

Consiglio Nazionale delle Ricerche

**ISTITUTO DI ELABORAZIONE
DELLA INFORMAZIONE**

PISA

**STIMA DI MASSIMA VEROSIMIGLIANZA DEI PARAMETRI
IN SISTEMI LINEARI CON INGRESSO NOTO.**

L. Bedini, F. Caroti Ghelli

Nota Interna

B79-17

Giugno

STIMA DI MASSIMA VEROSIMIGLIANZA DEI PARAMETRI IN SISTEMI LINEARI CON INGRESSO NOTO.

L. Bedini, F. Caroti Ghelli

INTRODUZIONE

Nello studio di sistemi complessi (biologici, ecologici, ecc...) si ricorre spesso all'uso di modelli che approssimano il comportamento del sistema a seguito di un qualche stimolo noto; in letteratura sono stati trattati diversi tipi di modelli lineari e proposti vari algoritmi di stima [1]. In generale la scelta del tipo di modello, nonché dell'algoritmo di stima dipende dal particolare problema sotto studio.

Nel presente lavoro si è fatto riferimento al problema di caratterizzare quantitativamente il comportamento di alcuni sistemi biologici, anche al fine di un eventuale ausilio diagnostico. In tale caso si ha la necessità di disporre di modelli funzionali, sufficientemente semplici per consentire una rapida stima dei parametri. Si è pertanto preso in considerazione la classe dei modelli lineari, discreti e a dimensione finita.

L'algoritmo di stima è basato sul metodo di massima verosimiglianza e fornisce pertanto stime asintoticamente corrette e consistenti. Tale algoritmo rappresenta una modifica di un algoritmo già riportato in letteratura [2].

Al fine di una maggiore semplicità di esposizione e di programmazione si è fatto riferimento a modelli con un

solo ingresso ed una sola uscita; il metodo è comunque facilmente estendibile al caso di più ingressi e/o più uscite.

Forme equivalenti di modelli

Si farà riferimento a modelli lineari, regolari, e a dimensione finita n , con un solo ingresso ed una sola uscita.

Tali modelli sono generalmente riportati sia in forma di una sola equazione differenziale di ordine n del tipo:

$$y^{(n)} + a_1 y^{(n-1)} + \dots + a_n y = c_1 u^{(n-1)} + c_2 u^{(n-2)} + \dots + c_n u \quad (1)$$

ove $a_1, a_2, \dots, a_n, c_1, c_2, \dots, c_n$ sono numeri reali, u l'ingresso ed y l'uscita; sia in forma di un sistema di n equazioni differenziali del primo ordine [1]:

$$\dot{\underline{x}} = F \underline{x} + \underline{g} u \quad (2)$$

$$y = \underline{h}^T \underline{x}$$

ove $\underline{x} = [x_1, x_2, \dots, x_n]^T$ rappresenta il vettore delle variabili di stato, F è una matrice $n \times n$, in generale ad elementi complessi, \underline{g} ed \underline{h} sono dei vettori.

Sostituendo nelle (2) al posto del vettore \underline{x} un vettore $\underline{z} = T \underline{x}$ (T matrice $n \times n$ non singolare) si hanno forme equivalenti del modello con matrici

$$F^* = T F T^{-1} \quad \underline{g}^* = T \underline{g} \quad (\underline{h}^*)^T = \underline{h}^T T^{-1}$$

Se il modello (2) è completamente raggiungibile [3], allora è possibile eseguire una trasformazione $\underline{z} = T \underline{x}$, in modo tale che la terna $F^*, \underline{g}^*, \underline{h}^*$ risulti nella forma (forma canonica di controllo):

$$F^* = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \\ -a_n & \dots & \dots & \dots & -a_1 \end{bmatrix} \quad \underline{g}^* = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad \underline{h}^* = \begin{bmatrix} c_n \\ \vdots \\ c_{n-1} \\ \vdots \\ c_1 \end{bmatrix} \quad (3)$$

la (3) esplicita la relazione esistente tra la rappresentazione di uno stesso modello nella forma (1) e (2).

L'utilizzazione di mezzi di calcolo digitali comporta la discretizzazione del modello continuo: come noto [4], detto T il periodo di campionamento prescelto, la discretizzazione del modello rappresentato in (2) risulta in:

$$\begin{aligned} \underline{x}(kT+T) &= A \underline{x}(kT) + \underline{b}u(kT) \\ \underline{y}(kT) &= \underline{h}^T \underline{x}(kT) \end{aligned} \quad (4)$$

ove $A = e^{FT}$ $\underline{b} = \underline{g} e^{FT}$.

In (4), con una opportuna scelta della scala dei tempi, si può sempre ritenere $T=1$.

Analogamente a quanto detto per il modello (2), se il modello discreto (4) è completamente raggiungibile, la terna A , \underline{b} , \underline{h} , può essere posta nella forma

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \\ -\bar{a}_n & \dots & \dots & \dots & -\bar{a}_1 \end{bmatrix} \quad \underline{b} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad \underline{h} = \begin{bmatrix} \bar{c}_n \\ \bar{c}_{n-1} \\ \vdots \\ \bar{c}_1 \end{bmatrix} \quad (5)$$

Un modello discreto del tipo (4), con $T=1$, può essere rappresentato con la seguente equazione alle differenze finite di ordine n :

$$y(k+n) + \bar{a}_1 y(k+n-1) + \dots + \bar{a}_n y(k) = \bar{c}_1 u(k+n-1) + \dots + \bar{c}_n u(k) \quad (6)$$

ove $\bar{a}_1, \bar{a}_2, \dots, \bar{a}_n, \bar{c}_1, \bar{c}_2, \dots, \bar{c}_n$ sono le quantità presenti in (5).

Le relazioni intercorrenti tra a_i ed \bar{a}_i , c_i e \bar{c}_i ($i=1, 2, \dots, n$) possono essere dedotte dalla (4) e dalla trasformazione eseguita per rappresentare la (4) nella forma (5).

La stima di massima verosimiglianza

Il problema della stima dei parametri viene posto nel modo seguente:

dato un modello discreto del tipo (4), con A , \underline{b} , \underline{h} nella forma (5), o (6) con $\underline{a} = [\bar{a}_1, \bar{a}_2 \dots \bar{a}_n]^T$, \underline{h} e lo stato iniziale $\underline{x}(1)$ non noti;

dato un modello delle osservazioni delle uscite del tipo $z(k) = y(k) + e(k)$, con $e(k)$ rumore gaussiano a media nulla, non correlato, con varianza σ^2 indipendente da k ;

dato un vettore $z(1), z(2) \dots z(M)$ di osservazioni delle uscite ed un vettore $u(1), u(2) \dots u(M)$ di osservazioni dell'ingresso;

stimare i parametri del modello \underline{a} ed \underline{h} e lo stato iniziale $\underline{x}(1)$ utilizzando il metodo di massima verosimiglianza.

Si tratta cioè, posto $\underline{z}_k = [z(1), z(2) \dots, z(k)]^T$, di massimizzare la funzione di verosimiglianza $L[\underline{z}_M / \underline{a}, \underline{h}, \underline{x}(1)]$. Come noto, essendo il rumore $e(k)$ gaussiano, non correlato, risulta

$$L[\underline{z}_M / \underline{a}, \underline{h}, \underline{x}(1)] = \prod_{k=1}^M P[\underline{z}_k | \underline{a}, \underline{h}, \underline{x}(1)] \exp\left(-\frac{1}{2} \sum_{k=1}^M [z(k) - \underline{h} \underline{x}(k)]^2\right) \quad (7)$$

E' quindi necessario massimizzare la (7), o, in forma equivalente, minimizzare la quantità

$$J = \sum_{k=1}^M [z(k) - \underline{h} \underline{x}(k)]^2 \quad (8)$$

subordinata alle condizioni date dalla (4) o, equivalentemente dalla (6).

Nel seguito, per brevità di scrittura, indicheremo più sinteticamente $\underline{x}(k)$ con \underline{x}_k ; $u(k)$ con u_k , $z(k)$ con z_k .

Il problema posto può essere risolto utilizzando i moltiplicatori di Lagrange: come noto [2], il minimo di J vincolato può essere determinato calcolando il minimo non

vincolato rispetto alle variabili $\underline{\lambda}_k$, \underline{x}_k ($k=1, 2, \dots, M$), \underline{a} ed \underline{h} della funzione

$$J^* = J + \sum_{k=1}^M \underline{\lambda}_{k+1}^T (\underline{x}_{k+1} - A \underline{x}_k - \underline{b} u_k) \quad (9)$$

Le condizioni necessarie per il minimo di J^* si ottengono eguagliando a zero le seguenti quantità:

$$\frac{\partial J^*}{\partial \underline{a}} = \frac{\partial J}{\partial \underline{a}} + \sum_{k=1}^M \underline{\lambda}_{k+1}^T \cdot (\underline{x}_{k+1} - A \underline{x}_k - \underline{b} u_k) \quad (10)$$

$$\frac{\partial J^*}{\partial \underline{h}} = \frac{\partial J}{\partial \underline{h}} = - \sum_{k=1}^M 2 \cdot [z_k - \underline{h}^T \cdot \underline{x}_k] \cdot \underline{x}_k \quad (11)$$

$$\frac{\partial J^*}{\partial \underline{x}_k} = - 2 \underline{h}^T [z_k - \underline{h}^T \underline{x}_k] + \underline{\lambda}_k - A^T \underline{\lambda}_{k+1} \quad (k=1, 2, \dots, M) \quad (12)$$

$$\frac{\partial J^*}{\partial \underline{\lambda}_k} = \underline{x}_{k+1} - A \underline{x}_k - \underline{b} u_k \quad (k=1, 2, \dots, M-1) \quad (13)$$

ove $\underline{\lambda}_{M+1} = 0$.

Si può quindi dimostrare che nei punti in cui sono soddisfatte la (12) e la (13) valgono le seguenti uguaglianze:

$$\left(\frac{\partial J}{\partial \underline{a}} \right)_V \Big|_{P=P_0} = \frac{\partial J^*}{\partial \underline{a}} \Big|_{P=P_0}; \left(\frac{\partial J}{\partial \underline{x}(1)} \right)_V \Big|_{P=P_0} = \frac{\partial J^*}{\partial \underline{x}(1)} \Big|_{P=P_0} \quad (14)$$

$$\left(\frac{\partial J}{\partial \underline{h}} \right)_V \Big|_{P=P_0} = \frac{\partial J^*}{\partial \underline{h}} \Big|_{P=P_0}$$

qualunque sia $P_0 = (\underline{a}_0, \underline{x}_0(1), \underline{h}_0)$, ove con $\left(\frac{\partial J}{\partial} \right)_V$ si intendono le derivate di J calcolate lungo l'equazione del vincolo espresso dalla (13).

E' facile verificare che le precedenti condizioni necessarie per il minimo di J^* costituiscono un sistema non lineare di $2 \cdot (M+1) \cdot n$ equazioni in altrettante incognite; la soluzione di tale sistema dovrebbe essere calcolata con metodi

iterativi che, anche per valori di M ed n relativamente piccoli, risulterebbero proibitivamente lunghi e di incerta convergenza se applicati globalmente a tutte le variabili.

Uguagliando a zero separatamente la (13) con $\underline{x}(1)$ ed \underline{a} assegnati, e la (12) con $\underline{\lambda}_{M+1}=0$, la soluzione può essere determinata limitando l'uso di metodi iterativi alla soluzione di un sistema con $3n$ variabili; tali $3 \cdot n$ variabili sono le componenti dei vettori \underline{a} , \underline{h} ed $\underline{x}(1)$ che possono essere stimati, ad esempio, utilizzando il metodo del gradiente, tenendo conto delle (14) per quanto riguarda il calcolo delle derivate; uguagliando a zero anche la (11), assegnati \underline{z}_k e \underline{x}_k ($k=1,2,\dots,M$) si ha, con semplici trasformazioni:

$$\underline{h} = \left[\sum_{k=1}^M \underline{x}_k \cdot \underline{x}_k^T \right]^{-1} \cdot \sum_{k=1}^M \underline{z}_k \cdot \underline{x}_k \quad (15)$$

Perciò l'uso di metodi iterativi può essere limitato alla soluzione di un sistema in $2 \cdot n$ variabili, costituenti i vettori \underline{a} ed $\underline{x}(1)$.

Scelto un metodo iterativo per la minimizzazione di J^* rispetto alle variabili \underline{a} ed $\underline{x}(1)$, basato sul calcolo delle derivate $\partial J^* / \partial \underline{a}$ e $\partial J^* / \partial \underline{x}(1)$, e precisati i "criteri di uscita" dal procedimento iterativo, l'algoritmo si configura secondo i passi seguenti:

- 1) si assegna un punto arbitrario $P_0 \equiv (\underline{a}^0, \underline{x}^0(1))$;
- 2) uguagliando a zero la (13) si ricavano $\underline{x}_2^0, \underline{x}_3^0 \dots \underline{x}_M^0$;
- 3) uguagliando a zero la (12) per $k=M, M-1 \dots 2$, posto $\underline{\lambda}_{M+1}^0=0$, si ricavano $\underline{\lambda}_M^0, \underline{\lambda}_{M-1}^0 \dots \underline{\lambda}_2^0$;
- 4) si calcola \underline{h}_0 utilizzando la (15);
- 5) si calcolano le derivate $\left(\frac{\partial J}{\partial \underline{a}} \right) \Big|_{P=P_0}$ e $\left(\frac{\partial J}{\partial \underline{x}(1)} \right) \Big|_{P=P_0}$ tenendo conto della (14); in particolare si noti che $\left. \frac{\partial J^*}{\partial \underline{x}(1)} \right|_{P=P_0}$ si ottiene utilizzando la (12) per $k=1$;

- 6) utilizzando le derivate in P_0 calcolate al punto 4), si individua un nuovo punto $P=P_0 + \Delta P$, ove ΔP è calcolato sulla base del metodo di minimizzazione adottato;
- 7) Si verifica se sono soddisfatti i "criteri di uscita"; altrimenti si pone $P_0=P$ e si ricomincia da 2).

La scelta di \underline{a}_0 in 1) può risultare critica per la convergenza del procedimento iterativo; si sono ottenuti buoni risultati assumendo come \underline{a}_0 iniziale il vettore che si ottiene minimizzando col metodo dei minimi quadrati la quantità:

$$\sum_{k=1}^{M-n} (z_{k+n} + a_1 \cdot z_{k+n-1} + \dots + a_n \cdot z_k - b_1 \cdot u_{k+n-1} + \dots - b_n \cdot u_k)^2 \quad (1.6)$$

rispetto alle incognite $\underline{a}=(a_1 a_2 \dots a_n)$ e $(b_1 b_2 \dots b_n)$.

La scelta di \underline{x}_0 non sembra essere altrettanto critica; nelle prove eseguite nel presente lavoro si è assunto $\underline{x}_0=0$.

Descrizione del sottoprogramma IDEN per la stima dei parametri.

Il sottoprogramma IDEN esegue la stima dei parametri di un modello lineare con un ingresso ed una uscita, con rumore gaussiano sovrapposto all'uscita, secondo il procedimento illustrato nel paragrafo precedente.

La stima viene eseguita, noto l'ordine N del modello, assegnati M valori consecutivi $U(J)$ ($J=1:M$) dell'ingresso e M valori consecutivi $Z(J)$ ($J=1:M$) dell'uscita misurata.

Il modello assunto è il seguente:

$$\underline{X}(k+1) = \underline{F}_A \underline{X}(k) + \underline{B}U(k)$$

$$Y(k) = \underline{C}^T \underline{X}(k) \quad Z(k) = Y(k) + E(k)$$

con $E(k)$ ($k=1:M$) rumore gaussiano a media nulla e varianza costante; viene fatta l'ipotesi che il modello sia rappresentato in forma canonica di controllo, pertanto i parametri da stimare sono gli elementi $A(j)$, ($j=1:N$), dell'ultima riga della matrice \underline{F}_A ; gli elementi $C(J)$, ($J=1:N$) del vettore \underline{C} ed il vettore degli stati iniziali. Per facilità di program-

mazione il vettore degli stati è memorizzato in $A(J+N)$, ($J=1 \div N$); il vettore C , il vettore Z delle osservazioni $Z(J)$ ed il vettore U degli ingressi $U(J)$ sono posti in area comune.

Il sottoprogramma IDEN è stato scritto in linguaggio FORTRAN IV ed è stato implementato sul sistema IBM370/168 con diverse variabili dichiarate in doppia precisione ; utilizza le seguenti subroutines:

DGELG(package SSP-IBM) per la soluzione di un sistema di equazioni lineari)

DFMCG(package SSP-IBM) per la minimizzazione eseguita col metodo del gradiente coniugato

FUNCT e FUNZ per il calcolo del vettore C , assegnato A , col metodo dei minimi quadrati, dell'errore ER e delle derivate $DA(J)$ ($J=1 \div 2N$)

DMFGR(package SSP-IBM) per calcolare il rango una matrice.

Lo schema a blocchi del sotto-programma è riportato in fig. 1; come già detto, la stima viene eseguita con un metodo iterativo che presuppone la scelta dei valori iniziali per il vettore $\underline{A} = A(1), A(2) \dots A(N)$ e per il vettore degli stati $A(N+1), A(N+2) \dots A(2N)$; mentre il vettore degli stati viene scelto arbitrariamente uguale a zero, per la scelta di \underline{A} vengono eseguiti dal sottoprogramma alcuni tentativi legati alla minimizzazione della (16); se tali tentativi non danno esito viene richiesto un punto iniziale all'operatore. Il parametro HESP è richiesto dalla subroutine DFMCG e viene utilizzato nel criterio per terminare le iterazioni eseguite nel metodo del gradiente coniugato.

Il sottoprogramma prevede i seguenti parametri di ingresso

N : dimensione del modello;

Z : vettore delle osservazioni in area comune;

U: vettore degli ingressi in area comune;
M: numero delle osservazioni in area comune.

Ha come parametri di uscita:

A(J), (J=1:N): parametri stimati del modello
A(J+N), (J=1:N): valori dello stato stimato
DA(J), (J=1:N): valori stimati delle derivate dell'errore
rispetto ai parametri A(J), (J=1:2N)
ER: errore definito come scarto quadratico medio
C(J), (J=1:N) valori del vettore C stimato in area comune.

Prevede inoltre le seguenti stampe:

IER IN DGELG = 1: avviso di tentativo non riuscito per
calcolare un set iniziale di parametri A(J), (J=1:N);
in tal caso il programma esegue altri tentativi ri-
ducendo il numero iniziale delle osservazioni; se
neppure tali tentativi hanno esito viene richiesto
un set iniziale all'operatore;

A ????: richieste di parametri iniziali A(J) (J=1:N)

HESP? richiesta del parametro HESP per la subroutine DFMCG
connesso con la precisione desiderata nel calcolo del
minimo; viene consigliato un valore compreso tra 10^{-4}
e 10^{-8} ;

IER =: stampa del valore assunto da IER nella subroutine
DFMCG;

COEFFICIENTI A: stampa dei parametri A(J), (J=1:N) stimati;

COEFFICIENTI C: stampa dei coefficienti C(J), (J=1:N) stimati;

STATO INIZIALE: stampa dello stato iniziale stimato;

ERRORE: stampa dell'errore stimato ER;

DERIVATE: stampa delle 2N derivate stimate;

1 CAMBIO 2 PROSEGUO 3 RETURN: richiesta di un numero (ICON) che determina le seguenti modalità di esecuzione: se tale numero è 1, il programma tenta un'altra stima con un set di parametri iniziali diversi (che verranno richiesti in seguito), se è 2 affina la stima effettuata eseguendo altre iterazioni, se è 3 termina e rientra.

IER IN FUNZ = - 1: avviso di matrice singolare nella stima dei parametri; in tale caso il programma riduce la dimensione della matrice singolare eliminando le righe e le colonne linearmente dipendenti e assegnando il valore zero ai C(k) corrispondenti;

PROBLEMA MAL POSTO: avviso della impossibilità di stimare nella subroutine FUNZ il vettore C con metodo dei minimi quadrati anche riducendo la dimensione della matrice singolare; in tal caso il programma pone ER = - 1 e ritorna.

TEMPO DI CALCOLO: stampa il tempo di calcolo impiegato per la stima, espresso in centesimi di secondo.

Risultati

Al fine di valutare le prestazioni del sottoprogramma IDEN, sono state eseguite alcune prove su modelli simulati assumendo come dati z_k ($k=1:M$) le quantità $z_k = y_k + e_k$ ove y_k è l'uscita deterministica del modello simulato con ingresso noto u_k ed e_k è un rumore gaussiano con media nulla e varianza σ_e^2 prefissata, generato attraverso la subroutine GAUSS del package SSP-IBM.

A titolo di esempio in tabella I sono riportati i risultati ottenuti per un modello con $N=3$, $M=60$,

$\underline{a} = [-2.4, 1.91, -0.504]^T$, $\underline{h} = [3.0, 1.0, 0.0]^T$, $x(1) = [0.0, 0.0, 0.0]^T$,
 per la sequenza di ingresso $\{1, 1, 0, 0, \dots, 0\}$

La percentuale di rumore è data dal rapporto tra la quantità σ_e e il valore medio del modulo dell'uscita del modello; i tempi T_C di calcolo sono espressi in centesimi di secondo.

L'errore E_{MP} costituisce lo scarto quadratico medio tra le osservazioni z_k generate e l'uscita del modello simulato; l'errore E è lo scarto quadratico medio tra le osservazioni e l'uscita del modello identificato. Per il caso esaminato e riportato in tab. I l'errore E è risultato minore dell'errore E_{MP} ; ciò significa che l'uscita del modello identificato costituisce un fitting, migliore, rispetto all'uscita del modello simulato, delle osservazioni z_k , come può vedersi dalla fig. 2.

TABELLA I.

rumore	\underline{a}	\underline{h}	$\underline{x}(1)$	E_{MP}	E	T_C
1%	-2.40 1.91 -.506	2.94 .69 -.001	-.006 -.09 .04	.214	.209	50
10%	-2.51 -2.12 -.60	3.39 -2.32 1.03	-.50 -.64 -.52	2.22	2.02	199

Bibliografia

- [1] Eykhoff P.: System Identification John Wiley & Sons, London 74.
- [2] Sage A.P., Melsa J.L.: System identification Academic Press, New York 1971.
- [3] Rubio J.E.: The theory of linear systems Academic Press, New York 1971.
- [4] Freeman H.: Discrete-time systems John Wiley & Sons, Inc. New York 65.

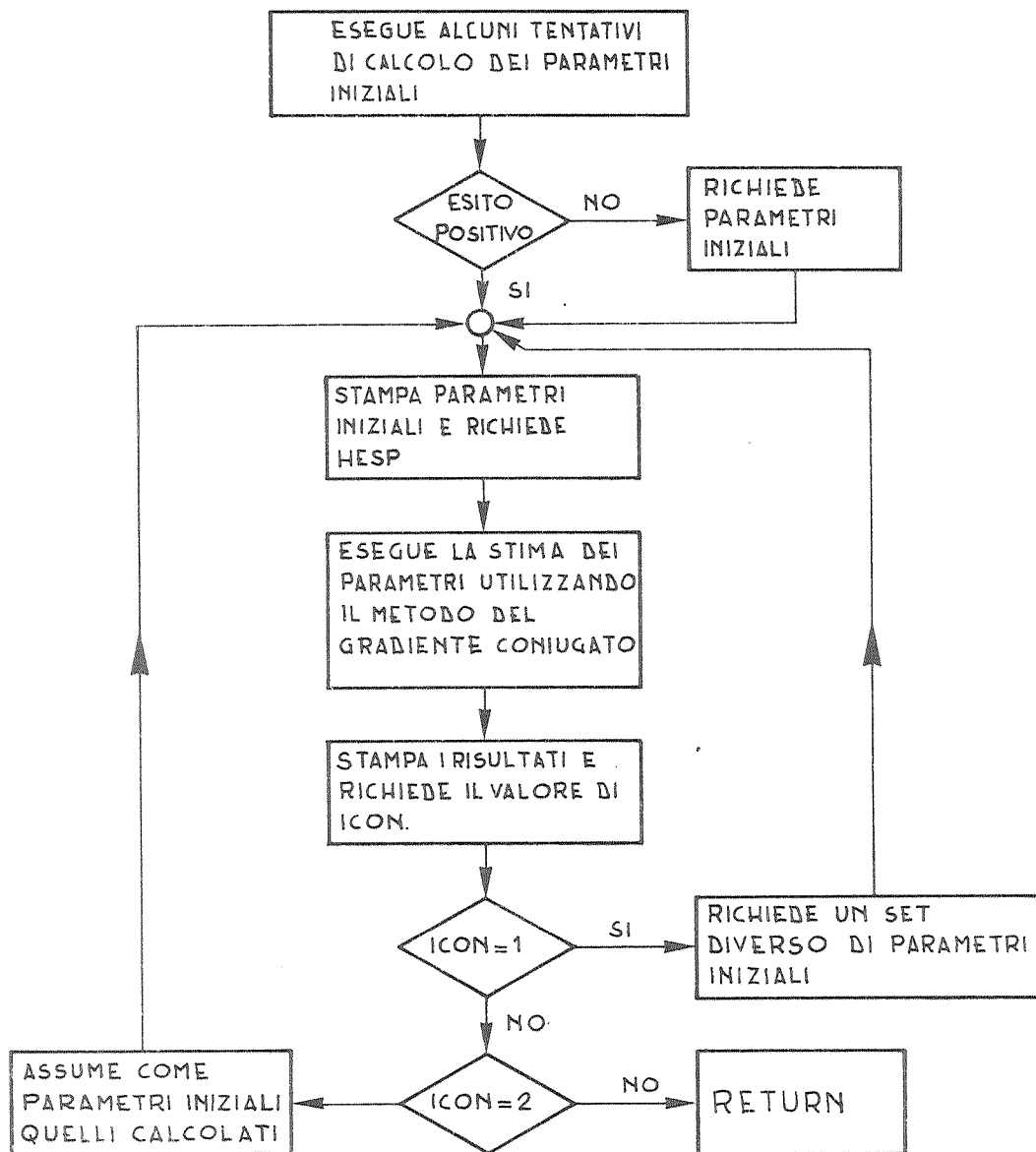
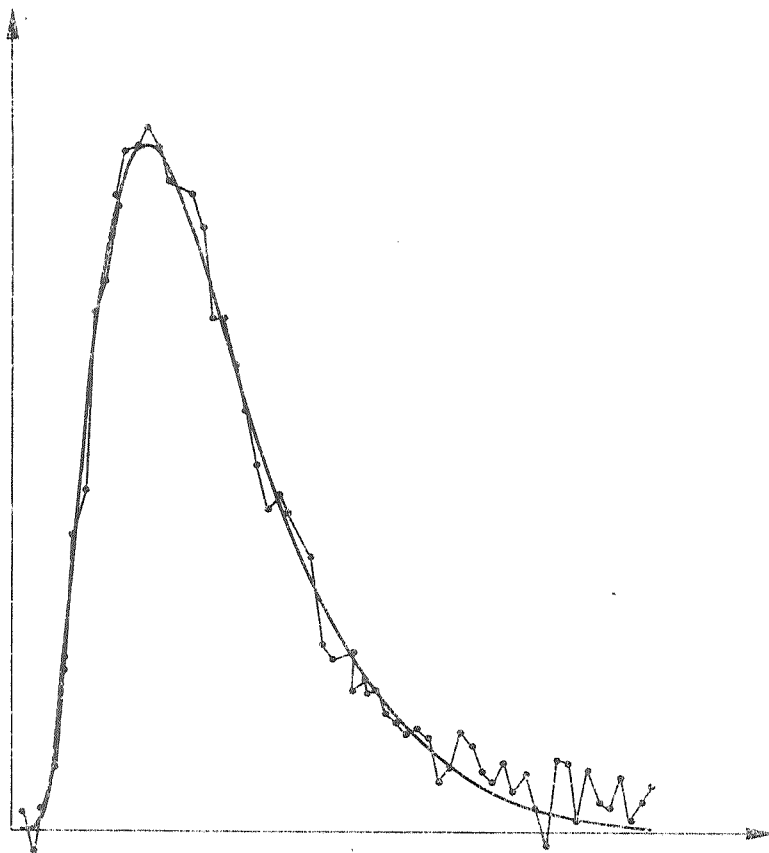
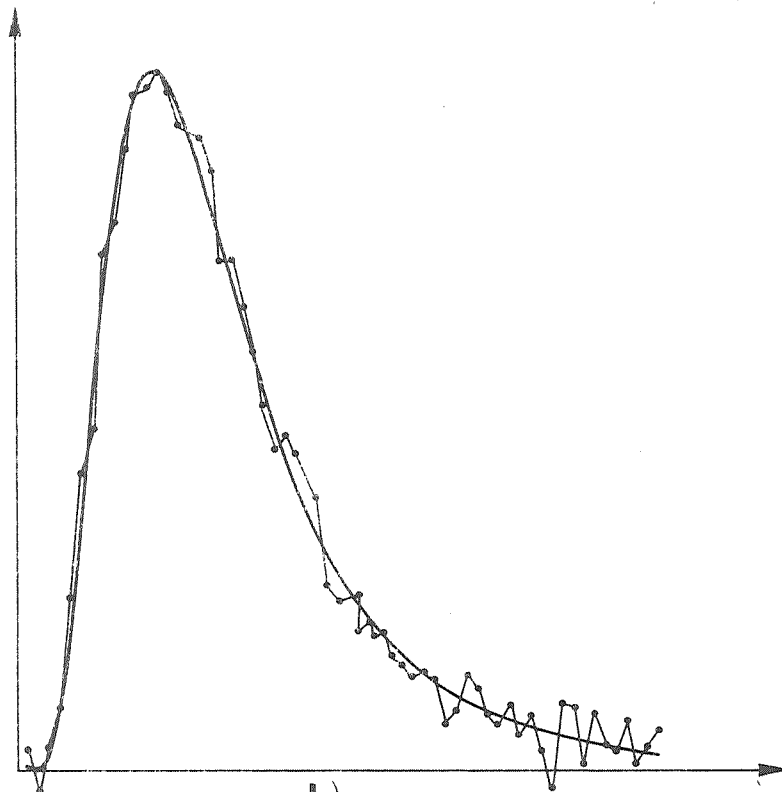


Fig.1- Schema a blocchi del sottoprogramma IDEN.



a)



b)

Fig.2- a) uscita teorica (—) e con rumore (—●) di un modello
 utilizzato per provare IDEN;
 b) uscita (—) del modello stimato.

```

SUBROUTINE IDEN(A,N,DA,ER)
C STIMA I PARAMETRI A(J) (J=1,N), LO STATO INIZIALE A(J)
C (J=N+1,2*N) DI UN MODELLO DI DIMENSIONE N; CALCOLA LE
C DERIVATE DA(J) (J=1,2*N) DELL'ERRORE ER RISPETTO AD A;
C CALCOLA IL VETTORE C(J) (J=1,N) DEI COEFFICIENTI DELL'IN-
C GRESSO.
C PARAMETRI DI INGRESSO:
C N: DIMENSIONE DEL SISTEMA
C Z: VETTORE DELLE OSSERVAZIONI (COMMON)
C U: VETTORE DEGLI INGRESSI (COMMON)
C M: NUMERO DELLE OSSERVAZIONI E DEGLI INGRESSI
C PARAMETRI DI USCITA:
C A(J) (J=1,N) :PARAMETRI DEL MODELLO
C A(J) (J=N+1,2*N): STATO INIZIALE
C DA(J) (J=1,2*N): DERIVATE DELL'ERRORE ER RISPETTO AD A
C ER: VALORE DELL'ERRORE IN A
C C(J) (J=1,N): COEFFICIENTI DEGLI INGRESSI NEL MODELLO (COMMON)
C SUBROUTINES DEL PACKAGE SSP-IBM USATE:
C DGELS, DFMCG, DMFGR
      IMPLICIT REAL*8 (A-G,O-Z)
      COMMON Z(1000),U(1000),C(10),M
      DIMENSION A(20),DA(20),DH(40),X(1010),S(400),L(20),SS(400)
      EXTERNAL FUNCT
C ESEGUE ALCUNI TENTATIVI PER CALCOLARE UN SET INIZIALE
C PER A(J) (J=1,N)
      NN=2*N
      MM=M-N
36 DO 30 I=1,NN
      D(I)=0.D+00
      DO 30 IT=1,MM
      IF(I.LE.N) D(I)=D(I)-Z(N+IT)*Z(N+IT-I)
      IF(I.GT.N) D(I)=D(I)+Z(N+IT)*U(I+IT-N-1)
30 CONTINUE
      DO 34 I=1,NN
      DO 34 J=1,NN
      JJ=(I-1)*NN+J
      S(JJ)=0.D+00
      DO 34 IT=1,MM
      IF(J.GE.I) GO TO 35
      K=(J-1)*NN+I
      S(JJ)=S(K)
      GO TO 34
35 IF(J.LE.N) S(JJ)=S(JJ)+Z(N-I+IT)*Z(N-J+IT)
      IF(I.GT.N) S(JJ)=S(JJ)+U(I-N+IT-1)*U(J-N+IT-1)
      IF(I.LE.N.AND.J.GT.N) S(JJ)=S(JJ)-Z(N-I+IT)*U(J-N+IT-1)
34 CONTINUE
      NNN=NN*NN
      NON=NM
      I=N+1
45 J=N+1
47 NEN=(J-1)*NON+I
      IF(S(NEN).EQ.0.) GO TO 46
      I=I+1
52 IF(I.LE.NON) GO TO 45
      GO TO 50
46 J=J+1
      IF(J.LE.NON) GO TO 47
      MWS=NON-1
      DO 48 J=1,NON

```



```

415  FORMAT(/,1X,' COEFFICIENTI C')
      WRITE(6,202) (C(J),J=1,N)
      WRITE(6,23)
23   FORMAT(1X,' STATO INIZIALE')
      WRITE(6,202) (A(J),J=NA,NO)
      ERX=ER/FLOAT(N)
      ER=DSQRT(ERX)
      WRITE(6,24) ER
24   FORMAT(1X,' ERRORE=',E15.7)
      WRITE(6,25)
25   FORMAT(1X,' DERIVATE')
      WRITE(6,202) (DA(J),J=1,NO)
2    WRITE(6,20)
20   FORMAT(1X, '//, 1X,' 1 CAMBIO A      2 PROSEGUO      3 RETURN')
      READ(5,*) ICON
      IF(ICON.EQ.1) GO TO 21
      IF(ICON.EQ.2) GO TO 1
      IF(ICON.EQ.3) RETURN
      END
      SUBROUTINE FUNCT(NN,A,ER,DA)
C    CALCOLA IL VETTORE DA DELLE DERIVATE DELL'ERRORE ER.
      IMPLICIT REAL*8 (A-G,O-Z)
      COMMON Z(1000),U(1000),C(10),M
      DIMENSION A(1),DA(1),XLP(10),XL(10),AX(10)
      DIMENSION X(1010)
      N=NN/2
      CALL FUNZ(A,N,ER,X)
      IF(ER.EQ.-.1D+01) GO TO 8
      DO 1 J=1,M
      XLP(J)=0.D+00
1     DA(J)=0.D+00
      DO 2 L=1,M
      I=N-L+1
      RX=0.D+00
      DO 3 KT=1,N
3     RX=RX-C(KT)*X(I+KT-1)
      XL(1)=2*(Z(I)+RX)*C(1)-A(N)*XLP(N)
      DO 4 J=2,M
4     XL(J)=2*(Z(I)+RX)*C(J)+XLP(J-1)-A(N-J+1)*XLP(N)
      DO 5 J=1,M
5     DA(J)=DA(J)+XLP(N)*X(I+N-J)
      DO 6 J=1,M
6     XLP(J)=XL(J)
2     CONTINUE
      DO 7 J=1,M
7     DA(N+J)=-XL(J)
8     RETURN
      END
      SUBROUTINE FUNZ(A,N,ER,X)
C    CALCOLA IL VETTORE C COL METODO DEI MINIMI QUADRATI E
C    L'ERRORE ER IN A.
      IMPLICIT REAL*8 (A-G,O-Z)
      COMMON Z(1000),U(1000),C(10),M
      DIMENSION D(100),A(1),IROW(10),ICOL(10),DB(100),CC(10)
      DIMENSION X(1)
      DO 1 J=1,N
1     X(J)=A(N+J)
      N1=N+1
      N2=N+M-1
      DO 2 J=N1,N2
      X(J)=U(J-N)
      DO 2 K=1,M
      L=J-K
      X(J)=X(J)-A(K)*X(L)

```

