The South African Institute of Computer Scientists
and
Information Technologists

Proceedings

of the

1996 National Research and Development Conference

Industry meets Academia

Interaction Conference Centre, University of Natal,
Durban.
26 & 27 September

Edited by
Vevek Ram

ISBN 0-620-20568-7

Cover printed by Natal Printers (Pty) Ltd, Pietermaritzburg
Copying by the Multicopy Centre, University of Natal, Pietermaritzburg
Binding by Library Technical Services, University of Natal, Pietermaritzburg

FOREWORD

This book is a collection of papers presented at the National Research and Development Conference of the Institute of Computer Scientists and Information Technologists, held on 26 & 27 September, at the Interaction Conference Centre, University of Natal, Durban. The Conference was organised by the Department of Computer Science and Information Systems of The University of Natal, Pietermaritzburg.

The papers contained herein range from serious technical research to work-in-progress reports of current research to industry and commercial practice and experience. It has been a difficult task maintaining an adequate and representative spread of interests and a high standard of scholarship at the same time. Nevertheless, the conference boasts a wide range of high quality papers. The program committee decided not only to accept papers that are publishable in their present form, but also papers which reflect this potential in order to encourage young researchers and to involve practitioners from commerce and industry.

The organisers would like to thank IBM South Africa for their generous sponsorship and all the members of the organising and program committees, and the referees for making the conference a success. The organisers are indebted to the Computer Society of South Africa (Natal Chapter) for promoting the conference among its members and also to the staff and management of the Interaction Conference Centre for their contribution to the success of the conference.

On behalf of the Organising Committee
Vevek Ram
Editor and Program Chair
Pietermaritzburg, September 1996

Organising Committee

Conference General Chairs
Mr Rob Dempster and Prof Peter Warren (UNP)

Organising Chair
Dr Don Petkov (UNP)

Secretariat
Mrs Jenny Wilson

Program Chair
Prof Vevek Ram (UNP)

Program Committee

Prof Peter Wentworth, Rhodes
Dr Milan Hajek, UDW
Prof Derek Smith, UCT
Prof Anthony Krzesinski, Stellenbosch
Dr Don Petkov, UNP
Mr Rob Dempster, UNP
Prof Peter Warren, UNP

Table of Contents

Real-Time Systems

Database Systems

Education

Practice and Experience

Information Technology and Organizational Issues

Abstracts

List of Contributors

Vladimir B Bajic
Centre for Engineering Research,
Technikon Natal,
P O Box 953
Durban 4000

C N Blewett
Department of Accounting
University of Natal
King George V Avenue
Durban 4001

Justin Cansfield
Department of Accounting
University of Natal
King George V Avenue
Durban 4001

Tom Considine
Apron Services (Pty) Ltd
P O Johannesburg
International Airport
1600

Eric Cloete
School of Electrical Engineering
Cape Technikon
Box 652
Cape Town

I Cloete
Computer Science Department
University of Stellenbosch
Stellenbosch
7600

O A Daini
Department of Computer Science
University of Botswana
Gaborone
Botswana

Nirvani Devcharan
Umgeni Water
Box 9
Pietermaritzburg
3200

P J A de Villiers
Department of Computer Science
University of Stellenbosch
Stellenbosch
7700

Ruth de Villiers
Department of Computer Science and
Information Systems
UNISA
Box 392, Pretoria, 0001

G J Erwin
Business Information Systems
University of Durban-Westville
Private Bag X54001
Durban 4000

G Ganchev
Computer Science Department
University of Botswana
PBag 0022
Gaberone, Botswana

J Geldenhuys
Department of Computer Science
University of Stellenbosch
Stellenbosch
7700

Louise Gibson
BIS, Dept Accounting & Finance
University of Durban
Pvt Bag X10
Dalbridge 4014

Mike Hart
Department of Information Systems
University of Cape Town
Rondebosch
7700

M. Hajek
Department of Computer Science
University of Durban-Westville
Pvt Bag X54001
Durban 4000

A C Hansen
Dept of Agricultural Engineering
University of Natal
Private Bag X01
Scottsville 3209

J M Hattingh
Department of Computer Science
Potchefstroom University for CHE
Potchefstroom 2520

Boris Jankovic
Centre for Engineering Research
Technikon Natal
P O Box 953,
Durban 4000

Paula Kotze
Department of Computer Science and
Information Systems
UNISA
Box 392
Pretoria, 0001

J W Kruger
Vista University
Soweto Campus
Box 359
Westhoven 2124

A C Leonard
Dept of Informatics
University of Pretoria
Pretoria
2000

Ben Laauwen
Laauwen and Associates
P O Box 13773
Sinoville
0129

Mari Le Roux
Information technology, development: project
leader
Telkom IT 1015
Box 2753
Pretoria 0001

P W L Lyne
Dept of Agricultural Engineering
University of Natal
Private Bag X01
Scottsville 3209

Rose Mazhindu-Shumba
Computer Science Department
University of Zimbabwe
Box MP167
Harare, Zimbabwe

Meredith McLeod
Centre for Engineering Research,
Technikon Natal,
P O Box 953
Durban 4000

D Moodley
Computer Management Systems
Box 451
Umhlanga Rocks
4320

Andrew Morris
P O Box 34200
Rhodes Gift
7707

R C Nienaber
Technikon Pretoria
Dept of Information Technology
Private Bag X680
Pretoria 0001

E Parkinson
Department of Computer Science
University of Port Elizabeth
Box 1600
Port Elizabeth 6000

Don Petkov
Department of Computer Science and
Information Systems
University of Natal
PBag x01
Scottsville 3209

Olga Petkov
Technikon Natal
Box 11078
Dorpspruit 3206
Pietermaritzburg

N Pillay
Technikon Natal
Box 11078
Dorpspruit 3206
Pietermaritzburg

V Ram
Department of Computer Science and
Information Systems
University of Natal
PBag x01
Scottsville 3209

Melinda Redelinghuys
Department of Computer Science and
Information Systems
UNISA
Box 392
Pretoria, 0001

Karen Renaud
Computer Science and Information Systems
UNISA
Box 392
Pretoria, 0001

H N Roux
Department of Computer Science
University of Stellenbosch
Stellenbosch
7700

T G Scott
Department of Computer Science
Potchefstroom University for CHE
Potchefstroom
2520

T Seipone
Department of Computer Science
University of Botswana
Gaborone
Botswana

Derek Smith
Department of Information Systems
University of Cape Town
Rondebosch
7700

Anette L Steenkamp
Department of Computer Science and
Information Systems
UNISA
Box 392
Pretoria, 0001

T Steyn
Department of Computer Science
Potchefstroom University for CHE
Potchefstroom 2520

H. Suleman
Department of Computer Science
University of Durban-Westville
Pvt Bag X54001
Durban 4000

A Vahed
Department of Computer Science
University of Western Cape
Private Bag X17
Bellville 7530

A Van der Merwe
Computer science and Informations Systems
UNISA
P O Box 392
Pretoria,0001

Tjaart J Van Der Walt
Foundation for Research and Development
Box 2600
Pretoria, 0001

K Vavatzandis
Department of Information Systems
University of Cape Town
Rondebosch
7700

Y Velinov
Dept Computer Science
University of Natal
Private Bag X01
Scottsville 3209

H Venter
Department of Computer Science
University of Port Elizabeth
Box 1600
Port Elizabeth 6000

H L Viktor
Computer Science Department
University of Stellenbosch
Stellenbosch
7600

R Von Solms
Department of Information Technology
Port Elizabeth Technikon
Private Bag X6011
Port Elizabeth 6000

A J Walker
Software Engineering Applications
Laboratory
Electrical Engineering
University of Witwatersrand
Johannesburg

P Warren
Computer Science Department
University of Natal
P/Bag X01
Scottsville 3209

Max Watzenboeck
University of Botswana
Private Bag 0022
Gaberone
Botswana

PARALLEL HIERARCHICAL ALGORITHM FOR IDENTIFICATION OF LARGE-SCALE INDUSTRIAL SYSTEMS

Boris Janković and Vladimir B. Bajić

Centre for Engineering Research, Technikon Natal,
P.O.Box 953, Durban 4000, Republic of South Africa
Tel.: (+27) 31-204-2560, Fax: (+27) 31-204-2560
e-mail: bajic.v@umfolozi.ntech.ac.za

Abstract

Parallel execution is the most powerful method of speeding up numerical analysis. However, numerical complexity increases much faster than the problem size. For this reason the computational effort required for the analysis of systems of big dimension, or generally speaking, large-scale systems, will be substantial even with application of parallel computer architectures. This leads to necessity for proper model reduction, as well as decomposition of the physical problem into a set of smaller-size problems. In this paper we propose a method which comprises both of these demands and results in an algorithm for parallel hierarchical identification of reduced-size large-scale models.

Introduction: Large-scale systems and related numerical problems

Large-scale systems (LSS), sometimes called complex systems, are usually defined as systems that consist of large number of interacting subsystems (Siljak 1983). Apart from large dimensionality of such systems, the nature of interactions between the subsystems can be particularly complex (Siljak 1983). Very often, there are difficulties even in identifying inputs and outputs of such systems (Siljak 1983). Also, numerical difficulties in analysis of even relatively low-order MIMO systems may be significant, and since in many cases LSS can be treated as MIMO systems, the same type of numerical problems is immanent to them too. One of the most distinguishing characteristics of LSS is the fact that "one-shot" approach methods can not be, in general, successfully used in the study of their behavior. This poses a significant problem if an on-line analysis of such systems is needed. For instance, in the case of self-adaptive controllers for LSS that change behavior relatively fast compared to dominant time constants of the LSS, obtaining a satisfactory model within a given time constraint may be a very difficult problem which is sometimes impossible to solve.

One of the most common approaches for speeding up calculations, in general, is utilization of parallel processing algorithms. In this paper we investigate some possible benefits by parallelism at the appropriate hierarchical level that may be used for implementation of faster self-adaptive control of LSS. There are several levels of parallelism related to algorithm execution. What they all have in common is that a sequential problem is somehow transformed into an equivalent form that is suitable for parallel processing. Conceptually, the lowest hierarchical level is embedded on instruction level, and it is closely related to underlaying (multi)processor architecture and corresponding compiler design (Malinowski *et al.* 1985). The initial sequential problem is still very much independent of this parallelism and details of tasks necessary to convert such a problem into one suitable for parallel processing are often hidden from the application level. SIMD (Single Instruction, Multiple Data) and MIMD (Multiple Instruction Multiple Data) are examples of parallel architectures (Malinowski *et al.* 1985).

Partially overlapping with this one is the next hierarchical level in which sequential, mathematical (mostly numerical) algorithms are transformed into parallel ones. There are many examples of these including partial differential equations, vector addition etc. (Lei *et al.* 1985). Sometimes, specialized parallel processors are specifically build for efficient execution these tasks. However, it is obvious that such algorithms and processor realizations can not be used for other types of

numerical problems. At this point it is interesting to note that analog computers are capable of fast simulations because all of their elements work in parallel (Pearce 1985).

Still higher hierarchical level of parallelism is obtained at the application level. In this case the initial problem is broken into a set of smaller-size problems which can be then solved in some phases independently of each other. For such a situation problem decomposition depends crucially on the nature and structure of the actual problem. For example, principles used for decomposition in circuit analysis problems are not applicable directly to ecological systems or economic systems. However, once obtained via decomposition approach at this level, subproblems are fairly autonomous and they are very loosely dependent on the underlying processor architecture. This goes up to the degree that subproblems can be solved independently on separate computers.

This problem decomposition approach is often the only way of reducing the rapid increase of the size of numerical problems as the problem size increases. Parallelism offered by parallel computer architectures will, naturally, significantly decrease the time necessary for solving the numerical problems, but even in the case of problem decomposition, the problem size can reach the level when this parallelism will not be fast enough to complete the task given within the prespecified time interval.

A very prominent class of systems that exhibits numeric difficulties are large-scale dynamic systems (*Si*ljak 1983). Transformation of problems associated with LSS analysis and design into a form suitable for parallel processing is not easy. These problems are not only due to the so-called "curse of dimensionality", but also due to the incomplete knowledge of subsystem interactions. Another example of the increased numerical complexity could be large-scale optimization problems. The most effective techniques for breaking up the original problem into a set of smaller ones are decomposition-coordination (Schoeffler 1971) and decomposition-aggregation methods (*Si*ljak 1983). The first one is very useful for hierarchical control, while the other is good for model reduction. In what follows we propose a technique that uses decomposition principle and model reduction in order to provide a significant parallelism in the identification of a reduced order models of LSS.

The results are illustrated by a simulation example.

Decomposition as problem size reduction technique

Following mainly Guardabassi (1982) let S represent a solution set of an abstract problem P. The problem P can be modelled (defined) as an ordered triple $P = (D, \pi, Z)$, where D is a data set (over which the problem is defined). Here, the mapping $\pi : D \ni d \rightarrow \pi(d) \in Z \ (Z = im\pi(D))$, is called an intrinsic mapping of the problem P. Therefore, finding a solution of a problem P for a given d means to find (any) element $s \in Z$, such that $s = \pi(d)$ and where $Z \neq \emptyset$. In this case we say that P has solution(s). If there is only one $s = \pi(d)$, $s \in Z$, $Z \neq \emptyset$, then we say that the problem P has a unique solution for d. In the case when $\forall d \in D$ there exist only one $\pi(d) \in Z$, the mapping π is called the intrinsic function of P. In the case $Z = \emptyset$, P has no solution for d. We will consider only the situation when π is a function.

When the problem $P = (D, \pi, Z)$ is a difficult (complex) one, we would like to transform it into some other (ideally equivalent) problem $P' = (D', \pi', Z')$, which is easier to solve. For the purpose of building our identification procedure, we will assume that the problem P' belongs to the class of composite problems. A composite problem of order N consists of N subproblems which may, or may not, be mutually dependent. Suppose that subproblem P_i of the composite problem P is defined by $P_i = (D_i, \pi_i, Z_i)$. From this, we see that it has its local data and local solution set, as well as its intrinsic function. To obtain the global solution of the composite problem P, it is necessary to consider the global data set D (which generally will not be union of local data sets i.e. $D \neq \bigcup_i D_i$), set of interaction functions which show dependencies of local data sets on other subsystems' local solutions, and finally, global solution function which relates global solution to local solutions.

A very useful tool for representing subsystem interactions, and generally the structure of LSS, is

graph theory. Each node can either represent a subproblem or its local data, and branches of the graph represent the subsystem interactions. If such a (di)graph is acyclic, then the local data for each of the subproblems P_k depends only on local solutions of subproblems P_i, $i = 1, 2, .., k-1$, and possibly itself, but not on other local solutions, i.e. on solution of P_i, $i > k$. In this case all subproblems can be solved in a sequence. However, if each subproblem has its own, autonomous, local data, that are not influenced in any way by other subsystems, then parallel solvers (algorithms) can be used . This is the principle that we will utilize in our approach to hierarchical identification.

Parallel Identification

From previous considerations it is clear that decomposition principle can be taken as the basis for implementation of parallel processing. It is successfully used in many aspects of LSS analysis, such as simulation (Malinowski *et al.* 1986). One problem, however, in which decomposition principle can not be applied directly is identification. The problem is that output of identification procedure is a model, so, initially, there is no model to decompose. To be able to transform an identification problem into a form suitable for parallel processing, we have to approach this indirectly.

When dealing with dynamic systems, similar considerations may be applied, but this time instead of considering problem models we consider systems. Clearly, we may have global systems representation (which is some form of input-output representation that hides its internal structure), as well as a composite model representation, in which the internal structure (i.e. subsystems and their interactions) is preserved to some extent. The relation between the two representations is given in Takahara (1982). Only for some special cases this relation will be isomorphic; in reality global models may have numerous equivalent representations as composite models.

This, however, is important for us as we can assume that model resulting from identification process will have its equivalent composite representation. Suppose that the result of identification of some given input/output data is a model M in global model representation. Based initially on Bajić (1995), it is argued in Janković (1996) that model M can be approximated by its composite representation M_{cm} in such a way that, if this representation is taken in a certain form, then a two-phase algorithm results: in the first phase some composite model components are identified in parallel, and in the second phase subsystem interactions are determined. The problem of matching the global model with the one in the composite representation has its counterpart in control theory where it is called "exact model matching procedure" (Moore and Silverman 1972). Our goal is to match some fictious, "assumed to be true" model M with a composite one using output feedback. This procedure due to way how it is implemented belongs to hierarchical type of identification. Analysis of the ability of such approach to approximate the global model can be found in Janković (1996). The particular composite forms for identification of SISO processes are given in Janković and Bajić (1996) and for MIMO systems in Bajić and Janković (1997).

In this paper we will combine the model reduction and parallel hierarchical identification to speed up the modeling necessary for self-adaptive controllers of LSS. With regard to model reduction, we take approach opposite to the one utilized by Obinata and Inooka (1976), Ouyang *et al.* (1897), who select system modes according to their contribution to power spectrum of output. In our case since there is no model initially, we use the method to discard the portion of power spectrum that has small contribution to total power. After reducing the bandwidth of such systems we can apply parallel identification procedure more efficiently to come to the subsystem models, and finally to the composite model.

LSS model description for parallel identification

In this section we give the mathematical description of the assumed LSS model to which the proposed parallel identification procedure will be applied. Let us assume that the original system, which is a SISO one with the input signal u and the output signal y, is composed of N subsystems S_j which mutually interact via the interconnection subsystem S_I. We assume that each of the subsystems S_j is described by the transfer function G_j and a dead-time L_j. Thus for the j-th subsystem S_j we

have the operator equation

$$y_j = G_j(s)e^{-L_j s}u_j = \frac{B_j(s)}{A_j(s)}e^{-L_j s}u_j$$
$$= \frac{b_{mj,j}s^{mj} + b_{mj-1,j}s^{mj-1} + \ldots + b_{1,j}s + b_{0,j}}{s^{nj} + a_{nj-1,j}s^{nj-1} + \ldots + a_{1,j}s + a_{0,j}}e^{-L_j s}u_j \quad (1)$$

where y_j and u_j represent the output and the input signals of S_j, respectively. We assume the original system will have reasonably good representation as a LSS of the form

$$y_j = G_j(s)e^{-L_j s}u_j, \quad j = 1, 2, ..., N \quad (2a)$$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ ... \\ y_N \end{bmatrix} \quad (2b)$$

$$y = \left(\mathbf{c}_{cm}e^{-Ls}\right)\mathbf{y} \quad (2c)$$

$$u_j = u + m_j, \quad j = 1, 2, ..., N \quad (2d)$$

$$\mathbf{m} = \begin{bmatrix} m_1 \\ ... \\ m_N \end{bmatrix} \quad (2e)$$

$$\mathbf{m} = \mathbf{A}_{cm}\mathbf{y} \quad (2f)$$

where $\mathbf{y}$ is the vector whose components are the outputs of subsystems, $\mathbf{m}$ is the vector of feedback signals, $\mathbf{A}_{cm}$ is the $N \times N$ feedback matrix, $\mathbf{c}_{cm}$ is the N-component vector combining the subsystem outputs, and L is the dead-time in the LSS model (1-2) separate from those included in the individual subsystems S_j.

The purpose of the identification is to determine the coefficients in the transfer functions G_j, $j = 1, 2, ..., N$ and dead-times L_j, $j = 1, 2, ..., N$, for subsystems S_j, as well as the matrix $\mathbf{A}_{cm}$, the vector $\mathbf{c}_{cm}$ and the dead-time L. The proposed parallel identification procedure by reduced-order models can be stated as follows. In the first step, the bandwidth of the output signal y is reduced by discarding those frequencies with insignificant power contribution. Secondly, transfer function models $G_j(s)e^{-L_j s}$ of orders n_1, $n_1+1, ..., n_1+N-1$ are identified. This can be done in parallel. In the final step, parameters $\mathbf{A}_{cm}$, $\mathbf{c}_{cm}$ and L of the composite model (1-2) are identified.

Complexity analysis

In this section we give a rough estimate of the complexity of the identification procedure proposed. We would like to analyze possible benefits of the proposed decomposition and identification method. In order to do this we must somehow relate the computational effort with the problem size. Let us assume that in the parameter estimation problem, the problem size is defined as the number of parameters entering the optimization procedure. For the transfer function models

$$y = G(s)e^{-Ls}u = \frac{B(s)}{A(s)}e^{-Ls}u = \frac{b_m s^m + b_{m-1}s^{m-1} + \ldots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \ldots + a_1 s + a_0}e^{-Ls}u \quad (3)$$

the number of parameters that need to be determined is

$$\Theta(G, n) \leq 2n + 1$$

since $m \leq n$ due to physical realizability. Thus in the worst case one gets

$$\Theta(G, n) = 2n + 1$$

This relation is linear. For the composite (LSS) model (1-2) we have

$$\Theta(CM,n)=\sum_{i=1}^{N}\Theta(G_i,n_i)+\Theta(IM,N)$$

where $\Theta(CM,n)$ is the number of parameters for composite model of order n, $\Theta(G_i,n_i)$ is the number of parameters for each of the subsystem's transfer function, where n_i is the order of $i-th$ subsystem, and $\Theta(IM,N)$ is the number of parameters for the interaction subsystem S_I.

To get the expression for the maximal number of parameters of the composite model (1-2) that needs to be determined simultaneously we note that, without loss of generality, for the numbers of parameters of subsystems S_j, $j=1,2,...,N$, the following holds

$$\Theta(G_1,n_1)\leq\Theta(G_2,n_2)\leq\ldots\leq\Theta(G_N,n_N)$$

as subsystems S_j can be numbered in such a way. Since all parameters of a subsystem S_j can be determined independently of the parameter identification procedures for S_i, $i\neq j$, then the number of parameters $\Theta(G_N,n_N)$ determines in a way the maximal computational effort and time needed in any branch of sequential processing for obtaining parameter estimates for any of the systems S_j in parallel computation. For the interaction subsystem S_I, if subsystem's dead-times are fixed, we can write

$$\Theta(IM,N)=N^2+N+1$$

where N^2 term comes from the matrix $\mathbf{A}_{cm}$, term N comes from vector $\mathbf{c}_{cm}$, and the last parameter is the composite model dead-time. It should be noted that the composite model (1-2) is of the order $n=\sum_{i=1}^{N}n_i$.

For the first step in parallel processing identification, the maximal number of parameters to be determined in an 'one-shoot' fashion is $2n_N+1$, as already asserted. For the second phase the number of parameters is N^2+N+1.

Let us assume that the number of instructions necessary for optimization of the n-parameter problem is proportional to the square of number of parameters, i.e. proportional to n^2. This is a very conservative assumption and in any real-world situation this ratio is much larger. However, even with this we will show a great advantage of the method proposed. So the total number of instruction needed for the largest sequential processing demand in any of the parallel processing branches will be

$$\begin{aligned} ins &= \Theta(G_N,n_N)^2+\Theta(IM,N)^2=(2n_N+1)^2+\left(N^2+N+1\right)^2 \\ &= (2n_1+2N-1)^2+\left(N^2+N+1\right)^2 \end{aligned} \tag{4}$$

For the composite model given as (3) in the 'one-shoot' approach the number of parameters that we need to determine simultaneously is

$$\begin{aligned} \Theta(G_{LS},n) &= 2n+1=2\sum_{i=1}^{N}n_i+1=2\sum_{i=1}^{N}(n_1+i-1)+1 \\ &= N^2+(2n_1-1)N+1 \end{aligned}$$

and the corresponding number of instructions will be

$$ins=\Theta(G_{LS},n)^2=\left[N^2+(2n_1-1)N+1\right]^2 \tag{5}$$

To have a measure of the relative complexity in the case of parallel identification approach and the 'one-shoot' approach we now form a ratio Φ between the number of instructions (4) and the number of instructions (5), and we call this the relative complexity index. This ratio takes into account the effect of decomposition. Thus we have

$$\Phi(N, n_1) = \frac{(2n_1 + 2N - 1)^2 + (N^2 + N + 1)^2}{[N^2 + (2n_1 - 1)N + 1]^2}$$

We see that Φ is function of the number of subsystems N and the lowest order n_1 of the subsystem models.

The comparison of relative complexity of the identification methods for the parallel processing identification and 'one-shoot' identification for the model (3) of the same model orders is given in Fig.1. Fifteen curves are shown, obtained for $n_1 = 2, 4, ..., 30$, and for $N = 2, 3, ..., 20$. The curve at the top is obtained for $n_1 = 2$, and the position of curves gradually goes down with the increase of n_1. The greatest reduction in the algorithm complexity is obtained for $n_1 = 30$ and for $N = 8$. This roughly corresponds to the 30 times reduced algorithm complexity in parallel identification method compared to the normal 'one-shoot' approach. Essentially, graphs in Fig.1 show that the parallel identification technique proposed makes a significant reduction in the required instructions during the identification (and the time required for that) compared to the 'one-shoot' approach..

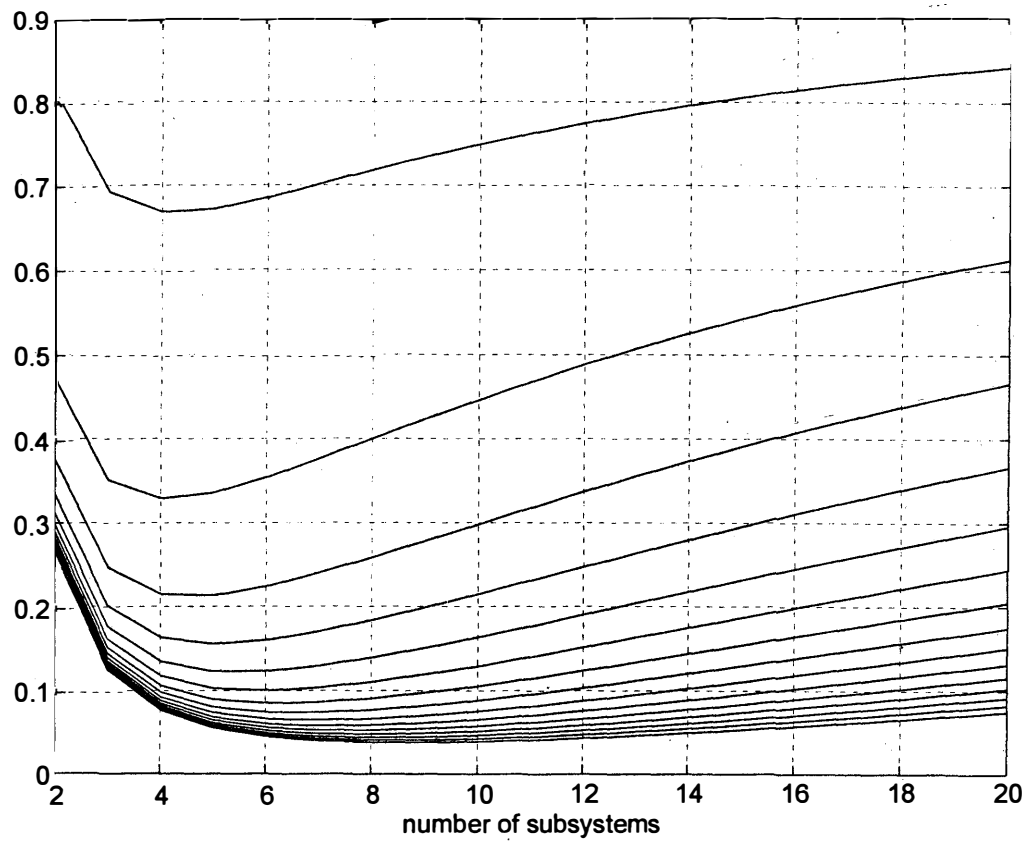

Fig. 1 Relative complexity curves

Example

The application of the method proposed is tested on the 3-channel autopilot model of order 74 (Simulink 1995). The input-output sequence is generated and a high frequency output components were truncated by passing the output signal y through a Butterworth filter. The final results of identification are shown in Figs. 2 and 3. As can be expected, the 'steady state' part of response is modelled more accurately because steady state is unaffected by low-pass filtering. This is clearly visible from Fig. 3 where the error between the response generated by the original model and the

response of the identified reduced order model is shown.

In the first phase three subsystem models of orders 2, 3 and 4 are identified. Note that identification of these three subsystems can be done in parallel. Subsystem dead-times were kept fixed to zero. The identified models in the first phase are as follows:

$$G_1 = \frac{1.1868s + 1.1189}{s^2 + 0.9201s + 1.1189} e^{-L_1 s}, \quad L_1 = 0$$

$$G_2 = \frac{3.6471s^2 + 8.6297s + 6.9719}{s^3 + 7.4694s^2 + 6.0564s + 6.9719} e^{-L_2 s}, \quad L_2 = 0$$

$$G_3 = \frac{0.8302s^3 + 8.2375s^2 + 32.6406s + 30.0417}{s^4 + 2.9463s^3 + 29.905s^2 + 24.2209s + 29.9163} e^{-L_3 s}, \; L_3 = 0$$

In the second phase the identified subsystems are coupled via the interconnection subsystem S_I to form the LSS model of the form (1-2). The parameters obtained in this phase are

$$\mathbf{A}_{cm} = \begin{bmatrix} 0.0078 & -0.3389 & -0.2879 \\ -0.0861 & -0.0457 & 0.0383 \\ 0.0334 & -0.1791 & 0.3577 \end{bmatrix}$$

$$\mathbf{C}_{cm} = \begin{bmatrix} 0.4234 & -0.3325 & 0.9200 \end{bmatrix}$$

$$L = 0.0031$$

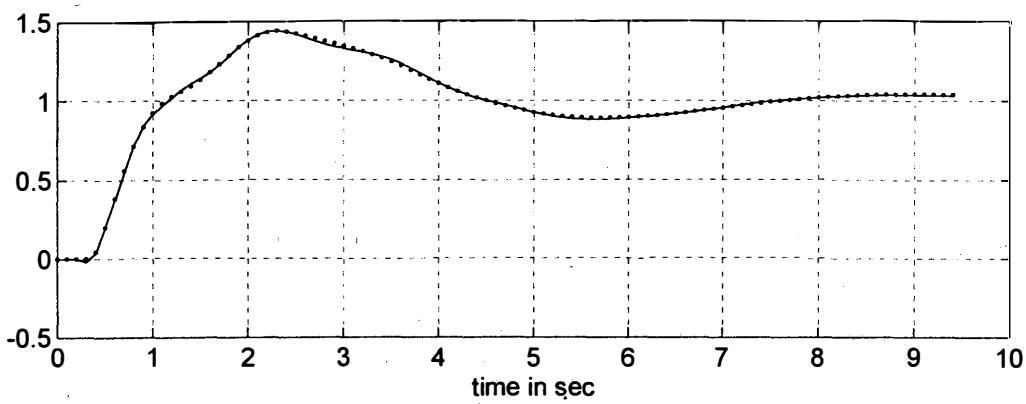

Fig. 2 Response of original model of the order 74 (solid) and reduced order model (dotted curve)

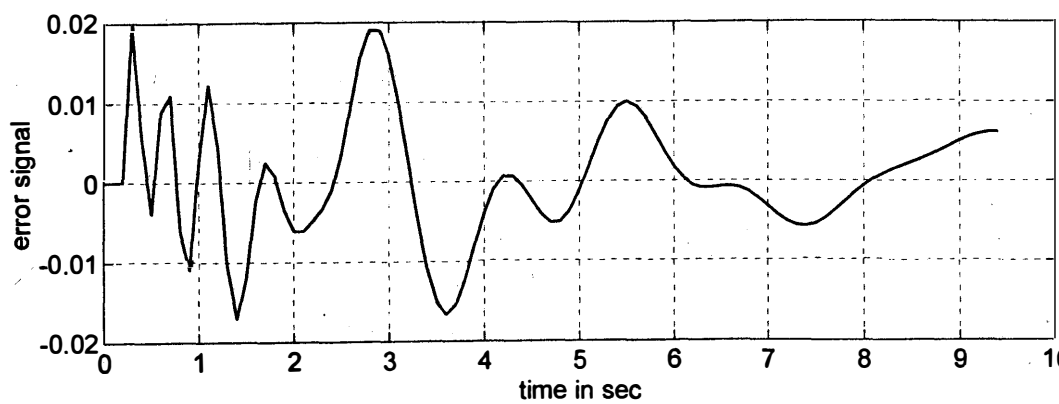

Fig. 3 Error of the response of the original model and reduced order model

Conclusions

A hierarchical method suitable for parallel identification of LSS is proposed. The method has advantages regarding computational complexity compared to the 'one-shoot' identification approach. Potential domain of application of this method is in the identification of complex industrial processes from real-time measurements. The method is presented in a form suitable for SISO systems, although it can be adapted to cater for MIMO system applications.

References

G.Authie and D. El Baz (1986). "A Multimicroprocessor for Parallel Processing", in "*Parallel Processing Techniques for Simulation*" (ed. M. A. Singh, A. Y. Allidina, B. K. Daniels), Plenum Press, pp. 229-238.

V.B.Bajic (1995). Private communication, January, RSA.

V.B.Bajić & B.Janković (1997). "Hierarchical identification pted for IFAC-IFIP-IMACS Conference, Belfort, France, May 20-22, 1997.

D.P.Bertsekas and J.N.Tsitsiklis (1989). "*Parallel and Distributed Computation - Numerical Methods*", Prentice-Hall.

G.Guardabassi (1982). "Composite Problem Analysis", *Large-Scale Systems*, Vol.3, pp.1-11.

B.Janković (1996) "*Hierarchical Identification of Large-Scale Systems*", Doctoral thesis, Technikon Natal, submitted.

B.Janković and V.B.Bajić (1996). "Reduced Optimization Complexity in Identification of Large-Scale SISO Processes", CSS '96, Brno, Czech Republic (to be published).

S.Lei, A.Y.Allidina and K.Malinowski (1985). "Clustering technique for rearranging ODE systems", *Proceedings of the First European Workshop on Parallel Processing Techniques for Simulation*, October, pp.31-44.

K.Malinowski, A.Y.Allidina and M.G.Singh (1986). "Decomposition-Coordination Techniques for Parallel Simulation", in "*Parallel Processing Techniques for Simulation*" (ed. M. A. Singh, A. Y. Allidina, B. K. Daniels), Plenum Press, pp.1-11.

K.Malinowski, A.Y.Allidina, M.G.Singh and W.D.Crorkin (1985), "Decomposition - coordination technique for parallel simulation, Part 1", *Large-Scale Systems*, Vol.9, pp.101-115.

B.C.Moore and L.M.Silverman (1972), "Model matching by state feedback and dynamic compensation", *IEEE Transactions on Automatic Control*, Vol.17, pp.491-496.

G.Obinata and H.Inooka (1976). "A Method for Modeling Linear Time-Invariant Systems by Linear Systems of Low Order", *IEEE Transactions on Automatic Control*, Vol.21, pp.602-603.

M.Ouyang, C.M.Liaw and C.T.Pan (1987). "Model Reduction by Power Decomposition and Frequency Response Matching", *IEEE Transactions on Automatic Control*, Vol.32, pp.59-62.

J.G.Pearse, P.Holliday and J.O.Gray (1985), "Survey of parallel processing in simulation", *Proceedings of the First European Workshop on Parallel Processing Techniques for Simulation*, October, pp.183-202.

J.D.Schoeffler (1971), *Static Multilevel Systems*, in *Optimization Methods for Large-scale Systems*, edited by David A. Wismer, McGraw-Hill.

*SIMULINK User's Guide (*1995), MATLAB, The MathWorks, Inc.

D.D.Siljak (1983). "Complex dynamic systems: dimensionality, structure and uncertainty", *Large-Scale Systems*, Vol.4, pp.279-294.

Y.Takahara (1982). "Decomposability conditions for general systems", *Large-Scale Systems*, Vol.3, pp.57-65.