e secondari. La stima dell'ordine n viene associata ai valori dominanti, una scelta che in alcuni casi si puó presentare soggettiva, a causa di una non evidente suddivisione nei due insiemi. Una volta scelto l'ordine si procede al calcolo delle matrici A, B, C, D del sistema, per esempio con il metodo a sottospazi di Kung ([5]). Vi sono poi diverse altre tecniche descritte in questa relazione di identificazione a sottospazi, non basate sull'identificazione della risposta impulsiva, bensí a partire da dati ingresso-uscita generici, e supportati sempre da tecniche numeriche. Sono il PBSID, PBSID_opt fornito dal prof.Chiuso, N4SID [2].

A. Sottospazi e catena chiusa

I metodi subspace identification trovano il loro utilizzo nell'identificazione di sistemi MIMO dinamici lineari a tempo discreto e dimensione finita di grandi dimensioni, dove i metodi a minimizzazione dell'errore di predizione sarebbero proibitivi per motivi tecnici. Da una prima analisi dei dati, valutati con l'uso degli strumenti forniti da Matlab, si evince quanto gli ingressi e le uscite in questo progetto siano fortemente correlati. A tale proposito é stato dunque necessario introdurre una estensione degli algoritmi di identificazione a sottospazi per operare in condizioni di anello chiuso, il cui studio ha subito uno sviluppo solo recentemente. Questa nuova classe di algoritmi consistenti per la catena chiusa, si basa sull'identificazione del predittore, in maniera molto simile al metodo ad errore di predizione (PEM), avvicinandosi in alcuni casi al suo comportamento. Le condizioni richieste per questo tipo di identificazione hanno richiesto di colmare un vuoto nelle procedure di realizzazione stocastica in presenza di feedback, cercando di eliminarne i termini indesiderati. Le procedure per l'identificazione con i metodi a sottospazi sono composte essenzialmente da due step:

1) costruzione di una base per lo spazio di stato attraverso una proiezione adatta delle sequenze dei dati.
2) data questa base, risolvere al tempo t e $t+1$ nel senso dei minimi quadrati ("least squares") il sistema:

$$\begin{cases} \hat{X}_{t+1} & \simeq & A\hat{X}_t + B\hat{U}_t + KE_t \\ Y_t & \simeq & C\hat{X}_t + E_t \end{cases} \tag{7}$$

In parole povere, i metodi subspace permettono di ricavare gli stati del sistema a partire dai dati I/O attraverso opportune proiezioni ortogonali o oblique degli sottospazi generati, definite in questi casi da matrici di Hankel a blocchi. Dopo di che, a partire dalla proiezione attraverso una decomposizione ai valori singolari (SVD) si ottiene una stima dell'ordine del sistema, la matrice di osservabilità estesa e la sequenza degli stati. Da questo si ricavano le matrici A,B,C,D come soluzione esatta o ai minimi quadrati di un opportuno sistema, e le matrici di covarianza Q,S,R dai residui. Si tenga presente che gli algoritmi a sottospazi differiscono essenzialmente sull'implementazione del primo punto, mentre il secondo é virtualmente uguale per tutti. Nelle prossime sezioni si descrivono qualitativamente gli algoritmi $PBSID$ e la sua versione ottimizzata $PBSID_{opt}$ data l'utilitá che ha avuto in questo progetto (vedi [2] per una trattazione esaustiva).

B. $PBSID$

Il PBSID ("'Prediction based subspace identification"') a differenza degli altri metodi a sottospazi tipo N4SID, utilizza differenti matrici di Hankel a blocchi, per operare in presenza di feedback. Si cerca cosí di risolvere il problema, introdotto dagli ingressi colorati, della non eliminazione del rumore additivo stocastico nella proiezione obliqua. Si utilizzano poi queste proiezioni oblique per ottenere una base per lo stato e

[2]Una tecnica di identificazione a sottospazi generica é descritta in [5]

la matrice di osservabilitá estesa; segue una decomposizione SVD per separarle e la risoluzione ai minimi quadrati per ricavare le matrici del sistema.

Dato il sistema lineare stocastico nella forma di innovazione:

$$\begin{cases} \mathbf{x}(t+1) &= A\mathbf{x}(t) + B\mathbf{u}(t) + K\mathbf{e}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t) + \mathbf{e}(t) \end{cases} \tag{8}$$

Per la trattazione si assume che:

- presenza di feedback tra I/O.
- la rappresentazione é minima.
- supposto D=0.
- si definisce $\overline{A} := A - KC$.
- il processo ergodico stazionario del secondo ordine $\mathbf{z}(t) := [\mathbf{y}^\top \mathbf{u}^\top]^T$ abbia densitá spettrale razionale e zeri tutti all'interno del cerchio unitario
- Il processo di rumore bianco **e**, di innovazione di **y** dato il passato di **z**, é definito come l'errore di predizione di un passo di $\mathbf{y}(t)$ dato il passato stretto di **z**(t) al tempo t.
- la differenza tra l'istante presente t e quello iniziale t_0 tende all'infinito rendendo trascurabile quest'ultimo termine. É un assunto necessario per assicurare la consistenza dello stimatore in catena chiusa, e la stima di infinite covarianze per ottenere l'asintotica efficienza degli stimatori, anche se questo mediamente puó non essere automatico.
- per semplificare l'analisi nell'articolo [2], si adotta un ordine ν di orizzonte futuro fissato e finito, sufficientemente grande.

Durante la trattazione, si useranno diversi strumenti la cui trattazione completa esula dallo scopo prefisso. La proiezione obliqua é la scomposizione in due sottospazi tra loro non ortogonali. Si denota come $\hat{E}_{||B}[\cdot|C_t]$, ossia la proiezione lungo lo spazio generato delle righe di B sullo spazio generato dalle righe di C_t. La formula estesa é:

$$\hat{E}_{||B_t}[A_t|C_t] = \hat{\Sigma}_{ac|b}\hat{\Sigma}^{-1}_{cc|b}C_t \tag{9}$$

Dove sotto le ipotesi di ergodicitá le matrici di covarianza campionaria (condizionata e non) sono:

$$\begin{aligned} \hat{\Sigma}_{ab} &:= \tfrac{1}{N}\textstyle\sum_{i=0}^{N-1} a_{t+i}b^\top_{t+i} \\ \hat{\Sigma}_{ab|c} &:= \hat{\Sigma}_{ab} - \hat{\Sigma}_{ac}\hat{\Sigma}^{-1}_{cc}\hat{\Sigma}_{cb} \end{aligned} \tag{10}$$

Infine si definisce la matrice di osservabilitá estesa:

$$\overline{\Gamma}^\top_\nu := [C^\top \quad \overline{A}^\top C^\top \quad \dots \quad (\overline{A}^\top)^{\nu-1}C^\top] \tag{11}$$

1) L'algoritmo: L'algoritmo PBSID per la ricerca di una base per lo spazio di stato usa diverse proiezioni oblique. La proiezione di ogni riga dell'uscita $Y_{t+h} \quad h = 0, ..., \nu$ puó essere vista con un modello VARX del tipo:

$$\begin{aligned} \hat{Y}_{t+h} &:= \hat{E}[Y_{t+h}|Z_{[t_0,t+h)}] \\ &= \hat{\Psi}_{1,h}Z_{t+h-1} + \ldots + \hat{\Psi}_{t+h-t_0,h}Z_{t_0} \end{aligned} \tag{12}$$

dalla quale attraverso l'algoritmo "'predictor-based"'(la P in apice) le proiezioni oblique asintoticamente in N, forniscono:

$$\begin{aligned} \hat{Y}^P_{t+h} &:= \hat{E}_{||Z_{[t,t+h)}}[Y_{t+h}|Z_{[t_0,t)}] \\ &= \textstyle\sum_{i=h+1}^{t-t_0+h} \hat{\Psi}_{i,h}Z_{t+h-i} \quad \simeq \quad C\overline{A}^{h-1}X_t \end{aligned} \tag{13}$$

Tutti i predittori fino a $h = T$ con tale approssimazione si possono raggruppare come:

$$\hat{Y}^P_{[t,T)} := \begin{bmatrix} \hat{Y}^P_t \\ \hat{Y}^P_{t+1} \\ \\ \vdots \\ \\ \hat{Y}^P_{T-1} \end{bmatrix} \simeq \overline{\Gamma}_\nu X_t \tag{14}$$

Dalla decomposizione ai valori singolari:

$$W_p^{-1}\hat{Y}^P_{[t,T)} := PDQ^\top \tag{15}$$

$$= \begin{bmatrix} P_n & \tilde{P}_n \end{bmatrix} \begin{bmatrix} D_n & 0 \\ 0 & \tilde{D}_n \end{bmatrix} \begin{bmatrix} Q_n^\top & \tilde{Q}_n^\top \end{bmatrix} \tag{16}$$

con una opportuna matrice peso W_p, é possibile stimare una matrice di osservabilitá, ottenuta ignorando i valori singolari che descrivono meno il sistema (ossia ponendo nell'equazione 16 la $\tilde{D}_n \simeq 0$) [3]. Allora:

$$\hat{\overline{\Gamma}}_\nu = W_P P_n D_n^{1/2} \tag{17}$$

quindi dalla 14 e sfruttando le equazioni precedenti si trova una base per lo spazio di stato

$$\hat{X}^{PBSID}_t := \hat{\overline{\Gamma}}^{-L}_\nu \hat{Y}^P_{[t,T)} \tag{18}$$

$$\hat{X}^{PBSID}_{t+1} := \hat{\overline{\Gamma}}^{-L}_\nu \hat{Y}^P_{[t+1,T)} \tag{19}$$

Dove $\hat{\overline{\Gamma}}^{-L}_\nu$ é l'inversa sinistra definita come

$$\hat{\overline{\Gamma}}^{-L}_\nu := (\hat{\overline{\Gamma}}^{-L}_\nu W_p^{-\top} W_p^{-1} \hat{\overline{\Gamma}}_\nu)^{-1} \hat{\overline{\Gamma}}^{-L}_\nu W_p^{-\top} W_p^{-1} \tag{20}$$

[3] questa analisi é ripresa nella descrizione del metodo PCA

C. $PBSID_{opt}$

Il metodo $PBSID_{opt}$ é numericamente equivalente alla stima di un modello VARX, utilizzante i medesimi passi per l'identificazione a sottospazi e con il pregio di avere una minore complessitá computazionale. Unisce cosí i pregi dello stimatore consistente CCA per la catena aperta e migliora il comportamento in retroazione rispetto a modelli SSARX/PBSID, nel senso della asintotica varianza. Infatti il PBSID é asintoticamente equivalente al CCA ("Canonical Correlation Analysis") per l'identificazione di serie di dati e in presenza di ingressi bianchi. Mentre la sua versione ottimizzata fornisce una minore o uguale asintotica varianza per ogni scelta del segnale di ingresso, anche colorati, (dimostrato in [3]). Comunque non é ancora del tutto chiaro il comportamento dell'efficienza in generale. Questa versione ottimizzata per i sistemi MIMO, presenta migliori prestazioni di identificazione in sistemi con contenuto alle basse frequenze (come le temperature in un edificio), la sostituzione del calcolo esplicito delle proiezioni oblique con passaggi di minimizzazione iterativa, la velocitá di calcolo e la produzione di modelli stabili e non affetti da bias. La selezione automatica dell'algoritmo dell'ordine del modello segue il criterio di rapporto tra valori singolari adiacenti e tra il valore massimo e minimo. Per quanto riguarda l'algoritmo, esso differisce dal PBSID originale, per il fatto che la stima del predittore 12 é formulato come un problema ai minimi quadrati pesati, dove la proiezione equivale, dopo opportune manipolazioni, a risolvere riga per riga il seguente sistema vettoriale:

$$\begin{aligned} Y &:= \begin{bmatrix} vec(Y_t) \\ vec(Y_{t+1}) \\ \vdots \\ vec(Y_T) \end{bmatrix} = S^P\Omega^P + \begin{bmatrix} vec(E_t) \\ vec(E_{t+1}) \\ \vdots \\ vec(E_T) \end{bmatrix} \\ &= S^P\Omega^P + E \end{aligned}$$

Dove si sono definite le nuove matrici:

$$\begin{aligned} \Xi_i &= C\overline{A}^i\Delta \quad \forall i = 1,\dots,\nu \\ \Delta &:= [\overline{A}^{(t-t_0-1)}[KB]\overline{A}^{(t-t_0-2)}[KB]\dots[KB]] \\ \Psi_{ij} &= C\overline{A}^{j-1}[KB] \quad \forall i,j = 1,\dots,\nu \end{aligned} \tag{21}$$

A noi basti sapere che attraverso uno stimatore di Markov si trova iterativamente una soluzione ottima $\hat{\Omega}^{Popt}$ come stima di Ω^P che comporta la piú piccola varianza asintotica basata sul modello.

$$Y := S^P\Omega^P + E \tag{22}$$

Con la soluzione $\hat{\Omega}^{Popt}$ di 22 é possibile rivedere i passaggi dell'algoritmo PBSID precedente, pervenendo al nuovo stato:

$$\hat{X}^{Popt}_{t+1} := (\hat{\overline{\Gamma}}^{Popt}_{\nu})^{-L} \begin{bmatrix} \hat{\Xi}^{Popt}_1 & \hat{\Psi}^{Popt}_{11} \\ \hat{\Xi}^{Popt}_2 & \hat{\Psi}^{Popt}_{22} \\ \vdots & \vdots \\ \hat{\Xi}^{Popt}_{\nu} & \hat{\Psi}^{Popt}_{\nu\nu} \end{bmatrix} Z_{[t_0,t+1)} \tag{23}$$

Dove $\hat{\overline{\Gamma}}^{Popt}_{\nu}$ é la matrice estesa di osservabilitá stimata (sostituendo $\hat{Y}^{Popt}_{t+k}$ a $\hat{Y}^{P}_{t+k}$ nella decomposizione 16). Similmente é possibile stimare la sequenza E_t, come:

$$\hat{E}^{Popt}_t := Y_t - \hat{E}[\hat{Y}^{Popt}_t | \hat{X}^{Popt}_t] \tag{24}$$

V. La validazione

L'ultima, ma non meno importante, fase dell'identificazione é la validazione del modello ricavato. Ci si chiede se il modello identificato spieghi i dati in maniera soddisfacente o se addirittura sia il caso di riconsiderare l'intero processo di identificazione. L'analisi delle prestazioni viene effettuata in condizioni sperimentali diverse da quelle di identificazione. Una metodologia é la suddivisione dei dati in due insiemi:

- l'uno per l'identificazione,
- il secondo per la validazione

Nel secondo set di dati si confrontano le uscite reali con quelle del modello con gli ingressi reali. Se tale fase comporta risultati insoddisfacenti é necessario rivedere il processo di identificazione, limitando il ricorso ad una nuova fase di raccolta dati che potrebbe risultare notevolmente onerosa, o non possibile.
In questo caso é comprensibile quale rilevanza assume il design dell'esperimento di raccolta dati, in modo tale da fornire dati il piú possibile ricchi di informazione per l'identificazione.

Lo scopo del simulatore in questo progetto é in parte diverso da quello precedente. É utile dal punto di vista di generazione di set di esperimenti (estate, inverno, ...) per validare il modello, ma é anche vero che partendo dal presupposto che il set di rilevazioni di temperatura sia molto limitato, ossia unico per zona[4], e per il fatto che gli ingressi esterni siano un andamento medio e non ripartibile nelle quattro coordinate geografiche, $DesignBuilder$ non offre possibilitá di selezione, nel confronto con i modelli identificati con i dati reali. A correlazione di quello appena detto, si tenga presente che la divisione in due metodologie della fase di validazione é giustificata dall'uso futuro del modello. Ossia se il modello verrá sfruttato come si presenta in catena aperta o come predittore. Nel nostro caso, se si vorrá conoscere gli andamenti all'interno dell'edificio avendo a disposizione anche un solo termostato all'interno si utilizza il predittore. Se invece, si avranno a disposizione solo gli andamenti della temperatura esterna, si usa il modello identificato nella maniera migliore possibile, ossia con tutti i dati disponibili.

[4]zona é intesa come una stanza, una entrata o un'area piú vasta non frazionata in tutti i lati da divisori

Nel caso di un predittore si definisce ottimo se rispetta le seguenti condizioni:

- predittore e l'errore di predizione devono essere totalmente scorrelati.
- la varianza dell'errore di predizione sia minima (fra tutti i predittori che soddisfano il primo punto).

A. metriche di valutazione

Come precedentemente accennato, la validazione di un modello si divide in due tronconi: la cross-validazione o la validazione del modello con metodi fpe, aic. La prima é valida qualora la quantitá di dati sia elevata e il limitare l'intervallo di identificazione non comporta errori apprezzabili, la seconda invece viene sfruttata in quei casi in cui il set di dati é limitato, ed é necessario, quindi usarli tutti per l'identificazione. La verifica della bontá del modello avviene con i criteri fpe o akaike su un set di modelli identificati dai stessi dati a cui vengono individualmente cambiati i parametri e ordine.

1) covarianza e correlazione: La covarianza $cov(x_1, x_2)$ in questo caso campionaria, é un indice della relazione lineare esistente tra due variabili, cioé della loro variabilitá relativa. La covarianza puó assumere valori diversi a seconda dell'unitá di misura con cui vengono riportate le variabili, e di conseguenza non é un buon indicatore. In generale, comunque, un valore assoluto elevato indica l'esistenza di una forte correlazione tra due variabili, viceversa nel caso il valore della covarianza tenda a zero. Mentre la correlazione $corr(x_1, x_2)$ di due variabili aleatorie é un indice adimensionale, compreso tra -1 e 1, della relazione lineare esistente tra due variabili normalizzate. Un valore di correlazione positivo indica che le due variabili tendono a crescere o a decrescere simultaneamente, mentre un valore negativo indica che tendenzialmente le due variabili hanno carattere opposto. Infine, due variabili linearmente indipendenti l'una dall'altra hanno correlazione nulla. Ció non significa che le due variabili sono indipendenti tra loro, cioè che sono ortogonali, ma solo che non esiste nessuna relazione lineare in grado di esprimere una variabile in funzione dell'altra; può però esistere una funzione di qualsiasi altro tipo in grado di mettere in relazione le due variabili. Queste definizioni permettono di capire ulteriormente il test di bianchezza e una sua contro-verifica. Quest'ultima viene attuata sempre per via simulativa con lo scopo di verificare se effettivamente l'errore di predizione fosse bianco, o almeno il suo andamento per ogni sensore tenda a rimanere all'interno dell'intervallo di confidenza.

2) test bianchezza: Il test di bianchezza é un test per decidere se un processo $X(\cdot)$ è bianco, sulla base di una sua realizzazione $x(1), x(2), \ldots, x(N)$. Nel contesto dell'identificazione il test di bianchezza é utile per verificare la bontá di un modello identificato mediante l'analisi del residuo, cioé dell'errore di predizione associato al modello identificato. Se il residuo é un processo bianco (a media nulla), allora il modello identificato é una buona descrizione della serie temporale/sistema dinamico in esame. Risulta utile anche nel valutare sulla base di una realizzazione se, una volta rimosse le eventuali componenti deterministiche, la componente stocastica di una serie temporale è un processo bianco.

La funzione in Matlab *resid* calcola delle stime della funzione di correlazione del residuo, e plotta un intervallo di confidenza [5] all'interno del quale tutte le correlazioni eccetto la prima (la varianza) si dovrebbero trovare. Se cos'í risulta, possiamo ritenerci soddisfatti del modello identificato. Nel caso invece, alcune residui siano significativamente al di sopra della soglia, il modello relativo non é corretto, i residui non sono incorrelati ed é auspicabile una modifica. É il caso in cui l'ordine del modello non sia corretto, o la stessa struttura o la lunghezza della sequenza dei dati non sia appropriata.

Il test dei residui deve essere usato con cautela ponendo particolare attenzione a un dettaglio pratico. Se si usano metodi a sottospazi (come N4sid appartente alla classe IV *instrumental variable* che si focalizzano sulla parte ingresso-uscita $F(z)$ nella relazione 25):

$$\mathbf{e}(t) = G(z)^{-1}[\mathbf{y}(t) - F(z)\mathbf{u}(t)] \tag{25}$$

viene consigliato in questa situazione di verificare l'indipendenza dell'errore e e di u, dando meno importanza al test di bianchezza per l'errore. Si sposta cosí l'attenzione sulla dinamica di $G(z)$ e non sulle proprietá del segnale attraverso $F(z)$ (maggiori informazioni su [**?**]). Come conseguenza di ció é lecito aspettarsi che i residui non soddisfino la proprietá menzionata prima di modello bianco. Non é comunque il nostro caso perché il sistema che usiamo 8 per identificare é di tipo *forward innovation* dove il disturbo é esplicitamente modellato come rumore bianco $\mathbf{e}(t)$, per cui il test di bianchezza sará uno strumento di analisi ampiamente sfruttato in validazione.

Partendo dall'assunto che $\epsilon(t)$ sia bianco e la sua funzione di covarianza $r_\varepsilon(\tau) = 0 \quad \tau \neq 0$, una volta che si costruisce la stima della funzione di covarianza:

$$\hat{r}_\varepsilon(\tau) = \frac{1}{N}\sum_{t=1}^{N-\tau} \epsilon(t+\tau)\epsilon(t) \tag{26}$$

Essendo validi gli assunti per $N \to \infty$:

$$\hat{r}_\varepsilon(\tau) \to 0 \qquad \tau \neq 0 \tag{27}$$

$$\hat{r}_\varepsilon(0) \to \lambda^2 = E\varepsilon^2(t) \tag{28}$$

Si considera cosí la quantitá normalizza per il test il rapporto:

$$x_\tau = \frac{\hat{r}_\varepsilon(\tau)}{\hat{r}_\varepsilon(0)} \tag{29}$$

3) test indipendenza: In relazione con il test di bianchezza, il test di indipendenza fornisce un esito positivo riguardo la bontá del modello, nel caso i residui sia incorrelati con gli ingressi passati. Eventuali picchi fuori dall'intervallo di confidenza indicano che il modello non é descritto come parte di quell'uscita relativa al corrispondente ingresso analizzato. Per esempio, un picco al ritardo (lag) k-esimo significa che l'uscita $y(t)$ originata dall'ingresso $u(t-k)$ non é propriamente descritta dal modello. Questo test si basa sull'assunto

[5] L'intervallo di confidenza corrisponde al range di valori dei residui che data la loro specifica probabilitá associata, per il sistema si possono ritenere statisticamente insignificanti.

ε sia indipendente dagli ingressi passati e di media nulla. Se i residui e gli ingressi sono indipendenti, allora

$$r_{\varepsilon u}(\tau) = E\varepsilon(t+\tau)u(t) = 0 \tag{30}$$

Si considerano valori sia positivi che negativi di τ per il test. Come affermato precedentemente se il modello non é accurato, é ragionevole che l'andamento di $r_{\varepsilon u}(\tau)$ per $\tau \geq 0$ sia lontano da zero, mentre per $\tau < 0$ puó non essere nullo dipendentemente dall'autocorrelazione di $u(t)$. Per esempio, se $u(t)$ é bianco allora $r_{\varepsilon u}(\tau) = 0$ per $\tau < 0$ anche se il modello é inaccurato, ma puó anche accadere che, nel caso il modello sia l'esatta descrizione del sistema, ossia $r_{\varepsilon u}(\tau) = 0$ per $\tau \geq 0$ si manifesti una $r_{\varepsilon u}(\tau) \neq 0$ per $\tau < 0$: questa é la manifestazione della presenza di feedback.

La quantitá normalizzata usata dal test é nella forma:

$$x_\tau = \frac{\hat{r}_{\varepsilon u}(\tau)}{[\hat{r}_\varepsilon(0)\hat{r}_u(0)]^{1/2}} \tag{31}$$

Dove:

$$\hat{r}_{\varepsilon u}(\tau) = \frac{1}{N}\sum_{t=1-min(0,\tau)}^{N-max(\tau,0)} \varepsilon(t+\tau)u(t) \tag{32}$$

4) errore quadratico medio: Un indice della qualitá del modello stimato diffusamente usato é l'MSE, ossia "'mean square error"'. Dato che per scarto o residuo viene definito come $\epsilon(t) = y_i - \hat{y}_i$ Esso é descritto dalla formula:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{33}$$

$$= Var\{\hat{y}\} + mean^2(\hat{y} - y) \tag{34}$$

MSE é un buon criterio di giudizio della bontá di un'equazione, perché é un indice della dispersione degli errori rispetto alla retta o iperpiano di regressione. Si ponga attenzione al fatto che si é fatta l'assunzione $\frac{1}{N-k-1} \simeq \frac{1}{N}$, per cui la somma dei quadrati dei residui é mediata sul numero elevato dei campioni senza sottrarre ad essi i gradi di libertá del sistema come in realtá la vera definizione di MSE richiama. L'equazione 34 come traduzione dell'MSE in media e varianza, esprime come queste proprietá della media e della varianza siano utilizzate nell'inferenza statistica per fornire informazioni sui parametri stimati di un'equazione di regressione; ad esempio, se si ha una stima degli errori, dalla varianza degli errori si puó ricavare un valore che indichi la probabilità di ogni punto di essere uguale al valore atteso e un intervallo, detto intervallo di confidenza, all'interno del quale avere il 95% di probabilità di trovare il valore atteso. Infine si tenga presente che questo criterio assume nomi diversi, ma con medesima formulazione, nel caso siano usati con set di dati diversi, ossia vengono utilizzati i criteri MSEP (Mean Square Error of Prediction) e MSECV (Mean Square Error of Cross-Validation) che sono computazionalmente identici, ma calcolati su dati di convalida o attraverso la cross-validation dei dati.

5) mappa zero-polo: Partendo dal presupposto che la stima del modello possa essere intrapreso come una allocazione zero-polo della fdt dei dati, la verifica della mappa zero-polo di ogni uscita o del modello identificato, fornisce una prima indicazione sull'esattezza dell'ordine del modello usato. Infatti in caso di zeri e poli molto vicini, suggerisce la possibilitá di una ulteriore riduzione del modello.

6) fit: Un'altro indice di valutazione sfruttato nella fase di validazione é la percentuale di varianza spiegata, ossia il fit in percentuale fornito in maniera automatica dalla funzione $compare.m$ di Matlab:

$$fit = 100 * \left[1 - \frac{norm(y_h - y)}{norm(y - \overline{y})}\right] \tag{35}$$

Dove y_h rappresenta l'uscita del modello con gli ingressi di validazione, mentre y sono le uscite di validazione e la norma di solito é la norma due. Da 35 si evince che se l'uscita h-esima di validazione é uguale a quella in uscita del modello, la differenza tra le due é nulla e l'attinenza ́del 100%. Essendo il sistema MIMO questo calcolo viene effettuato per ogni uscita in maniera automatica sempre dalla funzione $compare.m$, e verrá tenuto in considerazione nella valutazioni sperimentali, anche nell'analisi predittiva.

7) Fpe e Aic: I criteri per la scelta dell'ordine del modello in cui si cerchi di valutare la bontá del modello rispetto alle diverse possibili realizzazioni dei dati, assumono diverse connotazioni in relazioni al modello e alla tecnica di stima usata. In via generale, tali criteri vengono ottenuti penalizzando con una decrescita di una funzione perdita $V_N(\hat{\Theta}_N)$ in funzione dell'ordine o del set di modelli. La scelta del modello é associata al valore minimo dell'indice relativo. Una forma generale [7] di questo criterio é:

$$W_N = V_N(\hat{\Theta}_N)[1 + \beta(N,p)] \tag{36}$$

Dove $\beta(N,p)$ é funzione di N e dei parametri p, penalizzando modelli troppo complessi. Nel caso PEM, una forma realizzabile é di tale criterio é:

$$W_N = V_N(\hat{\Theta}_N)\frac{1 + p/N}{1 - p/N} \tag{37}$$

Nell'ipotesi di N sufficientemente grande le approssimazioni portano ad un corrispondente criterio, denominato *Akaike information criterion* (AIC).

$$W_N = NlogV_N(\hat{\Theta}_N) + 2p \tag{38}$$

Si tratta di un criterio di scelta della complessitá del modello di tipo statistico. Viene determinato minimizzando la distanza di Kullback tra la densitá di probabilitá dei dati osservati e quella che produrrebbe il modello in esame. Nel progetto in esame, esiste una routine che sfrutta tali criteri per determinare l'ordine del modello iterando piú modelli ARX con differenti ordini. Con tale ordine si setta l'orizzonte passato/futuro da fornire al funzione PBSIDopt. Si richiama il fatto che la regola per la selezione dell'ordine viene detta ed é consistente se la probabilitá di selezionare l'ordine sbagliato tende a zero quanto piú i dati tendono all'infinito.

VI. ANALISI RISULTATI

Per definire la bontá di uno stimatore dalla teoria siamo a conoscenza che la sola proprietá di asintotica correttezza non śufficiente a classificarne il comportamento, infatti é necessaria anche la proprietá di consistenza. La proprietá di consistenza permette di affermare che la qualitá della stima migliora all'aumentare dei campioni rilevati $N \to \infty$).
la trasformata di Fourier di ogni segnale mette in luce la presenza di componenti stagionali come somma di sinusoidi. Infatti si visualizzano dei picchi presenti nella trasformata che indicano in questo modo le pulsazioni delle sinusoidi che contribuiscono alla componente deterministica.

Le stime della funzione di correlazione effettuata con il comando $resid$ di Matlab ottenuta a partire dai dati e considerando i processi stazionari, rimangono al di fuori dell'intervallo di confidenza ,invece di tendere a zero (o al limite permanere sotto la soglia). Anche questa analisi é sinonimo della presenza di una componente deterministica, nonostante sia stata preventivamente tolta la media dei segnali. Infine un altro modo con cui si é potuto verificare questo é nella visualizzazione della mappa poli/zeri. Il modello identificato, presenta i poli vicino al cerchio di raggio unitario, ossia alla pulsazione che contribuisce alla componente deterministica, e in questo caso a uno che in statistica si traduce in presenza di un trend lineare.

É possibile che questa analisi si riveli solo a corredo del nostro scopo, infatti é ovvio che le nostre misure rivelino dei trend, ossia l'intercambio giorno/notte e temperature con andamenti sinusoidali. Si ritiene sufficiente nella fase di prefiltraggio del segnale, il solo togliere la media, evitando condizionamenti ulteriori.

Eventuali correlazioni tra i residui e gli ingressi per ritardi negativi non é necessariamente un indicazione di un modello non accurato. Mentre assume maggiore risvolto nell'analisi dei residui nel caso gli stessi, al tempo t, dipendano dagli ingressi futuri, sinonimo della presenza di feedback nel sistema. In tal caso si focalizzerá l'attenzione nella cross-validazione dei residui sugli andamenti relativi ai ritardi positivi.

All'atto dell'identificazione, nel caso le variabili siano tra loro linearmente correlate l'inversione di alcune matrici all'interno dell'algoritmo, puó portare a un cattivo condizionamento numerico creando problemi. É possibile risolvere ove questo accada, utilizzando metodi alternativi per la risoluzione di sistemi lineari di equazioni come la decomposizione QR.

Il segnale di ingresso é persistentemente eccitante di ordine 50 (dato fornito dalla funzione matlab *advice()*), per cui non dovrebbero sorgere problemi di identificabilitá, dato l'ordine del modello limitato. Infatti per identificare i parametri di un modello di ordine n, di solito l'ingresso deve essere persistentemente eccitante (PE) di ordine $2n$, ma questo valore indicativo varia da metodo a metodo. Il grafico ... rappresenta la temperatura esterna come ingresso e la sua densitá spettrale di potenza. Si verifica che il segnale é PE di ordine n se la sua densitá spettrale é diversa da zero per almeno n punti, meglio ancora se il segnale ha densitá spettrale per tutte le frequenze strettamente positiva, il che garantisce persistente eccitazione per ordini ben maggiori.

Aumentando l'ordine del modello, ad esempio fino a 19, come il toolbox ritorna di default, le prestazioni migliorano notevolmente il test di bianchezza fornisce quasi esito positivo, peró da un'analisi della mappa zero polo incrociata della fdt tra ogni uscita e ogni ingresso, del modello identificato, si evince che comunque ci sono cancellazioni zero-polo, indice di necessitá di una riduzione del modello. Resta comunque il fatto che la vicinanza dei poli/zeri nell'intorno di uno é associabile anche al tempo di campionamento, essendo che i modi del sistemi campionato, vanno come $e^{\Delta t}$

VII. DESIGNBUILDER IN QUESTE SIMULAZIONI

Note durante l'implementazione: al fine di identificare il modello con cui DesignBuilder determina le simulazioni, limitatamente all'andamento delle temperature, le scelte tipologiche e topologiche degli I/O sono la conseguenza dei seguenti ragionamenti. Come uscite il simulatore fornisce tre andamenti per le temperature:

- temperatura dell'aria T_a: calcolo della media delle temperature dell'aria.
- temperatura radiante T_r: la media della temperatura radiante (MRT) della zona, calcolata assumendo la persona al centro della stanza, senza dare peso a particolari superfici.
- temperature operativa T_{op}: media tra la temperatura dell'aria e quella radiante.

Essendo del tutto simili a meno della media, si sceglie l'andamento T_{op} della temperatura operativa, che ne riassume la media delle prime due, come espresso dalla formula $T_{op} = \frac{T_a+T_r}{2}$. Le temperature operative vengono fornite per ogni stanza, ma anche quelle medie per piano e per l'intero edificio cosa non necessaria e da non considerare all'atto dell'esportazione dei dati. Quest'ultime vengono prese come uscite del modello, mentre per quanto riguarda gli ingressi il discorso si complica. Nell'immediato il primo ingresso utile é la temperatura esterna a bulbo secco, poi nel tentativo di aderenza alla metodologia con cui sono stati rilevati i dati nella realtá, si é cercato di estrapolare dal simulatore i dati di irraggiamento (preferibilmente per ogni facciata).

Nel caso si abbia a disposizione un unico ingresso, ossia la temperatura esterna, nelle simulazioni non si sono, ovviamente, riscontrate differenze nel caso si identifichi con uno o con una matrice con colonne come copie dell' ingresso, eventualmente con rumore additivo bianco casuale. In sostanza l'andamento degli ingressi non é per niente sufficientemente "'ricco"' per una identificazione sufficiente. A tale proposito si é simulato un arco di periodo piú lungo rispetto alla settimana di raccolta dati reali, in modo tale da portare maggiore informazione, *ma non si é rilevato utile alla stima, in alcuni casi il modello é peggiorato, con errori medi di 16-18 °C. Ovviamente questa é la conseguenza di una notevole inconsistenza dei dati, di ingressi non sufficientemente persistentemente eccitanti, e di una matrice A del modello di stato praticamente instabile. Le ipotesi per l'applicabilitá della teoria asintotica (verifica) quindi, non sussistono, rendendo inutile la ricerca di deduzione di un modello. Si conclude come tale simulatore non si presta ai nostri scopi di identificazione, relegandolo*

alla sua attivitá principale, ossia alla simulazione termodinamica e principali componenti di progetto di riscaldamento e climatizzazione.

A questo punto si immette un nuovo ingresso, come la combinazione lineare della radiazione solare .....

Il metodo "'n4sid"' é lento e apporta maggiori errori al modello, mentre....

VIII. Il problema del modello nella regressione lineare

Nei problemi di controllo di sistemi dinamici, uno dei passi fondamentali è quello di poter ottenere un modello matematico che descriva il comportamento e le relazioni tra le variabili di ingresso o predditive e le variabili di uscita. Nel caso ideale il modello matematico è disponibile come set di equazioni ai principi primi esplicitato mediante equazioni di bilancio di materia, e di energia in cui i parametri sono coefficienti reali del processo, aventi significato fisico.
Nella stragrande maggioranza dei casi invece non si dispone del modello deterministico, questo significa che non sono noti o ricavabili alcuni parametri del modello. In questi casi si dispone invece di una collezione di misure delle variabili di interesse o di alcune grandezze ad esse collegate e si cerca di approssimare il reale comportamento del processo mediante un'equazione empirica di *regressione*.
Le tecniche di analisi statistica multivariata basate sul concetto di variabile latente forniscono gli strumenti necessari per estrarre da una moltitudine di dati le informazioni necessarie separandole dal rumore e dai disturbi presenti nel processo.
L'applicazione dei metodi di regressione ed analisi statistica multivariata permette di identificare o meglio stimare la relazione esistente tra i due set di variabili, di ingresso e di uscita; in questo modo si ottiene un modello matematico empirico del processo che ci permette di progettare un sistema di controllo. Tali tipi di modelli vengono denominati in letteratura come modelli black-box.
L'identificazione del modello è quindi una parte importante e ingegnisticamente onerosa nel controllo di qualunque sistema, come indicato nell'abstract noi cercheremo di identificare un modello termodinamico che descriva l'evolversi della temperatura all'interno di un edificio sotto l'effetto della temperatura esterna e della radiazione luminosa che interessa l'edificio su cui sono state svolte le misure sperimentali.

A. Regressione Lineare Multivariata

Cercheremo di fornire un quadro di insieme della teoria della regressione lineare multivariata per capire bene dove l'applicheremo nella soluzione del nostro problema evidenziandone i punti di forza che presenta. Presentiamo due metodi di regressione lineare multipla:

- La regressione ai minimi quadrati classici *Ordinary Least Squares Regression*
- La proiezione su Strutture Latenti *Projectin on Latent Structures*

1) Modelli Lineari Multivariati: Supponiamo di disporre di n misure sperimentali di k variabili indipendenti $x_1, \ldots, x_k$ e di altrettante misure della variabile dipendente y, con $n > k$. Un modello multivariato coinvolge una o più variabili dipendenti, rappresentate come funzione lineare di più variabili indipendenti o predittive.
Il modello di equazione lineare multivariata è:

$$\begin{aligned} y_i &= \alpha_0 + \alpha_1 x_{i1} + \ldots + \alpha_k x_{ik} + \epsilon_i \\ &= \textstyle\sum_{j=1}^{k} \alpha_j x_{ij} + \epsilon_i \quad \forall i = 1, \ldots, n \end{aligned} \tag{39}$$

dove y_i è l'i-esimo valore della variabile dipendente, $x_1, \ldots, x_k$ sono le k variabili indipendenti e $\alpha_0, \ldots, \alpha_k$ sono $k+1$ parametri matematici o fisici il cui valore viene stimato con metodi di regressione statistica in modo da minimizzare ϵ_i che è un errore stocastico sistematico che tiene conto del rumore di misura e di altre cause erratiche nella costruzione del modello.
L'applicazione del modello di regressione lineare ai dati di processo è resa complicata da numerose cause: sia la variabile dipendente che le variabili preditive sono affette da errore di misura, le variabili sono molto correlate tra loro, i dati sono imprecisi o in alcuni casi mancanti, a causa della rottura o malfunzionamento di un sensore. Inolre il rapporto segnale/rumore delle variabili può essere molto basso, introducendo rumore sistematico all'interno dell'equazione di regressione: questo fatto è molto importante, perchè induce nell'equazione di regressione, attraverso i coefficienti di regressione, a modellare anche l'errore misurato dalle variabili predittive e dipendenti per minimizzare l'errore ϵ_i.
Ciò significa che anche un'equazione di regressione statisticamente significativa che ottenga un buon fitting dei dati sperimentali e minimizzi l'errore in (39) può essere non utilizzabile se prima non viene separata l'informazione utile dal rumore.

B. Regressione lineare ai minimi quadrati

La regressione ai minimi quadrati (*Ordinary Least Squares*) risolve un sistema di n, equazioni lineari in cui sono incogniti $k+1$ coefficienti di regressione $\alpha_0 \ldots \alpha_k$ corrispondenti ad ogni variabile predittiva più l'intercetta, cercando di minimizzare la varianza dell'errore ϵ.
La regressione ai minimi quadrati è possibile solo se il numero di campioni disponibili è superiore al numero di variabili predittive; infatti se $n < k$ il sistema non è risolvibile in modo univoco, ma si hanno infinite soluzioni. Nel caso invece si abbia $n = k$ la soluzione esatta nel sistema lineare è possibile ed è unica: si ottiene per sostituzione o mediante algoritmi numerici come la decomposizione di Gauss. L'equazione di regressione multivariata ai minimi quadrati è data da:

$$\hat{y_i} = b_0 + b_1 x_{i1} + \ldots + b_k x_{ik} = b_0 + \sum_{j=1}^{k} b_j x_{ij} \quad \forall i = 1, \ldots, n$$

Dove $b_0 \ldots b_k$ sono le stime dei coefficienti di regressione parziale ottenute risolvendo il sistema di equazioni normali ricavate differenziando l'errore totale rispetto ad ogni incognita $\alpha_0 \ldots \alpha_k$. Il residuo e_i tiene conto dell'imprecisione che si

introduce utilizzando valori stimati non esatti per i coefficienti di regressione.
I residui, nel caso il modello sia esatto, hanno queste caratteristiche:

- si distribuiscono normalmente e non sono tra loro autocorrelati;
- hanno media nulla;
- sono non correlati con la variabile calcolata: $cov(e_i, \hat{y}_i)$

Il metodo di regressione ai minimi quadrati pùo essere applicato anche al caso di più variabili ottenendo un'espressione matriciale. Il sistema di equazioni normali del metodo di regressione ai minimi quadrati si scrive:

$$X^T X b = X^T y$$

la cui soluzione è

$$b = (X^T X)^{-1} X^T y$$

Abbiamo cosi l'equazione di regressione lineare multivariata ai minimi quadrati in forma matricile:

$$\hat{y} = X(X^T X)^{-1} X^T y$$

Tale soluzione minimizza la funzione MSE, dove le stime b_j dei coefficienti di regressione parziali α_j si caratterizzano perchè:

- minimizzano MSE indipendentemente dalla distribuzione degli errori e_i;
- più la deviazione standard è piccola più la stima dei coefficienti è precisa, se il modello soffre di multicollinearità l'errore standard è molto alto ;
- sono stime attendibili di α_j se le variabili predittive sono tra loro non correlate o se i residui e_i sono statisticamente indipendenti.

La soluzione del problema ai minimi quadrati è possibile solo se la matrice $(X^T X)^{-1}$ esiste, cioè se la matrice X ha rango massimo. Nel caso le variabili siano tra loro correlate e non sia possibile effettuare l'inversione della matrice $X^T X$, si deve rinunciare ad ottenere una stima ai minimi quadrati dei coefficienti di regressione, utilizzzando metodi alternativi per la risoluzione di sistemi lineari di equazioni come la decomposizione QR. Va rilevato infine che la regressione ai minimi quadrati classici si basa su un'assunzione che viene raramente rispettata dai dati di processo, ovvero l'esattezza delle misure sperimemtali delle variabili predittive: l'essenza di questo requisito fa cadere la validità dell'equazione di regressione ai minimi quadrati, perchè il modello non è in grado di verificare statisticamente l'esattezza delle stime dei coefficienti di regressione e l'attendibilità delle stime con dati di convalida.

SEZ INTRODUTTIVA

Quando si vuole risolvere un problema ai minimi quadrati in cui $\mathbf{Y} \in \mathbf{R^{nxp}}$ e $\mathbf{U} \in \mathbf{R^{nxm}}$ siano, rispettivamente, un'insieme di $\mathbf{p}$ e un insieme di $\mathbf{m}$ processi aleatori (ergodici del secondo ordine), realizzazioni di ingressi e uscite di un sistema, del tipo:

$$Y = \Theta \cdot U + E \tag{40}$$

con $\mathbf{E}$ rumore bianco di dimensione p e varianza $Var_{\mathbf{E}} = I_p$, con I matrice identitá, la soluzione, ovvero l'insieme dei parametri del sistema in questione che si vuol identificare, sarebbe data da:

$$\Theta = (UU^T)^{-1} \cdot U^T \cdot Y \tag{41}$$

Il problema principale che si può verificare è che $\mathbf{U}$ sia singolare, ovvero che i dati in ingresso formino una famiglia di dati linearmente dipendenti. In questo caso l'inversa di $\mathbf{U}\mathbf{U}^T$ non esiste; in realtá, anche utilizzando una pseudoinversa in luogo di un'inversa, i parametri identificati non sarebbero rappresentativi del sistema, e porterebbero un sistema instabile in presenza di set di dati diversi da quelli utilizzati in questa fase. É necessario dunque disporre di dati linearmente indipendenti:in questo modo, non solo si avrebbe una soluzione, ma la soluzione trovata sarebbe molto più consistente. Basti pensare che in (41), se i dati fossero linearmente dipendenti, l'espressione $(\mathbf{U}\mathbf{U}^\mathbf{T})^{-\mathbf{1}}$ fornirebbe dei valori molto elevati, quindi i parametri identificati avrebbero poca consistenza, dato che nel sitema piccole variazioni di $\mathbf{U}$ in (40) darebbero forti variazioni delle uscite $\mathbf{Y}$, invalidando l'identificazione effettuata. Una tecnica che ci permette di evitare simili problematiche è l'analisi delle componenti principali (**PCA**). Questa tecnica si basa sull'analisi degli autovettori e degli autovalori della matrice di convarianza dei dati, oppure equivalentemente sulla *decomposizione a valori singolari* (SVD). In breve, si ha un vettore

$$\mathbf{X} = \{\mathbf{x_1}; \mathbf{x_2} ... \mathbf{x_m}\}$$

in cui ogni componente $\mathbf{x_i}$ ha dimensione n, e la matrice $\mathbf{X}$ non é a rango pieno. Si vuole ridurre la dimensione m di $\mathbf{X}$, ovvero trovare una trasformazione lineare:

$$\mathbf{KLT} : R^{mxn} \rightarrow R^{lxn}$$

$$\mathbf{X} \longmapsto \bar{\mathbf{X}}$$

in modo che la matrice $\bar{\mathbf{X}}$ contenga le stesse informazioni di $\mathbf{X}$ ma sia di rango pieno, mettendo in evidenza le differenti caratteristiche tra i vettori delle misure.

FINE SESSIONE INTRODUTTIVA

IX. Metodi di regressione multivariati

I metodi di regressione multivariata sono strumenti di regressione nati dall'esigenza di superare i limiti della regressione ai minimi quadrati nel trattare variabili predittive fortemente correlate tra loro e affette da rumore. Le tecniche statistiche multivariate più conosciute sono la Proiezione su Strutture Latenti (PLS), e la Analisi delle Componenti Principali (PCA). I metodi di regressione multivariata vengono definiti anche come *Latent variable Multivariate Regression* in quanto presuppongono esista una struttura sottostante i dati descritta dalle variabili latenti; i metodi PLS e PCA sono strumenti per catturare questa struttura (Maggiori dettagli in[8]). compito dei metodi PCA e PLS é:

1) determinare la dimensione effettiva del sistema e cogliere la struttura delle correlazioni tra variabili, questo avviene mediante la definizione delle variabili latenti, tali variabili sono in numero inferiore alle variabili originali e tra loro ortogonali, esse costituiscono le direzioni del nuovo iperpiano, di dimensione ridotte rispetto alle dimensioni del processo, ma comunque in grado di descrivere i dati.
2) l'estrazione delle informazioni utili dai dati disponibili e la proiezione di tali informazioni lungo le direzioni prestabilite dalle variabili latenti.

Rispetto ad un metodo OLS classico, i metodi PCA e PLS presentano molti vantaggi,(vedi [9],[10],[11])

1) Sono capaci di gestire l'inclusione nel modello di variabili altamente correlate tra loro. In questo modo si evita di dover escludere una variabile importante dal modello solo perchè fortemente correlata con un'altra variabile già compresa nel modello, perdendo parte dell'informazione essenziale di un processo. Questo è un vantaggio rispetto al metodo OLS, che invece è incapace di gestire variabili tra loro correlate a causa della non invertibilità dell'inversa di una matrice singolare.
2) riducono le dimensioni del problema
3) permettono la regressione anche con dati di variabili predittive affette da errore, cosa non possibile con la regressione ai minimi quadrati classici,
4) minimizzano il rumore presente nelle variabili latenti mediando il rapporto segnale/rumore proveniente dalle variabili preditive.
5) forniscono stime più robuste dei coefficienti di regressione parziale.

X. Analisi delle Componenti Principali

Si riprende qui l'Analisi alle Componenti Principali, per capire e spiegare come funziona PLS.
L'Analisi delle Componenti Principali (PCA) individua, all'interno di un set di variabili, la struttura correlativa esistente tra le variabili stesse ed esplicita la dimensione effettiva dello spazio rappresentato da queste variabili mediante la costruzione delle variabili latenti. Data la matrice X dei dati delle variabili di processo, di dimensione n x k, dove n sono i campioni o misure disponibili e k le variabili predittive, il metodo PCA calcola dei vettori, detti componenti principali(PC) o variabili latenti (LV), che esprimono le direzioni di massima variabilità complessiva dei dati disponibili, cioè le direzioni lungo le quali si dispongono la maggior parte delle traiettorie delle variabili. Le variabili sono, per costruzione ortogonali tra loro; la prima variabile latente indica la direzione di massima variabilità delle variabili, la seconda variabile latente indica la seconda direzione di massima variabilità ortogonale alla prima. Per ottenere le componenti principali, il metodo PCA crea due nuovi vettori per ogni PC:

- un vettore *loading,* indicato con p ;
- un vettore *score,*indicato con il simbolo t.

Per ottenere i vettori *loadings*e gli *scores* si scompone la matrice X dei dati in una serie di k matrici di rango 1, ognuna delle quali rappresenta una componente principale e al cui somma è la matrice X stessa. Ognuna di queste matrici è il prodotto vettoriale dei due vettori di *loading* e *score* di ogni PC, formalmente:

$$X = t_1 p_1^T + \ldots t_a p_a^T + \ldots + t_k p_k^T = TP^T,$$

Con T e P matrici degli *score* e dei *loadings* ordinati come vettori colonna. Come detto, l'informazione ottenuta dalle variabili di processo è corrotta da rumore, di conseguenza non tutte le k direzioni principali saranno necessarie per rappresentare l'informazione contenuta nei dati. Se vengono mantenute le prime A direzioni principali o variabili latenti, la matrici X può essere rappresentata come:

$$X = t_1 p_1^T + \ldots + t_a p_a^T + t_r p_r^T + E_A = TP^T + E_A \quad (42)$$

con $A < k$ ed E_A matrice dell'informazione residua. Si presume che l'informazione utile sia contenuta nelle prime A variabili latenti, e il rumore nella matrice residua E_A.[6] L'equazione (42) può essere così interpretata: il processo monitorato attraverso le variabili contenute nella matrice X può essere descritto attraverso A variabili fittizie artificiali, dette variabili latenti, le quali assicurano di rappresentare la maggior parte della variabilità utile del processo e di non essere correlate tra loro grazie all'ortogonalità dei *loadings*. Le variabili latenti costituiscono le nuove variabili del sistema e sono combinazione lineare delle variabili predittive, colonne di X. Le variabili latenti hanno un'importanza diversa nel modello, proporzionale alla quantità di variabilità della matrice X che riescono a spiegare. Al vettore dei *loadings* e degli *scores* è possibile associare un'interpretazione geometrica:

- La matrice dei *loadings*

  $$P = [_1, \ldots, p_a, \ldots p_A]$$

  ha dimensioni A x p ed ogni elemento p_{aj} rappresenta il contributo della j-esima variabile predittiva alla a-esima variabile latente; geometricamente, p_{aj} è il coseno, compreso tra 0 e 1 che quantifica l'angolo di rotazione della a-esima variabile latente rispetto alla j-esima variabile predittiva nello spazio originario. In pratica ogni vettore p_a, di dimensioni 1 x p, definisce una nuova direzione nello spazio cartesiano in funzione delle variabili originali.

- la matrice degli *scores*

  $$T = [t_1, \ldots, t_a, \ldots, t_A]$$

  ha dimensione n x A; gli *scores* sono il risultato della proiezione della matrice X lungo le direzioni principali individuate dai vettori *loadings*, secondo la relazione

  $$T = XP.$$

[6] In realtà questo raramente accade, la determinazione del numero opportuno di variabili latenti da includere in un modello PLS rimane una questione ancora irrisolta. In assenza di certezze, il metodo più comunemente impiegato rimane quello della *cross-validation* (maggiori dettagli in [15])

A. SVD

Si consideri una matrice nxk $\mathbf{X}$ e sia $t = min(n, k)$. La decomposizione a valori singolari SVD di $\mathbf{X}$ é la fattorizzazione:

$$X = UDP^T$$

dove

1) $\mathbf{U}$ é una matrice n x t ortogonale [7];
2) $\mathbf{P}$ é una matrice kxt ortogonale;
3) $\mathbf{D}$=diag $\{d_1, ..., d_t\}$é una matrice diagonale con elementi $d_1 \geq ... \geq d_t \geq 0$;
4) Gli $r \leq t$ valori non nulli della diagonale della matrice $\mathbf{D}$ sono i valori singolari di $\mathbf{X}$;
5) Gli ultimi $t - r$ valori della matrice $\mathbf{D}$ possono esser scartati come le ultime $t - r$ colonne delle matrici $\mathbf{U}$ e $\mathbf{P}$.

Si ha infatti:

$$X = \begin{bmatrix} U1 & U2 \end{bmatrix} \cdot \begin{bmatrix} D1 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} P1^T \\ P2^T \end{bmatrix} \simeq U1 \cdot D1 \cdot P1$$

La matrice P é usualmente chiamata **matrice dei loadings**, e le sue colonne $p_1, ..., p_t$ sono i loadings. La matrice $\mathbf{T} = \mathbf{UD}$ viene chiamata invece **matrice degli scores**, e le sue rispettive colonne $t_1, ..., t_t$ scores.

B. Decomposizione autovalori-autovettori

Si consideri una matrice quadrata $\mathbf{A}$ di dimensioni n x n, e si consideri la sua decomposizione autovalori-autovettori:

$$A = P\Lambda P^T = \Sigma_{i=1}^{n} \lambda_i p_i p_i^T$$

Si ha che P é matrice ortogonale di colonne $p_1, ..., p_n$, e si consideri la matrice degli autovettori Λ in modo che gli autovettori siano ordinati in ordine crescente:

$$\lambda_1 \geq ... \geq \lambda_n$$

Si consideri una fattorizzazione di $\mathbf{A}$ come $A = X^T X$ e si consideri la decomposizione SVD di $\mathbf{X}$ trattata in precedenza.Risulta:

$$A = X^T X = PDU^T UDP = PD^2 P$$

dove si é sfruttata la proprietá di ortogonalitá della matrice U. Risulta che i valori sulla diagonale principale della matrice D^2, rispettivamente $d_1^2, ..., d_r^2$ sono gli autovalori non nulli della matrice XX^T o $X^T X$, entrambe di rango r. Le colonne di $\mathbf{P}$ sono autovettori di $X^T X$, corrispondenti agli autovalori non nulli. [8]

[7]ricordiamo che una matrice U é ortogonale se vale: $U \cdot U^T = I$, con I matrice identitá di dimensione n x n. Questo porta che, per le matrici ortogonali, $U^{-1} = U^T$

[8]**N.B.**: se X rappresenta un set di dati centrati, la matrice $X^T X/(n-1)$ é la matrice di covarianza di X.

C. *PCA*

La metodologia con cui si analizza la matrice di covarianza di un set di dati mediante decomposizione autovalori-autovettori e SVD viene denominata **Principal Component Analysis** (PCA). Tale metodologia sará utilizzata in seguito come strumento di analisi su dati multivariati. PCA puó esser considerato infatti uno strumento per analizzare la struttura di un insieme di dati multivariati, in particolare con il proposito di ridurne la dimensione: infatti, tramite PCA é possibile prendere i dati e proiettarli/ruotarli in uno spazio di dimensione minore, selezionando le direzioni nello spazio dei dati che forniscono la massima variazione, o equivalentemente, la maggior informazione. [9]
Intuitivamente, si posson riprendere alcune nozioni trattate in precedenza, partendo dalla decomposizione della matrice $\mathbf{X^T X}$ come $\mathbf{X^T X} = P\Lambda P^T$ e la decomposizione SVD di $\mathbf{X} = UDP^T$.Ricordando che P é stata definita *matrice dei loadings* e $T = UD = XP$[10] *matrice degli scores*, si ha che :

$$\mathbf{X} = t_1 p_1^T + ... + t_r p_r^T + \mathbf{E}$$

con $\mathbf{E}$ matrice contenente il 'rumore residuo', ovvero la parte della matrice $\mathbf{X}$ che le r componenti non riescono a spiegare. **QUESTO LO METTEREI DOPO, SUL CASO REALE, ORA CASO IDEALE SENZA RUMORE**. Geometricamente, si puó dire che:

- i singoli loadings p_i rappresentano delle direzioni, ortogonali tra loro, in uno spazio di dimensione r, ogniuna delle quali é data in funzione delle variabili originali;
- la matrice degli scores $\mathbf{T}$ é il risultato della proiezione della matrice $\mathbf{X}$ sullo spazio definito dalla matrice $\mathbf{P}$ dei loadings

D. Scelta del numero di componenti principali

Come presentato fin qui, PCA permette di selezionare le componenti principali di una matrice $\mathbf{X}$ e di decomporla come:

$$\mathbf{X} = \mathbf{T}_g \mathbf{P}_g^T + \mathbf{X}_{g+1}$$

dove $\mathbf{T}_g \mathbf{P}_g^T$ rappresenta la parte rilevante della matrice, mentre $\mathbf{X}_{g+1}$ è la parte trascurabile dei residui dati dagli eventuali $r - g$ termini rimanenti[11]. La problematica che si pone a questo punto, alquanto rilevante per i successivi trattamenti della matrice, è trovare una metodologia di scelta di g, ovvero quante componenti principali si devono selezionare in $\mathbf{X}$ per il problema che si sta trattando. Esiste un ampia scala metodologie di scelta di g, che variano a seconda della difficoltà di implementazione e del grado più o meno buono dell'ottimalità della soluzione che forniscono (vedi PRESS). Un semplice metodo di scelta si basa sugli autovalori di $\mathbf{X}$. Gli autovalori vengono normalizzati in valore e ad ognuno

[9]Questo tipo di approccio sará utilizzato in seguito per evitare vari problemi indotti dalla *collinearitá* dei dati.

[10]sfruttando la proprietá di ortonormalitá di P e sostituendo UD con T nella decomposizione SVD di $\mathbf{X}$;

[11]trascurabili , si intende, rispettivamente al problema che si sta affrontando. Si ricorda che r è pari al numero totale delle componenti principali di $\mathbf{X}$

viene assegnato un peso percentuale della somma di tutti gli autovalori:

$$peso(\lambda_i) = \frac{\lambda_i}{\lambda_1 + ... + \lambda_r} \times 100$$

e tale percentuale viene interpretata come la *variazione percentuale spiegata* dalla corrispondente componente principale. Tali pesi vengono quindi usati in modo cumulativo, così che il peso dato dalle prime g componenti è dato da:

$$\frac{\lambda_1 + \lambda_2 + ... + \lambda_g}{\lambda_1 + ... + \lambda_r} \times 100$$

Viene quindi fissata una regola di scelta di g come l'insieme di componenti che *spiegano* il *tot%* della matrice $\mathbf{X}$:in questo modo la scelta del numero di componenti da selezionare può esser automatizzata al variare della percentuale data. Un buon valore di percentuale da scegliere si assesta tra l'80 e il 90%. Una giustificazione a questa scelta può esser facilmente spiegata: se si considera la varianza del j-esimo score $\mathbf{t}_j$ come

$$s^2(\mathbf{t}_j) = \frac{1}{n-1} \|\mathbf{t}_j\|^2 = \frac{1}{n-1} \mathbf{t}_j^T \cdot \mathbf{t}_j = \frac{1}{n-1} \lambda_j$$

si nota come λ_j sia proporzionale alla varianza del corrispondente score. In particolare, tutte le componenti che hanno il rispettivo λ_j nullo non danno alcun peso e posson esser quindi scartate. Considerando quindi la varianza matriciale di $\mathbf{X}$

$$\mathbf{V}_x = \frac{1}{n-1} \mathbf{X}^T \mathbf{X} = \frac{1}{n-1} \sum_{k}^{j=1} \lambda_j \mathbf{p}_j^T \mathbf{p}_j$$

si può considerare λ_j come contributo (pesato) di $\mathbf{t}_j$ alla covarianza totale di $\mathbf{X}$ [12].

XI. PCR

PCR é l'acronimo di *Principal Component Regression*, ovvero regressione delle componenti principali. Con tale metodo si intende prelevare le componenti principali da una matrice di dati e regredire tali componenti mediante metodi lineari o minimi quadrati. A questo scopo si considerano le componenti principali di una matrice $\mathbf{X^T X}$, dove $\mathbf{X}$ é una matrice n x k di dati centrati. Ci sono fondamentalmente due metodologie per estrarre le componenti principali dalla matrice $\mathbf{X^T X}$, perfettamente equivalenti : SVD e decomposizione autovalori-autovettori (*eigenvalue decomposition*). Applicando il primo dei due metodi, si ottiene la forma ridotta:

$$X = U_r D_r P_r^T$$

dove con r si voglion indicare le matrici risultanti da SVD tolte le colonne/righe coorrispondenti agli ultimi $t - r$ autovalori nulli. Si definiscono gli *scores* come:

$$T_r = U_r D_r$$

una matrice con le colonne ortogonali, ma non necessariamente ortonormali [13]. Da $X = TP^T$ si ha che:

$$X^T X = P_r T_r^T T_r P_r^T = P_r \Lambda_r P_r^T$$

che é la decomposizione autovalori-autovettori di $X^T X$, in cui son state eliminate le colonne di P relative agli autovalori nulli di Λ.

Si può perciò pensare che la matrice $\mathbf{X}$ sia scomposta come:

$$\mathbf{X} = \mathbf{T}_g \mathbf{P}_g^T$$

e le caratteristiche importanti di $\mathbf{X}$ sian racchiuse in $\mathbf{T}_g$, matrice degli scores delle prime g componenti principali. Si ipotizza che $\mathbf{X}$ sia la matrice contenente m ingressi campionati in n istanti differenti, e $\mathbf{Y}$ sia la relativa matrice delle p uscite di un sistema lineare del tipo:

$$\mathbf{Y} = \mathbf{X} \cdot \mathbf{C} + \mathbf{F}$$

con $\mathbf{F}$ rumore del sistema.

Si puó a questo punto compiere *PCR*, ovvero *Multiple Linear Regression* utilizzando peró non l'intera matrice $\mathbf{X}$, ma $\mathbf{T}_g$ in luogo di questa. Si ottiene dunque:

$$\mathbf{Y} = \mathbf{T}_g \cdot \mathbf{C} + \mathbf{F_g} \tag{43}$$

e la relativa soluzione ai minimi quadrati é data da:

$$\hat{\mathbf{C}} = \left(\mathbf{T}_g^T \mathbf{T}_g\right)^{-1} \mathbf{T}_g^T \mathbf{Y} \tag{44}$$

Nel caso che si sta trattando, tale approccio va esteso, perché non adatto al problema: si vuole infatti trovare un sistema **dimamico** e non un legame **statico** tra ingressi e uscite del sistema. L'idea é dunque quella di compiere *PCA* sui dati d'ingresso e passare quindi ad una routine di identificazione a sottospazi l'insieme delle uscite e gli scores degli ingressi in luogo di questi ultimi. In questo modo, infatti, le problematiche trattate in **??**, comuni anche per tali tecniche ad identificazione, vengono a mancare permettendo così di ottenere un sistema più robusto e flessibile, sia dal punto di vista predittivo che in catena aperta. In realtà, si desidera un sistema in grado di accettare m ingressi, pari all'insieme totale di ingressi di cui si dispone inizialmente, e non l'insieme g ridotto.
Si può pensare quindi che se inizialmente il sistema si presenta nella forma:

$$\begin{cases} \mathbf{x}(t+1) = A\mathbf{x}(t) + B\mathbf{u}(t); \\ \mathbf{y}(t) = C\mathbf{x}(t) + D\mathbf{u}(t); \end{cases} \tag{45}$$

con $A \in R^{n*n}$,$B \in R^{n*m}$,$C \in R^{p*n}$ e $D \in R^{p*m}$, dati i loading $\mathbf{P}_g$ ricavati tramite il metodo *PCA* descritto in **??**, il sistema possa essere visto nella forma:

$$\begin{cases} \mathbf{x}(t+1) = A\mathbf{x}(t) + \hat{B}\mathbf{u}(t); \\ \mathbf{y}(t) = C\mathbf{x}(t) + \hat{D}\mathbf{u}(t); \end{cases} \tag{46}$$

Si ricorda infatti che da **??** è stata introdotta una trasformazione lineare sugli ingressi $\mathbf{u}(t)$ che ha portato una riduzione della dimensione dei processi (aleatori) in gioco.

[12] Si sottolinea come la somma degli autovalori sia pari alla varianza scalare di $\mathbf{X}$:

$$tr\left\{\mathbf{X}^T \mathbf{X}\right\} = (n-1)\left\{s^2(\mathbf{x}_1) + ... + s^2(\mathbf{x}_k)\right\}$$

[13] Infatti,

$$T^T T = DU^T UD = D^2 = \Lambda$$

Tale trasformazione è data dalla *matrice dei loading*, ed è possibile affermare quindi che le nuove matrici introdotte in 46 diventano [14]:

$$\begin{cases} B = B\mathbf{P}_g; \\ \hat{D} = D\mathbf{P}_g; \end{cases}$$

XII. IDENTIFICAZIONE A SOTTOSPAZI:PROBLEMATICHE E AFFINI

L'ottimalità della soluzione, ovvero il miglior sistema identificato possibile con un sottoinsieme di dati, verrebbe raggiunta comparando tutti i sistemi indentificati scegliendo di volta in volta tutte le possibili combinazioni di ingressi possibili per il sistema stesso. Naturalmente, a causa dell'elevato numero di sensori presenti, non è possibile procedere con questa metodologia: la complessità di un'algoritmo che risolve questo problema sarebbe difatti esponenziale. Nel precedente lavoro, son state affrontate le tecniche di selezione ingressi/uscite con metodologie greedy o con algoritmi genetici. Queste tipologie di tecniche non forniscono però una soluzione ottima, bensì partono con una soluzione sub-ottima cercando di migliorarla nell'intorno. L'obbiettivo è riuscire a compiere un'identificazione a sottospazi migliore, selezionando i dati in ingresso mediante algoritmi più intelligenti, che sfruttino strumenti quali l'analisi delle componenti principali.

METTERE MARCO ??

A. Algoritmo NIPALS

Descriveremo l'algoritmo NIPALS che ci permette di ottenere i vettori degli *scores* e dei *loadings* così come indicato in [14],[11]:

1) prendere un vettore colonna, cioè una variabile x_j di X e denominarla t_a: $x_j = t_a$ e porre $E_0 = X$
2) calcolare il vettore *loading* corrispondente:
$$p_a = Xt_a/t_a^T t_a$$
questo equivale a risolvere una regressione ai minimi quadrati delle colonne di X su t_a senza problemi di correlazione, perchè $t_a^T t_a$ è uno scalare; in questo modo si cattura anche la direzione di massima variabilità della matrice X. Successivamente si aggiornerà la matrice X con la matrice residua E_a.
3) normalizzare p_a a norma unitaria: $p_a = p_a/\left\|p_a\right\|$;
4) ricalcolare il vettore *score*:
$$t_a = Xp_a/p_a^T p_a$$
anche in questo caso si ha una regressione ai minimi quadrati senza problemi di inversione, perchè $p_a^T p_a$; è uno scalare
5) confrontare t_a nel punto 2 con t_a al punto 4: se la differenza relativa è nulla (con uno scarto di $10^{-8} - 10^{-6}$), passare al punto 6, altrimenti tornare al punto 2;
6) calcolare i residui della matrice X: $E_a = E_{a-1} - t_a p_a^T$

[14]si intende che la trattazione venga fatta tenendo conto di presentare le matrici degli ingressi come *numero processi* per *numero campioni*.In questo caso infatti si ha che in **??** l'equazione diventa

$$\mathbf{T}_g^T = \mathbf{P}_g \cdot \mathbf{X}^T$$

;

B. Algoritmo NIPALS per PCA

NIPALS é l'acronimo di *Nonlinear Iterative PArtial Least Square*, e si tratta di un algoritmo che implementa l'analisi delle componenti principali. Si vogliono estrarre le prime g componenti principali di $X^T X$, iniziando con l'autovalore piú grande (λ_1) e analizzando via via gli autovalori in maniera decrescente[15].

1) PSEUDOCODICE: L'algoritmo NIPALS viene inizializzato con $E_j = X$ e $j = 1$.Successivamente vengono compiuti i seguenti passi:

1) viene scelta una colonna qualsiasi di $\mathbf{X}_j$ e se ne assegna il valore a $\mathbf{t}_j$, impostandolo quindi come j-esimo *score*;
2) si calcola quindi il relativo loading $\mathbf{p}_j$ rispetto a $\mathbf{X}_j$, come $\mathbf{p}_j$
$$\mathbf{p}_j = \mathbf{X}_j\mathbf{t}_j/\left\|\mathbf{X}_j\mathbf{t}_j\right\|$$
3) si ricalcola $\mathbf{t}_j$ come proiezione di $\mathbf{X}_j$ su $\mathbf{p}_j$
$$\mathbf{t}_j = \mathbf{X}_j\mathbf{p}_j$$
;
4) se $\mathbf{t}_j$ è rimasto invariato dal passo 1 al passo 3, si continua, altrimenti si torna al passo 2;
5) si calcola il residuo:
$$\mathbf{X}_{j+1} = \mathbf{X}_j - \mathbf{t}_j\mathbf{p}_j^T$$
;
6) se $j = g$ allora si ferma l'algoritmo, altrimenti $j = j+1$ e si prosegue iterando dal passo 1.

Se $g = r$ allora l'algoritmo trova tutte le componenti principali. A questo punto le matrici $\mathbf{T}$ e $\mathbf{P}$ sono conformi alla relazione:

$$\mathbf{X} = \mathbf{T}\mathbf{P}^T \tag{47}$$

2) Proprietà: Per comprendere meglio come agisce l'algoritmo, vengono analizzati i passi in dettaglio, utilizzando le proprietà sopracitate per le matrici di *loadings* e *scores*. Si ponga $\lambda_j = \left\|\mathbf{X}_j\mathbf{t}_j\right\|$ e si scriva il passo 2 dell'algoritmo come:

$$\mathbf{X}_j\mathbf{t}_j = \lambda_j\mathbf{p}_j$$

Dalla relazione del passo 3 segue:

$$\mathbf{X}_j^T\mathbf{X}_j\mathbf{p}_j = \lambda_j\mathbf{p}_j$$

equazione che viene soddisfatta iterando dal passo 2 al passo 4 fino a che questi convergano ad una soluzione in norma d'errore minima. Da qui si nota facilmente come λ_j e $\mathbf{p}_j$ siano rispettivamente gli autovalori e gli autovettori della matrice $\mathbf{X}_j^T\mathbf{X}_j$.
Si dimostra ora come, nel passo 5, si formi una matrice $\mathbf{X}_{j+1} = \mathbf{X}_j - \mathbf{t}_j\mathbf{p}_j^T$ che è ortogonale a $\mathbf{t}_j$, j-esimo score. Si ha infatti:

$$\left(\mathbf{X}_j - \mathbf{t}_j\mathbf{p}_j^T\right)^T \cdot \mathbf{t}_j = \mathbf{X}_j^T\mathbf{t}_j - \mathbf{p}_j\mathbf{t}_j^T\mathbf{t}_j = \mathbf{X}_j^T\mathbf{X}_j\mathbf{p}_j - \mathbf{p}_j\lambda_j = \lambda_j\mathbf{p}_j - \mathbf{p}_j\lambda_j =$$

Si può dunque facilmente comprendere come, prendendo $\mathbf{t}_{j+1}$, una colonna a caso di $\mathbf{X}_{j+1}$ nel passo 1 dell'algoritmo, questa

[15]La metodologia con la quale si definisce il numero g di componenti principali da selezionare verrá trattata in seguito. Naturalmente dev'essere $g \leq r$

sia ortogonale, qualunque essa sia, ai precedenti scores $\mathbf{t}_j$ selezionati. Dopo g cicli si avrà dunque:

$$\mathbf{X} = \mathbf{t}_1\mathbf{p}_1 + \mathbf{t}_2\mathbf{p}_2 + ... + \mathbf{t}_g\mathbf{p}_g + \mathbf{X}_{g+1}$$

con $\mathbf{X}_{g+1} = \mathbf{0}$ se son state selezionate tutte le componenti, ovvero $g = r$.

XIII. Proiezioni su strutture latenti: PLS

La proiezione su strutture latenti (PLS) è un metodo di regressione che costruisce un'equazione di regressione lineare tra gli *scores* delle variabili predittive e gli *scores* delle variabili dipendenti; analogamente al metodo PCA, il metodo PLS decompone sia la matrice delle variabili predittive X che la matrice delle variabili dipendenti Y, creando direzioni principali che descrivano la massima variabilità di X cogliendo allo stesso tempo la massima covarianza tra gli *scores* di X e di Y. Le due matrici X e Y possono essere decomposte con due PCA secondo le relazioni esterne:

$$X = TP^T + E,$$
$$Y = UQ^T + F.$$

Se si utilizzano gli *scores* di X come variabili predittive e gli *scores* di Y come nuove variabili dipendenti, è possibile ottenere una equazione di regressione interna tra gli *scores* cioè si pone

$$U = Tb_{pls} + F^*$$

dove il vettore $b_{pls} = [b_1, \ldots, b_a, \ldots, b_A]^T$ è il vettore dei coefficienti di regressione tra le variabili latenti, F^* è la matrice degli scarti e b_{pls} è stato ottenuto con la regressione ai minimi quadrati tra gli *scores* di X e di Y:

$$b_{a,pls} = u_a^T t_a / t_a^T t_a.$$

e

$$\hat{u} = b_{a,PLS} t_a.$$

Il modello così calcolato non è però il migliore possibile, perchè non sempre gli *scores* di X e Y sono correlati tra loro; si può quindi ottenere un'equazione di regressione che rappresenta solo una minima parte della variabilità di Y o che non minimizza F^*. Si è quindi cercato di decomporre simultaneamente le matrici X e Y, ruotando parzialmente gli *scores* t_a in modo da renderli più correlati agli *scores* u_a. Per fare questo si sono introdotti anche i vettori *weights*, simili nel significato ai vettori *loadings*. I *loadings* e gli *scores* sono calcolati con l'algoritmo NIPALS, adattato alla regressione PLS:

1) prendere un vettore colonna o meglio una variabile y_j di Y e denominarlo $u_a : y_j = u_a$;
2) calcolare il vettore *weight* X corrispondente:

   $$w_a = Xu_a / u_a^T u_a$$

   regredendo X su u_a trovo la relazione lineare che permette di spiegare una variabile dipendente, u_a utilizzando l'informazione contenuta nella matrice delle variabili predittive. L'utilizzo del modello ai minimi quadrati, inoltre, garantisce che ciò è stato fatto prendendo la direzione di massima variabilità dei dati di X e minimizzando la matrice residua F^* .
3) normalizzare $w_a = w_a / \|w_a\|$;
4) calcolare il vettore *score* di $X : t_a = Xw_a / w_a^T w_a$;
5) calcolare il vettore *loading* di $Y : q_a = Yt_a / t_a^T t_a$;
6) normalizzare q_a a norma unitaria: $q_a = q_a / \|q_a\|$;
7) ricalcolare il vettore *score* di $Y : u_a = Yq_a / q_a^T q_a$;
8) confrontare il valore di t_a al punto 4 tra un'iterazione e l'altra: se la differenza relativa è nulla (con uno scarto di $10^{-8} \div 10^{-6}$) passare al punto 9, altrimenti tornare al punto 2;
9) calcolare il vettore *loading* di

   $$X : p_a = Xt_a / t_a^T t_a;$$

10) ortogonalizzare il *weight*:

    $$w_a = w_a p_a / p_a^T p_a;$$

11) calcolare il coefficiente di regressione:

    $$b_{a,pls} = u_a^T t_a / t_a^T t_a;$$

    la relazione interna è data da:

    $$\hat{u_a} = b_{a,pls} t_a;$$

    ma la forma degli *scores* e dei *loadings* è cambiata;
12) aggiornare le variabili:

    $$E_a = E_{a-1} - t_a p_a^T;$$
    $$F_a = F_{a-1} - u_a q_a^T;$$
    $$F_a^* = F_{a-1}^* - \hat{u_a} q_a^T;$$

In pratica si combinano gli algoritmi di decomposizione PCA per le due matrici X e Y tra loro, scambiando informazioni tra gli *scores* e i *loadings* delle variabili predittive e dipendenti, di modo che sia X che Y siano modellate dalle stesse variabili latenti. La matrice F^* minimizza grazie alla regressione ai minimi quadrati, la differenza tra errori calcolati e sperimentali: $F^* = Y - TQ^T$.
A differenza dell'algoritmo PCA gli *scores* t sono combinazioni lineari delle variabili predittive di X pesate secondo i *weights* w^*, e non secondo i *loadings* p.

$$T = XW^*;$$

La differenza tra *weights* e *loadings* della matrice X è che: i primi forniscono i pesi di ogni variabile predittiva a formare la variabile latente, i secondi trovano la direzione della variabile latente e spiegano la varianza delle singole matrici; a differenza del PCA, le due caretteristiche, contributo alla variabile latente e formazione della direzione principale, non coincidono.
Si nota inoltre che esiste una differenza tra i *weights* w e w^* legati dalla relazione: $W^* = W(P^T W)^{-1}$.
Un modello PLS opportunamente costruito contiene variabili latenti che descrivono la varianza di X e Y con valori decrescenti e cumulativamente simili; se ad esempio, un modello spiega la maggior parte della varianza di Y ma

poca varianza di X , significa che la matrice X contiene variabili poco correlate ad Y, che eventualmente danneggiano la modellizzazione.
La regressione condotta con metodo PLS è rigorosamente lineare: per questo motivo, spesso vengono suggeriti metodi per eliminare il carattere dinamico delle traiettorie, come il centramento, la normalizzazione, la trasformazione di variabili.

A. Interpretazione geometrica di PLS

PLS è un metodo di proiezione che geometricamente si può interpretare come la proiezione della matrice $\mathbf{X}$ (Un insieme di punti in uno spazio a K dimensioni) su un iperpiano A-dimensionale; in questo modo le coordinate dei proiettori (*scores*) $t_a: \quad a = 1, \ldots, A$ sono dei buoni predittori per $\mathbf{Y}$, figura 1. La direzione del piano è espressa come inclinazione, $\mathbf{p_{ak}}$, di ogni componente rispetto alle coordinate, $\mathbf{x_k}$. La pendenza è il coseno dell'angolo tra la direzione del piano di proiezione e l'asse delle ascisse.
PLS costruisce iterativamente un iperpiano nello spazio generato dai dati di $\mathbf{X}$; in questo modo tale iperpiano è la migliore approssimazione di $\mathbf{X}$, e allo stesso tempo, la posizione di dati proiettati sul piano, descritta dal valore degli *scores* $\mathbf{t_a}$ sono utilizzati per la predizione della risposta del sistema, $\mathbf{Y}$.

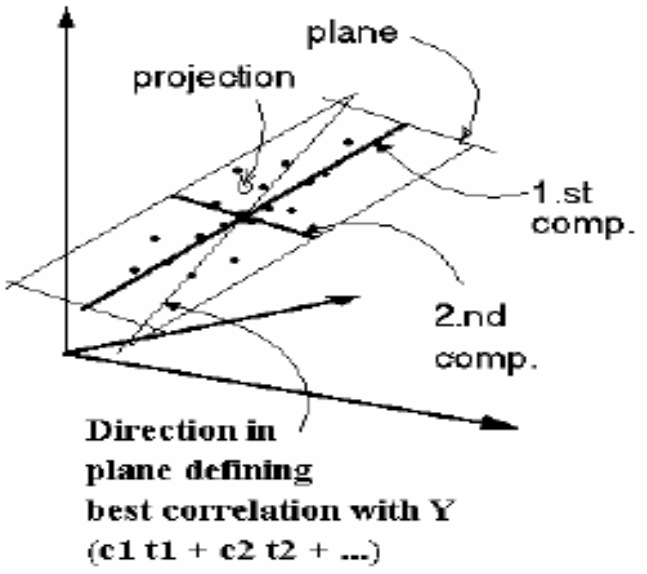

Fig. 1. Iperpiano generato dall'algoritmo PLS nello spazio dei punti di $\mathbf{X}$, si evidenzia l'ortogonalità delle componenti

1) Confronto geometrico tra OLS e PLS: La robustezza dell'algoritmo di regressione è riferita all'insensibilità del modello stimato al condizionamento errato dei dati e al rumore. Il grande vantaggio delle tecniche di proiezione su strutture latenti rispetto alla regressione lineare ordinaria sono illustrati geometricamente in figura 2 e in figura 3.
Tali figure evidenziano il caso di due ingressi e una uscita in presenza di collinearità dei dati e di rumore sovrapposto. Tutti i dati sono esattamente sulla stessa retta tranne un punto x, il quale è deviato per la presenza di rumore. Questi dati generano uno sottospazio a due dimensioni chiamiamolo, $\mathbf{X}$.
La regressione lineare ordinaria, proietta l'uscita $\mathbf{Y}$ ortogonalmente a $\mathbf{X}$. Purtroppo dato che il dato x è corrotto da rumore l'orientazione del piano $\mathbf{X}$ è pesantemente influenzata dalla collocazione del punto x. Il risultato è che la proiezione $\hat{Y}_{OLS}$ è altamente sensibile alla collocazione di x e quindi al rumore casuale.
Nel caso invece di algoritmo PLS si nota come sia richiesto un fattore, cioè una proiezione artogonale sul sottospazio unodimensionale $\mathbf{t_1}$ di $\mathbf{X}$. In questo caso la proiezione $\hat{Y}_{PLS}$ non è influenzata dalla collocazione di x, cioè è robusta al rumore. (maggiori dettagli in [19]).

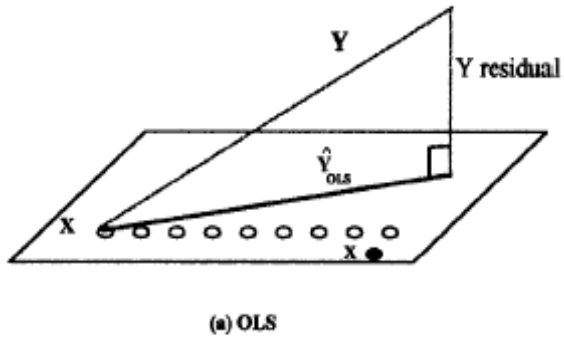

Fig. 2. Interpretazione geometrica di OLS. Un dato corrotto da rumore condiziona negativamente la regressione lineare

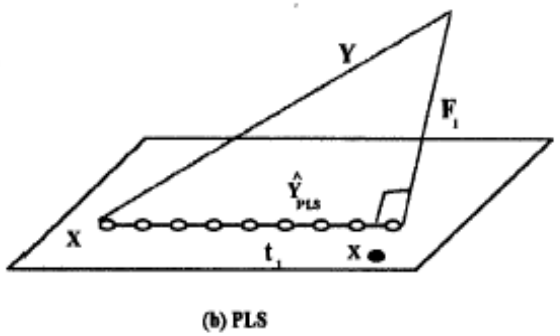

Fig. 3. Interpretazione geometrica di PLS. Il metodo PLS è robusto a collinearità dei dati e agli effetti del rumore sovrapposto

XIV. SELEZIONE DEI SENSORI CON PLS: CASO STATICO

Ciò che il nostro lavoro si propone, oltre al fatto di identificare un modello termodinamico è anche quello di ridurre il numero di sensori utilizzati mantenendo buona la qualità della stima della temperatura interna. La regressione PLS è negativamente influenzata dalla presenza di variabili predittive non correlate con la variabile dipendente. La presenza di variabili "inutili", rende necessario un maggior numero di variabili latenti per modellare i dati. Anche se questo metodo di regressione non è influenzato dalla presenza di variabili tra loro correlate, diventa necessario operare una scelta sulle variabili indipendenti, in modo da selezionare quelle maggiormente esplicative della variabile dipendente e aventi un elevato rapporto segnale/rumore, in modo da rendere le stime di qualità del prodotto più robuste e il modello meno influenzato

dal rischio di avarie dei sensori. La scelta delle variabili è contrastata tra due esigenze:

1) la mecessità di rappresentare nel modo migliore possibile i dati sperimentali, minimizzando il MSE, porterebbe a introdurre nel modello il maggior numero possibilie di variabili predittive. Con questo criterio vengono utilizzate variabili tra loro correlate nel modello e può condurre a *overfitting*, perchè introduce eccessive variabili nel modello che spiegano solo la parte della varianza dovuta al rumore e impedisce l'estrapolazione dei risultati dell'equazione di regressione al di fuori dell'intervallo in cui sono stati calibrati i dati;
2) la necassità di ottenere un modello il più possibile semplice e che minimizzi la varianza media del dato calcolato $var(\hat{y})$ richiederebbe di inserire ill minor numero possibile di variabili predittive.

Oltre a queste esigenze, di tipo statistico, si desidera che le variabili predittive prescelte siano ottenibili con apparecchiature poco costose e soggette a piccoli errori di misura e rumore. Esistono inn letteratura diverse soluzioni per risolvere problemi diversi:

- Quale criterio si deve adottare per determinare la superiorità di un'equazione di regressione rispetto ad un'altra e il sottoinsieme "migliore" di variabili predittive che determina il modello. Il criterio migliore rimane la minimizzazione dell'errore di predizione MSEP. Questo criterio, però pone il problema della distinzione tra modellizzazione dell'informazione e del rumore: un'equazione può riprodurre ottimamente i dati reali, ma essere inutile perchè dii fatto riproduce ile oscillazioni casuali introdotte dalla strumentazione di misura nei dati raccolti;
- Quale algoritmo è capace di selezionare il set più appropriato di variabili in modo da risulatare superiore agli altri rispetto al criterio di giudizio scelto, sapendo che la selezione dell'insieme ottimo dipende anche dallo stimatore che si intende costruire.

A. Selezione di variabili per PLS: VIP

In letteratura sono stati svilippati diversi metodi di eliminazione di variabili implementati sull'algoritmo NIPALS, tra cui: OSC(*Orthogonal Scatter Correction*) sviluppato in [16]; O-PLS (*orthogonal-PLS*) sviluppato in [17].
Tra quelli proposti il più utilizzato è il VIP, acronimo di *Variable Importance in Projection,* trattato in [18] il quale calcola un parametro di importanza per ogni variabile predittiva in base al valore pesato di ogni *loadings* nelle variabili latenti secondo:

$$VIP_j = \sqrt{\frac{p\sum_{a=1}^{A}\left(b_a^2 t_a^T t_a \left(w_{aj}/\left\|w_{aj}\right\|\right)^2\right)}{\sum_{a=1}^{A}\left(b_a^2 t_a^T t_a\right)}}$$

Tutte le variabili aventi il criterio $VIP_j > 1$ sono ammesse al set ridotto. VIP effettua sempre le stesse scelte indipendentemente dal rapporto segnale/rumore delle variabili, mentre varia se cambiano il numero di variabili latenti utilizzate dal modello. Il limite intrinseco di questi metodi è quello di costiture una relazione *statica* di ricerca delle variabili "ottime" per costruire un modello ridotto, noi abbiamo già lavorato in questa direzione ottenendo buoni risultati con l'analisi delle componenti principali e la relativa regressione, cercheremo invece di costruire un metodo dinamico che sfrutti le potenzialità di PLS per identificare un modello e scegliere dinamicamente un set opportuno di sensori pre creare un modello ridotto adatto al controllo e che dia buoni risultati inn fase di predizione.

XV. PLS: CASO DINAMICO, MODELLO ARX

Si esamina in questa sessione l'utilizzo e l'implementazione di un algoritmo di PLS dinamico. I medoti di selezione delle variabili fin qui analizzati come PCA permettono di ottenre delle informazioni sul numero di ingressi e uscite utilizzati in fase di identificazione attraverso un procedimento di carattere statico. Si cercherà qui invece di costruire una relazione dinamica che permetta di ridurre la dimensione del problema mantenendo tuttavia la qualità del modello identificato. Il modello ottenuto con questo tipo di tecnica sarà un modello ARX diversamente da quanto fatto con metodi PCA.
Per testarne l'acuratezza si utilizzeranno metodi di validazione comuni. Descriviamo l'idea proposta da Qin J. in [12] per costruire un modello ARX utilizzando dinamicamente l'algoritmo PLS e le sue potenzialità. Nei processi di identificazione caratterizzati da una grande mole di dati, l'algoritmo PLS viene utilizzato per creare un modello dei che spieghi bene i dati. Questo tipo di approccio ha delle forti limitazioni:

- il modello che ne risulta è complicato da aggiornare e difficilmente riesce ad inseguire i cambiamenti dinamici del modello,
- quando si devono manipolare grosse quantità di dati, con sistemi MIMO, come nel nostro caso la disponibilità di memoria fisica di un calcolatore diviene un fattore limitante.

Si cerca quindi di modificare l'approccio statico dell'algoritmo PLS a vantaggio di un algoritmo ricorsivo per un'identificazione dinamica del modello. L'idea è di fondere assieme due concetti, quello della regressione su strutture latenti di dati altamente correlati e affetti da rumore, e la possibiltà di operare con un modello ARX che tiene conto di n ritardi temporali antecedente all'istante $k-esimo$, in modo da avere una struttura dinamica. Diversi benefici vengono raggiunti utilizzando un algoritmo ricorsivo di PLS:

1) Il PLS ricorsivo aggiorna il modello quando un nuovo campione è disponibile,
2) Il PLS ricorsivo ha l'abilità di adattare la struttura correlativa del modello a seconda della dinamica del processo,
3) Il PLS ricorsivo tende a comprimere l'informazione e contenuta nei dati e risolvere il problema della disponibilità di memoria.

Un semplice modello stocastico auto-regressivo con ingressi esogeni è dato da:

$$\mathbf{y(k)} = \sum_{i=1}^{n_y} \mathbf{A_i y}(k-i) + \sum_{j=1}^{n_u} \mathbf{B_j u}(k-j) + \mathbf{v}(k); \qquad (48)$$

dove $\mathbf{y}(k)$, $\mathbf{u}(k)$, e $\mathbf{v}(k)$ sono il processo di uscita, di ingresso e il vettore di rumore con appropriate dimensioni per un sistema multi-input, multi-output.
A_i e B_j sono matrici dei soefficienti del modello che devono essere identificati. n_y, e n_u, sono i ritardi, o meglio quanti campioni precedenti all'istante $k-esimo$ vengono considerati per l'uscita e l'ingresso.
Per poter elaborare i dati con l'algoritmo PLS è necesasario definire il vettore:

$$\mathbf{x}^T(k) = \left[\mathbf{y}^T(k-1), \mathbf{y}^T(k-2), \ldots, \mathbf{y}^T(k-n_y), \right. \\ \left. \mathbf{u}^T(k-1), \mathbf{u}^T(k-2), \ldots, \mathbf{u}^T(k-n_u)\right. \quad (49)$$

dove la dimensione è m, cioè la somma di tutte le uscite e gli ingressi del sistema per il rispettivo ordine di ritardo n_y e n_u. Noi nella fase sperimentale cercheremo di trovare l'ordine ottimo del modello, ponendoci in questa prima fase nel caso in cui $n_y = n_u = n_{uy}$.
Una volta definita la matrice dei dati in questo modo si si costruiscono le matrici per n campioni:

$$\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(n)]^T \quad \in \mathbf{R}^{nxm}$$

$$\mathbf{Y} = [\mathbf{y}(1), \mathbf{y}(2), \ldots, \mathbf{y}(n)]^T \quad \in \mathbf{R}^{nxp}$$

dove p è la dimensione del vettore delle uscite $\mathbf{y}(k)$.
Definiamo la matrice dei valori da identificare del modello ARX come,

$$\mathbf{C} = \left[\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_{n_y}, \mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_{n_u}\right]^T \quad \in \mathbf{R}^{mxp},$$

le due matrici dei dati $\mathbf{Y}$, $\mathbf{X}$ sono in relazione secondo:

$$\mathbf{Y} = \mathbf{XC} + \mathbf{V}.$$

Ciò che l'algoritmo PLS restituisce e che ci serviranno per ricostruire la matrice $\mathbf{C}$ dei coefficienti del modello ARX, sono le matrici,

$$\mathbf{T}_n = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_m],$$

$$\mathbf{P}_n = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_m],$$

$$\mathbf{B}_n = diag(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m),$$

$$\mathbf{Q}_n = [\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_m].$$

[16] Le matrici di ingresso e di uscita possono essere riscritte in relazione alle matrici ottenute con PLS inn modo che:

$$\mathbf{X}_n = \mathbf{T}_n \mathbf{P}_n^T$$

$$\mathbf{Y}_n = \mathbf{T}_n \mathbf{B}_n \mathbf{Q}_n^T + \mathbf{F}_n$$

Dato che il numero di fattori considerato in questo caso è pari alla dimensione dell'ingresso, il residuo per la matrice $\mathbf{X}_n$ è scomparso, mentre in genarale la matrice residua $\mathbf{F}_n$ non è nulla per la presenza di rumore. Come già discusso se la matrice di covarianza $\mathbf{X}_n^T \mathbf{X}_n^T$ è singolare, e la soluzione ai minimi quadrati è mal condizionata. Per minimizzare il residuo nell'equazione di uscita utilizziamo il criterio ai minimi quadrati generalizzandolo nella forma:

$$(\mathbf{X}_n^T \mathbf{X}_n^T)\mathbf{C}_n = \mathbf{X}_n^T \mathbf{Y}_n \quad (50)$$

dato che le colonne di $\mathbf{T}_n$ sono mutuamente ortogonali, si può scrivere:

$$(\mathbf{X}_n^T \mathbf{X}_n) = \mathbf{P}_n \mathbf{T}_\mathbf{n}^\mathbf{T} \mathbf{T}_n \mathbf{P}_n^T = \mathbf{P}_n \mathbf{P}_n^T$$

$$\mathbf{X}_n^T \mathbf{Y}_n^T = \mathbf{X}_n^T \mathbf{T}_n \mathbf{B}_n \mathbf{Q}_n^T + \mathbf{X}_n^T \mathbf{F}_n = \mathbf{P}_n \mathbf{B}_n \mathbf{Q}_n^T \quad (51)$$

Il fatto che la matrice residua $\mathbf{F}_n$ è ortogonale a $\mathbf{X}_n$ nella regressione ai minimi quadrati è usato per derivare l'equazione (51). Sostituendo le precedenti relazioni nell'equazione (50), otteniamo:

$$(\mathbf{P}_n \mathbf{P}_n^T)\mathbf{C}_n = \mathbf{P}_n \mathbf{B}_n \mathbf{Q}_n^T$$

se la matrice di covarianza $\mathbf{X}^T\mathbf{X}$ è singolare la matrice $\mathbf{P}\mathbf{P}_n^T$ è pure singolare. Per cui la matrice dei coefficienti del modello ARX ottenuta con l'algoritmo PLS è data generalizzando l'inversa di una matrice, attraverso la pseudo-inversa:

$$\mathbf{C}_n^{PLS} = (\mathbf{P}_n \mathbf{B}_n^T)^+ \mathbf{P}_n \mathbf{B}_n \mathbf{Q}_n^T.$$

Si è così arrivati ad identificare il modello attraverso una procedura di tipo dinamico, che sfrutta le potenzialità dell'azione dell'algoritmo di PLS nel ricercare gli *scores* di massima predizione di

A. Scelta della complessità del modello ARX

Nell'algoritmo di identificazione analizzato precedentemente vi sono due parametri che giocano un ruolo chiave nella complessità di calcolo e nella bontà di stima del modello identificato, sono:

- numero di variabili latenti da considerare,
- numero di campioni precedenti l'istante k-esimo che vengono considerati nel modello ARX, nel nostro caso $n_u = n_y = n_{uy}$.

1) Numero di variabili latenti: Nell'analisi del numero di variabili latenti da considerare nel modello vengono utilizzati i criteri MSEP (Mean Square Error of Prediction) e MSECV (Mean Square Error of Cross-validation) che sono computazionalmente identici, ma calcolati sui dati di convalida o attraverso la cross-validation dei dati di calibrazione. (maggiori dettagli in [15] e in [21]) un altro criterio di scelta è il PRESS (PRediction Error Sum of Squares) che si ottiene come somma dei quadrati delle differenze tra ogni osservazione y_i e il valore calcolato $\hat{y}_{(i)}$ ottenuto dall'equazione di regressione costruita su tutti i punti tranne y_i. Formalmente

$$PRESS = \sum_{i=1}^{n} (y_i - \hat{y}_{(i)})^2$$

Il criterio di scelta indica come modello migliore quello che presenta il valore più piccolo di PRESS. Il criterio PRESS è molto simile a MSE e MSEP ma viene utilizzato solitamente nella cross-validation, infatti si ha

$$MSECV = PRESS/(n-k-1);$$

[16] Va notato che il numero dei fattori qui considerato è pari a alla dimensione dell'ingresso cioè m e corrisponde a condiderare tutte le variabili latenti calcolate dall'algoritmo PLS. Si cercherà nella fase sperimentale di limitare il numero di variabili latenti utilizzate attraverso i calcolo del RMSECV e RMSEP per diversi set di ingressi e uscite.

versioni diverse dello stesso criterio si hanno eliminando blocchi di dati invece che un singolo valore alla volta per la fase di calibrazione (dettagli in [11]).
E' da rilevare come la scelta del numero delle variabili latenti utilizzate in un modello PLS sia una questione ancora aperta in letteratura, lo dimostra la grande mole di articoli reperibili, ma tutti caratterizzati da una diversa impostazione delle ipotesi di lavoro e di verifica, in relazione allo scopo per il quale si cerca di ridurre il numero di variabili latenti. Noi abbiamo sviluppato alcune funzioni in *Matlab* che analizzano il comportamento del sistema, in dettaglio sono: *ipls* che sfrutta delle sottofunzioni per calcolare il RMSE di ogni modello, in base agli intervalli di dati che si desidera considerare e al numero di variabili predittive del modello; e *myplscv* che calcola e grafica il RMSEP. A tal proposito va sottolineato che la dimensione del problema e quindi del numero di variabili latenti massimo che possiamo considerare nel modello è per noi variabile, in quanto nel caso dinamico è funzione del:

- numero di sensori di ingresso al sistema,
- numero di sensori di uscita considerato,
- ordine del modello ARX considerato n_{uy}.

Si riportano di seguito alcuni grafici ottenuti in fase di simulazione:

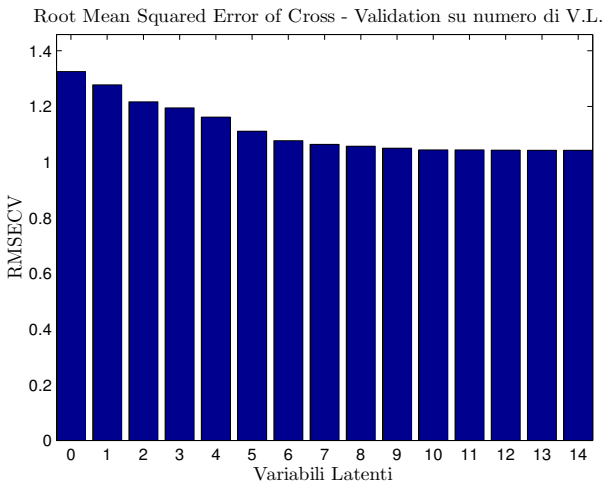

Fig. 4. Grafico di Root Mean Square Error of Cross - Validation con 14 ingressi e 54 uscite

In figura (4) è graficato il RMSECV nel caso di 14 ingressi e 54 uscite ottenuto applicando l'algoritmo in modo statico al set di dati disponibili. L'obiettivo è quello di un'analisi a priori della complessità del modello prima che i dati vengano organizzati in modo da formare la matrice $\mathbf{X} \in \mathbf{R}^{nxm}$ e di un successivo confronto.

La figura (5) è ottenuta considerando i dati come in figura (4) ma in questo caso viene calcolato il RMSEP che è un indice di predizione del modello PLS.

Le figure (6) e (7) graficano il RMSECV e il RMSEP nel caso descritto dall'algoritmo dinamico, sistemando cioè la matrice di ingresso del PLS come:

$$\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(n)]^T \quad \in \mathbf{R}^{nxm}$$

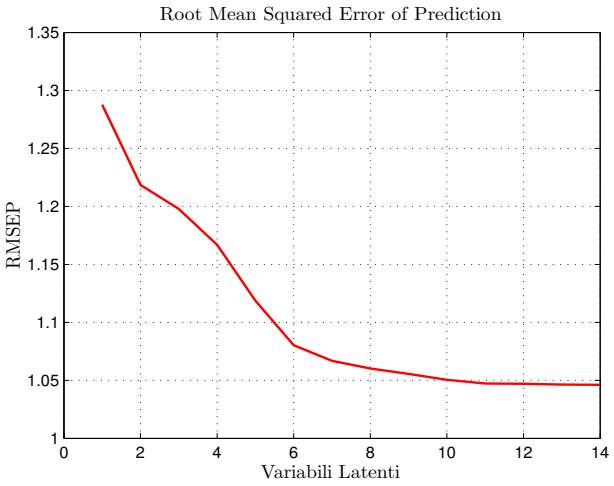

Fig. 5. Grafico di Root Mean Square Error of Prediction con 14 ingressi e 54 uscite

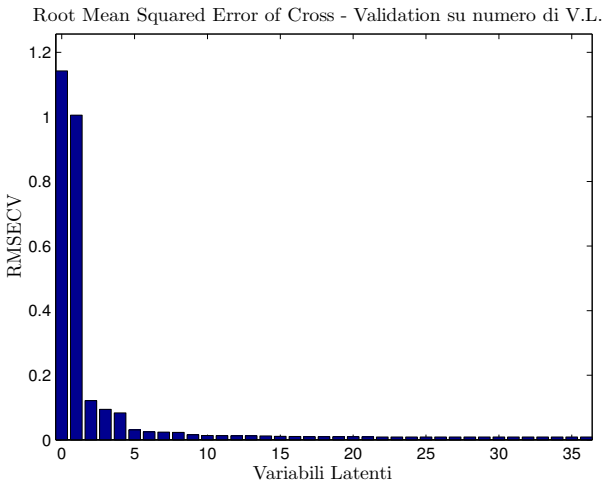

Fig. 6. RMSECV nel caso PLS dinamico troncato a circa 36 variabili latenti

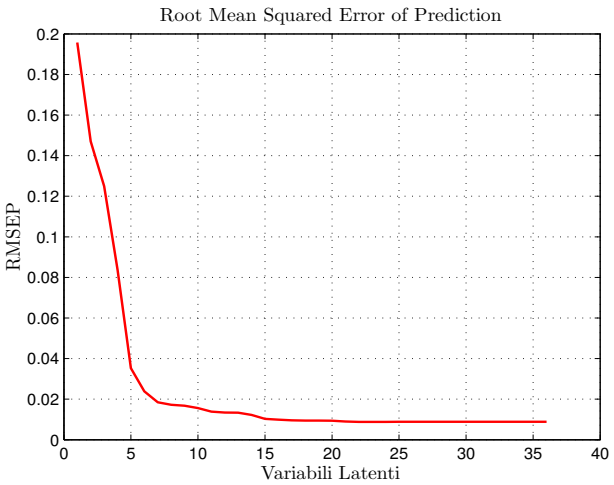

Fig. 7. RMSEP nel csao dinamico troncato a circa 36 variabili latenti

Confrontando il caso statico con il caso dinamico si nota come nel caso dinamico le prime $5 \div 6$ variabili latenti

spieghino ”meglio” il sistema di quanto non accada nel caso statico, infatti nel caso statico il RMSECV varia da circa 1,3 a 1 mentre nel caso dinamico la variazione va da circa 1,3 a 0.05, a dimostrazione del fatto che identificare il sistema con un procedimento dinamico porta a dei risultati più accurati.
Considerazioni analoghe possono essere fatte per il RMSEP.
Nel seguito identificheremo il sistema considerando le prime 5 o 10 variabili latenti inn modo da percepire la variazione nell'errore di identificazione.

2) Ordine del modello ARX, n_{uy}: La scelta dell'ordine da utlizzare nel modello ARX è uno dei passi dell'algoritmo PLS di non facile analisi in quanto:

- utilizzare la funzione *FPE* di *Matlab* richiede di disporre di un set di modelli ARX di ordine diverso costruiti con la funzione *ARX*, la quale crea una struttura per ogni modello contenente l'indice FPE del modello,
- considerando un modello MIMO di 14 ingressi reali e 54 uscite la memoria di sistema rappresenta il collo di bottiglia in fase di simulazione, di fatto rendendola impossibile.

Si è quindi cercato, in un primo momento un approccio diverso che permettesse di superare tali difficoltà.
Si sono simulati 20 modelli ARX di ordine crescente da $n_u = n_y = n_{uy} = 1$ a $n_u = n_y = n_{uy} = 20$ con 14 ingressi effettivi e 54 uscite. Si sono scelti 5 sensori campione per ognuno dei quali si è calcolato il MSE al variare dell'ordine del modello e si sono quindi graficati i risultati.

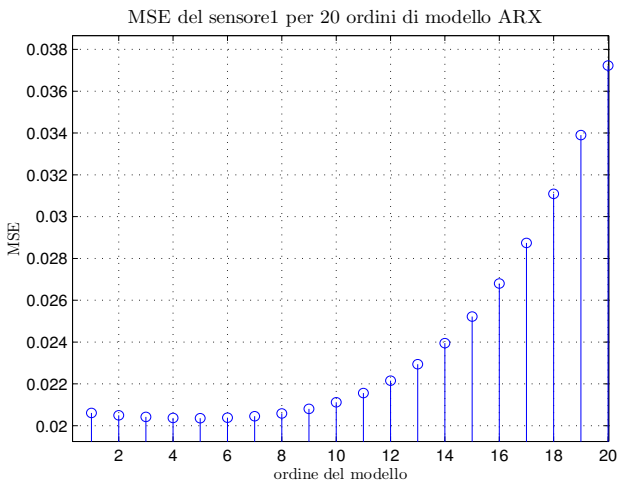

Fig. 8. MSE di un sensore per ordine di modello

Le figure(8, 9, 10, 11, 12) evidenziano come sia minima la variabilità del MSE tra i diversi modelli e in particolare come il minimo ordine vari in relazione al sensore scelto. Reputando scarso utilizzo solo di tale metodo si è cercato un modo per calcolare l'FPE dei diversi modelli identificati con l'algoritmo PLS dinamico. Decidendo di utilizzare le funzioni fornite nel IDtoolbox di *Matlab*, si è operato come segue:

1) identificato il modello attraverso il PLS dinamico,
2) estratte le matrici A e B dei coefficienti del modello ARX
3) risimulato le uscite del sistema identificato utilizzando come ingressi gli stessi ingressi di identificazione del PLS,

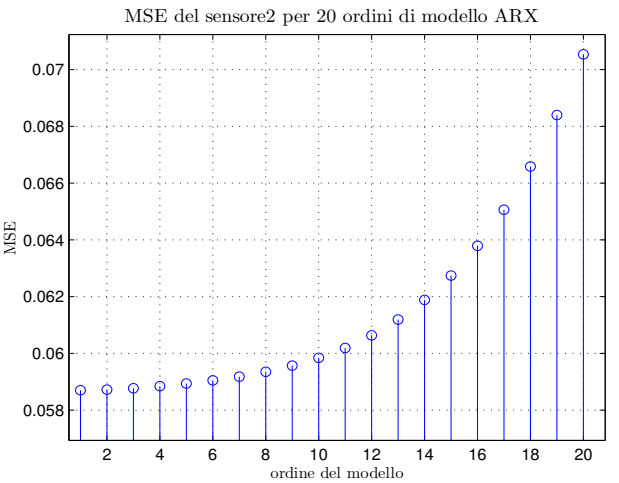

Fig. 9. MSE di un sensore per ordine di modello

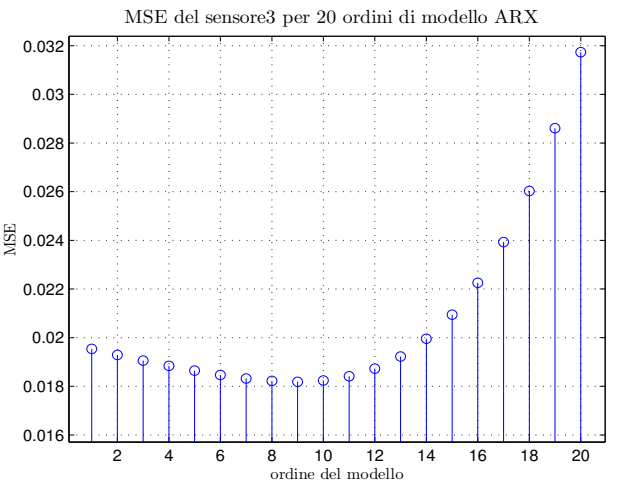

Fig. 10. MSE di un sensore per ordine di modello

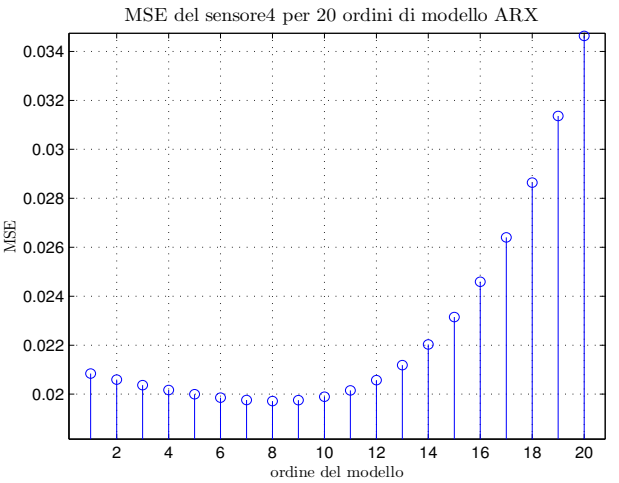

Fig. 11. MSE di un sensore per ordine di modello

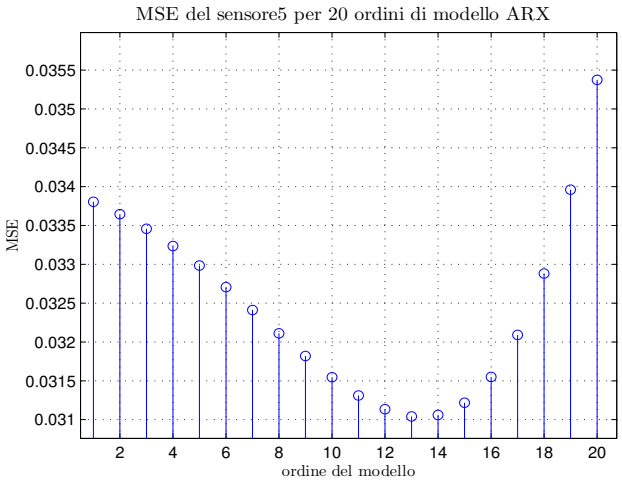

Fig. 12. MSE di un sensore per ordine di modello

4) a questo punto si sono identificati nuovamente i diversi modelli con la funzione *arx* utilizzando tutti gli ingressi di identificazione e come uscite quelle ottenute dalla simulazione del punto precedente.

Tale procedimento è stato eseguito su un set ridotto di 10 uscite in luogo di 54, questo per risolvere i problemi di disponibilità di memoria in fase di simulazione. L'analisi della figura (13) in relazioene ai precedenti grafici del MSE ci permette di dire che un buon ordine per il modello ARX è di $n_u = n_y = 9$.
Va tuttavia considerato che a priori ci si aspettava una maggiore variabilità nella bontà del modello in relazione all'ordine, osservazione parzialmente smentita sia dal calcolo di MSE che di FPE.

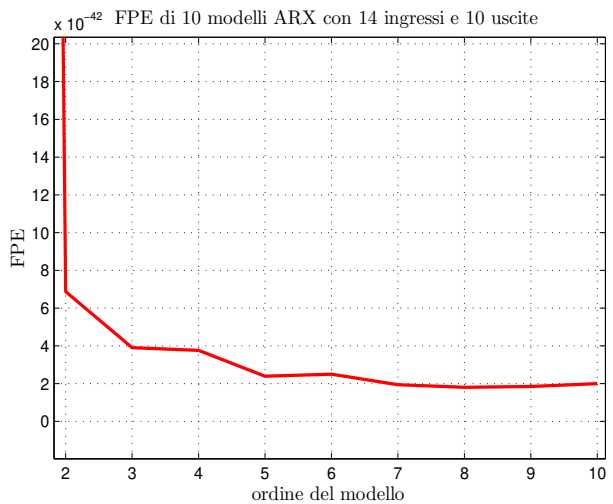

Fig. 13. Grafico dell'FPE per ordine di modello

XVI. PLS: Risultati sperimentali

Si presentano i risultati ottenuti identificando il modello con l'algoritmo PLS dinamico simulando i risultati su dati di validazione e variando il numero di variabili latenti utilizzate.

La funzione che calcola il modello ARX è *ARX_PLS* che sfrutta al suo interno la funzione *PLSF2*.
Va notato che nella funzione *ARX_PLS* la validazione del modello è stata ottenuta con due metodologie diverse:

1) Una utilizzando i dati di validazione e organizzandoli in modo da avere la matrice

$$\mathbf{X}_{\mathbf{VALIDAZIONE}} = [\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(n)]^T \quad \in \mathbf{R}^{nxm},$$

tale che l'uscita del sistema sia data come moltiplicazione per la matrice dei coefficienti
2) L'altra ricavando dalla matrice C_pls le due matrici dei coefficienti A e B che permettono di organizzare il modello ARX in forma canonica e simularne l'uscita utilizzando solo gli ingressi si validazione, $Uval$. Questa seconda metodologia rispecchia il reale utilizzo del modello anche per un futuro sviluppo di controllo.

Pur essendo la prima metodologia poco significativa si è scelto di inserirla ugualmente, in modo da confrontare il grado di acuratezza del modello in presenza di conoscenza a priori del valore dell'uscita negli istanti precedenti al k-esimo, e nel caso invece più realistico di disponibilità dei soli ingressi di validazione.
Di seguito si presentano i risultati ottenuti in fase di validazione settando:

- ordine del modello $n_u = n_y = 9$,
- utilizzando **5** variabili latenti

In figura(14) e figura(15) sono riportati i valori di MSE per tutti i 54 sensori di uscita, ottenuti attraverso i due metodi di validazione.

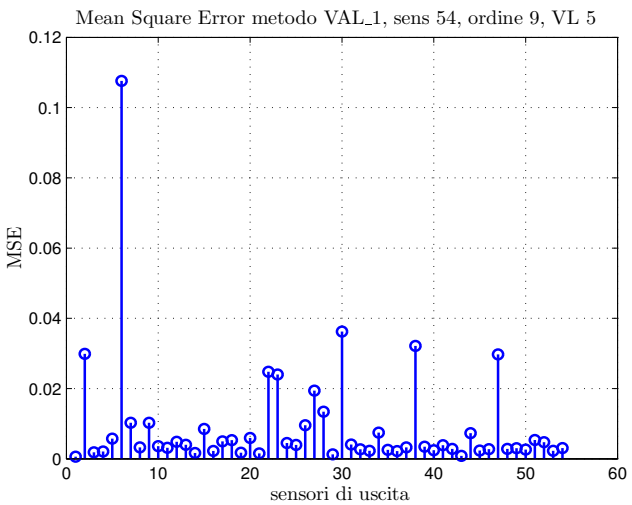

Fig. 14. MSE dei 54 sensori di uscita, **primo** metodo di validazione e **5 V.L.**

In figura(16) e in figura(17) sono riportati i grafici dell'andamento di 3 sensori scelti casualmente con il relativo MSE ottenuti sistemando i dati di validazione secondo il primo metodo.
In figura(19) e in figura(20) sono riportati i grafici dell'andamento di 3 sensori scelti casualmente con il relativo MSE ottenuti con il secondo metodo di validazione.
Di seguito si presentano i risultati ottenuti in fase di validazione settando:

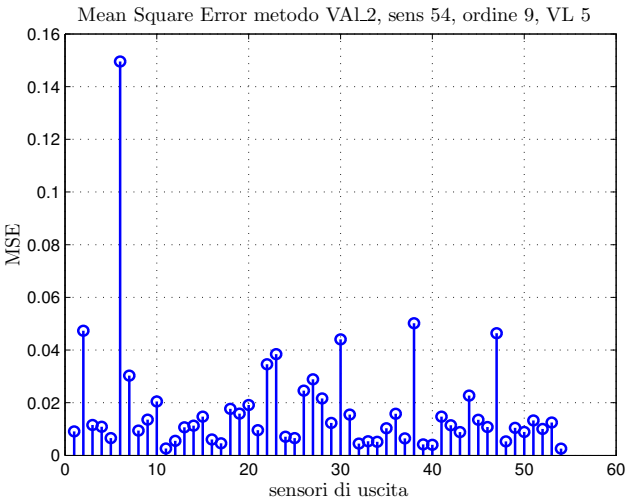

Fig. 15. MSE dei 54 sensori di uscita, **secondo** metodo di validazione e **5 V.L.**

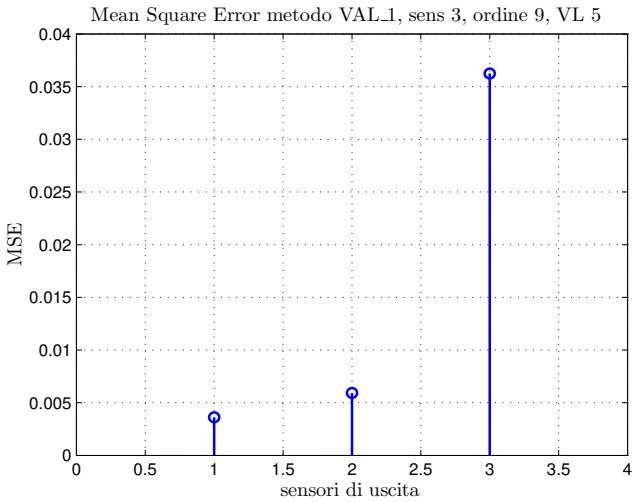

Fig. 17. MSE dei tre sensori di figura(16) relativo al metodo di **validazione 1** e **5 V.L.**

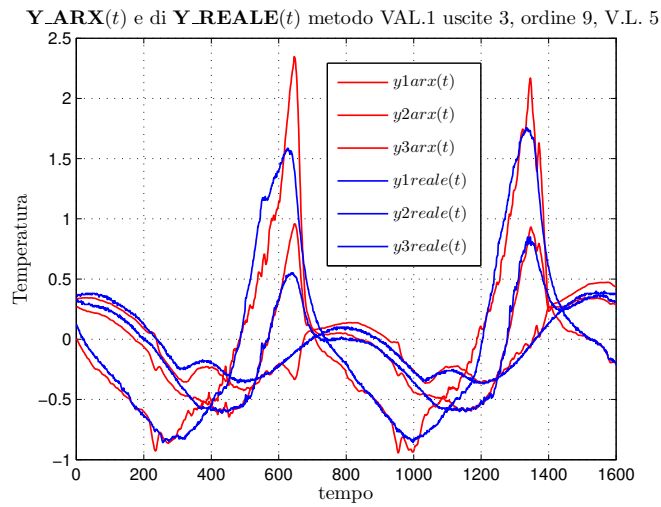

Fig. 16. Grafico di 3 sensori con andamento reale e in uscita dal modello ARX ottenuto con il **primo** metodo di validazione e **5 V.L.**

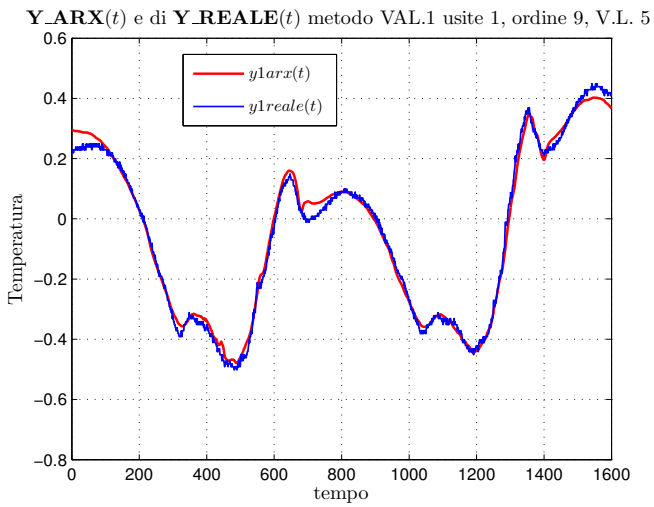

Fig. 18. Grafico di 1 sensore con andamento reale e in uscita dal modello ARX ottenuto con il **primo** metodo di validazione e **5 V.L.**

- ordine del modello $n_u = n_y = 9$,
- utilizzando **10** variabili latenti

In figura(22) e figura(23) è riportato l'andamento di 3 sensori scelti casualmente con il relativo MSE ottenuti con il secondo metodo di validazione.

XVII. Considerazioni

L'identificazione del modello avvenuta attraverso l'utilizzo di un algoritmo dinamico e sfruttando le potenzialità del PLS, ha il grande vantaggio di essere computazinalmente molto efficiente rispetto all'utilizzo della funzione di identificazione *arx* del IDToolbox di *Matlab* tanto più che tale funzione non porta ad alcun risultanto per scarsità di memoria di sistema in fase simulativa.[17]

Pur avendo ridotto la complessità del problema utilizzando di fatto al massimo 10 variabili latenti, il passo successivo è quello di costruire una relazione dinamica che permetta di limitare il numero di sensori utilizzati in fase di identificazione di un ordine di grandezza. L'obiettivo resta quello che la riduzione non deve peggiorare la qualità della stima della grandezza di interesse, nel nostro caso la temperatura interna. Gli sviluppi futuri quindi di tale modello è quello di porlo in forma di predizione come fatto per il caso statico con PCA e valutarne la bontà predittiva, costruendo e ricercando un modo dinamico di selezionare un set ridotto di sensori sia di ingresso che di uscita, catturando la massima variabilità possibile delle grandezze che ci interessano.

In questa fase la nostra idea è quella di combinare le due metodologie che abbiamo analizzato nel seguente modo:

1) Si esegue un'analisi delle componenti principali,
2) si riduce attraverso PCR il numero di sensori di ingresso,
3) si applica al set ridotto di sensori algoritmo di PLS dinamico,
4) si testa ill modello facendolo lavorare in forma di

[17] si dispone di un PC con 2 GByte di memoria RAM.

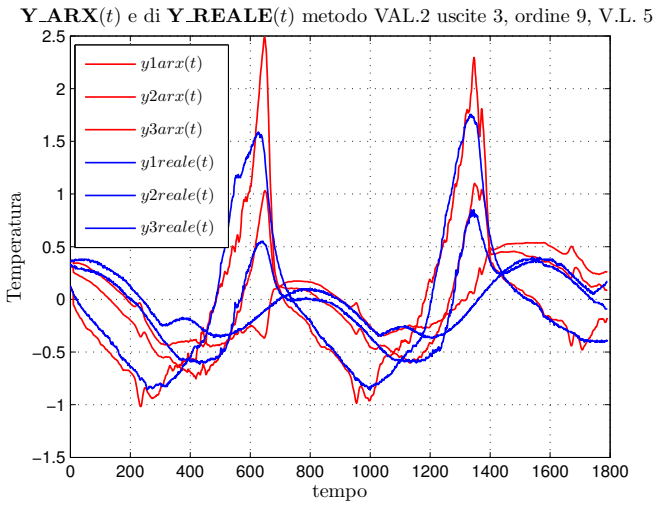

Fig. 19. Grafico di 3 sensori con andamento reale e in uscita dal modello ARX ottenuto con il **secondo** metodo di validazione e **5 V.L.**

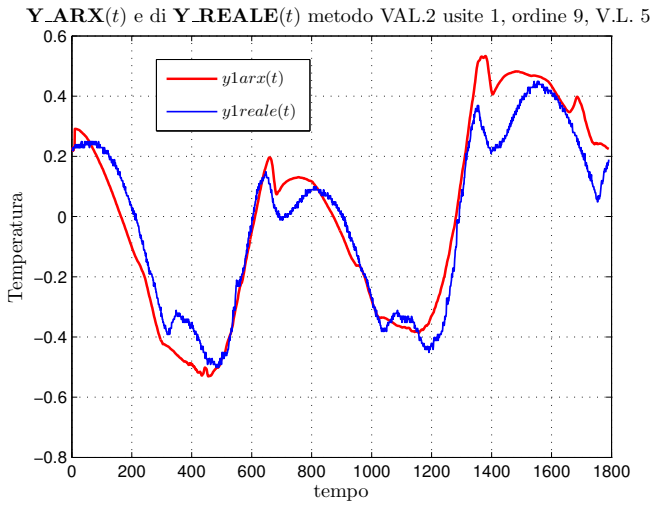

Fig. 21. Grafico di 1 sensore con andamento reale e in uscita dal modello ARX ottenuto con il **secondo** metodo di validazione e **5 V.L.**

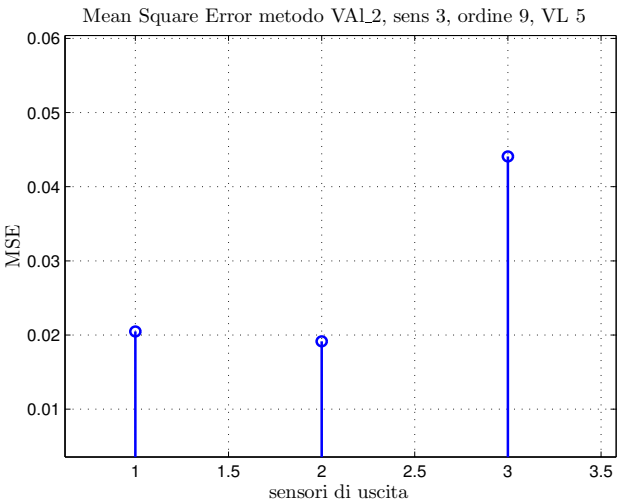

Fig. 20. MSE dei tre sensori di figura(19) relativo al metodo di **validazione 2** e **5 V.L.**

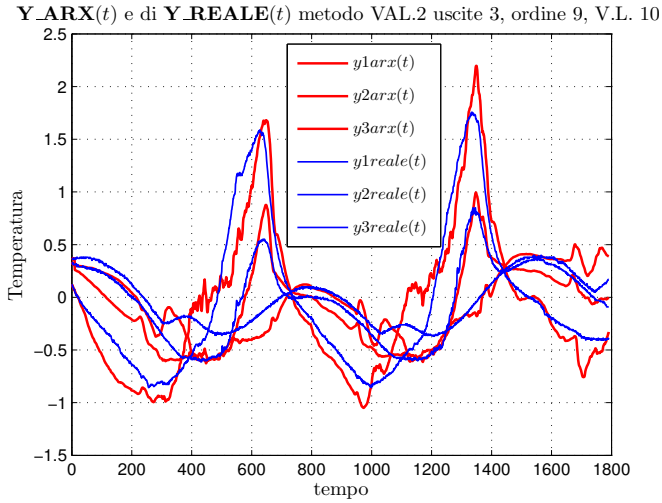

Fig. 22. Grafico di 3 sensori con andamento reale e in uscita dal modello ARX ottenuto con il **secondo** metodo di validazione e **10 V.L.**

predizione.

Segnaliamo come sviluppo futuro l'idea avuta da R. Muradore e F. Bezzo in [20] di fondere il concetto e le potenzialità del PLS con il metodo di identificazione a sottospazi, cioè la regressione statica del PLS viene sostituita da il metodo di identificazione a sottospazi creando un algoritmo dinamico di identificazione di un modello in spazio di stato con ingressi e uscite altamente correlati e affetti da rumore di processo. Va considerato che la portata teorica applicativa di questa idea è molto pesante e richiede una conoscenza dell'identificazione a sottospazi che al momento non possediamo. Altra questione è quella di poterlo implementare e testare nel nostro problema, passo anche questo di non facile attuazione. In letteratura si trovano soluzioni che risolvono il problema di ridurre il numero di sensori, utilizzando tecniche di posizionamento ottimo a priori per il successivo impiego di tecniche PLS, come in [22]. Questo tuttavia si allontana dalla nostra idea e dal nostro problema, noi cerchiamo un modo di ridurre i sensori utilizzati in fase di identificazione ma la cui collocazione è casuale, noi cerchiamo il set di sensori che porta il più alto grado di informazione per la fase di controllo e cerchiamo di farlo con un metodo dinamico.

XVIII. Conclusioni

Il lavoro svolto ha l'idea di fornire un quadro di insieme dello stato dell'arte nella soluzione del problema di identificazione termodinamica. Il lavoro si è concentrato nella ricerca di metodologie alternative rispetto al lavoro svolto lo scorso anno allo scopo di migliorare i risultati ottenuti sia in fase di identificazione che in fase predittiva. L'obiettivo che ci eravamo prefissi era quello di comparare parte dei risultati per capire dove le metodologie da noi applicate avevano reali vantaggi e dove invece il lavoro dello scorso anno permetteva di ottenere risultati più consistenti. Purtroppo un'analisi approfondita del lavoro dello scorso anno non è stata possibile, perchè mancava parte del codice di simulazione su

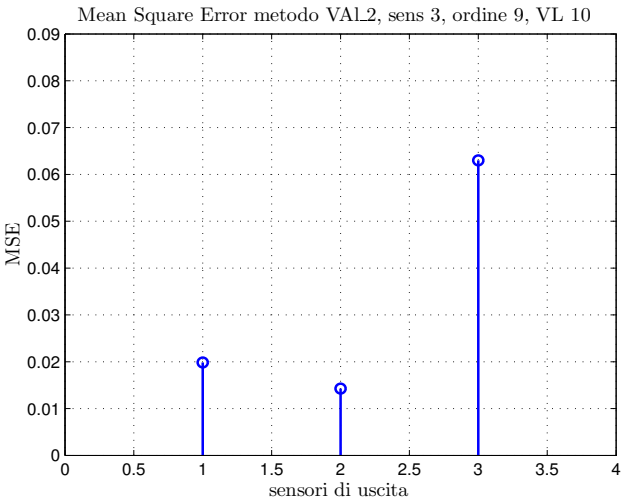

Fig. 23. MSE dei tre sensori di figura(22) relativo al metodo di **validazione 2** e **10 V.L.**

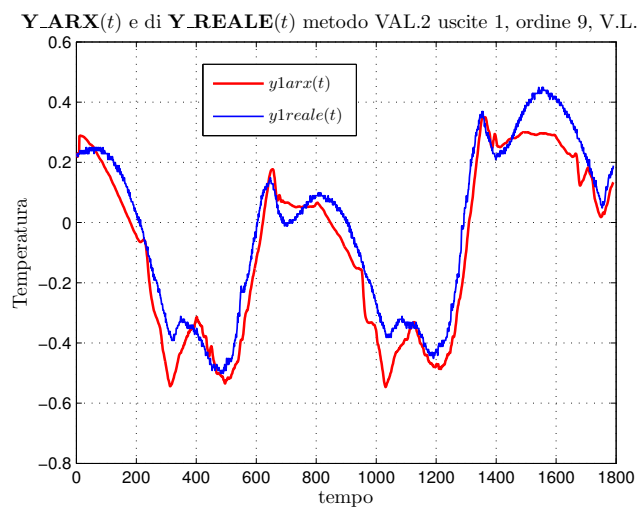

Fig. 24. Grafico di 1 sensori con andamento reale e in uscita dal modello ARX ottenuto con il **secondo** metodo di validazione e **10 V.L.**

cui venivano testati i metodi Greedy Search e Genetic Search, il tentativo di collaborare con il gruppo dello scorso anno per ripristinare le parti mancanti non è andato a buon fine.

XIX. Prove simulative

Nel caso che ci troviamo ad affrontare, un utilizzo delle tecniche fin'ora esposte é quanto mai necessario. Si é in presenza infatti di dati altamente correlati sia in ingresso che in uscita. Basta pensare infatti che

- per i *sensori interni* (*USCITE* del sistema): all'interno di una stanza, se come in questo caso le porte vengono tenute chiuse, le temperature misurate sono molto simili tra loro;

- per i *sensori esterni* (*INGRESSI* del sistema): all'esterno, le temperature dell'aria si posson ipotizzare molto simili tra loro; probabilmente l'irradiazione solare é la tipologia di ingressi selezionati che hanno maggior variabilitá, data la variazione di esposizione delle pareti della casa lungo l'arco della giornata.

É possibile affermare che, mentre la correlazione tra le uscite non crea nessun problema per quanto riguarda l'identificazione, la correlazione tra gli ingressi ne crea, e di molto influenti. Le problematiche sono quelle esposte in precedenza: se i dati sono dipendenti, la matrice degli ingressi non é invertibile, o lo é ma é anche molto vicina all'esser singolare, e questo porta a identificare un sistema molto sensibile e quindi inutilizzabile da ogni punto di vista.

Per tali simulazioni si sono utilizzate le misurazioni eseguite nell'anno precedente, prendendo in considerazione ingressi e uscite del sistema come trattati nel precedente progetto. Non è stato infatti possibile eseguire nuove misurazioni perchè la stuttura non è più agibile:tale procedura poteva esser di grande utilità perchè rilevazioni di misurazioni in un periodo dell'anno totalmente diverso dal precedente. In **??** viene riportato che le misurazioni son state eseguite tenendo le porte chiuse per ogni stanza:tale procedura è stata resa necessaria perchè la parte non-lineare di un sistema di interscambio di calore è dovuta al circolo di masse d'aria, ovvero la parte relativa alla convezione. La parte di procedura che non è stata compresa è perché si sono mantenuti chiusi gli scuri e tenute aperte le finestre:per le considerazioni fatte in precedenza ciò apporta spostamento di masse d'aria anche considerevoli, e induce perciò non linearità. Si sottolinea inoltre come nella maggior parte dei sistemi di simulazione venga considerato lo scambio di calore attraverso i vetri delle finestre e l'effetto che l'irradiazione solare provoca su queste per le variazioni di temperatura interne:era stato consigliato quindi di rieseguire le misurazioni tenendo finestre chiuse e scuri aperti, e tenendo tirate le tende per permettere ai raggi solari di penetrare nelle stanze. Inoltre, per il tempo che si disponeva, sarebbe stato interessante selezionare un periodo di grande variabilità climatica, allo scopo di trovare ingressi persistentemente eccitanti di ordine superiore a quello utilizzato: la settimana in cui sono state eseguite le rilevazioni, infatti, viene descritta per lo più con un tempo persistentemente sereno e con cambi di temperature per lo piú dolci, e non repentine. Si consiglia quindi, in caso del proseguimento di questo lavoro, di rieseguire le misurazioni in periodi di cambi stagionali (es. metà Febbraio-metà Marzo) in cui, almeno negli ultimi anni, si verificano molteplici variazioni di condizioni climatiche, anche all'interno di brevi periodi.

Si hanno quindi a disposizione:

- 53 (48) misure di temperatura interne, disposti uniformemente all'interno delle stanze (un sensore in ogni angolo + uno/due in centro per le stanze piú grandi);
- 14 (9) misure di temperatura e luminosita' esterne, suddivise in 5 di temperatura (uno per ogni facciata, 2 per il lato est della casa) e 4 sensori di radiazione solare.

Già nel lavoro precedente erano state rilevati degli errori dovuti per lo più a problemi hardware dei *T-mote*, che avevano costretto alla rimorzione di 2 sensori tra i 65 utilizzati. Da un esame più attento dei dati in uscita sono stati trovati altri sensori che soffrono di problematiche simili, e sono stati quindi rimosse dal set le misure relative ad altri 5 sensori **??**.

...figura... La differenza sostanziale che si vuole sottolineare da subito è che, mentre nel precedente progetto si è cercato di rimuovere "'fisicamente"' dei sensori, siano essi di ingresso o di uscita, tramite tecniche greedy, nel presente si sono utilizzate tecniche che consentono di rimuovere il "'ridondante"' dai set di misure, ma non per questo di rimuovere totalmente l'utilizzo di una specifica misura. Entrambe le tecniche che esporremo, infatti, riducono il set di ingresso o i set ingresso/uscita dei dati mediante trasformazioni lineari: i set di dati risultanti saranno frutto quindi di combinazioni lineari dell'insieme dei dati in considerazione, ma si continuerà a considerare comunque il set di dati nella sua interezza. In figura **??** viene riportata la disposizione fisica tridimensionale dei sensori, come specificato nella struttura dati fornita con il precedente progetto.

A. PCA: l'implementazione

Si é implementata una funzione denominata *KLT()* che compie *PCA* su un set di dati qualsiasi, e tale funzione é stata utilizzata principalmente per la riduzione delle variabili in ingresso. I passi che l'algoritmo esegue sono[18]:

- calcola le medie e le leva ai singoli vettori di misura;
- calcola la matrice di covarianza di $\mathbf{X}$ come:

$$Var_{\mathbf{X}} = \frac{1}{n-1}\mathbf{X}\mathbf{X}^T$$

;
- trova autovalori e autovettori di $Var_{\mathbf{X}}$:

$$Var_{\mathbf{X}} \cdot \mathbf{P} = \mathbf{P} \cdot \mathbf{D}^2$$

; dove $\mathbf{P}$[19] é la matrice contenente gli autovettori disposti sulle colonne, e $\mathbf{D}^2$ é la matrice diagonale contenente gli autovalori della matrice $Var_{\mathbf{X}}$;
- ordina in maniera decrescente gli autovalori, mantenendo l'ordinamento con i relativi autovettori;
- sceglie il numero g di componenti principali secondo una determinata metrica , ad esempio quella che pesa gli autovalori trattata in (**??**)
- seleziona le prime g colonne di $\mathbf{P}$, che rappresentano le direzioni lungo le quali i dati $\mathbf{X}$ hanno maggior rilevanza
- proietta i dati lungo le g direzioni date dai primi g autovettori:

$$\mathbf{T}_g = \mathbf{X} \cdot \mathbf{P}_g$$

.

La matrice risultante rappresenta gli *scores* di $\mathbf{X}$ e fungerá da ingresso per le routine di identificazione a sottospazi al posto dell'intera matrice $\mathbf{X}$. In questa maniera si ottiene un sistema in forma di stato come in **??** che ha peró g ingressi e p uscite:se si vuole mantenere il numero di ingressi pari a m si puó eseguire il procedimento precedentemente esposto in (**??**) Si considerano dunque i dati organizzati in maniera trasposta, ovvero $\mathbf{X} \in R^{kxn}$ dato come *processi* x campioni:la forma degli scores risulta quindi

$$\mathbf{T}_g^T = \mathbf{P}_g^T \cdot \mathbf{X}^T$$

e variando le matrici $\mathbf{B}$ e $\mathbf{D}$ come indicato in **??**i ottiene un sistema che accetta m ingressi.

[18]per l'implementazione della routine si é utilizzata la *scomposizione autovalori autovettori* della matrice di covarianza di$\mathbf{X}$ invece della *scomposizione SVD* della matrice dei dati $\mathbf{X}$, come presentato nella sezione —-: si é notato infatti che *eig()* si comporta computazionalmente molto meglio di *svd()* in presenza di una quantitá consistente di dati da trattare. In particolare, con la mole di dati a disposizione non è stato possibile compiere *SVD* perché necessita di troppo dispendio di memoria fisica (almeno l'implementazione fornita da `Matlab`).

[19]che corrisponde poi precisamente alla matrice dei loadings (rif.**??**)

XX. Conclusion

The conclusion goes here.

Appendix I
Caratteristiche tecniche principali di DesignBuilder:

- Interfaccia OPEN GL per grafica 3D fotorealistica.
- Motore di calcolo EnergyPlus v 1.4.
- Calcolo dinamico carichi termici e frigoriferi (ASHRAE Test certificate).
- Simulazione dei consumi di combustibile ed energia elettrica per periodi temporali a scelta.
- Collegamento a banche dati climatici mondiali.
- Calcolo della produzione della CO2 e verifica dei limiti di legge (Norma Inglese).
- Calcolo delle temperature reali nelle zone dell'edificio non climatizzate.
- Analisi corretto posizionamento pannelli solari e FV in funzione della posizione dinamica del sole nei vari periodi dell'anno.
- Archivi di strati elementari, strutture opache verticali e orizzontali, strutture vetrate verticali e orizzontali.
- Archivi di "Template" riferiti alle tipologie di attività sia di edifici,di impianti HVAC e di Illuminazione artificiale.
- Archivi riferiti a sistemi di schermatura interni ed esterni, ecct.
- Produzione di filmati AVI per simulazione ombre sull'edificio e rotazione del medesimo.
- Stampe in formato leggibile EXCEL e JPEG, Bitmap, DXF, ecct..
- Importazione/esportazione planimetrie DXF per agevolare costruzione edifici.
- Esportazione file per manipolazione esterna su EnergyPlus (casi complessi).
- Creazione zone termiche differenziate (attività,temperature,carichi endogeni,ecct).
- Check finale per Certificazione rendimento energetico edificio/impianto secondo normativa europea prEN 15217.

A. Tipici usi del programma sono:

- Valutare una serie di tipologie di facciate per quantificare gli effetti di surriscaldamento, utilizzo dell'energia e impatto visivo dell'edificio.
- Valutare l'uso ottimale della luce naturale. Modellare sistemi di controllo di illuminazione e calcolare i risparmi in illuminazione elettrica.
- Visualizzazione delle ombre solari in qualsiasi periodo dell'anno.

- Simulazione termica di edifici ventilati naturalmente.
- Calcolare le potenze di impianti di riscaldamento e condizionamento in tempi brevissimi.
- Aiuto nella comunicazione durante conferenze o presentazioni.

APPENDIX II
CENNI SU MODELLO DI CALCOLO NELLO STANDARD ITALIANO

Il modello di calcolo per la legislazione italiana fa' riferimento alla UNI EN 832, alla prEN 13790 e alla "'CTI Recommendation R30/3"'. CTI é il comitato termo-tecnico italiano, l'organismo chiamato a legiferare normative sull'energia e gli impianti termo-tecnici, mantenendo sempre un riferimento al gruppo CEN. La R03/3 rappresenta una semplificazione nella valutazione dei rendimenti energetici degli edifici, tale da permettere un adeguato intervallo di confidenza. Perció non permette simulazioni dinamiche, mentre semplifica le tipologie di ingresso-uscita dei simulatori. Un esempio di applicazione di questa metodologia é il software "'BestClass"' sviluppato dal politecnico di Milano. Esso stima il consumo di energia per il riscaldamento durante la stagione invernale e il DHW (consumo acqua calda). Purtroppo queste uniche caratteristiche limitano la completezza tecnica del pacchetto, non considerando, per esempio, i consumi di energia elettrica. Per una descrizione dei metodi di calcolo del sofware [1].

ACKNOWLEDGMENT

Questo procedimento può essere criticabile: ma di certo conduce a risultati interessanti
A.N. Kolmogorov
Se chiudete la porta agli errori, anche la verità ne resterà fuori.
R. Tagore
...dobbiamo rimandare il lettore alla letteratura, non senza pero avvertirlo che su questo argomento esiste una mole imponenete di materiale scritto da personaggi dalle dubbie credenziali scientifiche, che spesso si richiamano a fumose motivazioni neuro-biologiche che si rifanno a dei modelli primitivi del neurone introdotti nl 1945 da McCulloch e Pitts che sono stati successivamente dimostrati essere grossolanamente irrealistici dal punto di vista biologico. Ciononostante e invalso in questo settore l'uso di un linguaggio di tipo mistico-biologicoche poco o niente ha a che fare col soggetto e apparentemente serve unicamente a fare audience.
G. Picci

REFERENZE

[1] L. Tronchin, K. Fabbri, *Energy performance building evaluation in Mediterranean countries: Comparison between software simulations and operating rating simulation*, Energy & Buildings, 2007
[2] A. Chiuso *The role of Vector AutoRegressive modeling in Predictor Based Subspace Identification* Automatica, 43(6), June 2007, pp. 1034-1048.
[3] A. Chiuso *On the relation between CCA and predictor-based subspace identification* IEEE Transactions on Automatic Control, 52(10), 2007
[4] G. Picci *dispense corso di identificazione dei modelli* A.A. 2007/2008
[5] S.Bittanti *Identificazione dei modelli e sistemi adattativi* Pitagora Editrice Bologna 2004
[6] Lennart Andersson, Ulf Jönsson, Karl Henrik Johansson, and Johan Bengtsson *A manual for system identification* Gennaio 2006
[7] T.Söderström P.Soica *System Identification* Prentice Hall
[8] Burnham, A. J., J. F. MacGregor e R. Viveros (1999). *Latent Variable Multivariate Regression Modeling*, Chem. Intll. Lab. Syst. **48**, 167-180
[9] Kourti,T.e J.F. MacGregor. (1995)*Process Analysis, Monitoring and Diagnosis, Using Multivariate Projection Methods,* Chem. Intll. Lab. Syst. **28**, 3-21.
[10] Kresta, J. V., J.F. MacGregor e T.E. Marlin (1991), *Multivariate Statistical Monitoring of Process operating Performance,*Can.J. Chem. Engng. **69**, 35-47.
[11] Tonizzo M.(2006-2007), *Metodologie per la selezione delle variabili in ingresso nello sviluppo di sensori software basati su tecniche statistiche multivariate,* Barolo M., Faccotesi P., Tesi di laurea in Ingegneria Chimica anno accademico (2006-2007).
[12] Qin S.Joe,(1993)*Partial Least Squares Regression for Recursive System Identification* Fisher-Rosemount System, Inc. 1712 Centre Creek Drive, Austin
[13] *Identificazione termodinamica di un edificio* Toffoli, Baldan e Albertin A.A. 2006/2007.
[14] S. Wold, Erikson L. Trygg J. Kettaneh N. (2004)*The PLS method - partial least squares projections to latent structures and its application in industrial RDP*
[15] Baibing Li, Julian M., Elaine B.M.(2002) *Model selection for partial least squares regression* Chem. Intll. Lab. Syst. **64**, 79-89.
[16] Wold, S., Annti, F. Lindgren and J. Ohman (1998). *OSC of Near-infrared Spectra*, Chem. Intll. Lab. Syst. **44**, 175-185.
[17] Trygg, J. e S.Wold (2002). *Orthogonalizated Projection to latent Structure* J. Chemom. **16**, 4389-4401.
[18] Chong, I. G. e C.H. Jun(2005). *Performance* of Some Variable selection Methods when Multicollinearity is Present, Chem. Intll. Lab. Syst. **78**, 103-112.
[19] S.J. Qin (1997) *Recursive PLS algorithms for adaptive data modeling*, Computers and Chemical Engineering **22**, 503-514.
[20] R. Muradore, F. Bezzo (2006), *A framework for PLS-SIM integration*, 14^{th} IFAC Symposium on System Identification.
[21] M.Hubert, K. V. Branden (2003), *Robust Methods for Partial Least Squares Regression*.
[22] R. Muradore, F. Bezzo, M. Barolo, (2006) *Optimal sensor location for distributed-sensor systems using multivariate regression*, Computers and Chemical Engineering **30**, 521534