

# Estimatori per l'industria chimica di processo

di Roberto Baratti, Paolo Moretti, Stefania Tronci e Alberto Servida

Nel lavoro sono discusse alcune problematiche relative allo sviluppo di estimatori per la stima di variabili di processo difficilmente acquisibili in linea. In particolare, vengono illustrate due tipologie di estimatori: l'una basata sull'impiego di modelli fenomenologici, l'altra su modelli a reti neurali. Per questo ultimo caso verrà presentata un'applicazione reale, relativa alla stima della qualità del residuo di fondo di una unità vacuum.

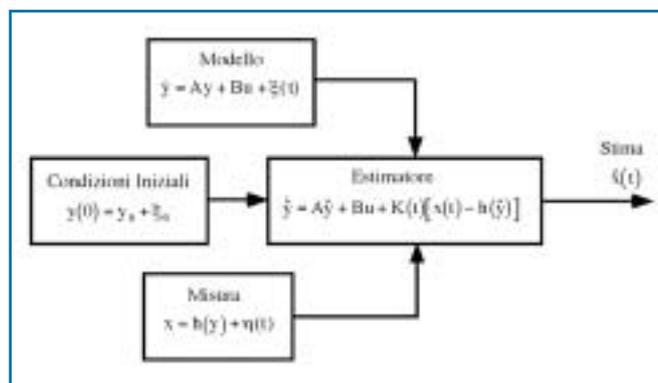


Figura 1 - Tipica architettura di un estimatore lineare

Attualmente, l'avvento dei sistemi di controllo multivariabili *model based* e dei sistemi di ottimizzazione richiede l'acquisizione di un numero maggiore di misure di campo per avere, in ogni istante, l'esatto assetto di funzionamento dell'impianto. Spesso, le variabili di processo principali (per esempio, composizioni, indici di qualità di prodotto, grado di polimerizzazione, concentrazione di biomassa) non sono direttamente acquisibili per mancanza di un adeguato sensore o sono misurabili a intervalli di tempo inaccettabili per le applicazioni in tempo reale. Per ovviare a questo inconveniente possono essere utilizzati sistemi inferenziali (estimatori) per predire i valori delle variabili di processo primarie sulla base di misure di variabili secondarie più facilmente acquisibili, quali per esempio temperature o portate. L'idea è quella di stimare le variabili primarie utilizzando variabili di processo secondarie in congiunzione a un modello del sistema esaminato. In principio, è possibile ricostruire lo stato di un sistema a partire da un insieme incompleto di misure indirette acquisite in tempo reale e da un modello dinamico del processo in esame [1, 2].

Gli estimatori sono classificati sulla base della natura sia dei sistemi esaminati (deterministici o stocastici) sia dei modelli di processo utilizzati (statici o dinamici, strutturati, non strutturati o ibridi) [3]. Gli estimatori sono efficacemente impiegati in una vasta gamma di applicazioni che spaziano dal monitoraggio delle prestazioni al controllo di processo, dalla riconciliazione dei dati allo sviluppo di sistemi di rilevazione di malfunzionamento di sensori/apparecchiature (*sensor/plant fault detection*).

La fase critica dello sviluppo di un estimatore è la formulazione di un adeguato modello di processo semplificato, che oltre a essere sufficientemente semplice da permettere simula-

zioni in tempo reale, deve anche garantire una rappresentazione adeguata del comportamento del sistema esaminato. I modelli di processo possono essere classificati in tre grandi famiglie:

- 1) modelli strutturati (a principi primi o fenomenologici);
- 2) modelli non strutturati (parametrici o "a scatola nera" - *black-box*);
- 3) modelli ibridi che integrano una modellizzazione non strutturata in una descrizione fenomenologica del sistema esaminato.

I primi si basano su una descrizione della fisica (fenomenologia) del processo utilizzando "principi primi", ovvero equazioni di conservazione della materia e dell'energia, equazioni costitutive (per esempio, relazioni di equilibrio), leggi cinetiche (velocità di reazione, di trasporto di materia e di energia). I secondi descrivono semplicemente il legame funzionale tra gli ingressi e le uscite del processo attraverso modelli parametrici che non hanno alcun fondamento fisico. Un tipico esempio della terza famiglia di modelli è costituito da un modello fenomenologico di reattore nel quale la cinetica di reazione è descritta con un modello parametrico.

Generalmente, nello sviluppo di estimatori sono impiegati modelli semplificati (ridotti) che vengono integrati con algoritmi di stima necessari per compensare le inevitabili carenze dei modelli approssimati. Nelle applicazioni dell'industria chimica di processo le tecniche di stima non lineare più comuni sono [4, 5]:

- 1) filtro di Kalman esteso (EKF), di semplice progettazione ma con difficoltà di sintonizzazione e scarsa robustezza;
- 2) osservatori geometrici (Geometric Observer, GO), robusti ma applicati solo in particolari processi;
- 3) osservatori high-gain (HG), stabili ma con procedure di sintonizzazione alquanto elaborate;
- 4) osservatori sliding-mode (SM) robusti ma di elaborata progettazione.

Tutte queste tecniche sono modificabili o adattabili per essere applicate ad un particolare processo chimico, con conseguen-

R. Baratti e S. Tronci, Dipartimento di Ingegneria Chimica e Materiali - Università di Cagliari; Paolo Moretti e Alberto Servida, Dipartimento di D.C.C.I. - Università di Genova. servida@unige.it

ti vantaggi e svantaggi legati al loro sviluppo. L'applicazione di questi osservatori è comunque ristretta a impianti non lineari osservabili ad anello aperto, mentre è stato evidenziato che la loro estensione a sistemi parzialmente osservabili non è immediata [6].

Un approccio alternativo a quelli descritti precedentemente è quello che si basa sull'impiego di sensori software che sfruttano una modellazione a reti neurali. I sensori software rappresentano una famiglia particolare di estimatori non strutturati in cui il processo è descritto tramite un modello a reti neurali.

Nel seguito verranno brevemente discusse le problematiche più importanti e critiche associate allo sviluppo di estimatori strutturati e di sensori software.

### Osservabilità

Prima di poter procedere allo sviluppo di qualunque tipo di estimatore o di sensore software è necessario verificare la possibilità di determinare tutte le variabili di stato,  $y$ , dalla conoscenza delle uscite (output) misurabili del sistema,  $x$ , e delle variabili manipolate,  $u$ . La verifica di ciò è basata sul concetto di osservabilità del sistema, la cui definizione è la seguente: "se, in un intervallo di tempo finito  $t_0 < t < t_f$ , qualunque stato iniziale  $y(t_0)$  può essere individuato dalla conoscenza degli ingressi,  $u(t)$ , e degli output del sistema,  $x(t)$ , allora il sistema è completamente osservabile".

È da notare che l'osservabilità di un sistema è indipendente dai disturbi essendo una proprietà intrinseca del sistema. Per brevità di trattazione ci limiteremo solamente al caso di sistemi lineari del tipo:

$$\begin{aligned} dy/dt &= \mathbf{A}y + \mathbf{B}u \\ \mathbf{x} &= \mathbf{C}y \end{aligned} \quad (1)$$

dove  $y$  è il vettore (di dimensione  $n$ ) degli stati e  $x$  il vettore (di dimensione  $l$ ) delle misure. Si può dimostrare che il sistema è completamente osservabile se il rango della matrice (di dimensione  $n \times nl$ ) di osservabilità  $\mathbf{L}$ :

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

è pari a  $n$ .

Vediamo un esempio per chiarire meglio il concetto. Consideriamo un reattore a mescolamento perfetto (Continuous Stirred Tank Reactor, CSTR) isoterma nel quale avvengono due reazioni in serie irreversibili del primo ordine,  $A \rightarrow B \rightarrow C$ ; le equazioni di bilancio materiale in forma adimensionale assumono la seguente forma:

$$\begin{aligned} dy_1/dt &= -(1+Da_1)y_1 + u_1 \\ dy_2/dt &= Da_1y_1 - (1+Da_2)y_2 + u_2 \end{aligned} \quad (3)$$

Supponiamo che si possa misurare solo la concentrazione della prima specie,  $x_1=y_1$ , la specie reagente di partenza. Per questo sistema le matrici  $\mathbf{A}$  e  $\mathbf{C}$  dell'eq. (1) sono:

$$\mathbf{A} = \begin{bmatrix} -(1+Da_1) & 0 \\ Da_1 & -(1+Da_2) \end{bmatrix} \quad (4)$$

$$\mathbf{C} = [1 \ 0]$$

di conseguenza per l'Eq. (2) la matrice di osservabilità:

$$\mathbf{L} = \begin{bmatrix} 1 & -(1+Da_1) \\ 0 & 0 \end{bmatrix} \quad (5)$$

risulta avere rango 1, e quindi, il sistema non è osservabile. Se ora ipotizziamo di poter misurare solo la seconda specie,  $x_1=y_2$ , così che:

$$\mathbf{C} = [0 \ 1] \quad (6)$$

allora la matrice di osservabilità:

$$\mathbf{L} = \begin{bmatrix} 1 & Da_1 \\ 0 & -(1+Da_2) \end{bmatrix} \quad (7)$$

ha rango 2 e quindi il sistema risulta essere completamente osservabile.

Abbiamo visto come nel primo caso la misura della concentrazione della prima specie,  $x_1=y_1$ , non fornisca sufficienti informazioni per determinare univocamente la concentrazione della seconda specie,  $x_2$ . Al contrario, nel secondo caso, dalla misura della concentrazione della seconda specie,  $x_1=y_2$ , è possibile determinare univocamente la concentrazione della prima,  $y_1$ .

Ciò ha una spiegazione fisica: infatti l'equazione che descrive l'evoluzione temporale di  $y_2$  dipende sia da  $y_1$  sia da  $y_2$ , mentre l'equazione che descrive  $y_1$  è indipendente da  $y_2$ . Quindi, misurando  $y_2(t)$  e conoscendo  $u(t)$  è possibile determinare univocamente  $y_1(t)$ . Mentre, se facessimo l'opposto, cioè misurassimo  $y_1(t)$  conoscendo  $u(t)$  non sarebbe possibile determinare univocamente  $y_2(t)$ .

### Estimatori (sensori) strutturati

In questo paragrafo verranno discusse alcune problematiche relative alla formulazione di estimatori strutturati che si basano sull'impiego di modelli fenomenologici. A seconda che si considerino, implicitamente o meno gli errori di modellizzazione e di misura si parlerà di estimatori (*estimator*) o di osservatori (*observer*) [7]. Questa distinzione non è solo formale come verrà chiarito nel seguito sviluppando entrambi i sistemi di stima per sistemi lineari.

#### Estimatori

Consideriamo il seguente sistema lineare stocastico:

$$\begin{aligned} dy/dt &= \mathbf{A}y + \mathbf{B}u + \xi(t), \quad y(0) = y_0 + x_0 \\ \mathbf{x} &= \mathbf{C}y + \eta(t) \end{aligned} \quad (8)$$

dove con  $\xi(t)$  abbiamo indicato gli errori dovuti alla modellazione e con  $\eta(t)$  quelli dovuti alla misura. Se il sistema è osservabile, allora è possibile derivare un estimatore per inferire tutti gli stati dalla conoscenza delle misure secondarie:  $x$ .

Sebbene ci siano diverse possibilità per la definizione della funzione obiettivo (minimi quadrati, massimo verosimiglianza, *likelihood*, minimo massimo errore ecc.), noi ci limiteremo a considerare come funzione obiettivo quella dei minimi quadrati, che nella loro forma pesata, per il sistema considerato è data da [7]:

$$I = \frac{1}{2} [y(0) - y_0]^T P_0^{-1} [y(0) - y_0] + \frac{1}{2} \int_0^{t_f} \{ (\dot{y} - Ay - Bu)^T Q^{-1} (\dot{y} - Ay - Bu) + [x(t) - C(t)y(t)]^T R^{-1} [x(t) - C(t)y(t)] \} dt \quad (9)$$

dove il primo termine minimizza l'errore quadratico delle condizioni iniziali, il secondo termine minimizza l'integrale dell'errore quadratico del modello e il terzo termine l'integrale dell'errore quadratico della misura. Supponiamo che gli errori  $\xi(t)$  ed  $\eta(t)$  del processo siano di natura gaussiana e completamente incorrelati sia nel tempo (rumore bianco, *white noise*) sia rispetto alle condizioni iniziali, ovvero supponiamo che siano soddisfatte le seguenti condizioni:

$$\begin{aligned} E(\xi(t)) &= 0 & E(\xi(t)\xi^T(t)) &= Q(t)\delta(t-\tau) \\ E(\eta(t)) &= 0 & E(\eta(t)\eta^T(t)) &= R(t)\delta(t-\tau) \\ E(y(0)) &= y_0 & E([y_0 - y(0)][y_0 - y(0)]^T) &= P_0 \\ E(\eta(t)y^T(t)) &= 0, & E(\xi(t)y^T(t)) &= 0, & E(\eta(t)\xi^T(t)) &= 0 \end{aligned} \quad (10)$$

dove E rappresenta l'aspettativa e  $\delta(t)$  rappresenta la funzione di Dirichlet. Applicando il metodo dei minimi quadrati alla funzione obbiettivo precedentemente descritta, si può dimostrare che un possibile filtro ottimo o estimatore è dato da:

$$\begin{aligned} \dot{\hat{y}} &= A\hat{y} + Bu + K(t)[x - C\hat{y}], & \hat{y}(0) &= \hat{y}_0 \\ K(t) &= P(t)C(t)^T R^{-1}(t) \end{aligned} \quad (11)$$

dove la matrice P rappresenta la stima della covarianza dell'errore ricavabile dall'equazione di Riccati:

$$P = PA^T + AP^T + Q - PC^T R^{-1} CP, \quad P(0) = P_0, \quad \dim P = \dim A \quad (12)$$

L'evoluzione temporale della covarianza dell'errore dipende dalle matrici R, che tiene conto degli errori di misura, e Q, che tiene conto degli errori di modellizzazione.

È da evidenziare che la covarianza del modello e delle misure viene utilizzata per la sintonizzazione (calibrazione) dell'osservatore. Detto estimatore è chiamato anche filtro di Kalman, ed è stato proposto da Kalman nel 1960 [8].

#### Osservatori

Come già accennato, gli osservatori sono estimatori per sistemi deterministici, ovvero per sistemi in cui gli errori sia della misura sia del modello sono trascurabili.

Analogamente agli estimatori, gli osservatori vengono utilizzati per la ricostruzione dagli stati di un sistema,  $y(t)$ , dalle misure disponibili,  $x(t)$ .

Consideriamo il sistema lineare deterministico:

$$dy/dt = Ay + Bu, \quad y(0) = y_0 \quad (13)$$

Se il sistema è osservabile, si può sviluppare il seguente osservatore (osservatore di Luenberger) [7]:

$$\frac{d\hat{y}}{dt} = A\hat{y} + Bu + K(x - C\hat{y}) \quad \hat{y}(0) = \hat{y}_0 \quad (14)$$

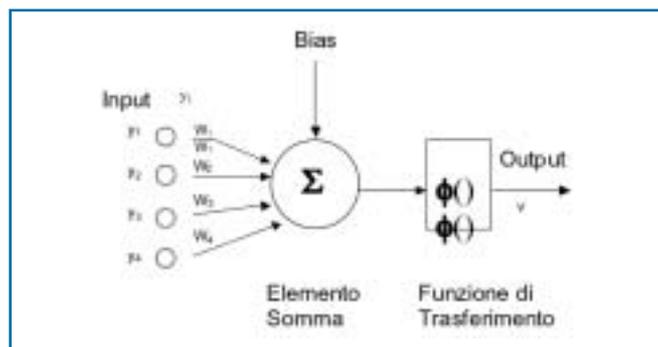


Figura 2 - Schematizzazione del singolo neurone artificiale

La costante di guadagno K dell'osservatore è scelta considerando la dinamica dell'errore di stima:

$$\begin{aligned} \epsilon &= y - \hat{y} \\ \dot{\epsilon} &= \dot{y} - \dot{\hat{y}} \\ \dot{\epsilon} &= Ay + Bu - [A\hat{y} + Bu + K(x - C\hat{y})] \\ \dot{\epsilon} &= (A - KC)\epsilon \end{aligned} \quad (15)$$

L'errore convergerà asintoticamente a zero se la sua dinamica è stabile, ovvero se gli autovalori della matrice (A - KC) hanno parte reale strettamente negativa. Segue che la scelta di K è effettuata in modo che gli autovalori dell'osservatore siano più negativi di quelli del sistema originale, così che la dinamica dell'osservatore sia più veloce del sistema originale.

#### Note

L'estimatore ottimo (filtro di Kalman) richiede la soluzione di  $(n^2+3n)/2$  equazioni differenziali ordinarie (Ordinary Differential Equations, ODE), dove: n è l'ordine del sistema (questo perché la matrice di covarianza è simmetrica:  $P_{ij}=P_{ji}$ ). Quindi per un sistema di ordine 3 si devono risolvere 9 ODE. Mentre, l'osservatore di Luenberger richiede solamente la soluzione di n ODE, in quanto ha una struttura più semplice.

L'esperienza ci dice che gli osservatori risultano essere meno robusti degli estimatori in presenza di errori sia di modellizzazione sia di misura. Inoltre, si deve tenere conto che i due sistemi di stima, precedentemente riportati, si riferiscono a sistemi lineari, cosa che assai raramente si trova nella realtà, infatti i processi chimici sono generalmente non lineari. Basti pensare a un reattore, che per semplicità assumeremo a mescolamento perfetto, dove avvenga una reazione non isoterma e dove la velocità di reazione dipenda dalla temperatura. In questo caso, il modello matematico che descrive il processo è costituito da due equazioni differenziali non lineari.

Per i sistemi non lineari, la struttura dell'estimatore rimane invariata rispetto a quella utilizzata per i sistemi lineari, sebbene il suo sviluppo comporti un maggiore onere matematico/computazionale. La Figura 1 rappresenta schematicamente la tipica architettura di un estimatore lineare.

#### Sensori software

Come già precedentemente accennato, con sensori neurali si indicano, tipicamente, modelli neurali utilizzati per la stima in linea di variabili di processo o di indici di prestazione difficilmente acquisibili in tempo reale [9] e [10]. Il cuore di un sen-

sore software è costituito da un modello neurale che cerca di descrivere, ovviamente in modo approssimato, la relazione funzionale tra le variabili di processo di ingresso (*input*) e quelle di uscita (*output*).

Il modello neurale è costituito da una rete artificiale neurale composta da elementi non lineari semplici, i neuroni, altamente interconnessi in una architettura che consente l'elaborazione dei "segnali" in modo parallelo. Il funzionamento di un singolo neurone artificiale è schematizzato nella Figura 2. L'"elemento somma" calcola la somma pesata degli ingressi (*input*) al neurone e questa informazione viene passata al blocco "funzione di trasferimento" che rielabora, attraverso l'applicazione della funzione di attivazione,  $\phi()$ , la somma calcolata trasformandola nell'uscita del neurone:

$$v = \phi(s) ; s = \sum_{j=1}^{p(=4)} w_j y_j \quad (16)$$

$w_j$  rappresentano i pesi delle connessioni.

Tra le funzioni di trasferimento più comuni possiamo menzionare [11]:

la funzione soglia:

$$\phi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases}$$

la funzione lineare:

$$\phi(v) = \begin{cases} 1 & v \geq 1/2 \\ v & -1/2 \geq v \leq 1/2 \\ 0 & v \leq -1/2 \end{cases}$$

la funzione sigmoide:

$$\phi(v) = \frac{1}{1 + \exp(-v)}$$

la funzione tangente iperbolica:

$$\phi(v) = \text{Th}(v) = \frac{1 - \exp(-v)}{1 + \exp(-v)}$$

In una rete neurale multistrato *feedforward* (RNMFF) i neuroni sono interconnessi in modo tale che l'uscita di un neurone viene trasmessa, e quindi rielaborata, da tutti i neuroni dello strato successivo come mostrato dalla Figura 3, che schematizza un rete RNMFF a tre strati con  $n$  ingressi (*input*), 2 neuroni nello strato nascosto (*hidden layer*) e 2 uscite. I pesi delle connessioni rappresentano i parametri del modello neurale e, nelle tipiche applicazioni di interesse dell'industria chimica di processo, vengono calcolati attraverso una fase di calibrazione (addestramento, *training*, e *cross-validation*) supervisionata. Ciò coincide con un problema di stima (*fitting*) dei parametri per confronto con dati sperimentali (le misure disponibili), problema che può essere risolto adottando diversi algoritmi di cui quello di *back-propagation* rappresenta sicuramente il più noto e utilizzato [11].

Le reti RNMFF hanno una struttura intrinsecamente statica

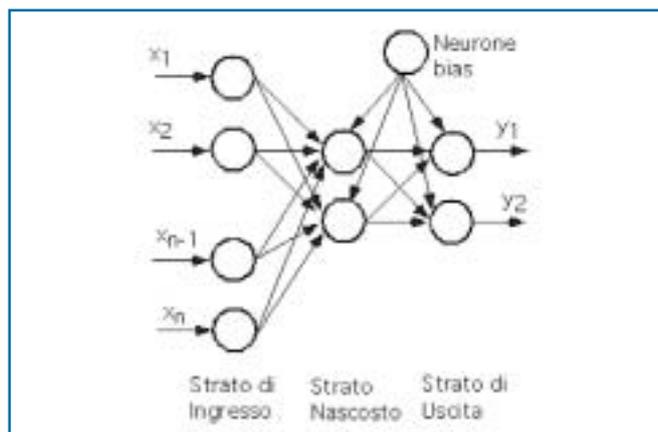


Figura 3 - Schematizzazione di una rete neurale multistrato feedforward a tre strati di architettura ( $n, 2, 2$ )

nel senso che sono dispositivi formulati per descrivere legami funzionali non lineari tra gli ingressi e le uscite. In realtà questa tipologia di rete neurale è utilizzata con successo anche nello sviluppo di sensori software in quanto le informazioni sulla dinamica del processo sono associate alle misure secondarie che fanno parte dell'insieme di ingressi al modello neurale.

Lo sviluppo di un sensore software è un processo articolato in più fasi: acquisizione dei dati di campo, pre-trattamento (*pre-processing*) dei dati acquisiti, selezione delle variabili di processo da utilizzare come ingressi alla rete neurale, calibrazione del modello neurale e convalida fuori linea (*off line*). A nostro avviso, le prime tre fasi sono particolarmente critiche per poter formulare un modello neurale efficiente, e di conseguenza un sensore software funzionale. È importante sottolineare che la formulazione ottimale di un sensore software è il risultato di un compromesso tra esigenze interpolanti ed esigenze di generalizzazione (predizione). Le prime si riferiscono alla capacità del modello neurale di descrivere i dati utilizzati nella fase di calibrazione, mentre le seconde fanno riferimento alla capacità del modello di rappresentare dati sperimentali mai utilizzati nella fase di calibrazione.

La raccolta dei dati di campo richiede l'acquisizione sia di misure di linea sia di misure fuori linea (le misure di laboratorio) e non deve assolutamente corrispondere a una mera raccolta di dati sperimentali: è necessario che in questa fase lo sviluppatore del sensore software analizzi con senso critico la qualità e il grado di rappresentatività dei dati sperimentali. È importante ricordare che la formulazione di modelli neurali è realizzata sulla base di dati storici di funzionamento (*data-driven models*) di conseguenza è importante che i dati utilizzati siano rappresentativi delle condizioni operative alle quali il sensore software dovrà operare.

Il pre-trattamento dei dati consente di identificare dati rappresentativi di condizioni di malfunzionamento degli strumenti analitici o delle apparecchiature, e di funzionamento fuori specifica. È in questa fase che le variabili di processo sono riscaldate: le variabili utilizzate come ingressi alla rete sono tipicamente riscaldate nell'intervallo (-1, 1) mentre le variabili che costituiscono le uscite della rete sono riscaldate nell'intervallo (0, 1).

La selezione delle variabili da impiegare come ingressi al sensore software neurale richiede una maggiore attenzione rispetto alla scelta delle variabili di uscita, che, in un certo senso, è una diretta conseguenza della specifica utilizzazione

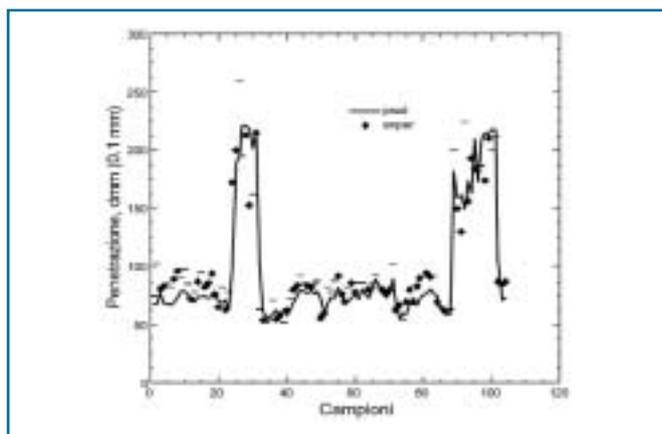


Figura 4 - Confronto tra le previsioni del sensore software e le misure sperimentali

del sensore software: infatti, le uscite del sensore corrispondono alle variabili di processo (indici di prestazione o di qualità di prodotto) che desideriamo stimare.

Già nella fase di pre-trattamento dei dati è possibile rendersi conto di quali variabili di processo influenzino in modo più significativo le uscite del sensore software. È essenziale selezionare, tra tutte le potenziali variabili, quelle che forniscono informazioni sufficienti per determinare variazioni di stato di funzionamento del processo. Per questa ragione è da evitare qualsiasi ridondanza e spesso le previsioni del sensore software possono essere sensibilmente migliorate se più variabili vengono raggruppate in un'unica ingresso alla rete neurale. Un esempio può chiarificare meglio quest'ultimo concetto. Nel caso delle colonne di distillazione, la qualità (purezza) del distillato dipende in modo significativo dal rapporto di riflusso, di conseguenza, se si volesse sviluppare un sensore software per stimare la qualità del distillato sarebbe meglio utilizzare come ingresso alla rete neurale il rapporto di riflusso piuttosto che "obbligare" la rete a determinare il legame funzionale tra composizione del distillato e le portate di distillato e di riflusso. Tutto ciò va a sfatare l'idea sbagliata di poter utilizzare la modellizzazione a reti neurali senza alcuna conoscenza del sistema che si desidera descrivere. Una comprensione, anche semplicemente qualitativa, dei fenomeni chimico fisici che regolano il sistema esaminato è di sicuro aiuto allo sviluppo di modelli neurale efficienti (*well designed*) con buone caratteristiche di generalizzazione: infatti, una buona conoscenza del processo consente una selezione ottimale delle variabili di processo da utilizzare come ingressi al modello neurale.

Per quanto riguarda la calibrazione del modello neurale è necessario prevenire il fenomeno di *overtraining*, che corrisponde a una eccessiva calibrazione della rete neurale sui dati di addestramento. Per evitare questo si adotta il metodo *cross validation*, che consiste nell'impiegare 2/3 dei dati di calibrazione per stimare i parametri (pesi) della rete e 1/3 dei dati per controllare l'errore di predizione della rete su dati non utilizzati direttamente per la stima dei parametri.

### Applicazione

In questa sezione verrà brevemente descritta l'applicazione di un sensore software (neurale) per la stima della penetrazione del residuo di una colonna di distillazione sottovuoto nell'as-

setto di lavorazione asfalto. In questa modalità operativa il residuo, utilizzato come base per la produzione di bitume, è caratterizzato in termini di penetrazione, misura fatta in laboratorio ad intervalli di 3-6 ore. La disponibilità in linea della penetrazione del residuo è critica per una migliore conduzione dell'unità, perché consentirebbe di intervenire prontamente nel correggere variazioni della qualità del prodotto.

I risultati mostrati nella Figura 4 si riferiscono a un sensore software sviluppato utilizzando dati storici (604 misure di penetrazione) relativi a un funzionamento di ca. 7 mesi (fine maggio 2001 - inizio gennaio 2002). Il modello neurale impiegato, di struttura (5,3,1), è stato calibrato utilizzando 500 misure di laboratorio (400 utilizzate per l'addestramento e 100 per la *cross validation*); la prova di predizione si riferisce alle 104 misure non utilizzate per la calibrazione del modello. In realtà, il sensore software aveva una struttura ricorrente esterna [11] in quanto uno dei 5 ingressi alla rete neurale era costituito dall'ultima misura sperimentale disponibile di penetrazione. Utilizzare valori storici come ingressi a modelli neurali è una tecnica per impartire caratteristiche dinamiche alle reti neurali e, talvolta, questo ne migliora le prestazioni. Il sensore software della Figura 4 è soddisfacente in quanto ha fornito un errore medio percentuale (13,0%) paragonabile a quello riportato dalla Norma Europea Standard (Eni 426), che indica per le misure di penetrazione errori di riproducibilità variabili dal 6 al 13% a seconda del valore di penetrazione misurato.

### Conclusioni

Nel lavoro abbiamo illustrato due "metodologie software" per lo sviluppo di sistemi inferenziali (estimatori) in grado di stimare variabili di processo, indici di prestazione o indici di qualità difficilmente acquisibili in linea. La prima famiglia di estimatori/osservatori, strutturati, è particolarmente adeguata per risolvere problemi di stima nel caso in cui si abbia un numero limitato di dati sperimentali storici, situazione per la quale è necessario descrivere il processo utilizzando un modello fenomenologico. La seconda famiglia di estimatori, sensori software neurali, sfrutta la modellizzazione a reti neurali (modellizzazione non strutturata), ed è particolarmente vantaggiosa quando i dati storici di funzionamento disponibili sono sufficienti per una buona calibrazione del modello neurale.

### Bibliografia

- [1] A. Gelb, A., Applied Optimal Estimation, MIT, Cambridge, 1974.
- [2] A.H. Jazwinski, Stochastic Processes and Filtering Theory, Academic Press, New York, 1970.
- [3] M. Soroush, *Comp. Chem. Engng.*, 1998, **23**, 229.
- [4] F.J. Doyle, *J. Proc. Cont.*, 1998, **8**, 339.
- [5] G.B. Wang *et al.*, *Chem. Eng. Sci.*, 1997, **52**, 787.
- [6] J. Alvarez, T. Lopez, *AIChE J.*, 1999, **45**, 107.
- [7] H.W. Ray, Advanced Process Control, McGraw Hill, New York, 1981.
- [8] R.E. Kalman, *Trans. ASME J. Basic Eng.*, 1960, **82**, 35.
- [9] D. Harold, *Control Eng. Europe*, 2001, June, 42.
- [10] G. Martin, *Chem. Eng. Prog.*, 1997, July, 66.
- [11] J.C. Principe *et al.*, Neural and Adaptive Systems: Fundamental through Simulations, John Wiley & Sons, New York, 2000.