A Thesis Submitted for the Degree of PhD at the University of Warwick

Permanent WRAP URL:  
http://wrap.warwick.ac.uk/138398

Copyright and reuse:
This thesis is made available online and is protected by original copyright. 
Please scroll down to view the document itself. 
Please refer to the repository record for this item for information to help you to cite it. 
Our policy information is available from the repository home page.

For more information, please contact the WRAP Team at: wrap@warwick.ac.uk
IDENTIFICATION FOR
DISTRIBUTED PARAMETER SYSTEMS

Carlos Silva Kubrusly

ABSTRACT

This thesis considers the parameter identification problem for systems governed by partial differential equations. The various identification methods are grouped into three disjoint classes namely: "Direct Methods", "Reduction to a Lumped Parameter System", and "Reduction to an Algebraic Equation".

The major subject investigated here is concerned with the applicability of stochastic approximation algorithms for identifying distributed parameter systems (DPS) operating in a stochastic environment, where no restriction on probability distributions is imposed. These algorithms are used as a straightforward identification procedure, converge to the real value of the parameters with probability one, and are suitable for on-line applications. In this way, a new identification

* This research was supported by CNPq. (Brazilian National Research Council) under Grant No. 3712/73.
IDENTIFICATION FOR
DISTRIBUTED PARAMETER SYSTEMS*

Carlos Silva Kubrusly

ABSTRACT

This thesis considers the parameter identification problem for systems governed by partial differential equations. The various identification methods are grouped into three disjoint classes namely: "Direct Methods", "Reduction to a Lumped Parameter System", and "Reduction to an Algebraic Equation".

The major subject investigated here is concerned with the applicability of stochastic approximation algorithms for identifying distributed parameter systems (DPS) operating in a stochastic environment, where no restriction on probability distributions is imposed. These algorithms are used as a straightforward identification procedure, converge to the real value of the parameters with probability one, and are suitable for on-line applications. In this way, a new identification

* This research was supported by CNPq. (Brazilian National Research Council) under Grant No. 3712/73.
method is developed for DPS described by linear models, driven by random inputs, and observed through noisy measurements. The very real case of noisy observations taken at a limited number of discrete points located in the spatial domain is considered. The proposed identification method assumes that a previous system classification has been performed, such that the model to be identified is known up to a set of space-varying parameters, where extraneous terms may be included.
ACKNOWLEDGEMENTS

I wish to thank my supervisor, Dr. Ruth F. Curtain, for her constant encouragement in all stages of this work. Her advice and observations were particularly valuable to me.

I am also grateful to Mr. Luca Braito for his assistance in computer programming.

Special thanks are due to Berenice for her efficient and careful typing of a difficult manuscript.
## CONTENTS

### INTRODUCTION

1

### CHAPTER 1  SOME BASIC ASPECTS IN SYSTEM IDENTIFICATION

1.1 — System Characterization, Classification and Identification  
1.2 — Lumped Parameter Systems and Distributed Parameter Systems  
1.3 — System Identification Versus State Estimation

3  
12  
16

### CHAPTER 2  DISTRIBUTED PARAMETER SYSTEMS IDENTIFICATION: A SURVEY

2.1 — Introduction  
2.2 — Classification of Methods  
2.3 — A Concise General Review  
2.4 — Conclusions

21  
25  
29  
40

### CHAPTER 3  MATHEMATICAL PRELIMINARIES

**PART I: MODELS FOR DPS**

3.1 — On a General Class  
3.2 — Linear Models

42  
44

**PART II: HIGHER ORDER FINITE-DIFFERENCES**

3.3 — Shift, Difference and Slope Operators  
3.4 — Approximating Derivatives

47  
59

**PART III: STOCHASTIC APPROXIMATION**

3.5 — The Dvoretzky's Theorem in Banach Spaces  
3.6 — A Proposed Stochastic Approximation Algorithm in Hilbert Space

66  
70
INTRODUCTION

The central theme in this thesis is the application of stochastic approximation theory, as a straightforward identification technique, for determining parameters in systems governed by partial differential equations (PDE).

In chapter 1 we begin by presenting some introductory aspects of the general problem of system identification. Fundamental concepts (such as: system characterization, classification and identification, as well as lumped (LPS) and distributed (DPS) parameter systems) are explained in order to make precise what kind of problem will be considered here.

A survey on the DPS identification field is presented in chapter 2. Before reviewing the various approaches used to face the problem, we introduce a new classification for the DPS identification methods. Briefly, these methods can be grouped into three disjoint classes: The first one uses optimization techniques directly on the model that describes DPS. The second class of methods is characterized by reducing the DPS to an equivalent LPS. In a similar way, the methods in the third class reduce the DPS to a set of algebraic equations.

Chapter 3 is concerned with the mathematical concepts and techniques that will be used later for identification purposes. It contains three independent parts: In part I we consider some classes of models for DPS described by PDE. Higher order finite-differences are introduced in part II, where the basic lemmas for model approximation are derived. Relevant aspects of the stochastic approximation theory,
as well as some applicable stochastic approximation algorithms in Hilbert space, are presented in part III.

The main results appear in chapter 4. There we propose a new method for identifying space-varying parameters in distributed systems. The DPS is supposed to be operating in a stochastic environment, and no restriction concerning probability distributions is imposed. A class of linear models (where extraneous terms may be included) driven by random inputs and observed through noisy measurements is considered. These measurements are taken at a limited number of discrete points located in the spatial domain. The theory is developed by assuming a one-dimensional spatial domain, but direct extensions to multi-dimensional spatial domains can be obtained as shown in section 4.7. Higher order finite-difference techniques are used to reduce the DPS to an equivalent discrete-time LPS. The parameters are then placed in an explicit form which is suitable for applying recursive identification schemes. In this way, stochastic approximation algorithms (as proposed in chapter 3, part III) are used as a straightforward on-line identification procedure, rather than a simple searching scheme for finding estimates previously obtained by means of any other optimization technique. These algorithms converge to the real value of the parameters with probability one.

Finally, the performance of the identification method is analysed in chapter 5. After a brief summary concerning second-order models, we present three examples dealing with parabolic and hyperbolic PDE. Conclusions and suggestions for further research are also included.

References are listed at the end of the thesis, and grouped according to chapter.
CHAPTER 1

SOME BASIC ASPECTS IN SYSTEM IDENTIFICATION

The general idea of System Identification is a very wide one and different authors dealing with this subject use the term "system identification" in some slightly different ways.

It is not our intention in this study to pose a formal or rigorous definition of system identification. Instead of this, we intend to present an informal and brief introduction on this topic as a starting point for the subsequent chapters.

The three stages of the identification procedure namely, system characterization, system classification and system identification are discussed in section 1.1. The meaning of lumped parameter system and distributed parameter system is explained in section 1.2. Finally, in section 1.3, the problems of system identification and state estimation are discussed and the difference between these two concepts is emphasized.

1.1 - SYSTEM CHARACTERIZATION, CLASSIFICATION AND IDENTIFICATION

One of the first attempts to explain the main concepts involved in System Theory was made by Zadeh [1]. Under the subtitle "Principal Problems of System Theory", he formulated twelve of the most important problems (both from theoretical and practical viewpoints), which are summarized below:
1) System Characterization
2) System Classification
3) System Identification
4) Signal Representation
5) Signal Classification
6) Systems Analysis
7) Systems Synthesis
8) System Control and Programming
9) System Optimization
10) Learning and Adaption
11) Reliability
12) Stability and Controlability

It would be helpful, for the purpose of our study, to add two important problems to the above list:
13) Observability
14) State Estimation (Filtering, Smoothing and Prediction)

As we are interested only in the meaning of the first three of those problems and mainly in the third one, we will discuss these in the light of Zadeh's paper.

*System Characterization:* "Representation of input-output relationships in mathematical form; transition from one mode of representation to another".

This problem is concerned both with the different ways in which the input-output relationship of a specific system can be represented (i.e., in terms of differential equations, state equations,
characteristic functions, frequency response functions, integral operators, etc.), and the forms which these representations assume for several types of systems (i.e., continuous-time, discrete-time, stochastic, deterministic, memoryless, finite-memory, causal, etc.).

System Classification: "Determination on basis of observations of input and output, of one among a specified class of systems to which the system under test belongs".

In the following we will call the "class of systems" by the class of models or simply the class, the elements of a class are obviously called by models or mathematical models, and the "system under test" will be called the system (some authors call it process or plant).

This kind of problem may be stated as follows:

Assume that

i) I is an index set,

ii) $C_\alpha$, $\alpha \in I$, are classes of models $M$,

iii) $F = \{C_\alpha; \alpha \in I\}$ is a family of these classes.

Suppose we are given a system $S$ and a family $F$, such that $S$ is characterized by $F$ and belongs to one of its classes, say $C_\alpha$. The problem is to determine $C_\alpha$ by observing the responses of $S$ to some different inputs.

---

1 The expression "$S$ belongs to $C_\alpha$" is not a very accurate one. What really belongs to $C_\alpha$ is a specific model, say $M^*$, which is in some sense "equivalent" to $S$. The meaning of "equivalent" will be precised later in this chapter.
Example 1. An important particular problem in classification is the following:

Let

i) \( I = \mathbb{Z}_+ = \{1, 2, 3, \ldots\} \) : the set of all positive integers.

ii) \( C_n \) be the class of all models \( M \) described by a single ordinary differential equation of order \( n \).

iii) \( F = \{ C_n ; n \in \mathbb{Z}_+ \} \) be the family of all these classes. That is, the set of all ordinary differential equations of any finite order.

Suppose \( S \) characterized by \( F \). Which means: suppose it is known that the system is represented in terms of an ordinary differential equation (this assumption concerns the System Characterization).

The question is: What is its order? In other words, which class \( C_n \in F \) does \( S \) "belong"? (or better: which class \( C_n \in F \) does \( M^* \), the "equivalent" model, belong?)

The problem of finding the class \( C_n \) is, sometimes, described as the black box approach. Roughly speaking, this means the determination of the structure (or "topology") of the system, considering it as a perfect "black box". On the other hand, assuming that some

---

2 All ordinary differential equations belonging to a given \( C_n \) (i.e., the models \( M \in C_n \)) are completely known up to a set of \( m \) (\( m > n \)) parameters, which are the coefficients of the differential equation. The problem of finding these parameters concerns to the System Identification, and it will be commented later in this section.
class of models, say $C_\kappa$, is available\(^3\) the determination of one element $M$ in $C_\kappa$ is, sometimes, called the opaque box approach. This is the subject of System Identification which is described below.

**System Identification:** "Determination, on basis of observations of input and output, of a system within a specified class of system to which the system under test is equivalent".

Observing the nomenclature introduced before (i.e., the meaning of class, model and system) the identification problem may be formulated as follows:

Given a class $C_\kappa$ (with each member of $C_\kappa$ completely characterized), the problem is to determine a model $M$ in $C_\kappa$ which is equivalent to the system $S$. Briefly: find $M \in C_\kappa$ such that $M$ is equivalent to $S$.

But what does the term "equivalent" mean in this particular case?

Assume that

i) $W$ is some space of inputs and $w$ a element of $W$.

ii) $y_S = y_S(w)$ is the system output and $y_M = y_M(w)$ is the model output, for some pre-selected input $w$ in $W$.

\(^3\) This assumption can be based on some "a priori" knowledge of the system's structure. This "a priori" knowledge can be thought as the result of a previous classification. We will make more comments about what we mean by "a priori" knowledge, later in this chapter.
The equivalence is often defined in terms of a cost function $J$ which is a functional of $y_S$ and $y_M$. That is:

$$J = J(y_S, y_M)$$

The model $M^*$ which is equivalent to the system $S$ will be one such that the cost function $J$ is minimized. Symbolically we have

$$M^* \equiv S \iff J(y_S, y_{M^*}) = \min_{M \in C_\alpha} J(y_S, y_M)$$

where the symbol $\equiv$ means "equivalent to" in the above sense.

So, when equivalence is defined by means of a cost function $J$, the identification problem is reduced to an optimization problem: find a model $M \in C_\alpha$ such that the cost function $J$ is minimum.

In such case, the questions related to the existence and uniqueness of the solution are the main problems of system identification. Some studies of these problems have been done by Bellman and Åström [2] for a simple class of linear systems.

The class $C_\alpha$ is called identifiable if the optimization problem has a unique solution.

At this stage some diagrams could be helpful to one visualize the identification procedure.

Let $Y$ be the space of outputs, such that:

$$y_S = y_S(w) \in Y ; \ w \in W$$

$$y_M = y_M(w) \in Y ; \ w \in W, \ M \in C_\alpha$$
Assume $\mathbb{Y}$ is a metric space$^4$ and define the cost function $J$ as the metric $d$ in $\mathbb{Y}$. That is:

$$J = d(y_M, y_S)$$

So, the identification problem (see fig. 1) can be formulated as follows: find a model $M \in \mathcal{C}$ such the distance from its output $y_M$ to the system output $y_S$ is the smallest possible.

The figure 2 shows the same situation represented by the engineering viewpoint; that is, by block diagrams (for simplicity the outputs are assumed to be scalars and no external noise is supposed to corrupt them).

![Diagram](image)

**Fig. 1**: $S \preceq M^* \subseteq \mathcal{C} \iff d(y_S, y_{M^*}) = \min_{M \in \mathcal{C}} d(y_S, y_M)$

$^4$ Actually, it would be necessary some further requirements (as linearity, inner product, and completeness) in the algebraic-topological structure of the output space $\mathbb{Y}$, when the identification problem is reduced to an optimization one. In this case $\mathbb{Y}$ becomes a Hilbert space.
Now let us return to the essential meaning of system identification, as formulated above. Generally speaking, there are two possible ways to determine, based on the observations of input and output, the mathematical model of a given physical (or economical, or sociological, or biological, etc.) system:

1) AXIOMATIC APPROACH: Mathematico-physical (or economical, or sociological, or biological, etc.) analysis based on the laws which govern the underlying "applied" subject.

2) EXPERIMENTAL APPROACH: Data analysis where the main information about the system is obtained by measurements.

Very often the first way is used in the system classification stage. That is, we can use the mathematico-physical analysis, based on the physical laws, as some "a priori" information about the structure of a physical system S, and so a class C* can be determined (i.e., system classification). Since a class C* is available, a model
M in $C_\infty$ may be determined by means of experimental measurements in the system S using, for example, the meaning of equivalence stated before (i.e., system identification).

It is important to notice what we mean by "a priori" knowledge (or "a priori" information). We will use this term in a very wide sense. That is, it will mean all knowledge (or information) we have about the system before we start the identification procedure.

From this viewpoint, the system classification can be considered, by itself, as "a priori" knowledge of the system's structure. So, if the solution of the classification problem provides us with some available class of models, say $C_\alpha$, all information we have about the models in $C_\alpha$ will be considered as "a priori". This will be the case even if the system classification was carried out by experimental analysis (some authors use the term "a priori" only for information obtained by means of non-experimental analysis).

**Example 2.** Assume it was determined that a given system S is represented in terms of ordinary differential equation (1st. step: System Characterization) which is linear (in the usual meaning) and of order n (2nd. step: System Classification).

Let $C_n$ be the class of all linear ordinary differential equations of order n.

So, using "a priori" information about the system S we could classify it as of class $C_n$.

Each linear ordinary differential equation of order n is a model belonging to the class $C_n$. There are infinitely many models in $C_n$ (actually the class $C_n$ is uncountable), and they are completely
characterized. The only difference between any two of these models is just a set of $n+1$ real constants, which are the coefficients of each linear ordinary differential equation in $C_n$. These coefficients are called the **parameters** of the model.

The question is: Which are these $n+1$ parameters? In other words, which model $M$ (represented by its $n+1$ parameters) is equivalent to the system $S$?

### 1.2 - LUMPED PARAMETER SYSTEMS AND DISTRIBUTED PARAMETER SYSTEMS

Before introducing the idea of "lumped parameter systems" and "distributed parameter systems", we will present an informal discussion about the meaning of the terms "dynamic systems" and "parametric models".

A system $S$ is said to be **instantaneous** if it is represented in terms of a mathematical model $M$ whose the outputs $y_M$ at any time $t$ depends only on the input values at the same time $t$. No past or future values of the input will affect the present value of the output. This may also be called a **zero-memory** or a **memoryless** system. Otherwise the system is said to be **dynamic** and to have **memory**.

If a dynamic system is one whose the model outputs do not depend on future values of the input, it is called **causal** (or **physical**, or **nonanticipatory**). If this is not the case the dynamic system is called **noncausal** (or **nonphysical**, or **anticipatory**). If a causal system is such that the model outputs depend on the past inputs only over
a finite period, say $T$, then it is said to have finite-memory, and $T$ is its memory length. 5

Systems can be represented in many different ways, as we have already seen in the last section. Now we will introduce two important disjoint classes of models for the system characterization problem: By nonparametric we mean such models described in terms of impulse response, transfer functions, covariance functions, spectral densities, etc. By parametric models we mean those ones described in terms of state equation (or more generally "dynamical equations", which means the set of equations that describes the unique relation between the input, the output and state), differential (or difference) equations (both partial and ordinary), etc. Loosely speaking, a model is said to be parametric when it is completely characterized by a set of parameters (which can be constants, time and/or space varying, state independent, etc.). In rough terms, this means that the identification procedure is reduced to a problem of finding a certain number of parameters which completely determines the underlined model. Otherwise, it is said to be nonparametric (e.g., when the identification procedure is reduced to a problem of finding a certain number of parameters which completely determines the underlined model.

5 We have avoided discussing the term dynamical system in order to keep this introductory chapter on an informal level, reducing the abstract mathematical notation to a minimum. Generally, "dynamical" and "dynamic" are slightly different concepts. In few words: "dynamical" has roughly the same meaning as "causal". For a detailed mathematical definition and interpretation of the axioms, the reader is referred to [3] - [6].
tion procedure is reduced to the problem of finding an impulse response function belonging to some specific function space). 6

When the system under study is represented by a parametric model, the terms parametric estimation 7 and structure identification are sometimes used to specify what we are calling "system identification" and "system classification", respectively.

Our main subject in this work will be the identification of a certain type of parametric models. So, it would seem to be a good choice to use the term "parametric estimation" instead of "system identification". But we will avoid (where possible) using the word "estimation" to specify an identification problem, reserving this term only for the problem of "state estimation". Later in this chapter, we will present some comments about the confrontation between the problems of system identification and state estimation.

Now let us return to the main topic of this section. Many authors [10] - [16] define lumped parameter systems and distributed parameter systems in some slightly different ways. Sometimes, the

---

6 The nonparametric representation has the advantage that it is not necessary to specify the order of the model explicitly. They are, however, intrinsically infinite-dimensional models. Interesting aspects of parametric versus nonparametric models can be found in the literature on time-series analysis [7] - [9].

7 The term "parametric estimation" is also used in some wider sense, even when the models are not classified as parametric ones but the identification problem is reduced to that of finding some unknown parameters (e.g., in the determination of a transfer function which is completely known up to a finite set of constant parameters).
properties which are used by one author as definition, are used by another one as consequence and "vice versa". For the purpose of our further studies a very brief and simple definition will be sufficient:

"A dynamic system that can be represented in terms of an ordinary differential (or difference) equation will be called a lumped parameter system (LPS). When it requires the use of partial differential equation to describe its dynamic behavior, it will be called a distributed parameter system (DPS)."

In brief words: LPS and DPS are characterized by finite and infinite-dimensional state space, respectively.

The meaning of the terms "lumped" and "distributed" can be better understood when the physical implications of the above definitions are more deeply analysed.

In a lumped parameter system the physical size of the system is not important, since the excitations are transmitted through the system instantaneously. This assumption is usually valid if the largest physical dimension of the system is small compared to the wavelength of the highest significant frequency considered. Also, in this case, the system can be decomposed into a finite number of components, each

---

8 The particular case of systems described by partial difference equation can be viewed as a result of an approximation method which reduces a DPS to a LPS.

9 Dynamic systems whose mathematical models are in the form of integral (or integro-differential) equation are also called DPS.
with a finite number of input and outputs. On the other hand, in a
distributed parameter system, the spatial configuration is important
and generally it has dimensions that are not small compared to the
shortest wavelength of interest.

Examples of physical systems that can be modeled by partial
differential equations will be considered in the next chapter. Supple­
mentary discussions concerning with applications and related topics
in DPS can be found in the literature dealing with the control prob­
lem (see, for example, [10] and [17]).

1.3 - SYSTEM IDENTIFICATION VERSUS STATE ESTIMATION

First of all, let us introduce the notion of state and
then the formulation of the state estimation problem.

The state notion: Actually, from a physical viewpoint, the
concept of state can be thought as a primitive one, and as such it
is not to be defined. Therefore some authors [18] - [26] present a
"formal definition" of state which is valid, and very useful indeed,
in the sense of giving a deeper insight about the meaning of state.
Generally speaking, "the state u of a dynamic system at time t = t_o,
is the amount of information at t_o that, together with any input w
belonging to an input space W and known for all t = t_o, determine
uniquely the "behavior" of the system". 10 But what does the term

10 Following our previous intention of keeping this introductory
chapter on an informal level, we have avoided discussing a more
sophisticated but precise definition of state based on abstract
mathematical concepts. The interested reader is referred to [27] - [31].
"behavior" mean in this case? It means "the state itself and the output" of the system. So, we may not use the above as a "precise definition" of state, since the intrinsic meaning of state is assumed to be known "a priori".

**Remark**: We talked about "state of a dynamic system". This is a slight abuse of nomenclature, since we have been using the terms "system" and "model" with different meanings. Actually the concept of state is inherent to that of "oriented abstract objects" [27] which means, in general terms, our mathematical models. Based on our previous terminology it would be more correct to say: state of a model $M$ that characterizes a dynamical system $S$. Or, when we are considering the equivalent model $M^*$, the "state of $S$" can be thought as the state of $M^*$. From now on, the term "state of a dynamic system" will be used in the above sense.

Some authors use (or abuse of) the term "system identification" or even "parameter estimation" to specify a state estimation problem. Our main goal in this section will be to emphasize the difference between these two concepts.

For sake of simplicity, we will concern ourselves with the particular problem of state estimation in lumped parameter systems (finite-dimensional case), that can be modeled by a linear ordinary differential equation (i.e., the Kalman-Bucy filter [32], [33]).

Currently there are lots of books [4], [34]-[68] at many different levels dealing with the state estimation problem in finite-dimensional case. The same is not true in the infinite-dimensional

---

11 For a general review see [69].
case, specially for distributed parameter systems described by partial differential equations [70]-[73]. A recent paper by Curtain [74] gives an unified survey of this field, emphasizing the mathematical problem of rigorously modelling distributed noise. For those readers who are familiar with Kalman-Bucy filter in lumped parameter systems, the quite readable but formal works of Meditch [75], [76] are suggested as background before becoming involved with the sophisticated mathematical aspects (such as Sobolev spaces and other necessary but non-trivial concepts) which are inherent to the study of state estimation for distributed parameter systems.

Problem Formulation: Let us consider a dynamic system S modeled by a linear ordinary differential equation, whose state as a function of time is an n-dimensional stochastic process \{u(t_1); t_1 \in T\}, where T is some appropriate index set (an ordered subset of the reals that has a minimum element called \(t_0\)). We are interested in knowing the value of \(u(t_1)\) for some fixed \(t_1\), but \(u(t_1)\) is not directly accessible to us. Suppose we can have access only to an observation process \{z(t); t_0 \leq t \leq T\} which is related to \(u(t_1)\) by means of a linear causal system.

Let us introduce some notation:

1) Denote an estimate of \(u(t_1)\) based on the measurements of the observation process \{z(t)\} by \(\hat{u}(t_1|t)\), such that

\[
\hat{u}(t_1|t) = K_{t_1} \{z(t)\}; \quad t_0 \leq z \leq t, \quad t \in T
\]

where \(K_{t_1}\) is some linear operator defined in the observation space.

2) Let \(\tilde{u}(t_1|t)\) be the estimation error which is defined by the relation

\[
\tilde{u}(t_1|t) = u(t_1) - \hat{u}(t_1|t)
\]
3) \( L[\hat{u}(t_1|t)] \) will be some admissible cost function of the estimation error. A typical example would be:

\[
L = L[\hat{u}(t_1|t)] = ||\hat{u}(t_1|t)||^2
\]

where \( || \cdot || \) means the usual norm in an n-dimensional Euclidean space.

4) Since \( u(t_1) \) and \( \hat{u}(t_1|t) \) are random vectors, it follows that \( \hat{u}(t_1|t) \) will also be a random vector and so \( L \) will be a random function. In order to get a useful measure of the error, we can define a performance criterion \( J \) as the mean value of \( L \), that is

\[
J[\hat{u}(t_1|t)] = E\{L[\hat{u}(t_1|t)]\}
\]

where \( E\{ \cdot \} \) stands for the expectation of a random variable in the usual meaning.

5) We say that an estimate \( \hat{u}(t_1|t) \) which minimizes \( J[\hat{u}(t_1|t)] \) is a "best" or optimal estimate.

6) The linear operator \( K_{t_1} \), which gives us the optimal estimate will be called the "best" or optimal linear filter.

**Problem statement:** Given the measures of the observation process \( \{z(t); t_0 \leq t < t^* \}; t \in T \}, \) determine the "best" estimation \( \hat{u}(t_1|t) \) of \( u(t_1) \); or equivalently, determine the "best" linear filter \( K_{t_1} \).

If \( t_1 > t \), the problem is one of prediction; if \( t_1 = t \), one of filtering; and if \( t_1 < t \), one of smoothing or interpolation.

So we have two distinct optimizations problems namely system identification and state estimation. The system identification (or parametric estimation, as we are dealing with parametric models) is concerned with the problem of finding a set of parameters that specifi-
3) \( L[u(t|t)] \) will be some admissible cost function of the estimation error. A typical example would be:

\[
L = L[u(t|t)] = ||\tilde{u}(t|t)||^2
\]

where \( || \cdot || \) means the usual norm in an \( n \)-dimensional Euclidean space.

4) Since \( u(t^1) \) and \( \tilde{u}(t^1|t) \) are random vectors, it follows that \( \tilde{u}(t^1|t) \) will also be a random vector and so \( L \) will be a random function. In order to get a useful measure of the error, we can define a performance criterion \( J \) as the mean value of \( L \), that is

\[
J[u(t^1|t)] = E[L[u(t^1|t)]]
\]

where \( E[ \cdot ] \) stands for the expectation of a random variable in the usual meaning.

5) We say that an estimate \( \hat{u}(t^1|t) \) which minimizes \( J[u(t^1|t)] \) is a "best" or optimal estimate.

6) The linear operator \( K_{e_{t^1}} \) which gives the optimal estimate will be called the "best" or optimal linear filter.

**Problem statement:** Given the measures of the observation process \( \{ z(t); t_0 < t < T \} \), determine the "best" estimation \( \hat{u}(t^1|t) \) of \( u(t^1) \), or equivalently, determine the "best" linear filter \( K_{e_{t^1}} \).

If \( t_1 > t \), the problem is one of prediction; if \( t_1 = t \), one of filtering; and if \( t_1 < t \), one of smoothing or interpolation.

So we have two distinct optimizations problems namely system identification and state estimation. The system identification (or parametric estimation, as we are dealing with parametric models) is concerned with the problem of finding a set of parameters that speci-
fies a model, say $M^*$, which is equivalent in some sense to the system $S$ (or, which is the "best" model, based on some performance criterion, among all models $M$ belonging to a pre-selected class of models $C_M$).

Differently, the state estimation concerns with the problem of finding the "best" estimate $\hat{u}$ of the state $u$ of a system $S$, which is presupposed to be fully characterized by a completely known model $M^*$.

In recent years many researchers dedicated a great deal of their attention to the identification problem for lumped parameter systems. A large number of books [35],[49],[53],[60],[62],[64],[66],[77]-[80], surveys [81]-[91] and comparisons of different methods [92]-[98], were written about this subject. On the other hand, the bibliography on identification of distributed parameter systems is not so large.

Very few books [60] deal, even superficially, with this subject; and very often the state estimation problem in systems described by partial differential equations is wrongly termed "identification" (or even, "parameter estimation"!). In the next chapter we present a survey of this field.\footnote{For previous surveys in DPS identification see [90] and [99].}
CHAPTER 2

DISTRIBUTED PARAMETER SYSTEMS IDENTIFICATION: A SURVEY

This chapter treats the parameter identification problem in distributed systems. The various identification methods are grouped into three disjoint classes, namely: "Direct Methods", "Reduction to a Lumped Parameter System" and "Reduction to an Algebraic Equation". Under this classification we give a general survey of the main approaches to the problem of identifying distributed parameter systems.

The meaning of "parametric models", "distributed parameter systems" and "systems identification" are to be understood as introduced in the previous chapter. Standard abbreviations such as DPS: Distributed Parameter System(s)
LPS: Lumped Parameter System(s)
ODE: Ordinary Differential Equation(s)
PDE: Partial Differential Equation(s)
will be used in this and later chapters.

2.1 - INTRODUCTION

It has become customary to begin a survey paper dealing with any problem in DPS, by remarking on two fundamental points:

1) Problems involved with DPS are much more difficult than those with LPS.
2) Very little literature has been written about DPS, compared with what has been done for LPS.

Actually these remarks still remain valid, but they now have much less significance than they had one or two decades ago. Not only are a large number of research papers being published in this field but also new techniques in Modern Mathematical System Theory have helped to lessen the gap between LPS and DPS identification methods.

Basically the main theoretical difficulty for identifying systems described by PDE, is due to the infinite dimensionality of the state space. Two approaches are normally used to face this problem; 1) Approximation of the infinite-dimensional model by a finite-dimension one, and 2) application of optimization techniques directly to the infinite-dimensional model. Recent works on optimization in abstract spaces, which contain the DPS and LPS identification as particular cases, have simplified the general concept of this later approach.

In modeling DPS, different authors assume different classes of parametric models, each one representing a particular case adapted to a specific physical system. The "best" choice for a class of models $C_\alpha$ would be a sufficiently large class, such that all DPS described by PDE could be represented by models $M$ belonging to some subclass of $C_\alpha$. This ideal assumption is not usually satisfied in practice; mainly because of the great difficulty in developing an identification method for such a wide class and, at the same time being applicable in non-restrictive conditions (such as on-line identification, normal operating record, noisy observations, finite number of measurements, nonzero input, random inputs, etc.). As it will be seen in further sections, the identification methods for general models present one or
more of those restrictions, either from theoretical or practical viewpoint. Discussions on mathematical description for DPS can be found in the literature dealing with the control problem (see, for example, [1] and [2]).

Another difficulty arises when we are dealing with numerical methods for distributed models. If the solution of PDE is available, it comes very often in the form of infinite series which must be (for numerical computation purposes) approximated by a finite one. If the explicit solution is not available, some approximation technique (such as finite-differences) will be required for simulation. So, sooner or later, we will be faced with approximation problems for numerical implementation of identification methods in DPS. Discussions about this topic can be found in [3]. For numerical methods in PDE see, for example, [4]-[14] (also see references in chapter 3 - part II).

It is obvious that in physical applications, the DPS identification is a more complex problem than LPS identification. One of the main reasons for that is due to the impossibility of taking measurements by using an infinite number of sensors continuously located all over the spatial domain. In this way, some kind of approximation may be (and usually is) necessary when dealing with real applications.

Concerning the second remark: The literature discussed in this chapter contains over 100 entries related only with DPS identification problem, and it is not exhaustive. Although lots of recent papers in this field are continuously appearing, the number of books (even those that dedicate few sections to the subject [15]) is still very scarce.
Before reviewing the various methods for identification we discuss briefly the underlying motivations.

Physical systems that can be modeled by PDE (i.e., distributed parameter physical systems) are often encountered in engineering applications: Antennas, wave guides, propagation of electromagnetic and mechanical waves, microwave tubes, transmission lines, gas lines, many fluid flow systems, heat exchangers, heat insulating slabs, mechanical torsion bars, vibrating beams and strings, physical structures, transportation, environmental and geological systems, chemical and nuclear reactors, nuclear plasma devices, and charged particles accelerators; are just a few examples of systems whose state variables are distributed in space.

Also the majority of industrial and technological systems are characterized by the same fact (e.g., aerospace, petroleum, power, steel, glass, cement and chemical industries; ferrous and nonferrous metallurgy; drying and evaporation machinery; rolling mills; etc.).

There are a wide range of identification problems for parametric models in the real world of distributed systems. Some examples of fundamental physical parameters appearing in DPS are listed below:

1) Electromagnetic properties (e.g., conductivity, permittivity, permeability, charge density, etc.)

2) Thermal properties (e.g., specific heat, thermal conductivity, heat transfer coefficient, etc.)

3) Gas and fluid properties (e.g., density, diffusion constant, viscosity, expansion and compressibility coefficients, etc.)

4) Material properties (e.g., elasticity modulus)
5) Chemical properties (e.g., activation energy, reaction velocity constant, etc.)

Much of classical and modern, practical and theoretical engineering has been concerned with this basic problem. The first efforts in identifying such fundamental parameters has been performed under rigorously controlled laboratory conditions (normally off-line identification assuming noiseless measurements). In this survey the major attention in given to the problem of identifying coefficients in parametric models\(^1\) for DPS from sequential data (time series), by using identification methods which can be performed under less restrictive environmental and computational conditions.

2.2 - CLASSIFICATION OF METHODS

In the next section we will discuss several identification procedures for DPS. Although each one of them treats the problem under different conditions, they can be grouped into three different classes.

**CLASS \(\Gamma_1\): (Direct Methods)** Consists of those methods that use optimization techniques directly to the distributed (infinite-dimensional) model.

**CLASS \(\Gamma_2\): (Reduction to a LPS)** Consists of those methods that reduce the DPS (described by a PDE) to a continuous or discrete-time LPS (described by ODE or difference equation).

\(^1\) Such as DPS driven by random inputs and observed through noisy measurements, experimental data, normal operating records, recursive on-line identification algorithms, etc.).
CLASS $\Gamma_3$: (Reduction to an Algebraic Equation) Consists of those methods that reduce the PDE to an algebraic equation. (If the time variable is involved, the class $\Gamma_3$ may be viewed as a subclass of $\Gamma_2$ when a finite time interval is discretized).

The methods in classes $\Gamma_2$ and $\Gamma_3$ are characterized by two stages (as opposed to those methods in class $\Gamma_1$ which have a "single" stage). The first stage is concerned with techniques for approximating the infinite-dimensional state space to a finite-dimensional one, and the second with techniques for parametric estimation.

It is important to note that, in the case of $\Gamma_1$, the techniques for parametric estimation are generally applied after that numerical approximations have been carried out. In this way, these techniques apply to finite-dimensional systems and not all present an infinite-dimensional analysis. Actually, from this viewpoint, the "single" stage characterizing the methods in class $\Gamma_1$ could be split into two sub-stages: The first one going up to the point where numerical approximation are used for computational purposes (in this sub-stage the methods work for infinite-dimensional spaces). The second is concerned with techniques used for parametric estimation, which are applied after that point. (In this second sub-stage, the majority of the methods do not present an infinite-dimensional treatment).

Since our classification is based on whether or not a method reduces the infinite-dimensional state space to a finite-dimensional one in order to apply known identification techniques, these two sub-stages appearing in class $\Gamma_1$ will not be emphasized.

Fig. 3 shows a diagram representing these three classes, where the paths (1), (2) and (3) correspond to $\Gamma_1$, $\Gamma_2$ and $\Gamma_3$ respec-
The bifurcation (2-A) and (2-B), appearing on path (2), characterizes the possibility of reducing the DPS to a discrete or continuous-time LPS. The link (2-3) represents the possibility of reducing to an algebraic equation via an ODE.

Fig. 3: Classification of the Identification Procedures for DPS.
tively. The bifurcation (2-A) and (2-B), appearing on path (2) characterizes the possibility of reducing the DPS to a discrete or continuous-time LPS. The link (2-3) represents the possibility of reducing to an algebraic equation via an ODE.

Fig. 3: Classification of the Identification Procedures for DPS.
Concerning the methods that will be discussed on section 2.3, we can classify them as follows (where the numbers between brackets correspond to the references listed at the end of this work).

**CLASS $\Gamma_1$:** Path (1): $[56]-[61],[75],[80],[81],[83],[88],[91]-[104]$.  
**CLASS $\Gamma_2$:**  
- Path (2-B): $[57],[65],[70]-[72],[76],[78],[84],[87],[89],[90]$.  
**CLASS $\Gamma_3$:**  
- Path (3): $[45],[46],[54],[55],[86]$.  
- Path (2-3): $[66]$.  

The majority of the methods belonging to $\Gamma_2$ and $\Gamma_3$ use the following techniques for stage I:

**Path (2-A):** Finite-differences\(^2\): $[50]-[53],[85]$.  
- Method of lines\(^3\): $[57],[87]$.  
- Method of characteristics (e.g., see $[73],[74]$): $[70]-[72]$.  

**Path (2-B):**  
- Galerkin's method (e.g., see $[71],[101]$): $[76]-[79]$.  
- First order perturbation: $[84]$.  
- Integral transformations: $[65]$.  

**Path (2-3):** Method of line + Integral transformations: $[66]$.  

**Path (3):**  
- Finite-differences: $[54],[55]$.  
- Integral transformations: $[45],[46],[86]$.  

Finally we mention the most pertinent techniques used for parametric estimation in the DPS identification problem. These tech-

\(^2\) Very often finite-difference techniques for approximating partial derivatives (e.g., see $[4],[8],[13]$) are also used by methods of class $\Gamma_1$ for numerical implementation (e.g., see $[80],[83],[97],[101]$).  
\(^3\) Basically, finite-difference techniques applied only over the spatial domain (e.g., see $[67]-[69]$).  

28
niques are concerned with stage II for methods in \( \Gamma_2 \) and \( \Gamma_3 \), and with the "single" stage for methods in \( \Gamma_1 \).

1) Gradient (Conjugate Gradient - Steepest Descent): [56]-[61], [71], [76]-[79], [80], [81], [97], [100], [101].

2) Stochastic Approximation: [50]-[53], [70], [72], [85].

3) Least Squares (Sequential Weighted Least Squares - Least Squares Filtering): [58], [59], [88], [89].

4) Kalman-Filter: [84].

5) Nonlinear Filtering: [77]-[79].

6) Nonlinear Programming: [83].

7) Maximum Likelihood: [51], [103].

8) Statistical Decision Theory: [51].

9) Pattern Search: [77], [79].

10) Quasilinearisation: [57], [61], [90].

11) Methods for Solution of Algebraic Equations: Class \( \Gamma_3 \).

Generally, in case of methods in class \( \Gamma_1 \), these estimation techniques are not developed using infinite-dimensional analysis, as commented before. For an interesting study concerning infinite-dimensional gradients of functionals over an infinite-dimensional parameter space, see Chavent [100].

2.3 - A CONCISE GENERAL REVIEW

The first attempts to identify parameters in DPS were mainly due to investigations dealing with the "Inverse Problem in Heat Trans-
Techniques are concerned with stage II for methods in $\Gamma_2$ and $\Gamma_3$, and with the "single" stage for methods in $\Gamma_1$.

1) Gradient (Conjugate Gradient - Steepest Descent): [56]-[61], [71], [76]-[79], [80], [81], [97], [100], [101].
2) Stochastic Approximation: [50]-[53], [70], [72], [85].
3) Least Squares (Sequential Weighted Least Squares - Least Squares Filtering): [58], [59], [88], [89].
4) Kalman-Filter: [84].
5) Nonlinear Filtering: [77]-[79].
6) Nonlinear Programming: [83].
7) Maximum Likelihood: [51], [103].
8) Statistical Decision Theory: [51].
9) Pattern Search: [77], [79].
10) Quasilinearisation: [57], [61], [90].
11) Methods for Solution of Algebraic Equations: Class $\Gamma_3$.

Generally, in case of methods in class $\Gamma_1$, these estimation techniques are not developed using infinite-dimensional analysis, as commented before. For an interesting study concerning infinite-dimensional gradients of functionals over an infinite-dimensional parameter space, see Chavent [100].

2.3 - A CONCISE GENERAL REVIEW

The first attempts to identify parameters in DPS were mainly due to investigations dealing with the "Inverse Problem in Heat Trans-
fer" [16]-[27]. Two different approaches were initially used to attack the identification problem: 1) Methods based on the analytical solution of PDE [28]-[38],[141], and 2) methods developed in frequency domain [39]-[42] (e.g., identification of some coefficients of a transfer function approximating a linear model for distributed systems).

The purpose of this section is a brief review of the more relevant literature dealing with parameter identification in distributed systems. The bibliography mentioned here has been published during the last decade, and it is widely available.

Some survey papers have already appeared in this field.

Kozhinsky and Rajbman [43] and Rajbman [44], discuss the work done in the Soviet Union. They present an extensive list of references and lots of applications. A general inspection of system identification problems (both for LPS and DPS) is also considered [44], including analysis of mathematical models accuracy [43].

A recent survey was presented by Goodson and Polis [3]. They consider a "step by step" approach which treats the identification prob-

---

4 Although the label "Inverse Problem" has originated from classical solutions of identification problems, it is still used in very recent papers [92]-[94], [103], [105], [106], where a modern abstract approach is applied to solve the old problem.

5 As observed in the last chapter, these models are classified as non-parametric ones. Since we are interested only in parametric models, such identification methods will not be discussed here.

6 This approach has been considered previously by the same authors in [77],[79]. See also [141] for a latter and concise version of [3].
lem by separating it into seven independent subproblems. As well as a collection of five suggestions for further research in this field, they also present a very interesting bibliographical analysis where the major techniques used for identifying DPS are displayed together with their respective frequencies of usage. The state estimation problem is also considered.

A large number of methods have been developed to solve the problem. The great majority of them is restricted to particular cases (such as specific classes of parametric models, boundary conditions, input signals and output measurements).

Ferdreaouille and Goodson [45],[46] used integration by parts (an extension of Shinbrot's technique) to reduce the distributed model to a set of algebraic equations. The method is applicable to nonlinear models (where extraneous terms may be included) and the case of space-varying coefficients is also considered. Normal operating records and experimental data may be used. The main limitations of the method are: 1) It is not convenient for on-line applications, 2) each model has to be considered separately, 3) it has a restricted applicability, since it requires the choice of a suitable function (which is not always easy or possible) to multiply the PDE in order to perform the integral transformation and, 4) no noise observations were assumed.

7 The "modulating function method" of reducing ODE to a set of algebraic equations proposed by Shinbrot [47], was previously utilized by Loeb and Cahon [48], and Takaya [49] for identifying parameters in LFS.
Zhivoglyadov and Kaipov [50] applied finite-difference [4], [8],[13] techniques to reduce a time-invariant DPS (whose model is not necessarily linear) to a discrete-time LPS. Estimates for constant unknown parameters were obtained by minimizing a performance criterion. In this way, they compute the gradient of a cost function for each different model. Assuming noisy observations taken at discrete points in space, a stochastic approximation algorithm is used as a searching scheme for finding these estimates. The method is suitable for on-line applications. In [51] they develop some DPS identification methods based on statistical decision theory, maximum likelihood and stochastic approximation. In [52] the accuracy of a stochastic approximation method is analysed. The work of Zhivoglyadov and his group is summarized in [53].

Collins and Khatri [54], [55], assumed a deterministic class of DPS described by a time-varying model which can be nonlinear in the dynamics, but must be linear in the parameters. Based on this linearity, the q-constant vector to be identified is placed in an explicit form, and finite-difference techniques are used to approximate the partial derivatives. This procedure reduces the identification problem to that of solving a q-dimensional linear algebraic equation, where a q x q -matrix must be inverted. The observations are taken at a finite number (>q) of points in time and/or space, and extraneous terms may be included in the original model. Normal operating data and on-line identification may be used, but measurements are assumed to be noiseless.

Seinfeld and Chen [56]-[61] developed methods for nonlinear DPS identification based mainly on the steepest descent algorithm. In [56], systems modeled by hyperbolic or parabolic PDE with constant pa-
parameters are identified. A steepest descent algorithm is used as an optimal scheme for minimizing a quadratic error criterion, where the concept of sensitivity coefficients is introduced. Analysis of output transformations and observability for DPS are developed, but no noise in the observations is assumed. The method is not convenient for on-line applications and requires integration of PDE. In [57] and [61] they used steepest descent, quasilinearization and collocation techniques for nonsequential (off-line) estimation of constant parameters. Optimal location of measurements is also considered. The identification of space-varying parameters is developed in [58] and [59], where two techniques are presented: steepest descent and least-squares filtering. In [60] they considered estimation of time and/or space-varying parameters, and also of those which govern the spatial domain. The identification problems are formulated as optimal control problems and necessary conditions for optimality are derived. The techniques of steepest descent and conjugate gradient are applied. In [57]-[61] they assume noisy observations.

Tzafestas [65] considered the estimation of constant parameters in linear stochastic DPS, which can be reduced to an equivalent LPS by means of integral transformations. Although the method is basically the same used by Perdreauville in [46] and no noisy observations have been considered, this was one of the first attempts to identify DPS driven by random inputs. He assumes a discrete-time analogue of the original model and uses integration by parts to get a canonical LPS. In [140] a particular class of discrete-time DPS is considered.

---

8 Studies concerning observability in DPS can be found in [1],[62]-[64].
rameters are identified. A steepest descent algorithm is used as an optimal scheme for minimizing a quadratic error criterion, where the concept of sensitivity coefficients is introduced. Analysis of output transformations and observability for DPS are developed, but no noise in the observations is assumed. The method is not convenient for online applications and requires integration of PDE. In [57] and [61] they used steepest descent, quasilinearization and collocation techniques for nonsequential (off-line) estimation of constant parameters. Optimal location of measurements is also considered. The identification of space-varying parameters is developed in [58] and [59], where two techniques are presented: steepest descent and least-squares filtering. In [60] they considered estimation of time and/or space-varying parameters, and also of those which govern the spatial domain. The identification problems are formulated as optimal control problems and necessary conditions for optimality are derived. The techniques of steepest descent and conjugate gradient are applied. In [57]-[61] they assume noisy observations.

Tzafestas [65] considered the estimation of constant parameters in linear stochastic DPS, which can be reduced to an equivalent LPS by means of integral transformations. Although the method is basically the same used by Perdreauville in [46] and no noisy observations have been considered, this was one of the first attempts to identify DPS driven by random inputs. He assumes a discrete-time analogue of the original model and uses integration by parts to get a canonical LPS. In [140] a particular class of discrete-time DPS is considered.

---

8 Studies concerning observability in DPS can be found in [1],[62]-[64].
Fairman and Shen [66] modified the method of Perdreaauville and Goodson [46] of reducing the model for DPS to a set of algebraic equations. They applied the method of lines to convert PDE into ODE. This way, they avoid two restrictions required in [46], namely: offline numerical integration and complete knowledge of the state, both over the spatial domain. In a second step they reduce the ODE to a set of algebraic equations by using the "moment functional method". This procedure has been applied to identification of constants parameters in one-dimensional wave and diffusion equations. A particular case of a time-varying coefficient was also considered. The observations were taken at a finite number of points in space, but were assumed to be noiseless.

Carpenter, Wozny and Goodson [70]-[72], used the method of characteristics [73], [74] to reduce a linear first-order PDE to a set of ODE. Estimates of unknown parameters (which may depend on the independent variables and states) were obtained by minimizing a quadratic performance criterion. Stochastic approximation algorithms were chosen as a recursive searching scheme for finding the estimates. They assumed noisy observations and limited available measurement transducers, but the on-line applicability of the method depends on the required performance criterion.

---

9 The method of lines (e.g., see [67]-[69]) reduces the DPS to a continuous-time LFS (i.e., difference-differential equations).

10 Basically, this method consists in multiplying the ODE by a suitable modulating function (they used a modified form of the Poisson probability density function) and then integrating by parts.
Ruban [75] presented an algorithm for identifying DPS by means of sensitivity functions. This method is similar to that proposed by Seinfeld in [56].

Polis, Goodson and Wozny [76]-[79] proposed a step by step approach to the DPS identification problem. They assumed an approximate solution for the distributed model, based on a finite set of orthogonal functions over the spatial domain. The Galerkin's criterion [7], [10] is used to reduce the PDE to a set of ODE. The constant parameters are then identified by means of known techniques for LPS identification. Three optimization schemes were considered to minimize a performance criterion: steepest descent, nonlinear filtering and pattern search. Decision for measurement locations in the spatial domain were taken based on the G-K observability [62]. Noisy observations and extraneous terms were considered.

Di Pillo and Grippo [80] applied the "epsilon technique" to estimate constant parameters and states in linear DPS. Noisy observations were taken at a finite number of points in the spatial domain, and finite-difference techniques were used for numerical implementation of the method. In [81] they proposed an alternative procedure, by using an approximate solution, to avoid finite-difference approximations.

Hamza and Sheiran [83] presented a method for identifying constant and time-varying parameters in DPS. Non-linear programming was applied to minimize a discrete version of an appropriate perfor-

---

11 Basically, the ε-technique consists in minimizing a new cost function, which is obtained by adding to the original one a penalty term. For details see [82].
mance criterion, where finite-difference techniques are used to approximate the partial derivatives. In the case of linear models and assuming the instantaneous error squared as a performance criterion, the method reduces to one similar to that proposed by Collins and Khatri in [55]. This identification procedure is suitable for on-line application and uses a limited number of sensors along the spatial domain. Examples considering noisy measurements, extraneous terms and experimental results were included.

Bhagavan and Nardizzi [84] considered the identification of DPS modeled by linear PDE with constant coefficients. First-order perturbations were used to reduce the problem to one of estimation in LPS by means of Kalman filtering. The method is suitable for on-line applications and assumes a finite number of noisy observations.

Several papers dealing with the identification problem in DPS were presented in IFAC symposiums and other international conferences. Some of them have already been commented on here.

Diamessis [86] used integral transformations and approximations by Chebychev polynomials to reduce a linear PDE with constant coefficients to a set of algebraic equations. The method is not suitable for on-line identification and does not assume noisy observations.

Luckinbill and Childs [87] applied the method of lines to reduce a quasilinear second-order PDE to an ODE, which is augmented by

12 In chapter 4 we propose a new identification method for a class of linear DPS operating in stochastic environment. The method is based on the stochastic approximation theory and a rather simplified version of it can be found in [85].
adjoining the constant parameters. A Newton-Raphson-Kantorovich expansion is used to solve the resulting linear boundary value problem. The identification procedure can operate under on-line conditions, but no noisy observations were considered.

Sherry and Shen [88] presented a method for parameter and state estimation in linear DPS, by using a sequential weighted least-squares algorithm. A finite number of noisy observations was considered, and the method is suitable for on-line applications.

Shridhar and Balasori [89] used splines to reduce the DPS to a continuous-time LPS. A recursive least-squares procedure was applied for parametric estimation.

Chaudhuri [90] proposed an identification method for DPS based on differential approximation and quasilinearization.

A modern abstract approach for DPS identification has been considered recently by Chavent [91]-[101] and others [102]-[106].

In [97] and [101], Chavent proposed an off-line identification method for DPS described by a general class of deterministic models, where no specific probabilistic treatment for noisy observations was considered. The method is developed by functional analysis techniques and based on the Lions’ [107] approach to control theory for DPS. It consists of minimizing a performance criterion (the quadratic error of output, which is nonquadratic with respect to the parameters), by using a conventional gradient technique (the steepest descent method was used). In order to compute the gradient of the performance criterion, he introduced the adjoint state (solution of adjoint state equation). The gradient is then derived as a functional of the adjoint and system states. Its computation requires the simul-
taneous solution of both system (given by the distributed model) and adjoint state equations. Fundamental problems in identification, such as existence, uniqueness and choice of minimization schemes, were discussed in some detail. Two types of models were considered: 1) Those with a finite number of constant parameters (finite-dimensional parameter space) and, 2) those with varying parameters as functions of independent variables, or states (infinite-dimensional parameter space). Applications were also included where discretization techniques, such as finite-differences, are applied for numerical implementation of the method, which was shown to work with a small number of measurements in space and/or time. In [100] he considered the identification of DPS modeled by parabolic PDE with space and state-varying parameters. Measurements are taken by a finite number of sensors located in the spatial domain. These sensors supply a mean value of the output over a small neighbourhood for each observation point. In this way, perturbations of measurements were considered, but a stochastic modeling for noisy observations is still lacking. As in [97], the method presents a rigorous mathematical treatment for minimizing a least square error criterion. An infinite-dimensional gradient of the performance criterion with respect to the parameters was defined, and expressed in terms of the system and adjoint states. The optimization algorithm was the steepest descent, and a detailed discussion on the uniqueness problem was also included.

Balakrishnan [102]-[104] considered the system identification problem (in particular, DPS) from a stochastic viewpoint. In [103] he presented a general abstract approach for identifying a class of linear DPS, previously considered by Lavrentiev [105] and Marchuk [106].
A stochastic formulation in Hilbert space was proposed, where additive white Gaussian noises are assumed to corrupt both input and observation process. The theoretical development was based on the semigroup theory of linear operators (as opposed to that proposed by Chavent in [97] and [100]), for time invariant systems operating under continuous-time assumption. The infinitesimal generator of a strongly continuous semigroup appearing in the model, was supposed to be dependent on unknown parameters. In order to obtain asymptotically unbiased and consistent estimates of those parameters, the "a posteriori" maximum likelihood technique was utilized. In [102] he considered the identification problem for both LPS and DPS, operating in a stochastic environment. The DPS case is just slightly mentioned, and shown to be included in the general model.

Other types of "identification" problem for DPS have appeared in the literature. Jones and Douglas [29], [30] identified a time-varying coefficient in a parabolic PDE. Since this coefficient necessarily appears in the boundary conditions and the identification method takes measurements only at the boundary, this problem is reduced to that of identifying boundary conditions. Ward and Goodson [108], [109] and Alvarado and Mukundan [139] also investigated the identification of boundary conditions. Wozny, Carpenter and Stein [110] presented a method for identifying Green's functions of DPS. Cannon [111] and Ikeda, Miyamoto and Sawaragi [142] considered the determination of unknown sources for a class of PDE. Saridis and Badavas [112], [113] identified solutions in DPS, but this was really a state estimation problem. The term "identification" was also used by Phillipson [114], [115] but again this was a state estimation problem.
Some practical subjects have stimulated several researches in this field. Problems related to physical structures, heating processes, transportation, economy, ecology, geology, chemical and environmental sciences; are just a few examples where applications and experimental results in identification of DPS have already been developed (see, for example, [16]-[26],[38],[39],[61],[97],[101],[116]-[136]).

Several conclusions can be drawn from the topics covered by this chapter. Some of them are presented in the next section. The classification introduced on section 2.2 can give us a general idea about the main techniques currently used for solving the identification problem in DPS. For a further collection of observations and comments concerning both with system identification and state estimation in DPS, the reader is referred to a previous survey by Goodson and Polis [3], [143].

2.4 - CONCLUSIONS

From what was discussed here we can select some basic points which deserve to be emphasized.

1) The reduction to finite-dimensional state space seems to be the most popular method used in the DPS identification problem (Methods of class $T_2$; the distributed model is approximated by a lumped one before any optimization is carried out).

2) Among the approximation techniques, finite-differences is one of the most used. Other techniques for dealing with distributed models, such as finite element method (e.g.,[137]), should be investigated for identification purposes.
3) As remarked before, sooner or later, we are faced with some approximation problem (either for reducing to a finite-dimensional state space, for numerical implementation, or for physical applications). The question of when to use approximation techniques, before or after the optimization, has no final answer yet. Athans [138] suggests that any approximation should be applied as late as possible in order to retain the distributed nature of the model, until numerical results are required.

4) Among the optimization techniques for parametric estimation the gradient method is the most popular. Its "probabilistic version"\textsuperscript{13}, stochastic approximation, have also been successfully used by several authors.

5) No general method for a large class of models operating in non-restrictive conditions has been developed, and only a few authors consider the case of stochastic environment (random inputs and noisy observations).

6) There has been a lot of recent literature in this field, but it is still difficult to make any comparisons, because of the different models considered.

7) Although other survey papers dealing with DPS identification have already appeared [3],[44],[143], this seems to be the first attempt to inspect the more relevant techniques in this field without confusing two different problems, namely: system identification, and state estimation.

\textsuperscript{13} From a particular viewpoint, the stochastic approximation algorithms may be thought as a "probabilistic version" of the gradient method.
CHAPTER 3

MATHEMATICAL PRELIMINARIES

PART I: MODELS FOR DPS

We have previously defined a DPS as a dynamic system that can be modeled by a PDE. We begin this chapter by presenting a formal description of such models, emphasizing a special subclass of linear models which will be considered in chapter 4 for identification purposes.

3.1 - ON A GENERAL CLASS

A general class of models for DPS described by PDE can be formally written as follows:

Dynamic equation: \[ L(u, a, s) = f(s) + w(s) \]  \hspace{1cm} (1)

Boundary conditions: \[ L_\Gamma(u)|_{x=x'} = f_\Gamma(x', t) + w_\Gamma(x', t) \]

Initial conditions: \[ L_0(u)|_{t=0} = f_0(x) + w_0(x) \]

where:

i) \( x \in X \), a simply connected open set in \( \mathbb{R}^n \); the spatial domain.

ii) \( x' \in \Gamma \), the boundary of \( X \).

42
iii) $t \in \mathbb{Z}$, an interval which can be $(0,T]$, or $(0,\infty)$: the time domain.

iv) $s = (x,t) \in \Omega = X \times \mathbb{Z}$.

v) $u(s)$ is a real-valued function on $\Omega$ belonging to an appropriate function space $H(\Omega)$: the dependent variable.

vi) $a(s)$ is a vector with a finite number of components which are real-valued functions on $\Omega$ belonging to an appropriate space: the parameter vector.$^1$

vii) $f(s), f_p(x',t)$ and $f_o(x)$ are real-valued functions on $\Omega, \Gamma \times \mathbb{Z}$ and $X$, belonging to appropriate function spaces: the input, boundary and initial functions, respectively.

viii) $w(s), w_p(x',t)$ and $w_o(x)$ are real-valued random fields$^2 \{w(s); s \in \Omega \subseteq \mathbb{R}^{n+1}\}, \{w_p(x',t); (x',t) \in \Gamma \times \mathbb{Z} \subseteq \mathbb{R}^{n+1}\}$ and $\{w_o(s); x \in X \subseteq \mathbb{R}^n\}$: disturbance processes corrupting the input, boundary and initial functions, respectively.

ix) $L, L_p$ and $L_o$ are partial differential operators. $L_p$ and $L_o$ are concerned with the boundary and initial conditions, and $L$ represents a parametric distributed model.

A brief word about what we mean by "appropriate" space: The appropriate space associated with the parameter vector $a(s)$ (i.e., the parameter space) will become clear later in the chapter 4 where the identi-

---

$^1$ A more general case can be considered, where the parameter vector $a(u,s)$ also depends on the dependent variable $u(s)$.

$^2$ The term "random field" is used to denote a collection of random variables indexed by points taking values in a subset of $\mathbb{R}^n$, as a natural extension of the concept of stochastic processes [1].
iii) $t \in \mathbb{Z}$, an interval which can be $(0,T]$, or $(0,\infty)$: the time domain.

iv) $s = (x,t) \in \Omega = \mathbb{X} \times \mathbb{Z}$.

v) $u(s)$ is a real-valued function on $\Omega$ belonging to an appropriate function space $H(\Omega)$: the dependent variable.

vi) $a(s)$ is a vector with a finite number of components which are real-valued functions on $\Omega$ belonging to an appropriate space: the parameter vector.\(^1\)

vii) $f(s)$, $f_\Gamma(x',t)$, and $f_0(x)$ are real-valued functions on $\Omega$, $\Gamma \times \mathbb{Z}$ and $\mathbb{X}$, belonging to appropriate function spaces: the input, boundary and initial functions, respectively.

viii) $w(s)$, $w_\Gamma(x',t)$, and $w_0(x)$ are real-valued random fields\(^2\) $\{w(s); s \in \Omega \subset \mathbb{R}^{n+1}\}$, $\{w_\Gamma(x',t); (x',t) \in \Gamma \times \mathbb{Z} \subset \mathbb{R}^{n+1}\}$, and $\{w_0(x); x \in \mathbb{X} \subset \mathbb{R}^n\}$: disturbance processes corrupting the input, boundary and initial functions, respectively.

ix) $L$, $L_\Gamma$, and $L_0$ are partial differential operators. $L_\Gamma$ and $L_0$ are concerned with the boundary and initial conditions, and $L$ represents a parametric distributed model.

A brief word about what we mean by "appropriate" space: The appropriate space associated with the parameter vector $a(s)$ (i.e., the parameter space) will become clear later in the chapter 4 where the identi-
fication problem is formulated. Of course, the structure of this space depends on the particular approach used for a specific identification problem. On the other hand, an appropriate space containing the independent variable $u$ (i.e., the state space) is a function (or distribution) space which comes to be suitable to deal with a specific problem in PDE (e.g., some Sobolev space $H(\Omega)^3$). Considering our particular approach to the DPS identification problem (see chapter 4), there will be no need to go into details related with $H(\Omega)$; and there are two basic reasons for that:

1) We assume the existence and uniqueness of the solution for a given DPS modeled by a particular PDE.

2) We consider (for identification purposes) an approximated finite-dimensional version for modeling DPS.

3.2 - LINEAR MODELS

A class of linear models of $n$th order can be written, most generally, as follows:

$$L(u, a, n) = \sum_{i=1}^{n+1} a_i(s) \frac{\partial^m}{\partial t^i \partial x_1^{i_1} \cdots \partial x_n^{i_n}} u(s)$$

3 Roughly speaking: $H(\Omega)$ is a Banach space of functions on $\Omega$, equipped with a suitable norm, such that all partial derivatives of $u$ up to the highest order involved in the model $L$ are in $L^p(\Omega)$. If $p = 2$, $H(\Omega)$ is a Hilbert space. For details see, for example, [2] and [3].
fication problem is formulated. Of course, the structure of this space depends on the particular approach used for a specific identification problem. On the other hand, an appropriate space containing the independent variable $u$ (i.e., the state space) is a function (or distribution) space which comes to be suitable to deal with a specific problem in PDE (e.g., some Sobolev space $H(\Omega)$). Considering our particular approach to the DPS identification problem (see chapter 4), there will be no need to go into details related with $H(\Omega)$; and there are two basic reasons for that:

1) We assume the existence and uniqueness of the solution for a given DPS modeled by a particular PDE.

2) We consider (for identification purposes) an approximated finite-dimensional version for modeling DPS.

3.2 - LINEAR MODELS

A class of linear models of $M$th order can be written, most generally, as follows:

$$L(u, a, s) = \sum_{i \leq n+1} a_i(s) \frac{\partial^m}{\partial t^i \partial x_1^{i_1} \cdots \partial x_n^{i_n}} u(s)$$

3 Roughly speaking: $H(\Omega)$ is a Banach space of functions on $\Omega$, equipped with a suitable norm, such that all partial derivatives of $u$ up to the highest order involved in the model $L$ are in $L^p(\Omega)$. If $p = 2$, $H(\Omega)$ is a Hilbert space. For details see, for example, [2] and [3].
where:

i) $M$ is a finite integer.

ii) $I = \{0, 1, \ldots, N\}$.

iii) $i_j \in I; \forall j = 0, 1, \ldots, n$.

iv) $i = (i_0, i_1, \ldots, i_n) \in I^{n+1}$; an $(n+1)$-tuple index.

v) $m = \sum_{j=0}^{n} i_j$ and such that $0 \leq m \leq M$.

So, in this case, the dynamic equation (1) can be written in the form

$$a_{i^*}(s) \frac{\partial^M}{\partial t^M} u(s) = \sum_{i \in I^{n+1}} a_i(s) \frac{\partial^m}{\partial t^i \partial x_1^{i_1} \cdots \partial x_n^{i_n}} u(s) + f(s) + w(s)$$

where $i^* = (N, 0, \ldots, 0) \in I^{n+1}$; or equivalently (assuming $a_{i^*}(s) \neq 0$,

$\forall s \in \Omega$) in a state representation:

$$\frac{\partial}{\partial t} u(s) = A(s) u(s) + b(f(s) + w(s))$$

where $A(s)$ is a $M \times M$-matrix of linear spatial-differential operators whose parameters may depend on $s = (x, t)$, $b$ is a vector in $R_M$ (e.g., $b = (0, \ldots, 0, 1)$) and $u(s)$ is a state vector with $M$ components.

Remark: We have defined $u(s)$ and $u(s)$ as "finite-size" vectors of functions representing parameters and state, respectively. It is important to note that:

1) The parameter space may be infinite-dimensional since any component $a_i(s)$ of $u(s)$ can be a function of $x$ for each time $t \in \mathbb{Z}$.

2) The state space is obviously an infinite-dimensional one, since for each time $t \in \mathbb{Z}$, the components of $u(s)$ are certainly functions of $x$.  

45
A particular case: In chapter 4 we will be particularly interested in the following special subclass of linear parametric models: 4

\[ L(u,x,t) = L_t^N u(x,t) - L_x^M u(x,t) \quad x \in X = \mathbb{R} \quad (2) \]

where \( L_t^N \) and \( L_x^M \) are linear partial differential operators such that

\[ L_t^N u(x,t) = \sum_{m=0}^{N} \beta(x) \frac{\partial^m}{\partial t^m} u(x,t) \]

\[ L_x^M u(x,t) = \sum_{m=1}^{M} \alpha(x) \frac{\partial^m}{\partial x^m} u(x,t) \]

and the time-invariant \((N+1)\)-dimensional parameter vector is given by

\[ a = a(x) = (\alpha(x), \xi(x)) \in \mathbb{R}^{M+1} \]

with

\[ \alpha(x) = (\alpha_1(x), \ldots, \alpha_M(x)) \in \mathbb{R}^M \]

\[ \xi(x) = (\xi_0(x), \ldots, \xi_N(x)) \in \mathbb{R}^{N+1} \]

4 For discussions (both from theoretical and practical viewpoints) on DPS whose models are included in such subclass, see for example [3]-[6].
The classical finite-difference method is a well known technique used to obtain approximations for partial differential equations, mainly for second-order equations (e.g., see [6]-[19]).

The main goal of this second part of chapter 3 is to introduce a brief discussion of finite-difference techniques for approximating higher order partial derivatives. No attempt will be made to present a rigorous treatment dealing with specific topics such as stability, errors, and other technical aspects concerning with numerical analysis of finite-difference techniques; since this subject is widely available in the current literature.5

The results obtained here will be used in the next chapter for reducing DFS to LPS.

3.3 - SHIFT, DIFFERENCE AND SLOPE OPERATORS

Notation:
1) The set of all nonnegative integers, of all positive integers, of all even integers (including zero) and of all odd integers are denoted by $\mathbb{Z}$, $\mathbb{Z}^+$, $\mathbb{Z}_e$ and $\mathbb{Z}_o$, respectively.

---

5 The interested reader is referred to the bibliography concerning with this second part of chapter 3, which is listed at the end of this work.
As usual the three dots (...) indicate a presumed understanding about what is omitted.

2) The symbols (−) and (″) denote integer valued functions defined on \( \mathbb{Z} \) as follows:

\[
\begin{align*}
\bar{m} &= \begin{cases} 
\frac{m}{2} & \text{if } m \in \mathbb{Z}_o \\
\frac{m+1}{2} & \text{if } m \in \mathbb{Z}_o 
\end{cases} \\
\hat{m} &= \begin{cases} 
\frac{m}{2} & \text{if } m \in \mathbb{Z}_o \\
\frac{m-1}{2} & \text{if } m \in \mathbb{Z}_o 
\end{cases}
\end{align*}
\]

Those functions will be extensively used in the remainder of this work and so we recall some of their main properties:

i) \( \bar{m} + \hat{m} = m \)

ii) \( \bar{m} - \hat{m} = \bar{m} \) \( \hat{m} + 1 = \hat{m} \)

iii) \( m+1 = \bar{m} + 1 \) \( \bar{m} - 1 = \hat{m} - 1 \)

iv) \( \bar{m} - \hat{m} = 2\bar{m} - m = m - 2\hat{m} = \begin{cases} 
0 & \text{if } m \in \mathbb{Z}_o \\
1 & \text{if } m \in \mathbb{Z}_o 
\end{cases} \)

Definition (D - 3.1): Let \( \mathcal{B}(\mathbb{R}) \) be the linear space of all bounded real-valued functions on \( \mathbb{R} \). We define the "p-th-order delta shift operator"
\[ S^p_\delta : B(R) \rightarrow B(R) \]

where \( \delta \) is a fixed positive real constant, as follows:

\[ S^p_\delta f(x) = f(x + p\delta) \]

for any \( f \in B(R) \) and \( p \in R \). \(^6\)

Remarks:

1) For any \( p \in R \), \( S^p_\delta \) is a linear operator on \( B(R) \) under the usual definition of addition and scalar multiplication.

2) Any finite set of operators \( \{S^p_\delta\} \) is linearly independent. That is, for any finite set of scalars \( \{\alpha_i\} \)

\[ \sum \alpha_i S^p_\delta = 0 \iff \alpha_i = 0 \text{ for all } i. \]

3) \( S^p_\delta \) is invertible for any \( p \in R \).

\[ S^p_\delta S^{-p}_\delta = S^{-p}_\delta S^p_\delta = S^0_\delta = I \]

where \( I \) stands for the identity operator on \( B(R) \).

Three special linear operators on \( B(R) \) are derived from \( S^p_\delta \) as follows:

**Definition (D - 3.2):** The "forward difference operator":

---

\(^6\) With \( \Theta = -p\delta \) we get \( S^p_\delta = S_\Theta \), the "delay" operator: \( S_\Theta f(x) = f(x - \Theta) \).

For further details on shift or delay operators see, for example, [20]. In the finite-differences literature the symbol \( E \) is used instead of \( S \) to denote shift (also called "displacement") operators.
\[ \Delta_{\delta} = S_{\delta}^1 - S_{\delta}^0 = S_{\delta} - I \ . \]

Definition (D - 3.3): The "backward difference operator":

\[ \Delta_{\delta}^{-1} = S_{\delta}^0 - S_{\delta}^{-1} = I - S_{\delta}^{-1} \ . \]

Definition (D - 3.4): The "m+1th-order centered slope operator":

\[ d_{\delta}^{m+1} = \frac{1}{\delta} \Delta_{\delta} (-1)^{m} d_{\delta}^{m} \ , \quad d_{\delta}^{0} = I \ . \]

Remark: \( \Delta_{\delta}^{-1} \) is not the inverse of \( \Delta_{\delta} \) since \( \Delta_{\delta} \Delta_{\delta}^{-1} = \Delta_{\delta}^{-1} \Delta_{\delta} \neq I \).

These operators, mainly \( S_{\delta}^{p} \) and \( D_{\delta}^{m} \), will be used for approximating higher order derivatives in the next section. But before going through that we need to prove the following results:

Proposition (P - 3.1):

\[ (S_{\delta}^{p} + a S_{\delta}^{q})^{m} = \sum_{i=0}^{m} a^{-i} \binom{m}{i} S_{\delta}^{mq + i(p-q)} \]

where \( p, q \) and \( a \) are reals (\( a \neq 0 \)), \( m \in \mathbb{Z} \) and

\[ \binom{m}{i} = m! / [(i!(m-i)!)] \ . \]

Proof:

Both cases, \( m=0 \) and \( m=1 \), are automatically satisfied by simple substitution. Now assume the following equivalent proposition:

\[ (a S_{\delta}^{q})^{-m} (S_{\delta}^{p} + a S_{\delta}^{q})^{m} = \sum_{i=0}^{m} a^{-i} \binom{m}{i} S_{\delta}^{i(p-q)} \ . \] (3)
Then
\[(aS_\delta^q)^-(m+1) (s_\delta^p + aS_\delta^q)^{m+1} = \]
\[= (aS_\delta^q)^-(m) (s_\delta^p + aS_\delta^q)^m a^{-1} s_\delta^{-q} (s_\delta^p + aS_\delta^q) = \]
\[= \sum_{i=0}^{m} a^{-i} \binom{m}{i} s_\delta^{i(p-q)} (a^{-1}s_\delta^{-q} + 1) = \]
\[= \sum_{i=0}^{m} a^{-(i+1)} \binom{m}{i+1} s_\delta^{i(p-q)} + \sum_{i=0}^{m} a^{-i} \binom{m}{i} s_\delta^{i(p-q)} = \]
\[= \sum_{i=1}^{m+1} a^{-i} \binom{m}{i-1} s_\delta^{i(p-q)} + \sum_{i=0}^{m} a^{-i} \binom{m}{i} s_\delta^{i(p-q)} = \]
\[= a^{0(m+1)} s_\delta^{0} + \sum_{i=1}^{m} a^{-i} \left[\binom{m}{i-1} + \binom{m}{i}\right] s_\delta^{i(p-q)} + a^{-(m+1)} \binom{m}{m} s_\delta^{(m+1)(p-q)}.\]

Since
\[\left\{\begin{array}{l}
\binom{m}{i-1} + \binom{m}{i} = \binom{m+1}{i} \\
\binom{m}{0} = \binom{m+1}{0} = 1; \quad \binom{m}{m} = \binom{m+1}{m+1} = 1
\end{array}\right.\]
we get:
\[(aS_\delta^q)^-(m+1) (s_\delta^p + aS_\delta^q)^{m+1} = \]
\[= a^{0(m+1)} s_\delta^{0} + \sum_{i=1}^{m} a^{-i} \binom{m+1}{i} s_\delta^{i(p-q)} + a^{-(m+1)} \binom{m+1}{m+1} s_\delta^{(m+1)(p-q)} = \]
\[= \sum_{i=0}^{m+1} a^{-i} \binom{m+1}{i} s_\delta^{i(p+q)} \]
which proves, by induction, the assumption (3). \(\square\)

**Particular cases:**

1) \(a = -1, p = 1\) and \(q = 0;\)
\[ \Delta^n_\delta = (s_\delta - I)^m = \sum_{i=0}^{m} (-1)^{m-i} \binom{m}{i} s^i_\delta. \]

2) \( a = -1, p = 0 \) and \( q = -1 \)

\[ \Delta^{-m}_\delta = (I - s^{-1}_\delta)^m = \sum_{i=0}^{m} (-1)^{m-i} \binom{m}{i} s^{i-m}_\delta. \]

**Proposition (P - 3.2):**

\[ D^m_\delta = \begin{cases} I & \text{if } m = 0 \\ \frac{1}{\delta^m} \prod_{i=1}^{m} (\Delta(-1)^{i-1}_\delta) & \text{if } m \in \mathbb{Z}_+ \end{cases} \]

where

\[ \prod_{i=1}^{m} \Delta(-1)^{i-1}_\delta = \Delta(-1)_\delta \Delta(-1)^2_\delta \ldots \Delta(-1)^{m-1}_\delta = \Delta(-1)^{m-1}_\delta \Delta(-1)^{m-2}_\delta \ldots \Delta(-1)_\delta \]

**Proof:**

a) \( m = 0 \) and \( m = 1 \), trivial by (D - 3.4).

b) \( m > 1 \): Assume

\[ D^m = \frac{1}{\delta^m} \prod_{i=1}^{m} (\Delta(-1)^{i-1}_\delta). \]

So, by (D - 3.4), we get:

\[ D^{m+1} = \frac{1}{\delta} \Delta(-1)^m_\delta \frac{1}{\delta^m} \prod_{i=1}^{m} (\Delta(-1)^{i-1}_\delta) = \]

\[ = \frac{1}{\delta^{m+1}} \prod_{i=1}^{m+1} (\Delta(-1)^{i-1}_\delta) \]

and the proof (by induction) is completed. \( \square \)
Lemma (1.3.1): The $m$th-order centered slope operator can be written in terms of a finite series of delta shift operators as follows:

$$D_m^\alpha = \frac{1}{\kappa^m} \sum_{i=0}^{m} (-1)^{m-i} \binom{m}{i} \Delta\delta_i^m; \quad m \in \mathbb{Z}$$

and this representation is unique.

Proof:

Existence:

a) $m = 0$ : $D_0^\alpha = \Delta_\delta^0 = I$ (trivial).

b) $m \in \mathbb{Z}_+$ : By (P-3.2) we have

$$\Delta^m D_\delta^\alpha = \prod_{i=1}^{m} \Delta^{-1} \delta^{i-1} = \Delta_\delta \Delta^{-1}_\delta \cdots \Delta^{-1}_\delta \delta^2 \Delta \delta = \begin{cases} \Delta_\delta^{-1} \cdots \Delta^{-1}_\delta \Delta_\delta & \text{if } m \in \mathbb{Z}_o \\ \Delta_\delta \Delta^{-1}_\delta \cdots \Delta^{-1}_\delta \delta^{-1} \Delta^{-1}_\delta \delta^{-1} & \text{if } m \in \mathbb{Z}_c \\ \end{cases}$$

$$= \begin{cases} (\Delta_\delta \Delta^{-1}_\delta)^{2m} & \text{if } m \in \mathbb{Z}_o \\ (\Delta_\delta \Delta^{-1}_\delta)^{-m} & \text{if } m \in \mathbb{Z}_c \\ \end{cases}$$

$$= (\Delta_\delta \Delta^{-1}_\delta)^m \begin{cases} I & \text{if } m \in \mathbb{Z}_o \\ \Delta_\delta & \text{if } m \in \mathbb{Z}_c \\ \end{cases}$$

But

$$\Delta_\delta \Delta^{-1}_\delta = (\delta - I) (\delta^{-1} - \delta) = \delta - 2I + \delta^{-1} = (\delta^{1/2} - \delta^{-1/2})^2,$$

then by (P-3.1) with $a = -1$ and $p = -q = \frac{1}{2}$ we get
\[(\Delta_{\delta} \Delta_{\delta}^{-1})^{m} = (S_{\delta}^{\frac{1}{2}} - S_{\delta}^{-\frac{1}{2}})^{2m} = \sum_{i=0}^{2m} (-1)^{i} \binom{2m}{i} (S_{\delta}^{\frac{1}{2}})^{i} (S_{\delta}^{-\frac{1}{2}})^{2m-i} . \]

Hence:

\[\delta_{\delta}^{m} \Delta_{\delta}^{m} = \sum_{i=0}^{2m} (-1)^{2m-i} \binom{2m}{i} \Delta_{\delta}^{i-m} \left\{ \begin{array}{ll}
I & \text{if } m \in \mathbb{Z}_e \\
\Delta_{\delta} & \text{if } m \in \mathbb{Z}_o
\end{array} \right. \]

b-i) \( m \in \mathbb{Z}_e \rightarrow 2m = m \) and the existence is proved for \( m \in \mathbb{Z}_e \).

b-ii) \( m \in \mathbb{Z}_o \rightarrow 2m = m-1 \) and \((-1)^{2m-i} = (-1)^{m-1-i} = (-1)^{m+1-i} \). So:

\[\delta_{\delta}^{m} \Delta_{\delta}^{m} = \sum_{i=0}^{m-1} (-1)^{m+1-i} \binom{m-1}{i} S_{\delta}^{i-m} (S_{\delta}^{-1}) = \]

\[= \sum_{i=0}^{m-1} (-1)^{m+1-i} \binom{m-1}{i} S_{\delta}^{i+1-m} - \sum_{i=0}^{m-1} (-1)^{m+1-i} \binom{m-1}{i} S_{\delta}^{i-m} = \]

\[= \sum_{i=1}^{m} (-1)^{m-i} \binom{m-1}{i-1} S_{\delta}^{i-m} + \sum_{i=0}^{m-1} (-1)^{m+1-i} \binom{m-1}{i} S_{\delta}^{i-m} = \]

\[= (-m+1) \binom{m}{0} S_{\delta}^{-m} + \sum_{i=1}^{m} (-1)^{m-i} \left[ \binom{m-1}{i-1} - (-1)^{m+1-i} \binom{m-1}{i} \right] S_{\delta}^{i-m} + \binom{m}{m} S_{\delta}^{m-m} = \]

\[= (-1)^{m} S_{\delta}^{-m} + \sum_{i=1}^{m} (-1)^{m-i} \binom{m}{i} S_{\delta}^{i-m} + \binom{m}{m} S_{\delta}^{m-m} = \]

\[= \sum_{i=0}^{m} (-1)^{m-i} \binom{m}{i} S_{\delta}^{i-m} \]

and the existence is proved for \( m \in \mathbb{Z}_o \).

Uniqueness:

Set

\[\alpha_{\delta}^{i} = (-1)^{m-i} \binom{m}{i} \frac{1}{S_{\delta}} \]

54
such that
\[ D_m^\delta = \sum_{i=0}^{m} \alpha_i S_{\delta}^{i-m}. \]

Now suppose also that
\[ D_m^\delta = \sum_{i=0}^{m} \beta_i S_{\delta}^{i-m}. \]

Then
\[ 0 = D_m^\delta - D_m^\delta = \sum_{i=0}^{m} (\alpha_i - \beta_i) S_{\delta}^{i-m} \]

by independence,
\[ (\alpha_i - \beta_i) = 0 \quad \text{for all } i = 0, 1, \ldots, m. \]

Lemma (L - 3.2): Let \( S_\delta \) and \( D_\delta \) be operators on \( B(\mathbb{R}) \) as defined before, and \( \{\delta_m \; ; \; m = 0, 1, \ldots, M\} \) be a set of \( M+1 \) real constants. Then:

1) There exists an unique set of \( M+1 \) real coefficients \( \{\alpha_m \; ; \; m = 1, 2, \ldots, M+1\} \) such that
\[ \sum_{m=0}^{M} \delta_m D_m^\delta = \sum_{m=1}^{M+1} \alpha_m S_m S_m^{M-1}. \]

2) Moreover,
\[ \sum_{m=1}^{M+1} \alpha_m = \delta_0. \]

3) and each of these coefficients is given in terms of \( \{\delta_m \; ; \; m = 0, 1, \ldots, M\} \)
such that

\[ D^m_\delta = \sum_{i=0}^{m} \alpha_i S^{i-m}_\delta. \]

Now suppose also that

\[ D^m_\delta = \sum_{i=0}^{m} \beta_i S^{i-m}_\delta. \]

Then

\[ 0 = D^m_\delta - D^m_\delta = \sum_{i=0}^{m} (\alpha_i - \beta_i) S^{i-m}_\delta \]

by independence,

\[ (\alpha_i - \beta_i) = 0 \quad \text{for all } i = 0, 1, \ldots, m. \]

\[ \square \]

Lemma (L - 3.2): Let \( S_\delta \) and \( D_\delta \) be operators on \( B(\mathbb{R}) \) as defined before, and \( \{6_m; m = 0, 1, \ldots, M\} \) be a set of \( M+1 \) real constants. Then:

1) There exists an unique set of \( M+1 \) real coefficients \( \{s_m; m = 1, 2, \ldots, M+1\} \) such that

\[ \sum_{m=0}^{M} 6_m D^m_\delta = \sum_{m=1}^{M+1} s_m S^{m-M-1}_\delta. \]

2) Moreover,

\[ \sum_{m=1}^{M+1} s_m = 6_0 \]

3) and each of these coefficients is given in terms of \( \{6_m; m = 0, 1, \ldots, M\} \).
as follows:

$$s_{i+M+1} = \sum_{m=m_b(i)}^{M} (-1)^{\overline{m}-1} {m \choose i+M} \frac{\delta m}{\delta m}$$

where

$$m_b(i) = \begin{cases} 
-2i & \text{if } i \leq -1 \\
0 & \text{if } i = 0 \\
2i-1 & \text{if } i \geq 1 
\end{cases}$$

for each $i = -M,-M+1,...,M$.

Proof:

1) By (L - 3.1):

$$\delta m p_m^m = \frac{\delta m}{\delta m} \sum_{i=0}^{m} (-1)^{m-i} {m \choose i} s_{i+m}^{m} =$$

$$= \frac{\delta m}{\delta m} \sum_{i=-m}^{m} (-1)^{m-i} {m \choose i+m} s_{i+m}^{m} \ldots (4)$$

Hence

$$\sum_{m=0}^{M} \delta m p_m^m = \sum_{m=0}^{M} \frac{\delta m}{\delta m} \sum_{i=-m}^{m} (-1)^{m-i} {m \choose i+m} s_{i+m}^{m} =$$

$$= \sum_{i=-M}^{M} r_i s_i^{M} \ldots (5)$$

for some set of $M+1$ coefficients $\{r_i; i = -M,...,M\}$. Moreover, this representation of $\sum_{m=0}^{M} \delta m p_m^m$ in terms of $\{s_{i}^{M};...,s_{i}^{M}\}$ must be unique by independence of $\{s_{i}^{M};...,s_{i}^{M}\}$. Then defining

$$s_i = r_{i-M-1}$$

we have proved the first part.
2) Let $\delta_m$ be written in the following form:

$$\delta_m = \sum_{i=0}^{m} \rho_m(i) s_{\delta}^{i-m}.$$  \hspace{1cm} (6)

So, by $(L - 3.1)$, we have

$$\rho_m(i) = \delta_m (-1)^{m-i} \binom{m}{i}.$$  \hspace{1cm} (7)

Since

$$\sum_{i=0}^{m} (-1)^{m-i} \binom{m}{i} = (-1)^{m} \sum_{i=0}^{m} (-1)^{i} \binom{m}{i} = \begin{cases} 1 & \text{if } m = 0 \\ 0 & \text{if } m \neq 0 \end{cases}$$

we get

$$\sum_{i=0}^{m} \rho_m(i) = \begin{cases} \delta_0 & \text{if } m = 0 \\ 0 & \text{if } m \neq 0 \end{cases}.$$  \hspace{1cm} (7)

Then, by (6) and the result of the first part, we have

$$\sum_{m=0}^{M} \delta_m \delta_m = \sum_{m=0}^{M} \sum_{i=0}^{m} \rho_m(i) s_{\delta}^{i-m} = \sum_{m=1}^{M+1} s_m s_{\delta}^{m-1}.$$  \hspace{1cm} (7)

Now, using (7), it comes:

$$\sum_{m=1}^{M+1} s_m = \sum_{m=0}^{M} \left( \sum_{i=0}^{m} \rho_m(i) \right) = \delta_0.$$  \hspace{1cm} (7)

3) Finally let us return to the equation (4) and (5). For each $m = 0, \ldots , M$ the coefficient of $s_{\delta}^{i}$ in (4) is

$$\frac{\delta_m}{\delta_m} (-1)^{m-i} \binom{m}{i-m},$$

and for each $i = -M, \ldots , M$ we have:
i) If $i = 0$, there exists $r_0$ in (5) for all $m = 0, \ldots, M$

ii) If $i = 1$, there exists $r_1$ in (5) for all $m$ such that $m \geq i$

iii) If $i = -1$, there exists $r_1$ in (5) for all $m$ such that $-m \leq i$

Then

$$r_i = \sum_{m=m_\#(i)}^{M} \frac{\delta_m (-1)^{m-i}}{\delta_m} (i, \bar{m})$$

where

$$m_\#(i) = \begin{cases} 0 & \text{if } i = 0 \\ \min \{ m : \bar{m} = i \} = \min \{ 2i-1, 2i \} = 2i-1 & \text{if } i \geq 1 \\ \min \{ m : \bar{m} = -i \} = \min \{ -2i, -2i+1 \} = -2i & \text{if } i \leq -1 \end{cases}$$

And the third part is proved since

$$r_i = r_{i-M-1}$$

Extension: Let $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ and define the following operator on $\mathcal{B}(\mathbb{R}^n)$, the linear space of all bounded real-valued functions on $\mathbb{R}^n$:

$$S_{\delta x_i}^p f(x) = f(x_1, \ldots, x_i + p \delta x_i, \ldots, x_n)$$

for any $f \in \mathcal{B}(\mathbb{R}^n)$ and $p \in \mathbb{R}$, where $\delta x_i > 0$ is fixed.

So $S_{\delta x_i}^p$, the "partial delta shift operator", is a natural extension of $S_{\delta x_i}^p$. Following this idea we can define "partial forward and backward difference operators".
and also a "m+1 th-order partial centered slope operator" on \( B(\mathbb{R}^n) \):

\[
\Delta_{\delta x_1} = S_{\delta x_1} - I
\]

\[
\Delta_{\delta x_1}^{-1} = I - S_{\delta x_1}^{-1}
\]

In this way, all the results obtained about "ordinary" operators \( S_{\delta}^p, \Delta_{\delta}, \Delta_{\delta}^{-1}, \text{ and } D_{\delta}^m \) on \( B(\mathbb{R}) \), have a direct extension to "partial" operators \( S_{\delta x_1}^p, \Delta_{\delta x_1}, \Delta_{\delta x_1}^{-1}, \text{ and } D_{\delta x_1}^m \) on \( B(\mathbb{R}^n) \).

3.4 - APPROXIMATING DERIVATIVES

The results of lemmas (L - 3.1) and (L - 3.2) are now applied to approximate certain class of linear PDEs as a natural extension of classical second-order finite-difference techniques. To begin with, we present a brief review on a particular type of approximation for ordinary derivatives.

Ordinary derivatives: Let \( f(x) \) belong to \( C^M(x_a, x_b) \), the space of all bounded real-valued functions on the interval \((x_a, x_b)\) which are continuous and have continuous derivatives up to the \( M \)th-order on \((x_a, x_b)\).

The derivative of \( f(x) \) on \((x_a, x_b)\) can be defined in terms
of operators $\Delta_{\delta}$, $\Delta^{-1}_{\delta}$ and $D_{\delta}$ since

$$
\frac{d}{dx}f(x) = \lim_{\delta \to 0} \frac{f(x) - f(x-\delta)}{\delta} = \lim_{\delta \to 0} \frac{f(x+\delta) - f(x)}{\delta} = \lim_{\delta \to 0} \frac{1}{\delta} \Delta^{-1}_{\delta} f(x) = \lim_{\delta \to 0} \frac{1}{\delta} \Delta f(x) = \lim_{\delta \to 0} D_{\delta} f(x).
$$

In the same way we can show by induction (e.g., see [21]-[23]) that higher order derivatives may also be written in terms of operators $\Delta_{\delta}^m$, $\Delta^{-m}_{\delta}$ or $D_{\delta}^m$:

$$
\frac{d^m}{dx^m}f(x) = \lim_{\delta \to 0} \frac{1}{\delta^m} \Delta_{\delta}^m f(x) = \lim_{\delta \to 0} \frac{1}{\delta^m} \Delta^{-m}_{\delta} f(x) = \lim_{\delta \to 0} \frac{1}{\delta^m} \left[ (\Delta_{\delta}^{-1})^{m-1} \Delta_{\delta}^{-m-2} \ldots \Delta_{\delta}^{-1} \Delta_{\delta} \right] f(x) = \lim_{\delta \to 0} D_{\delta}^m f(x).
$$

So forward, backward and centered operators ($\Delta_{\delta}^m$, $\Delta^{-m}_{\delta}$ and $D_{\delta}^m$) can be used to approximate derivatives at $x_o \in (x_n, x_b)$, for a sufficiently small $\delta$, as follows:

$$
\frac{d^m}{dx^m} f(x) \bigg|_{x=x_o} = \begin{cases}
\frac{1}{\delta^m} \Delta_{\delta}^{-m} f(x_o) : \text{(backward operators)} \\
\frac{1}{\delta^m} \Delta_{\delta}^m f(x_o) : \text{(centered operators)} \\
\frac{1}{\delta^m} D_{\delta}^m f(x_o) : \text{(forward operators)}
\end{cases}
$$

\footnote{For details concerning with this sort of (first-order) approximation see, for example, [21], [22], [24] and [25].}
Partial derivatives: A direct extension of the preceding approximation procedure can be done for partial derivatives as summarized below:

Let \( x = (x_1, \ldots, x_n) \in \Omega \), a simply connected open set in \( \mathbb{R}^n \) and \( f(x) \in C^m(\Omega) \), the space of all bounded real-valued functions on \( \Omega \) which are continuous and have continuous partial derivatives up to the \( m \)th-order on \( \Omega \).

The operators \( \Delta^m_{\delta x_1}, \Delta^{-m}_{\delta x_1} \) or \( D^m_{\delta x_1} \) can be used to write down partial derivatives of \( f(x) \) on \( \Omega \),

\[
\frac{\partial^m}{\partial x_1^m} f(x) = \lim_{\delta \to 0} \frac{1}{\delta x_1} \Delta^m_{\delta x_1} f(x) = \lim_{\delta \to 0} \frac{1}{\delta x_1} \Delta^{-m}_{\delta x_1} f(x) = \lim_{\delta \to 0} D^m_{\delta x_1} f(x)
\]

and so, assuming a sufficiently small \( \delta \), we can approximate partial derivatives (of a single independent variable \( \delta \)) at \( x_0 \in \Omega \) by using partial forward, backward and centered operators \( (\Delta^m_{\delta x_1}, \Delta^{-m}_{\delta x_1} \text{ and } D^m_{\delta x_1}) \) as follows:

\[
\frac{\partial^m}{\partial x_1^m} f(x) \bigg|_{x=x_0} = \begin{cases} 
\frac{1}{\delta x_1} \Delta^m_{\delta x_1} f(x_0); & \text{(backward operators)} \\
\Delta^m_{\delta x_1} f(x_0); & \text{(centered operators)} \\
\frac{1}{\delta x_1} \Delta^{-m}_{\delta x_1} f(x_0); & \text{(forward operators)} 
\end{cases}
\]

\(^8\) Here we are excluding the case containing cross-terms, such as \( \frac{\partial^2}{\partial x_1 \partial x_j} \), since they will not be of interest in our further studies.
Remark: Although we are not going to use approximation for cross-terms partial derivatives, it is interesting to notice that:

\[
\frac{\partial^m}{\partial x_1^{k_1} \ldots \partial x_n^{k_n}} f(x) \Big|_{x=x_0} = \lim_{\delta \to 0} \frac{1}{\delta^{k_1}_1 \ldots \delta^{k_n}_n} \prod_{i=1}^n \Delta^{k_i}_i f(x_0)
\]

where \( m = k_1 + \ldots + k_n \).

Finally let us consider the particular problem of approximating the linear partial differential operators introduced in (2).

Let the \( S_x \), \( S_t \), \( D_x^m \) and \( D_t^m \) be operators on \( B(R^2) \) as defined before. By lemmas (L - 3.1) and (L - 3.2) with \( (x_k, t_n) \in \Omega \subseteq R^2 \), the domain of \( u(x,t) \), and using centered operators for approximating partial derivatives we get:

\[
L^M_x u(x,t_n) \bigg|_{x=x_k} = \sum_{m=1}^M \alpha_m(x_k) D_x^m u(x_k, t_n)
\]

\[
= \sum_{m=1}^M \alpha_m(x_k) \frac{1}{\delta^m_x} \sum_{i=0}^{m-i} (-1)^{m-i} \binom{m}{i} \delta^{i-m}_x u(x_k, t_n) =
\]

\[
= \sum_{m=1}^{M+1} a_m(x_k) S_x^{m-M-1} u(x_k, t_n)
\]
In similar way

\[ L^N_t u(x_k, t) \bigg|_{t=t_n} = \sum_{m=0}^{N} \delta_m(x_k) \frac{1}{\delta^m_t} u(x_k, t_n) \]

\[ = \sum_{m=0}^{N} \delta_m(x_k) \frac{1}{\delta^m_t} \sum_{i=0}^{m} (-1)^{m-i} \binom{m}{i} S_{\delta^m_t}^{i} u(x_k, t_n) = \]

\[ = \sum_{m=1}^{N+1} c_m(x_k) S_{\delta^m_t}^{N+1-m} u(x_k, t_n) = \]

\[ = \sum_{m=1}^{N+1} c_m(x_k) u(x_k, t_n+(N-m+1)\delta_t) \]

where the coefficients \( a_m(x_k) \) and \( c_m(x_k) \) are such as introduced on lemma \((L - 3.2)\), and so (note that \( \omega_0(x) = 0 \)):

\[ \sum_{m=1}^{N+1} a_m(x_k) = 0 \]

\[ \sum_{m=1}^{N+1} c_m(x_k) = \delta_0(x_k) \]

Remarks:

1) When using this kind of centered approximation we must be sure that for a given \((x_k, t_n) \in \Omega\), \((x_k+(m'-N-1)\delta_x, t_n+(m''-N-1)\delta_t) \in \Omega\) (or at least in \( \bar{\Omega} \), the closure of \( \Omega \)) for all \( m' = 1, \ldots, N+1 \) and \( m'' = 1, \ldots, N+1 \).
2) Denoting

\[ a_k^m = a_m(x_k) \]
\[ c_k^m = c_m(x_k) \]

we get

\[ u_{k+j(n+i)} = u(x_k+j\delta x, t_n+i\delta t) \]

This simplified notation is extensively used in the next chapter.

3) The accuracy of this approximation technique increases for lower order models. The literature dealing with the classical finite-difference techniques [7]-[19] presents discussions on this subject, expressing the accuracy for models with \( M = 2 \) and \( N = 2 \).
STOCHASTIC APPROXIMATION

Stochastic Approximation is a recursive scheme which can be used for parametric estimation in a stochastic environment. Its origins are the works of Robbins and Monro [26], Kiefer and Wolfowitz [27], Blum [28] and the unified general approach given by Dvoretzky [29]. Presently there is a great deal of literature on this subject, both from theoretical and practical viewpoints. Some complete books on stochastic approximation have already been published [30], [31], and interesting surveys regarding mainly the applications are also available [32]-[42], [75].

This technique has been extensively used for identification purposes in memoryless systems and LFS (e.g., see [43]-[54]) and also, but not so extensively, for DFS identification [55]-[60]. Concerning the literature in system identification by stochastic approximation, it has become common practice to refer to Dvoretzky's work [29] and then to proceed directly to applicable algorithms. But it happens that this "bridge" linking theory and practice is not so obvious and the gap between them is not so narrow either.

For this reason we begin this third part on mathematical preliminaries by presenting the Dvoretzky theorem in abstract spaces. In an attempt to bridge that gap between theory and practice, we introduce some applicable stochastic approximation algorithms in Hilbert spaces. Particular cases involving algorithms for operators on finite-dimensional spaces are also included, since they will be required in the next chapter.
Wolfowitz [61] and Derman and Sacks [62] presented new proofs for the Dvoretzky theorem. In both cases, the proofs are given for real-valued random variables and in [62] they also presented an extension for finite-dimensional random vectors. Schmetterer [63] considered stochastic approximation algorithms, in particular the unified Dvoretzky's approach, in Hilbert spaces. The previous works were generalized by Venter [64], who proposed a wider class of algorithms in Hilbert spaces.

In his original paper, Dvoretzky [29] formulated an infinite-dimensional version for stochastic approximation algorithms in normed linear spaces, whose proof is a natural extension of the scalar (real) case. This theorem is formulated below in a simplified version, and the reader interested in its proof is referred to [29].

**Theorem (T - 3.1) (Dvoretzky):** Assume $x_0(\|x_0\| < \infty)$ a fixed point in $B$, a Banach space. Let $\{z(n); \ n = 0,1,2,\ldots\}$ be a $B$-valued random sequence, such that

$$\sum_{n=0}^{\infty} E\{\|z(n)\|^2\} < \infty \quad (8)$$

where $\| \|$ stands for the norm in $B$. Let $x(0)$ be a $B$-valued second-order random variable, and consider the following algorithm (a discrete-time dynamical system) in $B$:

$$x(n+1) = T_n x(n) + z(n) \quad (9)$$

where $\{T_n; B \to B; \ n = 0,1,2,\ldots\}$ is a family of bounded operators on $B$. 
If:

$$||T_n x - x_0|| \leq F_n ||x - x_0||$$

for any $x \in B$, and

$$E\{||T_n x(n) - x_0 + z(n)||^2\} \leq E\{||T_n x(n) - x_0||^2\} + E\{||z(n)||^2\}$$

(11)

for all $n = 0, 1, 2, \ldots$, where $\{F_n : n = 0, 1, 2, \ldots\}$ is a real sequence such that:

$$F_n > 0$$

(12)

for all $n$, and

$$\prod_{n=0}^{\infty} F_n = 0$$

(13)

Then $\{x(n)\}$ converges to $x_0$ in quadratic mean (q.m.) and with probability one (w.p.1): 9

$$\lim_{n \to \infty} E\{||x(n) - x_0||^2\} = 0$$

and

$$P\{\lim_{n \to \infty} x(n) = x_0\} = 1$$

9 The concepts of convergence "with probability one" (w.p.1), "almost certainly" (a.c.), "almost sure (or surely)" (a.s.), and "almost everywhere" (a.e.) are equivalent. For theoretical consideration see, for example, [1], [65]-[69].
Special case (3.3.1): Let B = H, a Hilbert space, and let \( \langle ; \rangle \) denote the inner product in H. Anyone of the (sufficient) conditions stated below can be used to substitute the condition (11) in the theorem:

\[
E\{z(n) \big| x(0), z(0), \ldots, z(n-1)\} = 0 \quad \text{w.p.} \ 1 \quad (11')
\]

for all \( n \), or

\[
\sum_{n=0}^{\infty} E \left\{ |\langle T_n x(n) - x_0, z(n) \rangle| \right\} < \infty . \quad (11'')
\]

Proof:

\[
E\{||T_n x(n) - x_0 + z(n)||^2\} =
\]

\[
= E\{||T_n x(n) - x_0||^2\} + E\{||z(n)||^2\} + \]

\[
+ E\{\langle T_n x(n) - x_0, z(n) \rangle\} + E\{\langle z(n) ; T_n x(n) - x_0 \rangle\} \leq
\]

\[
= E\{||T_n x(n) - x_0||^2\} + E\{||z(n)||^2\} + 2E\{|\langle T_n x(n) - x_0, z(n) \rangle|\}. \quad (11')
\]

\[
E\{\langle T_n x(n) - x_0, z(n) \rangle\} = E\{E\{\langle T_n x(n) - x_0, z(n) \rangle \big| x(n)\}\} =
\]

\[
= E\{\langle T_n x(n) - x_0, E\{z(n) \big| x(n)\}\rangle\} =
\]

\[
= E\{\langle T_n x(n) - x_0, E\{z(n) \big| x(0), z(0), \ldots, z(n-1)\}\rangle\} = 0 .
\]

In the same way:

\[
E\{\langle z(n) ; T_n x(n) - x_0 \rangle\} = 0 .
\]
Special case (3 - 3.1): Let $B = H$, a Hilbert space, and let $\langle ; \rangle$ denote the inner product in $H$. Anyone of the (sufficient) conditions stated below can be used to substitute the condition (11) in the theorem:

$$E\{z(n) \mid x(0), z(0), \ldots, z(n-1)\} = 0 \quad \text{w.p.1} \quad (11')$$

for all $n$, or

$$\sum_{n=0}^{\infty} E\{|\langle T_n x(n) - x_0 \mid z(n)\rangle\} < \infty \quad (11'')$$

Proof:

$$E\{|T_n x(n) - x_0 + z(n)|^2\} =$$

$$= E\{|T_n x(n) - x_0|^2\} + E\{|z(n)|^2\} +$$

$$+ E\{|\langle T_n x(n) - x_0 \mid z(n)\rangle\} + E\{|\langle z(n) \mid T_n x(n) - x_0\rangle\} \leq$$

$$\leq E\{|T_n x(n) - x_0|^2\} + E\{|z(n)|^2\} + 2 E\{|\langle T_n x(n) - x_0 \mid z(n)\rangle\}.$$  

$(11')$:  

$$E\{|\langle T_n x(n) - x_0 \mid z(n)\rangle\} = E\{E\{|\langle T_n x(n) - x_0 \mid z(n)\rangle \mid x(n)\}\} =$$

$$= E\{|\langle T_n x(n) - x_0 \mid z(n)\rangle\} =$$

$$= E\{E\{|\langle z(n) \mid x(n)\rangle\}\} =$$

$$= E\{E\{|\langle z(n) \mid z(0), z(0), \ldots, z(n-1)\rangle\}\} = 0.$$  

In the same way:

$$E\{|\langle z(n) \mid T_n x(n) - x_0\rangle\} = 0.$$
(11'): Set
\[ \sigma_n^2 = E\{\|z(n)\|^2\} + 2 E\{\|T_n x(n) - x_0 ; z(n)\|\} \]

So
\[ E\{\|T_n x(n) - x_0 + z(n)\|^2\} = E\{\|T_n x(n) - x_0\|^2\} + \delta_n^2 \]

where
\[ \sum_{n=0}^{\infty} \delta_n^2 < \infty \]

and that is sufficient to insure the theorem (for discussions, see [29] and [64]). \(\square\)

**Special case (S - 3.2):** Now set \(B = BL(R^k)\), the class of all bounded linear operators from \(R^k\) into itself. Let \(A^*\) and \(tr[A]\) denote the adjoint and the trace of an element \(A\) in \(BL(R^k)\), respectively. As in (S - 3.1), the conditions below can be used to substitute (11) in (T - 3.1):
\[ E\{Z(n)\big|X(0),Z(0),\ldots,Z(n-1)\} = 0 \quad w.p.1 \]

for all \(n\), or
\[ \sum_{m=0}^{\infty} E\{ tr[(T_n X(n) - X_0)^* Z(n)]\} < \infty. \]

---

10 If \(H\) is a real Hilbert space, the absolute valued appearing in (11') may be omitted.
Proof:

BL(R^k), with addition and scalar multiplication defined as usual, is a Hilbert space with the inner product defined as

\[ \langle A, B \rangle = \text{tr}[A^* B] \]

for all A, B in BL(R^k) [70]. So, by invariance of the trace (which is real) [71] and by equivalence of norms [72] in BL(R^k), the proof follows as in (3.1). □

3.6 - A PROPOSED STOCHASTIC APPROXIMATION ALGORITHM IN HILBERT SPACE

In this section we present two corollaries for the preceding theorem. A similar version (in R^1) can be found in [37], where the proof stating the connection with the Dvoretzky theorem is somewhat obscure.

First we need to prove the following lemma:

**Lemma (L - 3.3):** Let \( \{ \xi_n : n = 0, 1, 2, \ldots \} \) be a real positive sequence such that

\[ \frac{\xi_n}{\xi_{n+1}} \leq 1 + \xi_n + \xi_n \]

for all n, where:

\[ \xi_n \geq 0 ; \quad \sum_{n=0}^{\infty} \xi_n < \infty . \]
Define

\[ \phi(n,i) = \prod_{j=1}^{n-1} (1 - \zeta_j), \quad 1 < n \]
\[ \phi(n,n) = 1 \]

Then:

\[ \zeta_1 \phi(n+1,i+1) \leq k \zeta \]

for some finite positive constant \( k \).

Proof:

\[ \zeta_1 \phi(n+1,i+1) = \zeta_1 \prod_{j=i+1}^{n} (1 - \zeta_j) = \zeta_1 \prod_{j=i+1}^{n} \zeta_j \left( \frac{1}{\zeta_j} - 1 \right) = \]
\[ = \prod_{j=i}^{n-1} \zeta_j \left( \frac{1}{\zeta_j+1} - 1 \right) \zeta_n = \zeta_n \prod_{j=1}^{n-1} \left( \frac{\zeta_j}{\zeta_j+1} - \zeta_j \right) \leq \]
\[ \leq \zeta_n \prod_{j=1}^{n-1} (1 + \zeta_j) \leq \zeta_n \lim_{n \to \infty} \prod_{j=1}^{n-1} (1 + \zeta_j) = \]
\[ = \zeta_n k \quad 0 < k < \infty \]

since [73]:

\[ \zeta_n \to 0 \quad \text{and} \quad \sum_{n=0}^{\infty} \zeta_n < \infty \implies 0 < \prod_{n=0}^{\infty} (1 + \zeta_n) = k < \infty. \quad \square \]

**Corollary (C-3.1):** Let \( B = H \) be a Hilbert space, \( x_0 \) a fixed point in \( H \) \((||x_0|| < \infty)\), and consider the following algorithm in \( H \):

\[ x(n+1) = (1 - \zeta_n) x(n) + \zeta_n y(n) \quad (14) \]
where \( \{ c_n ; n = 0,1,2,\ldots \} \) is a real sequence, \( \{ y(n) ; n = 0,1,2,\ldots \} \) is a \( \mathbb{H} \)-valued random sequence, and \( x(0) \) is a second-order \( \mathbb{H} \)-valued random variable independent of \( \{ y(n) \} \).

If:

i) \( \zeta_n \in (0,1) \)

ii) \( \sum_{n=0}^{\infty} \zeta_n = \infty \)

iii) \( \sum_{n=0}^{\infty} \zeta_n^2 < \infty \)

iv) \( \mathbb{E} \{ ||y(n)||^2 \} < \infty \)

and

v-a) \( \{ \zeta_n \} \) as in (L - 3.3) \(^{11}\)

v-b) \( \mathbb{E} \{ y(n) \} = x_0 \); for all \( n \)

v-c) \( \sum_{n=0}^{n-1} \mathbb{E} \{ |y(i) - x_0 ; y(n) - x_0| \} < c < \infty \)

or

v') \( \mathbb{E} \{ y(n) - x_0 ; y(0), \ldots , y(n-1) \} = 0 \quad \text{w.p.1} \)

for all \( n \). \(^{12}\)

---

\(^{11}\) For example, the sequence 
\[
\zeta_n = \frac{1}{(an + b)^{1/2}}; \quad 0 < a \leq 1; \quad b > 1; \quad \frac{1}{2} < a \leq 1
\]
satisfies (i) - (iii), and (v-a) \(^{[73]}\).

\(^{12}\) Note that, from (v') we get (v-b) \(^{[65]}\), \(^{[66]}\).
Then:
\[ \lim_{n \to \infty} \mathbb{E}\{\|x(n) - x_0\|^2\} = 0 \]
and
\[ P\{\lim_{n \to \infty} x(n) = x_0\} = 1. \]

Proof:

By (14)
\[ x(n+1) = (1 - \zeta_n) [x(n) - x_0] + x_0 + \xi_n [y(n) - x_0]. \]

Set
\[ P_n = (1 - \zeta_n) \]
\[ z(n) = \xi_n [y(n) - x_0] \]
and define operators in \( H \), as follows:
\[ T_n x = P_n (x - x_0) + x_0 \]
for any \( x \) in \( H \). So we get an algorithm as in (9):

(9): \[ x(n+1) = T_n x(n) + z(n) \]
where the condition in (8), (10), (12) and (13) are satisfied:

(8): \[ \sum_{n=0}^{\infty} \mathbb{E}\{\|z(n)\|^2\} = \sum_{n=0}^{\infty} \xi_n^2 (\mathbb{E}\{\|y(n)\|^2\} - \|x_0\|^2) < \]
\[ < (\delta^2 - \|x_0\|^2) \sum_{n=0}^{\infty} \xi_n^2 < \infty. \]

(10): \[ \|T_n x - x_0\| = P_n \|x - x_0\|, \quad \text{for any } x \text{ in } H. \]

(12): \[ 0 < P_n < 1, \quad \text{since } 0 < \xi_n < 1. \]
(13): \[ \prod_{n=0}^{\infty} p_n = 0, \quad \text{since} \quad 0 < \xi_n < 1 \quad \text{and} \quad \sum_{n=0}^{\infty} \xi_n = \infty. \]

Then, in order to complete the proof, we just need to verify the condition (11) - theorem (T - 3.1) - or one of its equivalent forms, (11') or (11''), as in (S - 3.1). In the following we show that (v) \rightarrow (11') and (v) \rightarrow (11'').

(11'): \[ E\{z(n) \mid x(0), z(0), \ldots, z(n-1)\} = \xi_n E\{y(n) - x_o \mid x(0), y(0), \ldots, y(n-1)\} = \xi_n E\{y(n) - x_o \mid y(0), \ldots, y(n-1)\} = 0 \quad \text{w.p.1}, \quad \forall n \]

since \( x(0) \) is independent of \( \{y(n)\} \).

(11''): Let us rewrite the algorithm in (14):

\[ x(n+1) - x_o = (1 - \xi_n) [x(n) - x_o] + \xi_n [y(n) - x_o]. \]

So, for any fixed \( m \), we have:

\[ c_{xy}(n+1, m) = \phi(n+1, n) c_{xy}(n, m) + \xi_n c_{yy}(n, m) \quad (15) \]

where

\[ c_{xy}(n, m) = E\{\langle x(n) - x_o; y(m) - x_o \rangle \} \]
\[ c_{yy}(n, m) = E\{\langle y(n) - x_o; y(m) - x_o \rangle \} \]
\[ \phi(n, i) = \prod_{j=1}^{n-1} (1 - \xi_j) \quad i \neq n \]
\[ \phi(n, n) = 1 \]

---

13 See, for example, [74] pp. 146.
On iterating the inequality (15) we get:

\[ c_{xy}(n,m) \leq \phi(n,0) c_{xy}(0,m) + \sum_{i=0}^{n-1} \phi(n,i+1) \xi c_{yy}(i,m). \]

Since \(x(0)\) is independent of \(\{y(n)\}\) and \(E\{y(n) - x_0\} = 0\) for all \(n\),

\[ c_{xy}(0,m) = 0, \quad \text{for any } m. \]

Thus, setting \(m = n\), we have:

\[ c_{xy}(n,n) \leq \sum_{i=0}^{n-1} \phi(n,i+1) \xi c_{yy}(i,n). \]

Now note that:

\[ \sum_{i=0}^{n-1} c_{yy}(i,n) < c < \infty \quad \text{(by assumption (v-c))} \]

and

\[ \phi(n+1,i+1) \xi_k \leq k \xi_n \]

for some finite positive constant \(k\) (by assumption (v-a) and lemma (L - 3.3)).

Then:

\[ \sum_{n=0}^{\infty} E\{|T_n x(n) - x_0, z(n)|\} = \sum_{n=0}^{\infty} F_n \xi_n c_{xy}(n,n) \leq \]

\[ \sum_{n=0}^{\infty} \xi_n (1 - \xi_n) \sum_{i=0}^{n-1} \phi(n,i+1) \xi_k c_{yy}(i,n) = \]

\[ = \sum_{n=0}^{\infty} \xi_n \sum_{i=0}^{n-1} \phi(n+1,i+1) \xi_k c_{yy}(i,n). \]
\[ \sum_{n=0}^{\infty} \kappa \xi_n^2 \sum_{i=0}^{n-1} c_{yy}(i,n) \leq \kappa \sum_{n=0}^{\infty} \xi_n^2 < \infty \]

and (11") is satisfied. 

Finally we present a particular case of stochastic approximation algorithms for bounded linear operators on \( \mathbb{R}^k \).

**Corollary (C - 3.2):** The corollary (C - 3.1) remains valid if \( B = BL(\mathbb{R}^k) \), the class of all bounded linear operators from \( \mathbb{R}^k \) into itself, provided that the condition (v-c) is replaced by

\[ \sum_{i=0}^{n-1} \mathbb{E}\{\text{tr}[((Y(i) - X_0)^* (Y(n) - X_0)]\} < \infty. \]

**Proof:**
Define

\[ c_{XY}(n,m) = \mathbb{E}\{\text{tr}[(X(n) - X_0)^* (Y(m) - X_0)]\} \]

\[ c_{XY}(n,m) = \mathbb{E}\{\text{tr}[(Y(n) - X_0)^* (Y(m) - X_0)]\} \]

and the proof follows exactly as in (C - 3.1)\(^{14}\), by using the results obtained in (S - 3.2). \( \square \)

**Remarks:**

1) In [37] Fu proposed a similar version for stochastic approximation algorithms in \( \mathbb{R}^1 \), where direct extension to \( \mathbb{R}^k \) and \( BL(\mathbb{R}^k) \) can be

---

\(^{14}\) In this case the equality in (15) holds.
obtained.\textsuperscript{15} In his formulation he did not require the conditions (v-a) and (v-c) or (v'). So, if one believes in [37], a simplified version of (C - 3.1) in $\mathbb{R}^k$ and (C - 3.2) can be formulated by assuming only the conditions (i) - (iv) and (v-b). Note that, in this case,

$$z(n) = \frac{1}{n} [y(n) - x_0]$$

$$E\{y(n) - x_0\} = 0 \quad \forall n$$

$$E\{|y(n)|^2\} < \delta^2 < \infty$$

are being used to substitute the original sufficient condition

$$E\{z(n) | x(0), x(1), \ldots, x(n)\} = 0 \quad \text{w.p.1} \quad \forall n \quad (11')$$

assumed by Dvoretzky in [29] for the scalar (real) case.

2) Finally we remark, as proved by Venter [64], that conditions weaker than (11'), such as

$$\sum_{n=0}^{\infty} (E\{|E\{z(n) | x(0), x(1), \ldots, x(n-1)\}|^2\})^{\frac{1}{2}} < \infty \quad (16)$$

$$\sum_{n=0}^{\infty} |E\{z(n) | x(0), x(1), \ldots, x(n-1)\}| < \infty \quad \text{w.p.1} \quad (17)$$

are able to ensure the convergence for algorithms in Hilbert spaces, in quadratic mean and with probability one in case of (16), and with probability one in case of (17). (Note that (11') implies (16) which implies (17). For detailed discussion see [64]).

\textsuperscript{15} Such extensions to finite-dimensional spaces have already been successfully used for identification purposes in memoryless systems [45] and LFS [47], [49].
CHAPTER 4

IDENTIFICATION FOR A CLASS OF LINEAR DPS
USING STOCHASTIC APPROXIMATION

This is the central chapter of this work. It presents a new method for identifying distributed systems operating in a stochastic environment, where no restriction concerning probability distributions is imposed.

A class of linear models driven by random inputs and observed through noisy measurements is considered. These measurements are taken at a limited number of discrete points located in the spatial domain.

The method is classified as class \( \Gamma_2 \): First of all a time-space discretization is applied in order to approximate the infinite-dimensional model (described in terms of a linear PDE) by a finite-dimensional one (described in terms of a linear vector difference-equation). So, higher order finite-difference techniques are used to reduce the DPS to a discrete-time LPS (stage I, path 2-A). Thanks to the linearity in the parameters, the space-varying coefficients are placed in an explicit form, and are then identified by using recursive asymptotically unbiased stochastic approximation algorithms (stage II). The method is suitable for on-line applications and extraneous terms may be included in the original model.
4.1 - PROBLEM FORMULATION

Model description: Consider a DPS modeled by PDE and let $u(x,t) \in H(\Omega)$ denote the dependent variable, as follows:

$$L_t^N u(x,t) = L_x^M u(x,t) + \beta(x) w_x(t)$$

$$x \in X = (0, \mathcal{G}) ; \quad t > 0$$

where the linear partial differential operators

$$L_t^N u(x,t) = \sum_{m=0}^{N} \delta_m(x) \frac{\partial^m}{\partial t^m} u(x,t)$$

$$L_x^M u(x,t) = \sum_{m=1}^{M} \alpha_m(x) \frac{\partial^m}{\partial x^m} u(x,t)$$

are such as introduced in the first part of the last chapter, as well as the appropriate function space $H(\Omega)$, $\Omega = (0, \mathcal{G}) \times (0, \mathcal{G}) \in \mathbb{R}^2$.

The input disturbances $\{w_x(t); t=0\}$ are taken to be real-valued second-order stochastic processes for each $x$ in $(0, \mathcal{G})$. The real-valued space-varying parameters $\{a_1(x), \ldots, a_M(x)\}$, $\{\xi_0(x), \ldots, \xi_N(x)\}$ and $\beta(x)$ are supposed to be in $BV[0, \mathcal{G}]$, the space of all functions of bounded variation.

$\text{1} \ w_x(t)$ can be thought as an infinite-dimensional vector-valued second-order process. Actually, using a more sophisticated mathematical terminology, $\{w_x(t); \xi \in (0, \mathcal{G}) \times (0, \mathcal{G}) \in \mathbb{R}^2\}$ is a random field, rather than a stochastic process [1]. The existence and uniqueness of solutions for such stochastic DPS has been fully investigated by Curtain and Falb [2], [3].
variation on the interval \([0, \xi]\); and \(|\xi M(x)| > 0, \beta(x) \neq 0\), for all \(x \in (0, \xi)\).  

Observations: Assume that noisy observations are available over an equidistant\(^4\) partition \(P_x\) of \(X = [0, \xi]\).

\[
z(x_k,t) = h u(x_k,t) + d v(t); \quad t > 0
\]

where the observation noise \(\{v(t) ; t > 0\}\) is taken to be a real-valued second-order stochastic process, and \(h, d\) are real constants (\(h \neq 0\)).

Problem statement: To identify the set of \(M\) parameter functions \(\{\alpha_m; [0, \xi] \to \mathbb{R}; m = 1, \ldots, M\}\) appearing in the spatial-differential operator \(L_x^m\), based on noisy observations \(z(x_k,t)\). Under this formulation, the solution lies in \(BV[0, \xi]\), the infinite-dimensional parameter space. We consider here a finite-dimensional version:

\(^2\) By a partition \(P_{[a,b]}\) of the interval \([a,b]\), we mean a finite set of points \(x_k \in [a,b]\), \(k = 0, 1, \ldots, K\), such that \(a = x_0 < x_1 < \ldots < x_K = b\). A function \(f\) defined on \([a,b]\) is said to be of bounded variation if there is a constant \(f_o\) so that for any partition of \([a,b]\)

\[
\sum_{k=0}^{K-1} |f(x_{K+1}) - f(x_k)| < f_o < \infty
\]

\(^3\) Since \(L_x^M\) and \(L_x^N\) are defined for \(x \in (0, \xi)\), the values of \(\alpha_m, \xi_m\) and \(\beta\) in \(x=0\) and \(x=\xi\) have no significance for us. To ensure that these functions are in \(BV[0, \xi]\) we can define any real value for them in \(x=0\) and \(x=\xi\).

\(^4\) A partition is said to be "equidistant", if \(x_{k+1} - x_k\) is constant for all \(k = 0, 1, \ldots, K-1\).
"Identify \( \{x_m; \ m = 1, \ldots, M\} \) for points \( x_k \in P_x \). Since \( x_m \in \text{BV}[0, \ell] \), it is always possible to find an equidistant partition \( P_x \) of \([0, \ell]\) such that the finite sequence \( \{x_{m_k}; \ x_k \in P_x\} \) is a good (the goodness depending on how small we choose \( \delta_x = x_{k+1} - x_k \)) approximation of \( x_m(x) \ x \in [0, \ell] \), for all \( m = 1, \ldots, M \).

Boundary and initial conditions: A complete description of a physical DPS requires more information than is provided by the distributed model in (1). It is necessary to add some supplementary relations: initial (and/or terminal) and boundary conditions.

Let a set of initial conditions (IC) for the DPS modeled by (1) be given by:

\[
\frac{\partial^i}{\partial t^i} u(x,t) \bigg|_{t=0} = \zeta_i(x); \ x \in [0, \ell]; \ i = 0,1,\ldots,N-1 \quad (3)
\]

where the real-valued initial functions \( \zeta_i(x) \) are bounded and continuous on \([0, \ell]\).

Although the initial conditions written in (3) are not unique, they are quite representative for the great majority of physical systems modeled by (1). The same does not happen with a set of boundary conditions. Different experiments on the same system (e.g., one modeled by (1)), provide us with different types of "a priori" information which are expressed by different sets of boundary conditions.

---

We reserve the term "boundary condition" for conditions given only at the spatial boundary (in our case, at \( x=0 \) and \( x=\ell \)).
"Identify \( \{ \alpha_m ; m = 1, \ldots, M \} \) for points \( x_k \in P \). Since
\[ \alpha_m \in BV[0, \ell] \], it is always possible to find an equidistant partition
\( P_x \) of \([0, \ell]\) such that the finite sequence \( \{ \alpha_m(x_k) ; x_k \in P_x \} \) is a
good (the goodness depending on how small we choose \( \delta_x = x_{k+1} - x_k \))
approximation of \( \alpha_m(x) \in [0, \ell] \), for all \( m = 1, \ldots, M \).

Boundary and initial conditions: A complete description of
a physical DPS requires more information than is provided by the dis­
tributed model in (1). It is necessary to add some supplementary re­
lations: initial (and/or terminal) and boundary conditions.\(^5\)

Let a set of initial conditions (IC) for the DPS modeled by
(1) be given by:

\[
\frac{\partial^i}{\partial t^i} u(x,t) \bigg|_{t=0} = \varphi_i(x) ; \ x \in [0, \ell] ; \ i = 0, 1, \ldots, N-1 \quad (3)
\]

where the real-valued initial functions \( \varphi_i(x) \) are bounded and contin­
uous on \([0, \ell]\).

Although the initial conditions written in (3) are not unique,
they are quite representative for the great majority of physical systems
modeled by (1). The same does not happen with a set of boundary condi­
tions. Different experiments on the same system (e.g., one modeled by
(1)), provide us with different types of "a priori" information which
are expressed by different sets of boundary conditions.

\(^5\) We reserve the term "boundary condition" for conditions given only
at the spatial boundary (in our case, at \( x=0 \) and \( x=\ell \)).
Since our goal is the identification of parameters appearing only in the distributed model described by the dynamic equation (1), we have two possible ways to proceed: 1) We can assume a particular set of boundary conditions which is representative for many physical experiments in DFS. This approach can also be thought as if we could choose the experimental conditions (concerning with the boundary) on which the DFS would work for identification purpose. 2) The second way would be to assume a general (but not explicit) partial differential equation operating at the boundary.

We choose the first way because it will permit us to carry out a general identification procedure, without specifying the order of the operators $L_x^N$ and $L_t^N$, up to the point of computational implementation.

For simplicity assume a set of homogeneous boundary conditions (BC):

$$\frac{\partial^i}{\partial x^i} u(x,t) \bigg|_{x=0} = 0 ; \quad t \geq 0 ; \quad i = 0, 1, \ldots, N-1$$

$$\frac{\partial^i}{\partial x^i} u(x,t) \bigg|_{x=L} = 0 ; \quad t \geq 0 ; \quad i = 0, 1, \ldots, N-1$$

Remarks:

1) (Nonhomogeneous boundary conditions) The identification method that will be proposed here does not require homogeneous BC as a necessary condition for its applicability. We could have assumed nonhomogeneous boundary conditions such as
\[
\frac{\partial^i}{\partial x^i} u(x,t) \bigg|_{x=0} = h^0_i(t) ; \quad t \geq 0 ; \quad i = 0,1,\ldots,M-1

(4')
\]
\[
\frac{\partial^i}{\partial x^i} u(x,t) \bigg|_{x=\ell} = h^\ell_i(t) ; \quad t \geq 0 ; \quad i = 0,1,\ldots,M-1
\]

where $h^0_i(t)$ and $h^\ell_i(t)$ may be deterministic or random real-valued functions. Remarks will be made along this chapter in order to show that nonhomogeneous boundary conditions may also be considered.

2) (Random initial conditions) We could have assumed the initial functions $\zeta_i(x)$ as random functions rather than deterministic ones. But this assumption would not bring any further generalization to our identification method, which already assumes random inputs. Also note that, in case of homogeneous boundary conditions, we must have $\zeta_0(0) = g(\ell) = 0$.

4.2 - REDUCTION TO A FINITE-DIMENSIONAL STATE SPACE

**Space-time discretization:** The space and time-domain can be partitioned as follows:

1) Discretization of space-domain $\bar{X} = \{x: 0 \leq x \leq \ell \}$:

Define

\[ k \in \bar{S} = \{0,1,\ldots,K+M-1\} \]

where:

i) The integer $M$ is the order of the operator $L^M_x$.

ii) $K = \frac{\ell}{\delta_x} - M + 1$.

iii) The real constant $\delta_x > 0$ is such that $K$ is an integer $> M+1$. 

63
So, the function

\[ x_k = k \delta_x \]

from \( S \) into \( X \) defines a partition \( P_x \) of \( X \). The set \( S \) is called "the discrete space-domain".

2) Discretization of time-domain \( \{ t : t \geq 0 \} \):

Define

\[ n \in T = \{ 0, 1, \ldots \} \]

A function from \( T \) into \( [0, \omega) \) such as

\[ t_n = n \delta_t \]

where \( \delta_t > 0 \) is a real constant, defines a partition \( P_t \) of the interval \( [0, \omega) \). The set \( T \) is called "the discrete time-domain".

Remarks:

1) The sets

\[
\begin{align*}
S &= \{ \bar{M}, \bar{M}+1, \ldots, K+\bar{M}-1 \} \subset S \\
\bar{S} &= \{ \bar{M}, 2\bar{M}+1, \ldots, K+\bar{M}-1 \} \subset S \\
T' &= \{ \bar{N}, \bar{N}+1, \ldots \} \subset T \\
T &= \{ \bar{N}, \bar{N}+1, \ldots \} \subset T' \\
\bar{T} &= \{ N, N+1, \ldots \} \subset T
\end{align*}
\]

will be of particular interest in our further studies. The discretization procedure, as well as the location of \( S, \bar{S}, T', T \) and \( \bar{T} \), are shown in figures 4 and 5.
2) The condition $K > M + 1$ imposed on $\xi_x$ ensures that $S$ and $\mathcal{S}$ have at least $M + 1$ and $1$ points, respectively.

3) For reasons that will become clear later in this section, we assume $\xi_x$ such that

$$-\delta_x \mathcal{y}_{N-1}(x) \neq \mathcal{y}_N(x); \quad \forall x \in X$$

if $N \in Z_o^+$. This condition is always attainable since $|\mathcal{y}_N(x)| > \gamma > 0$ and $\mathcal{y}_{N-1}(x)$ is bounded $\forall x \in X$.

---

**Fig. 4: Discretization of Space-Domain.**
Approximating partial derivatives: Now we use space and time shift operators

\[ S_{x}^{t+1} u(x, t_n) = u(x, t_n + \delta t) \]

\[ S_{t}^{x} u(x, t_n) = u(x, t_{n+1}) \]

to write down approximations for partial derivatives as introduced in the second part of the last chapter. In this way, the identification problem formulated in an infinite-dimensional state space as in (1)-(4), can be reduced to the following finite-dimensional discrete version (The approximation procedure is illustrated on figure 6):
Approximating partial derivatives: Now we use space and time shift operators

\[ S_{t}^{\pm 1} u(x_k, t_n) = u(x_k, t_n \pm \delta_x) \]
\[ S_{t}^{\pm 1} u(x_k, t_n) = u(x_k, t_n \pm \delta_t) \]

\[ (x_k, t_n) P_x x P_t \]

to write down approximations for partial derivatives as introduced in the second part of the last chapter. In this way, the identification problem formulated in an infinite-dimensional state space as in (1)-(4), can be reduced to the following finite-dimensional discrete version (The approximation procedure is illustrated on figure 6):
\[ \sum_{m=1}^{N+1} a_k^m u_{k+m+(n-m)} - \sum_{m=1}^{N+1} a_k^m u_{k+m-(n-m)} + \beta_k u_k(n) \]

\((k,n) \in S \times T^*\)

**OBS:**
\[ z_k(n) = h_k u_k(n) + d v(n) \]
\((k,n) \in S \times T\)

**IC:**
\[ u_k(0), \ldots, u_k(H-1) \text{ given by } z_k(x_k) \]
\(k \in S\)

**BC:**
\[ u_k(0), \ldots, u_k(n-1) = 0 \]
\(u_k(n), \ldots, u_k(N-1) = 0\)
\(n \in T\)

**Proof:**

Let us use a simplified notation:

\[ u_k(n) = u(x_k, t_n) \]
\[ z_k(n) = z(x_k, t_n) \]
\[ v_k(n) = v(x_k, t_n) \]
\[ v(n) = v(t_n) \]
\[ \beta_k = \beta(x_k) \]

a) The discrete observation in (6) comes from (2) naturally, for all interior points \((x_k, t_n)\) such that \((n,k) \in S \times T^*\).

b) Use forward and backward operators (see last chapter, section 3.4)

\[ \frac{1}{\delta_x^m} \Delta_x^m, \frac{1}{\delta_x^m} \Delta_x^m \quad \text{and} \quad \frac{1}{\delta_t^m} \Delta_t^m \]

to approximate partial derivatives at boundary points \(x = x_0 = 0\),

87
\( x = x_{K+N-1} = \ell \) and \( t = t_0 = 0 \), respectively. So (7) and (8) are approximations for (3) and (4).

c) Use centered operators

\[
D^m_{\delta_x} \quad \text{and} \quad D^m_{\delta_t}
\]

to approximate partial derivatives at interior points \((x_k, t_n)\) such that \((k, n) \in S \times T'\). In this way, as shown in the second part of the preceding chapter, we have (5) from (1). Where the coefficients

\[
a^n_k = a_n(x_k) \quad \text{and} \quad b^n_k = b_n(x_k)
\]

are such as introduced on lemma (L - 3.2). □

Fig. 6: Continuous and Discrete Formulations.
Remarks:

1) By lemma \((L - 3.2)\) we have the following results (note that \(\alpha_0(x) = 0\)):

\[
a^i_{m+1} = \sum_{m=m_\alpha(i)}^{M} (-1)^{m-i} (i+m) \frac{\alpha_m(x_k)}{\delta_x^m}
\]

(9)

for each \(i = -\infty, -M+1, \ldots, \infty\) and \(k \in S\); where:

\[
m_\alpha(i) = \begin{cases} 
-2i & \text{if } i < -1 \\
1 & \text{if } i = 0 \\
2i - 1 & \text{if } i > 0
\end{cases}
\]

\[
c^N_{k+1} = \sum_{m=m_\gamma(j)}^{N} (-1)^{N-j} \left( \frac{\gamma^m}{\delta^m} \right) \frac{x_m(x_k)}{\delta^m}
\]

(10)

for each \(j = -\infty, -N+1, \ldots, \infty\) and \(k \in S\); where:

\[
m_\gamma(j) = \begin{cases} 
-2j & \text{if } j < -1 \\
0 & \text{if } j = 0 \\
2j - 1 & \text{if } j > 0
\end{cases}
\]

\[
\sum_{m=1}^{M+1} a^m_{k} = 0 \quad \forall k \in S
\]

(11)

\[
\sum_{m=1}^{N+1} c^m_{k} = \delta_0(x_k) \quad \forall k \in S
\]

(12)

2) In particular, from (10), the coefficient \(c^N_{k+1}\) is such that:

\[
c^N_{k+1} = \begin{cases} 
\frac{1}{\delta^N} \delta_N(x_k) & \text{if } N \in Z_0 \\
\frac{1}{\delta^N} \left[ \nu_N(x_k) + \delta_k \nu_{N-1}(x_k) \right] & \text{if } N \in Z_0
\end{cases}
\]
Remarks:

1) By Lemma (1 - 3.2) we have the following results (note that \( \alpha_0(x) = 0 \)):

\[
\begin{align*}
\alpha_k^{N+1} &= \sum_{m=n_k(i)}^M (-1)^{m-i} (i+m) \frac{\alpha_m(x_k)}{h_m} \\
\end{align*}
\]  

(9)

for each \( i = -M, -M+1, \ldots, M \) and \( k \in S \); where:

\[
\begin{align*}
m_k(i) &= \begin{cases} 
-2i & \text{if } i < 0 \\
1 & \text{if } i = 0 \\
2i-1 & \text{if } i > 0 
\end{cases}
\end{align*}
\]

for each \( j = -N, -N+1, \ldots, N \) and \( k \in S \); where:

\[
\begin{align*}
m_j(j) &= \begin{cases} 
-2j & \text{if } j < 0 \\
0 & \text{if } j = 0 \\
2j-1 & \text{if } j > 0 
\end{cases}
\end{align*}
\]

and:

\[
\begin{align*}
\sum_{m=1}^{M+1} c_k^m &= 0 \quad \forall k \in S \\
\sum_{m=1}^{N+1} c_k^m &= \gamma_0(x_k) \quad \forall k \in S
\end{align*}
\]  

(11)

(12)

2) In particular, from (10), the coefficient \( c_k^{N+1} \) is such that:

\[
\begin{align*}
c_k^{N+1} &= \begin{cases} 
\frac{1}{\delta_N} \gamma_N(x_k) & \text{if } N \in Z_0 \\
\frac{1}{\delta_t} \left[ \gamma_N(x_k) + \delta_t \gamma_{N-1}(x_k) \right] & \text{if } N \notin Z_0
\end{cases}
\end{align*}
\]
But we have already assumed that \( \chi_{N}(x) \neq 0 \) for all \( x \in X \) and \( \delta_t \chi_{N-1}(x) = -\delta_t \chi_{N-1}(x) \) for all \( x \in X \) if \( n \in \mathbb{Z}_0^+ \).

Then:

\[
c_k^{N+1} \neq 0 \quad \forall k \in S.
\]

3) Since \( \rho(x) \neq 0 \) for all \( x \in X \), we have:

\[
\rho_k \neq 0 \quad \forall k \in S.
\]

Equivalent discrete-time LP: Let \( \mathbb{R}^k \) denote the \( k \)-dimensional Euclidean space, \( \mathbb{B}L(\mathbb{R}^k) \) the normed linear space of all bounded linear operators from \( \mathbb{R}^k \) into itself, and define \( 6 \):

\[
\begin{align*}
\mathbf{u}(n) &= (u_{m+1}(n), \ldots, u_{K+M-1}(n)) ; \quad \forall n \in T \\
\mathbf{w}(n) &= (w_{m+1}(n), \ldots, w_{K+M-1}(n)) ; \quad \forall n \in T' - \text{random vectors in } \mathbb{R}^k \\
\mathbf{z}(n) &= (z_{m+1}(n), \ldots, z_{K+M-1}(n)) ; \quad \forall n \in T \\
\mathbf{d} &= d(1,1,\ldots,1) \in \mathbb{R}^k
\end{align*}
\]

\[
\mathbf{c}_m = \begin{bmatrix}
\mathbf{c}_{m+1} \\
\vdots \\
\mathbf{c}_{K+M-1}
\end{bmatrix} \in \mathbb{B}L(\mathbb{R}^k) ; \quad \forall m = 1,2,\ldots,N+1
\]

\[
\mathbf{b} = \begin{bmatrix}
\mathbf{b}_{m+1} \\
\vdots \\
\mathbf{b}_{K+M-1}
\end{bmatrix} \in \mathbb{B}L(\mathbb{R}^k)
\]

---

6 The quantities in \( \mathbb{R}^k \) and \( \mathbb{B}L(\mathbb{R}^k) \) are represented with respect to the standard basis in \( \mathbb{R}^k \) (i.e., \( \{e_i\} = (0, \ldots, 1, \ldots, 0) ; \quad i = 1, \ldots, k \) with a 1 in the \( i \)-th place and zeros elsewhere).
The approximate discrete version (5) - (8) can be written as a vector difference equation (discrete-time LFS), as follows:

From (5) and (8) (with $n \in T'$) we get

$$\sum_{m=1}^{N+1} c_m u(n+m-n-1) = A u(n) + B w(n)$$

$$c_{N+1} u(n) = -\sum_{m=1}^{N} c_m u(n+m-n-1) + A u(n) + B w(n).$$

Since $c_{N+1} \neq 0 \quad \forall k \in S$, $\exists c_{N+1}^{-1}$. Hence:

$$u(n+1) = \sum_{m=1}^{N} A_m u(n+m-n-1) + B w(n); \quad n \in T'$$

where $\{A_m \in BL(R^K); \; m = 1, \ldots, N\}$ and $B \in BL(R^K)$ are such that
Adding initial conditions supplied by (7)

\[ u(0), \ldots, u(N-1) \]

we get a full process description for all \( n \in T \), with observations given by (5)

\[ z(n) = h u(n) + d v(n) ; \quad n \in T . \]

Now define \( \{ y_m(n) ; \quad m = 1, \ldots, N \} \), sets of \( N \) random vectors in \( \mathbb{R}^N \) for each \( n \in T' \), as follows:

\[ y_m(n) = u(n+m-N-1) ; \quad n \in T' . \]

Then:

\[ y_m(0) = u(m-1) \quad \text{(initial state)} \]

\[ y_m(n+1) = y_{m+1}(n) ; \quad m \neq N \]

\[ y_N(n+1) = u(n+N) = \sum_{m=1}^{N} A_m y_m(n) + B_N v(n) \]

\[ y_N(n+1) = u(n) \quad \text{(output)} \]

In this way we get the following model of a linear discrete-time LFS, which is equivalent to the description given in (5) - (8):
\begin{align*}
\text{MODEL:} \quad \chi(n+1) &= F\chi(n) + Gw(n) ; \quad \chi(0) \text{ given} \quad (13) \\
\text{OBS:} \quad z(n) &= H\chi(n) + \delta \nu(n) ; \quad n \in T \\
\text{where:} \\
1) \quad \chi(n) &= (\chi_1(n), \ldots, \chi_N(n)) ; \quad n \in T' : \text{random vectors in } \mathbb{R}^{N \times K} \\
2) \quad F &= \begin{bmatrix}
0 & I \\
\vdots & \ddots & \ddots \\
0 & \cdots & 0 & I \\
A_1 & \cdots & A_{N+1} & \cdots & A_N
\end{bmatrix} \in BL(\mathbb{R}^{N \times K}) \\
3) \quad G &= \begin{bmatrix}
0 \\
\vdots \\
0 \\
0
\end{bmatrix} ; \quad \mathbb{R}^K \rightarrow \mathbb{R}^{N \times K} \\
4) \quad H &= h\begin{bmatrix}0 & \cdots & 0 & I & 0 & \cdots & 0 \end{bmatrix} ; \quad \mathbb{R}^{N \times K} \rightarrow \mathbb{R}^K
\end{align*}

The identity and null operators, I and 0, are in \( BL(\mathbb{R}^K) \). In case of \( H \), I is placed at \((N+1)\)th position.

\textbf{Remarks:}

1) \textbf{(Nonhomogeneous boundary conditions).} If we assume \((4')\) we get

\begin{align*}
\text{BC:} \quad u_0(n), \ldots, u_{N-1}(n) & \text{ given by } h_0^0(x_k) \\
u_{K+N}(n), \ldots, u_{K+N-1}(n) & \text{ given by } h_N^0(x_k) ; \quad n \in T
\end{align*}

instead of \((8)\), and the model in \((13)\) becomes:

\begin{align*}
\chi(n+1) &= F\chi(n) + Gw(n) + P u_T(n) ; \quad \chi(0) \text{ given}
\end{align*}
where:

1) $\mathbf{u}_T(n) = (u_0(n), \ldots, u_{M-1}(n), u_{K+M}(n), \ldots, u_{K+M-1}(n))$, in $\mathbb{R}^M$,
   represents the action of the boundary functions $h_0^0$ and $h_T^0$.
   It can be thought as an input (or disturbance) vector operating
   at the spatial boundary of the discrete version.

ii) $\mathbf{a}_T = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ u_T \\ \mathbf{B}_T \end{bmatrix}$ : $\mathbb{R}^M \rightarrow \mathbb{R}^{N \times K}$

iii) $\mathbf{B}_T = C_{N+1}^{-1} \mathbf{A}_T$ : $\mathbb{R}^M \rightarrow \mathbb{R}^K$

iv) $\mathbf{A}_T = \begin{bmatrix} a_1^1 & \ldots & a_M^1 \\ \vdots \\ a_1^M & \ldots & a_M^M \\ \vdots \\ \vdots \\ a_1^{M-1} & \ldots & a_M^{M-1} \\ a_1^{M+1} & \ldots & a_M^{M+1} \\ \vdots \\ \vdots \\ a_1^{M+2} & \ldots & a_M^{M+2} \\ \vdots \\ \vdots \\ \vdots \end{bmatrix}$ : $\mathbb{R}^M \rightarrow \mathbb{R}^K$

Note that the coefficients of $\mathbf{A}_T$ are those $\{a_{k}^{m}, k \leftrightarrow S, m = 1, \ldots, M+1\}$,
which are not coefficients of $\mathbf{A}$. This can be easily checked by
writing down the approximation of $L^H_x$ (i.e. $\sum_{m=1}^{M+1} a_k^m u_{k+m-M-1}(n)$)
$\forall k \in S$.

2) (Lower order models). It is quite obvious that the approximating model
   (5) (and so (13)) will be more accurate for cases of lower order
where:

1) \( \mathbf{u}_T(n) = (u_0(n), \ldots, u_{M-1}(n), u_{K+M}(n), \ldots, u_{K+M-1}(n)), \) in \( \mathbb{R}^N, \)

represents the action of the boundary functions \( h_i^0 \) and \( h_i^1. \)

It can be thought as an input (or disturbance) vector operating at the spatial boundary of the discrete version.

ii) \( \mathbf{C}_T = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} : \mathbb{R}^M \rightarrow \mathbb{R}^{N \times K} \)

iii) \( \mathbf{B}_T = \mathbf{C}_T^{-1} \mathbf{A}_T : \mathbb{R}^M \rightarrow \mathbb{R}^K \)

iv) \( \mathbf{A}_T = \begin{bmatrix} a_0^1 \\ \vdots \\ a_{2M-1}^1 \\ \vdots \\ a_0^{M+1} \\ a_2^{M+1} \\ \vdots \\ a_{K+M-1}^{M+2} \\ \ldots \\ a_{K+M-1}^{M+1} \end{bmatrix} : \mathbb{R}^M \rightarrow \mathbb{R}^K \)

Note that the coefficients of \( \mathbf{A}_T \) are those \( \{a_k^m, k \in S, m = 1, \ldots, M+1\} \),

which are not coefficients of \( \mathbf{A}. \) This can be easily checked by

writing down the approximation of \( L_x^M \) (i.e. \( \sum_{m=1}^{M+1} a_k^m u_{k+m-N-1}(n) \))

\( \forall k \in S. \)

2) (Lower order models). It is quite obvious that the approximating model (5) (and so (13)) will be more accurate for cases of lower order
as commented in chapter 3. For \( H \neq 2 \) and \( N \neq 2 \) (what represents the majority of cases of practical and theoretical interest), our general approximation procedure is reduced to the classical finite-difference technique (see [4] for a particular case with \( H=2 \) and \( N=1 \)).

A basic observation equation: Finally we present a relation expressing the observation dynamics, which is a fundamental step towards the identification procedure introduced in next sections.

Let \( \mathcal{Z}_n = \{n-n_H,n-n_{H+1},...,n+N\} \subset \mathcal{T} \)

and, for notational simplicity, define:

i) \( z_k(\mathcal{Z}_n) = \sum_{m=1}^{N+1} c_k^m z_k(n+m-n_H-1) \)

ii) \( v_k(\mathcal{Z}_n) = d \sum_{m=1}^{H+1} c_k^m v(n+m-n-H-1) + h \beta_k w_k(n) \)

iii) \( \mathbf{c}_k = (c_k^1,\ldots,c_k^{H+1}) \in \mathbb{R}^{H+1} \)

iv) \( z_k(n) = (z_{k-n_H}(n),\ldots,z_{k-n_H}(n)) \) : random vectors in \( \mathbb{R}^{H+1} \).

With \( (,\cdot) \) standing for the inner product in \( \mathbb{R}^{H+1} \), we claim that:

**Proposition (P - 4.1):** For all \((k,n) \in S \times \mathcal{T}\),

\[ z_k(\mathcal{Z}_n) = (z_k ; z_k(n)) + v_k(\mathcal{Z}_n) \]

95
Proof:

Since \( h \neq 0 \), we get from (6) and (7)

\[
\sum_{m=1}^{N+1} c_k^m z_k(n+m-H-1) = \sum_{m=1}^{M+1} c_k^m z_k(n+m-H-1)(n) + \\
+ d \left[ \sum_{m=1}^{N+1} c_k^m v(n+m-H-1) - v(n) \sum_{m=1}^{M+1} c_k^m \right] + h \beta_k w_k(n)
\]

for all \((k,n)\) such that \(k-\tilde{H}, \ldots, k+H \in S\) and \(n-\tilde{N}, \ldots, n+\tilde{N} \in T\), which means: \((k,n) \in \bar{S} \times \bar{T}\). But

\[
\sum_{m=1}^{N+1} c_k^m = 0; \quad \forall k \in S.
\]

Hence we have \( v_k(Z_n) \), independent of \( c_k \), as defined before. \(\square\)

4.3 - PARAMETERS IN EXPLICIT FORM

Our goal in this section is to deduce from (F - 4.1) a relation between the parameter vector

\[
\alpha_k = (a_{k}^{1}, \ldots, a_{k}^{N+1}) \in \mathbb{R}^{N+1}
\]

and the observations \( z_k(n) \), which is suitable for applying recursive identification algorithms. First, we introduce some notation:

Notation:

1) A pair of single bars, | |, stands for the absolute value (modulus) of a scalar quantity.
2) A pair of double bars, $\| \cdot \|$, denotes the standard Euclidian norm in $\mathbb{R}^k$, as well as the induced uniform norm of an operator (or its matrix) in $\mathcal{B}(\mathbb{R}^k)$. That is:

$$
\|A\| = \sup_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \max_{\|z\| = 1} \|Az\|
$$

for $x \in \mathbb{R}^k$, $A \in \mathcal{B}(\mathbb{R}^k)$ and $\|z\|^2 = \langle z ; z \rangle$.

3) A star, $\ast$, denotes the transpose of a matrix in the usual way; and the transpose of a vector when it is (notationally) written as a column vector. In this way we have:

$$
\bar{x} \ast y : \mathbb{R}^k \rightarrow \mathbb{R}^k
$$

$$
\bar{x} \ast y = \langle x ; y \rangle
$$

for any $x$ and $y$ in $\mathbb{R}^k$.

4) The symbols, $E\{ \}$ and Cov$\{ ; \}$, stand for the expectation and covariance operators, respectively.

5) The Kronecker delta function $\delta(i)$ is defined by

$$
\delta(i) = \begin{cases} 
1 & \text{if } i = 0 \\
0 & \text{otherwise}
\end{cases}
$$

Now, make the following assumptions on the linear discrete-time system described in (13) and (14):

**Assumption (A - 4.1): (Stability)** The space and time sampling rates, $\delta_x$ and $\delta_t$, are chosen such that the system in (13) is stable in the following sense: There exist constants $K_o$ and $\rho$ ($0 < \rho < 1$) such that

$$
\|F\|^n < K_o \rho^n , \quad \forall n \in \mathbb{T}
$$
Assumption (A - 4.2): (Stationarity) \( \{v(n) ; n \in T\} \) and \( \{w(n) ; n \in T\} \) are real and \( \mathbb{R}^K \)-valued second-order wide-sense stationary random sequences, respectively, with the following additional conditions:

i) \( E\{w(n)\} = 0 \)

ii) \( \text{Cov}\{w(i) ; w(j)\} = E\{w(i)w^*(j)\} = C_w \delta(i-j) \)

iii) \( E\{v(n)\} = \eta_v \)

iv) \( \text{Cov}\{v(i) ; v(j)\} = E\{v(i)v^*(j)\} - \eta_v^2 = \delta_v(|i-j|) \)

v) \( \text{Cov}\{v(i) ; w(j)\} = E\{v(i)w^*(j)\} = 0 \)

vi) \( w(n) \) and \( v(n) \) have finite moments up to the 4th order.

Where \( C_w \) is a symmetric positive definite matrix in \( BL(\mathbb{R}^K) \).

Assumption (A - 4.3): (Steady state) The initial state response is assumed to have died out before identification begins. Since the input disturbance \( w(n) \) and the observation noise \( v(n) \) are wide-sense stationary and the system under consideration is time-invariant, this assumption implies that the state \( x(n) \) and observation \( z(n) \) processes will also be wide-sense stationary during the identification procedure (see, for example, [5]-[7]).

Assumption (A - 4.4): (Finite transient time) Let \( N_T > N_N > 0 \) be a sufficient large integer such that the steady state assumption at the time \( n + N_T \) is valid for all \( n \in T \).

---

7 The term "random sequence" will be used to denote a "discrete-time stochastic process".
Finally, for each \( k \in S \), define:

i) \( \xi_k = -d^2 \sum_{m=1}^{N+1} c_k^m [\delta_v(m-N-1) + \eta_v^2] \)

ii) \( \xi_k = \xi_k(1, \ldots, 1) \in \mathbb{R}^{N+1} \)

iii) \( q_k = E\{z(n+N_t) \xi_k(z_{n+N_t})\} + \xi_k \in \mathbb{R}^{N+1} \)

iv) \( Q_k^{-1} = E\{z_k(n+N_t) \xi_k^*(n+N_t)\} \in \text{BL}(\mathbb{R}^{N+1}) \)

**Proposition (P - 4.2):** For all \( k \in S \) and \( n \in \mathbb{N} \),

\[
E\{z_k(n+N_t) \xi_k(z_{n+N_t})\} = -\xi_k
\]

**Proposition (P - 4.3):** There exists the symmetric positive definite matrix \( Q_k = [Q_k^{-1}]^{-1} \in \text{BL}(\mathbb{R}^{N+1}) \), for all \( k \in S \).

**Proposition (P - 4.4):** Both \( q_k \) and \( Q_k \) do not depend on \( n \).

---

By (12) we get:

\[
\xi_k = -d^2 \left[ \eta_v^2 \gamma_0(x_k) + \sum_{m=1}^{N+1} c_k^m \delta_v(m-N-1) \right]
\]

If \( \delta_v(i-j) = \delta_v^2 \delta(i-j) \), \( \delta_v^2 > 0 \), we have from (10):

\[
\xi_k = -d^2 \left[ \eta_v^2 \gamma_0(x_k) + \delta_v^2 \sum_{m=0}^{N} (-1)^\bar{m} \left( \begin{array}{c} m \\ \bar{m} \end{array} \right) \gamma_\bar{m}(x_k) \right]
\]

Note also that in [4] \( \xi_k \) is defined slightly differently. There, it is divided by \( Q_k^{-1} = \delta_t^{-1} \) for the particular case of \( N=1 \), \( \gamma_0 = -\gamma_1 \) and \( \gamma_1 = 1 \) (see section 4.6).
**Proof:**

(P - 4.2): Using finite induction we get the solution of the equation (13):

\[
\chi(n+N_t+1) = F^{n+1} \chi(n+N_t) + \sum_{m=0}^{n} F^{n-m} C \chi(m+N_t); \quad n \in \mathbb{Z}.
\]

By assumption (A - 4.3) and (A - 4.4), the first term in the right hand side is supposed to have died out for any time \(n < N_t\). Thus,

\[
\chi(n+N_t) = \sum_{m=0}^{n+N_t+1} F^{n+N_t-m-N_t-1} C \chi(m+N_t)
\]

and from (14) we get

\[
\chi(n+N_t) = R \chi(n+N_t) + d \psi(n+N_t).
\]

Now, with

\[
\psi(n+N_t) = d \sum_{m=1}^{N+1} c_k^m \psi(n+N_t+m-N_t-1) + h / k \psi(n+N_t),
\]

assumption (A - 4.2), and using the linearity of the expectation operator [1], we have:

\[
E\{\chi(n+N_t) \psi(n+N_t)\} = d d \sum_{m=1}^{N+1} c_k^m \left[ \psi(n+N_t) + \eta^2 \right] + \psi \psi(n+N_t)
\]

\[
= - \epsilon_k(1, \ldots, 1) \in \mathbb{R}^K.
\]

---

9 Actually, the assumptions (A - 4.3) and (A - 4.4) can be thought as a consequence of (A - 4.1) when one is considering the asymptotic behaviour. That is:

\[
0 \leq ||F^n|| \leq ||F||^n \leq K_o \rho^n; \quad 0 < \rho < 1
\]

\[
\lim_{n \to \infty} ||F^n|| \leq K_o \lim_{n \to \infty} \rho^n = 0 \Rightarrow F^n \to 0 \text{ as } n \to \infty
\]

100
Finally, define:

\[ J_k = \begin{bmatrix} \emptyset & I & \emptyset \end{bmatrix} : \mathbb{R}^K \to \mathbb{R}^{K+1} \]

where there are \( k-2 \) zero columns before \( I \in \text{BL}(\mathbb{R}^{K+1}) \) and, of course, \( K+\tilde{n}-\tilde{k}-1 \) zero columns after \( I \).

So:

\[ z_k(n) = J_k z(n) ; \quad \forall k \in \mathbb{S} . \]

Hence:

\[ E \{ z_k(n) \} = J_k E \{ z(n) \} \]

and (P - 4.2) is proved.

(P - 4.3): It is easy to show that:

\[ z(n+N) = H F^N \chi(n) + H F^{N-1} G \mu(n) + d v(n+N) ; \quad n \in T' \]

and

\[ H F^{N-1} = h \begin{bmatrix} 0 & \ldots & 0 & 1 \end{bmatrix} : \mathbb{R}^{N \times K} \to \mathbb{R}^K \]

\[ H F^N = h \begin{bmatrix} A_1 & \ldots & A_N \end{bmatrix} : \mathbb{R}^{N \times K} \to \mathbb{R}^K \]

Now define:

\[ \Phi = H F^N : \mathbb{R}^{N \times K} \to \mathbb{R}^K \]

\[ \Gamma = H F^{N-1} G = h B_N \in \text{BL}(\mathbb{R}^K) \]

\[ C_y = \text{Cov}\{ \chi(n+N) ; \chi(n+N') \} \in \text{BL}(\mathbb{R}^{N \times K}) \]

\[ C_z = \text{Cov}\{ z(n+N) ; z(n+N') \} \in \text{BL}(\mathbb{R}^K) \]

---

10 This result comes from equations (13) and (14) or, equivalently, from the solution for \( \chi(n) \), \( n \geq \tilde{n} \), as introduced at the beginning of this proof. Note that \( H F^m G = 0, \forall m = 0, \ldots, \tilde{n}-2 \).
Then
\[ z(n+N_t+n) = \Phi_{zv}(n+N_t) + \Gamma_{yv}(n+N_t) + \Delta_{yv}(n+N_t+N) ; \quad n \in T. \]

From (A - 4.3) and (A - 4.4), \( c_y \) and \( c_z \) do not depend on \( n \). So, by (A - 4.2), we get: \(^{11}\)

\[ c_z = \Phi c_y \Phi^* + \Gamma c_v \Gamma^* + \delta_{yv}(0) \Delta \Delta^* \]

and \( Q_k^{-1} \in BL(R^{n+1}) \) is given in terms of \( c_z \in BL(R^K) \) as follows:

\[ z_k(n) = J_k \hat{z}(n) \rightarrow c_k^{-1} = J_k c_z J_k^*. \]

Since \( c_w \neq 0 \) (positive definite) \( \in BL(R^K) \), and all eigenvalues of

\[ \Gamma = h \Delta = -h c_k^{-1} \Delta \in BL(R^K) \]

are different from zero \(^{12}\),

\[ \Gamma c_w \Gamma^* > 0 \rightarrow c_z > 0. \]

But, if \( c_z > 0 \), all principal minors of \( c_z \) are positive. In particular, all successive principal minors of the symmetric matrix

\[ Q_k^{-1} = J_k c_z J_k^* \in BL(R^{n+1}) \]

\(^{11}\) Note that:

\[ \text{Cov}(\Phi \gamma(n), \gamma(n)) = 0; \quad \forall n \in T. \]

\[ E\{\gamma(n+N_t)\} = 0 \rightarrow E\{\gamma(n+N_t)\} = 0 \rightarrow E\{z(n+N_t)\} = \gamma\Delta; \quad \forall n \in T. \]

\(^{12}\) Recall: \( 0 \leq ||A_k|| < \infty \) and \( 0 \leq ||c_k^{n+1}|| < \infty; \quad \forall k \in S. \)
are positive. Hence $Q^{-1}_k > 0$ (Sylvester's criteria. See, for example, [8], pp. 306). So, there exists the symmetric

$$Q_k = [Q^{-1}_k]^{-1} > 0 \in \text{BL}(R^{n+1}), \quad \forall k \in s.$$ 

(P - 4.4): This result comes directly from (A - 4.2) - (A - 4.4): wide-sense stationary random sequences (for details see, for example, [5]-[7]).

**Remark:** (Random initial state) Assume that all eigenvalues of the system matrix $F$ in $\text{BL}(R^{n \times K})$ are different from zero (or equivalently, $\det(F) \neq 0$), and define

$$\text{Cov}\{\hat{\gamma}(n) ; \hat{\gamma}(n)\} = \begin{cases} 
C_y(n) & \text{if } n \leq N_t \\
C_y & \text{if } n > N_t
\end{cases}$$

It is well known (e.g., see [5]-[7]) that

$$C_y(n+1) = FC_y(n)F^* + GC_yG^*$$

for all $n < T$. So, if the initial state

$$\hat{\gamma}(N) = (u(0), \ldots, u(N-1))$$

is a random vector in $R^{n \times K}$ such that

$$C_y(N) > 0$$

Note that, in case of $(M+K) < Z_0$,

$$Q^{-1}_{k_o} = J_{k_o} C_z J_{k_o}^* \in \text{BL}(R^{M+1})$$

is the $(M+1)$th "inner" of $C_z \in \text{BL}(R^K)$ [9], where $k_o = (K+M-1)/2$. 

103
in $BL(\mathbb{R}^{n \times k})$, we get
\[ C_y(n) > 0 \quad \forall n \in T, \]
in $BL(\mathbb{R}^{n \times k})$, and so:
\[ \Phi_y C_y \Phi^* > 0 \quad \Rightarrow \quad C_z > 0 \]
in $BL(\mathbb{R}^k)$. Hence, in this case, the conditions
\[ f(x) \neq 0 \quad \forall x \in X \]
and
\[ C_w > 0 \quad \text{in} \quad BL(\mathbb{R}^k) \]
can be omitted, since they are imposed only to ensure that $C_z > 0$ when the initial state is assumed to be deterministic.

Lemma (L - 4.1): (Explicit Parameter) Let $\varphi_k$ and $\varphi_\theta$ be as defined before. If the assumptions (A - 4.1) through (A - 4.4) are satisfied, then the parameter vector
\[ \alpha_k = (a_k^1, \ldots, a_k^{n+1}) \in \mathbb{R}^{n+1} \]
introduced in (P - 4.1) can be placed in an explicit form, as follows:
\[ \beta_k = \psi_k \varphi_k \]
for all $k$ in $\theta$.

Proof:
By (P - 4.1) through (P - 4.4), we have:
\[ Z_k(\varphi_{n+N_k}) = Z_k^*(n+N_k) \varphi_k + \psi_k(\varphi_{n+N_k}) \quad \forall k \in \theta \]
\begin{align*}
E\left\{ z_k(n+H_t) z_k'\left(\frac{n+H_t}{n}\right) \right\} &= E\left\{ z_k(n+H_t) z_k^*\left(\frac{n+H_t}{n}\right) \right\} E_k + E\left\{ z_k(n+H_t) \right\} E_k \left(\frac{n+H_t}{n}\right) \\
&= a_k^{-1} \frac{a_k}{a_k} - \xi_k.
\end{align*}

Then:
\[ a_k = q_k \left[ E\left\{ z_k(n+H_t) z_k\left(\frac{n+H_t}{n}\right) \right\} + \xi_k \right] = q_k a_k ; \quad \forall k \in S. \quad \Box \]

4.4 - PARAMETER IDENTIFICATION

General procedure: Now we consider the identification problem for DPS as formulated in section 4.4, by using the reduction to a finite-dimensional state space introduced in section 4.2. In this way, our identification procedure comprises two basic steps:

1st) By using noisy observations \( \{ z_k(n+H_t) ; n \in T \} \) available in each \( k \in S \), determine the coefficients \( \{ a_k^m ; m = 1, \ldots, M+1 \} \) appearing in the discrete version of the spatial-differential operator \( L_k^n \) as in (5). Since the observation process in \( R^{M+1} \)

\[ z_k(n+H_t) = ( z_k, z_k^*(n+H_t), \ldots, z_k^{M+1}(n+H_t) ) \]

is defined only for \( k \in S \) and

\[ a_k = ( a_k^1, \ldots, a_k^{M+1} ) \in R^{M+1} \]

is related to \( z_k(n+H_t) \) as in (4.1), this first step can be briefly stated as follows: "given \( \{ z_k(n+H_t) ; n \in T \} \) \( \forall k \in S \), determine \( a_k \) \( \forall k \in S \)."
In (L - 4.1) we get \( \rho_k \) in terms of \( Q_k \) and \( g_k \). But it is quite obvious that we cannot use the values of \( Q_k \) and \( g_k \) in order to perform the identification, because both of them depend on the knowledge of the matrix \( F \) (and so, they depend on the parameters \( c_k^m \)). Therefore, by using the stochastic approximation theory (chapter 3 - Part III) together with the explicit parameter lemma (L - 4.1), we may be able to present an on-line identification algorithm for \( g_k \), without computing the values of \( Q_k \) and \( g_k \). This first step is the central theme in the remainder of this section.

2nd) On the other hand, we also have the problem of recovering the parameters \( \{ \alpha_m(x_k^m); m = 1, \ldots, M \} \) from \( g_k \), for each \( k \in S \). This is a much easier problem than that concerning the first step, and it will be considered in the next section.

**Stochastic approximation algorithms:** First, by using the results developed in the third part of the preceding chapter, we present an auxiliary lemma for recursive estimation of \( g_k \) and \( Q_k^{-1} \).

**Lemma (L - 4.2):** Let \( \mathbf{z}_k(n+N_t) \), \( \mathbf{z}_k(n+N_t^*) \), \( \mathbf{F}_k \), \( \mathbf{g}_k \) and \( Q_k^{-1} \) be as defined before and consider the following algorithms in \( \mathbb{R}^{M+1} \) and \( \mathbb{B} \mathbb{L}(\mathbb{R}^{M+1}) \), respectively, for each \( k \in S \):

\[
\begin{align*}
Q_k(n+1) &= [1 - \lambda(n)] g_k(n) + \lambda(n) [\mathbf{z}_k(n+N_t) \mathbf{z}_k(n+N_t^*) + \mathbf{F}_k] ; \quad n \in \mathbb{T} \\
Q_k^{-1}(n+1) &= [1 - \mu(n)] Q_k^{-1}(n) + \mu(n) [\mathbf{z}_k(n+N_t) \mathbf{z}_k^*(n+N_t)] ; \quad n \in \mathbb{T}
\end{align*}
\]

\[14\] Note that, by (A - 4.1) and (A - 4.2), \( \| g_k \|_\infty \) and \( \| Q_k^{-1} \|_\infty \) for all \( k \in S \).
where \( \{\lambda(n); \ n \in \mathbb{T}\} \) and \( \{\mu(n); \ n \in \mathbb{T}\} \) are real sequences, \( s_k(0) \) is a second-order random vector in \( \mathbb{R}^{M+1} \) which is independent of \( \{z_{k}(n+N_t); \ n \in \mathbb{T}\} \) for each \( k \in S \), and \( Q_k^{-1}(0) \) is a second-order random matrix in \( \text{BL}(\mathbb{R}^{M+1}) \) which is independent of \( \{z_{k}(n+N_t); z^*_k(n+N_t); \ n \in \mathbb{T}\} \) for each \( k \in \mathbb{S} \).

If:

1) \( \lambda(n) \in (0,1); \quad \sum_{n=0}^{\infty} \lambda(n) = \infty; \quad \sum_{n=0}^{\infty} \lambda^2(n) < \infty \)

2) \( \mu(n) \in (0,1); \quad \sum_{n=0}^{\infty} \mu(n) = \infty; \quad \sum_{n=0}^{\infty} \mu^2(n) < \infty \)

Then:

1) \( P\left\{ \lim_{n \to \infty} s_k(n) = s_k \right\} = 1 \quad \text{and} \quad \lim_{n \to \infty} \mathbb{E}\{\|s_k(n) - s_k\|^2\} = 0 \)

2) \( P\left\{ \lim_{n \to \infty} Q_k^{-1}(n) = Q_k^{-1} \right\} = 1 \quad \text{and} \quad \lim_{n \to \infty} \mathbb{E}\{\|Q_k^{-1}(n) - Q_k^{-1}\|^2\} = 0 \)

for each \( k \in \mathbb{S} \).

Proof:

Set, for each \( k \in \mathbb{S} \),

\[
Y(n) = z_{k}(n+N_t) z_{k}(n+N_t) + \varepsilon_k; \quad x_0 = s_k
\]

\[
Y(n) = z_{k}(n+N_t) z^*_k(n+N_t); \quad x_0 = Q_k^{-1}
\]

in (C - 3.1) and (C - 3.2), and the proof follows directly by the results of section 3.6 and assumptions (A - 4.1) - (A - 4.4).

Before introducing the main identification theorem, we need to prove the following propositions:
Proposition (P-4.5): Let \( \{Q_k^{-1}(n) ; n \in \mathbb{T} \} \) be a sequence of random matrices in \( \text{BL}(\mathbb{R}^{k+1}) \) as defined in (L-4.2) for each \( k \in \mathcal{S} \).

If, in addition, \( Q_k^{-1}(0) \) is symmetric positive definite, then there exists the symmetric positive definite random matrix

\[
Q_k(n) = [Q_k^{-1}(n)]^{-1}
\]

in \( \text{BL}(\mathbb{R}^{k+1}) \) for all \( n \in \mathbb{T} \) and \( k \in \mathcal{S} \).

Proof:

From (16)

\[
Q_k^{-1}(n+1) = [-\lambda(n)] Q_k^{-1}(n) + \lambda(n) z_k(n+N_1^T) z_k^*(n+N_1^T).
\]

Since \( \lambda(n) \in (0,1) \) for all \( n \in \mathbb{T} \), \( Q_k^{-1}(0) \) is symmetric positive definite for each \( k \in \mathcal{S} \), and \( z_k(n+N_1^T) z_k^*(n+N_1^T) \) is symmetric positive semi-definite for all \( (k,n) \in \mathcal{S} \times \mathbb{T} \); \( Q_k^{-1}(n) \) is symmetric positive definite for all \( (k,n) \in \mathcal{S} \times \mathbb{T} \). The existence of \( Q_k(n) = [Q_k^{-1}(n)]^{-1} \), a symmetric positive definite random matrix in \( \text{BL}(\mathbb{R}^{k+1}) \), is thus guaranteed for all \( (k,n) \in \mathcal{S} \times \mathbb{T} \). \( \square \)

Proposition (P-4.6): Let \( A \in \text{BL}(\mathbb{R}^k), \ M \in \mathbb{R}^k \) and assume the existence of \( A^{-1} \) and \( (A+M M^*)^{-1} \).

If \( M A^{-1} M \neq -1 \), then:

\[
(A+M M^*)^{-1} = A^{-1} - (1+M^* A^{-1} M)^{-1} A^{-1} M M^* A^{-1}.
\]

\[\text{(15)}\]

Note that if \( A \succ 0 \), then:

\[
(3 A^{-1} \succ 0 \implies M A^{-1} M \preceq 0) \]

\[
(A+M M^*) \succ 0 \implies (A+M M^*)^{-1} \succ 0.
\]

108
Proof:
This is a trivial particular case of the "matrix inversion lemma" [6], or "method of modification" [10]. □

Now we can present a stochastic approximation algorithm for identifying the vector \( \xi_k \) through noisy observations \( \{z_k(n+N_t^+) ; n \in \mathbb{T} \} \).

**Theorem (T - 4.1):** Let \( I \) denote the identity matrix in \( BL(\mathbb{R}^{M+1}) \) and \( \xi_k \in \mathbb{R}^{M+1} \), for each \( k \in S \), the parameter vector as introduced on proposition (P - 4.1). Also let \( z_k(n+N_t^+) \), \( z_k(n+N_t^+) \) and \( \xi_k \) be as defined before, and consider the following algorithm in \( \mathbb{R}^{M+1} \), for each \( k \in S \):

\[
(SA-1): \quad \xi_k(n+1) = \frac{1}{1-\mu(n)} \left[ I - \mu(n) Q_k(n+1) z_k(n+N_t^+) z_k^*(n+N_t^+) \right] \xi_k(n) + \\
+ \lambda(n) Q_k(n+1) \left[ z_k(n+N_t^+) z_k(n+N_t^+) + \xi_k \right] ; \quad n \in \mathbb{T}
\]

with \( Q_k(n) \) taking values in \( BL(\mathbb{R}^{M+1}) \), for each \( k \in S \), given by

\[
(SA-2): \quad Q_k(n+1) = \left[ \frac{\mu(n) Q_k(n) z_k(n+N_t^+) z_k^*(n+N_t^+) Q_k(n)}{1-\mu(n) + \mu(n) z_k(n+N_t^+) z_k(n+N_t^+) Q_k(n)} \right] ; \quad n \in \mathbb{T}
\]

where \( \{\lambda(n) ; n \in \mathbb{T}\} \) and \( \{\mu(n) ; n \in \mathbb{T}\} \) are real sequences, \( \xi_k(0) \) is a second-order random vector in \( \mathbb{R}^{M+1} \) which is independent of \( \{z_k(n+N_t^+) ; n \in \mathbb{T}\} \) for each \( k \in S \), and \( Q_k(0) \) is a second-order random matrix in \( BL(\mathbb{R}^{M+1}) \) which is independent of \( \{z_k(n+N_t^+) z_k^*(n+N_t^+) ; n \in \mathbb{T}\} \) for each \( k \in S \).

---

Note: Since \( \{\alpha_m(x) \in BV[0,\epsilon] ; m = 1, \ldots, M\} \), \( \|\xi_k\| < \infty \) for all \( k \in S \), by (9).
If:

i) \( \lambda(n) \in (0,1) \); \( \sum_{n=0}^{\infty} \lambda(n) = \infty \); \( \sum_{n=0}^{\infty} \lambda^2(n) < \infty \)

ii) \( \mu(n) \in (0,1) \); \( \sum_{n=0}^{\infty} \mu(n) = \infty \); \( \sum_{n=0}^{\infty} \mu^2(n) < \infty \)

iii) \( Q_k(0) \) is a symmetric positive definite random matrix in BL(R^{n+1})

Then \( a_k(n) \) converges to \( a_k \) with probability one for each \( k \in S \):

\[ P\{\lim_{n \to \infty} a_k(n) = a_k \} = 1 ; \quad k \in S \] .

Proof:

For sake of simplicity we use the following notation:

\( \lambda = \lambda(n) \)
\( \mu = \mu(n) \)
\( z_k = x_k(n+1) \)
\( z = x(n+1) \)
\( q = q(n) \)
\( a = a(n) \)
\( Q = Q(n) \)
\( Q^{-1} = Q^{-1}(n) \)
\( Q_k^{-1}(n+1) = (1-\mu) Q^{-1} + \mu x z^* \) .

a) First of all let us prove the algorithm (3A-2). From (16)
So, by (P - 4.5) and (P - 4.6) with $Q = [Q^{-1}]^{-1}$, we get

$$Q_k(n+1) = [(1-\mu) Q^{-1} + \mu z z^*]^{-1} =$$

$$= \frac{Q}{1-\mu} - (1-\mu z^* \frac{Q}{1-\mu} z)^{-1} \frac{\mu}{(1-\mu)^2} Q z z^* Q =$$

$$= \frac{1}{1-\mu} \left[ Q - \frac{Q z z^* Q}{1-\mu + \mu z^* Q z} \right]$$

since $Q_k(0)$ is symmetric positive definite, and that proves (SA-2).

Moreover, note that:

$$Q_k(n+1) z = \frac{Q}{1-\mu} \left[ (1-\mu) z + [(z^* Q z) z - (z z^* Q) z] \right].$$

But

$$(z z^* Q) z = z (z^* Q z).$$

Hence

$$(z^* Q z) z - (z z^* Q) z = (z^* Q z) z - z (z^* Q z) = 0.$$

Then

$$Q_k(n+1) z = \frac{Q z}{1-\mu + \mu z^* Q z} \quad (17)$$

b) Now define:

$$r_k(n) = Q_k(n) x_k(n); \quad n \in \mathbb{N}$$

for each $k \in \mathcal{S}$, where $x_k(n)$ is given by (15). Thus from (15) and (SA-2), with

$$w = (1-\mu) + \mu z^* Q z$$

111
So, by \( P - 4.5 \) and \( P - 4.6 \) with \( Q = [Q^{-1}]^{-1} \), we get

\[
Q_k(n+1) = [(1 - \mu) Q^{-1} + \mu \overline{z} \overline{z}^*]^{-1} = \frac{Q}{1 - \mu} - (1 - \mu \overline{z}^* \frac{Q}{1 - \mu} \overline{z})^{-1} \frac{\mu}{(1 - \mu)^2} Q \overline{z} \overline{z}^* Q = \frac{1}{1 - \mu} \left[ Q - \frac{Q \overline{z} \overline{z}^* Q}{1 - \mu + \mu \overline{z}^* \overline{z} Q} \right]
\]

since \( Q_k(0) \) is symmetric positive definite, and that proves \((SA-2)\).

Moreover, note that:

\[
Q_k(n+1) \overline{z} = \frac{Q}{1 - \mu} \left[ (1 - \mu) \overline{z} + \left( \overline{z}^* Q \overline{z} \right) \overline{z} - \left( \overline{z}^* Q \overline{z} \right) \overline{z} \right] .
\]

But

\[
(\overline{z} \overline{z}^* Q) \overline{z} = \overline{z} (\overline{z}^* Q \overline{z}) .
\]

Hence

\[
(\overline{z}^* Q \overline{z}) \overline{z} - (\overline{z} \overline{z}^* Q) \overline{z} = (\overline{z}^* Q \overline{z}) \overline{z} - (\overline{z} \overline{z}^* Q) \overline{z} = 0 .
\]

Then

\[
Q_k(n+1) \overline{z} = \frac{Q \overline{z}}{1 - \mu + \mu \overline{z}^* \overline{z} Q} . \quad (17)
\]

b) Now define:

\[
\gamma_k(n) = Q_k(n) \gamma_k(n) ; \quad n \in \mathbb{F}
\]

for each \( k \in S \), where \( \gamma_k(n) \) is given by (15). Thus from (15) and \((SA-2)\), with

\[
\omega = (1 - \mu) + \mu \overline{z}^* \overline{z} Q \overline{z}
\]
we get:

\[ q_{k}^{(n+1)} = q_{k}^{(n+1)} q_{k}^{(n+1)} = q_{k}^{(n+1)} [(1 - \lambda) \omega + \lambda z \omega ] + \lambda q_{k}^{(n+1)} e_{k} \cdot \]

But

\[ q_{k}^{(n+1)} [(1 - \lambda) \omega + \lambda z \omega ] = \]

\[ = \frac{1}{1 - \mu} \left[ Q - \frac{\mu Q z z^*}{Q} \right] [(1 - \lambda) \omega + \lambda z \omega ] = \]

\[ = \frac{1}{1 - \mu} \left[ (1 - \lambda) \omega + \lambda z \omega - \frac{Q z}{Q} (\mu(1 - \lambda) z^* - \mu \lambda z \omega) \right] = \]

\[ = \frac{1}{1 - \mu} \left[ (1 - \lambda) \omega + \frac{Q z}{Q} (\lambda z \omega - \mu(1 - \lambda) z^* - \mu \lambda z \omega) \right] = \]

\[ = \frac{1}{1 - \mu} \left[ \lambda z \omega - \mu z \omega^* - \frac{1 - \lambda}{1 - \mu} z^* \omega \right] \]

Since

\[ \omega - \mu z^* z = 1 - \mu^2 \]

we have

\[ q_{k}^{(n+1)} = \frac{1 - \lambda}{1 - \mu} \omega + \frac{Q z}{1 - \mu + \mu z^* Q z} \left[ \lambda z - \frac{1 - \lambda}{1 - \mu} z^* \omega \right] + \]

\[ + \lambda q_{k}^{(n+1)} e_{k} \]

and so the algorithm (SA-1) comes through (17).

c) The assumptions about \( q_{k}(0) \) and \( q_{k}(0) \) plus (i) – (iii) are sufficient to ensure the convergence in lemma (L - 4,2). So if we define

17 Note that \( q_{k}(0) = [q_{k}(0)]^{-1} q_{k}(0) \), by definition of \( \{ q_{k}(n) ; n \in T \} \).
the events

\[ A_k = \{ \lim_{n \to \infty} Q_k^{-1}(n) = q_k^{-1} \} \]

\[ B_k = \{ \lim_{n \to \infty} g_k(n) = g_k \} \]

for each \( k \in S \), we get by (L - 4.2) that

\[ P(A_k) = P(B_k) = 1 . \]

Hence

\[ P(A_k \cup B_k) = 1 \]

\[ \therefore P(A_k \cap B_k) = P(A_k) + P(B_k) - P(A_k \cup B_k) = 1 . \]

That is

\[ P(A_k \cap B_k) = P(\lim_{n \to \infty} Q_k^{-1}(n) = c_{k,-1}^{-1} ; \lim_{n \to \infty} g_k(n) = g_k) = 1 \]

for each \( k \in \mathcal{S} \). We have already proved in (P - 4.5) the existence (and the uniqueness comes by definition of the inverse operator) of

\[ [Q_k^{-1}]^{-1} \quad \text{and} \quad [Q_k^{-1}(n)]^{-1} \]

for all \((k,n) \in \mathcal{S} \times T\). So we can define the event

\[ C_k = \{ \lim_{n \to \infty} [Q_k^{-1}(n)]^{-1} g_k(n) = [c_{k,-1}^{-1} g_k] \} \]

for each \( k \in \mathcal{S} \). Moreover,

\[ A_k \cap B_k \subseteq C_k . \]

That is, the existence and uniqueness of \([c_{k,-1}^{-1}]^{-1}\) and \([Q_k^{-1}(n)]^{-1}\)
for all \((k,n)\in S\times T\), ensure that the occurrence of \(A_k \cap H_k\) implies the occurrence of \(C_k\) for each \(k\in S\). Hence

\[
P(C_k) = P(A_k \cap H_k) = 1
\]

\[\therefore \quad P(C_k) = P\{\lim_{n \to \infty} [q_k^{-1}(n)]^{-1} q_k(n) = [q_k^{-1}]^{-1} q_k\} = 1\]

for each \(k\in S\). But

\[
q_k = q_k \quad q_k = [q_k^{-1}]^{-1} q_k \quad \text{(by lemma (L - 4.1))}
\]

\[
h_k(n) = q_k(n) \quad q_k(n) = [q_k^{-1}(n)]^{-1} q_k(n) \quad \text{(by definition)}
\]

for all \((k,n)\in S\times T\). Then:

\[
P(C_k) = P\{\lim_{n \to \infty} h_k(n) = q_k\} = 1
\]

for each \(k\in S\). □

Remarks:

1) (Constant parameters) Consider the particular case where the parameters \(\{\alpha_m ; m=1,\ldots,M\}\) in (1) are constant over \(X = (0,\xi)\). By (9) we can see that the components of \(\alpha_k\), the parameter vector defined in (P - 4.1), will be constant over all \(k\in S\) (i.e., \(\alpha_k = \alpha \in \mathbb{R}^{M+1}\) for all \(k\in S\); a space-invariant vector in \(\mathbb{R}^{M+1}\)). Thus, in order to perform the identification of \(\alpha\) using the algorithm (SA-1), it will be sufficient to take noisy measurements \(z_{k_o}(n+\eta_k)\) in just \(M+1\) points, say \(\{x_{k_o+\eta_1}, \ldots, x_{k_o+\eta_1}, \ldots, x_{k_o+\eta_{M+1}}\}\), located in the spatial domain \(X\), where \(k_o\) is any fixed point in \(S\). Moreover, the condition

\[\rho(x) \neq 0; \quad \forall x \in X\]
can be replaced by
\[ \beta_k = \beta(x_k) \neq 0 \quad \forall k = k_0 - N, \ldots, k_0, \ldots, k_0 + M \]
since, in this case, it is just required that the particular matrix
\( Q_k \in \mathbb{R}^{M+1} \) is positive definite (see (P - 4.3) and its proof).

2) (No noise condition) If \( d = 0 \) in the observation equation (2) we get
\[ E_k = 0, \quad \text{and so the identification algorithm (SA-1)} \]
\[ \text{does not depend on} \quad \text{the knowledge of} \quad \{ \delta_m(x_k) ; m = 0, \ldots, M \}. \]

3) (An indispensable information) Finally we recall that the information given in (11), that is:
\[ \sum_{m=1}^{M+1} a_k^m \quad \text{independent of} \quad \{ \delta_m(x_k) ; m = 1, \ldots, M \} \quad \text{for any} \quad k \in S, \]
represented a fundamental step in our identification procedure (when \( d \neq 0 \)), since it ensured that
\( \mu_k(x_k) \) in (P - 4.1) does not depend on
\( E_k \), and so the result of lemma (L - 4.1) could be achieved.

4.5 - RECOVERING THE ORIGINAL PARAMETERS

In this section we consider the 2nd step of the identification procedure. That is, we face the problem of determining the set of parameters \( \{ \alpha_m(x_k) ; m = 1, \ldots, M \} \) appearing in the distributed model (1-b), for each \( k \in S \).

To begin with, we recall the equation (9):
\[ a_k^{i+M+1} = \sum_{m=m_k(1)}^{M} (-1)^m \frac{\alpha_m(x_k)}{\delta_m(x_k)} \quad \text{for} \quad k \in S. \]
for each $i = -\bar{M}, -M+1, \ldots, \bar{M}$ and $k \in S$, where

$$m_\alpha(i) = \begin{cases} 
-2i & \text{if } i < -1 \\
1 & \text{if } i = 0 \\
2i - 1 & \text{if } i > 0 
\end{cases}$$

Or equivalently:

$$\alpha_{i+\bar{M}+1}^m = \sum_{m=1}^{\bar{M}} r_{im} \alpha_m(x_k)$$  \hfill (9')

where:

$$r_{im} = \begin{cases} 
(-1)^{m-i} \left( \frac{m}{i+\bar{m}} \right)^{\frac{1}{m-i}} & \text{if } m > m_\alpha(i) \\
0 & \text{if } m < m_\alpha(i) 
\end{cases}$$

Now define

$$\alpha_k = (\alpha_1(x_k), \ldots, \alpha_M(x_k)) \in \mathbb{R}^M; \ k \in S$$

$$R = [r_{im}]; \ \mathbb{R}^M \rightarrow \mathbb{R}^{\bar{M}+1}$$

and recall

$$\alpha_k = (\alpha_1^1, \ldots, \alpha_{\bar{M}+1}^M) \in \mathbb{R}^{\bar{M}+1}; \ k \in S$$

Then, by (9'), we get

$$\alpha_k = R \alpha_k; \ k \in S$$ \hfill (18)

The problem here is to recover the original parameters $\alpha_k \in \mathbb{R}^M$ from $\alpha_k \in \mathbb{R}^{\bar{M}+1}$, for each $k \in \mathcal{S}$. Since the stochastic approximation algorithm (SA-1) give us an estimate $\hat{\alpha}_k(n)$ of $\alpha_k$ for all $n \in \mathbb{T}$ and for each $k \in \mathcal{S}$, an estimate $\hat{\alpha}_k(n)$ of $\alpha_k$ is then supplied by means of equation (18)
for all $n \in \mathbb{T}$ and for each $k \in \mathcal{S}$:

$$a_k(n) = R \alpha_k(n); \quad k \in \mathcal{S}, \quad n \in \mathbb{T}.$$  \hfill (19)

But $a_k(n)$ and $\alpha_k(n)$ take values in $\mathbb{R}^{M+1}$ and $\mathbb{R}^M$, respectively. So it is not always possible to obtain a vector $\alpha_k(n)$ exactly satisfying the equation (19). A standard alternative consist of determining an estimate $\hat{\alpha}_k(n)$ of $\alpha_k(n)$ which best approximates a solution in the sense of minimizing the norm $\|a_k(n) - R \alpha_k(n)\|$ over all $\alpha_k(n)$. That is: a simple least-squares approach.

**Lemma (L - 4.3): (Least-Squares Estimate)** The estimate of $\alpha_k(n)$ which minimizes $\|a_k(n) - R \alpha_k(n)\|$ for each $(k,n) \in \mathcal{S} \times \mathbb{T}$ is given by

$$\hat{\alpha}_k(n) = (R^* R)^{-1} R^* a_k(n).$$

**Proof:**

By a direct inspection on the matrix $R$ we can conclude that it has linearly independent columns, and so the proof comes as in [11] pp. 83.

**Theorem (T - 4.2):** The random sequence in $\mathbb{R}^M \{ \hat{\alpha}_k(n); \ n \in \mathbb{T} \}$ obtained in (L - 4.3) converges to

$$\alpha_k = (\alpha_1(x_k), \ldots, \alpha_M(x_k)) \in \mathbb{R}^M$$

with probability one, for each $k \in \mathcal{S}$:

$$P\{ \lim_{n \to \infty} \hat{\alpha}_k(n) = \alpha_k \} = 1; \quad k \in \mathcal{S}. $$
Proof:

By (T - 4.1), (L - 4.3) and (18) we get:

\[ \hat{a}_k(n) = (R^* R)^{-1} R^* a_k(n) \xrightarrow{n \to \infty} (R^* R)^{-1} R^* a_k = (R^* R)^{-1} R^* a_k = a_k. \square \]

4.6 - AN EQUIVALENT PROCEDURE

We present here an equivalent procedure for the identification method developed in sections 4.2 through 4.5. It is based on formulating a slightly different version of the proposition (P - 4.1), as follows:

If we define

\[ i) \quad z_k(z_n) = \frac{1}{c_{k+1}} (z_k(z_n) - c_k z_k(n)) = \]

\[ = \sum_{m=1}^{N} \frac{c_m}{c_k} z_k(n+m-1) + z_k(n) \]

\[ = \frac{1}{c_{k+1}} z_k(z_n) \]

\[ ii) \quad v_k(z_n) = \frac{1}{c_{k+1}} v_k(z_n) \]

\[ iii) \quad a_k' = \frac{1}{c_{k+1}} (a_k - c_k a_{k+1}) = \]

\[ = \frac{1}{c_{k+1}} (a_k, \ldots, a_k, a_{k+1}, a_{k+2}, \ldots, a_{k+n+1}) \in \mathbb{R}^{n+1} \]

where:

\[ c_{k+1} = (0, \ldots, 0, 1, 0, \ldots, 0) \in \mathbb{R}^{n+1} \]

with 1 in the \((n+1)th\) position and zeros elsewhere.
iv) \( \xi_k^r = \frac{1}{c_{k}^{N+1}} \xi_k \in R^{M+1} \)

v) \( g_k^r = E \{ z_k(n+N_t) z_k^r(n+N_t) \} + \xi_k^r \in R^{M+1} \)

the results in (P - 4.1) and (L - 4.1) take the following form:

(P - 4.1): \( z_k^r(n) = \langle \xi_k^r ; z_k(n) \rangle + \xi_k^r \)

(L - 4.1): \( a_k^r = q_k q_k^r \)

and a similar version of the identification algorithm presented in (T - 4.1) is obtained when \( z^n, \xi_k, \) and \( a_k \) are replaced by \( z^n_r, \xi_k^r, \) and \( a_k^r, \) respectively:

(T - 4.1): \( a_k^r(n) \xrightarrow{n \to \infty} a_k^r \)

where:

\[ a_k^r(n) = a_k(n) \bigg|_{z^n = z^n_r \atop \xi_k = \xi_k^r} \]

in (SA-1).

Now define:

vi) \( \alpha_k^r = \frac{1}{c_k} \left( \frac{\alpha_1(x_k)}{\delta_x}, \ldots, \frac{\alpha_M(x_k)}{\delta_x} \right) \in R^M \)

vii) \( r_{im}^r = r_{im} \delta_m^x \)

viii) \( R' = \left[ r_{im}^r \right] : R^M \to R^{M+1} \)

So we get, from (9):

\[ e_k = R \alpha_k = c_k^{N+1} R' \alpha_k^r \]
and since

$$a_k' = \frac{c_k}{c_k^{N+1}} - \frac{c_{N+1}}{c_k^{N+1}} s_{N+1}$$

the equations (18) and (19) become:

(18): 

$$a_k' = R' a_k' - \frac{s_{N+1}}{c_k^{N+1}} s_{N+1}$$

(19): 

$$a_k'(n) = R' a_k'(n) - \frac{s_{N+1}}{c_k^{N+1}} s_{N+1}$$

In this way the least-squares estimate of $a_k'(n)$ is given by:

(L - 4.3): 

$$\hat{a}_k'(n) = (R'R')^{-1} R' \left[ a_k'(n) + \frac{s_{N+1}}{c_k^{N+1}} s_{N+1} \right],$$

and the convergence is proved as in (T - 4.2) by using (18), (T - 4.1) and (L - 4.3):

(T - 4.2): 

$$\frac{\hat{a}_k'(n)}{n} \xrightarrow{w.p.1} a_k' \quad.$$ 

This equivalent procedure was applied in [4] for a particular class of second-order models. Note that the original procedure identifies parameters appearing in the discrete version given by (5) (i.e., parameters of the matrix $A$), while the equivalent procedure identifies unknown parameters appearing in the system matrix $F$ given in (13) (more precisely, parameters of the matrix $A_{N+1} = c_{N+1}^{-1} (A - c_{N+1})$).
and since

$$\varphi_k = \frac{\varphi_k}{c_k} - \frac{\varphi_{k+1}}{c_k} \varphi_{k+1}$$

the equations (18) and (19) become:

(18): $$\varphi_k = R' \varphi'_k - \frac{\varphi_{k+1}}{c_k} \varphi_{k+1}$$

(19): $$\varphi_k(n) = R' \varphi'_k(n) - \frac{\varphi_{k+1}}{c_k} \varphi_{k+1}$$

In this way the least-squares estimate of $$\varphi_k(n)$$ is given by:

(L - 4.3): $$\hat{\varphi}_k(n) = (R'R)^{-1} R' \left[ \varphi'_k(n) + \frac{\varphi_{k+1}}{c_k} \varphi_{k+1} \right]$$

and the convergence is proved as in (T - 4.2) by using (18), (T - 4.1) and (L - 4.3):

(T - 4.2): $$\hat{\varphi}_k(n) \xrightarrow{n \to \infty} \varphi'_k.$$

This equivalent procedure was applied in [4] for a particular class of second-order models. Note that the original procedure identifies parameters appearing in the discrete version given by (5) (i.e., parameters of the matrix A), while the equivalent procedure identifies unknown parameters appearing in the system matrix F given in (13) (more precisely, parameters of the matrix $$A_{n+1} = C_{n+1}^{-1} (A - C_{n+1})$$).
Remark: (No noise condition) We can show that, if \( d = 0 \) in (2),
the stochastic approximation algorithm in (T - 4.1) also identifies

\[
\alpha_{k+1}^n = \sum_{m=0}^{M} (-1)^{m-j} \binom{m}{j} \frac{g_m(x_k)}{\delta_{x_k}}.
\]

Set

\[
R^n = \begin{bmatrix} R' & -c_{R+1} \\ c_{R+1} & -c_{R+1} \end{bmatrix} \in BL(R^{N+1})
\]

and note that there exists \( R'^{-1} \), since the independent columns of \( R' \)
are also independent of \( c_{R+1} \). Now defining

\[
\alpha_k^n = \begin{bmatrix} \alpha_k^n \\ c_k^{N+1} \end{bmatrix} = \begin{bmatrix} \alpha_1(x_k) \\ \vdots \\ \alpha_M(x_k) \\ \frac{g_{N+1}(x_k)}{\delta_{x_k}} \end{bmatrix} = R^{N+1}
\]

the equation (18) becomes:

\[
\alpha_k^n = R^n \alpha_k^n \in R^{N+1}
\]

and so we get a recursive estimate of \( \alpha_k^n \) directly from (T - 4.1)

\[
\alpha_k^n(n) = R^{-1} \alpha_k^i(n),
\]

since \( \alpha_k^i(n) \) does not depend on \( \{ \delta_m(x_k) ; m = 0, \ldots , M \} \) if \( d = 0 \).

4.7 - EXTENSION TO MULTI-DIMENSIONAL SPATIAL DOMAIN

So far we have been considering a one-dimensional spatial domain
\( x \in X \subset \mathbb{R}^1 \). Direct extensions of the theory developed in the previous
sections can be obtained for distributed models involving multi-dimen­sional spatial domains \( x \in X \subset \mathbb{R}^P \). In order to illustrate this, we perform
the main steps of the identification procedure for the following second-order \((M=2, N=1)\) linear model with two independent spatial variables \((x \in X \subset \mathbb{R}^2)\) and constant coefficients:

\[
\begin{align*}
\chi_0 u(x_1,x_2,t) + \chi_1 \frac{\partial}{\partial t} u(x_1,x_2,t) &= \\
&= \alpha_{11} \frac{\partial}{\partial x_1} u(x_1,x_2,t) + \alpha_{12} \frac{\partial}{\partial x_2} u(x_1,x_2,t) + \\
&+ \alpha_{21} \frac{\partial^2}{\partial x_1^2} u(x_1,x_2,t) + \alpha_{22} \frac{\partial^2}{\partial x_2^2} u(x_1,x_2,t) + \\
&+ \beta(x_1,x_2) w_{x_1,x_2}(t).
\end{align*}
\]

Since the solution method follows exactly as before, we will omit any comments and just the basic results will be presented in a concise form. The notation remains the same as in the preceding sections and, for simplicity, we consider homogeneous boundary conditions. The spatial domain \(X\) is taken to be an open square in \(\mathbb{R}^2\).

**Continuous formulation:** Infinite-dimensional state space.

\[
\begin{align*}
\text{MODEL:} & \quad \sum_{m=0}^{1} \gamma_m \frac{\partial^m}{\partial t^m} u(x,t) = \sum_{i=1}^{2} \alpha_{mi} \frac{\partial^m}{\partial x_i^m} u(x,t) + \beta(x) w_x(t) \\
& \quad x = (x_1,x_2) \in X = (0,\ell) \times (0,\ell) \subset \mathbb{R}^2 \quad ; \quad t > 0 \\
\text{IC:} & \quad u(x,0) = \varepsilon(x) \quad ; \quad x \in \bar{X} = [0,\ell] \times [0,\ell] \\
\text{BC:} & \quad u(x',t) = 0 \quad ; \quad x' \in \Gamma : \text{the boundary of } X \quad ; \quad t > 0 \\
\text{OBS:} & \quad z(x_k,t) = h u(x_k,t) + d v(t) \\
& \quad x_k \in P_x : \text{a partition of } \bar{X} \subset \mathbb{R}^2 \quad ; \quad t > 0
\end{align*}
\]
where:

i) The input disturbances \( \{ w_x(t) ; t \geq 0 \} \) are taken to be real-valued second-order stochastic processes for each \( x \) in \( (0, \ell) \times (0, \ell) \subseteq \mathbb{R}^2 \).

ii) The observation noise \( \{ v(t) ; t > 0 \} \) is a stochastic process defined as before.

iii) \( \{ \alpha_{m_i} ; m_i = 1, 2 \} ; \{ \beta_{m} ; m = 0, 1 \} \), \( h \) and \( d \) are assumed to be real constant parameters; and \( \delta_1 > 0 \), \( h > 0 \).

iv) \( \beta(x) > 0 ; \) for all \( x \in X \).

**Space-time discretization:** Finite-dimensional discrete version.

**MODEL:**

\[
\begin{align*}
&c_1 u_{k_1, k_2}(n) + c_2 u_{k_1, k_2}(n) = a_1 u_{k_1-1, k_2}(n) + \\
&a_21 u_{k_1, k_2-1}(n) + a_{22} u_{k_1, k_2}(n) + a_{23} u_{k_1, k_2+1}(n) + \\
&+ a_3 u_{k_1+1, k_2}(n) + \alpha_{k_1, k_2} w_{k_1, k_2}(n) \\
&(k_1, k_2) \in S \times S ; \quad n \in \bar{T}
\end{align*}
\]

**IC:**

\[
\begin{align*}
&u_{k_1, k_2}(0) = \xi_{k_1, k_2} ; \quad (k_1, k_2) \in S \times S
\end{align*}
\]

**BC:**

\[
\begin{align*}
&u_{0, k_2}(n) = u_{K+1, k_2}(n) = 0 \\
&(k_1, k_2) \in S \times S ; \quad n \in \bar{T}
\end{align*}
\]

**OBS:**

\[
\begin{align*}
&z_{k_1, k_2}(n) = \nu_{k_1, k_2}(n) + d v(n) \\
&(k_1, k_2) \in S \times S ; \quad n \in T
\end{align*}
\]
where:

i) \( x_{k_1} = k_1 \delta x_1 \in (0, t) \); \( i = 1, 2 \) (see fig. 7)

i-a) \( \delta x_1 = \delta x_2 = \delta x \)

i-b) \( x_1 \in S = \{0, 1, \ldots, K+1\} \Rightarrow S = \{1, 2, \ldots, K\} \)

i-c) \( K = \frac{\ell}{\delta x} - 1 \)

i-d) \( \delta x > 0 \) such that \( K \) is an integer \( \geq 3 \)

ii) \( t_n = n \delta t \approx 0 \)

ii-a) \( n \in \mathbb{N} = \{0, 1, \ldots\} \Rightarrow T = \{1, 2, \ldots\} \)

ii-b) \( \delta t > 0 \)

iii) \( u_{k_1, k_2}(n) = u(x_{k_1}, x_{k_2}, t_n) \)

iii-a) \( \frac{\partial}{\partial t} u(x_{k_1}, x_{k_2}, t) \bigg|_{t=t_n} \approx D_{\delta t} u(x_{k_1}, x_{k_2}, t_n) = \)

\[
= \frac{1}{\delta t} \left[ u_{k_1, k_2}(n+1) - u_{k_1, k_2}(n) \right]
\]

iii-b) \( \frac{\partial}{\partial x_1} u(x, t_n) \bigg|_{x=x_{k_1}} \approx D_{\delta x_1} u(x_{k_1}, x_{k_2}, t_n) = \)

\[
= \frac{1}{\delta x_1} \left[ u_{k_1+1, k_2}(n) - u_{k_1, k_2}(n) \right] ; \quad i = 1
\]

\[
= \frac{1}{\delta x} \left[ u_{k_1, k_2+1}(n) - u_{k_1, k_2}(n) \right] ; \quad i = 2
\]
\[
\begin{align*}
\text{iii-c)} & \quad \frac{\partial^2}{\partial x_1^2} u(x, t_n) \bigg|_{x_1 = x_k_1} = \frac{\partial^2}{\partial x_1^2} u(x_{k_1}, x_{k_2}, t_n) = \\
& = \frac{1}{\delta_x^2} \left\{ \begin{array}{l}
[u_{k_1-1, k_2}(n) - 2u_{k_1, k_2}(n) + u_{k_1+1, k_2}(n)] \quad ; \quad i = 1 \\
[u_{k_1, k_2-1}(n) - 2u_{k_1, k_2}(n) + u_{k_1, k_2+1}(n)] \quad ; \quad i = 2
\end{array} \right.
\end{align*}
\]

iv) \quad w_{k_1, k_2}(n) = w_{x_{k_1}, x_{k_2}}(t_n)

v) \quad v(n) = v(t_n)

vi) \quad z_{k_1, k_2}(n) = z(x_{k_1}, x_{k_2}, t_n)

\[
\begin{align*}
\text{vii)} & \quad a_1 = \frac{1}{\delta_x} \alpha_{21} \\
& \quad a_{21} = \frac{1}{\delta_x} \alpha_{22} \\
& \quad a_{22} = -\frac{1}{\delta_x} \left( \alpha_{11} + \alpha_{12} + \frac{2\alpha_{21}}{\delta_x} + \frac{2\alpha_{22}}{\delta_x} \right) \\
& \quad a_{23} = \frac{1}{\delta_x} \left( \alpha_{12} + \frac{\alpha_{22}}{\delta_x} \right) \\
& \quad a_3 = \frac{1}{\delta_x} \left( \alpha_{11} + \frac{\alpha_{21}}{\delta_x} \right)
\end{align*}
\]

viii) \[
\begin{align*}
& \quad c_1 = \delta_0 - \frac{\delta_1}{\delta_t} \\
& \quad c_2 = \frac{\delta_1}{\delta_t}
\end{align*}
\]

ix) \quad \beta_{k_1, k_2} = \beta(k_1, k_2)

125
Fig. 7: Discretization of the Space-Domain
\( \mathcal{X} = [0, \xi] \times [0, \xi] \).

**Equivalent discrete-time LPS:**

\[
\begin{align*}
\text{MODEL:} & \quad \mathbf{u}(n+1) = A \mathbf{u}(n) + B \mathbf{w}(n) ; \quad \mathbf{u}(0) \text{ given} \\
\text{OBS:} & \quad \mathbf{z}(n) = h \mathbf{u}(n) + d \mathbf{v}(n) ; \quad n \in \mathbb{T}
\end{align*}
\]

where:

1) \( \mathbf{u}(n) = (u_{11}(n), \ldots, u_{1K}(n), \ldots, u_{K1}(n), \ldots, u_{KK}(n)) \); \( n \in \mathbb{T} \)

2) \( \mathbf{w}(n) = (w_{11}(n), \ldots, w_{1K}(n), \ldots, w_{K1}(n), \ldots, w_{KK}(n)) \); \( n \in \mathbb{T} \)
Fig. 7: Discretization of the Space-Domain
\[ \mathbb{X} = [0, \ell] \times [0, c]. \]

Equivalent discrete-time LPS:

\[
\begin{align*}
\text{MODEL:} & \quad u(n+1) = A \, u(n) + B \, w(n) ; \quad u(0) \text{ given} \\
\text{OBS:} & \quad z(n) = h \, u(n) + d \, v(n) ; \quad n \in \mathbb{T}
\end{align*}
\]

where:

\[
\begin{align*}
i) \quad & u(n) = (u_{11}(n), \ldots, u_{1K}(n), \ldots, u_{K1}(n), \ldots, u_{KK}(n)) ; \quad n \in \mathbb{T} \\
ii) \quad & w(n) = (w_{11}(n), \ldots, w_{1K}(n), \ldots, w_{K1}(n), \ldots, w_{KK}(n)) ; \quad n \in \mathbb{T}
\end{align*}
\]
iii) \( \mathbf{x}(n) = (x_{11}(n), \ldots, x_{1K}(n), \ldots, x_{K1}(n), \ldots, x_{KK}(n)) ; \quad n \in \mathbb{T} \)

random vectors in \( \mathbb{R}^{K^2} \).

iv) \( \mathbf{d} = d(1,1,\ldots,1) \quad \in \mathbb{R}^{K^2} \)

v) \( A = \begin{bmatrix} A_2 & A_3 \\ A_1 & A_2 \\ & \ddots & \ddots \\ & & A_1 & A_2 \\ & & & A_1 & A_2 \end{bmatrix} \quad \in \mathbb{BL}(\mathbb{R}^{K^2}) \)

vi) \( A_1 = \frac{a_1}{c_2} I \quad \in \mathbb{BL}(\mathbb{R}^K) \)

vii) \( A_3 = \frac{a_3}{c_2} I \quad \in \mathbb{BL}(\mathbb{R}^K) \)

viii) \( A_2 = \frac{1}{c_2} \begin{bmatrix} a_{22} & a_{23} \\ a_{21} & a_{22} \\ & \ddots & \ddots \\ & & a_{21} & a_{22} \end{bmatrix} - \frac{a_1}{c_2} I \quad \in \mathbb{BL}(\mathbb{R}^K) \)

ix) \( B = \frac{1}{c_2} \begin{bmatrix} \beta_{11} \\ \beta_{1K} \\ & \ddots & \ddots \\ & & \beta_{K1} \\ & & & \beta_{KK} \end{bmatrix} \quad \in \mathbb{BL}(\mathbb{R}^{K^2}) \)
Identification Procedure: Since in this case we still have
\[ a_1 + \sum_{i=1}^{3} a_{2i} + a_3 = 0 , \]
a relation expressing the observation dynamics, as presented in (P - 4.1),
can be obtained as follows:

Define

1) \( \mathcal{S} = \{2, \ldots, K-1\} \subset S \)
2) \( z_{k_1,k_2}(\bar{z}_n) = c_1 z_{k_1,k_2}(n) + c_2 z_{k_1,k_2}(n+1) \)
3) \( \gamma_{k_1,k_2}(\bar{z}_n) = d(c_1 v(n) + c_2 v(n+1)) + h a_{k_1,k_2} w_{k_1,k_2}(n) \)
4) \( a = (a_1, a_{21}, a_{22}, a_{23}, a_3) \in \mathbb{R}^5 \)
5) \( z_{k_1,k_2}(n) = (z_{k_1-1,k_2}(n), z_{k_1,k_2-1}(n), z_{k_1,k_2}(n), z_{k_1,k_2+1}(n), z_{k_1+1,k_2}(n)) \); random vectors in \( \mathbb{R}^5 \).

Proposition (P - 4.1'): For each \((k_1, k_2) \in \mathcal{S} \times \mathcal{S}\) and \(n \in T\),
\[ z_{k_1,k_2}(\bar{z}_n) = \langle a, z_{k_1,k_2}(n) \rangle + \gamma_{k_1,k_2}(\bar{z}_n) . \]

Based on (P - 4.1') it can be shown that the parameter explicit
lemma (L - 4.1) has a similar version in case of multi-dimensional spatial
domain. First consider the assumptions (A - 4.1) - (A - 4.4) (stability,
stationarity, steady state and finite transient time), where the input
disturbance \( \{w(n) ; n \in \mathbb{T}\} \) is a random sequence in \( \mathbb{R}^{K^2} \) and so \( C_w \) is a sym-
metric positive definite matrix in \( \text{BL}(\mathbb{R}^{K^2}) \). Now define
\[ i) \quad \mathbf{E} = -d^2 [N^2_\nu \delta_0 + c_1 \delta_\nu(0) + c_2 \delta_\nu(1)] \]

\[ ii) \quad \mathbf{E} = \varepsilon(1,1,\ldots,1) \in \mathbb{R}^5 \]

\[ iii) \quad \mathbf{Q}_{k_1,k_2} = E\{z_{k_1,k_2}(n+N_t) z_{k_1,k_2}^*(n+N_t)\} + \bar{\mathbf{E}} \in \mathbb{R}^5 \]

\[ iv) \quad \mathbf{Q}_{k_1,k_2}^{-1} = E\{z_{k_1,k_2}(n+N_t) z_{k_1,k_2}^*(n+N_t)\} \in \mathbb{B}(\mathbb{R}^5) \]

\[ v) \quad J_{k_1,k_2} = \begin{bmatrix} 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \end{bmatrix} : \mathbb{R}^2 \rightarrow \mathbb{R}^5 \]

with \(I \in \mathbb{B}(\mathbb{R}^3)\) centered at the \([(k_1-1)K+k_2]\)th position and the 1s in the first and fifth rows, placed at \([(k_1-2)K+k_2]\)th and \([(k_1K+k_2)]\)th positions, respectively.

Since

\[ z_{k_1,k_2}(n) = J_{k_1,k_2} z(n) \]

the propositions \(P - 4.2) - (P - 4.5)\) remain valid if we replace \(k \in \mathbb{S}\) by \((k_1,k_2) \in \mathbb{S} \times \mathbb{S}\). Hence:

**Lemma (L - 4.1')**: (Explicit parameter) Let \(\mathbf{Q}_{k_1,k_2}\) and \(\mathbf{Q}_{k_1,k_2}^{-1}\) be as defined before. If the assumptions \((A - 4.1) - (A - 4.4)\) are satisfied, then the parameter vector

\[ \mathbf{a} = (a_1, a_2, a_3, a_2, a_3) \in \mathbb{R}^5 \]

introduced in \((P - 4.1')\) can be placed in an explicit form, as follows:

\[ \mathbf{a} = \mathbf{Q}_{k_1,k_2} \mathbf{Q}_{k_1,k_2}^{-1} \]
1) $\mathbf{v} = -a^2 \left[ \gamma_v \delta_0 + c_1 \delta_v(0) + c_2 \delta_v(1) \right]$

ii) $\mathbf{v} = \mathbf{v}(1,1,\ldots,1) \in \mathbb{R}^5$

iii) $\mathbf{Q}_{k_1,k_2} = E \{ \mathbf{z}_{k_1,k_2}(n+N_t), \mathbf{z}_{k_1,k_2}(n+N_t) \} + \mathbf{v} \in \mathbb{R}^5$

iv) $\mathbf{Q}_{k_1,k_2}^{-1} = E \{ \mathbf{z}_{k_1,k_2}(n+N_t), \mathbf{z}_{k_1,k_2}(n+N_t) \} \in BL(\mathbb{R}^5)$

v) $J_{k_1,k_2} = \begin{bmatrix} 0 & \ldots & 1 & 0 & \ldots \\ \vdots & \ddots & \vdots & \vdots & \ddots \\ \phi & \cdots & \mathbf{I} & \cdots & \phi \\ 0 & \ldots & 1 & \ldots & 0 \end{bmatrix} : \mathbb{R}^{k_2} \rightarrow \mathbb{R}^5$

with $I \in BL(\mathbb{R}^3)$ centered at the $[(k_1-1)K+k_2]$th position and the 1s in the first and fifth rows, placed at $[(k_1-2)K+k_2]$th and $[k_1K+k_2]$th positions, respectively.

Since

$\mathbf{E}_{k_1,k_2}(n) = J_{k_1,k_2} \mathbf{z}(n)$

the propositions (P - 4.2) - (P - 4.5) remain valid if we replace $k \in S$ by $(k_1,k_2) \in S \times S$. Hence:

Lemma (L - 4.1'): (Explicit parameter) Let $\mathbf{Q}_{k_1,k_2}$ and $\mathbf{Q}_{k_1,k_2}$ be as defined before. If the assumptions (A - 4.1) - (A - 4.4) are satisfied, then the parameter vector

$\mathbf{a} = (a_1,a_{21},a_{22},a_{23},a_3) \in \mathbb{R}^5$

introduced in (P - 4.1') can be placed in an explicit form, as follows:

$\mathbf{a} = \mathbf{Q}_{k_1,k_2} \mathbf{Q}_{k_1,k_2}$

[129]
for any \((k_1, k_2) \in S \times S\).

In this way, the parameter vector \(a\) can be identified through noisy observations \(z_{k_1, k_2}(n)\), by using stochastic approximation algorithms as presented in (T - 4.1) (with \(k \in S\) replaced by \((k_1, k_2) \in S \times S\)).

Remarks:

1) As far as the equivalent LPS is concerned, the computational complexity increases exponentially with the dimension of the spatial domain \(X\). For instance, assume \(X\) is an open rectangle in \(R^P\), and let \(K\) be a fixed integer \((\geq M+1)\) such that the discretization of \(X\) contains \(K^P\) interior points (i.e., \(K\) interior points for each discretized coordinate: \(S = \{1, 2, \ldots, K\}\)). So, as shown for the two-dimensional case (with \(N = 1\)), the equivalent LPS will be of order \((NK)^P\) (i.e., \(\chi(n)\) is in \(R^{(N\times K)^P}\)).

2) For identification purposes, the computational complexity increases only with the number of parameters to be identified (i.e., the identification algorithm \(a_k(n)\) is in \(R^{pM+1}\) for each \(k\)).
The performance of the identification method proposed in chapter 4 is analysed. After a brief summary concerning second-order models, we present three examples dealing with parabolic and hyperbolic PDEs. The chapter closes with a concise list of remarks including some conclusions and suggestions for further research in this field.

5.1 - SUMMARY: SECOND-ORDER MODELS

We present here a brief summary of the identification procedure developed in the last chapter, for second-order models ($M = 2$, $0 < N \leq 2$) with constant (space-invariant) parameters and one-dimensional spatial domain ($x \in (0, \ell)$). Two cases will be considered separately: $M = 2$, $N = 1$ and $M = N = 2$. ¹

¹ Recall that

\[
\begin{align*}
M = 2 & \quad \rightarrow \quad \tilde{M} = \bar{M} = 1 \\
N = \begin{cases} 
1 & \rightarrow \quad \tilde{N} = 0, \ \bar{N} = 1 \\
2 & \rightarrow \quad \tilde{N} = \bar{N} = 1
\end{cases}
\end{align*}
\]
Original DPS:

\[
\begin{align*}
&M=1: \quad \delta_0 u(x, t) + \delta_1 \frac{\partial}{\partial t} u(x, t) \\
&M=2: \quad \delta_0 u(x, t) + \delta_1 \frac{\partial}{\partial t} u(x, t) + \delta_2 \frac{\partial^2}{\partial t^2} u(x, t) \\
&= \alpha_1 \frac{\partial}{\partial x} u(x, t) + \alpha_2 \frac{\partial^2}{\partial x^2} u(x, t) + \beta(x) v(x) ; \quad x \in (0, \ell) ; \quad t > 0
\end{align*}
\]

IC: \[u(x, 0) = \xi_0(x) ; \quad x \in [0, \ell] \]

BC: \[u(0, t) = u(\ell, t) = 0 ; \quad t > 0 \]

OBS: \[z(x, t) = h u(x, t) + dv(t) ; \quad x \in [0, \ell] \]

Discrete version:

\[
\begin{align*}
&M=1: \quad c_1 u_k(n) + c_2 u_{k+1}(n) \\
&M=2: \quad c_1 u_{k-1}(n) + c_2 u_k(n) + c_3 u_{k+1}(n) \\
&= a_1 u_{k-1}(n) + a_2 u_k(n) + a_3 u_{k+1}(n) + \beta_k v_k(n) ; \quad (k, n) \in S \times T
\end{align*}
\]

IC: \[u_k(0) = \xi_0(x_k) ; \quad k \in S \]

BC: \[u_0(n) = u_{K+1}(n) = 0 ; \quad n \in \mathbb{T} \]

OBS: \[z_k(n) = h u_k(n) + dv(n) ; \quad (k, n) \in S \times T \]
where the coefficients \( a_1, a_2, a_3 \) and \( c_1, c_2, c_3 \) are given by:

\[
a_1 = \frac{\alpha_2}{\delta_x}
\]

\[
a_2 = -\frac{\alpha_1}{\delta_x} - 2 \frac{\alpha_2}{\delta_x}
\]

\[
a_3 = \frac{\alpha_1}{\delta_x} + \frac{\alpha_2}{\delta_x}
\]

\[
\begin{align*}
c_1 &= \delta_0 - \frac{y_1}{\delta_t} \\
c_2 &= \frac{y_1}{\delta_t}
\end{align*}
\]

\[
\begin{align*}
c_1 &= \frac{\delta_2}{\delta_t} \\
c_2 &= \delta_0 - \frac{y_1}{\delta_t} - 2 \frac{y_2}{\delta_t^2} \\
c_3 &= \frac{y_1}{\delta_t} + \frac{y_2}{\delta_t^2}
\end{align*}
\]

and the sets \( T, T', T \) and \( S \) are defined as follows:

\[
T = \{0,1,2,...\}
\]

\[
\begin{align*}
N=1: & \quad T = \{1,2,...\} \subseteq T' = T \\
N=2: & \quad T = \{2,3,...\} \subseteq T' = \{1,2,...\} \subseteq T
\end{align*}
\]

\[
S = \{1,2,...,K\}
\]

with

\[
K = \frac{t}{\delta_x} - 1 : \quad \text{an integer } \geq 3.
\]
Equivalent discrete-time LFS:

\[
\begin{align*}
\text{MODEL:} & \quad \chi(n+1) = F \chi(n) + G w(n) ; \quad \chi(N) \text{ given} \\
\text{OBS:} & \quad \xi(n) = H \chi(n) + d \nu(n) ; \quad n \in T \\
\text{where:} & \\
1) & \quad \underline{u}(n) = (u_1(n), \ldots, u_K(n)) ; \quad n \in T : \text{ in } \mathbb{R}^K \\
2) & \quad \underline{w}(n) = (w_1(n), \ldots, w_K(n)) ; \quad n \in T : \text{ in } \mathbb{R}^K \\
3) & \quad \underline{z}(n) = (z_1(n), \ldots, z_K(n)) ; \quad n \in T : \text{ in } \mathbb{R}^K \\
4) & \quad d = d(1,1, \ldots, 1) \in \mathbb{R}^K \\
5) & \quad B = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_K \end{bmatrix} \in BL(\mathbb{R}^K) \\
6) & \quad A = \begin{bmatrix} a_2 & a_3 \\ a_1 & a_2 & a_3 \\ & a_1 & a_2 \end{bmatrix} \in BL(\mathbb{R}^K) \\
N = 1: & \\
7) & \quad \chi(n) = \underline{u}(n) ; \quad n \in T = T : \text{ in } \mathbb{R}^K \\
8) & \quad \chi(N) = \chi(0) = \underline{u}(0) = (\xi_0(x_1), \ldots, \xi_0(x_K)) \\
9) & \quad F = A_1 = c_2^{-1}(c_1 - A) = \frac{1}{c_2}(A - c_1 I) = 
\end{align*}
\]
where:
\[
\begin{align*}
 a_i^j &= \frac{a_1}{c_2} = \alpha_i^j \\
 a_2^j &= \frac{a_2 - c_1}{c_2} = -\alpha_1^j - 2\alpha_2^j - \frac{c_1}{c_2} = 1 - \left(\alpha_1^j + 2\alpha_2^j + \delta \frac{\gamma_0}{\delta_1}\right) \\
 a_3^j &= \frac{a_3}{c_2} = \alpha_3^j + \alpha_2^j
\end{align*}
\]
with the coefficients $\alpha_i^j$ and $\alpha_2^j$ defined as follows:
\[
\begin{align*}
 \alpha_1^j &= \frac{1}{c_2} \frac{\alpha_1}{\delta_1} = \frac{1}{\delta_1} \frac{\delta_t}{\delta_x} \alpha_1 \\
 \alpha_2^j &= \frac{1}{c_2} \frac{\alpha_2}{\delta_1} = \frac{1}{\delta_1} \frac{\delta_t}{\delta_x} \alpha_2
\end{align*}
\]

ix-a) $G = B_1 = c_2^{-1} B = \frac{1}{c_2} B = \frac{\delta_t}{\delta_1} B \in \text{BL}(R^K)$

x-a) $H = h I \in \text{BL}(R^K)$

$N = 2$:

\[\begin{align*}
 \psi(n) &= (u(n-1), u(n)) ; \quad n \in T' \quad \text{in } R^{2K} \\
 \psi(N) &= \psi(1) = (u(0), u(1)) = \\
 &= (c_0(x_1), \ldots, c_0(x_K), \delta_t e_1(x_1) + e_0(x_1), \ldots, \delta_t e_1(x_K) + e_0(x_K))
\end{align*}\]
\[ F = \begin{bmatrix} 0 & I \\ A_1 & A_2 \end{bmatrix} \in BL(\mathbb{R}^{2K}) \]

\[ A_1 = -c_3^{-1} c_1 = -\frac{c_1}{c_3} I = -\frac{\gamma_2}{\gamma_1 \delta_t + \gamma_2} I \in BL(\mathbb{R}^K) \]

\[ A_2 = c_3^{-1}(A - c_2) = \frac{1}{c_3} (A - c_2 I) = \]

\[
\begin{bmatrix}
  a_1^1 & a_2^1 & a_3^1 \\
  a_1^2 & a_2^2 & a_3^2 \\
  & & \\
  a_1^3 & a_2^3 & a_3^3
\end{bmatrix}
\in BL(\mathbb{R}^K)
\]

where:

\[ a_1^i = \frac{a_1}{c_3} - \alpha_2 \]

\[ a_2^i = \frac{a_2 - c_2}{c_3} = -\alpha_1 - 2\alpha_2 - \frac{c_2}{c_3} = 2 - (\alpha_1 + 2\alpha_2 + \delta_1 \frac{\delta_2}{\delta_1 + \delta_2} + \delta_2) \]

\[ a_3^i = \frac{a_3}{c_3} = \alpha_1 + \alpha_2 \]

with the coefficients \( \alpha_1 \) and \( \alpha_2 \) defined as follows:

\[ \alpha_1 = \frac{1}{c_3} \frac{c_1}{\delta_x} = \frac{1}{\delta_1 \delta_t + \delta_2} \frac{\delta_2}{\delta_x} \alpha_1 \]

\[ \alpha_2 = \frac{1}{c_3} \frac{c_2}{\delta_x} = \frac{1}{\delta_1 \delta_t + \delta_2} \frac{\delta_1}{\delta_x} \alpha_2 \]
\[ F = \begin{bmatrix} 0 & 1 \\ A_1 & A_2 \end{bmatrix} \in \text{BL}(\mathbb{R}^{2K}) \]

\[ A_1 = -c_3^{-1} c_1 = -\frac{c_1}{c_3} I = \frac{-\lambda_2}{\delta_1 \delta_t + \delta_2} I \in \text{BL}(\mathbb{R}^K) \]

\[ A_2 = c_3^{-1}(A - c_2) = \frac{1}{c_3}(A - c_2 I) = \]

\[ \begin{bmatrix} a'_2 & a'_3 \\ a'_1 & a'_2 & a'_3 \\ \vdots & \vdots & \vdots \\ a'_1 & a'_2 & a'_3 \end{bmatrix} \in \text{BL}(\mathbb{R}^K) \]

where:

\[ a'_1 = \frac{a_1}{c_3} = \alpha'_2 \]

\[ a'_2 = \frac{a_2 - c_2}{c_3} = -\alpha'_1 - 2\alpha'_2 - \frac{c_2}{c_3} = 2 - (\alpha'_1 + 2\alpha'_2 + \frac{\delta_0}{\delta_1 \delta_t + \delta_2}) \]

\[ a'_3 = \frac{a_3}{c_3} = \alpha'_1 + \alpha'_2 \]

with the coefficients \( \alpha'_1 \) and \( \alpha'_2 \) defined as follows:

\[ \alpha'_1 = \frac{1}{c_3} \frac{c_1}{\delta_x} = \frac{1}{\delta_1 \delta_t + \delta_2} \frac{\delta_t^2}{\delta_x} \alpha_1 \]

\[ \alpha'_2 = \frac{1}{c_3} \frac{c_2}{\delta_x^2} = \frac{1}{\delta_1 \delta_t + \delta_2} \frac{\delta_t^2}{\delta_x^2} \alpha_2 \]
ix-b) $G = \begin{bmatrix} 0 \\ B_2 \end{bmatrix} : \mathbb{R}^K \rightarrow \mathbb{R}^{2K}$

$$B_2 = C^{-1}_3 B = \frac{1}{c_3} B = \frac{\delta_2^2}{\delta_1 \delta_2} B \in \text{BL}(\mathbb{R}^K)$$

x-b) $H = h \begin{bmatrix} 0 & I \end{bmatrix} : \mathbb{R}^{2K} \rightarrow \mathbb{R}^K$

Stochastic approximation algorithms for identification:

**1st case:** The original procedure:

(CA-1): $a(n+1) = \frac{1 - \lambda(n)}{1 - \mu(n)} \left[ 1 - \mu(n) Q(n+1) z_{k_0}^*(n+N_t) z_{k_0}^*(n+N_t) \right] a(n) +$

$$+ \lambda(n) Q(n+1) \left[ z_{k_0}^*(n+N_t) z_{k_0}^*(n+N_t) + \varepsilon \right] \quad \text{in} \, \mathbb{R}^3$$

with $Q(n)$ in $\text{BL}(\mathbb{R}^3)$ given by

(CA-2): $Q(n+1) = \frac{1}{1 - \mu(n)} \left[ Q(n) - \frac{\mu(n) Q(n) z_{k_0}^*(n+N_t) z_{k_0}^*(n+N_t) Q(n)}{1 - \mu(n) + \mu(n) z_{k_0}^*(n+N_t) Q(n) z_{k_0}^*(n+N_t)} \right]$

where:

i) $n \in \mathbb{N} = \{0,1,2,\ldots\}$

ii) $N_t$: finite transient time as in (A - 4.4)

iii) $k_0$: any fixed point in $S = \{2,\ldots,K-1\}$

iv) $\lambda(n) \in (0,1)$; $\sum_{n=0}^{\infty} \lambda(n) = \infty$; $\sum_{n=0}^{\infty} \lambda^2(n) < \infty$

v) $\mu(n) \in (0,1)$; $\sum_{n=0}^{\infty} \mu(n) = \infty$; $\sum_{n=0}^{\infty} \mu^2(n) < \infty$

137
vi) \( z_{k_0}'(\mathbf{a}_n) = \begin{cases} c_1 z_{k_0}(n) + c_2 z_{k_0}(n+1) & ; \ N = 1 \\ c_1 z_{k_0}(n-1) + c_2 z_{k_0}(n) + c_3 z_{k_0}(n+1) & ; \ N = 2 \end{cases} \)

vii) \( z_{k_0}(n) = J_{k_0} \mathbf{z}(n) = (z_{k_0-1}, z_{k_0}, z_{k_0+1}(n)) ; \text{ in } R^3 \)

viii) \( \mathbf{z} = -d^2 \tau^2 \delta_0 - d^2 \left\{ \begin{array}{l} c_1 \delta_0^2(0) + c_2 \delta_0^2(1) & ; \ N = 1 \\ c_1 \delta_0^2(1) + c_2 \delta_0^2(0) + c_3 \delta_0^2(1) & ; \ N = 2 \end{array} \right. \)

ix) \( \mathbf{z} = \mathbf{z}(1,1,1) \in R^3 \)

x) \( a(0) : \text{a second-order random vector in } R^3, \text{ independent of } \{ z_{k_0}(n+N_t) ; n \in \mathbb{T} \}. \)

xi) \( Q(0) : \text{a second-order symmetric positive definite random matrix in } BL(R^3), \text{ independent of } \{ z_{k_0}(n+N_t), z_{k_0}(n+N_t) ; n \in \mathbb{T} \}. \)

**2nd case:** An equivalent procedure:

(SA-1): \( \mathbf{a}'(n+1) = \frac{1 - \lambda(n)}{1 - \mu(n)} \left[ I - \lambda(n) Q(n+1) z_{k_0}(n+N_t) z_{k_0}(n+N_t)^\ast \right] \mathbf{a}'(n) + \lambda(n) Q(n+1) \left[ z_{k_0}(\mathbf{z}_{n+N_t}) z_{k_0}(n+N_t)^\ast + \mathbf{z}' \right] \in R^3 \)

with

vi') \( z_{k_0}'(\mathbf{z}'_n) = \begin{cases} z_{k_0}(n+1) & ; \ N = 1 \\ \frac{c_1}{c_3} z_{k_0}(n-1) + z_{k_0}(n+1) & ; \ N = 2 \end{cases} \)

138
ix') \( \xi' \), \( \xi' \) is a second-order random vector in \( \mathbb{R}^3 \), independent of 
\( \{ z_k(n+N_t) ; n \in \mathbb{N} \} \).

In both cases the algorithms converge with probability one:

\[
\mathbf{a}(n) \xrightarrow{\text{w.p.} 1} \mathbf{a} = (a_1, a_2, a_3) \in \mathbb{R}^3 \\
\mathbf{a}'(n) \xrightarrow{\text{w.p.} 1} \mathbf{a}' = (a'_1, a'_2, a'_3) \in \mathbb{R}^3
\]

**Recovering the original parameters:**

1st case: Using the original procedure:

\[
\mathbf{a} = \mathbf{R} \mathbf{a}'
\]

where:

i) \( \mathbf{a} = (a_1, a_2, a_3) \in \mathbb{R}^3 \\
ii) \mathbf{a} = (a_1, a_2) \in \mathbb{R}^2 \\
iii) \mathbf{R} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{R}_{01} & \mathbf{R}_{02} \end{bmatrix} = \frac{1}{\delta x} \begin{bmatrix} 0 & 1 \\ -2 & -2 \end{bmatrix} : \mathbb{R}^2 \rightarrow \mathbb{R}^3
\]

The least-squares estimate is given by

\[
\hat{\mathbf{a}}(n) = (\mathbf{R}^\mathbf{R})^{-1} \mathbf{R}^\mathbf{R} \mathbf{a}_k(n) \xrightarrow{\text{w.p.} 1} \mathbf{a}
\]
where:

iv) \( \hat{\alpha}(n) = (\hat{\alpha}_1(n), \hat{\alpha}_2(n)) : \text{ in } \mathbb{R}^2 \)

v) \( \alpha(n) = (a_1(n), a_2(n), a_3(n)) : \text{ in } \mathbb{R}^3 \)

vi) \((R^* R)^{-1} R^* = \frac{\delta_x}{3} \begin{bmatrix} -3 & 0 & 1 \\ 2\delta_x & -\delta_x & -\delta_x \end{bmatrix} : \mathbb{R}^3 \rightarrow \mathbb{R}^2 \)

So we get:

\( \hat{\alpha}_1(n) = \frac{\delta_x}{3} [-a_1(n) + a_3(n)] \)

\( \hat{\alpha}_2(n) = \frac{\delta_x^2}{3} [2a_1(n) - a_2(n) - a_3(n)] \)

2nd case: Using the equivalent procedure:

\( R' = R' \alpha' - \frac{c_{n+1}}{c_{n+1}} s_2 \)

where:

i) \( \alpha' = (a_1', a_2', a_3') \in \mathbb{R}^3 \)

ii) \( \alpha' = (\alpha_1', \alpha_2') \in \mathbb{R}^2 \)

\[ \begin{align*}
N = 1: \\
\alpha_1' &= \frac{1}{c_2} \frac{\alpha_1}{\delta_x} = \frac{1}{\delta_1} \frac{\delta_x}{\delta_1} \alpha_1 \\
\alpha_2' &= \frac{1}{c_2} \frac{\alpha_2}{\delta_x} = \frac{1}{\delta_1} \frac{\delta_x}{\delta_1} \alpha_2
\end{align*} \]
The least-squares estimate is given by

\[ \hat{a}'(n) = (R'^* R'^*)^{-1} R'^* \left[ \hat{a}'(n) + \frac{c_{N+1}^{2N+1}}{c_{N+1}} e_2 \right] \xrightarrow{n \to \infty} \alpha' \]

where:

\[ \text{ii) } \hat{a}'(n) = (\hat{a}_1'(n), \hat{a}_2'(n)) : \text{ in } \mathbb{R}^2 \]

\[ \text{iii) } a'(n) = (a_1'(n), a_2'(n), a_3'(n)) : \text{ in } \mathbb{R}^3 \]

\[ \text{iv) } e_2 = (0, 1, 0) \in \mathbb{R}^3 \]

\[ \text{v) } \frac{c_{N+1}}{c_{N+1}} = \begin{cases} \frac{c_{1}}{c_{2}} = \delta_{t} \frac{\delta_{t} - 1}{\delta_{1}} & \text{if } N = 1 \\ \frac{c_{2}}{c_{3}} = \delta_{t} \frac{\delta_{t} + \delta_{2}}{\delta_{1}} - \delta_{t} - 1 & \text{if } N = 2 \end{cases} \]

\[ \text{So we get:} \]

\[ \hat{a}_1'(n) = -a_1'(n) + a_3'(n) \]

\[ \hat{a}_2'(n) = \frac{1}{3} \left[ 2a_1'(n) - a_2'(n) - a_3'(n) + 1 - \delta_{t} \frac{\delta_{0}}{\delta_{1}} \right] \]
\[
\begin{align*}
N=2: & \quad \hat{\alpha}_1(n) = -a_1(n) + a_j(n) \\
& \quad \hat{\alpha}_2(n) = \frac{1}{3} \left[ 2 a_1(n) - a_2(n) - a_3(n) + 2 - \delta_t \frac{\gamma_0 \delta_1 + \delta_2}{\delta_1 \delta_2 + \delta_2} \right]
\end{align*}
\]

Remarks: If \( \alpha_1 = 0 \) is known "a priori", we get the least-squares estimate for \( \alpha_2 \) as follows:

1st case:
\[
\begin{align*}
\mathbf{a} &= \mathbf{R} \alpha_2 \\
\mathbf{R} &= \frac{1}{\delta_x^2} \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} : \mathbb{R}^1 \rightarrow \mathbb{R}^3
\end{align*}
\]
So:
\[
\hat{\alpha}_2(n) = \frac{1}{\delta_x^2} \left[ a_1(n) - 2 a_2(n) + a_3(n) \right]
\]

2nd case:
\[
\begin{align*}
\mathbf{a}' &= \mathbf{R}' \alpha_2' = \frac{\alpha_{N+1}}{\alpha_{N+1}} \mathbf{a}_2 \\
\mathbf{R}' &= \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} : \mathbb{R}^1 \rightarrow \mathbb{R}^3
\end{align*}
\]
So:
\[
\begin{align*}
N=1: & \quad \hat{\alpha}_2(n) = \frac{1}{6} \left[ a_1(n) - 2 a_2(n) + a_3(n) + 2 - 2 \delta_t \frac{\gamma_0}{\delta_1} \right] \\
N=2: & \quad \hat{\alpha}_2(n) = \frac{1}{6} \left[ a_1(n) - 2 a_2(n) + a_3(n) + 4 - 2 \delta_t \frac{\gamma_0 \delta_2}{\delta_1 \delta_2 + \delta_2} \right]
\end{align*}
\]

A block diagram: Figure 8 shows a block diagram for system simulation and identification. The DPS simulation is carried out by using the equivalent discrete-time LPS. The parameter vector \( \mathbf{a} \) is identified on-line, through noisy observations \( z(n+N_t) \), via stochastic approximation algorithms given in (SA-1) and (SA-2). The original parameters \( \mathbf{a} = (\alpha_1, \alpha_2) \) are recovered from \( \mathbf{a}(n) \) by means of a simple least-squares.
Fig. 8: SYSTEM SIMULATION AND IDENTIFICATION.
Examples illustrating this DPS identification method will be presented in sections 5.3 through 5.5. The following assumptions were made when performing those examples.

**Simulation:** The original DPS was simulated by using the equivalent discrete-time LPS, where the spatial domain

\[ X = (0, \pi) \]

(i.e., \( \ell = \pi \)) was discretized with

\[ \delta_x = \pi/6 \]

So,

\[ X = 5 \rightarrow \mathcal{S} = \{2, 3, 4\} \subset S = \{1, 2, 3, 4, 5\} \]

The \( M+1 \times 3 \) observation points in \( X \) (i.e., \( x_{k-1}, x_k, x_{k+1} \) where \( k_0 \) is any fixed point in \( \mathcal{S} \)), the constants \( h \) and \( d \), the time sampling rate \( \delta_t \), the space-varying input parameter \( \beta(x) \), and the initial functions \( g_0(x) \) and \( g_1(x) \) were chosen as follows:

\[
\begin{align*}
\text{i)} & \quad k_0 = 3 \quad \Rightarrow \quad \begin{cases} 
 x_{k_0-1} = (k_0 - 1) \delta_x = \pi/3 \\
 x_{k_0} = k_0 \delta_x = \pi/2 \\
 x_{k_0+1} = (k_0 + 1) \delta_x = 2\pi/3 
\end{cases} \\
\text{ii)} & \quad h = d = 1 \\
\text{iii)} & \quad \delta_t = \begin{cases} 
 1/2 & \text{if } N = 1 \\
 \sqrt{1/2} & \text{if } N = 2 
\end{cases}
\end{align*}
\]
5.2 - General Assumptions for Examples

Examples illustrating this DPS identification method will be presented in sections 5.3 through 5.5. The following assumptions were made when performing those examples.

Simulation: The original DPS was simulated by using the equivalent discrete-time LPS, where the spatial domain

\[ x = (0, \pi) \]

(i.e., \( \xi = \pi \)) was discretized with

\[ \delta_x = \pi/6 \]

So,

\[ K = 5 \rightarrow \delta = \{2, 3, 4\} \subseteq \delta = \{1, 2, 3, 4, 5\} \]

The M+1=3 observation points in \( x \) (i.e., \( x_{k_0-1}, x_{k_0}, \) and \( x_{k_0+1} \) where \( k_0 \) is any fixed point in \( \delta \)), the constants \( h \) and \( d \), the time sampling rate \( \delta_t \), the space-varying input parameter \( \beta(x) \), and the initial functions \( e_0(x) \) and \( g_1(x) \) were chosen as follows:

\[ \begin{aligned}
    & 1) \quad k_0 = 3 \\
    & \begin{cases}
    x_{k_0-1} = (k_0 - 1) \delta_x = \pi/3 \\
    x_{k_0} = k_0 \delta_x = \pi/2 \\
    x_{k_0+1} = (k_0 + 1) \delta_x = 2\pi/3
    \end{cases}
\end{aligned} \]

\[ ii) \quad h = d = 1 \]

\[ iii) \quad \delta_t = \begin{cases}
    1/2 & \text{if } N = 1 \\
    1/2 & \text{if } N = 2
\end{cases} \]

144
iv) \( \theta(x) = 2 \sin(x) \); \( x \in (0, \pi) \)

v) \( \varepsilon_0(x) = \varepsilon_1(x) = 0 \); \( x \in [0, \pi] \)

**Input disturbance:** \( \{w_k(n); n \in T^1\} \) was chosen to be uniformly distributed in \( (-\sqrt{3} \delta_w, \sqrt{3} \delta_w) \) and uncorrelated for all \( k \in S \), such that:

\[
\delta_w^2 = \mathbb{E}\{w_k^2(n)\}; \quad \text{for all } k \in S
\]

\[
\sigma_w^2 = \delta_w^2 I \in BL(R^5)
\]

with \( \delta_w = 1/2 \) (the example in section 5.3 also consider the case with \( \delta_w = 1 \)).

**Observation noise:** \( \{v(n); n \in T\} \) was chosen to be uniformly distributed in \( (-\sqrt{3} \delta_v, \sqrt{3} \delta_v) \) such that:

\[
\mathbb{E}\{v(n)\} = \eta_v = 0
\]

\[
\mathbb{E}\{v(i)v(j)\} - \eta_v^2 = \delta_v (i - j) = \delta_v^2 \delta(i - j)
\]

with \( \delta_v = 1/4 \).

**Stochastic approximation algorithms (SA-1) and (SA-2):** The identification was carried out by using the equivalent procedure; that is, the algorithm in (SA-1) was

\[
\mathbf{a}(n) = (a_1(n), a_2(n), a_3(n)) \xrightarrow{\frac{n+1}{n+3}} \mathbf{a}' = (a_1', a_2', a_3') \in R^3
\]

The following situation was assumed:

1) \( N = 100 \)

ii) \( \lambda(n) = \frac{1}{n+2} \)

iii) \( f^*(n) = \frac{1}{n+3/2} \)
iv) \[ \mathbf{a}'(0) = \begin{cases} (1/2, 0, 1/2) \in \mathbb{R}^3 & \text{when } N = 1 \\ (1, 0, 1) \in \mathbb{R}^3 & \text{when } N = 2 \end{cases} \]

v) \( Q(0) = I \in \text{BL}(\mathbb{R}^3) \)

5.3 - PARABOLIC PDE WITH ONE PARAMETER:

Our first example considers the identification of a single parameter \( \alpha_2 \) appearing in the "heat equation":

\[
\frac{\partial}{\partial t} u(x,t) = \alpha_2 \frac{\partial^2}{\partial x^2} u(x,t) + 2 \sin(x) w_x(t)
\]

(i.e: \( N = 1, Y_0 = 0, \delta_1 = 1 \) and \( \alpha_1 = 0 \) known "a priori"). The simulation was carried out by using

\[
\alpha_2 = \frac{1}{2} \left( \frac{\pi}{6} \right)^2 \quad \rightarrow \quad \alpha_2 = \frac{1}{4}
\]

and so, the constant vector to be identified by (SA-1) is

\[
\mathbf{a}' = (a_1', a_2', a_3') = (\alpha_2', \alpha_1' - 2\alpha_2', \alpha_2')
\]

Figures 9 and 10 show the performance of the identification procedure, where

\[
\hat{\mathbf{a}}_2(n) = \frac{1}{\delta} \left[ a_1(n) + 2 (1 - a_2(n)) + a_3(n) \right]
\]

\[
\hat{\mathbf{a}}'(n) = \| \mathbf{a}'(n) - \mathbf{a}' \|^2
\]

Two cases were considered:

1) \( \delta_w = 1/2 \)

ii) \( \delta_w = 1 \)
Fig. 9: Estimate Performance for $\alpha_2^*$. 

$\hat{\alpha}_2(0) = 0.5$

$\circ$ $\delta_w = 1/2$

$\triangledown$ $\delta_w = 1$

$\alpha_2 = 0.25$
Fig. 10: Error Performance.

\[ s'(0) = 0.375 \]

- \( \delta_w = 1/2 \)
- \( \delta_w = 1 \)
5.4 - PARABOLIC PDE WITH TWO PARAMETERS:

Now we consider the identification of $\alpha_1$ and $\alpha_2$ appearing in a parabolic equation, as follows:

$$\frac{\partial}{\partial t} u(x,t) = \alpha_1 \frac{\partial}{\partial x} u(x,t) + \alpha_2 \frac{\partial^2}{\partial x^2} u(x,t) + 2 \sin(x) w(x,t)$$

(i.e.: $N=1, \beta_0=0, \beta_1=1$). For simulating the DFS we assumed

$$\alpha_2 = \frac{1}{2} \left( \frac{\pi}{6} \right)^2 \quad \rightarrow \quad \alpha_2' = \frac{1}{4}$$

and two cases were considered (the first one representing a model with extraneous terms):

i) $\alpha_1 = 0 \quad \rightarrow \quad \alpha_1' = 0$

ii) $\alpha_1 = - \frac{\pi}{30} \quad \rightarrow \quad \alpha_1' = - \frac{1}{10}$

In this way, the constant vector to be identified by (SA-1) is:

$$\hat{\alpha}' = (a_1', a_2', a_j') = (\alpha_2', 1-(\alpha_1'+2a_2'), \alpha_1'+a_2').$$

The performance of the identification procedure is shown on figures 11-13, where

$$\hat{\alpha}_1'(n) = -a_1'(n) + a_j'(n)$$

$$\hat{\alpha}_2'(n) = \frac{1}{3} \left[ 2a_1'(n) - a_2'(n) - a_j'(n) + 1 \right]$$

$$\bar{a}'(n) = \|a'(n) - \bar{a}'\|^2$$

149
\( \hat{\alpha}_1(n) \)

\( \hat{\alpha}_1(0) = 0.0 \)

- \( \alpha_1 = 0 \)
- \( \alpha_1 = -\pi/30 \)

\( \alpha_1 = 0.0 \)

\( \alpha_1 = -0.1 \)

Fig. 11: Estimate Performance for \( \alpha_1^* \).
Fig. 12: Estimate Performance for $\alpha_2'$. 

$\hat{\alpha}_2'(0) = 0.5$

- $\alpha_1 = 0$
- $\alpha_1 = -\pi/30$

$\alpha_2' = 0.25$
Fig. 13: Error Performance.
Finally we present an example illustrating the identification of $\alpha_1$ and $\alpha_2$ appearing in a hyperbolic equation:

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \alpha_1 \frac{\partial}{\partial x} u(x,t) + \alpha_2 \frac{\partial^2}{\partial x^2} u(x,t) + 2 \sin(x) w_x(t)$$

(i.e. $N=2$, $\dot{y}_0 = \dot{y}_1 = 0$, $\dot{y}_2 = 1$). The simulation was carried out by assuming

$$\alpha_2 = \left(\frac{\pi}{6}\right)^2 \quad \longrightarrow \quad \alpha_1^i = \frac{1}{2}$$

and two cases considered:

i) $\alpha_1 = \frac{\pi}{30} \quad \longrightarrow \quad \alpha_1^i = \frac{1}{10}$

ii) $\alpha_2 = -\frac{\pi}{30} \quad \longrightarrow \quad \alpha_1^i = -\frac{1}{10}$

So, the constant vector to be identified by (SA-1) is

$$\vec{a} = (a_1^i, a_2^i, a_3^i) = (\alpha_2^i, 2-(\alpha_1^i + 2\alpha_2^i), \alpha_1^i + \alpha_2^i).$$

Figures 14 - 16 shown the performance of the identification procedure, where

$$\hat{a}_1^i(n) = -a_1^i(n) + a_3^i(n)$$

$$\hat{a}_2^i(n) = \frac{1}{3} \left[ 2a_1^i(n) - a_2^i(n) - a_3^i(n) + 2 \right]$$

$$\vec{a}^*(n) = \|\vec{a}^i(n) - \vec{a}\|^2$$
Fig. 14: Estimate Performance for $\alpha_1$.

$\hat{q}_1(0) = 0.0$

- $\alpha_1 = \pi/30$
- $\alpha_1 = -\pi/30$

$\alpha_1 = 0.1$

$\alpha_1 = -0.1$
Fig. 15: Estimate Performance for $\alpha_2$. 

$\hat{\alpha}_2(0) = 1.0$

$\alpha_2 = 0.5$

$\alpha_1 = \pi/30$

$\alpha_1 = -\pi/30$
Fig. 16: Error Performance.
5.6 - CONCLUSIONS

At the end of chapter 2 we have surveyed and commented on the general DPS identification problem. In this final section we discuss the particular identification method introduced in this work.

The stochastic approximation theory was used to identify parameters in distributed systems operating in a stochastic environment. Some basic points which characterize the method are listed below.

1) CLASS OF MODELS: The models we have considered here are described by PDE and have the following properties:

1-a) Linear, of order N in t and M in the scalar spatial variable x, with space-varying parameters.

1-b) Extensions to multi-dimensional spatial domain (x ∈ X ⊂ R^p) were also considered in section 4.7.

1-c) Although we have not considered cross-terms partial derivatives in our model, they can be treated using the same technique. But, in this case, the choice of which kind of approximation (backward, centered, or forward operators) must be decided for each model containing a particular type of cross-terms partial derivatives.

1-d) The space-varying parameters to be identified are those multipling spatial derivatives (parameters appearing in L^M_x, the spatial-differential operator), that is: \{α_m(x) ; m = 1, ..., M\}. Parameters appearing in L^N_t (i.e., \{γ_m(x) ; m = 0, ..., N\}) are assumed to be known "a priori".

1-e) In case of constant parameters to be identified, the identification procedure can be simplified as commented on page 114.
2) METHOD CLASSIFICATION: The method is classified as class $\Gamma_2$ (see section 2.2), and so it presents two stages: model approximation (stage I) and parametric estimation (stage II).

3) REDUCTION TO A FINITE-DIMENSIONAL STATE SPACE: In the first stage the method comprises two basic steps:

3-a) An equivalent discrete-time LPS obtained by using higher order finite-differences (section 4.2).

3-b) A fundamental observation equation given in (P - 4.1), which was possible thanks to the results obtained by finite-difference techniques (see remark 3 on page 115).

4) EXPLICIT PARAMETER: The parameters appearing in the discrete version (chapter 4, equation (5)) were placed in an explicit form (L - 4.1). In this way it was possible to use stochastic approximation algorithms as a straightforward identification procedure, rather than a simple searching scheme for finding estimates previously obtained by means of any other optimization technique.

5) INPUT DISTURBANCE: The input $\{w(n)\}$ was taken to be a zero mean "white" random sequence with positive definite covariance matrix $C_w$, as in (A - 4.2). It is also possible to consider positive semi-definite covariance matrix (or even zero input), if the initial state is assumed to be random, as commented on page 103. It is worth to remember here that, in order to use our identification procedure (based on a straightforward applicability of stochastic approximation via explicit parameter lemma (L - 4.1)), we cannot have deterministic inputs and deterministic initial state together. But in this case the reduction to an equivalent discrete-time LPS (stage I), as developed in section 4.2, can still be used for identification.
purposes by applying in stage II some other known technique (e.g., see surveys in LPS identification mentioned in chapter 1) for parametric estimation in LPS driven by deterministic inputs.

6) OBSERVATIONS: We have assumed that noisy observations are available at a finite number of discrete points equidistantly located in the spatial domain. In case of constant parameters just M+1 of those points are required, as remarked on page 114. Also, in some special cases, the measurements can be taken at pre-selected (not necessarily equidistant) observation points, as commented in [1].

7) OBSERVATION NOISE: The noise {v(n)} corrupting the observations is assumed to be uncorrelated with the input disturbance as in (A-4.2), and its statistics {η_v, 6_v(0), ..., 6_v(N)} are supposed to be known.

8) BOUNDARY CONDITIONS: Although the method has been developed using homogeneous BC, nonhomogeneous BC may also be considered (see remark on page 93), as well as random boundary conditions.

9) INITIAL CONDITIONS: Both, deterministic and random initial state, can be considered as commented before (see pages 83 and 103).

10) STOCHASTIC APPROXIMATION ALGORITHMS FOR IDENTIFICATION: The algorithms in (SA-1) and (SA-2) have the following properties:

10-a) No restriction on specific types of probability distributions is imposed.

10-b) Independence of the knowledge of: 1) The input disturbance covariance Σ_w, 2) the input space-varying parameter β(x), and 3) the output gain h.

10-c) Suitability for on-line identification.

10-d) Under no noise condition (i.e., d=0) the algorithm (SA-1) becomes independent of the knowledge of the parameters.
\{ \xi_m(x) ; \ m = 0, \ldots, N \} and a linear combination of them can also be identified, as remarked on pages 115 and 121.

11) **CONVERGENCE SPEED:** The examples confirmed that the identification procedure has a good convergence speed, which can be accelerated still further by changing the sequences \( \lambda(n) \) and \( \mu(n) \) (i.e., by choosing optimal sequences).

12) **SIMILARITIES WITH OTHER METHODS:** Although this seems to be the first attempt to identify distributed systems in a stochastic environment (random inputs and noisy observations) without imposing restrictions on probability distributions, some common points with previous works can be pointed out (for details see section 2.3):

12-a) A slightly similar deterministic version of the parameter explicit technique, that uses the DPS (with constant parameters) reduced to a set of algebraic equations, was applied by Collins and Khatri [2].

12-b) Stochastic approximation algorithms, as a searching scheme for finding estimates previously obtained by minimizing a performance criterion, were used by Zhivoglyadov and Kaipov [3]-[6] and Carpenter, Wozny and Goodson [7]. They considered noisy observations, but not random inputs.

12-c) In [8], [9] Tzafestas considered random inputs but assumed perfect observation of the state.

12-d) Balakrishnan [10] presented a rigorous theoretical formulation for a particular DPS identification problem in a Gaussian stochastic environment.
Suggestions for further research: Some areas in the DFS identification field where further work seems needed have already been commented on section 2.4. Here we extend these remarks, regarding mainly the method developed in chapter 4, by suggesting the following topics for further research:

1) Other techniques for reduction to a finite-dimensional state space (stage I), such as finite element methods or even more elaborated finite-differences, could be investigated towards the applicability of our stochastic approximation approach in stage II.

2) As commented before, the identification procedure can be accelerated by changing the sequences λ(n) and μ(n) appearing in (SA-1) and (SA-2). In this way, optimization studies could be done in order to determine a pair of sequences (λ^∗(n), μ^∗(n)), among those satisfying the conditions required in (T - 4.1), which maximizes the algorithm convergence speed.

3) More research is also needed regarding optimal placement of a fixed number of observation points.

4) In proposition (P - 4.1) we presented a basic observation equation. A similar relation, also involving the state, could be formulated as follows:

\[ z_k(n) = h(a_k, u_k(n)) + \nu_k(n) \]

where

\[ u_k(n) = (u_{k-N}(n), ..., u_{k+M}(n)) : \text{ a random vector in } \mathbb{R}^{M+1}. \]

In this way we get similar algorithms as in (SA-1) and (SA-2), with \( z_k(n+N_t) \) replaced by \( u_k(n+N_t) \) and \( \xi_k = 0 \). This approach presents three
Suggestions for further research: Some areas in the DFS identification field where further work seems needed have already been commented on section 2.4. Here we extend those remarks, regarding mainly the method developed in chapter 4, by suggesting the following topics for further research:

1) Other techniques for reduction to a finite-dimensional state space (stage I), such as finite element methods or even more elaborated finite-differences, could be investigated towards the applicability of our stochastic approximation approach in stage II.

2) As commented before, the identification procedure can be accelerated by changing the sequences $\lambda(n)$ and $\mu(n)$ appearing in (SA-1) and (SA-2). In this way, optimization studies could be done in order to determine a pair of sequences $(\lambda^*(n), \mu^*(n))$, among those satisfying the conditions required in (T - 4.1), which maximizes the algorithm convergence speed.

3) More research is also needed regarding optimal placement of a fixed number of observation points.

4) In proposition (P - 4.1) we presented a basic observation equation. A similar relation, also involving the state, could be formulated as follows:

$$z_k(n) = h(z(n) ; w_k(n)) + \nu_k(n)$$

where

$$w_k(n) = (u_{k-H}(n), ..., u_{k+H}(n)) ; \text{ a random vector in } \mathbb{R}^{M+1}.$$ 

In this way we get similar algorithms as in (SA-1) and (SA-2), with $z_k(n+N_t^*)$ replaced by $w_k(n+N_t^*)$ and $\xi_k=0$. This approach presents three
basic advantages:

1. The "a priori" knowledge of both noise statistics and the parameters \( \gamma_m(x); m = 0, \ldots, M \) would no longer be necessary.

2. Space-varying observation noise (i.e., \( v_k(n) \)) could be considered.

3. The "indispensable" (in the former approach) information supplied by finite-difference techniques (i.e., \( \sum_{m=1}^{M+1} a_k^m \) constant. See equation (11) in chapter 4 and also remark 3 on page 115) would not be required in this case, and so a wider class of approximation techniques could be applied in stage I.

On the other hand, the accessibility of the state \( u_k(n) \) must be assumed, which represents the main disadvantage. In order to by-pass the state accessibility requirement, the identification could be done by using in the new version of (SA-1) and (SA-2) estimates \( \hat{u}_k(n) \), instead of \( u_k(n) \), obtained by means of the Kalman-Bucy Filter. This approach has been previously considered for LPS identification in [11].

5) It would be useful to have some comparison of effectiveness of the different approaches for the DPS identification problem. In performing such comparisons, one must keep in mind that: The literature in this field considers a wide range of particular models operating in quite diverse conditions. So a critical evaluation involving a large number of methods could become a difficult task.
REFERENCES

CHAPTER 1


CHAPTER 2


[36] CANNON J. R. - "Determination of Unknown Coefficients \( k(u) \) in the Equation \( \nabla \cdot k(u) \nabla (u) = 0 \) from Overspecified Boundary Data", J. of Math. Anal. & Appl., Vol. 18, No. 1, pp. 112-114, Apr. 1967.


* These references came to the author's knowledge later when the chapter was fully typed, so they do not follow the previous order.


183


CHAPTER 4


CHAPTER 5


