



Dottorato di Ricerca in Ingegneria Civile

Graduate School in Civil Engineering

Sede: Facoltà di Ingegneria - Università di Pavia - via Ferrata 1 – 27100 Pavia – Italy

Dottorato di Ricerca in Ingegneria Civile XI Ciclo

Structural System Identification: Advanced Approaches and Applications

Tesi di Dottorato
Ing. Aurora Angela Pisano

Relatore:

Prof. Lucia Faravelli

Correlatore:

Prof. Mario Di Paola

Controrelatore:

Prof. Aldo Cauvin

February, 1999

A papà e mamma

Dottorato di Ricerca in Ingegneria Civile

Settore:	Ingegneria
Sede Amministrativa non consortile:	Università degli Studi di PAVIA
Durata del dottorato in anni:	3
Numero studenti:	3 ogni anno
Periodo formativo estero in mesi:	come previsto dal regolamento del Dottorato di Ricerca
Numero minimo di corsi:	6

Recapiti



Dipartimento di Meccanica Strutturale
via Ferrata 1 - 27100 Pavia - Italy
Tel. 0382 / 505450 Fax 0382 / 528422



Dipartimento di Ingegneria Idraulica e Ambientale
via Ferrata 1 - 27100 Pavia - Italy
Tel. 0382 / 505300 Fax 0382 / 505589

Coordinatore

CASCIATI Fabio - Professore Ordinario di Scienza delle Costruzioni (H07A)

Dipartimento di Meccanica Strutturale
via Ferrata 1 - 27100 Pavia – Italy Tel. 0382 / 505458 Fax 0382 / 528422
e-mail: fabio@dipmec.unipv.it

Collegio dei Docenti

- CAUVIN Aldo - Professore Ordinario di Teoria e Progetto delle Costruzioni in Calcestruzzo Armato e Precompresso (H07B)
- CIAPONI Carlo - Professore Associato di Idraulica (H01A)
- FARAVELLI Lucia - Professore Ordinario di Sicurezza e Affidabilità delle Costruzioni (H07A)
- FUGAZZA Mario - Professore Associato di Sistemazione dei Bacini Idrografici (H01B)
- GOBETTI Armando - Professore Associato di Dinamica delle Strutture (H07A)
- MACCHI Giorgio - Professore Ordinario di Tecnica delle Costruzioni (H07B)
- MOISELLO Ugo - Professore Ordinario di Idrologia (H01B)
- PAPIRI Sergio - Professore Associato di Infrastrutture Idrauliche (H01B)
- SALA Roberto - Professore Associato di Macchine (H04C)

Organizzazione del corso

Il dottorato di ricerca in *Ingegneria Civile* presso la Facoltà di Ingegneria dell'Università degli Studi di Pavia è stato istituito nell'anno accademico 1994/95 (X ciclo).

Il corso consente al dottorando di scegliere tra due curricula: idraulico o strutturale. Egli svolge la propria attività di ricerca rispettivamente presso il Dipartimento di Ingegneria Idraulica e Ambientale o quello di Meccanica Strutturale.

Durante i primi due anni sono previsti almeno sei corsi, seguiti da rispettivi esami, che il dottorando è tenuto a sostenere. Il Collegio dei Docenti, composto da professori dei due Dipartimenti, organizza i corsi con lo scopo di fornire allo studente di dottorato opportunità di approfondimento su alcune delle discipline di base per entrambe le componenti idraulica e strutturale. Corsi e seminari vengono tenuti da docenti di Università nazionali ed estere.

Il Collegio dei Docenti, cui spetta la pianificazione della didattica, si è orientato ad attivare ad anni alterni corsi sui seguenti temi:

- Meccanica dei solidi e dei fluidi
- Metodi numerici per la meccanica dei solidi e dei fluidi
- Rischio strutturale e ambientale
- Metodi sperimentali per la meccanica dei solidi e dei fluidi
- Intelligenza artificiale

più corsi specifici di indirizzo.

Al termine dei corsi del primo anno il Collegio dei Docenti assegna al dottorando un tema di ricerca da sviluppare sotto forma di tesina entro la fine del secondo anno; il tema, non necessariamente legato all'argomento della tesi finale, è di norma coerente con il curriculum, scelto dal dottorando (idraulico o strutturale).

All'inizio del secondo anno il dottorando discute con il Coordinatore l'argomento della tesi di dottorato, la cui assegnazione definitiva viene deliberata dal Collegio dei Docenti.

Alla fine di ogni anno i dottorandi devono presentare una relazione particolareggiata (scritta e orale) sull'attività svolta. Sulla base di tale relazione il Collegio dei Docenti, "previa valutazione della assiduità e dell'operosità dimostrata dall'iscritto", ne propone al Rettore l'esclusione dal corso o il passaggio all'anno successivo.

Il dottorando può svolgere attività di ricerca sia di tipo teorico che sperimentale, grazie ai laboratori di cui entrambi i Dipartimenti dispongono, nonché al Laboratorio Numerico di Ingegneria delle Infrastrutture.

Il "Laboratorio didattico sperimentale" del Dipartimento di Meccanica Strutturale dispone di:

1. una tavola vibrante che consente di effettuare prove dinamiche su prototipi strutturali;
2. opportuni sensori e un sistema di acquisizione dati per la misura della risposta strutturale;
3. strumentazione per la progettazione di sistemi di controllo attivo e loro verifica sperimentale;
4. strumentazione per la caratterizzazione dei materiali, attraverso prove statiche e dinamiche.

Il laboratorio del Dipartimento di Ingegneria Idraulica e Ambientale dispone di:

1. un circuito in pressione che consente di effettuare simulazioni di moto vario;
2. un tunnel idrodinamico per lo studio di problemi di cavitazione;
3. canalette per lo studio delle correnti a pelo libero.

A partire dall'anno accademico 1997/98 al dottorando viene data la possibilità di frequentare la "Scuola Avanzata di Formazione Integrata" dell'Istituto Universitario Studi Superiori, che si articola in tre anni e la cui finalità è quella di integrare le attività post-laurea di tipo specialistico con studi a carattere interdisciplinare adatti ad assicurare un più ampio bagaglio culturale.

Abstract

This thesis deals with the identification problem of structural systems subjected to dynamic loads. In particular, following a recent stream of the research activities in this field, the present study is focused on some numerical techniques based on the elaboration of time series functions of input-output of the structure. The time series, for real problems, are obtained from records, derived by testing apparatus and are then subjected to data acquisition problems. This latter aspect is here not treated, even if it is an active field of research, and the characteristic and the nature of the input-output data are assumed known.

The main purpose of this study is the development of system identification methods based on different principles and the validation of these methods for structural engineering applications. Moreover it is pointed out how the choice of the system identification technique depends on the specific final purpose of the identification.

The subject of structural identification techniques has been developed considering theoretical formulations and related numerical applications.

First the system realization theory is used for the identification of a multi-degree of freedom time-invariant linear system. This time domain technique ensures a good approximation of the response of the system, modeled through finite dimensional, first order, difference equations. Moreover the technique provides a systematic approach to the model order determination using the singular value decomposition method applied to the Hankel matrix of the Markov parameters of the system. Starting from a realization of the system is quite simple to derive its modal parameters, while is not so easy to derive mechanical quantities such as damping and stiffness. On the other side, realizations of a linear system are models which accurately express the system

dynamics, for this reason they are particularly suitable for control applications, as proposed in the following.

A second approach to the system identification problem utilizes neural networks. Neural networks are considered “black box” models. The unknown coefficients of the models are called “connection weights” and they are determined through an iterative procedure finalized to a minimization of the least squared error of the model. Neural networks present the ability to elaborate a large amount of data and to realize an associative memory.

In the following, multi-layer feedforward neural networks will be used to solve damage detection problems. In this context it is proposed to use the transfer functions of the structure, evaluated in different, simulated damage conditions, as input to the network.

The proposed approach to the damage identification problem allows to obtain satisfactory results even for complex structures as well as for non linear structures and for real time monitoring problems.

Finally a probabilistic approach to the system identification problem is proposed. In this context the potential models, which represent a particular class of stochastic models having solution in a probabilistic sense, are introduced. In fact the associated reduced Fokker-Planck equation can be solved in closed form and then, for this class of equations, the probability density function of the response is known. Under appropriate hypothesis (such as stationary and non parametric input, time invariant system, probability potential function expressible in polynomial form) the structural parameters of the system (damping and stiffness) can be derived solving a set of algebraic equations having as unknowns the aforementioned parameters and as coefficients the statistical moments of some functions related to the response.

All the presented methods are low influenced by the presence of noises in the input-output data.

It is opinion of the writer that the effectiveness and the friendly applicability of these methods make them useful tools for the solution of the identification problem in the structural engineering field.

Sommario

Questa tesi affronta il problema dell'identificazione di sistemi strutturali soggetti ad azioni dinamiche. In particolare, in linea con gli attuali interessi della ricerca in questo settore, l'attenzione è stata orientata verso quelle tecniche identificative basate sull'elaborazione delle serie temporali in ingresso e in risposta alla struttura, ottenibili per via sperimentale o attraverso simulazione numerica.

Le caratteristiche delle serie di input-output sono assunte note; non verranno pertanto trattati i problemi relativi all'acquisizione e caratterizzazione dei dati che costituiscono oggetto di altri settori di ricerca.

Attraverso questo studio, ci si è proposti di sviluppare metodi identificativi, basati su principi sostanzialmente diversi, e di appurarne le potenzialità e gli effettivi limiti di applicabilità ai sistemi strutturali, mettendo in evidenza come la loro scelta sia subordinata allo specifico scopo finale dell'identificazione.

Contributi alla ricerca sono presenti sia in termini di applicazioni che di sviluppi teorici.

Si è considerato dapprima il problema dell'identificazione di un sistema lineare tempo-invariante a più gradi di libertà. Si è dunque utilizzata la teoria di realizzazione dei sistemi che porta, nota la risposta impulsiva della struttura (parametri di Markov), alla determinazione delle matrici di stato del sistema. Questa tecnica, agendo nel dominio del tempo, consente una buona approssimazione della risposta del sistema, modellato attraverso equazioni alle differenze finite del primo ordine (rappresentazione in variabili di stato).

Il metodo utilizza una decomposizione della matrice di Hankel dei parametri di Markov del sistema, basata sul valore singolare, per determinare tra le infinite possibili realizzazioni quella di dimensioni minime.

Nota una realizzazione del sistema risulta immediato ricavare i parametri modali dello stesso, mentre non è altrettanto facile derivarne i parametri di dissipazione e di rigidità. Per questo motivo le applicazioni a problemi di diagnostica strutturale consentono solo in via qualitativa di individuare la presenza di danno nella struttura. D'altro canto i modelli di realizzazione riproducano accuratamente la dinamica del sistema così da trovare una naturale applicazione nei problemi di controllo strutturale, come viene proposto nel seguito.

Un secondo approccio al problema identificativo è quello che prevede l'utilizzo di reti neurali. Le reti neurali sono dei modelli, non basati fisicamente, in cui i parametri incogniti, pesi di connessione, vengono determinati alla fine di un processo iterativo che porta alla minimizzazione dell'errore quadratico medio totale della rete. Questi modelli sono in grado di elaborare un numero considerevole di dati e sviluppano una memoria associativa che consente loro di mettere in relazione serie di dati, aventi caratteristiche intrinseche simili, a classi corrispondenti.

Nel seguito si utilizzeranno reti neurali, del tipo "feedforward" a più strati, per la risoluzione di problemi di diagnostica strutturale. È da sottolineare che l'efficacia di questo strumento di calcolo dipende, in parte, dal tipo di dati che deve elaborare e dalla loro organizzazione.

Nella tesi si propone di utilizzare le funzioni di trasferimento della struttura, valutate in diverse (simulate) condizioni di danno, come input alla rete neurale, associando a ciascuna funzione di trasferimento una corrispondente classe di danno.

L'approccio proposto consente di ottenere buoni risultati in campo diagnostico anche in presenza di strutture complesse e a comportamento non lineare. Esso si presta inoltre ad essere impiegato in problemi di monitoraggio in tempo reale.

Infine si propone un approccio probabilistico al problema identificativo. In questo contesto vengono introdotti i modelli a potenziale che costituiscono una classe di modelli stocastici aventi soluzione in senso probabilistico. Infatti

l'equazione ridotta di Fokker-Planck, associata ai modelli a potenziale, ammette soluzione in forma chiusa e dunque per queste equazioni è nota analiticamente la funzione densità di probabilità del processo risposta.

Utilizzando questi modelli, sotto opportune ipotesi (tra cui input stazionario e non parametrico, sistema tempo invariante, potenziale di probabilità esprimibile in forma polinomiale), il problema dell'identificazione dei parametri caratterizzanti il sistema strutturale (smorzamento e rigidità) si traduce nella risoluzione di un sistema di equazioni algebriche aventi per incognite i parametri suddetti e per coefficienti i momenti statistici di alcune funzioni della risposta.

I metodi presentati hanno in comune il pregio di risentire limitatamente della presenza di disturbi nei dati utilizzati. È opinione della scrivente che essi siano di facile applicabilità, nonché di comprovata efficacia, costituendo strumenti fondamentali per la risoluzione del problema identificativo in campo strutturale.

Acknowledgements

I would like to express my sincere gratitude to Prof. Lucia Faravelli who gave me the opportunity to carry out this Ph.D work. Her useful suggestions, patient guidance as well as her human support were essential.

I would like to thank Prof. Mario Di Paola from the *Dipartimento di Ingegneria Strutturale e Geotecnica* of the University of Palermo for his constant encouragements and stimulating discussions during the course of this thesis.

Special thanks go also to Prof. Fabio Casciati and to Prof. Aldo Cauvin for the attention reserved to my work and the suggestions given to improve it.

Last I would like to express my appreciation for all the people of the *Dipartimento di Meccanica Strutturale* of the University of Pavia for their hospitality.

Contents

Abstract	IX
Sommario	XI
Acknowledgements	XIV

Chapter

1 Introduction	
1.1 Outline of the dissertation	5
2 Eigensystem Realization Theory for System Identification	
2.1 Introduction	7
2.2 Equivalent representations for structural dynamic systems	9
2.2.1 Second order differential equation	10
2.2.2 The state space representation	10
2.2.3 Representation of the system through Markov parameters	12
2.3 Eigensystem realization algorithm	15
2.4 Kalman filter	21
2.5 The Observer Kalman filter	23
2.6 A practical application	26
2.6.1 Preliminary considerations	26
2.6.2 The implemented procedure	27
2.6.3 The control law	30

2.6.4	A numerical example	32
2.7	Conclusion	43

3 The Neural Network Approach for Damage Assessment and Safety Analysis

3.1	Introduction	45
3.2	The network architecture	47
3.3	Learning algorithms	51
3.3.1	The back-propagation learning rule	51
3.3.2	The modified back-propagation algorithm	55
3.4	The proposed approach to the damage detection problem	58
3.4.1	The damage characteristic function	58
3.4.2	Numerical applications	61
3.4.3	Example 1	61
3.4.4	Example 2	69
3.4.5	Example 3	73
3.4.5.1	Reliability of the structure via fragility curve	73
3.4.5.2	Application and results	76
3.5	Conclusion	85

4 Potential Systems for System Identification

4.1	Introduction	87
4.2	The Fokker Planck equation	89
4.3	Generalized stationary potential	92
4.4	A reduced class of potential system	94
4.5	Some fundamental properties	97
4.6	Identification of SDOF non-linear system	102
4.6.1	Identification of the stiffness parameters	102
4.6.2	Identification of the damping parameters	104
4.6.3	Numerical application	104
4.7	Error effected data	107
4.8	Identification of MDOF system	110
4.9	Conclusion.....	114

5 Conclusions and Remarks	115
References	119
Appendices	
Appendix 2.A	
2.1 A The Mode Singular Value (MSV).....	125
2.2 A The Modal Amplitude Coherence (MAC).....	127
Appendix 2.B	
2.1 B The Markov parameters of the system	125
Appendix 2.C	
2.1 C The Kalman filter	130
2.1 C The Riccati equation	133
Appendix 2.D	134
Appendix 3.A	
3.1 A Foundation of Neural Networks	138
3.2 A Schemes of Neural Networks	141
Appendix 3.B	
3.1 B The learning algorithms.....	146

Chapter 1

Introduction

To identify a structure means essentially to be able to reproduce its behavior under an external excitation. This can be achieved by solving a general system identification problem or by solving a structural system identification problem.

The system identification problem is generally stated as the construction of mathematical models from observer data which can best fit the input-output relations, regardless to physical interpretations, while structural system identification involves the determination of the intrinsic structural parameters, such as normal modes, mode shapes, stiffness and damping parameters.

The general subject of system identification originally began in the area of electrical engineering and only later it has been extended to the fields of the mechanical and civil engineering; here it has become an increasingly important area of research because structural identification can be viewed as a first and fundamental step in *(i)* estimating and monitoring the health of an existent structure, *(ii)* in control problems and *(iii)* to perform reliability analysis of the structure.

A very extensive and diverse literature on the subject is available. Several methods have been proposed and they can be grouped in different classes. Generally a distinction is made between parametric and non-parametric

methods, time domain and frequency domain methods and, in the field of civil engineering, also between static and dynamic methods.

General surveys on the subject can be found in Ljung (1987), Kozin and Natke (1986), Juang (1994), Natke and Campel (1997) and in several review papers, Kailath et al. (1979), Eikhoff (1981), Isermann (1981), Imai et al. (1989), among others.

It is not the aim of this work to present a comprehensive survey of achievements in this field and of the state of art.

Among the various methods only three will be treated in the detail: the first, based on the system realization theory, is applicable to linear systems, the second and the third, based respectively on neural networks and on potential systems theory, are applicable to linear and non-linear systems.

In the follows, to each of these methods a chapter will be dedicated.

Of course the aforementioned methods are not unconditionally applicable to every identification problem, on the contrary they have strong limitations and must be used appropriately. The general considerations reported in the following, and schematized in fig.(1.1), could help to define a correct approach to the problem and to choose the more adequate system identification technique.

- First it is important to have in mind the purposes of the system identification procedure, in fact different purpose may need a different identification process to perform a system identification task. If one deals with control problems, such as control of large aerospace structures, the final goal could be the development of a control strategy. The identification task is then to find a model of the structure which adequately will describe the input and output map. On the other hand, if monitoring and damage detection in a structure are the purposes of the study, the system identification task could be to identify the properties of the dynamic system such as stiffness, damping frequencies and their changes in time. Finally, if the final goal of the study is to develop a vulnerability analysis of the structure during its operating time, the applied system identification technique should be able to capture the modifications of characteristic functional of the structure and use it to

determine the probability of failure of the structure.

- The second concept that is important to outline regards the selection of a set of models. To give a broad definition, a model of a system is a framework describing the relationship among the system variables in terms of mathematical expressions like integral or differential equations. The selection of a model, in a system identification problem, directly depends on the purpose of the identification and influences the choice of the identification technique. Usually physical based models (the ones containing the parameters of the system) lead to parametric system identification, while black box models imply non-parametric identification. Moreover the model may be deterministic or stochastic depending of the type of the external forces and on the characteristics of the structure under investigation.

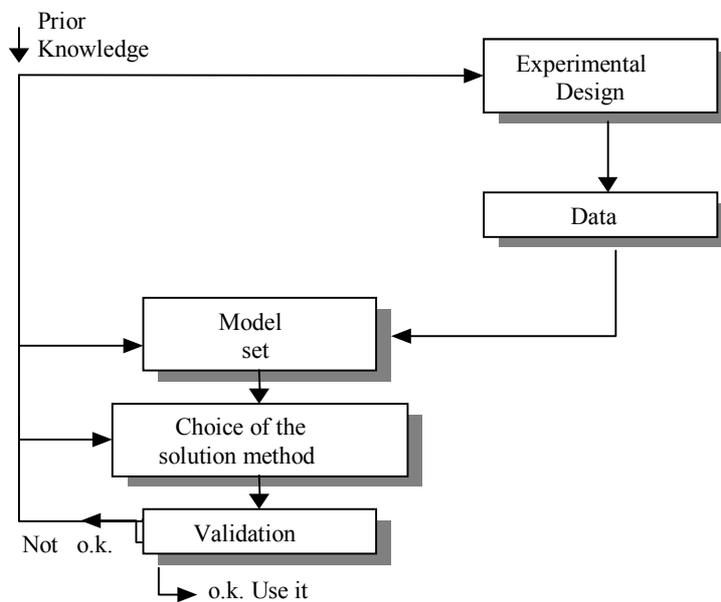


Figure 1.1: System identification procedure scheme.

For a given system, the choice of a correct model is of fundamental importance, but, very often, really difficult. Real structural systems are generally large in size and complex in behavior so that their accurate mathematical idealization is not an easy task; the options for input-output observational data are usually limited and the measured data could contain noises; finally in case of damage the system behavior may be highly non-linear. From these considerations one can see that with the model the first inaccuracies are introduced into the problem. Model inaccuracies effect the accuracy of the prediction.

- To reduce the uncertainties a validation of the model is necessary. This involves various procedures to assess how the model relates to the observed data, if available, or to simulated data in other cases; in fact in such contexts data generated according to some well-defined mathematical rules are assumed as *true*, which, of course, is an idealization.

The present work is primarily aimed to the development and validation of system and structural system identification techniques. The practical applications in structural engineering problems is also stressed. A particular attention is devoted to those techniques based on the elaboration of measured input-output data. In this context three totally different approaches to the problem will be developed showing how each of these is particularly suitable to solve a specific problem such as damage detection, control and monitoring of the structure.

The large amount of literature existing on this subject makes necessary to give some guidelines about the effective potentiality of the various methods in the field of structural engineering.

New research contributions are presented through numerical applications, in the second and third chapters, and through theoretical developments in the fourth chapter.

1.1 Outline of the dissertation

The content of the present thesis is organized as follows:

The second chapter deals with the identification of a multi-degree of freedom linear system by means of the eigensystem realization theory. Throughout the chapter will be shown how the realization of the system, combined with the observer Kalman filter theory, leads to the formulation of a systematic and stable procedure for the identification of the dynamics of the system. These informations will be used to treat the damage detection problem in a structural system and, together with an implemented control law, to control the dynamic system behavior.

Chapter three deals with the problems of monitoring and damage assessment in a structure using the neural network approach. Multi-degree of freedom, linear and non-linear systems will be analyzed. Particular attention will be given to the feedforward, multi-layers network type. As learning set for the network the transfer functions of the structure, evaluated in different damage conditions, will be used. The chapter provides also several numerical applications.

Chapter four shows a probabilistic approach to the system identification problem. To this end a particular class of potential systems will be introduced and it will be shown how the parameters of the system can be identified solving a set of algebraic equations having as coefficients the statistical moments of the system response and as unknowns the parameters itself.

Single and multi-degree of freedom stochastic models will be studied.

Finally chapter five contains the general conclusions and some remarks for further research.

Chapter 2

Eigensystem Realization Theory for System Identification

2.1 Introduction

In structural, mechanical and aerospace engineering fields, a large number of system identification techniques are available in order to develop a mathematical model of the system using experimental data. Furthermore for linear systems, numerous algorithms are based on the state-space representation of the system.

This chapter deals with the identification of the parameters of the state-space model of the system, starting from given input-output data. These parameters may then be used to solve specific problems such as damage identification and control of a structural system.

The theoretical framework of the system realization theory is founded on the works of Kalman (1963) and Ho and Kalman (1965). Among the infinite number of possible realizations of a system, with the same degree of accuracy, there is one that has minimum order. Minimum realization means a model with the smallest state-space dimension. Ho and Kalman showed that the minimum

realization problem is equivalent to a representation problem involving sequences of matrices containing the Markov parameters of the system; being the Markov parameters the pulse response of the discrete time system.

This technique was combined with the singular value decomposition technique by Zieger and McEwen (1974), for the treatment of noisy data. The evolutions of the research in this direction resulted in the development of the Eigensystem Realization Algorithm (ERA) by Juang and Pappa (1985). The ERA algorithm provides modal parameter estimation and a modal reduction of the dynamic system.

The analysis in the time domain, for obtaining Markov parameters of the system from the input and output data, usually presents the drawback of inverting an input matrix, which becomes particularly large for lightly damped systems. For this reason, rather than identify the system Markov parameters, which may exhibit very slow convergence, one can use an asymptotically stable observer so that a stable discrete state space model for the system is identified.

This concept is implemented in the so-called Observer/Kalman filter identification algorithm (OKID), Juang (1992). A fundamental innovation in this approach lies in the presence of the observer that allows an artificial compression of the data and as a consequence an improvement of the identification analysis. The method can be regarded as an adaptive filtering approach, which does not require prior statistical information and does not rely on sample correlation or covariance calculations. Moreover, one can assign the desired poles for the observer so that it is possible to specify the decay rate of the observer Markov parameters to be determined from the data.

The uniqueness and invertibility of the transformation from the observer Markov parameters to the system Markov parameters are also ensured by a matrix formulation, Phan et al. (1992). Once the observer Markov parameters are obtained, OKID uses the eigensystem realization algorithm to realize the state space model from the observer and the system Markov parameters.

This approach makes identification possible not only for the open-loop system, but also for an associated observer that can be used in controller design.

Beside the observer/Kalman filter identifier an equivalent state-space frequency-domain method has been developed, for the cases in which frequency response data are available instead of time histories. The method involves directly the frequency response function to estimate the Markov parameters using a rational matrix description to fit the frequency data and then recalls ERA to obtain the state space model from the Markov parameters. In this way the problems connected with the aliasing effects, introduced during the inverse discrete Fourier transform of the frequency response function, are avoided.

Another interesting aspect, from a structural engineering point of view, is that it is possible to transform the system state-space based realization model into the corresponding physical coordinate based structural model, (Alvin and Park, 1994). Furthermore it can be shown that it is possible to carry out a unique set of structural parameters from an infinite set of equivalent realization models. The existence of this link between the two classes of system identification techniques, (the one based on the state-space representation and the other on the second order structural dynamic equation) makes the methods based on the state space representation more attractive due to their simplicity and robustness.

In the following the problem of damage detection and subsequent restoration of a structural system will be treated by means of the system realization theory. To this end a control algorithm, based on the assignment of the poles, is also implemented. To clarify the adopted procedure, some fundamental concepts regarding the state-space representation of a structural system, the singular value decomposition, as well as the Kalman filter theory and the poles assigned control law will be reviewed.

2.2 Equivalent representations for structural dynamic systems

Models in the time domain as well as in the frequency domain may be used to represent a linear dynamic system. The models are said equivalent when they

reproduce the same input-output mapping. The necessity to have different representations of the same system is due to the fact that some models are more suitable for computation than others.

In the following a brief resume of equivalent representations for a linear dynamic structure will be presented. Starting from the equation of motion, the transformation into the state space form and a representation as weighting sequence will be reported.

2.2.1 Second order differential equation

Typically a linear, time-invariant dynamic system, which may represent a real structure in a discrete form, having a finite dimension, say p , is described by a set of p second order differential equations expressed in matrix form as:

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{C}_0\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{F}(\mathbf{q}, t) \quad (2.2.1.1)$$

where the familiar symbols \mathbf{M} , \mathbf{C}_0 and \mathbf{K} are the p -order square matrices of mass, damping and stiffness respectively; $\ddot{\mathbf{q}}(t)$, $\dot{\mathbf{q}}(t)$ and $\mathbf{q}(t)$ are vectors of acceleration, velocity and displacement, while $\mathbf{F}(\mathbf{q}, t)$ is the forcing vector function.

2.2.2 The state-space representation

The above set of second order differential equations may be transformed into a first order system of differential equations by means of the following definitions:

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{q}(t) \\ \dot{\mathbf{q}}(t) \end{bmatrix}; \quad \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C}_0 \end{bmatrix}; \quad (2.2.2.1 \text{ a-b})$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\hat{\mathbf{B}} \end{bmatrix}; \quad \mathbf{F}(\mathbf{q}, t) = \hat{\mathbf{B}}\mathbf{u}(t); \quad (2.2.2.1 \text{ c-d})$$

where the product $\hat{\mathbf{B}}\mathbf{u}(t)$ represents the forcing function being $\hat{\mathbf{B}}$ the $(p \times m)$ influence matrix, with m the number of inputs, and $\mathbf{u}(t)$ the input force vector.

Then denoting with \mathbf{A} the $(n \times n)$ state matrix of the system, with $n = 2p$, and denoting with $\mathbf{x} \in R^n$ the state vector of the system, a more compact form of the system is given by:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t). \quad (2.2.2.2)$$

If r structural responses are available then a matrix output equation can be written in the form:

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t). \quad (2.2.2.3)$$

Here $\mathbf{y} \in R^r$ is the output variable, \mathbf{C} is an $(r \times n)$ output influence matrix for the state vector \mathbf{x} , including only velocity and displacement, and \mathbf{D} is an $(r \times m)$ direct transmission matrix. The matrix \mathbf{D} is the output influence matrix for the acceleration and disappears when accelerations are not measured in output.

Equations (2.2.2.2) and (2.2.2.3) constitute a continuous time state space model of the structural system. Formally the same equations can be written in the discrete case, when the input-output data are available in discrete form.

In this case, eqs.(2.2.2.2) and (2.2.2.3) are described by difference equations instead of differential equations, as:

$$\mathbf{x}(k+1) = \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \mathbf{u}(k) \quad (2.2.2.4)$$

$$\mathbf{y}(k) = \mathbf{C}_d \mathbf{x}(k) + \mathbf{D}_d \mathbf{u}(k) \quad (2.2.2.5)$$

$\mathbf{x}(k)$ means the vector \mathbf{x} computed at the discrete time $k \cdot \Delta t$, where Δt indicates the sampling time interval. The notation d refers to the discrete representation of the system. The discrete time matrices may be computed by the following series expansion of the corresponding matrices in continuous time:

$$\mathbf{A}_d = e^{\mathbf{A}\Delta t} = \mathbf{I} + \mathbf{A}\Delta t + \frac{1}{2!}(\mathbf{A}\Delta t)^2 + \frac{1}{3!}(\mathbf{A}\Delta t)^3 + \dots \quad (2.2.2.6)$$

$$\mathbf{B}_d = \int_0^{\Delta t} e^{\mathbf{A}\tau} d\tau \cdot \mathbf{B} = \left[\mathbf{I}\Delta t + \frac{1}{2!}\mathbf{A}(\Delta t)^2 + \frac{1}{3!}\mathbf{A}^2(\Delta t)^3 + \dots \right] \cdot \mathbf{B}. \quad (2.2.2.7)$$

These series converge if the matrix \mathbf{A} is asymptotically stable. Equations (2.2.2.6) and (2.2.2.7) allow switching from the continuous to the discrete time formulation.

As will appear clear in the following, if an observer is introduced into the system then the eigenvalues of the new state matrix can be moved in a way that their real parts are negative.

In the following for simplicity, the system matrices in the discrete form will be denoted without the subscription d , the use of difference or differential equations will make clear if they refer to the discrete or to the continuous model.

2.2.3 Representation of the system through Markov parameters

As already introduced, the Markov parameters of a system constitute the unit pulse response sample of the system. They can be easily obtained, for given initial conditions, by solving eqs.(2.2.2.4) and (2.2.2.5) in terms of previous input $\mathbf{u}(i)$, $i=0, \dots, k$.

For $\mathbf{x}(0) = \mathbf{0}$ the solution for the output $\mathbf{y}(k)$ is given by:

$$\mathbf{y}(k) = \sum_{i=0}^k \mathbf{Y}(k-i) \mathbf{u}(i) \quad (2.2.3.1)$$

where the constant matrices in the summation are the Markov parameters defined as:

$$\mathbf{Y}(0) = \mathbf{D}, \quad \mathbf{Y}(1) = \mathbf{CB}, \quad \mathbf{Y}(2) = \mathbf{CAB}, \quad \mathbf{Y}(k) = \mathbf{CA}^{k-1}\mathbf{B}. \quad (2.2.3.2)$$

Equation (2.2.3.1) can be written in matrix form as:

$$\bar{\mathbf{y}} = \bar{\mathbf{Y}} \bar{\mathbf{U}} \quad (2.2.3.3)$$

with $\bar{\mathbf{y}}$, $\bar{\mathbf{Y}}$ and $\bar{\mathbf{U}}$ having dimensions (rxk) , $(rxmk)$ and $(mkxk)$ respectively.

From eq.(2.2.3.1) it can be seen that the output of a linear system can be expressed through a weighted sum of the applied input, with the weights being the Markov parameters of the system. Thus the Markov parameters characterize the unit impulse input-output relationships of the system and as thus they are unique for the system. Any equivalent realization of the system must preserve the Markov parameters.

Moreover the solution of eq.(2.2.2.4) has the form:

$$\mathbf{x}(k) = \sum_{i=1}^k \mathbf{A}^{i-1} \mathbf{B} \mathbf{u}(k-i). \quad (2.2.3.4)$$

It is easy to show that any system has an infinite number of possible realizations, as in fact if a generic transformation of the state vector is introduced, say $\mathbf{x}=\mathbf{Tz}$, with \mathbf{T} being a non-singular square matrix, eqs.(2.2.2.4) and (2.2.2.5) can be rewritten as:

$$\mathbf{z}(k+1) = \mathbf{T}^{-1}\mathbf{AT} \mathbf{z}(k) + \mathbf{T}^{-1}\mathbf{B} \mathbf{u}(k) \quad (2.2.3.5)$$

$$\mathbf{y}(k) = \mathbf{C} \mathbf{T} \mathbf{z}(k) + \mathbf{D} \mathbf{u}(k). \quad (2.2.3.6)$$

From this new set of equations it can be noted that the effect of the input $\mathbf{u}(k)$ on the output $\mathbf{y}(k)$ has not changed, so the new set of matrices $\mathbf{T}^{-1}\mathbf{A}\mathbf{T}$, $\mathbf{T}^{-1}\mathbf{B}$ and $\mathbf{C}\mathbf{T}$ as well as \mathbf{A} , \mathbf{B} , \mathbf{C} (\mathbf{D} is not affected by the transformation) represent a realization of the system.

The coordinate transformation \mathbf{T} is called a *similarity* transformation because the eigenvalues λ of \mathbf{A} are invariant under this transformation. Because of this, the two matrices \mathbf{A} and $\mathbf{T}^{-1}\mathbf{A}\mathbf{T}$ are said to be similar. Moreover all the similarity transformations preserve the Markov parameters of the system. Referring to eq.(2.2.3.2) the Markov parameters are given by:

$$\mathbf{Y}(0) = \mathbf{D} \quad (2.2.3.7)$$

$$\mathbf{Y}(k) = \mathbf{C}\mathbf{T}(\mathbf{T}^{-1}\mathbf{A}\mathbf{T})^{k-1}\mathbf{T}^{-1}\mathbf{B} = \mathbf{C}\mathbf{A}^{k-1}\mathbf{B}.$$

Another important property of the similarity transformation is that it preserves the transfer functions \mathbf{H} from \mathbf{u} to \mathbf{y} . In fact applying the Laplace transform to eqs.(2.2.3.5) and (2.2.3.6) one obtains:

$$\begin{aligned} \mathbf{H}(s) &= \frac{\mathbf{y}(s)}{\mathbf{u}(s)} = \mathbf{C}\mathbf{T}(s\mathbf{T}^{-1}\mathbf{T} - \mathbf{T}^{-1}\mathbf{A}\mathbf{T})^{-1}\mathbf{T}^{-1}\mathbf{B} + \mathbf{D} = \\ &= \mathbf{C}\mathbf{T}(s\mathbf{I} - \mathbf{A}^{-1})\mathbf{B} + \mathbf{D}. \end{aligned} \quad (2.2.3.8)$$

where s is the Laplace variable.

Due to the fact that no restriction has been introduced in the choice of \mathbf{T} it is therefore proved that there are an infinite number of equivalent state-space representations that produce the same input-output relationship.

A realization problem starts from the knowledge of the Markov parameters and can be stated as follows:

given a sequence of Markov parameters $\mathbf{Y}(k)$ of the system, determine a realization $[\mathbf{A}, \mathbf{B}, \mathbf{C}]$ which best approximates the given Markov sequence according to some measure of accuracy :

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \quad (2.2.3.9)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) \quad (2.2.3.10)$$

such that :

$$\mathbf{Y}(0) = \mathbf{D} \quad (2.2.3.11)$$

$$\mathbf{Y}(k) \approx \mathbf{C}\mathbf{A}^{k-1}\mathbf{B}$$

where \mathbf{x} is the resultant state vector.

The order of the realization is related to the number of inputs and outputs regardless to the order of the physical system and this, in some sense, is an advantage. On the other hand care should be taken to ensure that there is a sufficient amount of information in order to entirely reproduce the behavior of the system; as will be shown an optimal compromise can be reached.

Solution of the realization problem concerns issues such as model order, uniqueness, noise, model measures to be fitted, etc.

2.3 Eigensystem Realization Algorithm

As previously stated, the problem of a system realization consists of finding a set of constant matrices \mathbf{A} , \mathbf{B} , \mathbf{C} from the Markov parameters for which equations (2.2.2.4) and (2.2.2.5) are satisfied. Among the infinite solutions the one with minimum order is obviously the most attractive.

Ho and Kalman (1965) introduced the principles of the minimum realization theory for which from the infinite number of realizations for a system it can be extracted a class that has minimum order. All the minimum realizations have the same set of eigenvalues. This concept allowed the development of a technique that provides a systematic approach to model order determination, for given accuracy, and the derivation of the discrete state space model.

The algorithm developed by Ho and Kalman uses the discrete-time shift of the Markov parameters, which are used to form a Hankel matrix, defined as follows:

$$\mathbf{H}_{\alpha\beta}(k) = \begin{bmatrix} \mathbf{Y}(k+1) & \mathbf{Y}(k+2) & \dots & \mathbf{Y}(k+\beta) \\ \mathbf{Y}(k+2) & \mathbf{Y}(k+3) & \dots & \mathbf{Y}(k+\beta+1) \\ \dots & \dots & \dots & \dots \\ \mathbf{Y}(k+\alpha) & \mathbf{Y}(k+\alpha+1) & \dots & \mathbf{Y}(k+\alpha+\beta-1) \end{bmatrix}. \quad (2.3.1)$$

Juang and Pappa (1985) introduced the use of the singular value decomposition (SVD) of the Hankel matrix to generalize the Ho-Kalman algorithm to structural system identification. The new algorithm, known as Eigensystem Realization Algorithm, ERA, expresses the measured Hankel matrix as:

$$\mathbf{H}_{\alpha\beta}(k) = \mathbf{R} \mathbf{\Sigma} \mathbf{S}^T = \sum_{i=1}^{N_{\max}} s_i \mathbf{r}_i \mathbf{z}_i^T \quad (2.3.2)$$

where \mathbf{R} and \mathbf{S} are orthonormal ($\alpha \times \alpha$) and ($\beta \times \beta$) matrices, composed of column vectors \mathbf{r}_i and \mathbf{z}_i respectively, and $\mathbf{\Sigma}$ is an ($\alpha \times \beta$) matrix with the singular values s_i of $\mathbf{H}_{\alpha\beta}(k)$ on the main diagonal and zero elsewhere; N_{\max} is the minimum of α and β .

Minimum model order is determined by minimizing the matrix norm between the measured and realized Hankel matrix $\mathbf{H}_{\alpha\beta}(k)$ and $\hat{\mathbf{H}}_{\alpha\beta}(k)$, so that:

$$\min \left\| \mathbf{H}_{\alpha\beta}(k) - \hat{\mathbf{H}}_{\alpha\beta}(k) \right\|. \quad (2.3.3)$$

The realized Hankel matrix $\hat{\mathbf{H}}_{\alpha\beta}(k)$ can be determined by truncating the SVD series expansion at a suitable order $N < N_{max}$ such that $s_{N+1} < \varepsilon \cong 0$. Therefore using the measured Hankel matrix $\mathbf{H}_{\alpha\beta}(0)$ the realized Hankel matrix $\hat{\mathbf{H}}_{\alpha\beta}(0)$ is:

$$\hat{\mathbf{H}}_{\alpha\beta}(0) = \mathbf{R}_N \boldsymbol{\Sigma}_N \mathbf{S}_N^T \quad (2.3.4)$$

where $\boldsymbol{\Sigma}_N$ is a diagonal ($N \times N$) matrix whose diagonal elements are the singular values up to order N of $\mathbf{H}_{\alpha\beta}(0)$ decreasing ordered; while \mathbf{R}_N and \mathbf{S}_N are in general rectangular matrices ($\alpha \times N$) and ($\beta \times N$) respectively, they are still orthonormal such as $\mathbf{R}_N^T \mathbf{R}_N = \mathbf{S}_N^T \mathbf{S}_N = \mathbf{I}$ and are obtained by extracting the first N columns of \mathbf{R} and \mathbf{S} respectively. On the other hand the Hankel matrix can be directly expressed by means of the state matrix \mathbf{A} , the controllability \mathbf{P}_α and observability \mathbf{Q}_β matrices by means of:

$$\mathbf{H}_{\alpha\beta}(k) = \mathbf{P}_\alpha \mathbf{A}^k \mathbf{Q}_\beta \quad (2.3.5)$$

where

$$\begin{aligned} \mathbf{P}_\alpha &= [\mathbf{C} \quad \mathbf{C}\mathbf{A} \quad \mathbf{C}\mathbf{A}^2 \quad \dots \quad \mathbf{C}\mathbf{A}^{\alpha-1}]^T; \\ \mathbf{Q}_\beta &= [\mathbf{B} \quad \mathbf{A}\mathbf{B} \quad \mathbf{A}^2\mathbf{B} \quad \dots \quad \mathbf{A}^{\beta-1}\mathbf{B}]. \end{aligned} \quad (2.3.6)$$

For $k = 0$ and assuming the new order N for the system, making a comparison between eqs.(2.3.4) and (2.3.5) a relationship between \mathbf{R}_N , \mathbf{S}_N , $\boldsymbol{\Sigma}_N$ and \mathbf{P}_α and \mathbf{Q}_β can be found. One possible choice is the following:

$$\mathbf{P}_\alpha = \mathbf{R}_N \boldsymbol{\Sigma}_N^{1/2}; \quad \mathbf{Q}_\beta = \boldsymbol{\Sigma}_N^{1/2} \mathbf{S}_N^T. \quad (2.3.7)$$

whit this choice \mathbf{P}_α and \mathbf{Q}_β have an equal distribution of the singular values of $\mathbf{H}_{\alpha\beta}(0)$, and in fact it is easy to show that the controllability and observability grammians are equal and diagonal, which implies an internally balanced realization, therefore the realized system is observable and controllable.

Using eqs.(2.3.7) for $k = 1$ eq.(2.3.5) becomes:

$$\mathbf{H}_{\alpha\beta}(1) = \mathbf{R}_N \boldsymbol{\Sigma}_N^{1/2} \hat{\mathbf{A}} \boldsymbol{\Sigma}_N^{1/2} \mathbf{S}_N^T \quad (2.3.8)$$

from which a minimum order realization can be obtained as:

$$\hat{\mathbf{A}} = \boldsymbol{\Sigma}_N^{-1/2} \mathbf{R}_N^T \mathbf{H}_{\alpha\beta}(1) \mathbf{S}_N \boldsymbol{\Sigma}_N^{-1/2}, \quad \hat{\mathbf{B}} = \boldsymbol{\Sigma}_N^{1/2} \mathbf{S}_N^T \mathbf{E}_m \quad (2.3.9.a,b)$$

$$\hat{\mathbf{C}} = \mathbf{E}_r^T \mathbf{R}_N \boldsymbol{\Sigma}_N^{1/2}, \quad \hat{\mathbf{D}} = \mathbf{Y}(0) \quad (2.3.10.a,b)$$

where $\mathbf{E}_m^T = [\mathbf{I}_m \ \mathbf{O}_m \ \dots \ \mathbf{O}_m]$ and $\mathbf{E}_r^T = [\mathbf{I}_r \ \mathbf{O}_r \ \dots \ \mathbf{O}_r]$. The symbol \mathbf{I}_m denotes the $(m \times m)$ identity matrix, \mathbf{O}_m the $(m \times m)$ null matrix and m and r respectively the dimension of input and output.

When, for a given set of Markov parameters, a minimum realization is found, from the identified state matrix it is possible to derive its eigenvalues and eigenvectors. The eigen-quantities can be used to produce a modal representation of the system. The modal realization is specified by the matrices $\boldsymbol{\Lambda}$, $\boldsymbol{\Psi}^{-1} \hat{\mathbf{B}}$ and $\hat{\mathbf{C}} \boldsymbol{\Psi}$, being $\boldsymbol{\Lambda}$ a diagonal matrix containing the eigenvalues λ_i of the state matrix and $\boldsymbol{\Psi}$ the corresponding eigenvectors matrix. The real part of $\boldsymbol{\Lambda}$, into the continuous time model, gives the modal damping rates, while the imaginary part gives the damped natural frequencies.

As specified above this kind of realization is internally balanced and the singular value decomposition of the Hankel matrix ensures, in some sense, the

exclusion of possible noises in the data and in fact the degree of observability and controllability of the system decreases with the amplitude of the singular values. Juang and Pappa (1986) studied the effect of the noise on modal parameters estimation by ERA, establishing some relationship between the singular values and the characteristic of the noise. The modal amplitude coherence (MAC) and the mode singular value (MSV) are quantities that allow the accuracy of the identified system to be established. The calculation of these quantities are then included in the ERA algorithm, and if these indicators determine a reduction of the order of the model the program returns to the representation (2.2.3.2), re-evaluates the Markov parameters and repeats the procedure. More details on the MAC and MSV factors can be found in appendix (2.A).

The flowchart shown in fig.(2.1) summarizes the primary steps of the system realization algorithm.

The advantages in using this particular realization, over others based on the Hankel matrix decomposition approach, are that it is a minimal order realization in absence of noise and for a defined accuracy, as shown above, and because it is a balanced realization.

The Markov parameters of the system can be obtained through discrete Fourier transforms and spectral analysis (see appendix 2.B) or they can be directly derived in the time domain, using ARMA models, with a benefit of providing filter gains of an asymptotically stable observer model, as will be specified in the following sections.

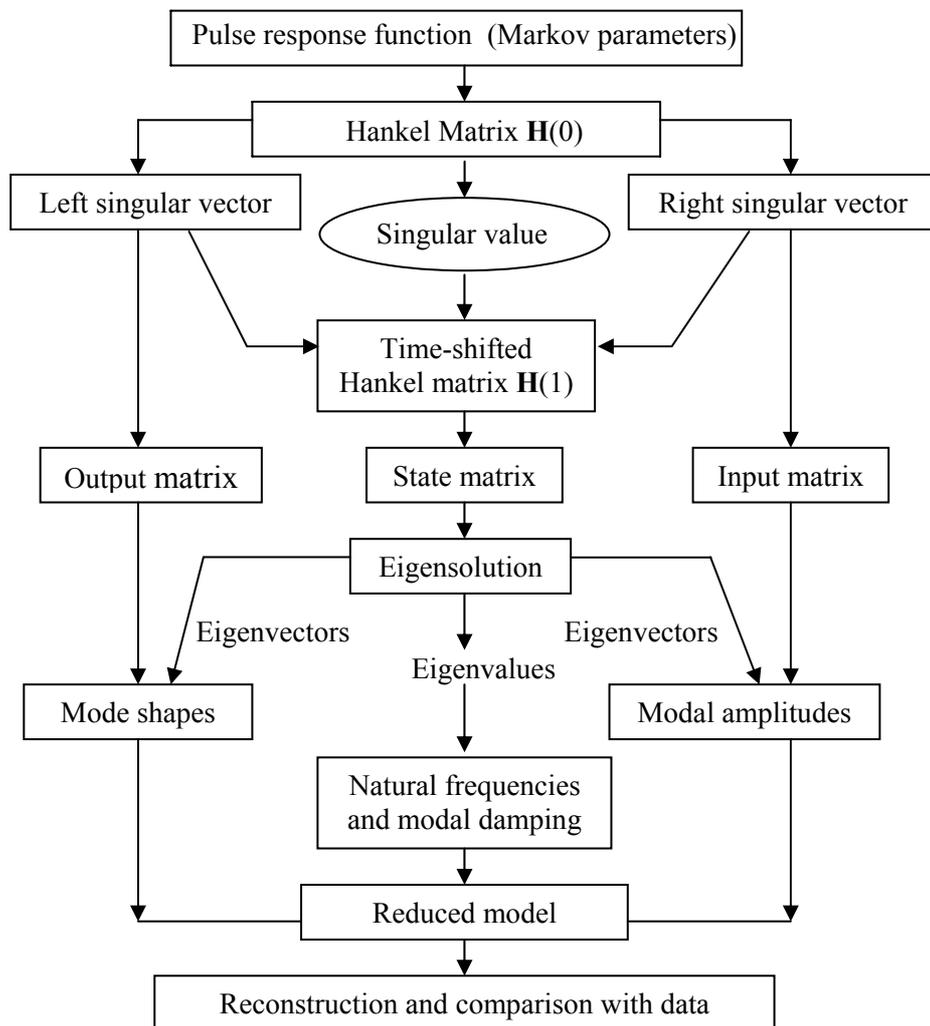


Figure 2.1: Flowchart for the ERA
 After Juang J.N. (1994) 'Applied System Identification'.

2.4 Kalman filter

The discrete time-domain model introduced by eqs.(2.2.2.4) and (2.2.2.5) does not account for the possibility of model inaccuracy and the presence of noise in the measured data. However these quantities may be included in the model which assumes the form:

$$\mathbf{x}(k+1) = \mathbf{A} \mathbf{x}(k) + \mathbf{B} \mathbf{u}(k) + \mathbf{e}_1(k) \quad (2.4.1)$$

$$\mathbf{y}(k) = \mathbf{C} \mathbf{x}(k) + \mathbf{D} \mathbf{u}(k) + \mathbf{e}_2(k) \quad (2.4.2)$$

where the two new vectors $\mathbf{e}_1(k)$ and $\mathbf{e}_2(k)$ represent the processes of noise due to disturbances and modeling inaccuracies and the measurement noise due to the inaccuracy of the sensors. Classically these processes are modeled as zero-mean and Gaussian, therefore they are assumed independent each other.

By minimizing the state estimation error $\mathbf{e}(k) = \mathbf{x}(k) - \hat{\mathbf{x}}(k)$ between the estimate state and the real state, one arrives to define a Kalman filter.

Suppose now that the estimated state can be written in recursive form as:

$$\hat{\mathbf{x}}(k+1) = \mathbf{F}(k) \hat{\mathbf{x}}(k) + \mathbf{H}(k) \mathbf{u}(k) + \mathbf{K}(k) \mathbf{y}(k) \quad (2.4.3)$$

where $\mathbf{F}(k)$, $\mathbf{H}(k)$ and $\mathbf{K}(k)$ are respectively $(n \times n)$, $(n \times r)$ and $(n \times m)$ time variant matrices, to be determined in a way that the estimation error has zero mean value and is minimum.

The estimation error $\mathbf{e}(k+1)$ can be written as:

$$\begin{aligned} \mathbf{e}(k+1) = \mathbf{x}(k+1) - \hat{\mathbf{x}}(k+1) = & [\mathbf{A} - \mathbf{F}(k) - \mathbf{K}(k)\mathbf{C}] \hat{\mathbf{x}}(k) + \\ & + [\mathbf{B} - \mathbf{H}(k) - \mathbf{K}(k)\mathbf{D}] \mathbf{u}(k) + [\mathbf{A} - \mathbf{K}(k)\mathbf{C}] \mathbf{e}(k) ; \end{aligned} \quad (2.4.4)$$

then the expected value of $\mathbf{e}(k+1)$ will be equal to zero only if:

$$\mathbf{F}(k) = \mathbf{A} - \mathbf{K}(k)\mathbf{C}; \quad \mathbf{H}(k) = \mathbf{B} - \mathbf{K}(k)\mathbf{D}; \quad \hat{\mathbf{x}}(0) = \mathbf{E}[\mathbf{x}(0)] = \mathbf{0}. \quad (2.4.5)$$

From the above positions the following Kalman filter derives:

$$\hat{\mathbf{x}}(k+1) = [\mathbf{A} - \mathbf{K}\mathbf{C}] \hat{\mathbf{x}}(k) + [\mathbf{B} - \mathbf{K}\mathbf{D}] \mathbf{u}(k) + \mathbf{K} \mathbf{y}(k) \quad (2.4.6)$$

$$\mathbf{y}(k) = \mathbf{C} \hat{\mathbf{x}}(k) + \mathbf{D} \mathbf{u}(k) + \boldsymbol{\varepsilon}(k). \quad (2.4.7)$$

Define:

$$\bar{\mathbf{A}} = [\mathbf{A} - \mathbf{K}\mathbf{C}], \quad \bar{\mathbf{B}} = [\mathbf{B} - \mathbf{K}\mathbf{D} \quad \mathbf{K}], \quad \mathbf{v}(k) = \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{y}(k) \end{bmatrix} \quad (2.4.8)$$

then eq.(2.4.6) can be replaced by the following:

$$\hat{\mathbf{x}}(k+1) = \bar{\mathbf{A}} \hat{\mathbf{x}}(k) + \bar{\mathbf{B}} \mathbf{v}(k). \quad (2.4.9)$$

Then, indicated with $\boldsymbol{\varepsilon}(k)$ the residual, defined as the difference between the real and the estimated measurement, given by:

$$\boldsymbol{\varepsilon}(k) = \mathbf{y}(k) - \hat{\mathbf{y}}(k) \quad (2.4.10)$$

the Kalman filter can be rewritten in terms of estimated quantities as:

$$\hat{\mathbf{x}}(k+1) = \mathbf{A} \hat{\mathbf{x}}(k) + \mathbf{B} \mathbf{u}(k) + \mathbf{K} \boldsymbol{\varepsilon}(k) \quad (2.4.11)$$

$$\hat{\mathbf{y}}(k) = \mathbf{C} \hat{\mathbf{x}}(k) + \mathbf{D} \mathbf{u}(k). \quad (2.4.12)$$

The matrix \mathbf{K} is named the Kalman filter gain and must be determined to minimize the estimation error. When the system reaches the steady state condition, the Kalman filter gain is a constant ($n \times r$) matrix, having the form:

$$\mathbf{K} = \mathbf{A}\mathbf{P}\mathbf{C}^T [\mathbf{E}_2 + \mathbf{C}\mathbf{P}\mathbf{C}^T]^{-1} \quad (2.4.13)$$

\mathbf{E}_2 being the covariance matrix of the process $\mathbf{e}_2(k)$ and \mathbf{P} the covariance matrix associated with the residual process and fulfilling the Riccati equation:

$$\mathbf{P} = \mathbf{A}\mathbf{P}\mathbf{A}^T - \mathbf{A}\mathbf{P}\mathbf{C}^T [\mathbf{E}_2 + \mathbf{C}\mathbf{P}\mathbf{C}^T]^{-1} \mathbf{C}\mathbf{P}\mathbf{A}^T + \mathbf{E}_1 \quad (2.4.14)$$

\mathbf{E}_1 being the covariance matrix of the error process $\mathbf{e}_1(k)$. For more details the reader can refer to appendix (2.C).

The Kalman filter is essentially a method of sequential least square estimation. In the last decades extensive work has been done on this subject because, at steady state, the Kalman filter presents the attractive quality of having a closed form solution, the Riccati equation, for its gain matrix. However it presents the drawback that the covariance of the input and the output measurements must be known. These quantities are difficult to evaluate.

In the following sections, under appropriate hypothesis, an important relationship between the Kalman filter gain and the matrix gain of an observer identifier will be evidenced, showing that an optimal realization can be obtained for an observer identifier having a Kalman gain matrix.

2.5 The Observer Kalman filter

The procedure described in section (2.3) and implemented in the ERA algorithm is practically useful if the order of the Markov parameters involved in the problem is relatively low. However, for lightly damped systems, the order k for which $\mathbf{A}^k \cong 0$ is really large which makes impossible to derive the Markov parameters from eq.(2.2.3.3); the matrix $\bar{\mathbf{U}}$ becomes ill conditioned. In order to overcome this problem the damping of the system is artificially increased by inserting an observer in the system with the purpose of making the system as stable as desired.

The discrete time state-space observer model of a dynamic system has the form:

$$\mathbf{x}(k+1) = \bar{\mathbf{A}} \mathbf{x}(k) + \bar{\mathbf{B}} \mathbf{v}(k) \quad (2.5.1)$$

$$\mathbf{y}(k) = \mathbf{C} \mathbf{x}(k) + \mathbf{D} \mathbf{u}(k) \quad (2.5.2)$$

where the following definitions have been adopted:

$$\bar{\mathbf{A}} = [\mathbf{A} + \mathbf{G}\mathbf{C}], \quad \bar{\mathbf{B}} = [\mathbf{B} + \mathbf{G}\mathbf{C} \quad -\mathbf{G}], \quad \mathbf{v}(k) = \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{y}(k) \end{bmatrix}. \quad (2.5.3)$$

In the above, \mathbf{G} is an arbitrary $(n \times r)$ matrix, known as gain matrix, which allows the eigenvalues of the state matrix to move, as is evident from the first of eq.(2.5.3), making $\bar{\mathbf{A}}$ asymptotically stable. The observer Markov parameters of the system given in eq.(2.5.1) can be used, as the corresponding system without observer, to describe the input-output map. So, similarly to eq.(2.2.3.3), the following relationship holds:

$$\bar{\mathbf{y}} = \bar{\mathbf{Y}}_0 \mathbf{V} \quad (2.5.4)$$

where:

$$\bar{\mathbf{y}} = [\mathbf{y}(0) \ \mathbf{y}(1) \ \mathbf{y}(2) \ \dots \ \mathbf{y}(k-1)] \quad \rightarrow \ (r \times k) \quad (2.5.5)$$

$$\bar{\mathbf{Y}}_0 = [\mathbf{D} \ \mathbf{C}\bar{\mathbf{B}} \ \mathbf{C}\bar{\mathbf{A}}\bar{\mathbf{B}} \ \dots \ \mathbf{C}\bar{\mathbf{A}}^{k-2}\bar{\mathbf{B}}] \quad \rightarrow \ r \times [(r+m)(k-1) + m] \quad (2.5.6)$$

$$\mathbf{V} = \begin{bmatrix} u(0) & u(1) & u(2) & \dots & u(k-1) \\ & v(0) & v(1) & \dots & v(k-2) \\ & & v(0) & \dots & v(k-3) \\ & & & & v(0) \end{bmatrix} \rightarrow [(r+m)(k-1) + r] \times k \quad (2.5.7)$$

The observer Markov parameters are then obtained as:

$$\bar{\mathbf{Y}}_0 = \bar{\mathbf{y}} \mathbf{V}^T [\mathbf{V} \mathbf{V}^T]^{-1} \quad (2.5.8)$$

if $[\mathbf{V} \mathbf{V}^T]^{-1}$ exists, otherwise $\mathbf{V}^T [\mathbf{V} \mathbf{V}^T]^{-1}$ is replaced by the a pseudo-inverse matrix \mathbf{V}^+ defined as $\mathbf{V} \mathbf{V}^+ \mathbf{V} = \mathbf{V}$.

Methods to define \mathbf{G} correctly are discussed in Phan et al.(1992), where it is also outlined how the discrete time observer model given by eqs.(2.5.1) and (2.5.2) is formally identical to a steady state Kalman filter when $\mathbf{G} = -\mathbf{K}$ and $\boldsymbol{\varepsilon}(k) = 0$. This implies that also their Markov parameters are equal. This is an important concept due to the stability of the Kalman filter.

From the latter, it can be concluded that any observer satisfying eq.(2.5.8) produces the same input-output map of a Kalman filter if the data length is sufficiently long and the order of the observer sufficiently large so that the truncation error is negligible. Therefore, when reduced to the system order, the identified observer has to be a Kalman filter, that implies $\mathbf{G} = -\mathbf{K}$.

The concepts expressed in this section have been used in the so called Observer/Kalman filter and implemented in a program Observer Kalman Filter Identifier (OKID). Once a reduced model for the system has been identified, using the observer equations, the eigensystem realization theory, as described in section (2.3), is applied to find a realization of a state-space model and the corresponding eigensolution.

2.6 A Practical Application

2.6.1 Preliminary considerations

System identification techniques can be repetitively applied for damage assessment and structural control (Faravelli et al. 1996). In the following numerical application the eigen-characteristics of a typical space structure, modeled through a finite element technique, are identified before and after damage using the eigensystem realization theory.

The action of linear piezoelectric actuators is also modeled with the double purpose of allowing a correct identification of the structure, during standard operational requirements, and controlling the structural behavior in a damaged state, in order to partially maintain its overall functionality until a restoring intervention is achievable.

The problem is addressed in two steps: first the system identification technique is used for damage assessment and then a suitable control law is applied in order to achieve a dynamic response of the damaged structure slightly different from that of the undamaged structure. For the purpose of the present application, a truss structure properly equipped with sensors (for the output monitoring) and actuators (for force inputs) is considered.

The problem can be stated as follows:

1. choose the locations of actuators to excite the structural modes of interest and place the sensors to measure these modes. An external random excitation is also considered;
2. identify a state-space able to adequately describe the input-output properties of the intact structure;
3. simulate a severe damage in the structure by removing one or more rods;
4. identify a state-space model for the damaged system;
5. control the system by means of an output feedback strategy.

The identification procedure is not intended to locate the damage in the truss, but to ascertain that the damage has occurred and to identify the system dynamics in the damaged case.

2.6.2 The implemented procedure

In the present study, the intact structure has been identified through OKID with a state-space model in the following form:

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u} \quad (2.6.2.1)$$

$$\mathbf{y} = \mathbf{C} \mathbf{x} + \mathbf{D} \mathbf{u} \quad (2.6.2.2)$$

where $\mathbf{x} \in \mathbb{R}^n$ denotes the state of the system, $\mathbf{u} \in \mathbb{R}^m$ indicates the input (or control) variable and $\mathbf{y} \in \mathbb{R}^r$ is the output variable. All \mathbf{x} , \mathbf{u} , \mathbf{y} are functions of time t . Moreover, without loss of generality, it has been assumed that $m < r < n$.

When the input are forces the outputs are displacements (as in this case), the matrix \mathbf{D} is zero and eq.(2.6.2.2) assumes the form:

$$\mathbf{y} = \mathbf{C} \mathbf{x} . \quad (2.6.2.3)$$

It is important to note that the physical meaning of the state is not known, because the OKID algorithm only chooses the set of matrices \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} which best correlate the input with the output data. This difficulty can be avoided as shown in the following. Two cases are possible. First, suppose that the order of the system is equal to the number of outputs. In this case the matrix \mathbf{C} is square and can be inverted. Introduce the following state transformation:

$$\mathbf{z} = \mathbf{C} \mathbf{x} \quad (2.6.2.4)$$

to obtain:

$$\dot{\mathbf{z}} = \tilde{\mathbf{A}} \mathbf{z} + \tilde{\mathbf{B}} \mathbf{u} \quad (2.6.2.5)$$

$$\mathbf{y} = \mathbf{z} \quad (2.6.2.6)$$

where:

$$\tilde{\mathbf{A}} = \mathbf{C} \mathbf{A} \mathbf{C}^{-1} \quad (2.6.2.7)$$

$$\tilde{\mathbf{B}} = \mathbf{C} \mathbf{B}. \quad (2.6.2.8)$$

Due to this transformation, the state variables have now a clear physical meaning, since the system outputs coincide with the displacements at the nodes of the structure.

However, when the order of the system is larger than the one of measured outputs ($n > r$) \mathbf{C} becomes rectangular and therefore is not invertible.

To proceed as above, it is first necessary a partition of the state vector as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_r \\ \mathbf{x}_l \end{bmatrix} \quad (2.6.2.9)$$

with $\mathbf{x}_r \in \mathbb{R}^r$, $\mathbf{x}_l \in \mathbb{R}^l$ and $l = n - r$ in a way that :

$$\mathbf{y} = [\mathbf{C}_1 \quad \mathbf{C}_2] \begin{bmatrix} \mathbf{x}_r \\ \mathbf{x}_l \end{bmatrix}. \quad (2.6.2.10)$$

Next, it is introduced an augmented output vector as:

$$\mathbf{y}_a = \begin{bmatrix} \mathbf{y} \\ \mathbf{x}_l \end{bmatrix} = \mathbf{C}_a \begin{bmatrix} \mathbf{x}_r \\ \mathbf{x}_l \end{bmatrix} \quad (2.6.2.11)$$

where:

$$\mathbf{C}_a = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \\ \mathbf{0} & \mathbf{I}_l \end{bmatrix}. \quad (2.6.2.12)$$

After this artificial extension of the outputs, the state-space model can be transformed as follows:

$$\mathbf{z} = \mathbf{C}_a \mathbf{x} \quad (2.6.2.13)$$

which now is of the form of eq.(2.6.2.3).

Following the above procedure the system representation becomes:

$$\dot{\mathbf{z}} = \tilde{\mathbf{A}} \mathbf{z} + \tilde{\mathbf{B}} \mathbf{u} \quad (2.6.2.14)$$

$$\mathbf{y} = \tilde{\mathbf{C}} \mathbf{z} \quad (2.6.2.15)$$

being:

$$\tilde{\mathbf{A}} = \mathbf{C}_a \mathbf{A} \mathbf{C}_a^{-1} \quad (2.6.2.16)$$

$$\tilde{\mathbf{B}} = \mathbf{C}_a \mathbf{B} \quad (2.6.2.17)$$

$$\tilde{\mathbf{C}} = [\mathbf{I}_r \quad \mathbf{0}]. \quad (2.6.2.18)$$

Note that now the state variable transformation of eq.(2.6.2.13) allows one to give a clear physical meaning to the identified state-space system. And then, by applying the same kind of transformation to the state-space models of the undamaged and of the damaged structure a comparison can be made; the meaning of the state variables being the same. In the present study system identification techniques have been used to detect the state of the structure during the life time. This has been done by applying independent and time varying forces through the actuators, which could be built with piezoelectric elements, and by considering an external disturbance also applied to the

structure. The first identification might be used to assess and refine the quality of the finite element model of the structure. Successive identifications of the system, showing changes in the low frequency modes of the structure, can point out that damage, intended as stiffness reduction, has occurred in the structure. This would lead to the decision of replacement/refurbishment of the structure during its service life.

2.6.3 The control law

Suppose now that at a certain instant the procedure described in the previous paragraph shows that a structural damage is occurred. The problem is how to restore, at least partially, the dynamic behavior of the structure so as to fit it with the corresponding one of the intact truss.

The procedure is as follows. Assume that the vibration of the truss is constrained to keep its low frequency characteristics in the presence of a structural damage. A control strategy can be obtained to guarantee that the main operational specifications are satisfied. To this end, the intact and the damaged structure are first identified and the corresponding state-space model are transformed using eqs.(2.6.2.5)-(2.6.2.18). In general the intact and the damaged structures show a different dynamic behavior due to the different eigen-structure of the two systems. In order to recover the dynamics of the damaged structure, the latter is forced to behave like the intact one by means of a static output feedback controller. In particular, the chosen controller strategy is to match the identifiable eigenvalues and corresponding right eigenvectors of the intact structure to the corresponding ones of the damaged structure.

It is recalled the well known theorem (Szinathkumar et al., 1978):

given a controllable and observable system and the assumptions that the matrices \mathbf{B} and \mathbf{C} are of full rank, then $\max(m, r)$ closed loop eigenvalues can be assigned and $\max(m, r)$ eigenvectors (are reciprocal vectors by duality) can be partially assigned with $\min(m, r)$ entries in each vector arbitrarily chosen using gain output feedback, i.e., with a control law:

$$\mathbf{u} = -\mathbf{S} \mathbf{y} . \quad (2.6.3.1)$$

The gain matrix \mathbf{S} can be evaluated according to the method by Andry et al. (1983). In particular for each eigenvalue λ_i , the achievable eigenvector \mathbf{v}_i^a is given by:

$$\mathbf{v}_i^a = \mathbf{L}_i \mathbf{L}_i^+ \mathbf{v}_i^d \quad (2.6.3.2)$$

where \mathbf{v}_i^d is the desired eigenvector while \mathbf{L}_i is defined as:

$$\mathbf{L}_i = (\lambda_i \mathbf{I} - \tilde{\mathbf{A}})^{-1} \tilde{\mathbf{B}} \quad (2.6.3.3)$$

and $(.)^+$ denotes the appropriate pseudo-inverse of $(.)$. Finally the feedback matrix can be computed by the following relation:

$$\mathbf{S} = \mathbf{U} (\tilde{\mathbf{C}}\mathbf{V})^{-1} \quad (2.6.3.4)$$

where \mathbf{V} is the matrix whose columns are the achievable eigenvectors and \mathbf{U} the matrix whose columns are related to the achievable eigenvalues and eigenvectors (Andry, Shapiro and Chung, 1982).

Note that, according to the Srinatkrumar's result, in general it is not possible to assign completely the right eigenvectors of the controlled system. Instead the above procedure tends to guarantee that the desired eigenvectors are best approximated in a least square sense.

It must be stressed that care must be taken in the evaluation of the pseudo-inverse of \mathbf{L}_i because this matrix is ill conditioned. To avoid numerical difficulties, a singular value decomposition has been used, where the singular values that are smaller than a fixed minimum value have been neglected.

2.6.4 A numerical example

For numerical purposes, a 9-bay truss structure was chosen, as shown in fig.(2.2). The truss structure is composed of 46 graphite/epoxy beams and is clamped at one end, precisely at nodes 1 and 20. The beams have a hull circular cross section with an inner radius of 0.033 m and an outer radius of 0.035 m . Each non-diagonal element is a 1.50 m long truss. The finite element analysis of the dynamical behavior of the structure was obtained by means of ABAQUS 5.4 software.

At the upper right end node of the structure, node 11 in the figure, is applied an external excitation normal to the longitudinal axis of the truss. The excitation is modeled as a Gaussian white noise to simulate the action of an external disturbance.

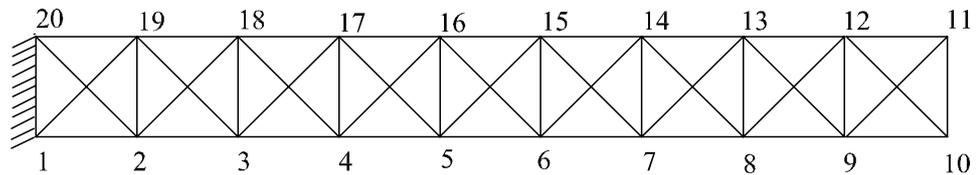


Figure 2.2: Nine-bay truss structure.

Actuators are supposed to be located as beams between nodes 1 and 2, between nodes 5 and 16 and between nodes 17 and 3. The placed of the actuators was based upon simplified controllability considerations (singular value analysis). In the numerical analysis of the model it was assumed that the stiffness of the actuators is equivalent to that of the beam elements of the truss. The displacements in a direction normal to the longitudinal axis of the truss were recorded at nodes 3, 5, 8, 11, 14 and 17 for the purpose of implementing the control strategy; subsequently the displacements at the upper right end node of the truss were recorded to compare the behavior of the intact, damaged and controlled structure.

A state-space model of the intact truss was obtained with the previously described identification techniques. The order of the identified state-space model of the structure is 14. The actual output at the sensors' locations ,i.e., the response of the structure as obtained through finite element analysis, is practically coincident with the response of the identified model, as can be seen in fig.(2.3).

The values of the first three lower modes of the intact structure as obtained by the finite element analysis and by OKID are compared on table (2.1). Hence the high accuracy of the identification is confirmed. The presence of structural damage has been simulated for several different cases by completely removing one or two beams from the structure. In the following the results from four different possible damaged conditions of the structure are reported. Precisely, in the first case is analyzed the behavior of the structure when the beams between node 17 and 18 and between 19 and 20 are removed. In a second case the beams between node 2 and 19 and between node 3 and 18 are removed. The third and the fourth cases investigate respectively the situations in which just one beam is assumed damaged and precisely the one between node 1 and 19 and the one between node 4 and 16 respectively.

In appendix (2.D) the matrices **A**, **B**, **C** obtained by OKID in the undamaged condition are reported together with the corresponding matrices, obtained considering the damaged condition determined in case 1. The control matrix **S** is also reported.

Figures (2.4-2.11) report the response, in terms of displacement, of the damaged structure (darker line), together with the corresponding controlled response (lighter line) in (a) and with the response of the intact structure (lighter line) in (b), under the same external excitations.

This comparison is repeated for all the four above mentioned cases and considering the effects on two nodes of the truss. The first positioned at the free extreme of the truss, (node 11) and the second positioned in the middle of the truss (node 5). The results reported in the figures prove the effectiveness of the adopted strategy. In fact, for these cases and for others, following the control strategy outlined in the previous paragraph, the application of an output

feedback control law was simulated in such a way that the first three complex eigenvalues, corresponding to structural modes below 100 Hz, and the components of the right eigenvectors of the undamaged structure can be assigned to the damaged structure to restore the dynamic behavior of the truss.

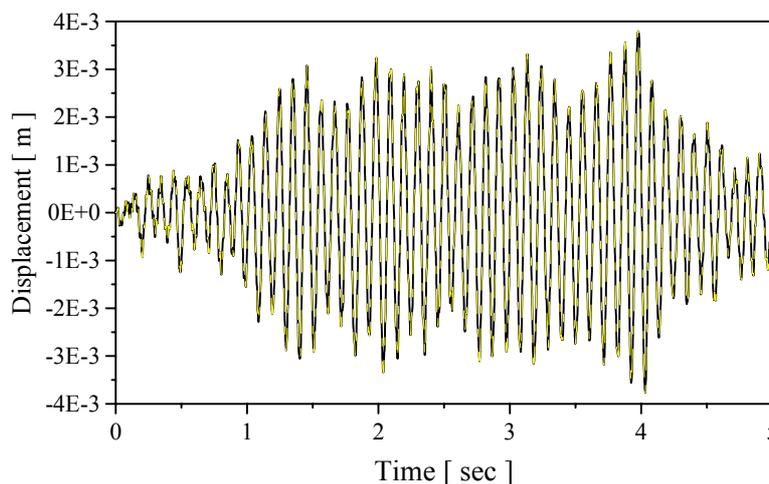


Figure 2.3: — Response of the structure using the FEM model;
- - Response of the structure using OKID

The proposed approach gave successful results for all of the cases presented.

For the identification of the truss the forces applied at node 11 have been modeled as white noises.

The action of the controller restores the dynamic behavior of the truss. The displacement time history of the controlled structure is very similar to that of the intact structure, while the displacement history of the damaged structure is quite different from it except for the response at node 5 in the third case, fig. (2.9.a).

However it should be reminded that the main goal of the control strategy adopted was to restore the three lower modes of the truss, rather than to exactly maintain a certain displacement history at some fixed point of the truss.

Mode	FEM	OKID
I	9.5	9.5
II	53.2	53.7
III	81.9	83.8

Table 2.1: Lower modes of the truss [Hz].

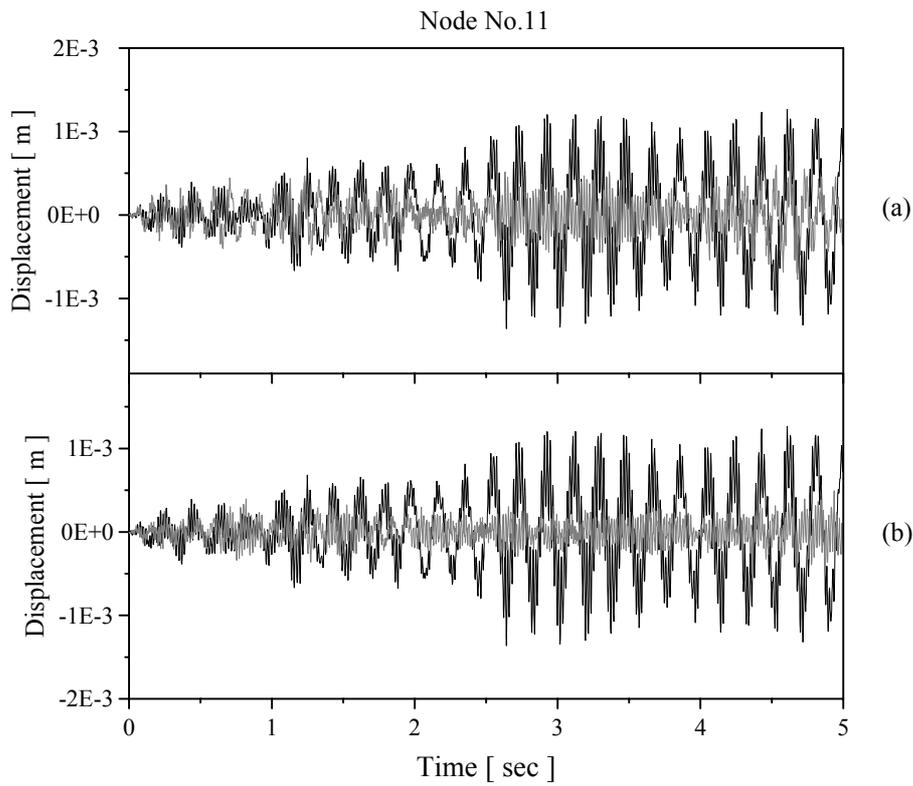


Figure 2.4: (case 1)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
- (b) Response of the damaged (dark line) and of the intact (light line) structure.

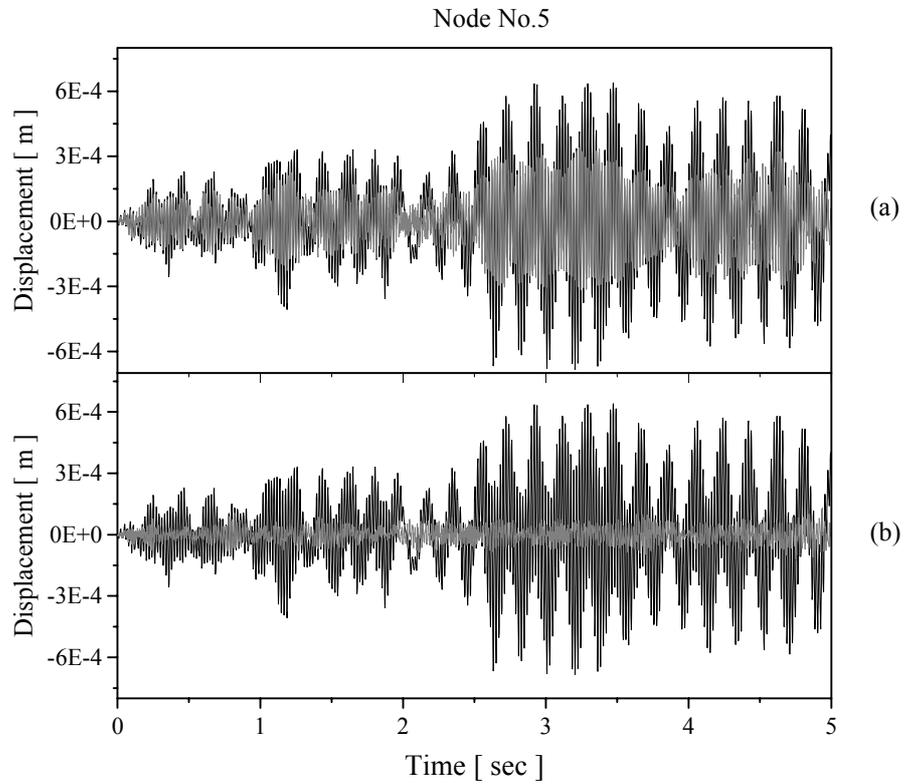


Figure 2.5: (case 1)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
(b) Response of the damaged (dark line) and of the intact (light line) structure.

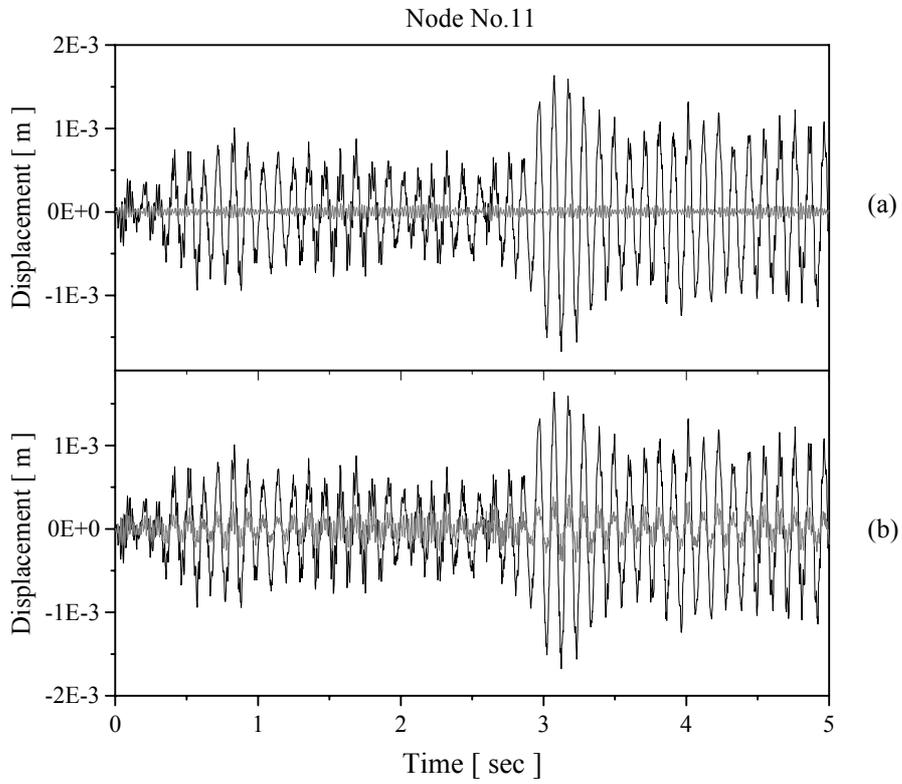


Figure 2.6: (case 2)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
- (c) Response of the damaged (dark line) and of the intact (light line) structure.

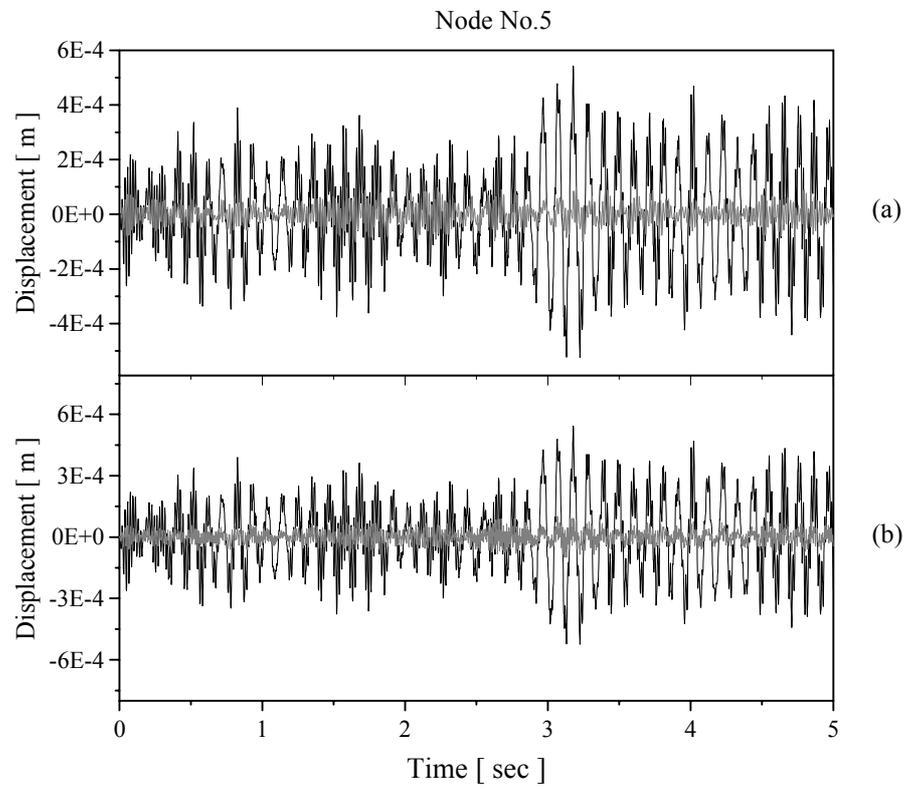


Figure 2.7: (case 2)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
- (b) Response of the damaged (dark line) and of the intact (light line) structure.

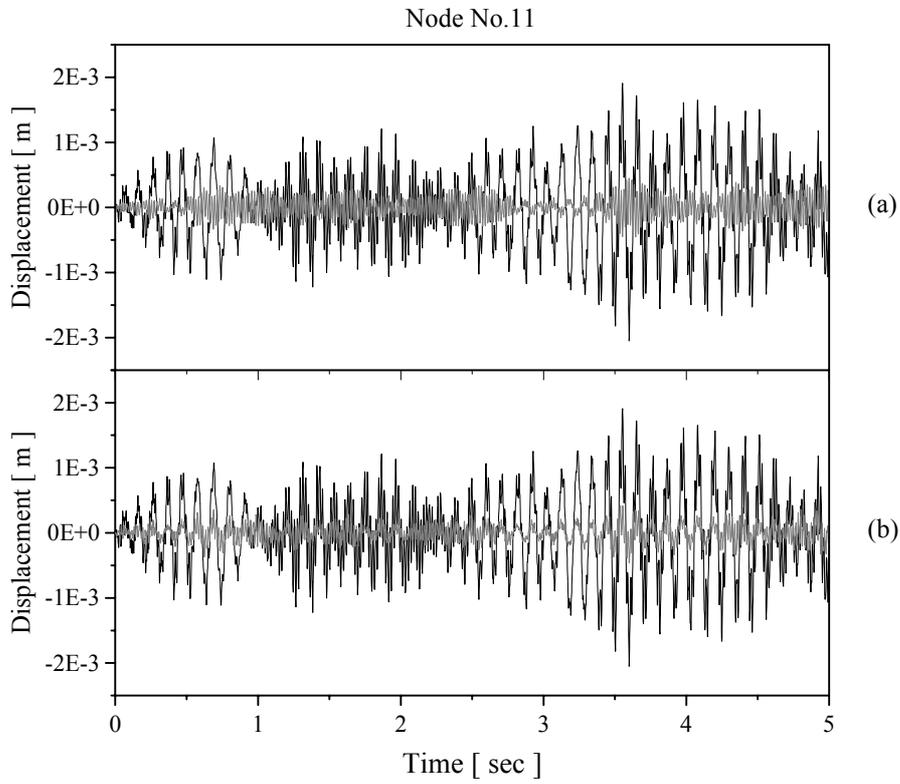


Figure 2.8 (case 3)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
(b) Response of the damaged (dark line) and of the intact (light line) structure.

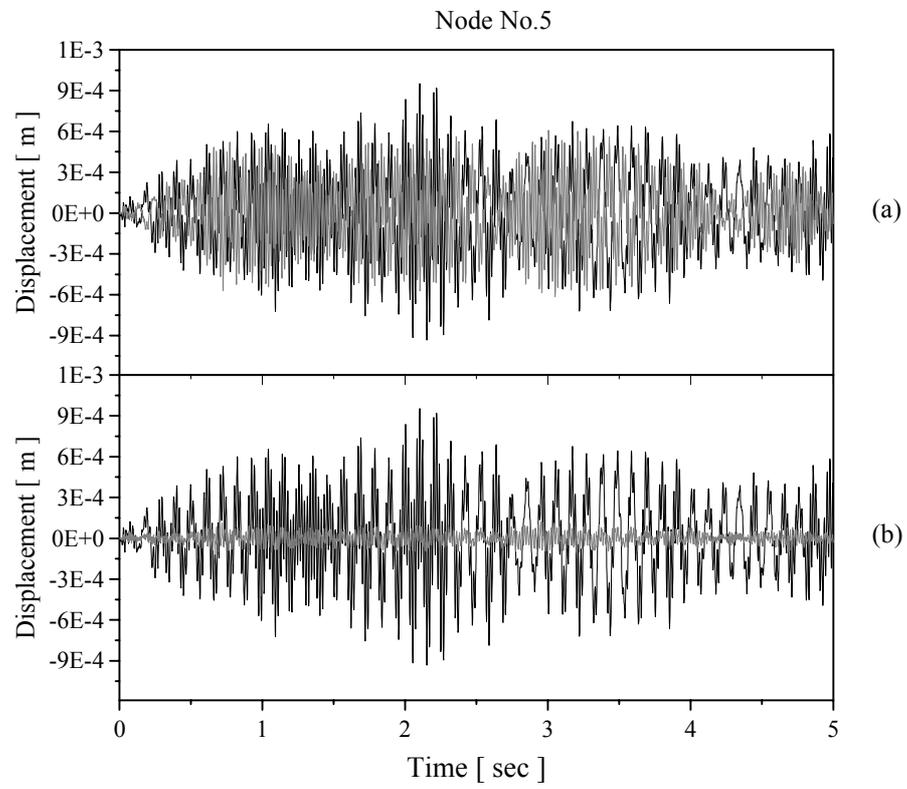


Figure 2.9: (case 3)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
(b) Response of the damaged (dark line) and of the intact (light line) structure.

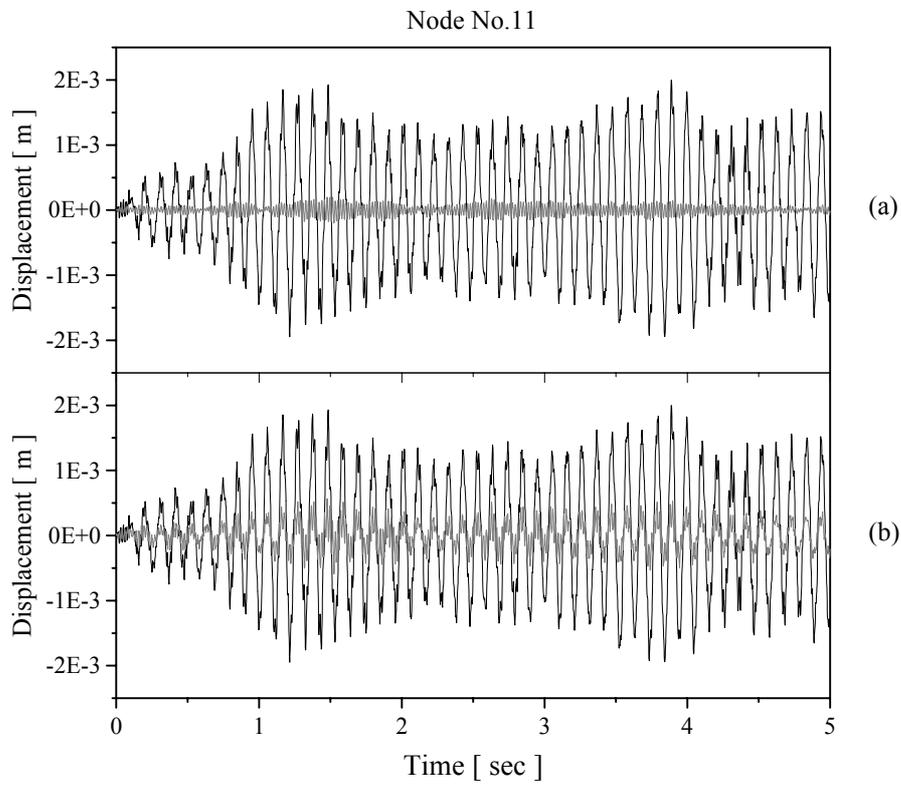


Figure 2.10: (case 4)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
(b) Response of the damaged (dark line) and of the intact (light line) structure.

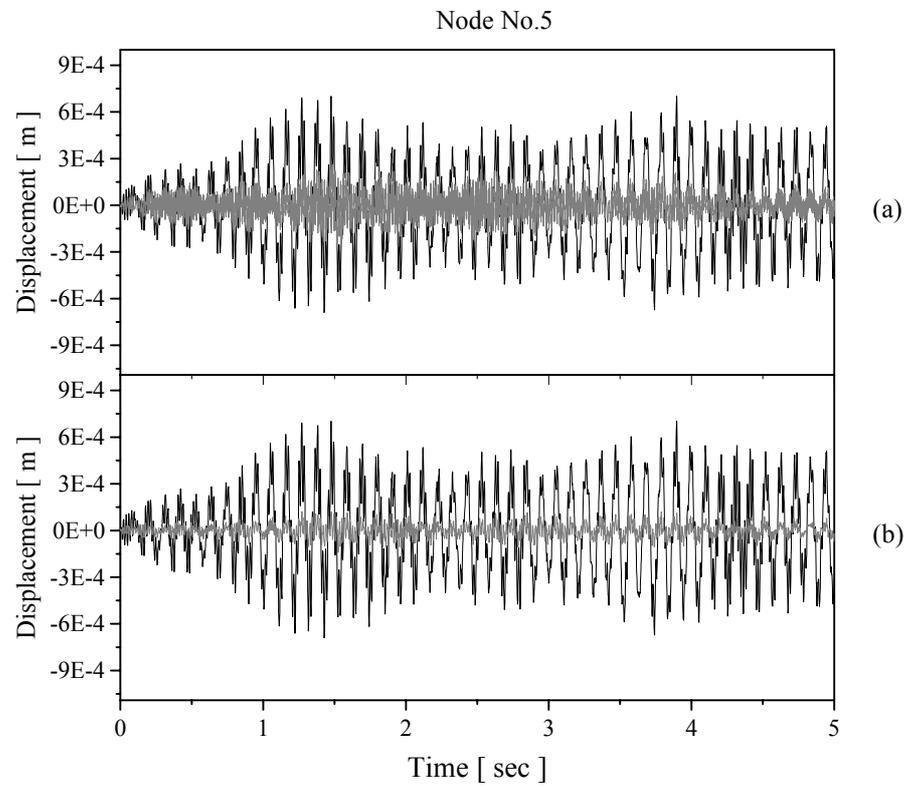


Figure 2.11: (case 4)

- (a) Response of the damaged (dark line) and of the controlled (light line) structure.
(b) Response of the damaged (dark line) and of the intact (light line) structure.

2.7 Conclusion

In this chapter some fundamental concepts regarding the system realization theory have been reviewed. The theory has been implemented in order to have a state variables vector with a clear physical meaning and used, together with a pole assigned law, for damage detection and control in a truss structure. Moreover the observe-Kalman filter identification technique was used to simulate the behavior of a truss structure for state monitoring and damage recovery. The action of linear actuators was model to excite the truss, in order to identify a state-space model of the structure under a known force input, and to restore the dynamic behavior of the structure once the damage has occurred.

An output feedback control law has been determined through an eigen-structure assignment algorithm in such a way that the control strategy imposes the lower frequency modes of the intact structure to be recovered, thus partially restoring the dynamic behavior of the damaged truss. The study outlines the highly effective results that can be obtained through a combined use of system identification / eigen-structure assignment technique.

The method described is simple to handle and the effectiveness of the procedure is clear from the reported results. It should be noted that for the simple truss examined the assignment of the first three modes was sufficient. However, care must be taken when trying to extrapolate the results to other configurations. In more complicated structures and for specific combinations of loads, problem such as spillover, here not present, may appear and make the problem more complicated.

Chapter 3

The Neural Network Approach for Damage Assessment and Safety Analysis

3.1 Introduction

Artificial neural networks have been studied for more than three decades and are rapidly expanding areas of research across many disciplines. They provide a useful tool for their ability to learn from experience, extracting from previous examples conclusions for the new ones and to select from available data the essential characteristic, neglecting irrelevant data.

Neural networks are used for classification (Faravelli, 1994), pattern matching (Chen, 1991), pattern completion, noise removal, linear and non-linear optimization (Tank and Hopfield, 1986), control (Casciati and Faravelli, 1995, Zak et al., 1990), reliability problem (Casciati, Faravelli and Pisano, 1997) and many other problems.

The structure of artificial neural networks can be considered, in a rough manner, similar to that of the brain system. They consist of a number of processing elements (PE), neurons, each of which can have many inputs but only one output. A non simple connection between different neurons is possible through a connection law. Connections serve the specific purpose to determine

the direction of information flow. The weight associated with each connection is an adjustable value, which modulates the amount of output signal passing from one processing element to the adjacent one and changes through a learning process. If the sum of the inputs to a given neuron exceeds a certain threshold, determined by a differential equation, the neuron “fires” and sends a signal to another neuron.

The neural network models can be classified according to various criteria such as their learning methods (supervised versus unsupervised), and architectures (feedforward versus recurrent, output types, node types, connections weight, and so on).

The architecture of the network is determined on how the internal neuron connection are arranged and by the nature of the connection. How the weights of the connections are adjusted or trained to achieve a desired behavior of the network is governed by its learning algorithm. There are networks that can generate dynamically their architecture during the learning process (Adeli and Hung, 1995), but, in general, every network is set up to solve a specific problem and the network architecture is not adaptive.

In a feedforward network the neurons are generally grouped into layers, the signal flow from the input layer to the output layer via unidirectional connections. The number of hidden layers and the number of neurons in each layer depend on the problem. Usually error back-propagation algorithm is used for supervised learning of multi-layer feedforward networks, (Rumelhart, Hinton and Williams, 1986). The method consists in a generalization of the least-squared scheme. Using this algorithm the error signal between the desired and the actual system output is propagated back through the network to adjust the weights of the processing elements in the various layers. This is just one of the several algorithms following the development of the neural network theory. More details are given in next sections and in appendix (3.A).

A very extensive and recent literature can be found on the subject of neural networks also for the specific applications in the field on mechanics and structural engineering. Review papers, Simpson (1992) and Zeng (1998)

provide a synthesis of the developments and applications of neural networks as well as an important font of references.

Recently neural networks are used for diagnostic problems. An earlier work is due to Wu, Gabussi and Garrett (1992) who used a feedforward neural network to localize damage in a shear type frame. The damage was simulated as local stiffness reduction. To train the network the Fourier spectrum of the system response was used. Povich and Lin (1994), following a similar approach, tried to identify the damage in the structure by means of a feedforward neural network trained by the responses in terms of acceleration of the structure subjected to dynamic loads.

Another recent work is due to Ceravolo and De Stefano (1995) who proposed the neural network approach for damage localization in a supported beam training the network with the natural frequencies of the structure identified in different damage conditions.

In the following a three layer feedforward neural network will be used for damage detection in a structural system. As proposed by Faravelli and Pisano (1997) the network will be trained using the transfer functions of the structure in different damage conditions.

In the following sections a brief review of fundamental concepts regarding the choice of the network architecture and size, the learning rule and the preset parameters, as well as some fundamental definitions, are discussed. The use of a neural network for damage detection and safety problems will be also presented. In particular it will be shown how the adopted neural network is able to work either for linear or non-linear structures, either for simple or complex structures.

3.2 The Network Architecture

The architecture of a neural network is defined by its layers and connections; one or more layers can be present between the input and output layers. Usually each unit receives input from all the units in the next layer and no communication is allowed between units in the same layer. Although in a

recurrent network this condition can be different. A multi-layer neural network presents a strong map ability also for non-linear problems. A typical example of such neural network architecture is shown in fig.(3.1), where the directions of the activation propagation and of the error propagation are also evidenced.

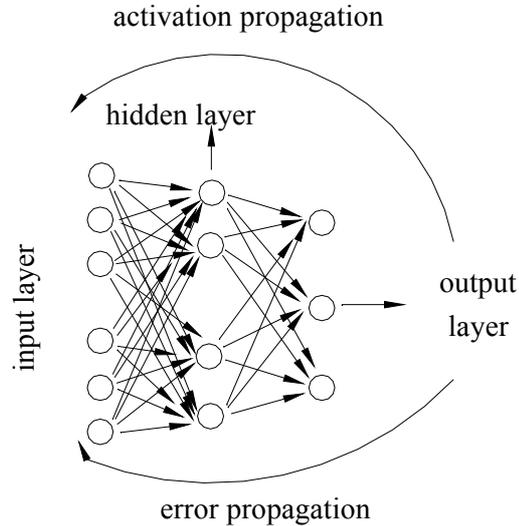


Figure 3.1: Typical scheme of Multilayer Neural Network.

The behavior of a feedforward neural network depends also on the number of layers and nodes in its architecture. So a natural question arising during the construction of a multi-layered neural network is, how many hidden layers and how many neurons in each layer have to be used to achieve the desired mapping.

Based on the Kolmogorov theorem which states that:

any continuous functions $f(x_1, \dots, x_n)$ defined in \mathbf{I}^n ($n \geq 2$) can be represented in the form:

$$f(\mathbf{x}) = \sum_{j=1}^{2n+1} \chi_j \left(\sum_{i=1}^n \psi_{ij}(x_i) \right)$$

where χ_j are continuous functions of one variable and ψ_{ij} are monotone functions not dependent of f and \mathbf{I} the unit closed interval $[0,1]$; Hecht-Nielsen (1987) proved the following Kolmogorov's mapping neural network existence theorem :

any continuous function of $\varphi : \mathbf{I}_n$ to \mathbf{R}_m , $\mathbf{Y} = \varphi(\mathbf{X})$, can be implemented exactly by a multilayer neural network, which have n neurons in the input layer (the input vector \mathbf{X}), $2n+1$ neurons in the hidden layer and m neurons in the output layer (the output vector \mathbf{Y}).

The theorem of Hecht-Nielsen theoretically proves that any continuous function can be reproduced by a three-layer neural network.

With reference to the Rumelhart-Hinton-Williams multilayer network Funahashi (1989) proved that:

any continuous mapping $f : \mathbf{K} \rightarrow \mathbf{R}^m$ defined by $\mathbf{x} = (x_1 \dots x_n) \rightarrow (f_1(\mathbf{x}) \dots f_m(\mathbf{x}))$ can be approximated in the sense of uniform topology on K by input-output mappings of k -layers networks whose output functions for the hidden layers are non-constant, bounded and monotone increasing continuous functions (i.e. sigmoid) and whose output functions for input and output layers are linear.

Where \mathbf{K} is a compact subset of \mathbf{R}^n and k is an integer ≥ 3 .

The starting point of the proof for the one hidden layer case is an integral formula proposed by Irie-Miyake (1988) and from this the proof of the theorem can be generalized to the case of any number of hidden layers by induction. The Funahashi theorem has been extended by Harmik et al.(1989) to the case of non continuous activation function. The proposed results evidenced how standard multilayer feedforward neural networks are capable of approximating any measurable function to any desired degree of accuracy. From this derives that this kind of networks are universal approximators.

However, despite the Hecht-Nielsen and the Funahashi existence theorems, it has been pointed out (Flood et al.,1994) that there are many solution surfaces

that are extremely difficult to model using a network with just one hidden layer. In these cases a two hidden layers network is recommended.

Beside the problem of establishing the number of the hidden layers there is also the one concerning the choice of the number of neurons and of input data to utilize to train the network. It could appear obvious that increasing the number of the neurons provides better results. In practice, can happen that this implies a deviation from the trend of the solution surface. Similar considerations are valid also with regard to the amount of input data presented to the network.

One practical solution is to consider a range of different configurations and try until the best results are obtained. Although, instead of using a so generic approach, one could find more attractive to adopt a specific rule, taking in mind that a general solution to the problem does not exist.

One suggestion (Rogers, 1994) is represented by the following relationship:

$$p = 1 + h(n + m + 1)/m \quad (3.2.1)$$

where n and m have the meaning specified above, while p is the number of training pairs and h is the number of hidden nodes. The equality expressed by eq.(3.2.1) implies the network to be determined; the number of equations equating the number of the unknown weights. Otherwise if p is greater or less than the value from eq.(3.2.1) the network is over or under determined.

In conclusion, it has been remarked that the architecture of the network includes also the kind of connection existing between the neurons in each layer. Several different networks have been developed in the last decade for different types of applications. For the purpose of this work only a multilayer feedforward neural network will be analyzed with major detail, while, among the remaining networks only some notices will given in appendix (3.A).

3.3 Learning Algorithms

In a broad sense the definition *learning* for a neural network means to change the weight connection values between different neurons until the optimal performance of the network is reached. This essentially means to minimize the output error between the network output and the target output for a specific set of input patterns.

Several procedures are available for changing the values of the connection weights; the back-propagation learning rule, in its different versions, the Hebbian learning rules and its derivatives, the principal component learning, the competitive learning, the stochastic learning etc. are such of examples.

In the following particular attention will be reserved to the back-propagation learning rule, while a brief review of the most common learning rules is summarily reported in appendix (3.B).

3.3.1 The back-propagation learning rule

The back-propagation learning rule is the one most in use to train non-linear, multi-layered networks to perform function approximation, pattern association and pattern classification. The rule allows to minimize the sum-squared error of the network and this is done by changing the values of the network weights and biases in the direction of the steepest descent path with respect to the error; in this way the network is able to establish the desired mapping between input and output vectors. At the beginning the network uses the input vectors or random vectors to produce a first set of output vectors and then compares these with the target output.

Let us consider first the case with no hidden layers. If differences are detected, then the network starts to adjust the weights of the connections by means of the following rule:

$$\Delta_p w_{ji} = lr \delta_{pj} i_{pi} \quad (3.3.1.1)$$

where lr indicates the so-called learning rate and is a constant value, δ_{pj} represents the difference between the target input for j -th component of the output pattern for pattern p , say t_j , and the j -th element of the actual output pattern produced by the presentation of the input pattern p , say o_j , i_i is the i -th element of the input pattern.

To derive eq.(3.3.1.1) the approach followed by Rumhart et al.(1986), will be adopted. For linear units, it mainly consists in showing that eq.(3.3.1.1) minimizes the total error of the network. To this end it is shown that the derivative of the error measure with respect to each weight is proportional to the weight change given by the delta rule (with negative constant of proportionality). This corresponds to performing steepest descent path on a surface in weight space whose height, at any point, is equal to the error measure.

The error measure between the expected and the actual outputs on input-output pattern p is defined as:

$$E_p = \frac{1}{2} \sum_j (t_{pj} - o_{pj})^2 \quad (3.3.1.2)$$

Let $E = \sum E_p$ be the total measured error. The following relation holds true:

$$\frac{\partial E}{\partial w_{ji}} = \sum_p \frac{\partial E_p}{\partial w_{ji}} \quad (3.3.1.3)$$

It must be shown that the following equality fulfilled:

$$-\frac{\partial E_p}{\partial w_{ji}} = \delta_{pj} \cdot i_{pj} \quad (3.3.1.4)$$

and in fact, being the right hand side of eq.(3.3.1.4) proportional to the weights change, this equality ensures that the change of the weights results in minimizing the error E.

The left hand side of equation (3.3.1.4) can be re-written in the form:

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial w_{ji}} \quad (3.3.1.5)$$

The first part on the right hand side of eq.(3.3.1.5) gives the error changes with the output of the j -th unit and in explicit form has the expression:

$$\frac{\partial E_p}{\partial o_{pj}} = -(t_{pj} - o_{pj}) = -\delta_{pj} \quad (3.3.1.6)$$

while the second part tells how the output changes with the weights.

Because of the linearity hypothesis $o_{pj} = \sum_i w_{ji} i_{pi}$, one gets:

$$\frac{\partial o_{pj}}{\partial w_{ji}} = i_{pi} \quad (3.3.1.7)$$

Substituting eqs.(3.3.1.6) and (3.3.1.7) in eq.(3.3.1.5) the relation (3.3.1.4) is proved.

The back-propagation learning rule can be easily generalized to the cases in which the units are not linear and the hidden layers are present between the input and the output layer.

In this case it is required that the following relationship holds:

$$-\frac{\partial E_p}{\partial w_{ji}} \propto \Delta_p w_{ji} \quad (3.3.1.8)$$

Making the same steps as above one writes:

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial net_{pj}} \frac{\partial net_{pj}}{\partial w_{ji}} \quad (3.3.1.9)$$

with $net_{pj} = \sum_i w_{ji} o_{pi}$ and $o_i = i_i$ if unit i is an input unit and for non linear activation functions:

$$o_{pi} = f_j(net_{pj}) \quad (3.3.1.10)$$

then the second factor of eq.(3.3.1.9) assumes the form:

$$\frac{\partial net_{pj}}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}} \sum_k w_{jk} o_{pk} = o_{pi} \cdot \quad (3.3.1.11)$$

Defining:

$$-\frac{\partial E_p}{\partial net_{pj}} = \delta_{pj} \quad (3.3.1.12)$$

then equ.(3.3.1.9) gives the desired relationship.

To compute δ_{pj} one observes that:

$$\frac{\partial E_p}{\partial net_{pj}} = \frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial net_{pj}} \quad (3.3.1.13)$$

from which using eqs. (3.3.1.10) and (3.3.1.6) one derives:

$$\delta_{p,j} = -\frac{\partial E_p}{\partial net_{p,j}} = (t_{p,j} - o_{p,j})f'_j(net_{p,j}) \quad (3.3.1.14)$$

for any output unit u_j . If u_j is a hidden unit, with a similar procedure, the following expression for $\delta_{p,j}$ can be derived:

$$\delta_{p,j} = f'_j(net_{p,j}) \sum_k \delta_{p,k} w_{kj} \quad (3.3.1.15)$$

Through eqs.(3.3.1.14) and (3.3.1.15) the $\delta_{p,j}$ are obtained for all kind of units in a recursive way. Starting from this point the weights of the network can be derived by means of an equation formally equal to eq.(3.3.1.1).

The same quantities $\delta_{p,j}$ are used to evaluate the changes of the threshold values by means of the following relationship:

$$\Delta_p \theta_j = lr \delta_{p,j} \quad (3.3.1.16)$$

3.3.2 The modified back-propagation algorithm

Pure back-propagation presents some drawbacks especially when, using a non-linear network, the error surface assumes a complicate shape. Problems related to the stability of the algorithm could appear and the solution of the network training becomes trapped in a local minimum of the error surface.

Also the choice of the learning rate size is problematic. The learning rate specifies the amount of changes that are made in the weights and in the biases at each iteration. Small learning rates may guarantee a stable training but they make the learning slow. Rumelhart et al.(1986) state that for the most rapid learning, lr should be as large as possible without leading to oscillation. Even in the unlikely case that one value of lr is optimal at one stage of the learning, there is no guarantee that the same value of lr will be appropriate to any other stage of the learning process.

From the above considerations it comes out the necessity to introduce some improvements to the back-propagation algorithm in order to speed up the training of the network and to ensure its convergence.

Following a work of Volg et al. (1988), the following modifications are considered:

1. Update the network weights after all the patterns to be learned have been presented to the network instead to update these values after each pattern is presented to the network, as implemented in the classical back-propagation algorithm. That is because the reduction of the network error with respect to only one pattern will not, in general, ensure a reduction of the error with respect to all the other patterns which the system is to learn.
2. Make the learning rate lr dynamically changing, so that the algorithm utilizes a near-optimum lr , as determined by the local optimization topography, instead of keeping the learning rate constant during all the training.
3. Introduce a *momentum* factor allowing the network to respond not only to the local gradient but also to the recent trends in the error surface. Acting like a low pass filter, momentum allows the network to neglect small features in the error surface and to escape the problem of local minima.

Introducing these variations into the Rumelhart algorithm, eq.(3.3.1.1) changes into:

$$\Delta w_{ji}(m+1) = lr \sum_p \delta_{pj} o_{pi} + \alpha \Delta w_{ji}(m) \quad (3.3.2.1)$$

where m indicates the iteration number, since the weights are updated only one per iteration through all the patterns, as evident from the summation over p present in the formula.

The learning rate lr is no more a constant value, its variation is determined on whether or not an iteration decreases the total error for all patterns. Practically if the new error exceeds the old one by more than a prefixed ratio, typically 1.04, then the new weights, biases and outputs are discarded; in addition the learning rate is decreasing, typically using a factor 0.7. On the other hand, if the new error is less of the old one, the learning rate is increasing, typically by a factor 1.05.

Practically this procedure increases the learning rate, saving time, but only if the network can learn without large error increase; when the learning rate is too high to guarantee a decrease of the error its value is decreased until stable learning resumes.

The same philosophy is adopted to set up the value of the momentum α . Momentum is introduced in eq.(3.3.2.1) to adding to the weight's changes, used the simple back-propagation rule, a quantity proportional to the weights calculated in a previous iteration.

If during the training the network produces a total error greater than the prefixed value, the step is repeated and α is set equal to zero. Then the weights are re-arranged using the gradient rule, changing, if necessary, the learning rate. When a successful step is taken α is reset to its original value, typically 0.95. Following this strategy as long as the topography of the error surface is relatively uniform and smooth the memory, implicit in α , will aid convergence. If, however, a step results in a degradation of the performance of the system, then clearly a change in the direction of the optimization is required and the implicit knowledge of α could be misleading.

In the next section a feedforward neural network, trained using the back-propagation algorithm, modified with adaptive learning rate and momentum, is implemented for damage detection problems.

3.4 The proposed approach to the damage detection problem

In the present section the validity of the neural network approach for structural damage identification will be carry out. The proposed procedure will be tested through different numerical examples including the case in which the non-linear behavior of the structure is considered.

The detection of damage in complex structures presents, in general, serious difficulties related to the low sensitivity of the system parameters to the presence of structural defects. The problem becomes even more complicate when one looks for the non-linear behavior of the structure.

The monitoring of the structure is generally realized by comparing some characteristics quantities of the system at various stages of its life. As it is well known, a unique solution of the system identification “inverse problem” does not in general exist. In most cases, it is rather difficult to identify, for an assigned structure, one or more functionals depending on the severity and on the location of the damage, due to the fact that the dynamic behavior of the structure can change even significantly when damage is present.

The idea is to have a tool able to monitor the structure starting from the elaboration of easily acquirable data, i.e, the response of the structure, in terms of acceleration, measured in suitable points. In the following, damage is detecting by using of the transfer functions of the structure as characteristic functions and a feedforward neural network is trained to recognize the actual state among the undamaged and different possible damaged conditions in the structure.

3.4.1 The damage characteristic function

As already outlined in the general introduction, one of the major difficulties in the damage detection problem consists in the identification of a characteristic function that can be determined easily and shows good sensitivity to structural

modifications. For structures under dynamic loads the transfer functions between a given load and the response of the system can be adopted.

So if a series of experiments or simulations of the structural response to a known input is performed, in different damage conditions, the transfer functions can be easily built by applying the Fourier analysis. Changes of these functions can be related with damage evidence in the structure.

Physical reasoning leads to the following definition of the transfer function. Let $\mathbf{Y}(t)$ be the vector of the measured output and $\mathbf{U}(t)$ the vector of the applied input, then the well known relationship between the two vectors holds:

$$\mathbf{Y}(s) = \mathbf{H}(s) \mathbf{U}(s) \quad (3.4.1.1)$$

where s represents the Laplace variable, $\mathbf{Y}(s)$ and $\mathbf{U}(s)$ the Laplace transform of the corresponding vectors $\mathbf{Y}(t)$ and $\mathbf{U}(t)$ and $\mathbf{H}(s)$ is the transfer matrix of the system. Each column of $\mathbf{H}(s)$ being the transfer function between a specific input signal and a corresponding output.

The transfer function matrix can be expressed as function of the characteristic matrices of the system through the following:

$$\mathbf{H}(s) = \mathbf{C} (s\mathbf{I} - \mathbf{A}^{-1}) \mathbf{B} + \mathbf{D} \quad (3.4.1.2)$$

where the matrices \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} are respectively the state matrix of the system, the input influence matrix and the output influence matrices, as already specified in section (2.2.2), while \mathbf{I} denotes the identity matrix. The representation given by eq.(3.4.1.2) outlines how any structural modification, determining a change of the state matrix, is immediately reflected into the transfer function of the system.

The concept appears even more clear if one looks to the polynomial representation of the transfer function, and in fact:

$$H(s) = \frac{\prod_{i=1}^m (s - z_i)}{\prod_{i=1}^n (s - p_i)} \quad (3.4.1.3)$$

The real and the imaginary parts of the poles p_i give respectively the damping and the frequencies of the system; quantities that change when the mechanical characteristic of the structure are modified.

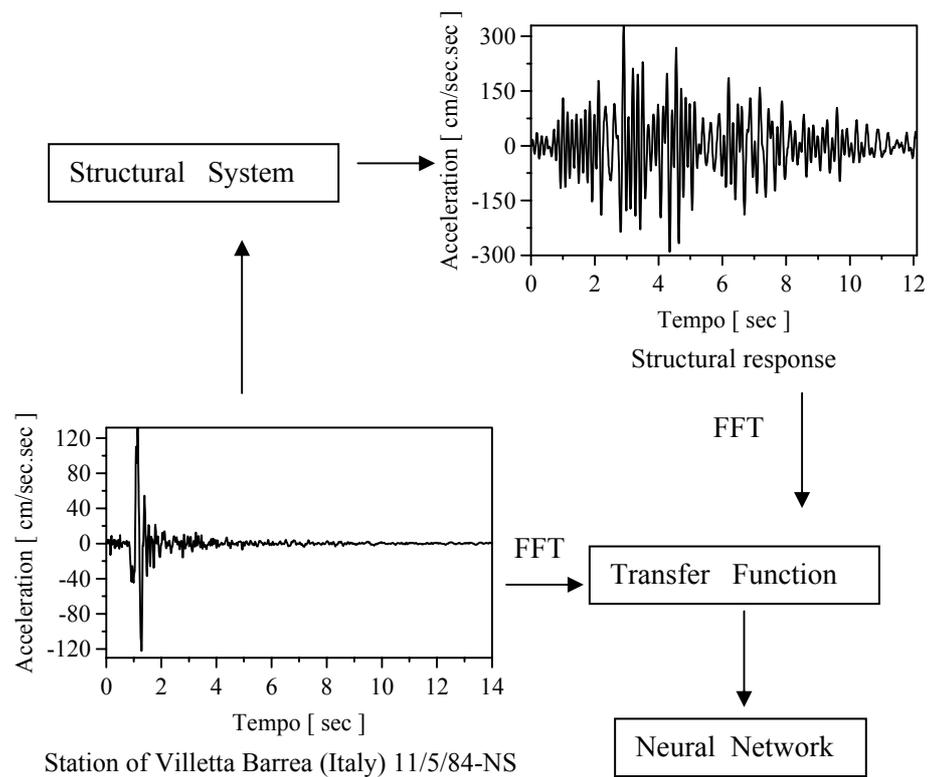


Figure 3.2: The damage detection approach.

The transfer function can be assumed as characteristic of the behavior of the structure even when the hypothesis of linearity is removed. The last of the examples reported in this chapter will make evident how the transfer functions of the structure change significantly when one or more plastic hinges are present, so that different transfer functions can be related to a different number of plastic hinges in the structure.

The differences of the transfer functions are elaborated by a neural network to reproduce the relationship between the transfer function and the condition of damage in the structure.

The flowchart of fig.(3.2) schematizes the proposed approach to the damage detection problem.

3.4.2 Numerical Applications

In the following section three numerical examples will be reported. Each example has the purpose to reveal different potentialities of the neural network tool. The first two examples only deal with the damage detection problem, but analyze structures of different complexity showing how a simple neural network is not able to perform the desired pattern association if has to elaborate an excessive amount of data. In this case, corresponding to the study of a complex structure, arises the necessity to utilize network hierarchically organized. In both the studied cases the behavior of the structure is supposed to be elastic during all the analyses. In the third example the hypothesis of linearity for the structure is removed and the final goal of the study is to show the validity of the neural network approach also in this case and to develop a vulnerability analysis scheme based on real-time monitoring.

3.4.3 Example 1

As first example the four stories frame of fig.(3.3) is considered. In the figure the geometrical and mechanical characteristics, for each structural element, are reported together with the type of profile and the permanent loads.

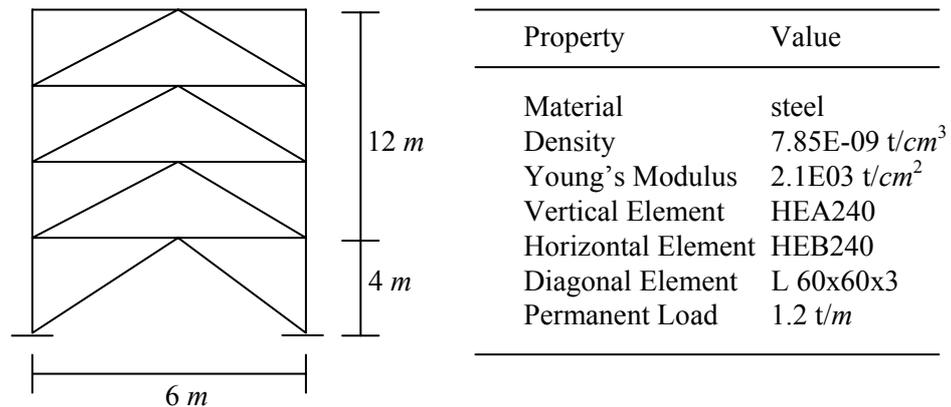


Figure 3.3: Structural model (example 1).

The dimensions of the structure are chosen so that, after a seismic event of given characteristic, the breaking of one of the diagonal elements is admissible.

Using Abaqus 5.5 the dynamic analysis of the structure is carried out. Different accelerograms are applied at the base of the structure while the responses, also in terms of acceleration, are evaluated at the last floor. The analyses are repeated in different conditions, first considering the structure undamaged and then considering each time one diagonal element damaged. That was necessary to create different kinds of structural responses for the network and to teach it to recognize the differences.

For the training stage, the first three accelerograms from the following Italian stations were considered, while the last one was utilized at the end to test the network behavior.

1. Station of Feltre 15/9/76 – Component NS
2. Station of Villetta Barrea 11/5/84 – Component NS
3. Station of Maiano 11/6/76 – Component NS
4. Station of Cassina Sant'Elia 7/5/84 – Component NS.

The records were scaled to have the same peak acceleration of 280 cm/sec^2 .

Some preliminary remarks on the input accelerations are required. The seismic acceleration records were selected from the databank SMCAT.

As a preliminary step all the data were processed to correct the instrument errors introduced in the data during their record. The program PROX, (Casciati et al. 1994) which produces a filtering of the seismic registration in the frequency domain was used for this purpose.

Once the inputs are ready and the responses of the structure calculated, the transfer functions of the structure can be easily built, as specified in section (3.4.1). Each transfer function is associated to each possible considered damage condition. In the present study five possible conditions are hypothesized for the structure: no damage, damage in the first, second, third and fourth floor. Therefore there are five possible state conditions for each of the three inputs, so that the network is activated with an input consisting of 15 vectors each of one contains 125 points of the corresponding transfer function. Figures (3.4 a-d) show the differences between the transfer function obtained when the structure does not present damage with the transfer function obtained when one of the four damaged conditions presented above appears.

The three layer feedforward neural network utilized to solve the damage detection problem was chosen after several attempts necessary to define an optimal network configuration. The two hidden layers contain 30 and 15 neurons respectively and process the data using a log-sigmoidal activation function. Although in the output layer a linear activation function has proved better results.

The target output of the network is represented by a five elements vector contained values from 0 to 4 corresponding to the five possible state conditions of the structure. Precisely 0 corresponds to the undamaged condition, 1, 2, 3 and 4 correspond to the cases with damage in the first, second, third and fourth floor.

The network problem is now to realize the association between the transfer functions in input and the target output. Beyond the definition of the neurons number in each layer, a preliminary study was carried out in order to define the optimal value for the learning rate.

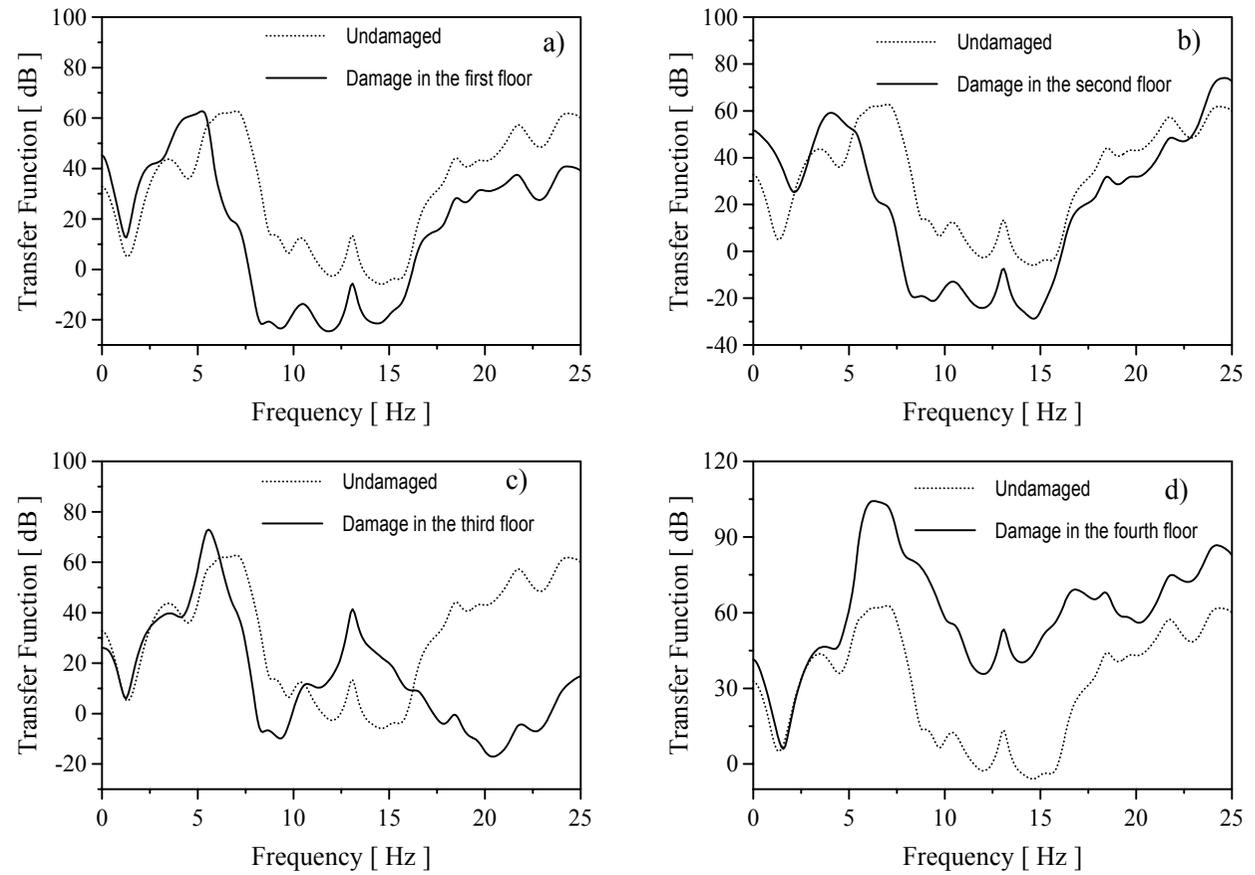


Figure 3.4: Comparison between the transfer functions of the intact and the damaged structure.

Maintaining invariant all the other characteristics of the network, this was first trained for different values of lr . The network was set up to stop its training if one of the following conditions holds:

- 1) the number of iterations exceeds 20000;
- 2) the desired lower value for the total error of the network is reached;
- 3) an upper value for the total error of the network is exceeded.

The last condition means instability of the algorithm.

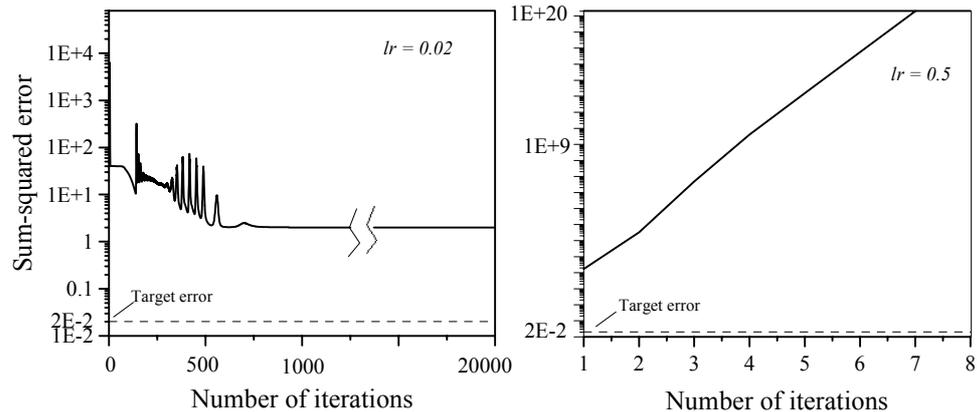
First it was considered the simple back-propagation method. The lower threshold value for the network error was fixed in 0.02, while the upper value in $1E+20$. The learning rate was changed in a range from 0.05 to 0.005 as reported in table (3.1). In the same table the number of iterations, the training time and the final value of the total error of the network are also reported. From these values derives that only with lr equal to 0.006 the network reaches the final goal. Figures (3.5) and (3.6) give the trend of the total error of the network versus number of iterations. They show two typical situations that can be verified when the simple back-propagation algorithm is utilized. Figure (3.5) outlines how the network can fall in a local minimum of the error surface, while fig.(3.6) evidences how a too large learning rate can create instability and in fact the total error increases rapidly with each iteration.

The same set of tests was repeated using the previous network trained by the modified back-propagation learning rule, as introduced in section (3.3.2). The results reported in table (3.2) reveals the sensible improvement of the network efficiency. The admissible network error was fixed in 0.0002 and it was reached very quickly in all the examined cases. Figures (3.7 a,b) and (3.8 a,b) show the trend of the total error of the network and of the corresponding learning rate starting from an initial value for lr of 0.05 and 0.005 respectively.

It should be pointed out that also the modified back propagation, for given initial condition, can have the problem to fall in a local minimum during the training. When the desired behavior for the network has been reached its weights and biases are saved. At this point the network represents a model of the system able to reproduce its behavior. The network can then be tested using the fourth accelerogram of the above list. Table (3.3) reports the output of the

network trained with BP for different values of lr . The target values are also reported (in bold). Likewise table (3.4) reports the output of the network trained with MBP.

As appears evident from these numbers the results coming out from the example are satisfactory and the network is always able to identify the damage and its position in the structure.



Figures (3.5-3.6): Trend of the sum-squared error of the network versus number of iterations using BP method.

Back-propagation Method			
lr	Epochs	c.p.u time second	Final value of the total error
0.05	20000	610.82	7.52
0.03	20000	593.03	20.0
0.02	20000	592.65	2.0
0.006	253	9.22	0.02
0.005	20000	593.85	2.0

Table 3.1: Training values using BP method.

Back-propagation Method			
lr	Epochs	c.p.u time second	Final value of the total error
0.05	122	5.71	0.0002
0.03	112	5.33	0.0002
0.02	91	4.78	0.0002
0.006	92	5.27	0.0002
0.005	94	4.83	0.0002

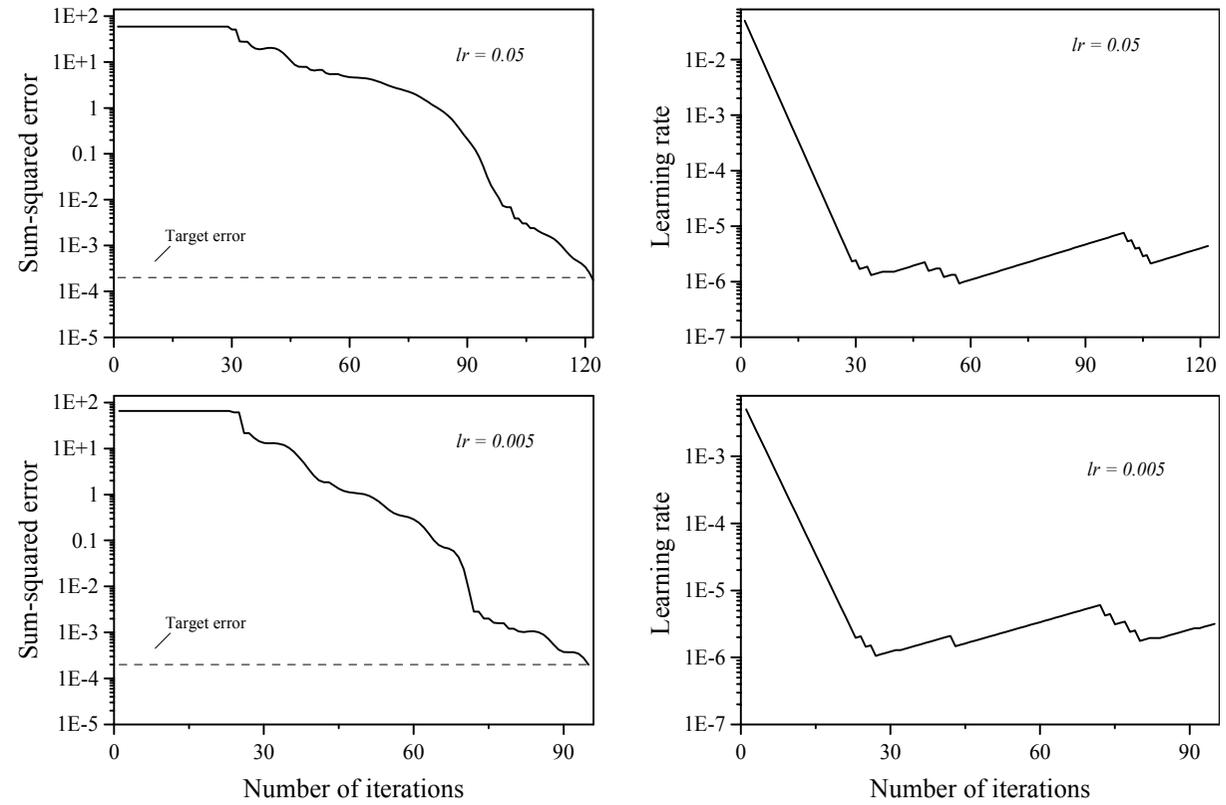
Table 3.2: Training values using BP method.

Back-propagation Method					
lr	0.05	0.03	0.02	0.006	0.005
0	0.712	1.500	0.000	0.050	0.000
1	1.249	1.500	1.500	1.003	1.500
2	1.249	1.500	1.500	1.996	1.500
3	3.478	1.500	3.000	2.951	3.000
4	3.509	4.000	4.000	4.002	4.000

Table 3.3: Target and NN output using BP method.

Modified Back-propagation Method					
lr	0.05	0.03	0.02	0.006	0.005
0	0.001	0.000	0.000	0.000	0.001
1	0.999	0.994	1.000	1.003	0.999
2	2.003	2.004	1.998	2.001	1.998
3	3.003	3.000	2.997	2.994	2.993
4	4.003	4.000	3.996	4.000	3.998

Table 3.4: Target and NN output using MBP method.



Figures (3.7 a,b - 3.8 a,b): Network error and learning rate versus number of iterations.

3.4.4 Example 2

The second example deals with the problem of damage detection in a multi-bay planar truss having the same structural scheme and characteristics of the one introduced in section (2.6.4). Following the strategy adopted in the previous example all the possible damage conditions of the structure are considered. Although, in this case, due to the major number of elements in the structure a major number of possible damage conditions has to be assumed.

If the network has to classify many patterns then overlapping between different classes is possible. For this reason, in the present study, instead of using a single network, four sub-networks were trained to identify damage in the entire structure. Referring to fig.(2.6.4) the classes of longerons, diagonals and battens are considered. The classes are denoted as B, C, and D respectively. For each class a three layers feedforward neural network, trained using the back-propagation algorithm, modified with adaptive learning rate and momentum, is considered. The input neurons correspond to the transfer function ordinates arising from the analysis of the structure for which a single element within that class is damaged.

A fourth neural network, denoted as A, is trained to distinguish between:

- a) Undamaged structure
- b) Structure with damage in class B
- c) Structure with damage in class C
- d) Structure with damage in class D

The flowchart of fig.(3.9) shows and clarifies the adopted strategy.

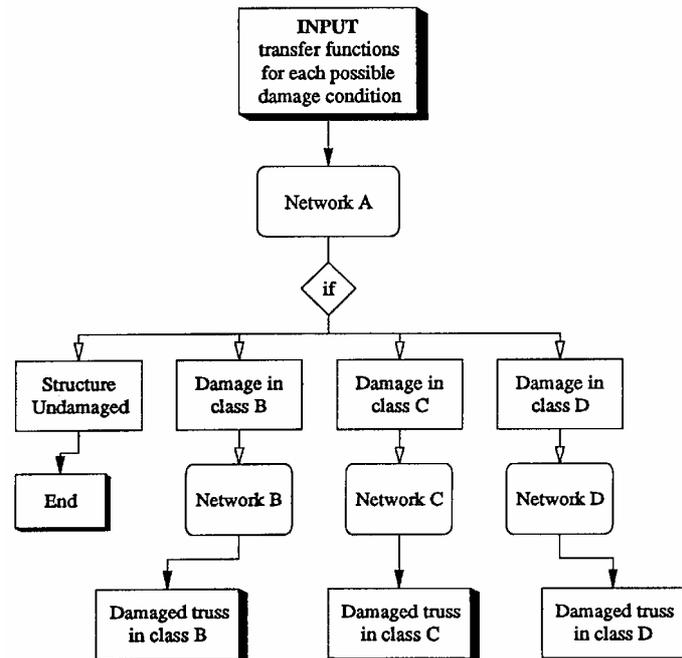


Figure 3.9: Flowchart of the global neural network.

The structure is considered subjected to white noise excitation, applied at node 11 in direction y . At the same node two sensors are located to measure the displacement response of the structure in the directions x and y . First the dynamical behavior of the structure is simulated using F.E. analysis by the above mentioned code Abaqus 5.5, then for each possible damage situation, the transfer functions are built considering for the displacements along the directions x and y .

In perfect analogy with the first example, fig.(3.10) shows the changes of the transfer functions due to the damage in the structure.

In order to train the network, four load conditions were considered. For each of them $2n$ transfer functions are available, (where n is the number of elements in the truss structure). Using the output in direction y it is easily to understand

how the transfer functions of the structure, when the upper longitudinal element is removed, are the same of the one obtained removing the lower longitudinal element for fixed bay. The same happens when diagonal elements are removed.

Hence the procedure adopted for damage detection does not lose generality if one considers the symmetry of the structure. The geometrical conditions of the structure allows the reduction of the number of damaged situations from 46 to 27.

For example, if attention is focused on network B, the training is performed using the transfer functions of the structure when the elements from 1 to 9 are removed one by one.

The amount of data is reduced, but on the other side the network does not locate uniquely the damage, because the outputs can be associated to two possible damage conditions. To avoid this problem the transfer functions in the direction x are also considered. Figure (3.11) shows that in this case transfer functions are no longer symmetric. The flowchart of fig.(3.9) maintains its validity but another step has been taken to distinguish which element is damaged between the two candidates.

The characteristics of each neural network are the same of the one utilizes in the first example. And for each of them the same general considerations can be repeated.

	Time [sec]	Epochs	Efficiency
Network A	1.883	20.000	100%
Network B	2.070	20.000	90%
Network C	903	8.897	100%
Network D	432	4492	100%

Table 3.5: Neural Networks data..

Table (3.5) shows the time training for each network and its efficiency in damage detection. The results show that the network A was fully efficient in the classification of the damaged class. Networks C and D worked as well as

network A and were able to exactly indicate the damaged element. Only network B presents some ambiguity in detecting the damaged elements between two near longerons, such as elements 2 and 3 or 21 and 22.

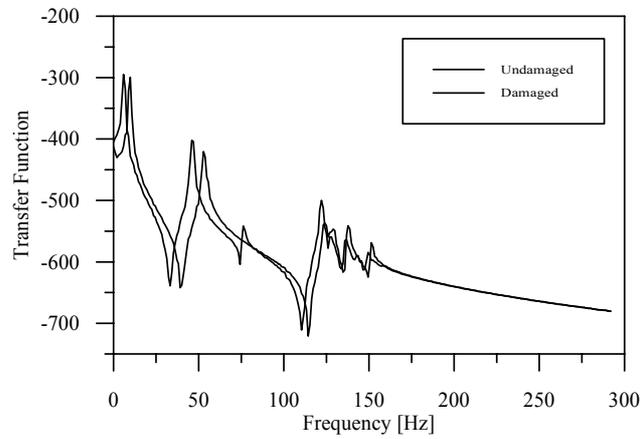


Figure 3.10: Transfer functions for damaged and undamaged structure.

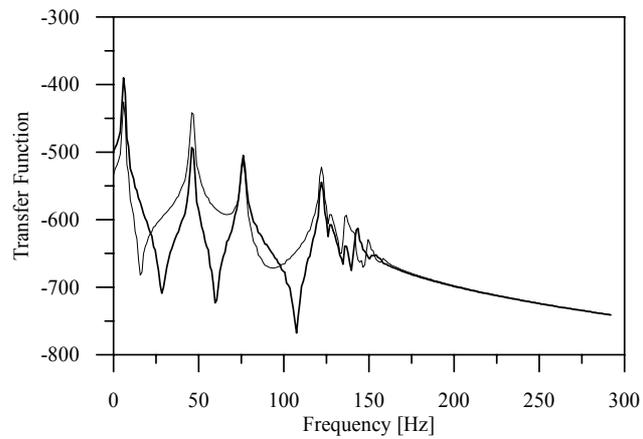


Figure 3.11: Transfer Functions of two symmetric elements.

3.4.5 Example 3

In the previous two examples the efficiency of using a neural network trained by the transfer functions of the structure was proved for damage detection either in a simple frame or in a multi-bay truss. In both cases the hypothesis of linear elastic behavior of the structure was made and the damage was simulated removing an element.

The following example deals with two problems: the non material linear behavior and the fragility analysis of the structure. Considering again the transfer functions of the structure, the results show that they change significantly when one or more plastic hinges are activated. The network recognizes the differences between the transfer functions and establish the degree of damage present in the structure, this information can be used to update the probability of local failure of the structure through fragility curves, independently evaluated via simulation.

To make the reader familiar with the safety analysis problem, some fundamental concepts on the subject will be presented in the next section.

3.4.5.1 Reliability of the structure via fragility curve

In the safety analysis of complicate structural systems it is often convenient to distinguish between uncertainty on input loads and uncertainty on system proprieties. In particular in seismic risk, uncertainty on earthquake loads is typically represented through a hazard function $H_S(s)$, which gives the rate at which an appropriately defined site intensity parameter S exceed the value s . The other component of the global uncertainty cover the system behavior and resistance and it is generally modeled by a random vector $X = (x_1, \dots, x_n)^T$ or a random process $x(t)$. The conditional failure probability or fragility function $p_f(s)$ is the probability of system failure given that $S = s$. The functions $H_S(s)$ and $p_f(s)$ can be combined into:

$$v_f = \int_{-\infty}^{+\infty} H_S(s) p_f(s) ds \quad (3.4.5.1.1)$$

to assess the structural reliability (Casciati and Faravelli, 1991).

The fragility of such a system may depend on one or more limit state functions $g_i(\mathbf{x}, \boldsymbol{\tau})$, being $\boldsymbol{\tau} = \{\tau_1, \dots, \tau_n\}^T$ a parameter vector (for example time).

If one or more of these functions are negative for some fixed vector $\boldsymbol{\tau}$, the structure will fail for such a system state.

The fragility of the structure can then be expressed as:

$$p_f = \int_{g(\mathbf{x}, \boldsymbol{\tau}) \leq 0} f_x(\mathbf{x}, \boldsymbol{\tau}) d\mathbf{x} \quad (3.4.5.1.2)$$

where for fixed $\boldsymbol{\tau}$, $f_x(\mathbf{x}, \boldsymbol{\tau})$ is the probability density function (p.d.f.) of the random basic variable vector \mathbf{X} and $g(\mathbf{x}, \boldsymbol{\tau})$ is an appropriate envelope of the $g_i(\mathbf{x}, \boldsymbol{\tau})$.

In studying the structure of limit state functions, we often encounter the problem that these functions are not known explicitly. Another problem is also the mathematical computation of failure probabilities.

Beside the Monte Carlo method, several approximation methods were proposed under the assumption that the limit state function is known in analytical form. Many authors (Hohenbickler & Rackwitz, 1981; Breitung 1984; Der Kiureghian, 1989; Madsen, Krenk & Lind, 1986) proposed to estimate the reliability by means of the so-called reliability indices. The basic idea of these methods, presented in a paper by Hasofer and Lind (Hasofer & Lind, 1974), is to identify in the failure domain the point or the points where the probability density is maximum and then to approximate the failure probability by an integral of the p.d.f. over a domain which is bounded by a hyperplane or a quadratic surface obtained by a suitable Taylor expansion in these points. Originally this method was developed for standard normal random vectors. In this case the points with larger values of the p.d.f. in the failure domain are the points with minimal distance to the origin. After this minimal distance is found, the limit state function is replaced by a first or second order Taylor expansion. Let β be this minimal distance, the probability is given by:

$$p_f \cong \Phi(-\beta) \quad (3.4.5.1.3)$$

where $\Phi(\cdot)$ represents the cumulative Gaussian distribution. The methods based on the linearization of the limit state function are known as FORM (First Order real Methods). Methods using quadratic expansions at the minimal distance points are called SORM (Second Order Reliability Methods).

When the analytical form for the limit state function is not available a polynomial approximation can be derived from the results of numerical experiments appropriately planned (response surface technique (Faravelli, 1989)). The fragility analysis of the structure under investigation is defined as in the following.

Structural elements undergo progressive deterioration when subjected to intense cyclic loading. The damage of the structure is assumed to be localized and identified by the procedure defined above.

The limit events are the achievement of excessive inelastic rotation in a single section and the activation of the collapse mechanism. These events are combined to provide:

- 1) the fragility of the undamaged structure for the progressive activation of excessive inelastic rotations and eventually for the activation of a collapse mechanism;
- 2) the fragility of the structure for different initial levels of damage.

The structural fragility in terms of local rupture is characterized by:

$$\mathcal{G}_i \geq \mathcal{G}_{Li} \quad (3.4.5.1.4)$$

where \mathcal{G}_i is the inelastic rotation and \mathcal{G}_{Li} its limit value. The latter one is assumed to be random, while the inelastic rotation turns out to be random due to the stochastic nature of the external excitation.

Using the probabilities of local failure, obtained for each section and for different initial levels of damage in the structure it is possible to build some fragility curves for the system under investigation.

The total probability of failure, given a seismic intensity, is evaluated in terms of mode failure probabilities as:

$$p_f = P(F_1) + P(F_2) - P(F_1 \cap F_2) + P(F_3) - P(F_1 \cap F_3) - P(F_2 \cap F_3) + P(F_1 \cap F_2 \cap F_3) + \dots ; \quad (3.4.5.1.5)$$

where F_i denotes the event of occurrence of the i -th local rupture and $P(F_i)$ the corresponding probability of occurrence. The event $(F_i \cap F_j)$ means that the local failure simultaneously occurs in both the i -th and j -th plastic hinge. Given the matrix of covariance function between the inelastic rotations, one can calculate the matrix of covariance of the local modes and evaluate p_f by second order bounds (Ditlevsen and Madsen, 1996).

3.4.5.2 Application and results

The non linear dynamic analyses of a plane frame is developed by using the finite element code DRAIN-2D which assumes that each beam element behaves elastically and that plastic hinges might be form only at the two extreme sections. The constitutive law between the bending moment and the hinge rotation is assumed to be elastic perfectly plastic.

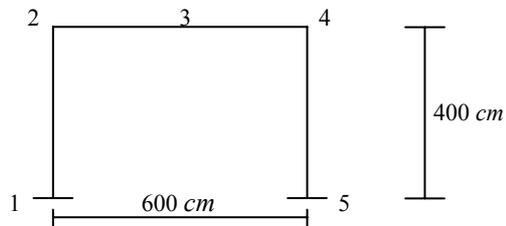


Figure 3.13: Structural scheme.

The numerical calculations were developed with reference to the simple frame of fig.(3.12), where the sections of interest, in which plastic rotation can potentially develop, are pointed out. The geometry and material properties are given in Table (3.6).

This numerical example consists of two parts: the first one deals with the network training, for damage identification purposes; the second part deals with the reliability analysis.

Property	Value
Young's Modulus	2.1E 6 <i>kg/ cm²</i>
Strain hardening ratio	0.01
Yield moment for the beam	646.800 <i>kg cm</i>
Yield moment for the column	239.800 <i>kg cm</i>

Table 3.6: Numerical values adopted in the numerical example.

Five seismic acceleration records were selected from the databank SMCAT. All the time histories present the characteristic of being significant over a time of 20 second. The selected accelerograms were scaled to produce different level of damage in the structure, the damage being the activation of a plastic hinge in any section of the structure.

Starting from the undamaged condition, and using one of the selected accelerograms, five seismic intensities, S_j ($j = 0, 1, 2, 3, 4$) were identified to provide an elastic behavior of the structure and the activation of one, two, three and four plastic hinges, respectively.

Using DRAIN 2D, the response in terms of acceleration at node 4 was calculated for each of the five accelerograms described above. Using Fourier analysis the transfer functions corresponding to each time response were derived. Figure (3.14) shows the transfer functions of the system for different levels of damage. Other analyses, using a different accelerogram, properly scaled, prove that these functions are invariant for given number of plastic

hinges and given level of limit plastic rotation. This allows one to use the transfer functions as characteristic functions also in this case.

In order to distinguish between the five resulting shapes of transfer function a feedforward neural network was trained to associate each transfer function with a Boolean vector of five elements corresponding to the five sections of interest along the frame. The elements of the Boolean vector have value 0 if the corresponding section does not present a plastic rotation, 1 otherwise.

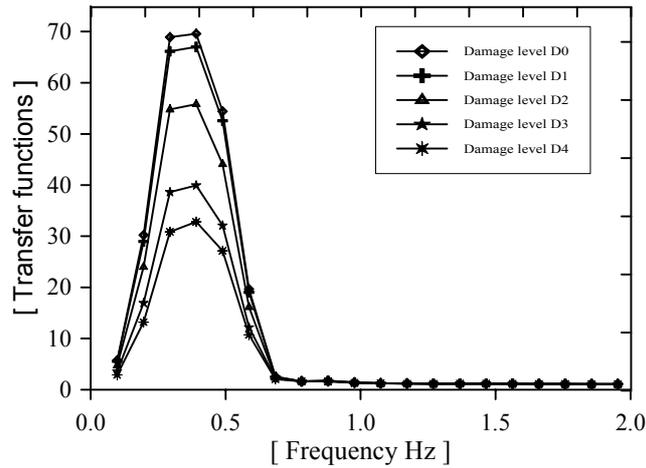


Figure 3.14: Transfer functions of the structure for different levels of damage D_j .

The network has two hidden layers having respectively 100 and 50 neurons and was programmed with a learning rate variable in a range of 2% and a tolerance error of 0.01. After its training, the network was tested using the transfer functions of the structure obtained starting from different damaged conditions for different accelerograms scaled to given values of intensity.

The results of this test are presented in table (3.7), where the output of the network, during a test, are compared with the target ones (in bold).

The trained network can now be considered as an effective tool to identify the number of plastic hinges in the structure.

0.0034	0	0.0164	0	0.0094	0	0.0284	0	0.9671
0.0342	0	0.9770	1	0.9787	1	0.9821	1	0.9940
0.0038	0	0.0044	0	0.0108	0	0.0045	0	0.0010
0.0229	0	0.0267	0	0.9658	1	0.9908	1	0.9872
0.0112	0	0.0095	0	0.0316	0	0.9721	1	0.9826

Table 3.7: Example of neural network output.

The second part of this study concerns the probability analysis. To obtain the fragility curves of the system, it was necessary to produce the statistics of the inelastic rotations \mathcal{G}_i for each section of interest in the structure and for assigned initial damaged condition. The distribution of these variable is assumed to be Gaussian and numerical simulations were developed to assess their means and standard deviations.

The limit value of the inelastic rotation \mathcal{G}_{Li} , is characterized on the basis of experimental values obtained from laboratory tests (Casciati and Faravelli, 1989): the mean is set equal to 0.032 and the coefficient of variation to 0.2.

<i>Intensity</i>	S_0	S_1	S_2	S_3	S_4
<i>Section</i>					
1	0.0	0.0	0.0006	0.0046	0.0075
	0.0	0.0	0.0009	0.0031	0.0040
2	0.0028	0.0098	0.0151	0.0195	0.0198
	0.0027	0.0060	0.0070	0.0048	0.0059
4	0.0029	0.0100	0.0143	0.0247	0.0272
	0.0040	0.0065	0.0053	0.0049	0.0071
5	0.0	0.0	0.0005	0.0046	0.0076
	0.0	0.0	0.0008	0.0031	0.0040

Table 3.8: Mean and variance of the four maximum inelastic rotations from a sample of 5 accelerations.

Two different conditions were analyzed: the undamaged structure was first considered and the statistics calculated using different earthquakes scaled by five different factors.

Table (3.8) reports the statistics of the maximum inelastic rotations computed as sample of five accelerograms.

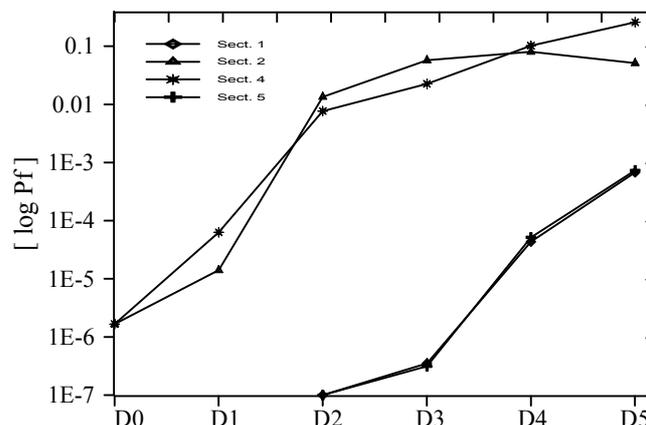


Figure 3.15: Fragility vs damage for intensity S_0 of the occurring seismic event.

In table (3.8) S_i indicate the seismic intensity parameter.

Once the means and the standard deviations of the inelastic rotation were assessed, for each section, the limit state function given in eq.(3.4.5.1.4) and the corresponding probability of failure, eq.(3.4.5.1.3), gives rise to the updated curves in figures (3.15-3.19).

The combined use of neural network and fragility curves provides a systematic approach to define a probability of local failure of the structure during its operating life. Of course the method presents the non-negligible defect to require a priori studies of the structure to define the fragility curves either for given initial damage either for given earthquake intensity. On the other hand, the proposed approach represents a practical way to associate, in real time, the presence of damage to a probability of failure of the structure.

The preliminary numerical results show a promising accuracy of the method.

<i>Section</i>	<i>Intensity</i>				
	S_0	S_1	S_2	S_3	S_4
1	0.0	0.0	0.0008	0.0052	0.0082
	0.0	0.0	0.0012	0.0029	0.0040
2	0.0102	0.0176	0.0224	0.0257	0.0261
	0.0073	0.0096	0.0099	0.0087	0.0098
4	0.0087	0.0158	0.0193	0.0286	0.0312
	0.0070	0.0092	0.0074	0.0082	0.0104
5	0.0	0.0	0.0008	0.0053	0.0082
	0.0	0.0	0.0011	0.0028	0.0041

Table 3.9: Mean and variance of the four maximum inelastic rotations for detected initial damage D_1 .

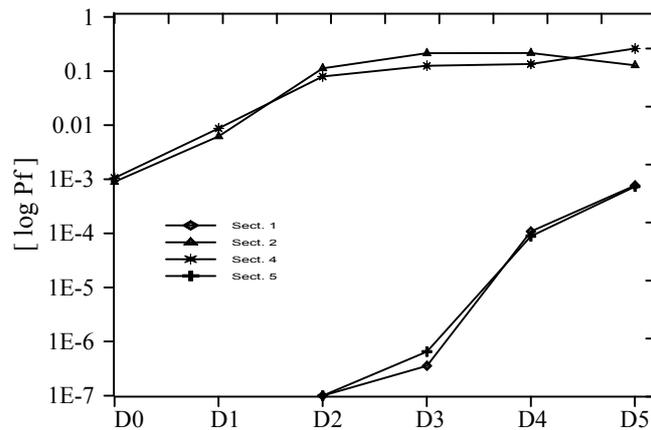


Figure 3.16: Fragility vs damage for intensity S_1 of the occurring seismic event.

<i>Intensity</i>	S_0	S_1	S_2	S_3	S_4
<i>Section</i>					
1	0.0002	0.0002	0.0011	0.0061	0.0091
	0.0003	0.0004	0.0015	0.0027	0.0041
2	0.0146	0.0218	0.0261	0.0282	0.0286
	0.0087	0.0106	0.0104	0.0105	0.0116
4	0.0125	0.0190	0.0222	0.0303	0.0329
	0.0071	0.0089	0.0072	0.0089	0.0114
5	0.0001	0.0001	0.0010	0.0060	0.0090
	0.0001	0.0001	0.0013	0.0026	0.0040

Table 3.10: Mean and variance of the four maximum inelastic rotations for detected initial damage D_2 .

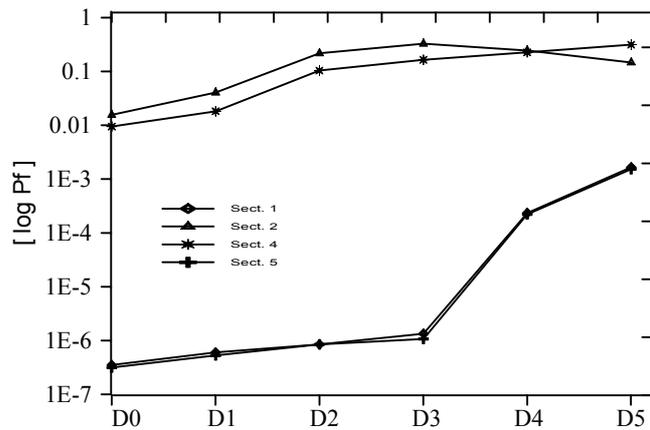


Figure 3.17: Fragility vs damage for intensity S_2 of the occurring seismic event.

Intensity	S_0	S_1	S_2	S_3	S_4
Section					
1	0.0036	0.0043	0.0054	0.0117	0.0132
	0.0033	0.0038	0.0040	0.0051	0.0064
2	0.0198	0.0218	0.0252	0.0275	0.0286
	0.0055	0.0067	0.0066	0.0077	0.0070
4	0.0225	0.0234	0.0260	0.0311	0.0337
	0.0033	0.0038	0.0038	0.0069	0.0072
5	0.0039	0.0043	0.0053	0.0114	0.0129
	0.0033	0.0036	0.0040	0.0052	0.0065

Table 3.11: Mean and variance of the four maximum inelastic rotations for detected initial damage D_3 .

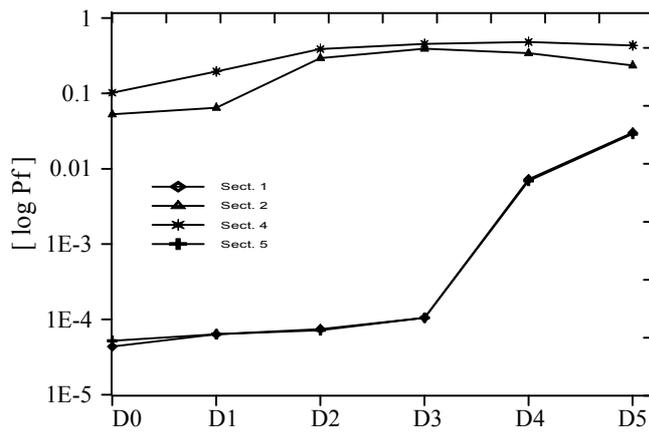


Figure 3.18: Fragility vs damage for intensity S_3 of the occurring seismic event.

Intensity	S_0	S_1	S_2	S_3	S_4
Section					
1	0.0067	0.0068	0.0079	0.0134	0.0151
	0.0045	0.0046	0.0050	0.0073	0.0079
2	0.0187	0.0215	0.0221	0.0243	0.0254
	0.0047	0.0062	0.0064	0.0077	0.0070
4	0.0262	0.0262	0.0274	0.0324	0.0347
	0.0053	0.0054	0.0058	0.0083	0.0086
5	0.0067	0.0067	0.0077	0.0131	0.0149
	0.0046	0.0046	0.0050	0.0074	0.0080

Table 3.12: Mean and variance of the four maximum inelastic rotations for detected initial damage D_4 .

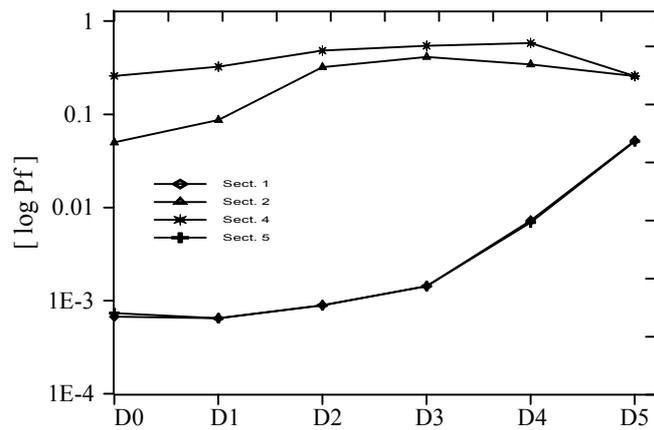


Figure 3.19: Fragility vs damage for intensity S_4 of the occurring seismic event.

3.5 Conclusion

This chapter has dealt with the damage detection problem in a structural system by means of a feedforward neural network. It has pointed out that the transfer functions of the structure, derived from dynamic analyses, starting from different damage conditions, contain all the information about the mechanical behavior of the structure and then can be regarded as characteristic functions. Moreover this property is also maintained in the non-linear field.

Throughout the numerical applications have been evidenced how a three layer feedforward neural network, appropriately organized, can identify correctly the damage in a simple structure.

Moreover it has been shown how the same problem can be successful treated also in complex structures using a set of network each of one has to solve a sub-problem in a sub-structure.

Finally the proposed approach, combined with the knowledge of the fragility curves of the structure, has been applied to evaluate the probability of local failure of the structure.

Chapter 4

Potential Systems for System Identification

4.1 Introduction

Real systems have random nature. Their stochastic behavior can be related either to the statistical variation of the materials or to the randomness of the external excitation. Modeling the dynamic behavior even for a simply deterministic structure subject to random forces is not a trivial problem, especially when the non-linear behavior is investigated.

Mathematically exact solutions are available only under appropriate basic hypothesis. When the random excitation can be modeled as a Gaussian white noise process, so that the response of the structure is a Markov process, then the probability density function of the system response can be evaluated in a closed form by solving the so called Fokker Planck equation. The latter shows how the probability of the response of the structure evolves in time.

For non-linear systems complete analytical solution of the Fokker Planck equation are known only for some special first order system (Caughey, 1971; Gardiner, 1983). Then to solve the problem different attempts of numerical procedure approaches have been proposed for higher order cases (Naess and Johnsen, 1983; Spencer and Bergman, 1993).

For higher order non-linear systems analytical solutions are found in literature only for the reduced Fokker Planck equation in which the time derivative term is assumed equal to zero. The resolution of this equation brings to the knowledge of the unconditional probability density function of the system response when it reaches the state of the statistical stationarity. A stationary solution of the system is obtainable only under some conditions (Lin and Cai, 1995) namely:

- the random excitations applied on the system are stationary processes (that is the case for the stationary Gaussian white noise);
- the system parameters are time invariant;
- the energy input from the random excitations is statistically balanced by energy output from dissipation, if any dissipation energy exist in the system.

The advantages of knowing the exact stationary solution are evident. For this reason much research effort has been devoted to the identification of such class of systems whose analytical probabilistic response could be obtained in a closed form.

Potential systems are interesting since:

- 1) they are able to reproduce different types of non linearity;
- 2) they represent a class of systems from which the probability density function of the response is known.

Both these characteristics make potential systems useful also for identification problems, as it will appear clear in the following.

A general definition of potential system is first due to Lin and Cai (1988); the response of these systems is characterized by a probability density function having an exponential form. This class of system, known as *generalized stationary potential*, includes all the particular classes of system having analytical solution, obtained by other authors in previous research studies, and allows also to consider the action of parametric forces. Moreover the results obtained by Lin and Cai were further generalized by To and Li (1991).

In the following sections only some fundamental properties of the potential systems will be reported; for a more extensive treatment of the subject the reader can refer to (Lin and Cai, 1995).

The main part of this chapter is devoted to the presentation of a particular class of potential system whose analytical properties make them versatile to be used for identification purposes. Moreover, using the concept of equivalent non linearization, systems not included in the potential class can be identified through them, this is because they share the same response probability distribution.

4.2 The Fokker Planck equation

Analytical solutions for non linear systems subjected to random excitations are mainly based on the assumption that the system response can be modeled as a Markov stochastic process. Reminding that if $X(t)$ is a random process a sufficient condition to be a Markov process is that its increments are independent in any two non-overlapping time increments. Markov processes are just a mathematical approximation of a physical process, justified only by the fact that increments of random processes, in non-overlapping time, can be assumed stochastically independent.

The conditional probability $\text{Prob}[X(t) \leq x \mid X(t_0) = x_0]$ of a Markov process $X(t)$ is known as *transition probability distribution function* and its derivative, if exists, gives the *transition probability density function* defined as:

$$q(x, t \mid x_0, t_0) = \frac{\partial}{\partial x} \text{Prob}[X(t) \leq x \mid X(t_0) = x_0] \quad (4.2.1)$$

while the higher order probability densities assume the form:

$$p(\mathbf{x}_1, t_1; \mathbf{x}_2, t_2; \dots; \mathbf{x}_n, t_n) = q(\mathbf{x}_n, t_n \mid \mathbf{x}_{n-1}, t_{n-1}) q(\mathbf{x}_{n-1}, t_{n-1} \mid \mathbf{x}_{n-2}, t_{n-2}) \dots \quad (4.2.2)$$

$$q(\mathbf{x}_2, t_2 \mid \mathbf{x}_1, t_1) p(\mathbf{x}_1) \quad t_1 < t_2 < \dots < t_n$$

showing how the behavior of a Markov process at several instants can be obtained from the initial probability density and the transition probability density.

If the initial state of a Markov process is known then equation (4.2.1) alone characterizes the stochastic process. Making a limit for $t-t_0 \rightarrow \infty$ the stationary probability density can be derived :

$$p(\mathbf{x}) = \lim_{t-t_0 \rightarrow \infty} q(\mathbf{x}, t | \mathbf{x}_0, t_0) . \quad (4.2.3)$$

Equation (4.2.3) represents the *unconditional probability density function* of the process $\mathbf{X}(t)$ because when t goes to infinity the influence of the initial condition will disappear. The higher order probability densities at the stationary state can be written as:

$$p(\mathbf{x}_1, t_1; \mathbf{x}_2, t_2; \dots; \mathbf{x}_n, t_n) = q(\mathbf{x}_n, t_n - t_{n-1} | \mathbf{x}_{n-1}) q(\mathbf{x}_{n-1}, t_{n-1} - t_{n-2} | \mathbf{x}_{n-2}) \dots q(\mathbf{x}_2, t_2 - t_1 | \mathbf{x}_1) p(\mathbf{x}_1) \quad t_1 < t_2 < \dots < t_n . \quad (4.2.4)$$

The evolution in time of the transition probability density of a Markov process is governed by the well-known Fokker Planck equation :

$$\frac{\partial}{\partial t} q + \frac{\partial}{\partial x_j} (a_j q) - \frac{1}{2} \frac{\partial^2}{\partial x_j \partial x_k} (b_{jk} q) + \frac{1}{3!} \frac{\partial^3}{\partial x_j \partial x_k \partial x_l} (c_{jkl} q) - \dots = 0 \quad (4.2.5)$$

where the symbols a_j , b_{jk} and c_{jkl} represent the derivative moments of the Markov process :

$$a_j = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} E[X_j(t + \Delta t) - X_j(t) | \mathbf{X}(t) = \mathbf{x}] \quad (4.2.6)$$

$$b_{jk} = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} E\{[X_j(t + \Delta t) - X_j(t)][X_k(t + \Delta t) - X_k(t)] | \mathbf{X}(t) = \mathbf{x}\} \quad (4.2.7)$$

$$c_{jkl} = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} E \{ [X_j(t + \Delta t) - X_j(t)][X_k(t + \Delta t) - X_k(t)] \cdot [X_l(t + \Delta t) - X_l(t)] | \mathbf{X}(t) = \mathbf{x} \} . \quad (4.2.8)$$

If the process is also Gaussian then the Fokker Planck equation assumes the reduced form:

$$\frac{\partial}{\partial t} q + \frac{\partial}{\partial x_j} (a_j q) - \frac{1}{2} \frac{\partial^2}{\partial x_j \partial x_k} (b_{jk} q) = 0 \quad (4.2.9)$$

with the derivative cumulants of order higher than two being equal to zero.

For non-linear oscillators exact solutions are known in the stationary condition when q is independent of time, then equation (4.2.9) is reduced to the form:

$$\frac{\partial}{\partial x_j} (a_j p) - \frac{1}{2} \frac{\partial^2}{\partial x_j \partial x_k} (b_{jk} p) = 0 . \quad (4.2.10)$$

Another important property to outline is that an one-to-one correspondence between the Fokker Planck equation, describing the probability density function of a Markov process, and the Itô stochastic differential equation, describing the process, exists.

The Markov process $X(t)$, following the Itô theory, must satisfy the following differential equation:

$$dX(t) = m(X, t) dt + \sigma(X, t) dB \quad (4.2.11)$$

where $B(t)$ is a unit Wiener process such as:

$$E[dB(t_1) dB(t_2)] = \begin{cases} 0 & t_1 \neq t_2 \\ dt & t_1 = t_2 = t \end{cases} \quad (4.2.12)$$

while m and σ are called respectively drift and diffusion coefficients and are related to the coefficients of the reduced Fokker Planck equation by means of :

$$a(x, t) = [m(X, t)]_{X=x} \quad (4.2.13)$$

$$b(x, t) = [\sigma^2(X, t)]_{X=x} \quad (4.2.14)$$

Equations (4.2.13) and (4.2.14) make possible to pass from the drift and diffusion coefficients of the Itô differential equation to the first and second derivative moments of the Fokker Planck equation by simple replacing the random process $X(t)$ with the state variable x .

Analogous relationship can be written considering an n -dimensional Markov process. The advantages in working with the Itô differential equations consist in the fact that any arbitrary function of a Markov process can be derived quite simply using the Itô differential rules.

For sake of completeness it must be reminded that there are other type of differential equations able to represent a Markov process like the Stratonovich type stochastic differential equations. But they will not be used in the following.

4.3 Generalized stationary potential

Let consider a non-linear system, whose associated reduced Fokker Planck equation can be written in the form:

$$\frac{\partial}{\partial x_j} G_j = 0 \quad (4.3.1)$$

with:

$$G_j = a_j(\mathbf{x})p(\mathbf{x}) - \frac{1}{2} \frac{\partial}{\partial x_k} [b_{jk}(\mathbf{x})p(\mathbf{x})] \quad (4.3.2)$$

the function G_j is called probability flow in the j -th direction.

Suppose now that the probability flow vanishes for any direction j , then the stochastic system under investigation is said to belong to the class of stationary potential systems (Stratonovich, 1963).

For this class of systems the stationary probability function can be expressed in exponential form:

$$p(\mathbf{x}) = Ce^{-\phi(\mathbf{x})} \quad (4.3.3)$$

where C is a non negative normalization constant while ϕ is an appropriate probability potential function. By substituting eq.(4.3.3) in eq.(4.3.1) one obtains:

$$b_{jk} \frac{\partial \phi}{\partial x_k} = \frac{\partial}{\partial x_k} b_{jk} - 2a_j \quad (4.3.4)$$

which represents a system of n partial differential equations.

The stochastic system is said belonging to the class of a stationary potential systems if there exists a consistent function ϕ that satisfies all these equations. It can be shown that this is possible if the matrix containing all the diffusion coefficients b_{jk} is invertible. To extend the class of systems for which eq.(4.3.1) has a solution Cai and Li proposed to split the drift and the diffusion coefficients in two parts:

$$a_j = a_j^1 + a_j^2 \quad (4.3.5)$$

$$b_{jk} = b_{jk}^j + b_{jk}^k \quad (4.3.6)$$

where a_j^1 represents a portion of the drift in the j -th direction which is required to balance the differential diffusion and maintain equal to zero the probability flow in the j -th direction, while a_j^2 is a portion of the drift related to the circularity flow.

Substituting eqs.(4.3.3), (4.3.5) and (4.3.6) in eq.(4.3.1) it can be shown that this equation can be satisfied if the following conditions hold:

$$b_{jk}^j \frac{\partial \phi}{\partial x_k} = \frac{\partial}{\partial x_k} b_{jk}^j - a_j^1 \quad (4.3.7)$$

$$\frac{\partial}{\partial x_j} a_j^2 = a_j^2 \frac{\partial \phi}{\partial x_j}. \quad (4.3.8)$$

Equation (4.3.7) is related to the vanishing probability flow, while eq.(4.3.8) with the circulatory flow. The problem is solvable if a consistent ϕ function can be obtained from these equations.

If a given system with the spitting of the drift and of the diffusion coefficients satisfies eqs.(4.3.7) and (4.3.8) then the system is said to belong to the class of generalized potential systems. It is important to note that these two last equations reproduce a relationship existing between the system parameters and the spectral level of the excitation. These equations are adopted to develop an exact solution technique of the Fokker Planck equation (Cai and Lin, 1995).

The class of generalized stationary potential is the more general among the classes of potential system and allows treating systems with non-linear damping and subjected to multiplicative excitations.

4.4 A reduced class of potential systems

In the following section a particular class of potential models will be presented. This class can be thought as a particular case of the more general model proposed by To and Li (1991) and for this reason is called *Reduced*

Potential Model (MPR), Cavaleri (1998). The reason to look at a less general class of models lies in the fact that the MPR are really simple to treat mathematically and moreover they allow to model the behavior of a wide class of real non-linear systems.

In the following the capital letter will be used to refer to a random process, while the corresponding small letter indicates its domain.

Let consider the stochastic system:

$$\ddot{X} + l(X, \dot{X}) = f_i(X, \dot{X})W_i \quad (i=1, 2, \dots, n) \quad (4.4.1)$$

where l and f_i are, in general, non linear functions, while W_i are Gaussian white noises having delta type correlation function:

$$E[W_i(t)W_j(t+\tau)] = 2\pi K_{ij}\delta(\tau) \quad (4.4.2)$$

K_{ij} being the spectrum matrix components of the vector process W .

According to To and Li, (1991) the system (4.4.1) belongs to a class of potential systems if the following condition is verified:

$$l(x, \dot{x}) = \pi \dot{x}_i K_{ij} f_i(x, \dot{x}) f_j(x, \dot{x}) \cdot \left[\lambda_y \frac{d\phi_0(\lambda)}{d\lambda} - \frac{\lambda_{yy}}{\lambda_y} \right] + \\ - \pi K_{ij} f_i(x, \dot{x}) \frac{\partial f_j(x, \dot{x})}{\partial \dot{x}} + \frac{\lambda_x}{\lambda_y} + \frac{g(x)}{\lambda_y} e^{\phi_0(\lambda)} \quad (4.4.3)$$

where $g(x)$ and λ are positive arbitrary functions, while $\lambda_y, \lambda_x, \lambda_{yy}$ are obtained from the relationship between the probability potential ϕ and the coefficients of the reduced Fokker Planck equation associated to the system (4.4.1) written in the state variable representation.

Assume now that:

$$i) \quad f_i(x, \dot{x}) = 1; \quad \forall i \quad (4.4.4)$$

$$ii) \quad \Lambda = \frac{1}{2} \dot{X}^T \dot{X} + \int_0^x r(x) dx = H(X, \dot{X}) \quad (4.4.5)$$

$$iii) \quad \phi(\lambda) = Q(h) = \sum_{i=1}^n a_i h^i . \quad (4.4.6)$$

Condition *i)* means that the system is subjected only to external excitations; *ii)* $r(x)$ indicates the restoring force so that eq.(4.4.5) gives the total energy of the system; therefore *iii)* means that we only deal with the class of potential systems in which the potential function is given in polynomial form.

If moreover the function g is assumed equal to zero over all its domain of definition, then the function (4.4.3) can be rewritten as:

$$l(x, \dot{x}) = \frac{dQ(h)}{dh} K_{ij} \dot{x} + r(x) . \quad (4.4.7)$$

Considering the equations given above the governing equation of the stochastic system can be written as:

$$\ddot{X} + \pi \dot{X} K_{ij} \left[\frac{dQ(H)}{dH} \right] + r(X) = W_j \quad (4.4.8)$$

that represents the class of reduced potential models under study. For this class of models the expression of the probability density function is known :

$$p_{x\dot{x}}(x, \dot{x}) = C \exp(-Q(h)). \quad (4.4.9)$$

4.5 Some fundamental properties

To focus the attention on some fundamental properties of the MPR systems, let us consider a one degree of freedom non linear system having the governing equation:

$$\ddot{X} + \pi \dot{X} K \left[\frac{dQ(H)}{dH} \right] + r(X) = W \quad (4.5.1)$$

with $Q(h)$ expressible as a potential series of h such as:

$$\pi \dot{X} K \left[\frac{dQ(H)}{dH} \right] = \sum_{j=1}^k j a_j H^{j-1}. \quad (4.5.2)$$

Making the substitution $Y_1 = X$ and $Y_2 = \dot{X}$ equation (4.5.1) can be written in the state variable form as:

$$\begin{cases} \dot{Y}_1 = Y_2 \\ \dot{Y}_2 = -\frac{dQ(H)}{dH} Y_2 - r(Y_1) + W \end{cases} \quad (4.5.3)$$

that in the Itô representation becomes :

$$\begin{cases} d\dot{Y}_1 = Y_2 dt \\ d\dot{Y}_2 = -\left[\frac{dQ(H)}{dH} Y_2 + r(Y_1) \right] dt + dB(t). \end{cases} \quad (4.5.4)$$

Using the classical rules of the differential stochastic calculus applied to the function $H(Y_1, Y_2)$ one obtains:

$$\begin{aligned} dH = & \frac{\partial H}{\partial Y_1} dY_1 + \frac{\partial H}{\partial Y_2} dY_2 + \frac{1}{2} \frac{\partial^2 H}{\partial Y_1^2} (dY_1)^2 \\ & + \frac{1}{2} \frac{\partial^2 H}{\partial Y_2^2} (dY_2)^2 + \frac{\partial^2 H}{\partial Y_1 \partial Y_2} dY_1 dY_2 \end{aligned} \quad (4.5.5)$$

and neglecting the higher order infinitesimal in dt follows:

$$dH = \frac{dQ(H)}{dH} Y_2^2 dt + Y_2 dB + \frac{(dB)^2}{2} \quad (4.5.6)$$

where for sake of brevity the dependence on time of the variables H , Y , dB , is omitted.

In stationary conditions taking the expected value and dividing by dt eq.(4.5.6) becomes:

$$\dot{E}[H] = -E \left[\frac{dQ(H)}{dH} Y_2^2 \right] + \frac{s}{2} = 0 \quad (4.5.7)$$

where s is the intensity level of the white noise W .

Reapplying the same Itô rule to the process $H^i(Y_1, Y_2)$ and repeating the same steps that have brought to eqs.(4.5.6) and (4.5.7) the following relationship holds:

$$\dot{E}[H^i] = -iE \left[H^{i-1} \frac{dQ(H)}{dH} Y_2^2 \right] + i \frac{s}{2} E[H^{i-1}] + \frac{i(i-1)}{2} s E[H^{i-2} Y_2^2] = 0. \quad (4.5.8)$$

If eq.(4.4.5) is inserted in eq.(4.5.8) this assumes the form:

$$-\sum_{j=1}^k j a_j E[H^{i+j-1} Y_2^2] + \frac{s}{2} E[H^{i-1}] + \frac{(i-1)}{2} s E[H^{i-2} Y_2^2] = 0. \quad (4.5.9)$$

For $i=1, 2, \dots, k$ eq.(4.5.9) gives a set of equations in the unknown parameters a_j . The applicability of eq.(4.5.9) is restricted by the fact that it is necessary to calculate double integrals of the type:

$$E[H^n Y_2^2] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} h^n Y_2^2 p_{Y_1 Y_2}(y_1, y_2) dy_1 dy_2. \quad (4.5.10)$$

This drawback can be overcome every time is possible to split the double integral in the product of two simple integrals.

An example is reported in To and Li (1991) for the particular case in which the restoring force has the expression:

$$r(X) = f^2 \operatorname{sgn}(X) |X|^\nu \quad \nu > 0 \quad (4.5.11)$$

in this case it can be shown that the fundamental relationship holds:

$$\alpha = \frac{E[H^i Y_2^2]}{E[H^{i+1}]} = \text{constant}; \quad \forall i \quad (4.5.12)$$

in particular α is equal to one if ν is set equal to one; moreover from a numerical study results that eq.(4.5.12) is true for polynomial form of $r(X)$.

From this result, eq.(4.5.9) can be replaced by the following:

$$-\alpha \sum_{j=1}^k j a_j E[H^{i+j-2} Y_2^2] + \frac{s}{2} E[H^{i-1}] (1 + \alpha(i-1)) = 0. \quad (4.5.13)$$

Using the differential stochastic calculus some important characteristics of the moment's response can be derived.

First it is important to remind that the response acceleration \dot{Y}_2 process of any system subjected to a white noise input is itself a white noise, although the mean values of the product processes $Y_1\dot{Y}_2$ and $Y_2\dot{Y}_2$ are finite numbers.

Recalling the Itô differential equation for a general dynamic system one can write:

$$d\mathbf{Y}(t) = f(\mathbf{Y}, t) dt + g(\mathbf{Y}, t) dB(t) \quad (4.5.14)$$

where \mathbf{Y} is a vector having components $[Y_1(t) Y_2(t)]^T$.

For $t = t_1$ eq.(4.5.14) becomes :

$$d\mathbf{Y}(t_1) = f(\mathbf{Y}, t_1) dt_1 + g(\mathbf{Y}, t_1) dB(t_1) . \quad (4.5.15)$$

Now multiplying right and left members of eq.(4.5.15) by $Y(t_2)$, using the Kronecker algebra and taking the expected value follows:

$$E[d\mathbf{Y}(t_1) \otimes Y(t_2)] = E[f(\mathbf{Y}, t_1) \otimes Y(t_2)] dt_1 ; \quad t_2 \leq t_1 \quad (4.5.16)$$

being $E[g(\mathbf{Y}, t_1) dB(t_1) \otimes Y(t_2)] = 0$ for the *not-anticipative* property of the Wiener process B .

It must be noted that in stationary conditions the left hand side of eq.(4.5.16) gives the right derivative of the Y correlation function

$$\frac{\partial}{\partial t_1} R_Y(t_1 - t_2) = \begin{bmatrix} E \left[\dot{Y}_1(t_1) Y_1(t_2) \right] \\ E \left[\dot{Y}_1(t_1) Y_2(t_2) \right] \\ E \left[\dot{Y}_2(t_1) Y_1(t_2) \right] \\ E \left[\dot{Y}_2(t_1) Y_2(t_2) \right] \end{bmatrix} \quad (4.5.17)$$

while the right hand side of eq.(4.5.16), written in extended form, returns:

$$E[f(\mathbf{Y}, t_1) \otimes \mathbf{Y}(t_2)] = \begin{bmatrix} [Y_2(t_1) Y_1(t_2)] \\ [Y_2(t_1) Y_2(t_2)] \\ \left(-\sum_{j=1}^k ja_j H^{j-1} \right) [Y_2(t_1) Y_1(t_2) - r(Y_1(t_1)) Y_1(t_2)] \\ \left(-\sum_{j=1}^k ja_j H^{j-1} \right) [Y_2(t_1) Y_2(t_2) - r(Y_1(t_1)) Y_2(t_2)] \end{bmatrix}. \quad (4.5.18)$$

For $t = t_1 = t_2$ making a comparison between the right hand side of eqs.(4.5.17) and (4.5.18) and reminding that in stationary condition $E[Y_1 \dot{Y}_2 H^k] = 0$, from the last two equations the fundamental relationships can be derived:

$$E[\dot{Y}_2(t) Y_1(t)] = -E[r(Y_1) Y_1] \quad (4.5.19)$$

$$E[\dot{Y}_2(t) Y_2(t)] = -E\left[\left(-\sum_{j=1}^k ja_j H^{j-1}\right) Y_2^2(t)\right]. \quad (4.5.20)$$

On the other hand specializing eq.(4.5.13) for $i=1$ results:

$$-\alpha \sum_{j=1}^k ja_j E[H^{j-1} Y_2^2] + \frac{S}{2} = 0 \quad (4.5.21)$$

and then

$$E[\dot{Y}_2(t) Y_2(t)] = -\frac{S}{2} = -\pi k_0. \quad (4.5.22)$$

Attention must be given to eqs.(4.5.19) and (4.5.22); the first gives the relationship existent between the moment response of the structure and the

restoring force parameters while the second links some moment response of the structure with the intensity level of the external excitation which can be unknown in the identification problem.

4.6 Identification of SDOF non-linear systems

4.6.1 Identification of the stiffness parameters

The concepts introduced in the previous sections will be utilized to identify non-linear system belonging to the class of reduced potential systems. The proposed identification procedure consists in successive steps and different cases can be distinguished.

Suppose, for example, that the restoring force is linear and in the classical form:

$$r(Y_1) = f^2 Y_1 \quad (4.6.1.1)$$

where f is the natural frequency of the system; in this case eq.(4.5.19) becomes:

$$E[\dot{Y}_2(t)Y_1(t)] = -f^2 E[Y_1^2] \quad (4.6.1.2)$$

from which the parameter f , directly connected to the stiffness of the system, can be derived if the system response is known in terms of displacement and acceleration.

In a more general case a polynomial form can be assumed for the restoring force:

$$r(Y_1) = \sum_{i=1}^m b_i Y_1^{2i-1} \quad (4.6.1.3)$$

in this case to identify the m stiffness parameters b_i the scalar eq.(4.6.1.2) is not enough.

Other relationships could be derived as in the following. With the position (4.6.1.3) the Itô representation of the system in the form of eq.(4.5.4) can be replaced by:

$$\begin{cases} d\dot{Y}_1 = Y_2 dt \\ d\dot{Y}_2 = -\sum_{j=1}^k ja_j H^{j-1} Y_2 dt - \sum_{i=1}^m b_i Y_1^{2i-1} dt + dB(t). \end{cases} \quad (4.6.1.4)$$

On the other hand if the process $Y_1^{2n-1}Y_2$ is differentiated, using the Itô differential rule and the substitution made in (4.6.1.4), one obtains:

$$\begin{aligned} d(Y_1^{2n-1}Y_2) &= (2n-1)Y_1^{2n-2}Y_2^2 dt - Y_1^{2n-1} \left(\sum_{j=1}^k ja_j H^{j-1} Y_2 \right) dt + \\ &- Y_1^{2n-1} \left(\sum_{i=1}^m b_i Y_1^{2i-1} \right) dt + Y_1^{2n-1} dB(t). \end{aligned} \quad (4.6.1.5)$$

Dividing by dt eq.(4.6.1.5) and taking the expected value results in:

$$(2n-1)E[Y_1^{2n-2}Y_2^2] = E \left[Y_1^{2n-1} \left(\sum_{i=1}^m b_i Y_1^{2i-1} \right) \right] \quad (4.6.1.6)$$

where the *not-anticipative* property of the Wiener process has to be taken into account together with the stationary condition property $E[Y_1^{2n-2}Y_2^2 H^m] = 0 \quad \forall n, m$.

For $n=1, 2, \dots, m$ eq.(4.6.1.6) gives a set of independent equations needed to identify the m b_i parameters.

4.6.2 Identification of the damping parameters

Also for the identification of the damping parameters different situations can be distinguished. If the power level of the white noise input to the system is known, for $i=1,2, \dots, k$ a set of k independent equations in the unknown parameters a_i can be derived from eq.(4.5.13). Otherwise, if the power level of the input is unknown as first step is necessary to identify it. This can be done by means of eq.(4.5.22). This procedure will be explained through the following numerical application.

4.6.3 Numerical application

Let consider the single degree of freedom system:

$$\ddot{X} + (a_1 + a_2 H)\dot{X} + f^2 X = W \quad (4.6.3.1)$$

where W is a white noise having assigned power spectral level; moreover assume for the system parameters the following value:

$$a_1 = 0.02 \text{ sec}^{-1}, \quad a_2 = 0.002 \text{ cm}^2 \text{ sec}, \quad f = 1.4 \text{ sec}^{-1}.$$

Because experimental data are not available, the kinematics of the system will be derived by numerical integration of eq.(4.6.3.1).

Starting from the knowledge of the velocity and the displacement of the system, eq.(4.5.22) is applied to derive the level of the external excitation. Figure (4.1) shows how the correct value is reached asymptotically with increasing sample length. This because we are assuming to work in ergodic condition.

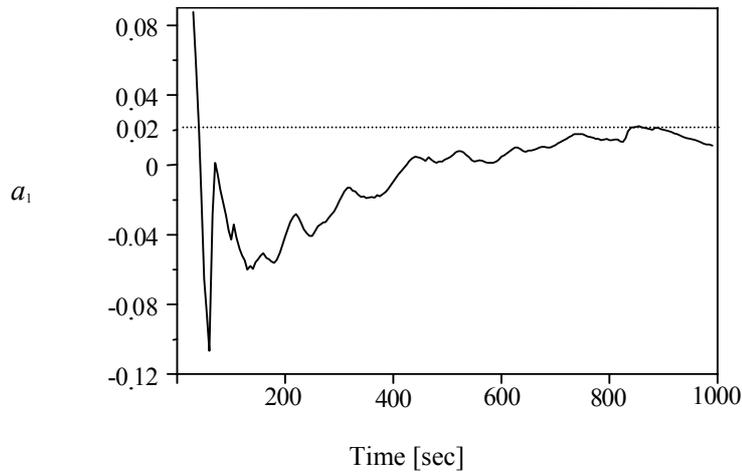


Figure 4.1: Trend of the identified damping parameter.

To identify the stiffness parameter eq.(4.5.19) as well as eq.(4.6.1.6) could be used. From the application of the first equation one obtains the result shown in fig.(4.2), again the correct value is reached asymptotically.

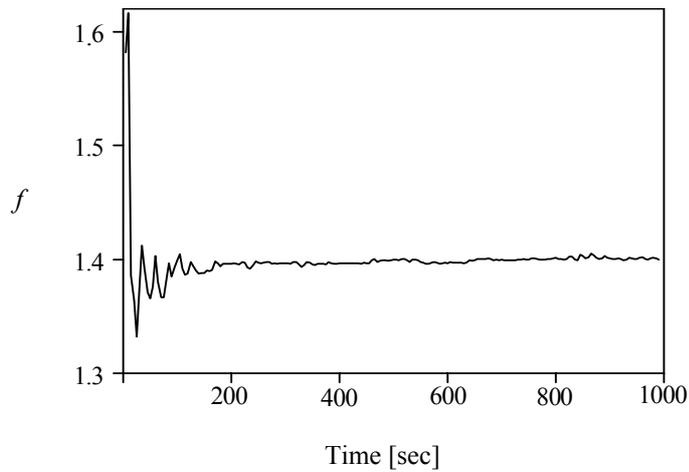


Figure 4.2: Trend of the identified stiffness parameter.

As a second case the shape of the restoring force is assumed polynomial

$$r(Y) = b_1 Y_1 + b_2 Y_1^3 + b_3 Y_1^5 \quad (4.6.3.2)$$

with $b_i (i=1,2,3)$ constant. Then the set of resolving equation becomes:

$$\begin{cases} E[Y_2^2] = b_1 E[Y_1^2] + b_2 E[Y_1^4] + b_3 E[Y_1^6] \\ 3E[Y_1^2 Y_2^2] = b_1 E[Y_1^4] + b_2 E[Y_1^6] + b_3 E[Y_1^8] \\ 5E[Y_1^4 Y_2^2] = b_1 E[Y_1^6] + b_2 E[Y_1^8] + b_3 E[Y_1^{10}] \end{cases} \quad (4.6.3.3)$$

from which the following values for b_1, b_2, b_3 are derived:

$$b_1 = 1.4; \quad b_2 = 0; \quad b_3 = 0.$$

As a second step the dissipation parameters are derived applying eq.(4.5.13) evaluated for $i = 1$ and for $i = 2$.

$$\begin{cases} \alpha (a_1 E[Y_2^2] + 2a_2 E[HY_2^2]) = \frac{S}{2} \\ \alpha (a_1 E[HY_2^2] + 2a_2 E[H^2 Y_2^2]) = \frac{S}{2} E[H](1 + \alpha) \end{cases} \quad (4.6.3.4)$$

from which the system returns $a_1 = 0.017; \quad a_2 = 0.0026$.

Also in this case it can be assumed unknown the number k of the α_i parameters and $m \geq k$ equations can be inserted in the system (4.6.3.4). In this case, as for the stiffness parameters estimation, the system will return only k parameters different from zero.

From this simple example can be seen how versatile the presented system identification technique is.

Moreover some considerations are necessary:

- one limitation of the method lies on the fact that the type of the external excitation must be known (white noise), in order to have a response system modeled as a Markov process;
- the method is valid only for systems belonging to the class of reduced potential system;
- the level of the external excitation, as well as the observed time, influence the accuracy of the procedure. Especially the damping coefficients undergo the influence of two factors as shown in table (4.1).

Time [sec]	a_1	a_2	b_1	k_0
300	-0.020	0.097	1.396	1
600	0.005	0.067	1.396	1
900	0.020	0.022	1.401	1
300	-0.035	0.0062	1.398	10
600	0.022	0.0030	1.397	10
900	0.021	0.0028	1.404	10
300	-0.090	0.0041	1.405	30
600	0.035	0.0032	1.400	30
900	0.017	0.0026	1.400	30

Table 4.1: System parameters estimation.

4.7 Error effected data

As specified in the previous sections, the presented system identification technique needs the knowledge of the input-output signals of the system. As for any dynamic based identification technique, this makes the procedure easily applicable in real case. Nevertheless real signals are influenced by noises.

The reason of this lies either in the adopted equipment for the data acquisition or in the adopted procedure for the data processing. Starting by saying that environmental noises cannot be totally eliminated from real data, it is important to verify that the chosen identification technique is robust, that means that the parameter estimation of the system are slightly effected by noises.

Suppose to have a set of different sensors, then their acquisition is classically modeled by a sum of a real measure (displacement, velocity, acceleration or other) plus an error. The errors associated to different measures are modeled as independent, zero mean, white noise (Jazwinski, 1970).

As a consequence eqs.(4.5.13), (4.5.22) and (4.6.1.6) respectively become:

$$-\alpha \sum_{j=1}^k ja_j E[\tilde{H}^{i+j-2}(Y_2 + e_2)^2] + \frac{s}{2} E[\tilde{H}^{i-1}](1 + \alpha(i-1)) = 0 \quad (4.7.1)$$

$$E[(\dot{Y}_2 + e_3)(Y_2 + e_2)] = -\frac{s}{2} \quad (4.7.2)$$

$$(2n-1)E[(Y_1 + e_1)^{2n-2}(Y_2 + e_2)^2] = E\left[(Y_1 + e_1)^{2n-1} \left(\sum_{i=1}^m b_i (Y_1 + e_1)^{2i-1} \right)\right] \quad (4.7.3)$$

where:

$$\tilde{H} = \frac{1}{2}(Y_2 + e_2)^2 + \frac{1}{2} \sum_{i=1}^m b_i (Y_1 + e_1)^{2i-1} \quad (4.7.4)$$

and e_1, e_2, e_3 are the three independent white noises associated respectively to the displacement, velocity and acceleration of the system. Using eqs.(4.7.1), (4.7.2) and (4.7.3) the previous example is repeated. The intensity of e_1, e_2, e_3 has be chosen equal to the 5% of the maximum value of the associated processes. The results, reported in table (4.2), show how the identification

technique remains efficient. The system parameters are just little influenced by the noise introduced during their estimation.

Time [sec]	a_1		a_2		b_1		k_0
	5% noise	10% noise	5% noise	10% noise	5% noise	10% noise	
300	-0.020	-0.015	0.096	0.084	1.396	1.391	1
600	0.005	0.006	0.066	0.059	1.396	1.390	1
900	0.019	0.020	0.022	0.020	1.402	1.396	1
300	-0.034	-0.027	0.0062	0.0050	1.397	1.3889	10
600	0.022	0.024	0.0030	0.0029	1.395	1.386	10
900	0.025	0.027	0.0027	0.0026	1.402	1.393	10
300	-0.090	-0.075	0.0040	0.0040	1.404	1.395	30
600	0.033	0.025	0.0032	0.0031	1.398	1.392	30
900	0.013	0.010	0.0028	0.0026	1.404	1.398	30

Table 4.2: System parameters estimation with noisily data.

An interesting result comes out from eq.(4.7.2) as in fact:

$$E[(\dot{Y}_2 + e_3)(Y_2 + e_2)] = E[\dot{Y}_2 Y_2] + E[\dot{Y}_2 e_2] + E[e_3 Y_2] + E[e_2 e_3] = \quad (4.7.5)$$

$$E[\dot{Y}_2 Y_2] = -\frac{s}{2}$$

because from the hypotheses of zero mean, independent disturbances follows:

$$\begin{aligned}
E[\dot{Y}_2 e_2] &= E[\dot{Y}_2] E[e_2] = 0 \\
E[Y_2 e_3] &= E[Y_2] E[e_3] = 0 \\
E[e_2 e_3] &= E[e_2] E[e_3] = 0
\end{aligned} \tag{4.7.6}$$

From eq.(4.7.5) follows that the level estimation of the input is not influenced by noise in the measurements.

4.8 Identification of MDOF system

The above described procedure can be extended to the case of multi-degree of freedom system.

Let $\mathbf{X}(t)$ be an n -vector of stochastic processes, obtained by solving the following set of differential equations:

$$\ddot{\mathbf{X}} + \pi \dot{\mathbf{X}} \mathbf{K} \left[\frac{dQ(\mathbf{H})}{d\mathbf{H}} \right] + \mathbf{r}(\mathbf{X}) = \mathbf{W} \tag{4.8.1}$$

being \mathbf{W} the n -vector of white noise processes whose correlation matrix is given in the form:

$$E[\mathbf{W}(t_1) \mathbf{W}(t_2)] = \pi \mathbf{K} \delta(t_1 - t_2) \tag{4.8.2}$$

and $\mathbf{r}(\mathbf{X})$ is a vector of non-linear functions of \mathbf{X} (non-linear stiffness). It will be noted that the damping vector term is related to the strength of the white noise and on the scalar function Q of the total energy of the whole system.

If the differential equations are given in the form (4.8.1), then the stationary probability density function is given in a closed form solution as follows:

$$p_{\dot{x}}(\mathbf{x}, \dot{\mathbf{x}}) = q \exp(-Q(h(\mathbf{x}, \dot{\mathbf{x}}))). \quad (4.8.3)$$

The total energy H of the system has the form:

$$H(\mathbf{X}, \dot{\mathbf{X}}) = \frac{1}{2} \sum_{j=1}^n \dot{x}_j^2 + \int_0^{x_1} \int_0^{x_2} \dots \int_0^{x_n} \mathbf{r}^T(y_1, y_2, \dots, y_n) d\mathbf{y}. \quad (4.8.4)$$

The first step for the identification procedure consists in the approximation of the original non-linear system through the following expression:

$$\ddot{\mathbf{X}} + \mathbf{C}(\dot{\mathbf{X}}) + \mathbf{r}(\mathbf{X}) + \mathbf{f}(\mathbf{X}, \dot{\mathbf{X}}) = \mathbf{W} \quad (4.8.5)$$

with a non-linear system belonging to the class of potential systems given in eq.(4.8.1). In eq.(4.8.5) \mathbf{C} is the damping matrix and $\mathbf{f}(\mathbf{X}, \dot{\mathbf{X}})$ is a non-linear vector. The error $\boldsymbol{\varepsilon}$ obtained in substituting the original system (4.8.5) with the equivalent non-linear system (4.8.1) has the form:

$$\boldsymbol{\varepsilon} = \pi \dot{\mathbf{X}} \mathbf{K} \left[\frac{dQ(\mathbf{H})}{d\mathbf{H}} \right] - \mathbf{C}\dot{\mathbf{X}} - \mathbf{f}(\mathbf{X}, \dot{\mathbf{X}}) \quad (4.8.6)$$

If \mathbf{K} is a non singular matrix, then the appropriate cost function to be minimized in order to find the parameters in the function $Q(\mathbf{H})$ is :

$$E[\varphi(\boldsymbol{\varepsilon})] = E[\boldsymbol{\varepsilon}^T \mathbf{K}^{-1} \boldsymbol{\varepsilon}] \quad (4.8.7)$$

and if $Q(\mathbf{H})$ is given in polynomial form, such as:

$$Q(\mathbf{H}) = \sum_{j=1}^m a_j H^j, \quad (4.8.8)$$

the cost function has to be minimized with respect to the parameters a_j of the function $Q(\mathbf{H})$.

By performing the derivatives of $\varphi(\boldsymbol{\varepsilon})$ with respect to the unknown parameters a_j the following linear system is obtained:

$$\begin{bmatrix} E[\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] & 2E[H\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] & \dots & mE[H^{m-1}\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] \\ 2E[H\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] & 4E[H^2\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] & \dots & 2mE[H^m\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] \\ \dots & \dots & \dots & \dots \\ mE[H^{m-1}\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] & 2mE[H^m\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] & \dots & m^2E[H^{2m-1}\dot{\mathbf{X}}^T \mathbf{K} \dot{\mathbf{X}}] \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_m \end{bmatrix} = \frac{1}{\pi} \begin{bmatrix} E[B] \\ 2E[HB] \\ \dots \\ mE[H^{m-1}B] \end{bmatrix}; \quad (4.8.9)$$

being B equal to:

$$B = [\mathbf{C} \dot{\mathbf{X}} + \mathbf{f}(\mathbf{X}, \dot{\mathbf{X}})]^T \dot{\mathbf{X}}. \quad (4.8.10)$$

In compact form the system (4.8.9) can be written as:

$$\bar{\mathbf{A}} \bar{\mathbf{a}} = \bar{\mathbf{b}}. \quad (4.8.11)$$

At this stage the method of stochastic non-linearization requires a tremendous computation effort, being the procedure iterative since the various entries of the matrix $\bar{\mathbf{A}}$ depend on the unknown parameters a_j . So that starting from a set of coefficients a_j a stochastic average has to be computed by the realtion:

$$E[H^j \dot{X}^T K \dot{X}] = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} h^j \dot{x}^T K \dot{x} dx_1 dx_2 \dots dx_n \quad d\dot{x}_1 d\dot{x}_2 \dots d\dot{x}_n \quad (4.8.12)$$

However a fundamental relationship similar to that of a single degree of freedom system (eq.(4.5.12)) is valid, such as:

$$E[H^j \dot{X}^T K \dot{X}] = \gamma E[H^{j+1}] \quad (4.8.13)$$

Using eq.(4.8.13), eq.(4.8.9) can be written in the form:

$$\gamma \begin{bmatrix} E[H] & 2E[H^2] & \dots & mE[H^m] \\ 2E[H^2] & 4E[H^3] & \dots & 2mE[H^{m+1}] \\ \dots & \dots & \dots & \dots \\ mE[H^m] & 2mE[H^{m+1}] & \dots & m^2E[H^{2m}] \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_m \end{bmatrix} = \bar{\mathbf{b}} \quad (4.8.14)$$

since $H(t)$ is a scalar process, then the various entries of the matrix $\bar{\mathbf{A}}$ can be computed by performing an integral of the type:

$$E[H^k] = \int_0^{\infty} h^k p_H(h) dh \quad (4.8.15)$$

instead of multiple integrals of the type of eq.(3.8.12).

The probability density function $p_H(h)$ can be evaluated in a closed form by means of eq.(4.8.3).

Once the first row of the matrix $\bar{\mathbf{A}}$ is evaluated, all the other rows can be computed by means of the recursive relationship:

$$-\sum_{j=1}^k j a_j \alpha E[H^{i+j-1}] + E[H^{i-1}] tr \mathbf{K} + (i-1) \alpha E[H^{j-2}] = 0. \quad (4.5.9)$$

The identification procedure of a SDOF system, just described in section (4.7), remains still applicable for the case of the MDOF system.

4.9 Conclusion

In this chapter a probabilistic approach to the structural identification problem has been presented. To solve the problem a particular class of potential system has been considered. This class of models is called “reduced potential system” (MPR). The MPR models are characterized by a polynomial potential function and are suitable to model non-linear structure.

For the case of white noise input, applying the rules of the differential stochastic calculus, an identification algorithm has been derived.

Using this algorithm the unknown parameters of the structure (damping and stiffness) have been identified solving sets of algebraic equations involving the statistical moments of some functions related to the response of the structure.

It has also been shown that the presented procedure is slightly effected by the presence of noises in the data.

Chapter 5

Conclusions and Remarks

This thesis has treated the identification problem of structural systems, starting from the knowledge of input-output series functions.

It has been evidenced how the choice of the system identification procedure and of the system model depends on the specific purpose of the identification.

These preliminary choices are essential and effect the final results.

Identification represents the first step to solve classical problems, in the structural engineering field, as control problems, monitoring and damage detection problems and structural reliability problems.

Throughout the thesis three different approaches to the problem have been developed and the obtained results can be summarized as follows:

- for time-invariant, multi-degree of freedom linear systems, the system realization procedure has shown good potential in reproducing the dynamic response of the structure. The method, starting from the Markov parameters of the system, has allowed to obtain a realization of the system with the smallest state-space dimension. Moreover the introduction of an “observer” has ensured the stability of the identification algorithm. Using the identified state-space model, the modal parameters of the structure, which are used for qualitative damage identification, have been

determined. This kind of approach has resulted particularly useful in control applications. To this end the procedure has been implemented together with an assignment control law, to control the modes (under 100 Hz) of a typical space structure. For the purposes of the application, a truss structure equipped with piezoelectric elements has been considered. The piezoelectric elements have had the double function to act as sensors (for the output monitoring) and as actuators (to give the necessary control loads). In order to obtain, from the identified model, outputs having a clear physical meaning a transformation of the state variables has been introduced. The study has outlined the highly effective results that can be achieved through the combined use of system identification and eigenstructure assignment technique.

In further researches some other aspects of the problem could be explored. One aspect regards the possibility to have a systematic approach in transforming the state-space realization model into the corresponding second order structural model. Another point that should be investigated in major details regards the stability of the control algorithm. In fact for complicated structures and for particular combinations of loads, problems, such as spillover, may be present.

- neural networks, appropriately trained, have been successfully used in multi-degree of freedom linear and non-linear systems. It has been shown that a feedforward, multi-layer neural network is able to uniquely associate to a given function, depending on the structural response, informations about the state of the structure.

The transfer functions of the structure have been used as input to the network, being the transfer functions characteristic of the dynamic system behavior. The same choice has been adopted for the damage identification in a simple shear type structure having a non material linear behavior. In this case the ability of the neural network in recognizing the presence and the position of one or more plastic hinges in the structure has been shown.

This information has been used to update the probability of local failure of the structure through fragility curves, independently evaluated via simulation.

This last problem is open for further researches, since the prospective to have an efficient tool to evaluate the probability of failure of a structure in real time is really attractive.

- to give a probabilistic approach to the system identification problem, the class of reduced potential systems has been studied. This class of models has been extracted from the class of stationary potential systems under the hypotheses of non-parametric and Gaussian input and a potential function expressible in polynomial form. It has been shown that, in stationary conditions, the mechanical parameters of the structure can be identified solving a set of algebraic equations whose coefficients are given by the statistical moments of some functions of the response of the structure. The procedure has maintained its validity also when the errors associated to the different data, introduced into the model, have been considered.

References

- ABAQUS (1994). Version 5.4 Hibbit, Karlsson and Sorensen.
- Adeli H. and Hung S. (1995). *Machine Learning. Neural Network, Genetic Algorithms and Fuzzy System*, Diane D. Cerra Ed.
- Alvin K.F. and Park K.C. (1994). "Second Order Structural Identification Procedure via State-Space-Based System Identification", *AIAA Journal*, Vol.32, No.2, pp.397-406.
- Andry A.N., Shapiro E.Y. and Chung J.C. (1983). "Eigenstructure Assignment for Linear Systems", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-19, No.5, pp.711-729.
- Breitung K. (1984). "Asymptotic approximations for multinormal integrals", *J. of Engineering Mechanics, ASCE*. Vol.110, No.3, pp.357-366.
- Carpenter G.A. and Grossberg S. (1988). "The ART of adaptive pattern recognition by a self-organising neural network", *Computer*, March 1988, pp.77-88.
- Casciati F. and Faravelli L. (1991). *Fragility Analysis of Complex Structural Systems*, Research Studies press Ltd. Taunton, U.K.
- Casciati F. and Faravelli L. (1995). "Signal Recognition for Active Structural Control", *Smart Mater. Struct*, Vol.4, pp.9-14.
- Casciati F. and Faravelli L. (1995). "Signal recognition for active structural control", *Smart Mater. Struct*, No.4, pp.9-14.
- Casciati F. et al. (1994). "Some aspects of record processing in seismic structural design", *Proc. of 2-th Int. Conf. on Earthquake Resistant Construction and Design. 15-17 June 1994*. Berlin, Germany, pp.19-26.

- Casciati F., Faravelli L. and Pisano A. (1998). "Damage Detection in Structural Systems Using a Dynamic Method", *Proc. Eur. Conf. on Safety and Reliability (ESREL'98)*, Trondheim, NW, 16-19 June, Vol.2, pp.947-953.
- Caughey T.K. (1971). "Nonlinear Theory of Random Vibration", *Advances in Applied Mechanics*, Vol.11, pp.209-253.
- Cavaleri L. (1998). "L'impiego dei modelli a potenziale nella identificazione strutturale", PhD Thesis, Universita' di Palermo.
- Chen C.H. (1991). *Neural Networks in Pattern Recognition and their Applications*, World Scientific, Singapore.
- Control, Vol.1, No.2, pp.152-192.
- Der Kiureghian A. (1989). "Measures of structural safety under imperfect state of knowledge", *J. of Engineering Mechanics, ASCE*. Vol.115, No.5, pp.1119-1140.
- Ditlevsen O. and Madsen H. (1996). *Structural reliability Methods*. Jhon Wiley, Chichester, U.K.
- Eikhoff P. (1981). *Trends and progress in system identification*, Pergamon press, Elmsford N.Y.
- Faravelli L. (1989). "Response surface approach for reliability analysis", *J. of Engineering Mechanics, ASCE*. Vol.115, No.12, pp.2763-2781.
- Faravelli L. (1994). "Blocking Problems in the Analysis of Random Field", *Probability and Materials*, pp.177-195.
- Faravelli L. and Pisano A. (1997). "A Neural Network Approach to Structure Damage Assessment", *Proc. Int. Conf. on Intelligent Information Systems*, Bahamas 8-10 Dec., pp.585-588.
- Faravelli L., Pisano A., Mengali G. and Pieracci A. (1996). "Use of System Identification Techniques for Damage Assessment and Recovery of Space Truss Structures", *Proc. Of the Seventh Int. Conference on Adaptive Structures*, Rome Sept. 23-25, pp.
- Flood I. and Kartam N. (1994). "Neural Networks in Civil Engineering I: Principles and Understanding", *ASCE J. Comput. In Civil Eng.*, Vol.8, No.2, pp.131-148.

- Funahashi K.I. (1989). "On the Approximate Realization of the Continuous Mappings by Neural Networks", *Neural Networks*, Vol.2, pp.183-192.
- Gardiner, C.W. (1983). *Handbook of Stochastic Methods for Physics, Chemistry and the Natural Sciences*, Springer-Verlag, Berlin.
- Hebb D. (1949). *Organization of Behavior*, New York: Wiley.
- Hecht-Nielsen R. (1987). "Kolmogorov Mapping Neural Networks Existence Theorem", *IEEE Int. Conf. On Neural Networks*, Vol.3, pp.11-23.
- Ho B.L. and Kalman R.E. (1965). "Effective Construction of Linear State-Variable Models from Input / Output data", *Proc. of the 3-th Annual Allerton Conference on Circuit and System Theory*, pp.449-459.
- Hohenbickler M. and Rackwitz R. (1981). "Non normal dependent vectors in structural safety", *J. of Engineering Mechanics, ASCE*. Vol.107, No.6, pp.1227-1249.
- Hopfield J.J. (1982). "Neural Networks and physical systems with emergent collective computational abilities", *Proc. of the National Accademy of Sciences*, No.79, pp.2554-2558.
- Imai H. et al. (1989). "Fundamental on system identification on structural dynamics", *Probabilistic Engineering Mechanics*, Vol.4, No.4, pp.162-173.
- Irie B. and Miyake S. (1988). "Capability of three-layered Perceptron", *IEEE Int. Conf. On Neural Networks*, Vol.1, pp.641-648.
- Isermann R. eds. (1981). "Special Issue on System Identification", *Automatica*, Vol.17, No.1.
- Jazwinski A.H. (1970). *Stochastic processes and filtering theory*, Academic Press, New York.
- Juang J.N. (1992). *Applied System Identification*, PTR Prentice Hall Englewood Cliffs, New Jersey.
- Juang J.N. and Pappa R.S. (1985). "An Eigensystem Realization Algorithm for Modal Parameter Identification and Modal Reduction", *Journal of Guidance, Control and Dynamics*, Vol.8, No.5, pp.620-627.
- Juang J.N. and Pappa R.S. (1986). "Effect of Noise on Modal Parameters Identified by the Eigensystem Realization Algorithm", *Journal of Guidance, Control and Dynamics*, Vol.9, No.3, pp.294-303.

- Juang J.N., Horta L.G. and Phan M. (1992). "System / Observer / Controller Identification Toolbox", NASA Technical Memorandum 107566.
- Kailath T., Mchra R.K. and Mayne D. Q. eds. (1974). "Special Issue on System Identification", *IEEE Trans. Automatic Control*, Vol. AC-19, No.6.
- Kalman R.E. (1963). "Mathematical Description of Linear Dynamic Systems", *SIAM Journal of*
- Kohonen T. (1989). *Self-Organization and Associative Memory*, 3-th eds., Berlin: Springer-Verlag.
- Kozin F. and Natke H.G. (1986). "System Identification Techniques", *Structural Safety*, No.3, pp.269-316.
- Lin Y.K. and Cai G.Q. (1988). "Exact Stationary-Response Solution for Second Order Nonlinear Systems under Parametric and External White-Noise Excitations", *ASME Journal of Applied Mechanics*, Vol.55, No.3, pp.702-705.
- Lin Y.K. and Cai G.Q. (1995). *Probabilistic Structural Dynamics*, McGraw-Hill Book.
- Ljung L. (1987). *System Identification Theory for the user*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- Madsen H., Krenk S. and Lind N.C. (1986). *Method of Structural Safety*. Prentice Hall Inc., Englewood Cliffs, N.Y.
- McCulloch W.S. and Pitts W.H. (1943). "Logical Calculus of the Ideas Immanent in Nervous Activity", *Bull Math. Biophys*, 5, 115.
- Naess A. and Johnsen J.M. (1993). "Response Statistics of Nonlinear, Compliant Offshore Structure by the Path Integral Solution Method", *Probabilistic Engineering Mechanics*, Vol.8, No.2, pp.91-106.
- Natke H.G. and Cempel C. (1997). *Model-Aided Diagnosis of Mechanical Systems*, Springer-Verlag, Berlin Heidelberg.
- Oja E. (1982). "A simplified neural model as a principal component analyzer", *J. Math. Biol.*, Vol.15, pp.267-273.
- Pham D.T. and Oztemel E. (1994). "Control chart pattern recognition using learning vector quantization networks", *Int. J. Production Research*, Vol.32, No.3, pp.721-729.

- Phan M. et al. (1992). "Linear System Identification via an Asymptotically Stable Observer", NASA technical paper 3164.
- Povich C.R. and Lim T.W. (1994). "An Artificial Neural Network Approach to Structural Damage Detection Using Frequency Response Functions", *AIAA Paper 94-1751-CP*.
- Rogers J.L. (1994). "Simulating Structural Analysis with Neural Networks", *ASCE J. Comput. In Civil Eng.*, Vol.8, No.2, pp.252-265.
- Rosenblatt R. (1962). *Principles of Neurodynamics*, Washington DC: Spartan Books.
- Rumelhart D.E., Hinton G.E. and Williams R.J. (1986). "Learning Internal Representations by Error Propagation", *Parallel Data Processing*, Rumelhart D. and McClelland Eds., MIT Press, Cambridge, Vol.1, pp.318-362.
- Sanger T. (1989). "Optimal unsupervised learning in a simple-layer linear feedforward neural network", *Neural Networks*, Vol.2, pp.459-473.
- Simpson P.K. (1996). "Foundations of Neural Networks", IEEE Technology Update Series, Simpson P.K. Ed., pp.1-23.
- Spencer B.F. and Bergman L.A. (1993). "On the Numerical Solution of the Fokker-Planck Equation for Nonlinear Stochastic Systems", *Nonlinear Dynamics*, Vol.4, pp.357-372.
- Srinathkumar S. (1978). "Eigenvalue-Eigenvector Assignment Using Output Feedback", *IEEE Transactions Automatic Control*, AC-23, No.1, pp.79-82.
- Stratonovich R.L. (1963). *Topics in the Theory of random Noise*, Vol.1 Gordon and Breach, N.Y.
- Tank D.W. and Hopfield J.J. (1986). "Simple Neural Optimization Networks: an AID converter, signal decision circuit and ", *IEEE Trans.Circuit Syst.*, CAS Vol.33, No.5, pp.533-541.
- To C.W.S and Li D.M. (1991). "Equivalent nonlinearization of nonlinear systems to random excitations", *Probabilistic Engineering Mechanics*, Part2. Vol.6, Nos.3 and 4, pp.184-192.

- Volg. T.P. et al. (1988). "Accelerating the Convergence of the Back-Propagation Method", *Biological Cybernetics*, No.59, pp.257-263.
- Widrom B. and Hoff M.E. (1960). "Adaptive switching circuits", IRE WESCON Convention Record, No.4, pp.96-104.
- Wu X., Gabussi and Garrett.(1992). "Use of Neural Networks in Detecting of Structural Damage", *Computer & Structures*, Vol.42, No.4, pp.649-659.
- Zak S.H. et al. (1990). "Control of Dynamic Systems via Neural Networks", *Neural Networks: Concepts, Applications and Implementations*, Antognetti P. et al. Eds. Prentice-Hall, Englewood Cliffs N.J. Vol.II.
- Zeiger H.P. and McEwen A.J. (1997). "Approximate Linear Realization of Given Dimension via Ho's Algorithm", *IEEE Transactions Automatic Control*, AC-19, No.2, pp.153- .
- Zeng P. (1998). "Neural Computing in Mechanics", *Applied Mech. Rev.*, Vol.51, No.2, pp.173-197.

Appendices

Appendix 2.A

2.1 A The Mode Singular Value (MSV)

As specified in section (2.2.3), for a linear system, the map from input \mathbf{u} to output \mathbf{y} can be fully described by the Markov parameter sequence:

$$\mathbf{Y} = [\mathbf{D} \quad \mathbf{CB} \quad \mathbf{CAB} \quad \mathbf{CA}^{l-2}\mathbf{B}] \quad (2.1.1 \text{ A})$$

where l is the number of Markov parameters and \mathbf{A} the diagonal eigenvalue matrix. This sequence is co-ordinate independent and unique.

Let the input and output matrices be partitioned as:

$$\mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{bmatrix}; \quad \mathbf{C} = [c_1 \quad c_2 \quad \dots \quad c_n] \quad (2.1.2 \text{ A})$$

where n is the number of modal co-ordinates. Each Markov parameter can then be written as a linear combination of n components each of one gives the contribute of a different mode, for example:

$$\mathbf{Y}(2) = \mathbf{C}\mathbf{A}\mathbf{B} = \sum_{i=1}^n c_i \lambda_i b_i . \quad (2.1.3 \text{ A})$$

Therefore each co-ordinate has a sequence of Markov parameters described as following:

$$\mathbf{Y}(i) = [c_i b_i \quad c_i \lambda_i b_i \quad \dots \quad c_i \lambda_i^{l-2} b_i] \quad i=1,2,\dots,n; \quad (2.1.4 \text{ A})$$

and the total Markov parameter sequence becomes:

$$\mathbf{Y} = \left[\mathbf{D} \sum_{i=1}^n \mathbf{Y}(i) \right] \quad i=1,2,\dots,n . \quad (2.1.5 \text{ A})$$

From the representation (2.1.5 A) it is immediate to understand that each modal co-ordinate contributes to the pulse response sample by the individual modal sequence $\mathbf{Y}(i)$, which can be quantified by taking its maximum singular value, i.e.:

$$\begin{aligned} \text{MSV}_i &= \left\{ |c_i| \left(1 + |\lambda_i| + |\lambda_i|^2 + \dots + |\lambda_i^{l-2}| \right) |b_i| \right\}^{1/2} = \\ &\cong \sqrt{\frac{|c_i| |b_i|}{(1 - |\lambda_i|)}} \end{aligned} \quad (2.1.6 \text{ A})$$

where the approximation is valid if $|\lambda_i|$ is less than 1 and l is sufficiently large.

The MSV is a method of characterizing the contribution of each identified mode to the identified model pulse response history. And in fact, is reasonable that a mode that has larger contribution to the identified model's pulse response has a large contribution to the system's pulse response.

2.2 A The Modal Amplitude Coherence (MAC)

Define the sequence:

$$\mathbf{q}_i = [b_i \quad \lambda_i b_i \quad \dots \quad \lambda_i^{l-2} b_i] \quad i = 1, 2, \dots, n; \quad (2.2.1 A)$$

which represents the time history reconstructed from the identified eigenvalue λ_i and the row vector b_i . The sequence \mathbf{q}_i is called the modal amplitude time history of the i -th mode because it represents the temporal contribution of the i -th mode associated with the output matrix c_i to the Markov parameter sequence \mathbf{Y} .

The MAC factor is defined as in the following:

$$\text{MAC}_i = \frac{|\bar{\mathbf{q}}_i \mathbf{q}_i^*|}{\sqrt{|\bar{\mathbf{q}}_i \bar{\mathbf{q}}_i| |\mathbf{q}_i^* \mathbf{q}_i^*|}} \quad (2.2.2 A)$$

where $\bar{\mathbf{q}}_i$ has the same meaning of \mathbf{q}_i but is calculated in presence of noise data. The apex (*) indicates the complex conjugate. If MAC_i is equal to one then the two vectors $\bar{\mathbf{q}}_i$ and \mathbf{q}_i coincide and the model reproduces the pulse response data.

Appendix 2.B

2.1 B The Markov parameters of the system

The Markov parameters of the system can be derived using the spectral analysis. Given pairs of input-output signals, $\mathbf{u}(t)$ and $\mathbf{y}(t)$, the auto spectral and cross spectral densities functions can be derived using the Fourier transform.

The one-side spectral density functions have the following form:

$$\begin{aligned} G_{u_i y_j}(f_k) &= \frac{2}{N} E[\mathbf{U}_i(f_k, N)^* \mathbf{Y}_j(f_k, N)] \\ G_{u_i u_j}(f_k) &= \frac{2}{N} E[\mathbf{U}_i(f_k, N)^* \mathbf{U}_j(f_k, N)] \\ G_{y_i y_j}(f_k) &= \frac{2}{N} E[\mathbf{Y}_i(f_k, N)^* \mathbf{Y}_j(f_k, N)] \end{aligned} \quad (2.1.1 B)$$

where the operator E denotes averaging of the spectral densities over multiple test trials and $\mathbf{U}_i(f_k, N)$ and $\mathbf{Y}_i(f_k, N)$ are the discrete Fourier transforms of $\mathbf{u}_i(t)$ and $\mathbf{y}_i(t)$. The apex (*) indicates the complex conjugate.

From the auto and cross spectral density functions the frequency response functions (FRFs) and the coherence functions are computed as:

$$H_{u_i y_j}(f_k) = \frac{G_{u_i y_j}(f_k)}{G_{u_i u_j}(f_k)} \quad (2.1.2 B)$$

$$\gamma^2_{u_i y_j}(f_k) = \frac{|G_{u_i y_j}(f_k)|^2}{G_{u_i u_j}(f_k) G_{y_i y_j}(f_k)}. \quad (2.1.3 B)$$

The model estimation problem may be solved either in the frequency domain, using the discrete FRFs as defined above, or in the time domain. In this last case, the FRFs must be transformed using an inverse Fourier transformer into the discrete impulse response functions, otherwise known as Markov parameters.

The Markov parameters are then given by:

$$\mathbf{h}(t_n) = IFFT(\mathbf{H}(j\omega_k)) \quad (2.1.4 B)$$

where *IFFT* symbolizes the inverse of the discrete fast Fourier transform as implemented using the *FFT*-based algorithm.

Appendix 2.C

2.1 C The Kalman Filter

Recalling the notations introduced in section (2.4) a dynamic system with noise may be mathematically described, in the discrete time domain, by the following equations:

$$\mathbf{x}(k+1) = \mathbf{A} \mathbf{x}(k) + \mathbf{B} \mathbf{u}(k) + \mathbf{e}_1(k) \quad (2.1.1 \text{ C})$$

$$\mathbf{y}(k) = \mathbf{C} \mathbf{x}(k) + \mathbf{D} \mathbf{u}(k) + \mathbf{e}_2(k) \quad (2.1.2 \text{ C})$$

while the estimated state can be described by the following Kalman filter:

$$\hat{\mathbf{x}}(k+1) = [\mathbf{A} - \mathbf{K}\mathbf{C}] \hat{\mathbf{x}}(k) + [\mathbf{B} - \mathbf{K}\mathbf{D}] \mathbf{u}(k) + \mathbf{K} \mathbf{y}(k) \quad (2.1.3 \text{ C})$$

$$\mathbf{y}(k) = \mathbf{C} \hat{\mathbf{x}}(k) + \mathbf{D} \mathbf{u}(k) + \boldsymbol{\varepsilon}(k), \quad (2.1.4 \text{ C})$$

where the present symbols have the meaning already introduced in section (2.4).

The state estimation error is defined as the difference between the real and the estimated state, such as:

$$\mathbf{e}(k) = \mathbf{x}(k) - \hat{\mathbf{x}}(k) \quad (2.1.5 \text{ C})$$

and has to be minimized to give an optimal estimation of $\mathbf{x}(k)$, given $\mathbf{y}(k)$ and $\mathbf{u}(k)$.

The estimation error at time $k+1$ can be obtained, combining eqs.(2.1.1 C-2.1.4 C) and the residual $\mathbf{e}(k) = \mathbf{y}(k) - \hat{\mathbf{y}}(k)$, by the expression:

$$\mathbf{e}(k+1) = \mathbf{x}(k+1) - \hat{\mathbf{x}}(k+1) = [\mathbf{A} - \mathbf{K}(k)\mathbf{C}] - \mathbf{K}\mathbf{e}_2(k) + \mathbf{e}_1(k). \quad (2.1.6 C)$$

From eq.(2.1.6 C) derives that the error $\mathbf{e}(k+1)$ at the time step $k+1$ depends on the previous error $\mathbf{e}(k)$ and on the noises $\mathbf{e}_1(k)$ and $\mathbf{e}_2(k)$. However, under the assumption of independence between the current error and the current noises and the assumption of independence between the noises, the following relationships hold:

$$E[\mathbf{e}(k) \mathbf{e}_1^T(k)] = 0; \quad E[\mathbf{e}(k) \mathbf{e}_2^T(k)] = 0; \quad E[\mathbf{e}_1(k) \mathbf{e}_2^T(k)] = 0; \quad (2.1.7 C)$$

Taking the expected value of eq.(2.1.6 C) yields:

$$E[\mathbf{e}(k+1)] = [\mathbf{A} - \mathbf{K}(k)\mathbf{C}]E[\mathbf{e}(k)] \quad (2.1.8 C)$$

with $\mathbf{e}_1(k)$ and $\mathbf{e}_2(k)$ having zero mean.

Equation (2.1.8 C) allows to write $E[\mathbf{e}(k)]$ as function of the expected value in previous steps:

$$E[\mathbf{e}(k)] = [\mathbf{A} - \mathbf{K}(k-1)\mathbf{C}][\mathbf{A} - \mathbf{K}(k-2)\mathbf{C}] \dots [\mathbf{A} - \mathbf{K}(0)\mathbf{C}]E[\mathbf{e}(0)] \quad (2.1.9 C)$$

If now is assumed a zero initial condition, such as $\mathbf{x}(0) = \hat{\mathbf{x}}(0) = 0$ eq.(2.1.9 C) returns $E[\mathbf{e}(0)] = 0$, that means that the estimation error has zero mean.

Starting from eq.(2.1.8 C) the error covariance dynamic can be derived as:

$$\begin{aligned}
\mathbf{P}(k+1) &= \mathbf{E}[\mathbf{e}(k+1)\mathbf{e}^T(k+1)] = \\
& [\mathbf{A} - \mathbf{K}(k)\mathbf{C}] \mathbf{E}[\mathbf{e}(k)\mathbf{e}^T(k)] [\mathbf{A} - \mathbf{K}(k)\mathbf{C}]^T + \\
& + \mathbf{K}(k) \mathbf{E}[\mathbf{e}_2(k)\mathbf{e}_2^T(k)] \mathbf{K}^T(k) + \mathbf{E}[\mathbf{e}_1(k)\mathbf{e}_1^T(k)].
\end{aligned} \tag{2.1.10 C}$$

Define the covariances of the noises as \mathbf{E}_1 and \mathbf{E}_2 , then the above equation assumes the form:

$$\mathbf{P}(k+1) = [\mathbf{A} - \mathbf{K}(k)\mathbf{C}] \mathbf{P}(k) [\mathbf{A} - \mathbf{K}(k)\mathbf{C}]^T + \mathbf{K}(k) \mathbf{E}_2 \mathbf{K}^T(k) + \mathbf{E}_1. \tag{2.1.11 C}$$

In order to obtain an optimal estimation for the state, one criterion for choosing $\mathbf{K}(k)$ is to minimize the expected value of the squared norm of $\mathbf{e}(k)$.

$$J_k = \mathbf{E}[\mathbf{e}^T(k)\mathbf{e}(k)] = \text{trace}[\mathbf{P}(k)] \tag{2.1.12 C}$$

To obtain the minimum of J_k with respect to $\mathbf{K}(k)$ the following equality must be satisfied:

$$\frac{\partial [\text{trace}[\mathbf{P}(k+1)]]}{\partial \mathbf{K}(k)} = -2 [\mathbf{A} - \mathbf{K}(k)\mathbf{C}] \mathbf{P}(k) \mathbf{C}^T + 2 \mathbf{K}(k) \mathbf{E}_2 = 0 \tag{2.1.13 C}$$

Solving with respect to $\mathbf{K}(k)$ one obtains:

$$\mathbf{K}(k) = \mathbf{A} \mathbf{P}(k) \mathbf{C}^T [\mathbf{R} + \mathbf{C} \mathbf{P}(k) \mathbf{C}^T]^{-1} \tag{2.1.14 C}$$

that gives the Kalman filter gain matrix.

2.2 C The Riccati equation

Substituting eq.(2.1.14 C) into (2.1.11 C) and observing that $\mathbf{P}(k) = \mathbf{P}(k)^T$, the following relationship can be derived:

$$\mathbf{P}(k+1) = \mathbf{A}\mathbf{P}(k)\mathbf{A}^T - \mathbf{A}\mathbf{P}(k)\mathbf{C}^T [\mathbf{E}_2 + \mathbf{C}\mathbf{P}(k)\mathbf{C}^T]^{-1} \mathbf{C}\mathbf{P}(k)\mathbf{A}^T + \mathbf{E}_1. \quad (2.2.1 \text{ C})$$

known as discrete Riccati equation.

If the time-variant error covariance $\mathbf{P}(k)$ reaches a steady value then can be considered a constant \mathbf{P} . In this case the Riccati equation is an algebraic equation given by:

$$\mathbf{P} = \mathbf{A}\mathbf{P}\mathbf{A}^T - \mathbf{A}\mathbf{P}\mathbf{C}^T [\mathbf{E}_2 + \mathbf{C}\mathbf{P}\mathbf{C}^T]^{-1} \mathbf{C}\mathbf{P}\mathbf{A}^T + \mathbf{E}_1. \quad (2.2.2 \text{ C})$$

Here the steady state Kalman filter gain matrix becomes constant.

Appendix 2.D

State matrix **A**, undamaged condition:

0.9956	0.0550	-0.0067	-0.0021	-0.0007	-0.0001	0.0003	-0.0006	0.0000	0.0002	0.0000	0.0001	0.0000	0.0000
-0.0674	0.9979	-0.0074	-0.0421	0.0053	-0.0062	-0.0001	-0.0002	0.0012	0.0006	0.0001	0.0002	0.0002	-0.0001
0.0057	0.0366	0.9508	0.3184	-0.0096	0.0132	0.0000	0.0013	-0.0023	-0.0015	-0.0002	-0.0005	-0.0004	0.0003
0.0111	0.0416	-0.3307	0.9374	0.0099	-0.0072	-0.0018	0.0029	0.0013	-0.0003	0.0002	-0.0003	0.0001	0.0000
-0.0166	-0.0200	0.0008	0.0278	0.6930	0.6863	-0.0003	0.0026	-0.0074	-0.0046	-0.0006	-0.0015	-0.0014	0.0009
-0.0035	-0.0044	-0.0255	0.0177	-0.7135	0.6986	-0.0197	0.0320	0.0063	-0.0038	0.0019	-0.0040	0.0002	0.0006
0.0128	0.0031	-0.0113	-0.0146	0.0167	0.0171	0.6234	0.7751	-0.0360	-0.0215	-0.0030	-0.0067	-0.0066	0.0037
0.0028	0.0062	0.0032	-0.0105	0.0282	-0.0186	-0.7812	0.6137	-0.0916	-0.0429	-0.0096	-0.0087	-0.0147	0.0081
0.0099	0.0084	-0.0007	-0.0158	0.0209	0.0289	-0.0504	0.0734	0.6113	0.7591	-0.0455	0.1531	0.0172	-0.0278
-0.0139	0.0012	0.0135	0.0118	-0.0156	-0.0244	0.0215	-0.0275	-0.7574	0.6457	0.0585	-0.0562	0.0158	0.0027
-0.0018	0.0000	0.0020	-0.0008	-0.0018	-0.0062	-0.0075	0.0124	-0.0085	-0.0435	0.8515	0.5035	0.0327	-0.0293
-0.0011	-0.0049	-0.0038	0.0060	-0.0075	-0.0012	0.0116	-0.0045	-0.1514	-0.0752	-0.5068	0.8290	0.1641	-0.0711
0.0141	0.0025	-0.0122	-0.0144	0.0113	0.0303	-0.0070	0.0018	-0.0005	-0.0245	0.0439	-0.1661	0.6968	-0.6889
0.0068	-0.0144	-0.0163	0.0055	-0.0045	0.0066	0.0049	0.0135	0.0369	-0.0062	0.0293	-0.0547	0.7109	0.7154

State matrix A damaged condition (case 1):

135

0.9956	0.0549	-0.0063	-0.0025	0.0000	0.0001	-0.0006	0.0008	0.0000	0.0002	0.0000	0.0003	0.0000	0.0000
-0.0676	0.9980	-0.0043	-0.0437	0.0073	-0.0071	-0.0001	0.0002	0.0003	0.0000	0.0015	0.0007	0.0000	0.0000
0.0064	0.0323	0.9483	0.3203	-0.0127	0.0148	0.0006	-0.0016	-0.0008	-0.0003	-0.0027	-0.0017	-0.0001	-0.0001
0.0125	0.0424	-0.3331	0.9378	0.0056	-0.0081	0.0034	-0.0044	0.0005	-0.0009	0.0012	-0.0007	0.0000	0.0000
-0.0115	-0.0331	-0.0129	0.0401	0.6814	0.7082	0.0050	-0.0033	-0.0011	-0.0010	-0.0032	-0.0026	-0.0002	-0.0002
-0.0063	0.0133	-0.0149	0.0025	-0.7310	0.6808	0.0406	-0.0507	0.0031	-0.0099	0.0027	-0.0093	-0.0001	-0.0004
-0.0109	-0.0108	0.0027	0.0186	-0.0276	-0.0178	0.6488	0.7480	0.0077	0.0052	0.0199	0.0108	0.0010	0.0009
-0.0014	-0.0096	-0.0070	0.0157	-0.0390	0.0436	-0.7559	0.6433	0.0444	-0.0027	0.0958	0.0422	0.0050	0.0034
0.0003	0.0003	-0.0008	-0.0019	-0.0013	-0.0022	0.0227	-0.0362	0.8745	0.4689	0.0129	0.0395	0.0004	0.0028
-0.0002	-0.0028	-0.0014	0.0029	-0.0057	0.0064	-0.0100	-0.0017	-0.4788	0.8637	0.1525	0.0602	0.0079	0.0038
0.0051	0.0063	0.0034	-0.0106	0.0118	0.0136	0.0644	-0.0689	0.0326	-0.1469	0.6486	0.7360	0.0066	0.0080
-0.0020	0.0075	0.0040	-0.0044	-0.0145	0.0068	-0.0255	0.0340	-0.0507	0.0546	-0.7391	0.6698	0.0082	0.0068
-0.0070	-0.0001	0.0051	0.0040	-0.0162	-0.0164	0.0000	-0.0169	-0.0038	-0.0190	-0.0032	-0.0115	0.6483	0.6890
-0.0034	0.0022	0.0047	-0.0003	-0.0073	-0.0058	-0.0009	-0.0049	-0.0078	-0.0018	0.0027	0.0056	-0.6705	0.6930

Matrix **B**, undamaged case:

1.0e-004*				
0.0000	0.2218	0.0057	0.0018	
0.0000	0.2205	0.0017	-0.0004	
0.0001	-0.1547	0.0207	0.0146	
-0.0001	-0.3040	0.0199	0.0168	
-0.0004	0.5447	0.0861	0.0263	
0.0013	0.1767	0.0281	0.0154	
0.0022	-0.2763	-0.0334	0.0553	
-0.0005	-0.1089	-0.0146	-0.0357	
-0.0081	-0.2909	-0.0662	0.0299	
0.0049	0.2614	0.0400	-0.0154	
0.1056	0.0256	0.3099	0.1834	
-0.0096	0.0731	-0.0227	-0.0257	
0.0162	-0.3042	0.0557	0.0244	
0.0064	0.0138	0.0085	0.0027	

Matrix **B**, damaged case:

1.0e-004*				
0.0000	0.2221	0.0057	0.0018	
0.0000	0.2224	0.0012	-0.0010	
0.0002	-0.1479	0.0209	0.0155	
-0.0008	-0.3057	0.0173	0.0171	
-0.0021	0.5637	0.0861	0.0433	
0.0049	0.0771	0.0115	0.0146	
-0.0020	0.3180	0.1031	-0.0308	
0.0069	0.1113	0.0079	-0.0054	
0.1026	-0.0183	0.3494	0.0798	
0.0312	0.0265	0.0931	0.0151	
0.0313	-0.1783	-0.0378	0.0230	
-0.0050	-0.0371	0.0432	0.0268	
-0.0750	0.1339	0.3414	-0.0540	
-0.0221	0.0420	0.0985	-0.0706	

Matrix C, undamaged case:

0.0078	-0.0083	0.0027	-0.0013	0.0001	-0.0002	0.0000	0.0001	0.0000	-0.0001	0.0000	0.0000	0.0000	0.0000
0.0283	-0.0261	-0.0084	0.0016	0.0005	-0.0002	-0.0001	0.0001	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000
0.0006	-0.0008	0.0007	-0.0003	0.0003	-0.0001	-0.0001	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0047	-0.0052	0.0026	-0.0011	0.0005	-0.0003	-0.0001	0.0001	0.0001	-0.0001	0.0000	0.0000	0.0000	0.0000
0.0197	-0.0189	-0.0024	0.0000	-0.0006	0.0001	0.0000	-0.0001	-0.0001	0.0000	0.0000	0.0000	0.0000	0.0000
0.0154	-0.0153	0.0001	-0.0007	-0.0007	0.0001	0.0001	-0.0001	-0.0001	0.0000	0.0000	0.0000	0.0000	0.0000

Matrix C, damaged case:

0.0006	-0.0008	0.0007	-0.0003	0.0002	-0.0002	0.0001	-0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0047	-0.0052	0.0026	-0.0011	0.0004	-0.0004	0.0001	-0.0002	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000
0.0078	-0.0083	0.0027	-0.0013	0.0000	-0.0002	0.0000	-0.0001	0.0000	0.0000	0.0001	-0.0001	0.0000	0.0000
0.0284	-0.0261	-0.0085	0.0013	0.0004	-0.0004	0.0002	-0.0001	0.0000	0.0000	0.0001	0.0001	0.0000	0.0000
0.0197	-0.0189	-0.0024	-0.0002	-0.0006	0.0001	-0.0001	0.0001	0.0000	0.0000	-0.0001	0.0000	0.0000	0.0000
0.0154	-0.0153	0.0001	-0.0008	-0.0006	0.0002	-0.0002	0.0001	0.0000	0.0000	-0.0001	-0.0001	0.0000	0.0000

Matrix S

1.0e+010 *					
-0.8423	-1.3260	1.8624	0.0485	0.2095	-0.8638
1.9458	3.0954	-4.3661	-0.1253	-0.4686	2.0274
0.1922	0.3309	-0.4816	-0.0241	-0.0360	0.223

Appendix 3.A

3.1 A Foundations of Neural Networks

The perceptron is one of the early attempts to build intelligent and self-learning systems using simple components. It was derived from the brain neuron and introduced by McCulloch-Pitts (1943).

Figure (3.1 A) shows a typical structure of a perceptron, modeled by the following mathematical equation:

$$y_i = 1\left(\sum w_{ij}x_j + s_i - \theta_i\right). \quad (3.1.1 A)$$

In eq.(3.1.1 A) $1(u)$ represents the unit step function having value 1 if $u > 1$ and zero otherwise, x_i is the input signal coming from the output of another neuron, s_i represents the outside input, w_{ij} are the connection weights, u_i denotes the internal state of the neuron, θ_i the threshold value, finally f_i and y_i are respectively the activation function and the neuron output.

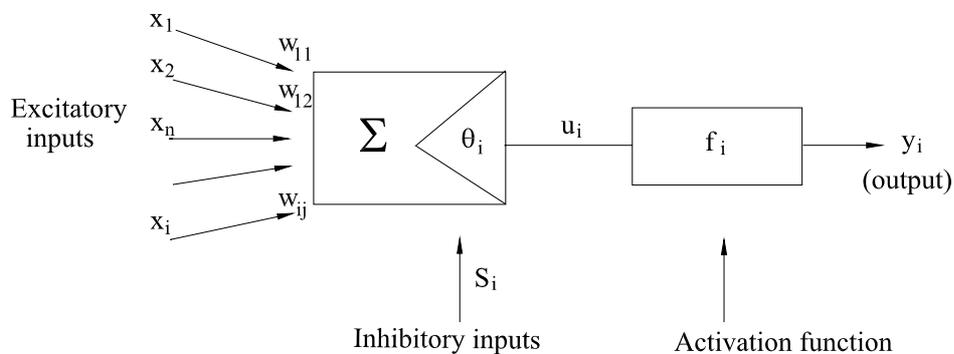


Figure 3.1 A: The McCulloch-Pitts neuron model.

The connection weights are adjusted during the learning process. Connection weights that are positive valued are excitatory while those with negative values are inhibitory. If a connection weight is zero this is equivalent as not having a connection present.

The McCulloch-Pitts model is a discrete model and the neuron output can have only the states 0 or 1. A development of this model is given by the Hopfield (1982) neuron having a time dependent activation function.

The activation function has the specific purpose to map the neuron output from a possible infinite domain to a pre-specified range. Theoretically activation functions can have any form, but usually the ones employed by the majority of neural networks are: linear functions, step and ramp functions, sigmoids, logsigmoids and Gaussian functions. Apart the linear function all the others introduce a non-linearity in the network. Figure (3.2 A) reports the above mentioned functions.

Because a single neuron can act only in a simple way, the power of neural network computation comes from connecting neurons into networks. Neurons with simple properties and interacting according to simple rules can perform complex functions. This is the reason why the neural networks are also defined as massively parallel devices. It has been shown that the way in which the neurons are connected and the number of layers can influence the capability of a network as well as its stability and convergence.

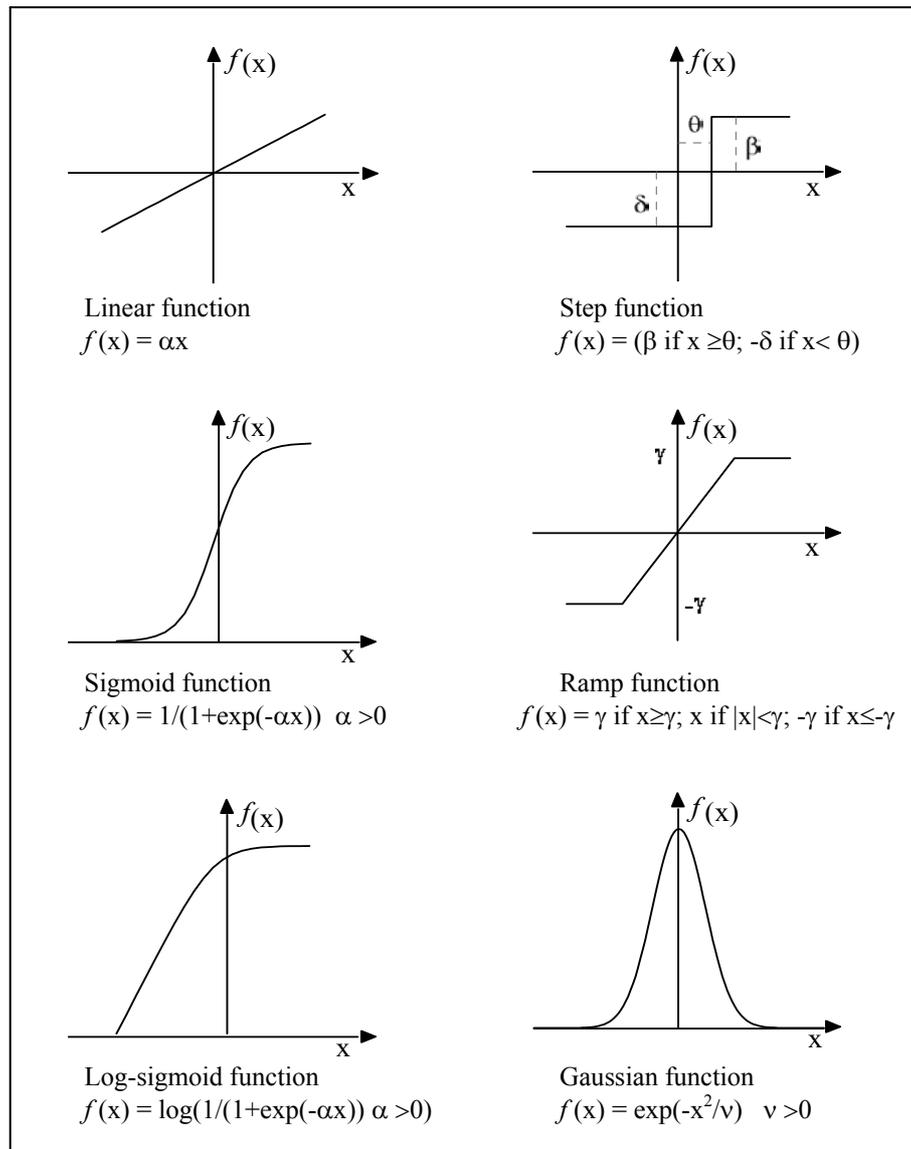


Figure 3.2 A: Example of activation functions.

3.2 A Schemes of Neural Networks

In the following a brief review of the networks commonly used in practical applications is reported.

Hopfield network: this is a typical recurrent network having signal layer of neurons, each connected to all the others. The weights of the network are assigned as follows:

$$w_{ij} = \begin{cases} \frac{1}{N} \sum_{c=1}^P x_i^c x_j^c & i \neq j \\ 0 & i = j \end{cases} \quad (3.2.1 A)$$

where x_i^c is the i -th component of the training input pattern for class c , P the number of classes and N the number of neurons, x_i^c can assume value $+1$ or -1 .

Figure (3.3 A) shows a Hopfield network.

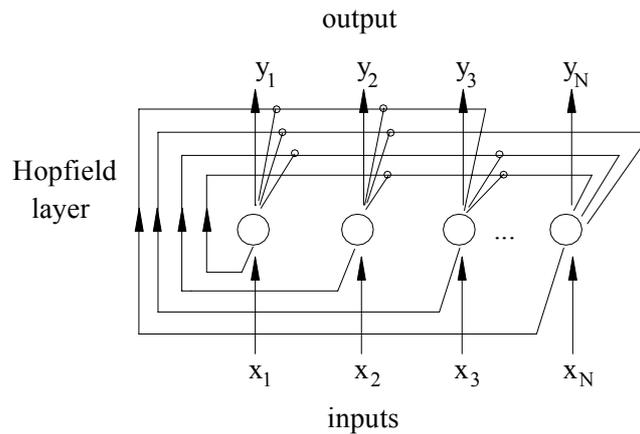


Figure 3.3 A A Hopfield network.

Starting from an initial condition given by the inputs, the network produces an output which is then feedback to become the new input. The network stops

its iterations when the output vector converges to one of the design equilibrium point vectors by means of the following law:

$$y_i(k+1) = f \left[\sum_{j=1}^N w_{ij} y_j(k) \right] \quad 1 < i < N \quad (3.2.2 A)$$

being $f(x)$ a step function defined assuming value 1 if $f(x)$ is greater than zero, -1 otherwise.

Elman and Jordan network: both these networks are feedback networks due to the presence of an additional hidden layer called *context* or *state* layer that receives feedback signals. Figures (3.4 A) and (3.5 A) show the scheme of an Elman net and a Jordan net respectively. As is evident from these figures, the differences between the two networks lies in the fact that in the Elman net the feedback signal comes from an ordinary hidden layer, while in a Jordan net the feedback signal comes from the output layer. With both nets, the outputs of neurons in the state layer are feedforward the hidden layer

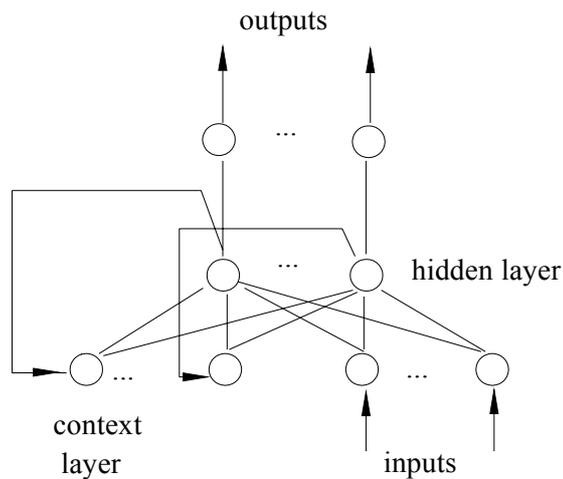


Figure 3.4 A: Scheme of an Elman network.

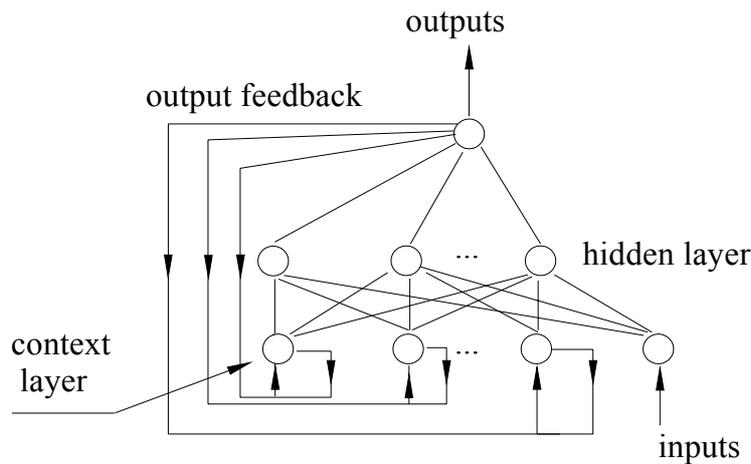


Figure 3.5 A: Scheme of a Jordan network.

The recurrent connection allows the Elman network to both detect and generate time varying patterns, the only requirement is that the hidden layer must have enough neurons.

A sufficient number of neurons can reproduce any complex function.

Kohonen network: this is a self organizing or competitive network. The neurons learn to recognize groups of similar input vectors in such a way that neurons physically close together in the neuron layer respond to similar input vectors. The output vector can contain only 1 or 0 values. To determine the winner neuron the Euclidean distance between a reference vector and the input vector is used. The reference vector contains all the weights of the connections with the given output neuron. The winning output neuron is the one (or more than one) whose reference vector is closest to the input pattern. Only the winning neuron's weights get updated. Thus the neuron whose weight vectors are closest to the input vector is updated to be even closer. As a result the winning neuron is more likely to win the competition the next time a similar vector is presented and less likely to win when a very different input vector is presented. A drawback of this logic is that some neurons may not always be activated, some neuron weight vectors may start so far from any input vectors

and never win the competition. This problem has been overcome in advanced competitive networks, using biases to give neurons which are rarely winning an advantage over neurons which win very often. Figure (3.6 A) shows a scheme of a Kohonen network.

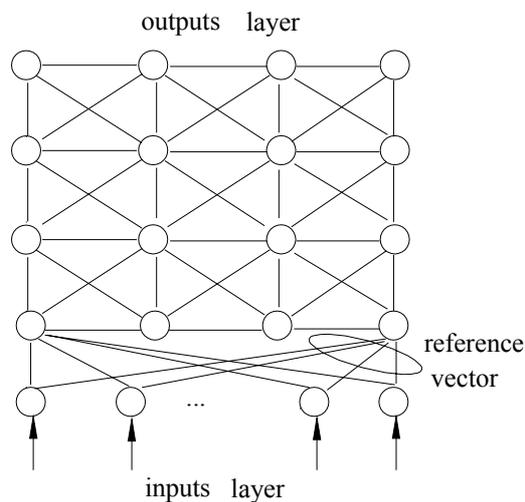


Figure 3.6 A: Scheme of a Kohonen network.

Linear vector quantization network (LVQ): LVQ is a method for training competitive layers in a supervised manner (Pham and Oztemel, 1994); on the other hand LVQ networks learn to classify input vectors into defined target classes. The network is characterized to have an hidden layer consistent of Kohonen neurons. When an input pattern is supplied to the network, the hidden neuron whose reference vector is closest to the input pattern wins the competition for being activated. The output neuron connected closer to the winning neuron will produce a “1” output, “0” all the others. LVQ networks can classify any set of input vectors, this is an advantage with respect to the perceptron neurons. The only requirement is that the competitive layer must have enough neurons.

Many others networks of different architecture are present in literature; among these particularly interesting are the Cerebellar Model Articulation Control (CMAC) network that can be considered a supervised feedforward network with a fuzzy associative memory; the Group Method of Data Handling (GMDH) network that has a structure changing during the training until the accuracy of the desired mapping is achieved. Each GMDH neuron is an Adaptive Linear Element (Adeline) with non linear processor.

Multi-Layer Perceptron: the perceptron model was introduced by Rosenblatt (1958). Actually *perceptron* refers to a large class of neural models. The core idea of the perceptron is the incorporation of learning into the McCulloch-Pitts neuron model. One of the many perceptrons that Rosenblatt studied is the back-coupled perceptron that anticipates the currently used back-propagation model.

A scheme of a multi-layer perceptron network is given in figure (3.3). Neurons in the hidden layer only act as buffer for distributing the input signals to neurons in the hidden layer then each neuron in hidden layer sums up its input signals x_j after having weighting them with the strengths of the respective connections w_{ij} from the input layer and computes its output y_i as a function of this sum, as for examples in eq.(3.2.1).

Appendix 3.B

3.1 B The learning algorithms

This appendix reviews some of the learning algorithms rule used from the networks presented in appendix 3.2 A.

Hebbian learning: the rule proposed by Hebb (1949), in its simplest version, synthesizes the concept for which if a neuron, say A, is active and this activity causes the fire of a connected neuron B, then the efficiency of the synaptic connection between A and B should be increased, which is described in mathematical form by:

$$w_{ij}^{new} = w_{ij}^{old} + a_{ki} b_{kj} \quad (3.1.1 B)$$

where w_{ij} denotes the weight connections between neurons i and j , while a_{ki} and b_{kj} , with $k=1,2,\dots,m$, and $j = 1,2,\dots,n$, are respectively the input and output vectors. The Kohonen network implies this kind of learning rule. The neuron output values and the weight connections can be bounded in a suitable range by using the following equations:

$$w_{ij}^{new} = \frac{1}{k} (a_{ki} b_{kj} + (k-1) w_{ij}^{old}) \quad (3.1.2 B)$$

$$w_{ij}^{new} = \begin{cases} 1 & \text{if } a_{ki} b_{kj} = 1 \\ 1 & \text{if } w_{ij}^{old} = 1 \\ 0 & \text{otherwise} \end{cases} \quad (3.1.3 B)$$

Equation (3.1.3 B) assigns a binary value to a connection depending on the values in input and in output of the neuron. Networks that imply this kind of learning rule have the capacity to store a big amount of information.

Differential Hebbian learning: this algorithm can be considered as an extension of the previous one to capture the temporal changes that accrue in the input and output sequences.

$$w_{ij}(t+1) = w_{ij}(t) + \Delta x_i(t) + \Delta y_j(t-1) \quad (3.1.4 B)$$

where $\Delta x_i = x_i(t) - x_i(t-1)$ is the change in the i -th input neuron at time t , and $\Delta y_j = y_j(t-1) - y_j(t-2)$ is the change at j -th output neuron at time $t-1$.

Principal component learning: this kind of rule acts in a way that the weights of the network are principal components of the input data patterns. These components are found as minimum orthogonal set vectors of the covariance matrix. Once the basis set has been found, any vector in the space can be obtained by linear combination of the vectors in the basis. Examples of principal component learning rules are present in the works of Oja (1982) and Sanger (1989) and are expressed by the following equations:

$$w_{ij}^{new} = w_{ij}^{old} + b_{kj} (\alpha a_{ki} - \beta b_{kj} w_{ij}^{old}) \quad (3.1.5 B)$$

where α and β are non-zero constants, and

$$w_{ij}^{new} = w_{ij}^{old} + \gamma_k \left(a_{ki} b_{kj} - b_{kj} \sum_{h=1}^i y_h w_{jh} \right) \quad (3.1.6 B)$$

where γ_k is a time-decreasing learning parameters.

Networks implying these kind of rules are limited to have linear processing elements.

Competitive learning: is a two steps procedure that couples the recall process with the learning process in a two-layer neural network. This is the rule implemented in a competitive network. As a first step the winning PE has to be determined and this, as reported in the previous section for the Kohonen network, is made possible by determining the reference vector closest to the input pattern. Then as a second step the connection values of the winning reference vector have to be adjusted; the equation that allows this correction is given by:

$$w_{ij}^{new} = w_{ij}^{old} + \alpha(t)y_j(a_{ki} - w_{ij}^{old}) \quad (3.1.7 B)$$

where $\alpha(t)$ is a non-zero time-decreasing function. This algorithm has been modified to give to all neurons the same probability to be a winner.

Stochastic learning: stochastic learning uses random processes, probability and energy relationships to adjust the weight connections. This kind of processing learning is employed in multi-layer networks where the outputs of a hidden layer are randomly changed. To establish if these changes improve the network performance, the network energy is used as indicator. If the energy after the change is slower then the change is kept otherwise the change is accept according to a pre-chosen probability distribution. This procedure is repeated until the network becomes *stable* and for each pattern pair in the data set, then the collected data are used to statistically adjust the weights. The probabilistic acceptance of higher energy state allows the neural network to escape local minima.