Processidentifiering - overheadbilder

Gustavsson, Ivar

1979

Document Version:
Förlagets slutgiltiga version

Link to publication

Citation for published version (APA):
Gustavsson, I. (1979). *Processidentifiering - overheadbilder*. (Technical Reports TFRT-7166). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:
1

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

CODEN: LUTFD2/(TFRT-7166)/1-248/(1979)

PROCESSIDENTIFIERING - OVERHEADBILDER

I Gustavsson

Department of Automatic Control
Lund Institute of Technology
August 1979

PROCESSIDENTIFIERING - OVERHEADBILDER

I Gustavsson

Detta arbete har till stor del understötts av Styrelsen för Teknisk Utveckling under anslag nr 78-3763.

Dokumentutgivare
04T0 Lund Institute of Technology
Handläggare Department of Automatic Control
06T0
Författare
08T0 I Gustavsson

Dokumentnamn
04T6 REPORT
Utgivningsdatum
06T4 August 1979

Dokumentbeteckning
04T6 LUTFD2/(TFRT-7166)/1-248/(1979)
Ärendebeteckning
06T6

10T4

Dokumenttitel och undertitel
18T0
PROCESSIDENTIFIERING - OVERHEADBILDER
(PROCESS IDENTIFICATION - TRANSPARENCIES)

Referat (sammandrag)
26T0 This report consists of the gathered transparencies from a course in Process Identification, which was held twice during the spring of 1979 at the Department of Automatic Control, Lund Institute of Technology.

Referat skrivet av
42T0 I Gustavsson

Förslag till ytterligare nyckelord
44T0

Klassifikationssystem och -klass(er)
50T0

Indextermer (ange källa)
52T0

Omfång
56T0 248 pages

Övriga bibliografiska uppgifter
56T2

Språk
58T0 Swedish

Sekretessuppgifter
60T0

ISSN
60T4

ISBN
60T6

Dokumentet kan erhållas från
62T0 Department of Automatic Control
Lund Institute of Technology
Box 725, S-220 07 LUND 7, SWEDEN

Mottagarens uppgifter
62T4

Pris
66T0

1. INLEDNING

Denna rapport innehåller de samlade overheadbilderna från den kurs i Processidentifiering, som gavs vid Institutionen för reglerteknik, Lunds Tekniska Högskola vid två tillfällen våren 1979.

Syftet med denna kurs var att informera om hur man effektivt utför processidentifiering, dvs bestämmer dynamiska processmodeller ur experimentella data. Avsikten var också att göra kursdeltagarna så förtrogna med ett interaktivt programpaket för identifiering, IDPAC, att de själva efter kursen skulle kunna lösa enklare identifieringsproblem med hjälp av IDPAC. I kursen presenterades ett flertal identifieringsmetoder. En översikt gavs angående de praktiska problem som är förenade med processidentifiering. Bakgrunden till och användningen av IDPAC behandlades. Några industriella tillämpningar presenterades. En stor del av kurstiden upptogs av kursdeltagarnas egna datorkörningar med IDPAC.

Denna samling overheadbilder måste ses i sitt sammanhang med den just beskrivna kursen. Avsikten med overheadbilder är att de skall åtföljas av en muntlig framställning. Denna rapport har framtagits för att användas vid framtida upprepningar av kursen i fråga. Däremot kan det vara svårt för någon som ej har någon kännedom om identifiering att få någon behållning av enbart denna samling overheadbilder. För den som redan vet en del om identifiering kan rapporten emellertid ge värdefulla tips om viktiga begrepp och problem och ge inspiration till ytterligare studier i bland annat de referenser som ges.

Före varje avsnitt finns en kortare innehållsförteckning och i förekommande fall referenser. Mera allmänna referenser är de två böckerna:

Eykhoff P: System Identification - A Survey. Wiley 1974.

Goodwin G C and R L Payne: Dynamic System Identification - Experiment Design and Data Analysis. Academic Press 1977.

samt artikeln:

Åström K J and P Eykhoff: System Identification - A Survey. Automatica 7, 123-162, 1971.

1:2

Referenser för tillämpningarna som presenterades är:

Åström K J and C G Källström: Identification of Ship Steering Dynamics. Automatica 12, 9-22, 1976.

Källström C G and K J Åström: Experiences of System Identification Applied to Ship Steering Dynamics. Preprints 5th IFAC Symposium on Identification and System Parameter Estimation, Darmstadt, 1979.

Ekström L, R Hänsel, L H Jensen and L Ljung: A Dynamic Model of a Part of an Air-conditioned Building. Report 3073, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1974.

INNEHÅLLSFÖRTECKNING

2. IDENTIFIERING - EN ÖVERSIKT

K J ÅSTRÖM

SYSTEM MODELING AND IDENTIFICATION

1. INTRODUCTION
2. MODELS & MODELING
3. MODELING FROM PHYSICS
4. MODELING FROM PROCESS MEASUREMENTS
5. IDENTIFICATION
6. INTERACTIVE COMPUTING
7. TUNING & ADAPTATION
8. CONCLUSIONS

MODELING

WHY USE A MODEL?

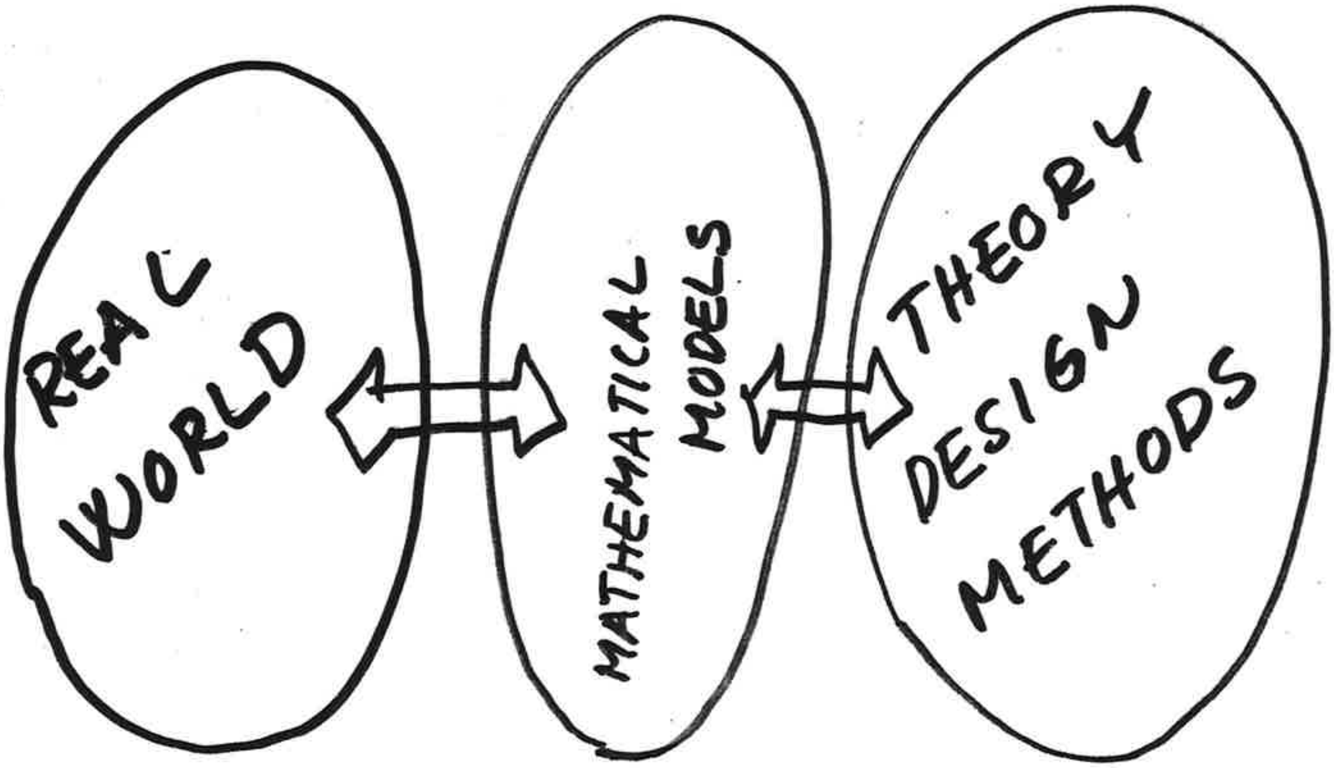
- O COMPACT SUMMARY OF KNOWLEDGE (NEWTON & KEPPLER VS TYCHO BRAHE)
  - COMMUNICATION
  - EDUCATION
- O EASIER TO WORK WITH MODELS THAN REAL LIFE
  - DESIGN
  - OPTIMIZATION
- O SOMETIMES A NECESSITY
  NO ALTERNATIVE AVAILABLE

CAUTION!

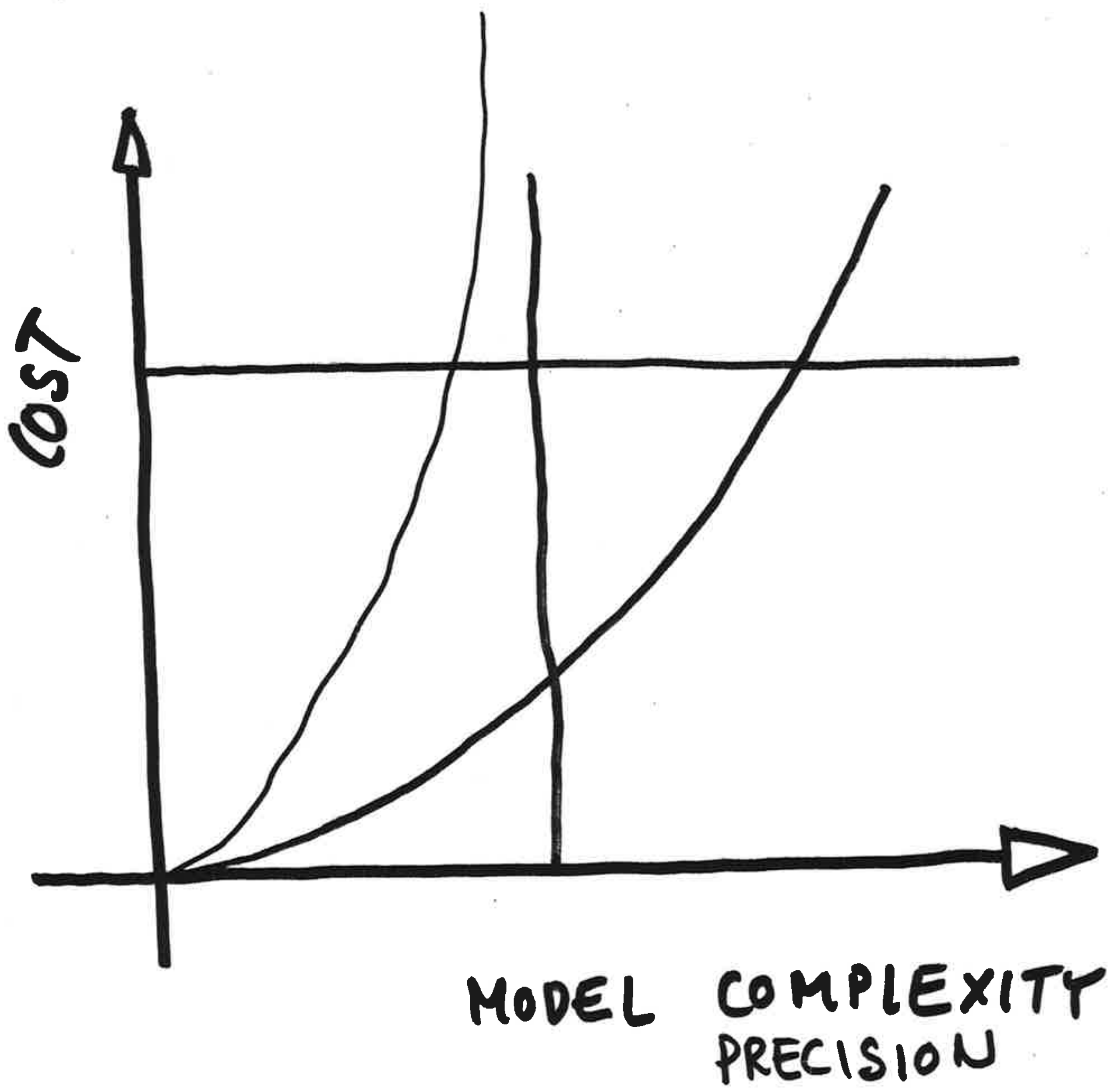
"WHEN MAP DISAGREES WITH NATURE TRUST NATURE"
SWEDISH ARMY MANUAL

EMPIRICS

MODELING IS OFTEN MORE TIMECONSUMING THAN INITIALLY ANTICIPATED



REAL WORLD
MATHEMATICAL MODELS
THEORY DESIGN METHODS



COST
MODEL COMPLEXITY
PRECISION

MODELING

PHILOSOPHY: HIERARCHY OF MODELS

WHY MODELS?

- COMPACT SUMMARY OF KNOWLEDGE
  (NEWTON VS KEPPLER & TYCHO BRAHE)
- EDUCATION
- COMMUNICATION
- PROCESS REDESIGN
- SELECTION OF SENSORS (STRUCTURE
- CONTROL DESIGN
- TEST OF CONTROL SYSTEMS

MODELING OF DISTURBANCES

LEARN FROM THE PROCESS

- PHYSICAL LAWS
- EXPERIMENTS
- IDENTIFICATION

SPECIFICS FOR CONTROL

GOAL CLEAR?

PROCESS DESIGN

REGULATOR STRUCTURE
SENSORS & ACTUATORS

REGULATOR DESIGN

- KNOWLEDGE ABOUT DYNAMICS MAY GIVE A POSSIBILITY TO RESOLVE DIFFICULT DESIGN COMPROMIZES (AIRPLANE FBW!)

- MOVE CONTROL DESIGN CLOSER TO PROCESS DESIGN

- KNOWLEDGE ABOUT CONTROL & DYNAMIC PROBLEMS MAY GIVE A POSSIBILITY TO AVOID PROBLEMS THROUGH GOOD PROCESS DESIGN

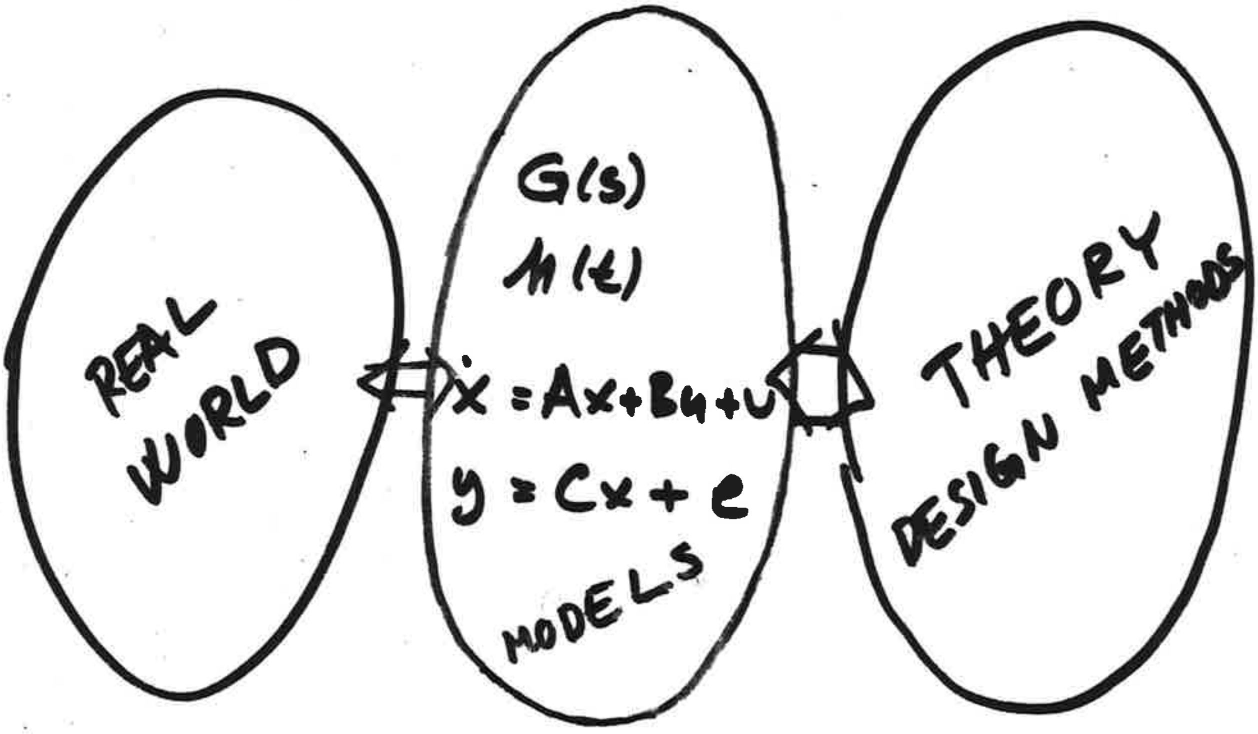
PROCESS DIAGNOSTICS

MODELS REFLECT MUCH OF SYSTEM BEHAVIOUR?

- LOG DATA DURING NORMAL OP.
- FIT MODELS
- EVALUATE PERFORMANCE

HAS BEEN USED TO EVALUATE PACKAGED SYSTEMS IN PAPER INDUSTRY

INTRODUCTION



FREQUENCY RESPONSE

PULSE TESTING

LEAST SQUARES

MAXIMUM LIKELIHOOD

COMPUTERS

TRADE SIMPLICITY IN EXPERIMENTS FOR COMPUTATIONS?

MODELING BASED ON PHYSICAL PRINCIPLES

- SPECIFY PURPOSE OF MODEL

  DEFINE SYSTEM BOUNDARIES

  INPUTS, OUTPUTS, DISTURBANCES

  ⇒ QUALITATIVE MODEL

- WRITE BALANCE EQUATIONS

  MASS
  MOMENTUM
  ENERGY

  VARIABLES REQUIRED TO DESCRIBE STORAGE OF THESE ARE CALLED STATE VARIABLES.

- WRITE CONSTITUTIVE EQUATIONS

  HOOKES
  ARRHENIUS
  THERMODYNAMICAL STATE EQ.

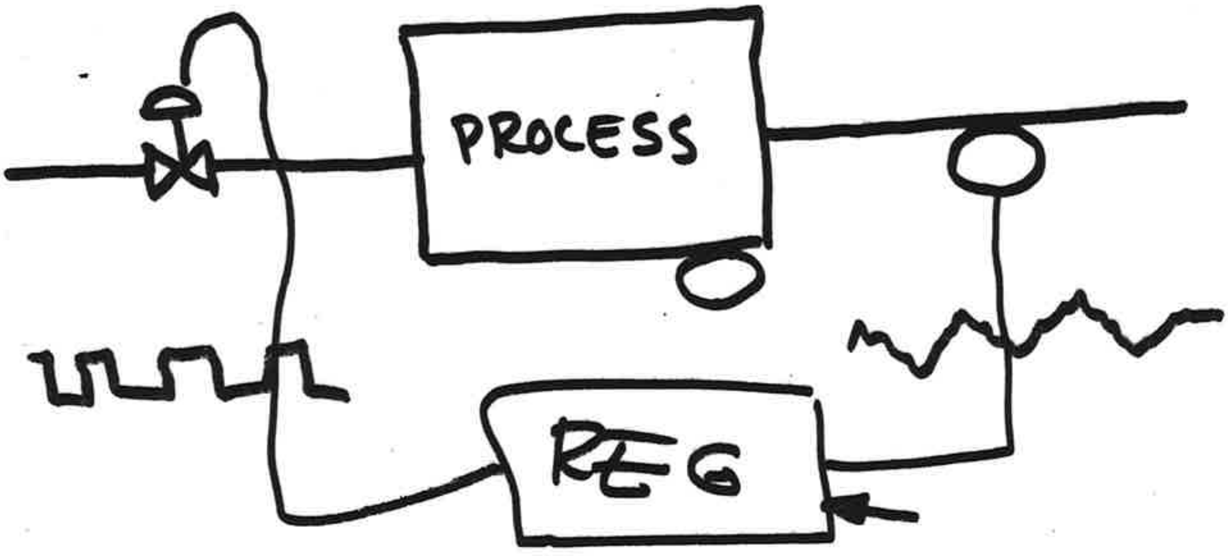
  ⇒ QUANTITATIVE MODEL

- VALIDATE !

  DIFFICULTY

  APPROX, TIME, COST

MODELING BY PROCESS EXPERIMENTS



- EXPERIMENTAL PLANNING
- CHOOSE MODEL STRUCTURE
- PARAMETER ESTIMATION
- VALIDATION

MODELSTRUCTURES $\mathcal{M}$

"BLACK, GREY & WHITE BOXES"

EXTERNAL DESCRIPTIONS:

$$Y(s) = G_\theta(s)\,U(s) + H_\theta(s)\,E(s)$$

$$y(t) + A_1 y(t-1) + \dots + A_n y(t-n) =$$
$$B_1 u(t-1) + \dots + B_n u(t-n) +$$
$$e(t) + C_1 e(t-1) + \dots + C_n e(t-n)$$

INTERNAL DESCRIPTIONS:

$$\frac{dx}{dt} = f(x, u, v, \theta)$$
$$y = g(x, u, e, \theta)$$

$$\frac{dx}{dt} = Ax + Bu + v$$
$$y = Cx + Du + e$$

IDENTIFICATION

- $\mathcal{B}$ EXPERIMENTAL PLANNING
- $\mathcal{B}$ MODEL STRUCTURE
- $\mathcal{B}$ PARAMETER ESTIMATION
- $\mathcal{B}$ VALIDATION

$\mathcal{D}$ INPUT/OUTPUT DATA FROM SYSTEM $\mathcal{S}$ UNDER EXPERIMENT CONDITION $\mathcal{X}$

$\mathcal{M}$ CLASS OF MODELS

$\mathcal{C}$ CRITERION

METHODS
ALGORITHMS
THEORY
EMPIRICS

EX: TRADE $\mathcal{X}$ FOR COMPUTATIONS

THEORY

- STANDARD STATISTICAL THEORY

  $\mathcal{S} \in \mathcal{M}$ DOES $\hat{\mathcal{M}}(\mathcal{D}, \mathcal{X}, \mathcal{C}) \rightarrow \mathcal{S}$

  AS LENGTH OF $\mathcal{Z} \rightarrow \infty$?

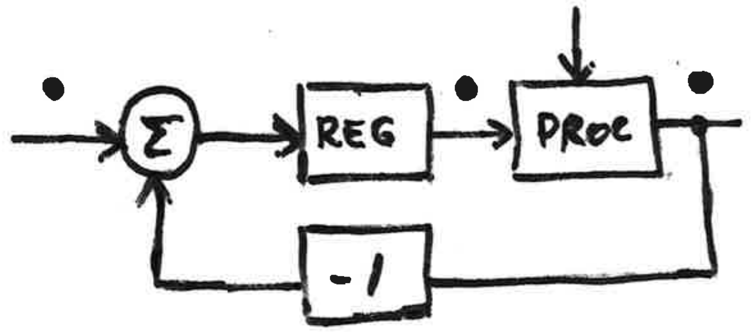
SYSTEM (PARAMETER) IDENTIFIABILITY

CONSISTENCY

ACCURACY

- INFLUENCE OF $\mathcal{X}$

EXPERIMENTS WITH FEEDBACK



SELECTION OF INPUT SIGNALS

- PARAMETRIZATION OF MODELS

UNDERSTANDING, INSIGHT

SCREENING OF MODELS

PROBLEM $\mathcal{S} \notin \mathcal{M}$!

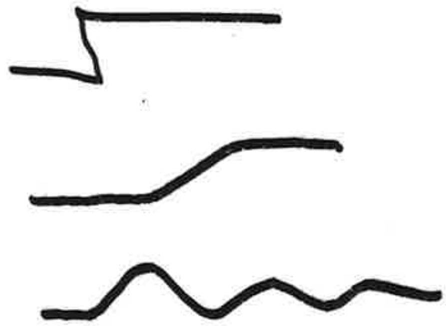
CHECK LIST FOR MODELING

PROCESS DESCRIPTION

- PICTURES
- MIKEY MOUSE
- FLOW DIAGRAMS
- (BE AWARE OF STANDARDS IN DIFFERENT FIELDS!)

MATHEMATICAL MODELS

- PURPOSE
- ASSUMPTIONS
- EQUATIONS
- NORMALIZATION, DIMENSIONFREE PARAMETERS
- NOTICE SI STANDARD FOR UNITS AND SYMBOLS
- RANGE OF VALIDITY
- INPUTS OUTPUTS STATES
- PARAMETERS



STEADY STATE PROPERTIES

- STEADY STATE GAINS
- RELATIONS TO PHYSICAL PARAMETERS

NONLINEAR SIMULATION

- SIMULATION PROGRAMS
- (COMPUTER PROGRAMS IS THE ONLY WAY TO DESCRIBE AND HANDLE MODELS FOR COMPLEX SYSTEMS)
- TRANSIENT RESPONSE

LINEARIZATION

- EQUATIONS
- NORMALIZATIONS, DIMENSIONFREE PARAMETERS
- RANGE OF VALIDITY
- TRANSFER FUNCTIONS
- TRANSIENT RESPONSES
- TIME CONSTANTS, GAINS, BODEDIAGRAMS
- RELATIONS TO PHYSICAL PARAMETERS

APPROXIMATIONS

- BASED ON PHYSICAL ARGUMENTS
- SYSTEM THEORETIC

VALIDATION

- EXPERIMENTAL DATA
- HOW FAR CAN WE TRUST THE MODEL?

REFERENCES

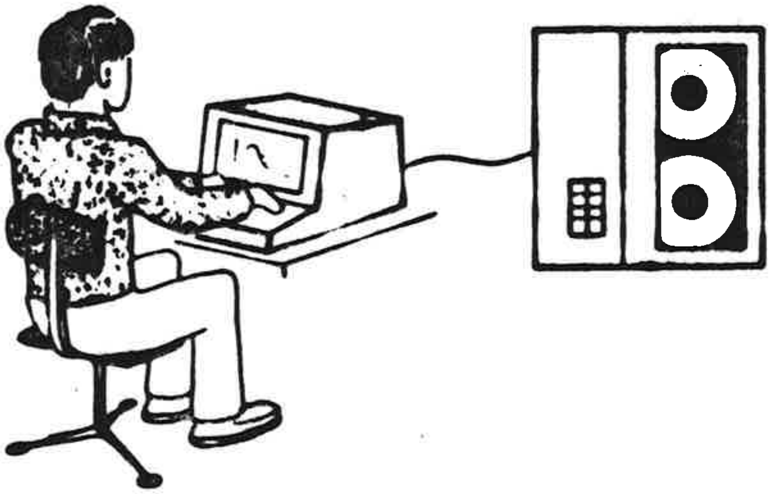
COMPUTER AIDED ANALYSIS AND DESIGN

BACKGROUND

MANY METHODS ARE CONCEPTUALLY SIMPLE
BUT THEIR DETAILS MAY BE MESSY

SOLUTION

COMBINE MAN'S INTUITION WITH THE COMPUTERS
CALCULATING CAPACITY



EXAMPLES

SIMNON
IDPAC
MODPAC
SYNPAC

EMPIRICS

- MANY METHODS WORK WELL ON SIMULATED DATA BUT NOT ON PLANT DATA ($\mathcal{S}$ & $\mathcal{M}$)!
- ML + LS A REASONABLE COMBINATION
- INTERACTIVE COMPUTING!!!

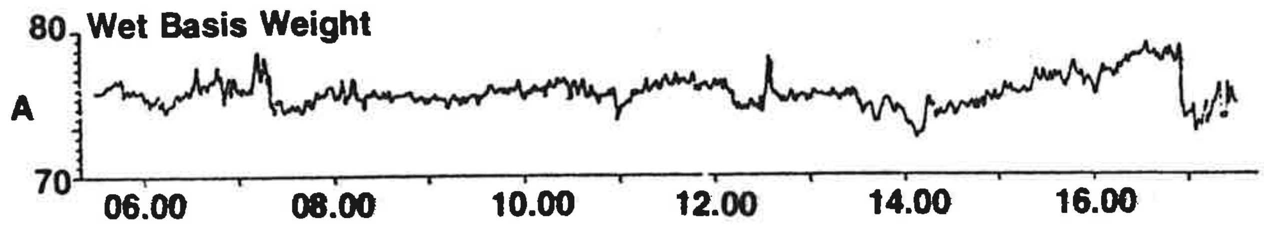
```
MOVE DK WORK ← DT DATA (1 2)
PLOT WORK
TREND WORK(2) ← WORK(2) 1
ML PAR1 ← WORK 1
ML PAR2 ← WORK 2
ML PAR3 ← WORK 3
RESID RES ← PAR2 WRK 20
DETER DET ← PAR2 WRK(1)
PLOT NL WRK(2) DET
```

APPLICATIONS OVERVIEW

- DRUM BOILER
- NUCLEAR REACTOR
- ELECTRIC GENERATOR
- DISTILLATION COLOMN
- EVAPORATORS
- MIXER SETTLER
- PAPER MACHINES
- DIGESTERS
- HEATING & VENTILATION
- HEAT EXCHANGERS
- HEAT ROD
- SHIP STEERING
- ACTIVATED SLUDGE PROCESS
- BIOLOGICAL SYSTEMS
  - BALANCING
  - WORKLOAD - HEART RATE

ASSESSMENT OF BENEFITS OF CONTROL

DATALOGGING:

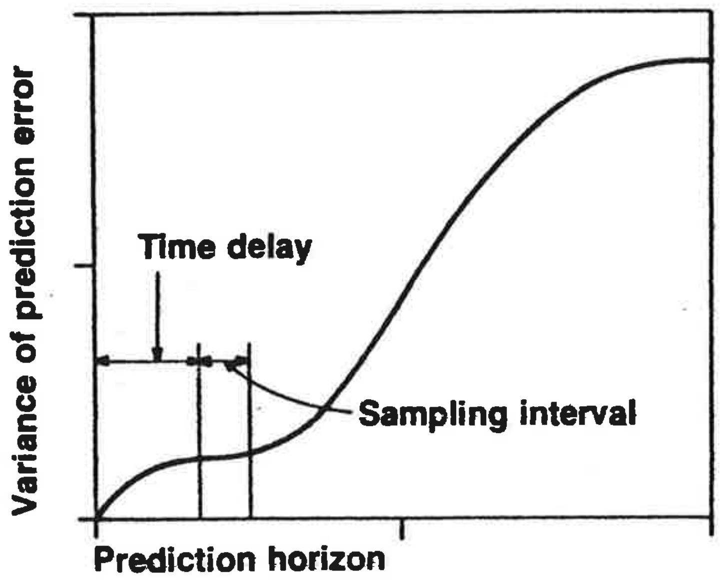


PROCESS IDENTIFICATION: PROCESS MODEL

PREDICTION

ERROR ANALYSIS

HEDGE

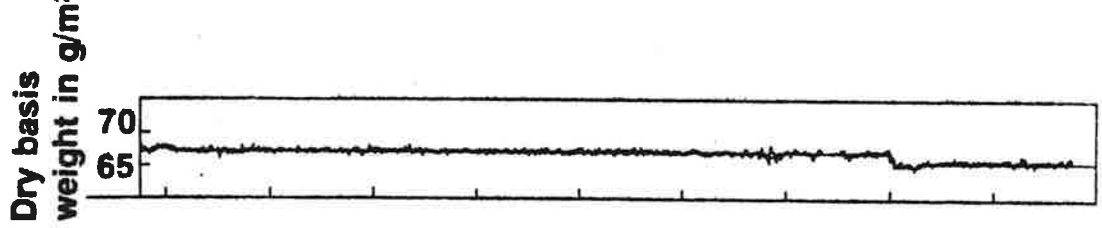


2:/7

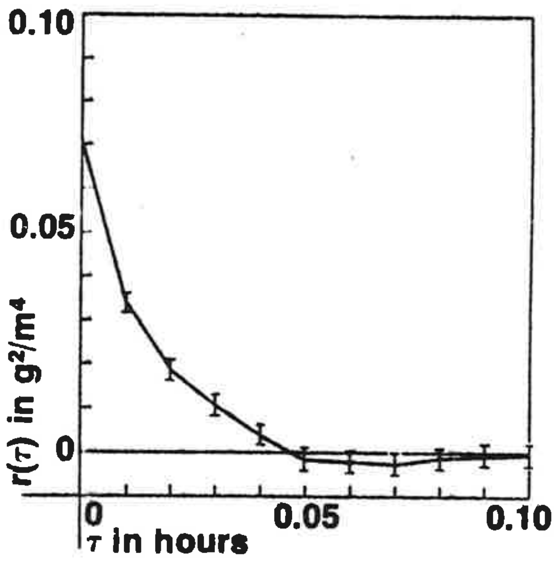
2:18

ASSESSMENT OF NEED FOR RETUNING OF MINIMUM VARIANCE CONTROL

LOG CONTROLLED OUTPUT DURING NORMAL OPERATION

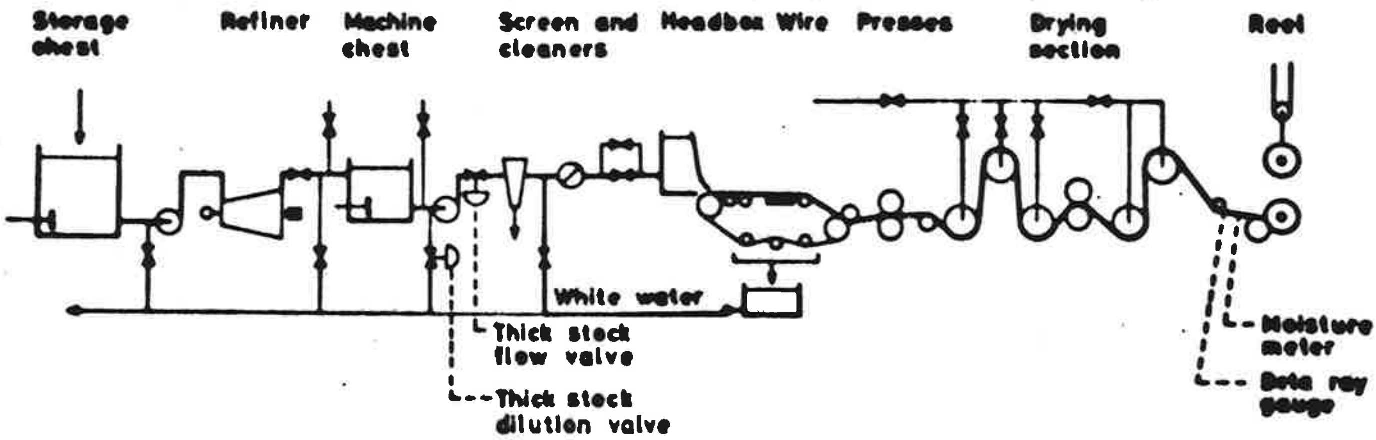


CALCULATE COVARIANCE FUNCTION OF OUTPUT (COV Y)



USE KNOWLEDGE OF PROCESS DEAD TIME AND SAMPLING PERIOD TO TEST IF COVARIANCE FUNCTION SATISFIES MINIMUM VARIANCE CONDITIONS

BASIS WEIGHT CONTROL OF PAPERMACHINE



SECOND ORDER MODEL

TWO TIME DELAYS

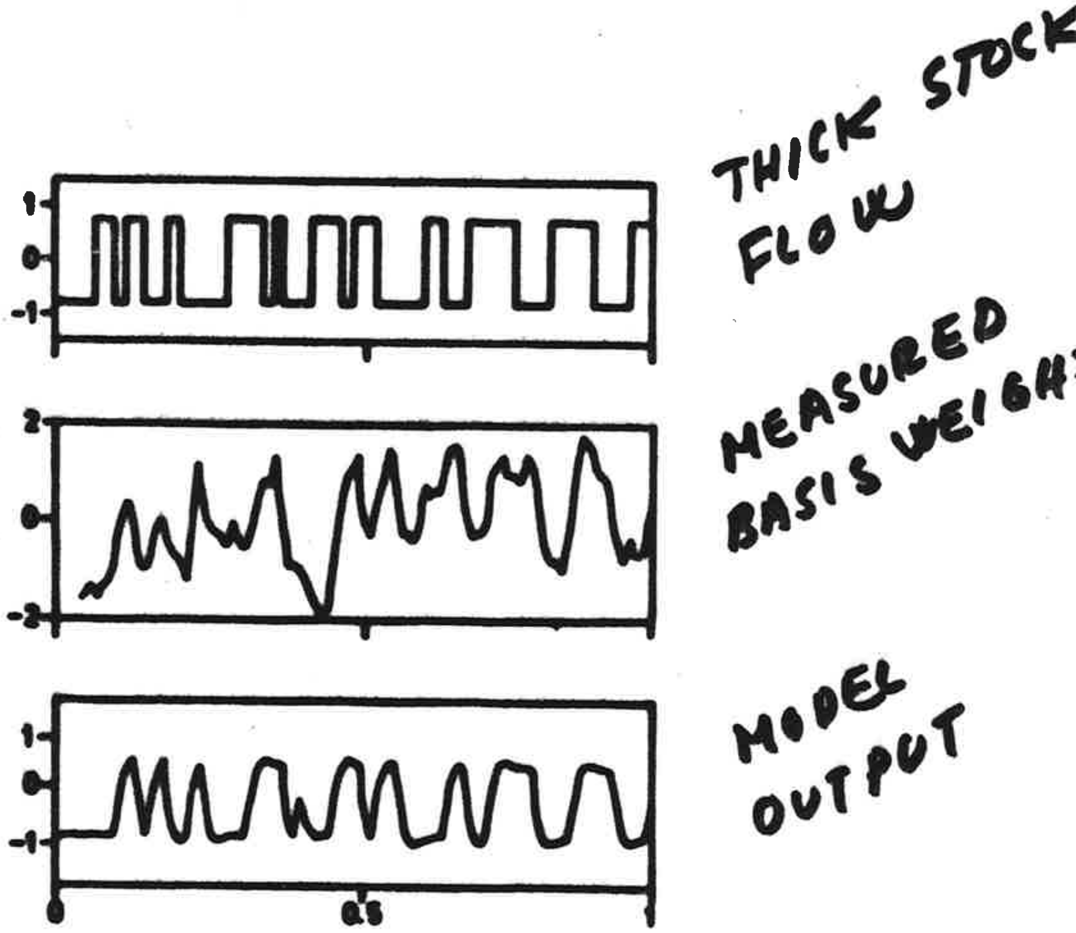
SEVEN PARAMETERS

$$y(t) - 1.283\,y(t-1) + 0.45\,y(t-2) = 4.6\,u(t-2) - 4.05\,u(t-3)$$

$$\Delta y(t) = \frac{4.61q - 4.05}{q^2 - 1.283q + 0.495}\,\Delta u(t-2) + 0.382\,\frac{q^2 - 1.438q + 0.550}{q^2 - 1.283q + 0.495}\,e(t)$$

REF K. J. Å. INTRODUCTION TO STOCHASTIC CONTROL THEORY

2:20



THICK STOCK FLOW
MEASURED BASIS WEIGHT
MODEL OUTPUT
1
0
-1
2
0
-2
1
0
-1
0
0.5
1

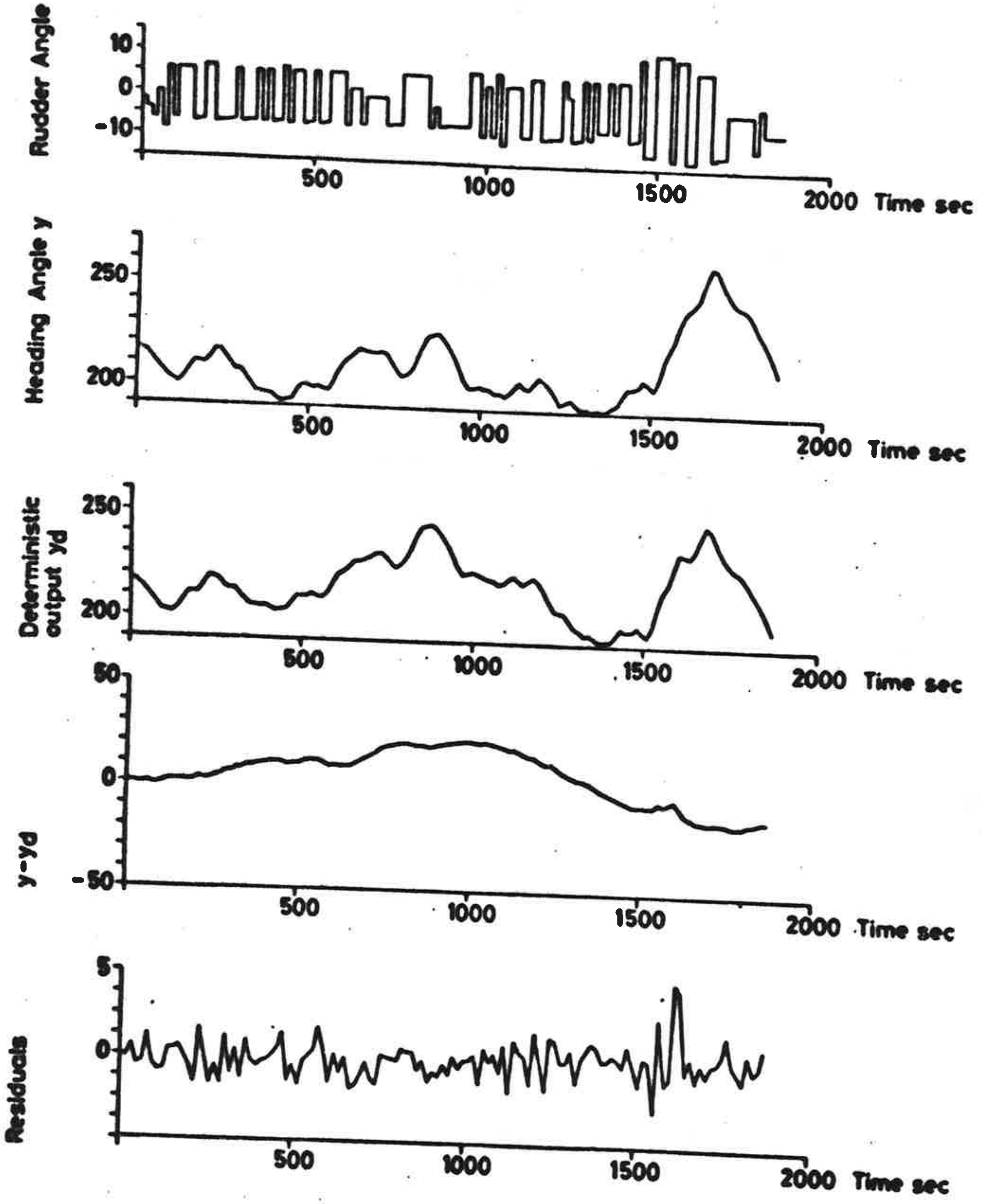
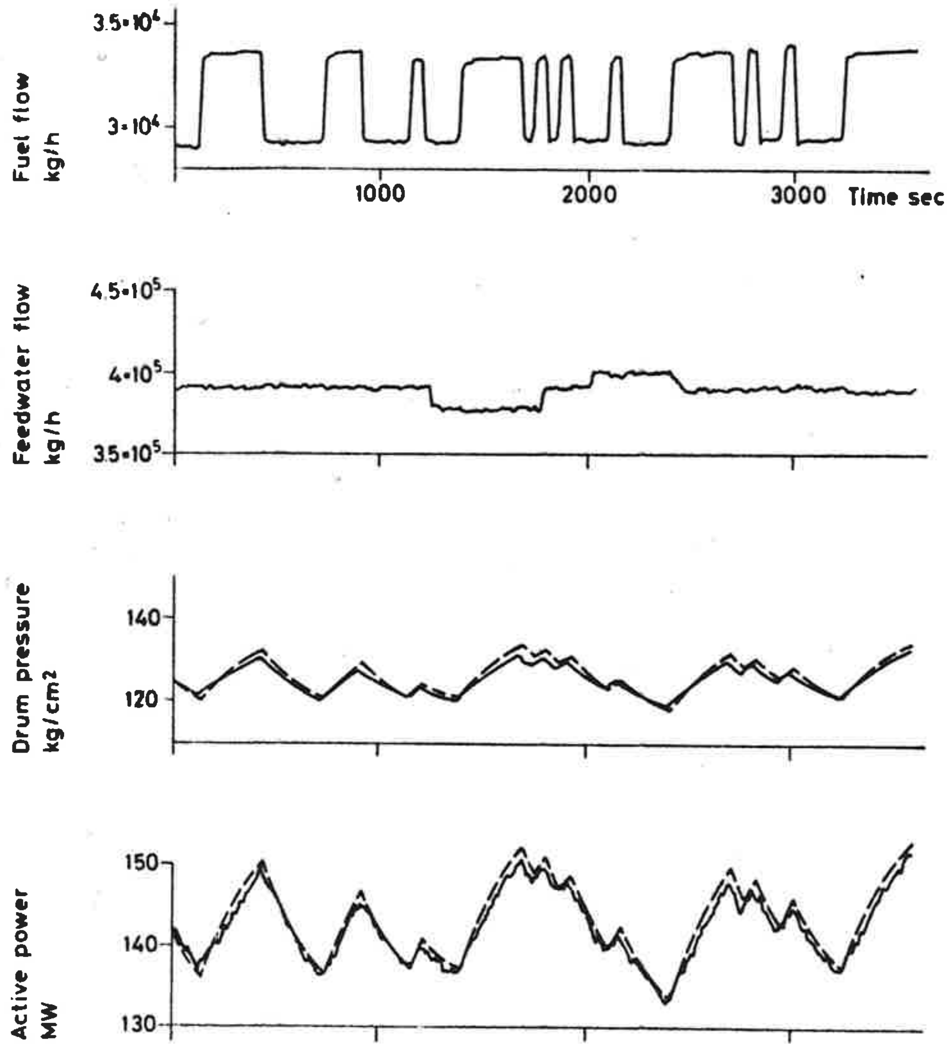
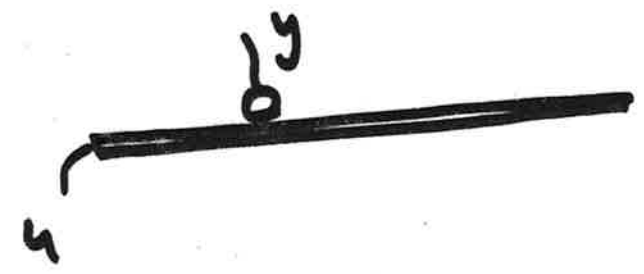


Fig. 3 Results of an experiment designed to determine the ship steering dynamics.

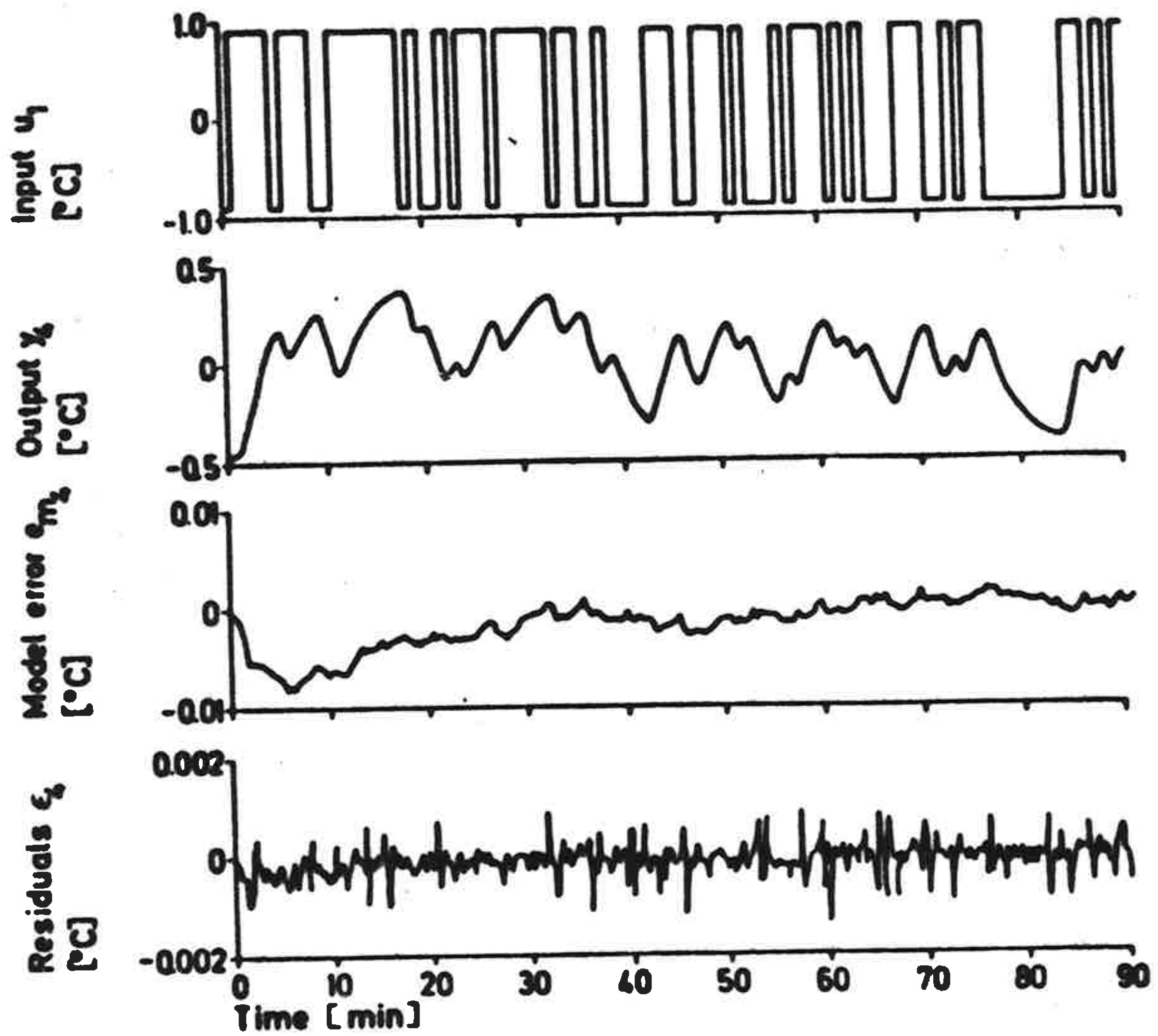
2:22

THERMAL BOILER
KARL EKLUND



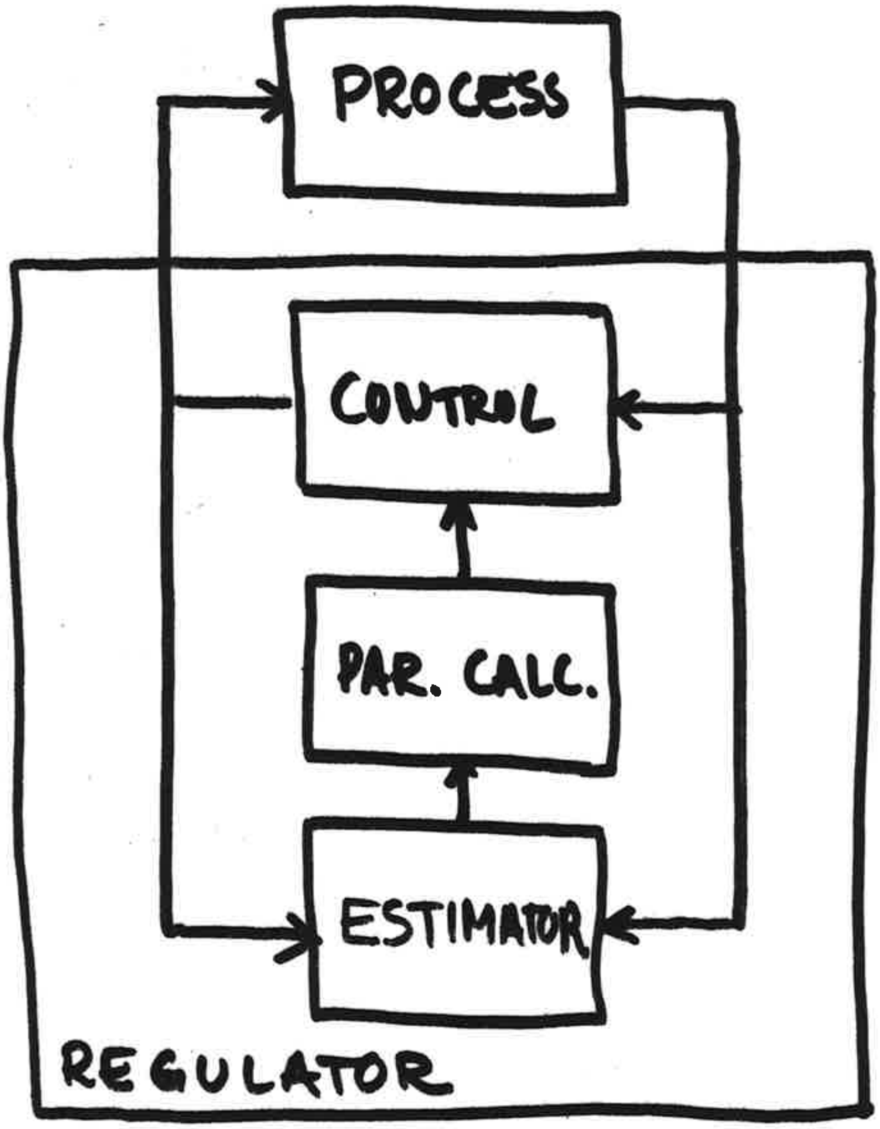


HEAT ROD IDENTIFICATION
BO LEDEN



$$\frac{\partial u}{\partial t} + \theta_1 u = \theta_2 \frac{\partial^2 u}{\partial x^2}$$

TUNING & ADAPTATION

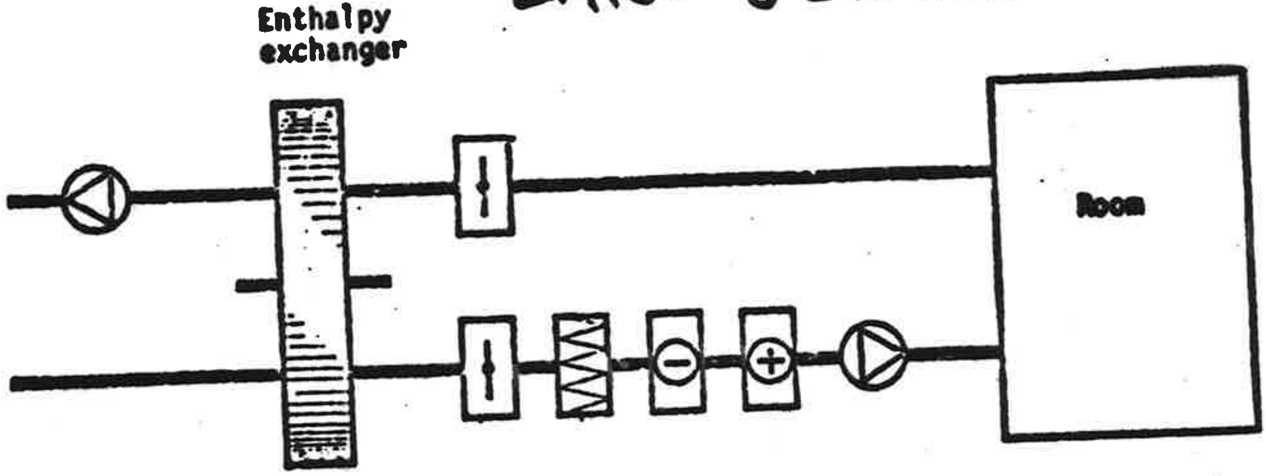


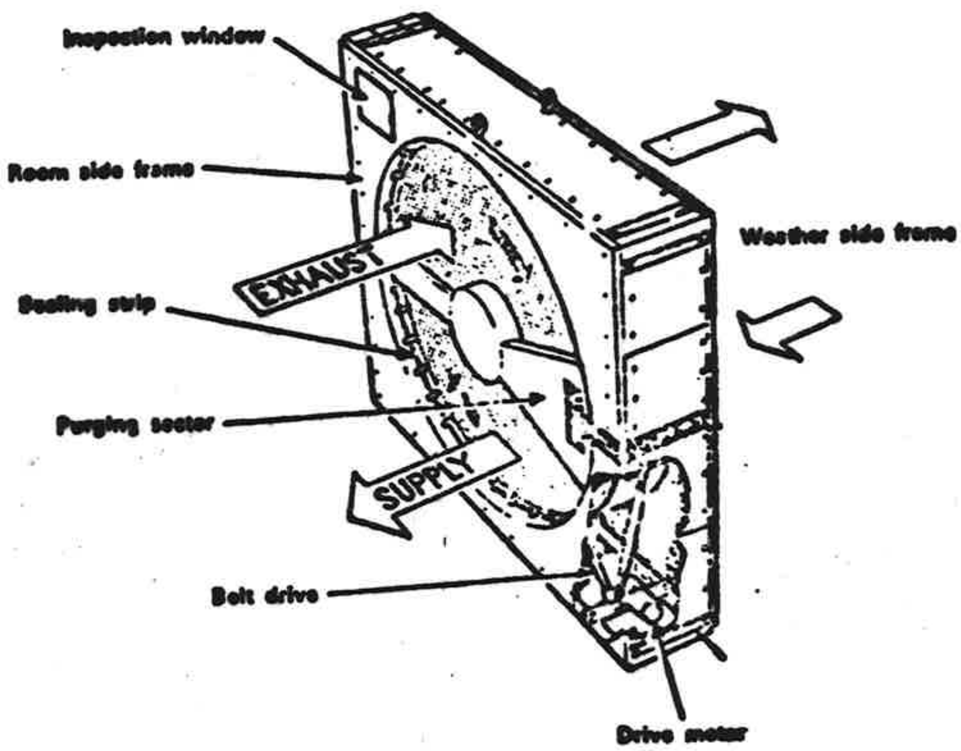
INTERPRETATION

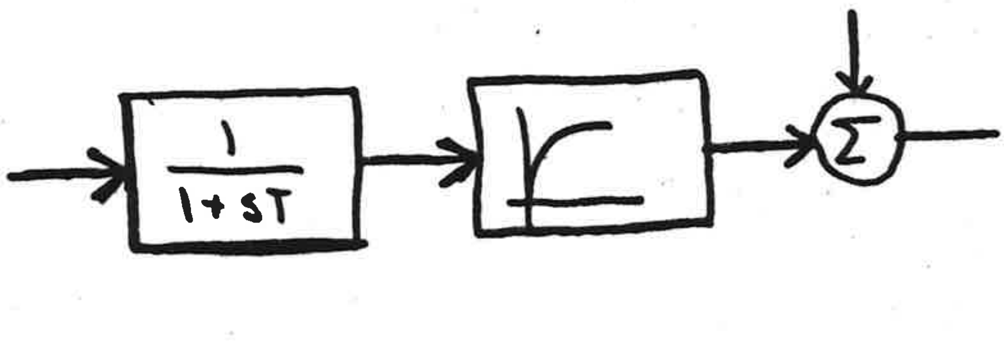
PERIODICAL TUNING

ADAPTATION

ENTHALPY EXCHANGER
LARS JENSEN

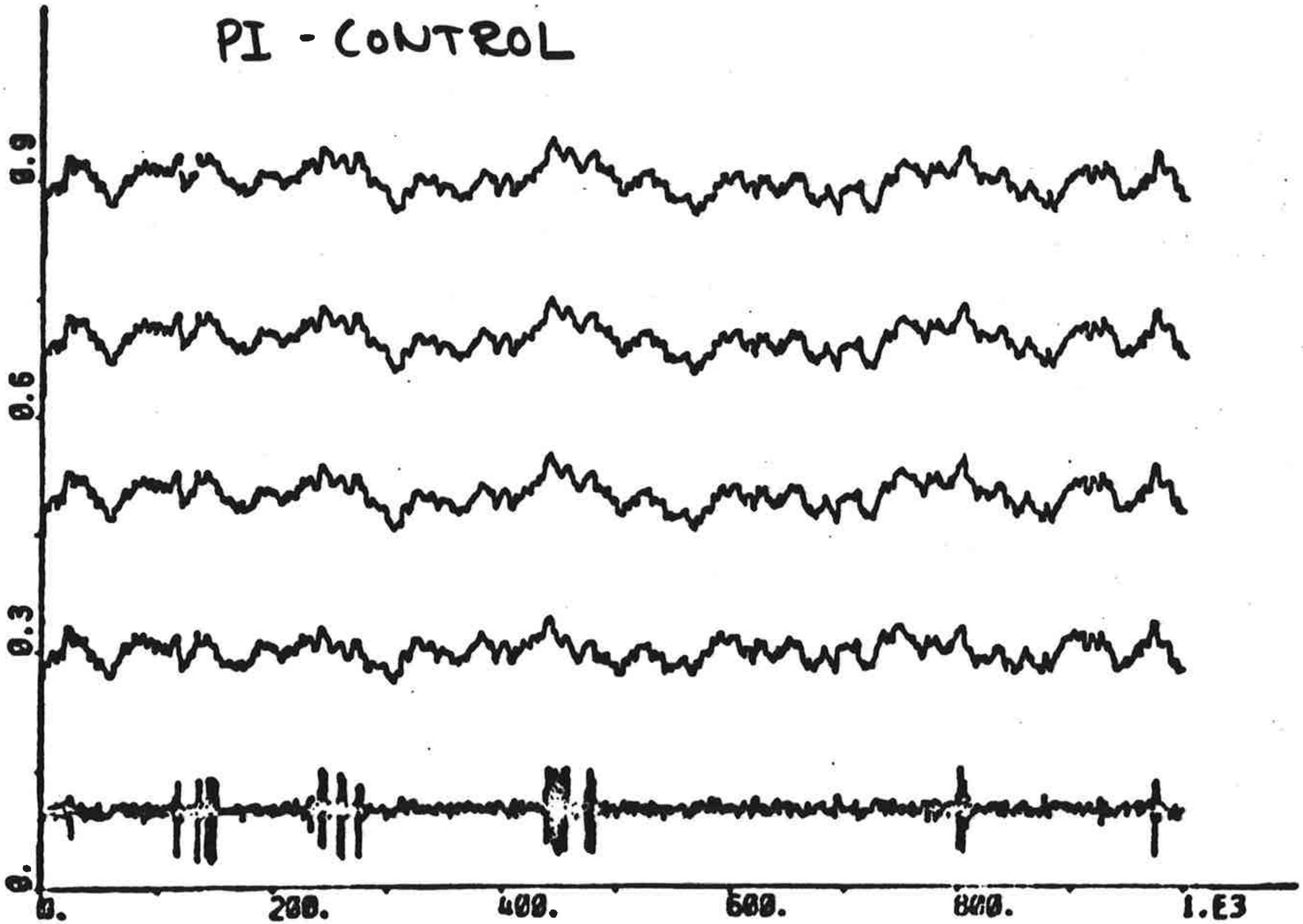


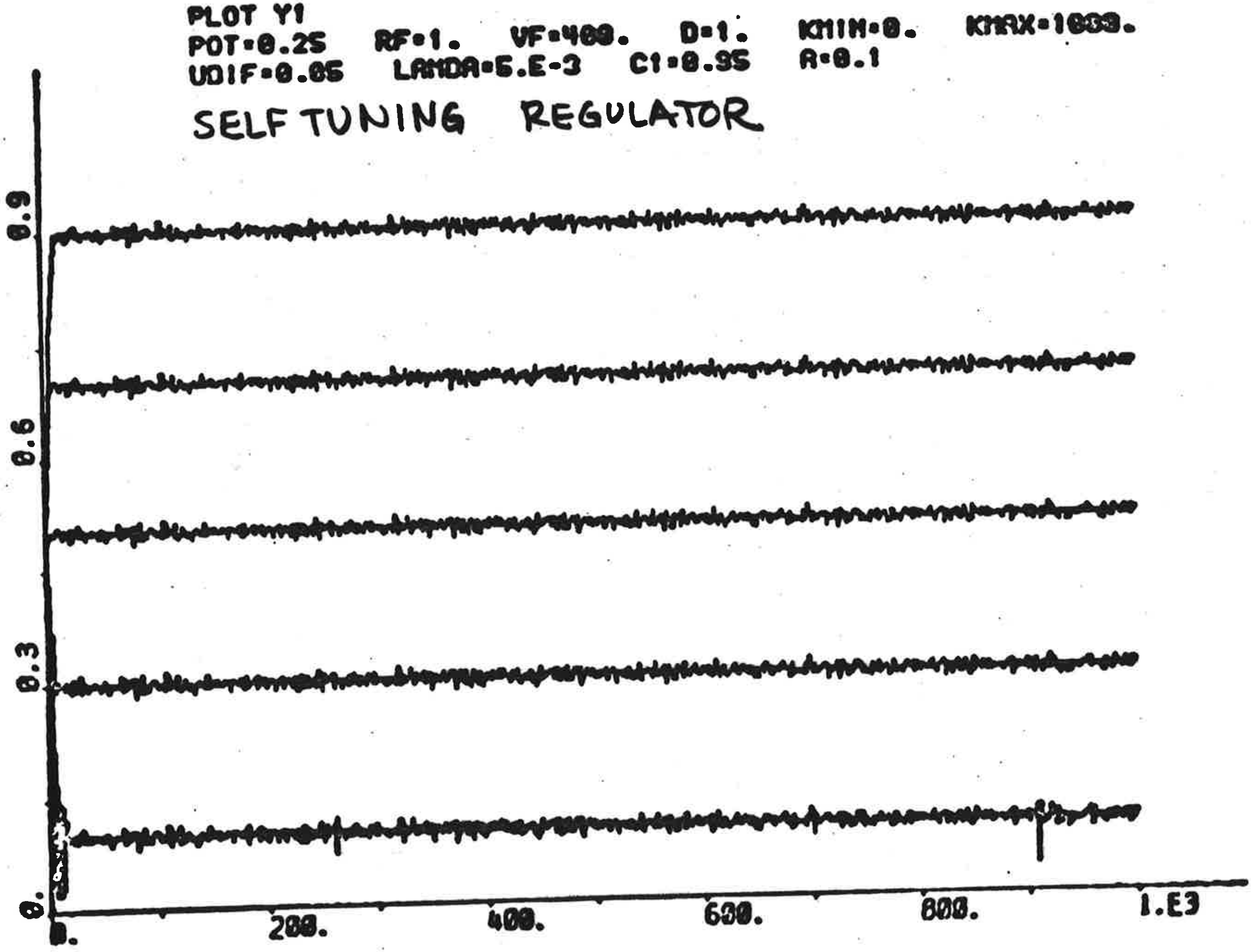




2:96

PLOT Y1
POT=0.25 RF=0. D=4.E-3 KMIN=1. KMAX=1. UDIF=0.03
LAMDA=5.E-3 CI=0.95 A=0.1




PLOT Y1
POT=0.25 RF=1. VF=400. D=1. KMIN=0. KMAX=1000.
UDIF=0.05 LAMDA=5.E-3 C1=0.95 A=0.1
SELF TUNING REGULATOR
0.9
0.6
0.3
0.
0.
200.
400.
600.
800.
1.E3

1:22

CONCLUSIONS

- FAMILY OF MODELS
  MATCH MODELS TO PROBLEMS
- IDENTIFICATION TECHNIQUES USEFUL FOR MODELING: LEARN FROM THE PROCESS
- CHOOSE TECHNIQUE THAT FITS THE PROBLEM
  - PULSE & FREQUENCY RESPONSE
  - CORRELATION & SPECTRAL ANALYSIS
  - PARAMETER ESTIMATION LS, MV
- INTERACTIVE COMPUTING MAKES METHODS EASY & CHEAP TO LEARN AND USE
- PROCESS DIAGNOSTICS
- TUNING & ADAPTATION

The Role of System Identification in Process Modeling

Von Professor K.J.Åström, Lund

1. INTRODUCTION

The notion of a mathematical model is fundamental to science and engineering. A model is a very useful and compact way to summarize our knowledge about a process. A model is also a very effective tool for education and communication. For control engineering, models are significant because virtually all existing control theory is based on the assumption that mathematical models of the process, its environment and the criterion are available. The models are also used to select the structure of the control system, appropriate sensors and actuators. They are also useful for process design.

It is important to emphasize the danger of believing that a process can be characterized by one mathematical model. It is much more fruitful to represent a process with a hierarchy of models, ranging from very detailed and complex simulation models of whole processes to the 'back of an envelope model' which is easily manipulated analytically. The simple models are used for exploratory purposes and to obtain the gross features of the system behaviour. The very complicated simulation models, which also may contain pieces of the real process, are used for a detailed check of the control system to make sure that nothing has been neglected. The complicated models take a long time to develop and they are costly to maintain. They do, however, reproduce the properties of the real system with high fidelity and they are a necessity for design of critical processes. Between the two extremes there may be many different types of models which are suited for design of control systems. The crucial problem is to steer between oversimplification with the danger of disaster and overcomplication which is too expensive. The trademark of good engineering is to choose the right model for each specific purpose.

There are in principle two different sources from which models can be obtained, from prior experiences e.g. in terms of physical laws, or by experimentation on a process. Modeling based on physical knowledge was covered by prof Profos's lecture. The purpose of this lecture is to discuss system identification i.e. modeling based on experimentation on a process. When it is attempted to obtain a specific model it is of course beneficial to combine both approaches.

Classical control theory was based on the idea to model a dynamical system by a transfer function or an impulse response. Such a model is referred to as an external description because it gives only a relation between the system input and output i.e. the external variables. The success of classical control theory can partly be attributed to the fact that there were powerful experimental techniques, frequency and transient response analysis, which made it possible to obtain the appropriate models. These classical methods for system identification are still very useful for process modeling and they should always be kept in mind even if they have largely vanished from most current papers on automatic control.

The so-called modern control theory is largely based on a process model in terms of a state-equation. This is called an internal model because the state model describes explicitly all the internal couplings between the inputs, outputs and the state-variables. The problem of obtaining suitable internal descriptions for different process is one of the major problems when attempting to apply modern control theory. In special areas like the aerospace field it was frequently possible to derive the desired models from basic physical laws. However, in other areas, like industrial process control it was not possible to obtain the desired models from basic physical laws, and process experiments thus became a necessity. Much of the current research in system identification has been inspired by the desire to obtain process models from process experiments.

In control system design it is also important to have models of disturbances. The external models are often given in terms of spectral densities and covariance functions. When using internal representations the disturbances are instead represented as outputs of dynamical systems driven by white noise. Models for disturbances can only, rarely, be determined from first principles. Process experiments combined with system identification is thus often the only possibility to model disturbances.

The purpose of this paper is to give an overview of system identification techniques and their use in process modeling. A brief discussion of system identification methods is given in section 2. Interactive computing is very significant for the practical solution of system identification problems because it helps the problem solver to combine his intuition with extensive calculations. Interactive computing is probably also the only way to make many identification methods easily accessible for engineers working in industry. Interactive computer software for system identification is discussed in section 3. Practical aspects and experiences of applying system identification methods for process modeling are discussed in section 4. Design of adaptive control systems is a particular area of applications. One possibility is to use a controller which includes an on-line parameter estimator. This is discussed in section 5. The main conclusions are given in section 6. It can be safely said that the classical techniques for system identification like frequency and transient response analysis together with techniques like the least squares and the maximum likelihood method provides the process modeler and control designer with powerful tools that are worthwhile to master.

2. SYSTEM IDENTIFICATION

Some aspects on the system identification problem that are useful for the applications are given in this section. For more details we refer to the survey papers [1] and [2], the books [3],[4],[5],[6] and the proceedings from the IFAC Symposia on system identification in Prague 1967, 1970, the Hague 1973 and Tbilisi 1976.

It was mentioned in the introduction that identification was the experimental aspect of process modeling. In particular system identification includes

1. Experimental planning
2. Selection of model structure
3. Parameter estimation
4. Validation

Experimental planning includes the decision to make open- or closed-loop experiments, selection of input signals and sampling rates. It also includes considerations of many of the practical problems that are associated with performing experiments in an industrial environment. The experiment will result in data D in the form of records of inputs and outputs from the process. The selection of model structure is frequently based on physical principles or on a priori knowledge of the process dynamics. The purpose of the parameter estimation is to determine the parameters of the model based on the experimental data. Model validation is the procedures used to ensure that the model obtained is reasonable. This frequently requires more experiments.

In practice the procedure is iterative. When investigating a process where the a priori knowledge is poor it is reasonable to start with transient and frequency response analysis to get crude estimates of the dynamics, the region of linearity, and the disturbances. Based on these results it can then be attempted to derive physical models where the results of the frequency response analysis are used to motivate various approximations. The results of the preliminary investigation can then be used to plan suitable experiments where the plant is perturbed and the output observed. The data obtained are then used to estimate the unknown parameters. New experiments are done for the validation. Based on the results and the experience obtained, the model may be improved and new experiments can be planned etc.

The different aspects of system identification will now be discussed in more detail.

Experimental planning

It is often difficult and costly to perform experiments on real industrial processes. It has therefore been a desire to develope methods that will relax the constraints on the experiments at the expense of increased computations. While many classical methods depended strongly on the input to be of a precise form, e.g. sinusoid, the newer techniques can handle virtually any type of input signal. The only requirements on the input signal is that it should excite all the modes of the process sufficiently (persistant excitation).

There is a substantial literature on the planning statistical experiments [7],[8]. The purpose is to find optimal designs of experiments. In process modeling this corresponds to finding an optimal input signal. Considerable research has been devoted to this problem [9],[10]. All results on optimal input design are, however, based on the assumption that a model of the process is known. This means that the results can only be used when a reasonably good a priori knowledge of the dynamics of the process and its environment is available. Good applications are known. The results may, however, also be strongly misleading if the process dynamics differs from the a priori assumptions. The results on design of optimal inputs are also restricted because it is frequently assumed that the process is open loop during the experiment.

The possibility to base system identification on data obtained under closed loop control of processes have been explored recently [11].The results obtained are very useful from the point of view of applications. The main difficulty with data obtained from a process under feedback is that it may be impossible to determine the desired models i.e. lack of identifiability. It has, however, been demonstrated that identifiability can be recovered if the feedback is sufficiently complex. It helps to make the feedback nonlinear, time-varying and to change the set points. A practical way to make the feedback time-variable is to switch between different linear feedbacks. There are cases where data from closed loop experiments will give better results than open loop experiments [11]. A practical way to arrange the experiments is e.g. to provide the process with a self-tuning regulator to minimize the fluctuations in process variables and to change the set-point of the regulator with as large signal as possible.

Model structures

The model structures used are derived from prior knowledge of the dynamics of the process and its environment.

In some cases the only a priori knowledge available is that the process can be described as a linear system in a particular operating range. It is then natural to use general representations of linear systems. Such representations are often called black box models. Typical examples of black box models are the transfer function model

$$U(s) = G(s)U(s) + H(c)E(s) \quad (2.1)$$

and the difference equation model

$$y(t) + A_1y(t-1) + \ldots + A_ny(t-n) =$$
$$= B_1u(t-1) + \ldots + B_uu(t-n) + e(t) + C_1e(t-1) +$$
$$+ \ldots + C_ue(t-n) \quad (2.2)$$

where u is the input, y the output and e is a white noise disturbance. The parameters as well as the order n in the vector difference equation (2.2) are considered as unknown parameters.

Sometimes it is possible to apply known physical laws to derive models of the process which only contain a few parameters. Such models are commonly referred to as <u>white box models</u>. For lumped parameter processes such models may be of the form

$$\frac{dx(t)}{dt} = f(x(t),\ u(t),\ v(t),\ \theta)$$

$$y(t) = g(x(t),\ u(t),\ e(t),\ \theta) \tag{2.3}$$

where u is the input, y the output, x the state, e and v disturbances and θ a vector of unknown parameters. Linear models where

$$f(x,u,v,\theta) = A(\theta)x + B(\theta)v + v$$

$$g(x,u,e,\theta) = C(\theta)x + D(\theta)u + e \tag{2.4}$$

are particularly common.

For distributed parameter processes the model (2.3) is replaced by a partial differential equation.

In many practical cases the models may be composed of parts which are black box models and parts which are white box models. Such models are called <u>grey box models</u>. Notice that a significant trend in the recent development is to attempt to model both the process dynamics and the disturbances. This is of course in close agreement with the needs of the control engineer because without disturbances there is no control problem.

Criteria

When formulating an identification problem a criterion is introduced to give a quantity expressing how well a model M fits the experimental data $\mathcal{D}$. The criteria can be postulated. By making statistical assumptions it is also possible to derive criteria from probabilistic arguments. Criteria can therefore be viewed from two points of view. They are often expressed as

$$V(e) = \int_0^T h(e(t))dt \tag{2.5}$$

or for discrete time systems

$$V(e) = \sum_{t=0}^{N} h(e(t)) \tag{2.5'}$$

where e is the input error, the output error or the generalized error. See [1]. The prediction error is a typical example of a generalized error. The function h is frequently chosen as a quadratic but it is also possible to have many other forms. Particularly it may be useful to have functions which do not grow as rapidly as e^2 for large e. See [12].

The first formulation, solution and application of an identification problem was given by Gauss in his famous determination of the orbit of the planet Ceres [13]. Gauss formulated the identification problem as an optimization problem and introduced the principle of least squares in the following way:

> "Therefore, that will be the most probable system of values of the unknown quantities p, q, r, s, etc., in which the sum of the squares of the differences between the observed and computed values of the functions V, V', V", etc. is a minimum."

Ever since, the least squares criterion has been used extensively. Nowadays the least squares method (LS) commonly refers to a method where not only the criterion is quadratic but also the model is such that the errors (i.e. the differences between the observed and computed values) are <u>linear in the parameters</u>. The solution of the problem can then be given in closed form. It should, however, always be remembered that least squares is often chosen for mathematical convenience. This was clearly pointed out by Gauss:

> "Denoting the differences between observation and calculation by Δ, Δ', Δ", etc., the first condition will be satisfied not only if ΔΔ + + Δ'Δ' + Δ"Δ" + etc., is a minimum (which is our principle), but also if $\Delta^4 + \Delta'^4 + \Delta''^4$ + etc., or $\Delta^6 + \Delta'^6 + \Delta''^6$ + etc., or in general, if the sum of any of the powers with an even exponent becomes a minimum. But of all these principles ours is the most simple; by the others we should be led into the most complicated calculations."

Because of the simplicity of the least squares problem it is always tempting to use this formulation. It is, however, useful to remember that if the identification problem is solved using a digital computer there is no particular reason to choose a quadratic criterion.

When the disturbances of a process are described as stochastic processes, the identification problem can be formulated as a statistical parameter estimation problem and the whole artillery of statistical estimation methods becomes available. The maximum likelihood method is a popular technique which has many attractive statistical properties. See e.g. [14], [15] and [16]. This method can also be interpreted as a least squares criterion if the quantity to be minimized is taken as the sum of squares of the prediction errors or more precisely in the case of discrete time observations at times $t_0, t_1, \ldots, t_N$ the criterion is given by

$$V(\theta) = N/2 \log \det R + \frac{1}{2} \sum_{1}^{N} \varepsilon^T(t_i) R^{-1} \varepsilon(t_i) + \frac{Np}{2} \log 2\pi \tag{2.6}$$

where $\varepsilon(t_i)$ are the prediction errors

$$\varepsilon(t_i) = y(t_i) - \hat{y}(t_i | t_{i-1}) \tag{2.7}$$

The maximum likelihood criterion (2.6) is based on the assumption that the prediction errors ε are normally distributed. Notice, however, that the criterion (2.6) can still be postulated even if the prediction errors are not normal. The corresponding identification method then becomes a prediction error method.

Parameter estimation methods

The parameter estimation problem can be formulated as follows. Given data $\mathcal{D}$, in the form of input-output records from a process, a class of models $\mathcal{M}$ and a criterion C. Find a model in the class $\mathcal{M}$ which fits the data as well as possible according to the criterion C.

There are many possibilities to combine experimental conditions, model classes and criteria. There are also many different ways to organize the calculations of the estimate. Consequently there is a large number of different identification methods available. It is useful to remember, however, that they are all based on the same principle and that they only differ in the choice of model structures, criteria and organization of the calculations.

One broad distinction is between on-line methods and off-line methods. The on-line methods give estimates in real time as the measurements are obtained. The on-line methods are the only alternative if parameters are timevarying and when the estimates must be produced in real time. The on-line algorithms available are also frequently simpler to program than the off-line methods. The draw-back with the on-line methods is that they are less reliable. They may not necessarily converge. Even if they converge they may converge to the wrong solution. In many cases the off-line methods will also give estimates with higher precision. Off-line techniques are therefore preferrable unless the processes are timevarying or it is necessary to obtain estimates in real time.

Choice of methods

The large number of identification methods available are of course very confusing for an industrial engineer who is primarily interested in having a tool to obtain a model. Several attempts to compare different identification methods have also been made. See e.g. [19]. The comparisons are largely inconclusive in the sense that there is no method that is universally best. Fortunately it appears, however, that the choice of techniques is not crucial. Personally I would recommend a prospective user to learn the classical methods (frequency and transient response analysis, correlation and spectral analysis), least squares with extensions and maximum likelihood. The least squares method is very simple and easy to understand. Under some circumstances it will give estimates with the wrong mean values (bias). This can, however, be overcome by using various extensions like multistage least squares, extended least squares and generalized least squares. The major drawback by least squares is that it requires a model structure which is linear in the parameters. The maximum likelihood method is a very general technique which can be applied to a wide variety of model structures.

2:34

Model validation

When a model has been obtained from experimental data it is necessary to check the model in order to reveal its inadequacies. Black box models should be given particular attention in this respect. For model validation it is useful to determine step responses, impulse responses, poles and zeros, model- and prediction errors etc. Calculation of statistical quantities like correlation of prediction errors and cross correlations between inputs and prediction errors can also be revealing. Since the purpose of the model validation is to scrutinize the model with respect to inadequacies it is useful to look for quantities that are sensitive to model changes.

Provided that assumptions on the data generation can be made, many useful results can be obtained. For example it is sometimes possible to determine the statistical properties of the estimates for large data sets. Assuming that the mechanism which generated the data is known it is also possible to analyse if the estimates converge with increasing data sets. In particular if the model structure is flexible enough to include the data generating mechanism it is then also possible to obtain conditions such that the estimates will converge to their "true values". Statistical methods can also be used to decide between models having different structures. For example, the choice between the models having a different number of parameters can be formulated as a hypothesis test using the test quantity

$$t = \frac{V_1 - V_2}{V_2} \cdot \frac{N - p_2}{p_2 - p_1}, \quad p_2 > p_1 \qquad (2.8)$$

where V_i is the loss function (e.g. the negative logarithm of the likelihood function) of the model having p_i parameters and N the number of sampling points. The model with more (p_2) parameters is preferred if the value t is sufficiently large.

An interesting approach to this problem has recently been given by Akaike [17] who suggests using the criterion

$$AIC = -2 \log (ML) + 2p \qquad (2.9)$$

where ML is the maximum likelihood and p is the number of parameters. Akaike's criterion, which is based on information theoretic considerations, is equivalent to (2.8) if V_1 is close to V_2. Other tests are given in [18].

When the identification problem is formulated as a statistical parameter estimation problem, there are many ideas and results from statistics that can be exploited. For example it is possible to assign accuracies to the parameter estimates by using the second derivative of the likelihood function. The statistical approach requires, however, that certain assumptions are made on the mechanism which generated the data i.e. the real process. This is most unpleasant because the real process is often nonlinear, timevarying, and infinite dimensional and little is known about it.

Great care should therefore be used when the results of statistical analyses are interpreted. It has been found empirically that many methods work very well on simulated data but very poorly on real data. This reflects the fact that certain results are sensitive to variations in the data generation and it indicates the needs for research into the problem of mismatch between the model structure and the data generation. A particular problem of overfitting clearly illustrates what can happen. If a model which has too many parameters is fitted to a given data set, an extremely good fit can of course be obtained for a particular data set. The high order model may, however, be very poor when applied to another data set. It is therefore a good practical rule to work with at least two data sets. One set is used for the identification and the other for the validation.

3. INTERACTIVE COMPUTING

It is a substantial effort to solve a system identification problem for an industrial process if no prior experience and no software is available. The effort can be reduced substantially if good computer software is available. In particular it has been our experience that the time and effort can be reduced substantially if suitable software for interactive computing is available.

Interactive computing requires an efficient man-machine interface. A graphical display which can be used to show curves is a necessity. Interactive software allows the problem solver to combine his insight and intuition with extensive calculation. It also gives a direct link between the problem solver and the computer without needing programmers as intermediaries.

An interactive program package for system identification, IDPAC, has been in operation since 1972 at the Department of Automatic Control at the Lund Institute of Technology. The program which was developed by Wieslander [20], [21] was originally written for a process computer PDP 15/35. It has also been run on other computers. To illustrate the potentials of interactive computing the main features of IDPAC will be described.

The program has facilities for input - output, editing and display of data. It includes several estimation procedures like correlation and spectral analysis, least squares and maximum likelihood estimation. It has facilities for simulation and model analysis. The program is command driven, which means that the user initiates the different operations by typing commands on a terminal. The program also has a MACRO facility, which means that a user can combine several commands. In this way it is possible both to have a large flexibility for the experienced user and to allow for a simple use of standardized procedures for an inexperienced user. An example of the use of the program is given below.

```
1. MOVE DK WORK ← DT DATA (1 3)
2. PLOT WORK
3. TREND ← WORK (2)  1
4. ML PAR1 ← WORK 1
5. ML PAR2 ← WORK 2
6. ML PAR3 ← WORK 3
```

The first command simply moves the columns 1 and 3 on the data file DATA from magnetic tape to a work area on the disc. The second command plots the data on the graphical display. The third command removes a first order trend from the second column in the file WORK. The commands 4, 5 and 6 perform Maximum Likelihood estimation of the parameters in the model (2.2) for orders 1, 2, and 3 using the data in the file WORK. The estimated parameters are stored in the files PAR1, PAR2, and PAR3.

The analysis of the models can proceed as follows.

```
7. RESID RES ← PAR2 WRK 20
```

This means that the residuals of the model with parameters PAR2 are computed and stored in the file RES. In this computation the covariance function of the residuals and the cross covariance function between the input and the residuals are also computed and automatically displayed. The commands

```
8. DETER DET ← PAR2 WORK (1)
```

computes the deterministic output of the model with parameters PAR2 when the input is the process input WORK (1) and the disturbances neglected. The command

```
9. PLOT NL WORK (2) DET
```

finally plots the process output WORK (2) as separate points and the output of the simulated model.

Command-driven programs like IDPAC have several advantages. The commands can be read from a file on disc instead from the input terminal. By combining this with the macro facility it is easy to obtain new commands simply by combining already existing commands. In this way it is easy to generate commands for multistage least squares, extended least squares, by simple combinations of the basic least squares command. In this way IPDAC is almost like a special language for system identification.

The use of a macro will be demonstrated using an example. Assume that a transfer function model (2.2) has been estimated using a parameter estimation scheme which also estimates the parameter uncertainties. Since the transfer function parameters and their uncertainties do not give much physical insight it is useful to make a Monte Carlo simulation of the responses of a system whose parameters have a distribution with the estimated means and covariances. This is simple in principle but tedious

2:36

to program. Using the MACRO facility the problem is solved as follows.

```
1. MACRO MCSIM Y ← MOD U NL
2. FOR I = 1 TO NL
3. RANPA P ← MOD
4. DETER Y(I) ← P U
5. NEXT I
6. PLOT Y
7. END
```

This macro generates the new command

```
MCSIM Y ← MOD U NL
```

which performs NL number of MonteCarlo simulations of a system MOD having uncertain parameters. The input signal is U and the output signals are stored as columns in the file Y. The first line is simply the macro definition. Lines 2 and 5 controls the iteration. The third line generates a parameter vector P by sampling a gaussian distribution whose mean value is the estimated parameters θ and whose covariance is the estimated covariances R. These are stored in the file MOD. The fourth line is a simple simulation command. It generates the output Y from a model with parameters P having the input U.

Having defined the Macro it can now be used as follows:

```
1. ML MODEL ← DATA 2 SA
2.       SAVE COMAT
3. LET NPLX. = 100
4. INSI U NPLX
5.     > PULSE
6. MCSIM Y ← MODEL U 6
```

The first command is an ML-command to generate a second order ML model from the measured data stored in the file DAT. The argument SA in the command 1 means that a special command is required. The second line specifies that the covariance matrix should be saved and stored in the file called MODEL. The third command defines that the variable NPLX should be given the value 100. A signal called U of length NPLX is defined in statement 4 and statement 5 specifies this signal to be a unit pulse. Command number 6 finally calls the macro command that was just generated. The curves shown in Figure 1 are then displayed on the graphic screan.

The experiences with the interactive package IDPAC have been very good. The program has made it possible to analyse results from industrial experiments quickly and at a reasonable cost. The program has also been a very useful teaching aid. It has made it possible to teach system identification efficiently in a short time (about a week) both to students in the university and to engineers in industry. The program package is now being used by a number of industries.

4. EXAMPLES OF USES OF SYSTEM IDENTIFICATION

The purpose of this section is to illustrate uses of system identification for process modeling by some examples. The examples are chosen to reflect different ways in which the identification methods can be used. The first example, paper machine dynamics, shows the use of classical spectral analysis and black box modeling for the purpose of assessing the benefits of control, design of control laws and diagnostic checking of controllers in operation. The application includes modeling of process dynamics and disturbances. The second example, drum boilers, illustrates how identification methods are used to get an appreciation of the necessary complexity for a model whose purpose is to give the main features of nonlinear drum boiler dynamics. The third example, ship dynamics, illustrates how system identifica-

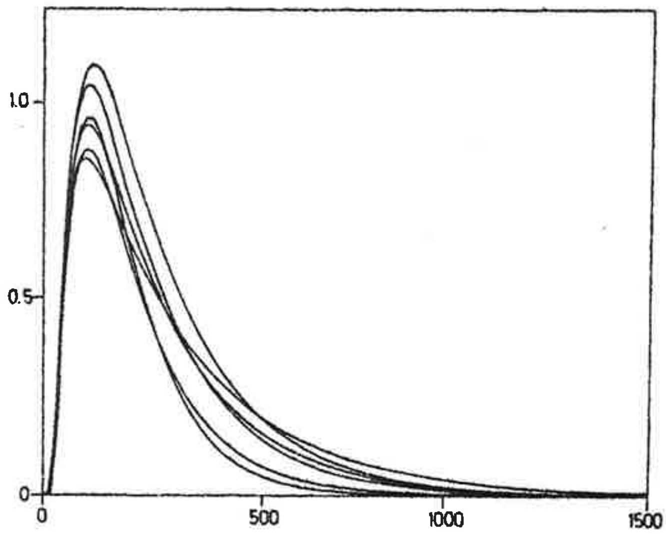


Figure 1. Monte Carlo simulation of a model of ship dynamics. The curves show how the uncertainties in the parameters of a transfer function model are reflected in uncertainties in the impulse response.

tion methods are used to determine parameters that are of physical relevance. It also illustrates identification of disturbance models. The fourth example demonstrates identification of parameters in partial differential equation models. The examples are all based on experiments using real data. The following discussion is limited to the main characteristics of the examples because they are discussed in detail elsewhere.

Paper machine dynamics

The purpose of modeling in this case was twofold: to determine the performance that could be expected from an optimal basis weight controller and to determine the optimal control strategies. The particular application is described in detail in [22] and [23], where also further references are given.

A schematic diagram of the process is shown in Figure 2. The output to be controlled is the basis weight and the control variable is the thick stock flow. The process dynamics is characterized by mixing dynamics and a transport delay. It is easy to determine the transport delay but it is difficult to determine the mixing dynamics from physical considerations. There are no possibilities to determine the characteristics of the disturbances from a priori data.

The disturbances were investigated by analysing the fluctuations in basis weight during normal operation. Probability distributions, spectral densities, and covariance functions were determined. The variability of these characteristics with time were also explored. The analysis revealed that it was reasonable to characterize the fluctuations in basis weight as a stationary stochastic process. The measured spectral density also indicated that a significant contribution to the disturbances were in a low frequency range (lower than 0.001 rad/s). Since the time delay of the process is of the order of 100 s and the dominant time constants are also of the same magnitude the preliminary investigations indicated that improvements could indeed be obtained by a better control. To provide quantitative numbers autoregressive models were fitted to the data. These models have the form

$$A(q^{-1})y(t) = C(q^{-1})e(t) \qquad (4.1)$$

where A and C are polynomials in the backward shift operator q^{-1}, i.e.

$$A(q^{-1}) = 1 + a_1q^{-1} + \ldots + a_nq^{-n}$$

$$C(q^{-1}) = 1 + c_1q^{-1} + \ldots + c_nq^{-n},$$

y is the measured output and e white noise. The order n of the model (4.1) was determined using a combination of hypothesis testing, analysis of the residuals and zeros of the polynomials $A(q^{-1})$ and $C(q^{-1})$. Having obtained the model (4.1) it is straightforward to show that the error in predicting the stochastic process y k steps ahead is given by

$$E\,[y(t+k) - \hat{y}(t+k|t)]^2 = \lambda^2 \left[1 + \sum_1^{k-1} f_i^2\right]$$

where f_i are the coefficients in the series expansion

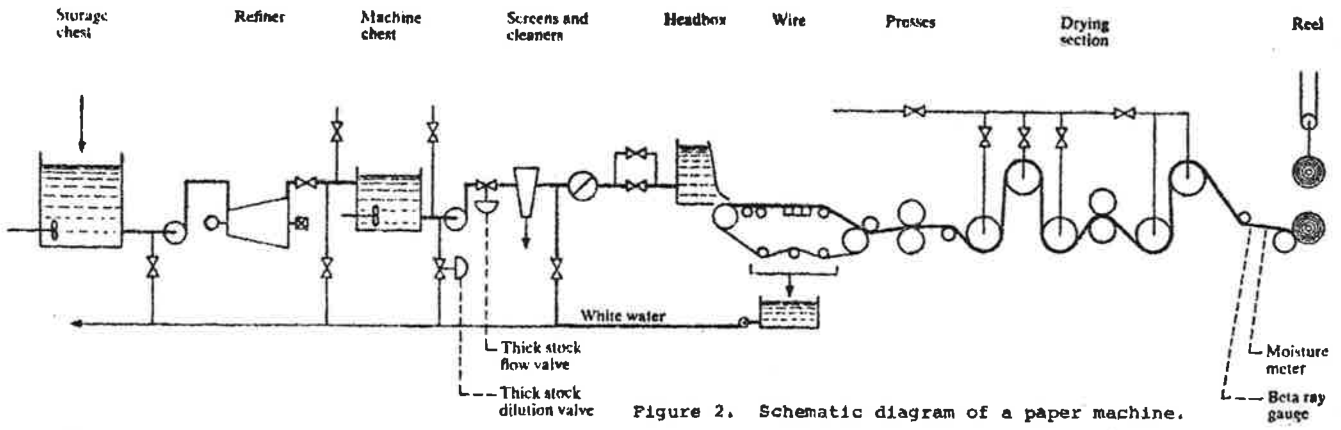


Figure 2. Schematic diagram of a paper machine.

2:38

$$\frac{C(q^{-1})}{A(q^{-1})} = 1 + f_1q^{-1} + f_2q^{-2} + \ldots$$

which are easily determined by long division and λ is the standard deviation of the e:s in the model (4.1), see [24].

Figure 3 shows a graph of the prediction errors versus k. With this graph it is easy to evaluate the benefits that can be derived from optimal control. For a stable minimum phase system it can be shown that a regulator which minimizes the variance of the process output will give a control error which is equal to the error in predicting the output over a time horizon which is the sum of the timedelay in the process and one sampling interval. The effects of choosing different sampling rates in the controller can similarly be estimated from the graph. In the particular situation shown in Figure 3 the variance can thus be reduced by a factor of 9. This means that the standard-deviation will be reduced from about 1.5 g/m^2 to 0.5 g/m^2. For a large paper machine this corresponds to a considerable saving.

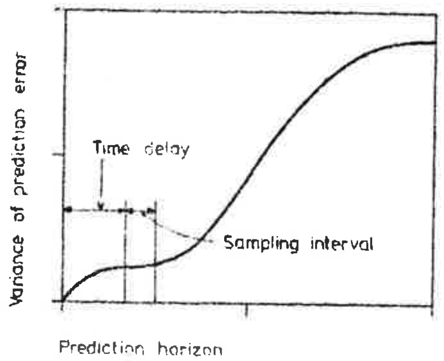


Figure 3. Variance of the error in predicting the disturbances over different time intervals.

Being convinced that there is indeed an incentive to be gained from an improved control it is worthwhile to pursue the investigations. To provide more accurate estimates it is necessary to know the process dynamics. Identification experiments were planned based on crude estimates of process dynamics. In the experiments the thick stock flow valve was perturbed and the basis weight was measured. The amplitude of the perturbations were chosen so that the quality would not change significantly. The frequency content was determined based on the timedelay and the preliminary estimates of the time constants. The shape of the signal is not critical. Several experiments were made in order to see if the process model changes with time and for the purpose of model validation. A typical result is shown in Figure 4. Black box models of the form (2.2) were fitted to the data using the maximum likelihood method. The order n was determined using hypothesis testing and the model was validated by determining the correlation function of the residuals and the cross correlations between residuals and inputs. The validation was also done using other data sets. Having a model (2.2) it is straightforward to determine minimum variance control strategies and the minimum variance that could be obtained if a minimum variance strategy would have been used during the experiment. When the control law is obtained it is easy to program it into a process computer to obtain the desired result. This technique of process modeling and control design was applied in the Gruvön paper mill in Sweden in 1964-65. It is reported in [22], [23]. The theory required is given in [24].

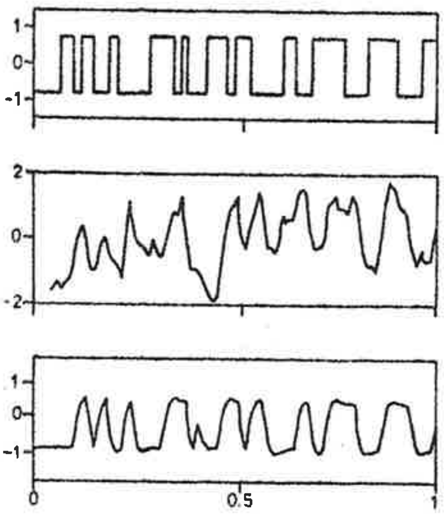


Figure 4. Input and output from a process identification experiment designed to determine the basis weight dynamics. The model output and the prediction errors are also shown in the figure. From [22].

Notice in particular that in this case the process modeling will not only give the process model and an optimal controller, but it will also give the smallest possible variance that can be achieved. The technique is therefore useful for feasibility studies.

The type of process modeling described here has been applied to many different processes to design control-laws. The modeling has also been applied for diagnostic purposes to find out if a specific regulator has a performance that is close to the optimal. When a minimum variance controller is used this can also be tested simply by computing the covariance function of the output. See F i g u r e 5. The covariances can also be used to tune the controller as will be discussed in section 5.

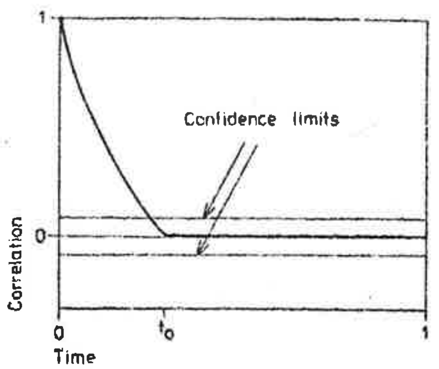


Figure 5. Covariance of control error for minimum phase system with a minimum variance controller.

Drum boiler dynamics

The purpose of this modeling exercise was to obtain a simple model for a drum boiler which could be used to determine the "storage capacity" and the possible "power reserve" of a drum boiler. The physics of the boiler are fairly well understood although the phenomena involved are fairly complicated. The key questions were related to the choice of suitable approximations and lumping of distributed parameter system. The work was based on a set of identification experiments performed on a 160 MW boiler. Perturbations in fuel flow, steam flow and feedwater flow were introduced in the experiment. All regulators were removed when these perturbations were introduced. The result of a typical experiment is shown in F i g u r e 6. System identification methods were used in the following way. Simple transfer function models (2.2) relating the important process variables to the perturbed inputs were first fitted using the maximum likelihood method. The orders of the models were determined using a combination of statistical and deterministic methods. The results clearly showed that low order models were sufficient. Physical models having the appropriate complexity were then derived. The results of the system identification were used as a guide when deciding between different approximations and different ways of lumping the distributed parameter phenomena. The parameters of the physical models were then estimated from the data. The simple model is given by

$$\frac{dx}{dt} = - \alpha_1 x^{9/8} u_2 + \alpha_2 u_1 - \alpha_3 u_3$$

$$y = \alpha_4 x^{9/8} u_2$$

where the inputs u_1, u_2 and u_3 correspond to fuel flow, turbine valve position and feedwater flow, the output y is output power and the statevariable x is drum pressure. The performance of this model is illustrated in F i g u r e 6. A further discussion of this particular application is given in [25], [26], [27], [28], and [29].

Ship steering dynamics

This application is described in [30], [31], and [32]. The equations governing ship

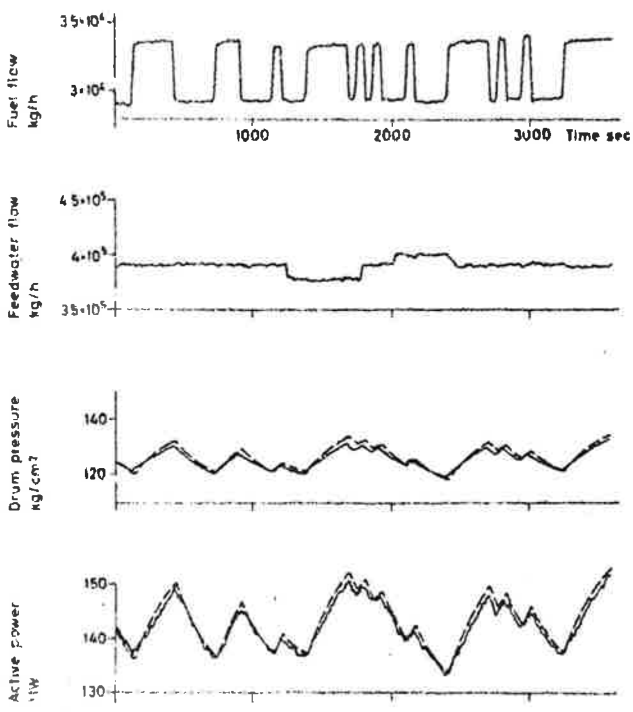


Figure 6. Data and results of identification experiment designed to determine a model of a thermal boiler. From [26].

steering dynamics are wellknown. The linearized equations do, however, contain certain parameters called the hydrodynamic derivatives. One object was to determine these parameters based on experiments on different ships. Another object was to model the disturbances acting on the ship. In the experiments the rudder of the ship was perturbed and the corresponding motion was determined by measuring the heading angle, the yaw rate and the velocity components. Different model structures were determined using hydrodynamic theory. The parameters of the models were then determined using the maximum likelihood method. A general purpose maximum likelihood identification program LISPID was used in this application. This program is described in [33]. An interesting feature of this problem was that it was natural to use a continuous time model whose parameters were fitted using discrete time measurements. One example of a model structure used is listed below.

Process dynamics

$$\begin{bmatrix} \frac{L}{V^2}(m'-Y_v') & \frac{L^2}{V^2}(m'x_G'-Y_r') & 0 \\ \frac{L}{V^2}(m'x_G'-N_v') & \frac{L^2}{V^2}(I_z'-N_r') & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dv \\ dr \\ d\psi \end{bmatrix} = \begin{bmatrix} \frac{1}{V}\theta_1 & \frac{L}{V}\theta_2 & \theta_7 \\ \frac{1}{V}\theta_3 & \frac{L}{V}\theta_4 & \theta_7\theta_8 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v(t) \\ r(t) \\ \psi(t) \end{bmatrix} dt + \begin{bmatrix} 0 & \theta_{11} \\ 0 & \theta_{12} \\ 0 & \theta_{13} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta(t_k-\tau) \\ U1 \end{bmatrix} + e(t_k),$$

$$k = 0, 1, \dots, N$$

Input delay

$$\tau = T|\sin\theta_{23}|$$

Measurement model

$$\begin{bmatrix} v_1(t_k) \\ v_2(t_k) \\ \dot{r}(t_k) \\ \dot{\psi}(t_k) \end{bmatrix} = \begin{bmatrix} \alpha_2 & L_1\alpha_2 & 0 \\ \alpha_3 & -L_2\alpha_3 & 0 \\ 0 & 1/\alpha_1 & 0 \\ 0 & 0 & 1/\alpha_1 \end{bmatrix} \begin{bmatrix} v(t_k) \\ r(t_k) \\ \psi(t_k) \end{bmatrix} + \begin{bmatrix} \alpha_1\theta_5 & \theta_9 \\ -\alpha_1\theta_6 & \theta_{10} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta(t-\tau) \\ U1 \end{bmatrix} dt + dw$$

Noise covariance

$$R_1 = \begin{bmatrix} |\theta_{14}| & \sqrt{(|\theta_{14}||\theta_{15}|)}\sin\theta_{16} & 0 \\ \sqrt{(|\theta_{14}||\theta_{15}|)}\sin\theta_{16} & |\theta_{15}| & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad R_2 = \begin{bmatrix} |\theta_{17}| & 0 & 0 & 0 \\ 0 & |\theta_{18}| & 0 & 0 \\ 0 & 0 & |\theta_{19}| & 0 \\ 0 & 0 & 0 & 0{\cdot}01 \end{bmatrix}$$

Initial state

$$\begin{bmatrix} v(t_0) \\ r(t_0) \\ \psi(t_0) \end{bmatrix} = \begin{bmatrix} \theta_{20}/\alpha_2 \\ \alpha_1\theta_{21} \\ \alpha_1\theta_{22} \end{bmatrix}$$

This form of the model is natural from the physical point of view because the first equation represents conservation of momentum and the second conservation of angular momentum. Notice also that the derivatives are not solved explicitely in the model and that the parameters also appear in the characterization of the covariances of the disturbances. Another interesting characteristic of the problem was that the measurements were fairly costly and that the sampling rate was not constant during the experiment. This inconvenience was overcome at the expense of computing time using the identification program LISPID. System identification theory was useful in many stages of the project. An analysis of the identifiability conditions showed the parameter combinations that could be determined under different experimental conditions. The first experiments were made under open loop conditions. This is, however, rather inconvenient because the ship may deviate considerably from the desired course during the experiment. When the consequences of making experiments under closed loop conditions were fully understood, it was possible to use the autopilot during the experiment. The perturbations are then introduced as set point changes or as changes of the parameters in the autopilot. Such experiments are much easier and also less costly to perform than the open loop experiments. Investigations of nonlinar ship steering dynamics are now under way.

2:41

Determination of thermal conductivity

All the previous examples have dealt with estimation of parameters in lumped parameter system. A distributed parameter system will now be discussed. The purpose of the modeling in this case is to determine the thermal diffusivity of a material. This is commonly done by investigating the propagation of heat in a rod of the material of interest. The temperature of one or both end points of the rod is perturbed and temperatures along the rod are measured. The conduction of heat along the rod is governed by the partial differential equation

$$\frac{\partial u}{\partial t} + \theta_1 u = \theta_2 \frac{\partial^2 u}{\partial x^2}$$

where $u(t,x)$ is the temperature at time t and position x on the rod. The parameter θ_1 is the thermal diffusivity and θ_2 represents the heat losses. Results of experiments by Leden [34], where the temperature of one end point has been perturbed, are shown in Figure 7. The estimates of the parameters θ_1 and θ_2 were calculated using the maximum likelihood method. The partial differential equation was approximated using finite differences. The consequences of various approximations were explored in [34]. The result was that the thermal diffusivity could be determined with a remarkable accuracy. The advantage compared

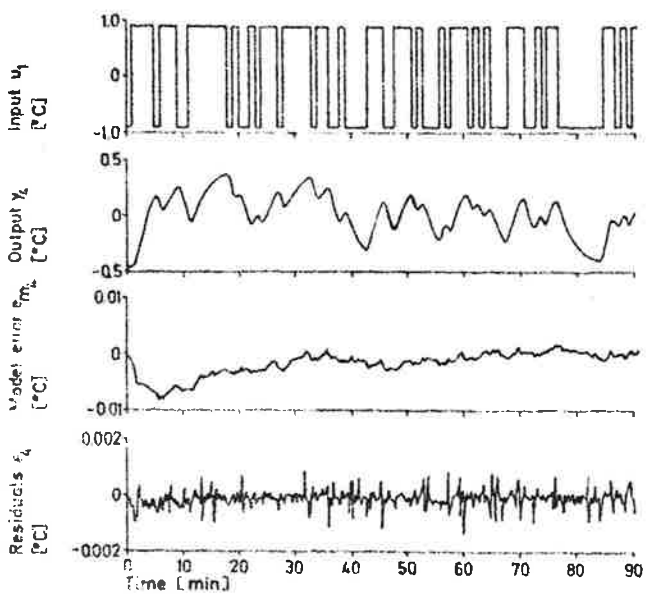


Figure 7. Data and results of identification experiment designed to determine the thermal diffusivity of copper. From [34].

with previous methods is that the requirements on the experimental conditions can be relaxed in return for increased calculations.

Summary

The examples given here will hopefully illustrate the use of system identification in process modeling. Further examples are given in [35]. A general experience obtained from these and many other applications is that the interactive software is very useful. Another experience is that it is useless to waste computing time on bad data. As a rule it is recommended that the persons who are doing the parameter estimation are also participating in the experiments.

5. ON-LINE IDENTIFICATION AND ADAPTIVE CONTROL

In many cases the problem facing a control engineer is simply to design a controller for an existing process using available sensors. Many problems of this type are solved simply by installing a three term controller and tuning its parameters. In some cases when the variable controlled is a major quality variable it may be justified to be more ambitious and to attempt to obtain an optimal regulator. It is then necessary to obtain a model for the process and the disturbances. This can be done as was described in the example on paper machine dynamics in the previous section. It is then necessary to go through the following steps:

1. Plan and perform experiments
2. Parameter estimation
3. Design of control laws
4. Implementation

For safety's sake it is advisable to repeat steps 1 and 2 twice to make sure that the results are correct. System identification clearly plays an important role in the procedure. Notice, however, that in this case the control designer has little interest in the process model as such. The model is only used as an intermediate result in order to obtain the control law. It has been my experience that with luck the problem can be solved in a week or two provided the problem solver is familiar with the technique and that he has appropriate software available. The method will also give valuable insight because it tells the best regulation that can possibly be obtained under the given circumstances and how the results are influenced by different sampling rates. Still there are few control loops that will merit such an effort of engineering. It is therefore meaningful to investigate if there are other alternatives which will lead to the desired result. An obvious way is to use an on-line parameter estimation scheme to adjust the parameters of the controller. Such regulators are called self-tuning. A block diagram of a self-tuning regulator is shown in Figure 8.

The self-tuning regulator can be thought of as being composed of three parts: an on-line parameter estimator, a controller with variable parameters, and a block which calculates suitable controller parameters based on the estimated parameters. The block diagram of Figure 8 only shows the tuning of feedback loops. Feedforward loops can be tuned in the same manner.

Regulators which include an on-line parameter estimator have been proposed for a long time. Recent developments, [36], [37], [38], [39], [40], [41], [42], have shown that regulators of this type have many interesting and useful properties. Roughly speaking

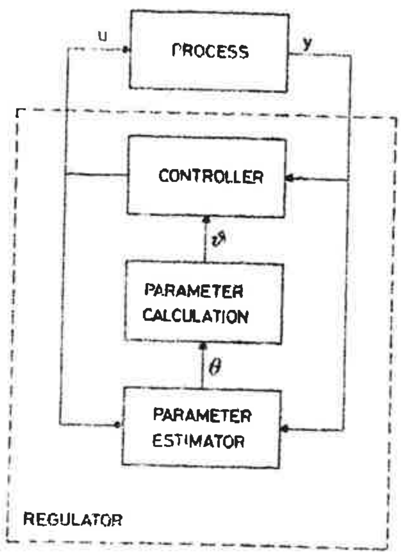


Figure 8. Block diagram of a regulator with on-line parameter estimation.

the regulators can be designed as follows: Use a design principle which will give a closed loop system with the desired performance if the parameters are known. Select an on-line parameter estimator. Determine the relation between the estimated parameters and the controller parameters. In addition it may be attempted to take parameter uncertainties into account when designing the control laws and to introduce extra perturbation signals to improve system identification. There are consequently many possible alternatives. The closed loop systems obtained are not easy to analyse because they are nonlinear, stochastic and timevarying. Some progress has, however, been made towards the understanding of the regulators. In several cases they have shown to have unexpectedly nice properties like the ability to stabilize any minimum phase system and convergence toward the minimum variance regulator that could be designed if the process and the disturbance characteristics were known.

Regulators with on-line system identification can be used in several different ways. They can be used as tuners i.e. they will be connected to ordinary control loops for certain periods to tune the parameter values for better performance. The regulators can also be used in applications where the characteristics of the process and its environment is changing continuously.

The basic self-tuning regulator STURE1 which is composed of a least squares parameter estimator and a minimum variance controller with one feedforward compensation requires only about 35 lines of FORTRAN code. The feasibility of using such regulators have been demonstrated by industrial experiments in control of paper machines [43] [44] and digesters [45], ore crushers [46], heat exchangers [47], and super tankers [48]. The ore crusher experiment was particularly interesting because a plant in Kiruna was controlled by a computer in Lund using teleprocessing over a distance of 1800 km. Several of the applications have also been in continuous operation for several years. Below we will give two examples of uses of self-tuning regulators.

Paper machine control

The problem of basis weight control of a paper machine was discussed in the previous section. A realistic simulation of the behaviour of a minimum variance controller and a self-tuning regulator is shown in Figure 9. The simulation is based on disturbances measured in a plant and an estimated process model. It represents the true situation quite well. The self-tuning regulator was initialized with all its parameters equal to zero. It is seen from the basis weight curve that the self-tuning regulator is inferior to the minimum variance regulator for the first 30 minutes. After 60 minutes there are, however, very small differences between the outputs obtained using the two regulators. It is perhaps even more instructive to look at the control signals generated by the different regulators. It is clearly seen from Figure 9 that the self-tuner is sluggish initially but after 60 minutes the self-tuning regulator produces virtually the same control signal as the minimum variance regulator. Since the paper machine dynamics does not change rapidly, good

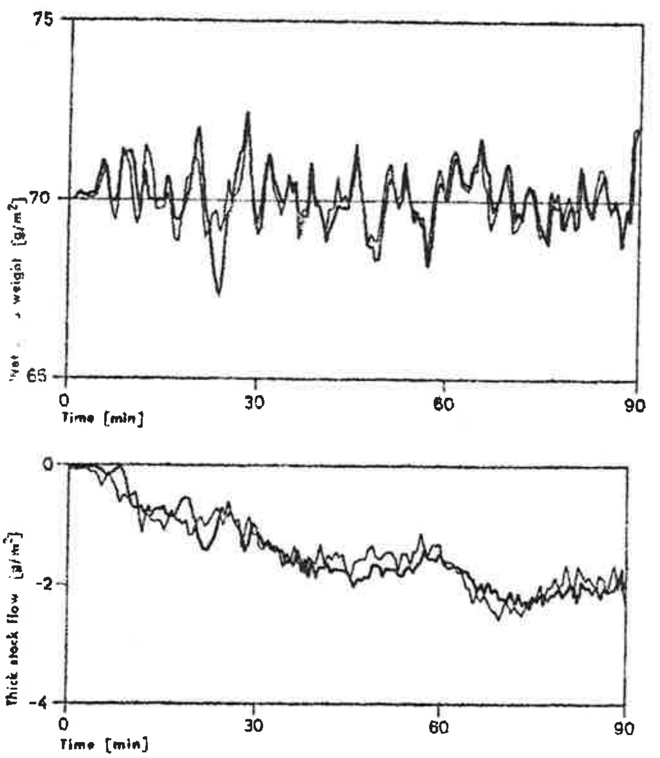


Figure 9. Simulation of self-tuning (thick lines) and minimum variance control (thin lines) of the basis weight of a paper machine. From [38].

regulator parameters will be obtained after one or two hours of operation. The self--tuner can then be replaced by a controller having fixed parameters. Comparing the effort required to go through process experiment, system identification, control design, and implementation with the effort required to implement a self-tuning regulator we find that the self-tuning regulator offers a substantial saving in terms of engineering work required. The idea to use on-line identification for self-tuning has the additional advantage that the process can be observed carefully during the tuning phase when the parameters are changing.

The basis weight loop as well as many of the other applications of the self-tuning regulator represent comparatively complex control loops. The following example due to Jensen [47] shows that the self-tuner can be conveniently used even for very simple control loops.

Self-tuning control of a heat-exchanger

A block-diagram of a special heat-exchanger which is used in some comfort control systems is shown in Figure 10. The system dynamics can be characterized as a first order lag followed by a severe nonlinearity. The process can conveniently be controlled using an integrating regulator. Due to the nonlinearity it is, however, necessary to change the gain of the regulator depending on the operating level.

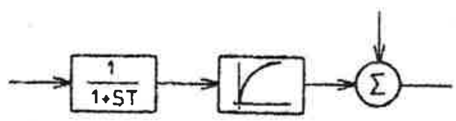


Figure 10. Block diagram of simple nonlinear system.

Figure 11 illustrates what happens if a constant gain controller is used. It is apparently difficult to get a good compromise. In Figure 11 the gain is obviously too high for low levels and too low for high levels. Wild oscillations at low levels are avoided by limiting the controller output. Figure 12 shows the results obtained by using a self-tuning regulator. In this case only the gain of the process is estimated. This requires only three additions, three multiplications, and one addition extra per iteration. This is a very modest increase of the computational effort. A comparison of Fig. 11 and Fig. 12 shows clearly the advantage of using a self--tuning regulator in this particular case.

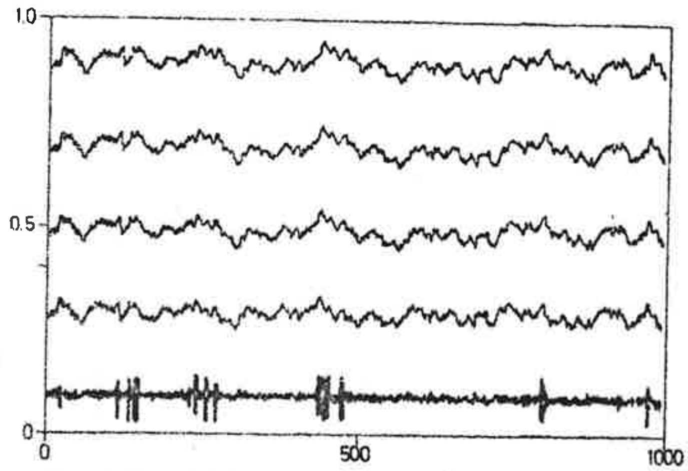


Figure 11. Outputs of the system in Fig. 10 with a fixed gain integrating controller.

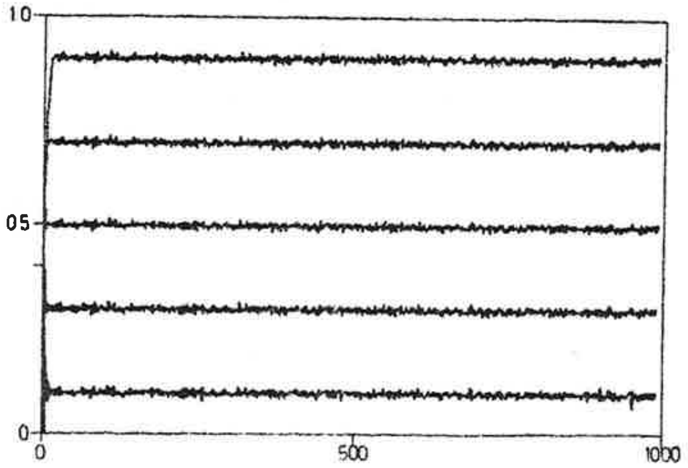


Figure 12. Outputs of the system in Fig. 10 with a self-tuning integrating controller.

6. CONCLUSIONS

The purpose of this paper has been to give an overview of system identification methods and their use for process modeling. It is clear from the examples given that classical as well as recent methods of system identification are excellent tools for the process modeler because they allow him to learn from experiments on the process. It has been mentioned that it may be timeconsuming and tedious for the industrial engineer to learn to master the system identification techniques but that the availability of interactive computer software will drastically reduce the time and effort required. It has also been emphasized that system identification methods are useful not only as modeling tools but also as an integral part of self-tuning regulators. The power of such regulators has also been shown. It therefore appears that the process modeler and the control designer can benefit from acquiring a basic knowledge of system identification methods and their uses.

7. REFERENCES

[1] Åström, K J and Eykhoff, P: "System identification - a survey". 2nd IFAC Symposium on Identification and Process Parameter Estimation, Prague. Also in Automatica 7 (1971) 123-162.

[2] Nieman, R E, Fisher, D G, and Seborg, D E: "A review of process identification and parameter estimation techniques". Int J Control 13 (1971) 209-264.

[3] Eykhoff, P: "System identification - parameter and state estimation". Wiley, New York 1974.

[4] Isermann, R: "Prozessidentifikation". Springer-Verlag, Berlin 1974.

[5] Unbehauen, H, Göhring, B, and Bauer, B: "Parameterschätzverfahren zur Systemidentifikation". Oldenbourg, München 1974.

[6] Graupe, D: "Identification of systems". Krieger, New York 1976.

[7] Cox, D R: "Planning of experiments". Wiley, New York 1958.

[8] Fedorov, V V: "Theory of optimal experiments". Academic Press, New York 1972.

[9] Mehra, R K: "Optimal input signals for parameter estimation in dynamic systems - survey and new results". IEEE Trans AC-19 (1974) 753-768.

[10] Goodwin, C G, Zarrop, M B, and Payne, R L: "Coupled design of test signals sampling interval, and filters for system identification". IEEE Trans AC-19 (1974) 748-752.

[11] Gustavsson, I, Ljung, L, and Söderström, T: "Identification of processes in closed loop - Identifiability and accuracy aspects". Preprints IV IFAC Symposium on Identification and System Parameter Estimation, Tiblisi 1976.

[12] Polyak, B T and Tsypkin, Ya Z: "Robust (noiseproof) identification". Preprints of IV IFAC Symposium on Identification and System Parameter Estimation, Tiblisi 1976.

[13] Gauss, K F: "Theoria motus corporum coelestium". Göttingen 1809. English translation published by Dover.

[14] Åström, K J and Bohlin, T: "Numerical identification of linear dynamic systems from normal operating records". Second IFAC Symposium on the Theory of Self-Adaptive Control Systems, Sept 1965. Also in Hammond, P H (editor) "Theory of Self-Adaptive Control Systems", Plenum Press 1966.

[15] Balakrishnan, A V: "Technique of system identification". Lecture notes, Institute Electrotecnio, University of Rome, 1969.

[16] Mehra, R K: "Identification of stochastic linear systems". Proc IEEE 1969 Symp on Adaptive Processes, Pennsylvania State University.

[17] Akaike, H: "Use of an information theoretic quantity for statistical model identification". Proc 5th Hawaii Intl Conf on System Science, 1972, pp 249-250.

[18] Kashyap, R L: "A Bayesian comparison of different classes of dynamic models using empirical data". Report TR-EE 76-40 School of Electrical Engineering, Pardue University, West Lafayette, Indiana, Sept 1976.

[19] Iserman, R: "Comparison of different identification methods". Fourth IFAC Symposium on Identification and System Parameter Estimation, Tiblisi 1976.

[20] Wieslander, J: "IDPAC users guide, revision 1". Report 7605, Dept of Automatic Control, Lund Institute of Technology, 1976.

[21] Wieslander, J and Gustavsson I: "IDPAC - An efficient interactive identification program". Paper 3.3, Preprints IV IFAC Symposium on Identification and System Parameter Estimation, Tiblisi 1976.

[22] Åström, K J: "Computer control of a paper machine - an application of linear stochastic control theory". IMB J Res & Dev 11 (1967) 389-405.

[23] Åström, K J: "Applications to paper making in curriculum and methods in control and systems engineering, vol II". IBM Corporation 1970.

[24] Åström, K J: "Introduction to stochastic control theory". Academic Press, New York 1970.

[25] Eklund, K: "Linear drum boiler - turbine models". Report 7117, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1971 (ph D dissertation).

[26] Åström, K J and Eklund, K: "A simplified nonlinear model of a drum-boiler turbine unit". Int J Control 16 (1972) 145-169.

[27] Åström, K J: "Modeling and identification of power system components" in Handschin, E (editor): Real-time control at electric power systems. Elsevier, Amsterdam 1972.

[28] Eklund, K and Gustavsson, I: "Identification of drum boiler dynamics" in Eykhoff (editor): Proc 3rd IFAC Symposium on Identification and System Parameter Estimation, Part 2. The Hague / Delft, The Netherlands, June 1973. North-Holland, Amsterdam 1973.

[29] Åström, K J and Eklund, K: "A simple non-linear drum boiler model". Int J Control 22 (1975) 739-740.

[30] Åström, K J and Källström, C G: "Application of system identification techniques to the determination of ship dynamics" in Eykhoff (editor): Proc 3rd IFAC Symposium on Identification and System Parameter Estimation. The Hague/Delft, The Netherlands, June 1973. North-Holland, Amsterdam 1973.

[31] Åström, K J and Källström, C G: "Identification of ship steering dynamics". Automatica 12 (1976) 9-22.

[32] Åström, K J, Norrbin, N H, Källström, C G, and Byström, L: "The identification of linear ship steering dynamics using masimum likelihood parameter estimation". Report 1920-1, The Swedish State Shipbuilding Experimental Tank, Gothenburg, Sweden, Dec 1974.

[33] Källström, C G, Essebo, T, and Åström, K J: "A computer program for maximum likelihood identification of linear multivariable stochastic systems". Preprints fourth IFAC Symposium on Identification and System Parameter Estimation, Tiblisi 1976.

[34] Leden, B: "Identification and dead-beat control of a heat diffusion process". Report 7508, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden, April 1975 (ph D dissertation).

[35] Gustavsson, I: "Survey of applications of identification in chemical and physical processes". Automatica 11 (1975) 3-24.

[36] Peterka, V: "Adaptive digital regulation of noisy systems". Paper 6.2, Preprints 2nd IFAC Symposium on Identification and Process Parameter Estimation, Prague 1970.

[37] Åström, K J and Wittenmark, B: "On the control of constant but unknown systems". Paper 37.5, Proc 5th IFAC World Congress, Paris, June 1972.

[38] Åström, K J and Wittenmark, B: "On self-tuning regulators". Automatica 9 (1973) 185-199.

[39] Wittenmark, B: "Self-tuning regulators". Report 7312, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1973.

[40] Åström, K J, Borisson, U, Ljung, L, and Wittenmark, B: "Theory and applications of adaptive regulators based on recursive parameter estimation". 6th IFAC World Congress, Boston, 1975.

[41] Åström, K J and Wittenmark, B: "Analysis of a self-tuning regulator for non-minimum phase systems". Preprints IFAC Symposium on Stochastic Control, Budapest 1974.

2:46

[42] Clarke, D W and Gawthrope, P J: "Self-tuning controller". Proc IEE 122 (1975) 929-934.

[43] Cegrell, T and Hedqvist, T: "Successful adaptive control of paper machines". Automatica 11 (1975) 53-59.

[44] Borisson, U and Wittenmark, B: "An industrial application of a self-tuning regulator". Preprints IFAC Symposium on Digital Computer Applications to Process Control, Zürich 1974.

[45] Cegrell, T and Hedqvist, T: "A new approach to continuous digester control". Preprint IFAC Symp on Digital Computer Applications to Process Control, Zürich 1974.

[46] Borisson, U and Syding, R: "Self-tuning control of an ore crusher". Automatica 12 (1976) 1-7.

[47] Jensen, L and Hänsel, R: "Computer control of an enthalpy exchanger". Report 7417, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1974.

[48] Källström, C G: "Adaptive autopilots for supertankers". Ph D dissertation, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden (to appear).

3. FREKVENS- OCH TRANSIENTANALYS

Björn Wittenmark

Avsnittet om frekvensanalys bygger på en rapport:

Åström K J: Lectures on System Identification: Frequency Response Analysis. Report 7504, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1975.

FREKVENS- OCH TRANSIENTANALYS

FREKVENSANALYS

- BAKGRUND
- EXEMPEL - UBÅT, ÖGA
- SVÅRIGHETER
- KORRELATIONSMETOD
- EXEMPEL - KRAFTNÄT

TRANSIENTANALYS

- BAKGRUND
- EXEMPEL
- BEGRÄNSNINGAR

SAMMANFATTNING

ÖVERFÖRINGSFUNKTIONEN

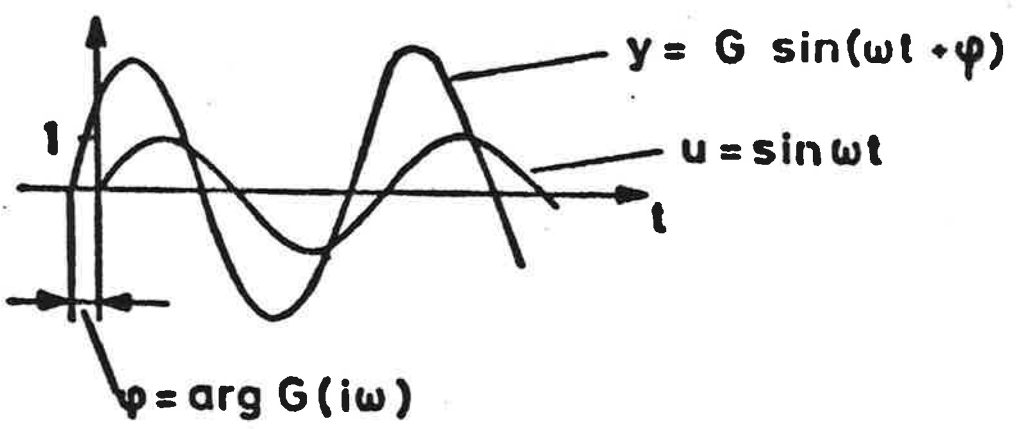
$$Y(s) = \int_0^\infty e^{-st} h(t)\, dt \quad U(s)$$

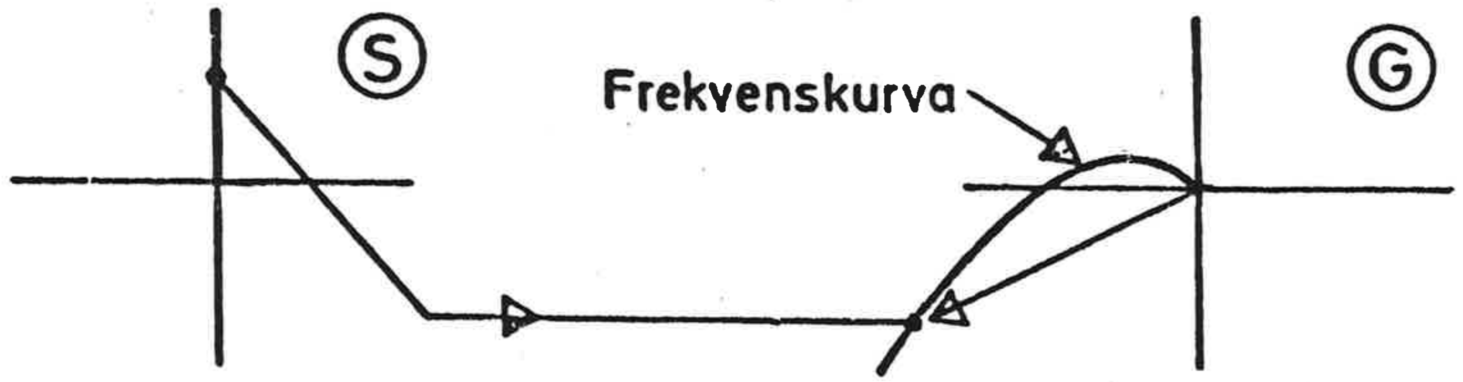
$$Y(s) = \int_0^\infty e^{-st}\, y(t)\, dt$$

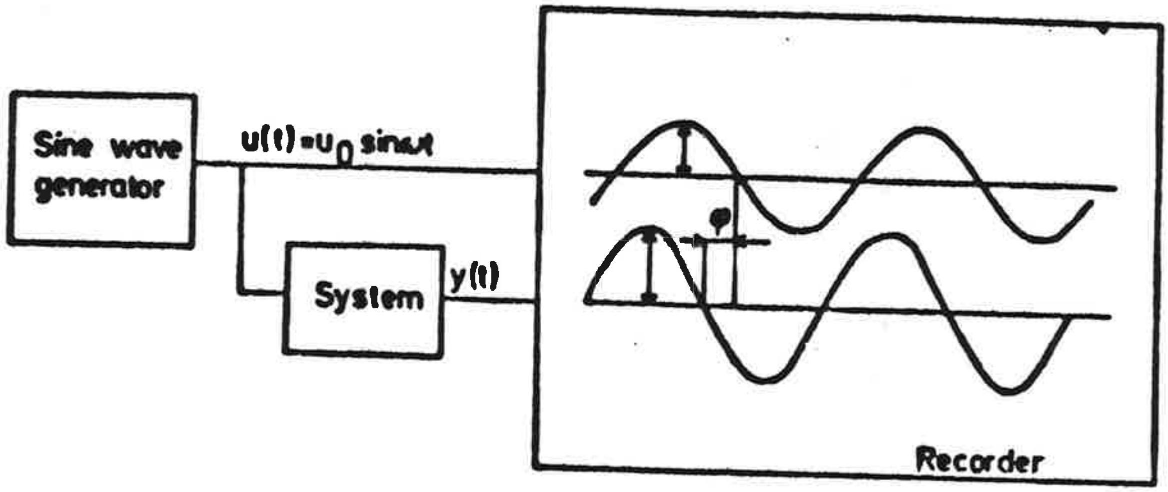
FÖR SYSTEMET

$$G(s) = \frac{b_1 s^{n-1} + \ldots + b_{n-1}}{s^n + a_1 s^{n-1} + \ldots + a_n}$$

FYSIKALISK TOLKNING







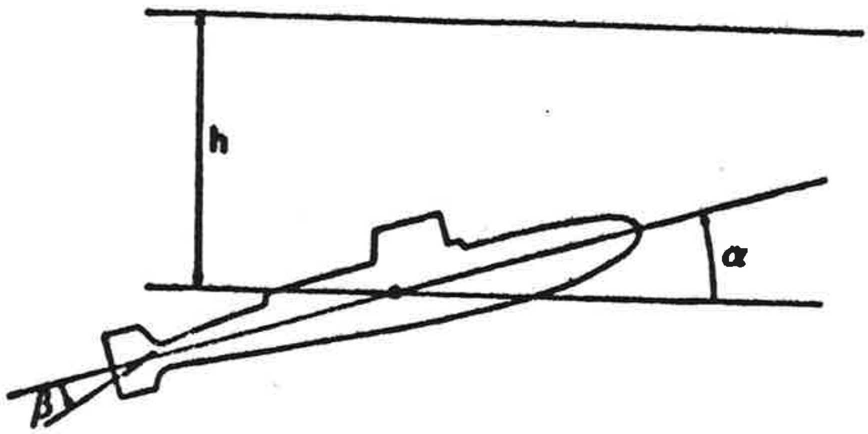
$$u(t) = u_0 \, \mathrm{Im}\{e^{i\omega t}\}$$

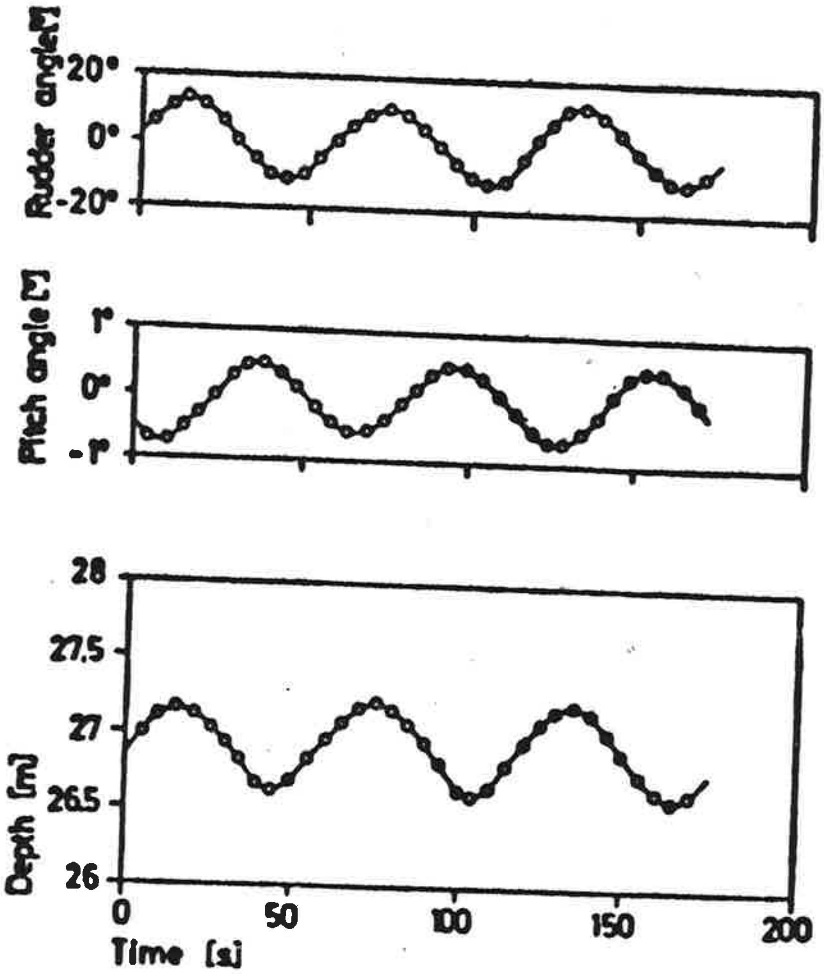
$$y(t) = u_0 \, |G(i\omega)| \, \mathrm{Im}\{e^{i\omega t + \arg G(i\omega)}\}$$

STABILA SYSTEM

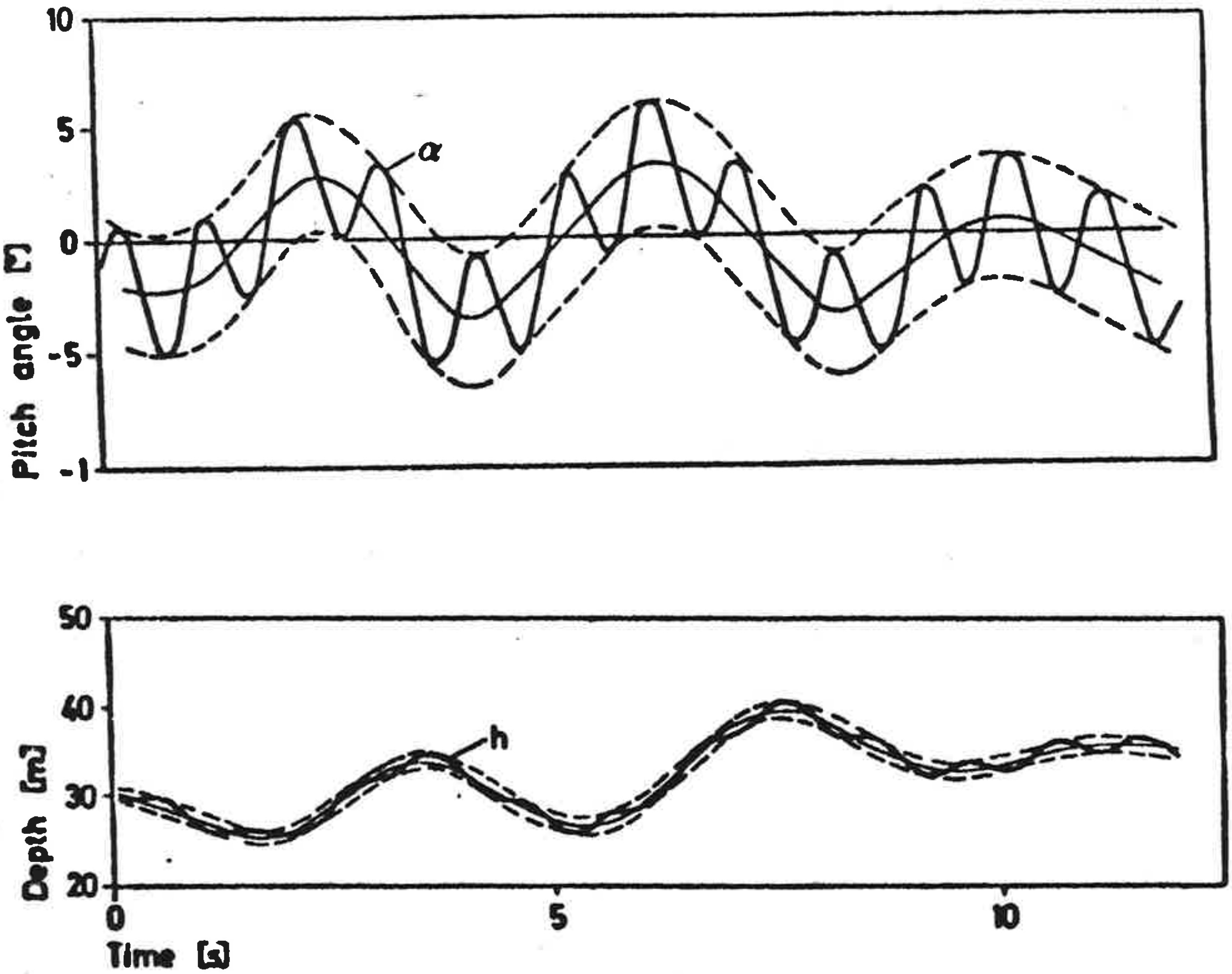
3:4

UBÅTEN GARDE PERSSON





3: 5



3:6

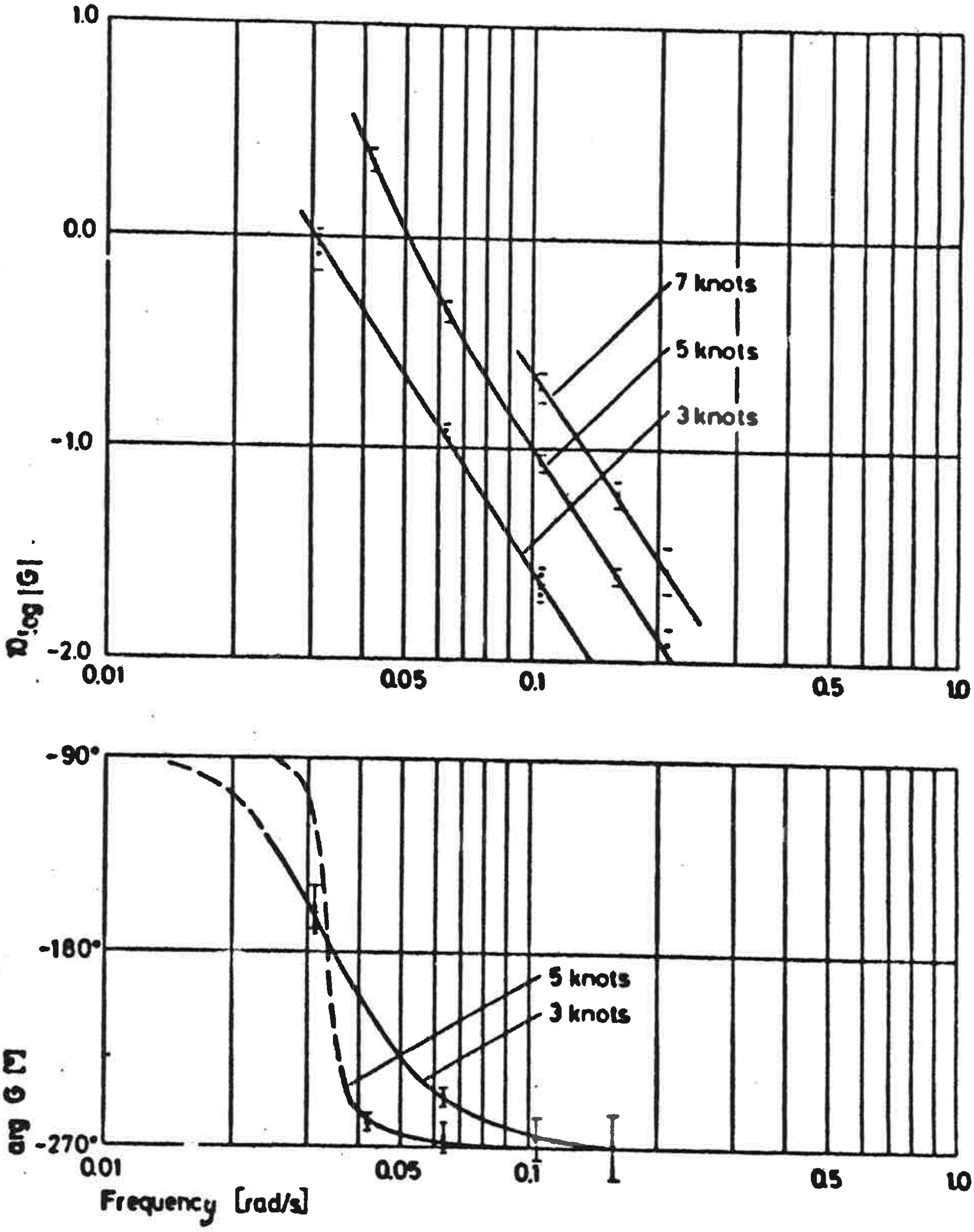


Figure 2.8 - Bode diagram of the transfer function relating depth to rudder angle obtained when applying frequency response to /determine submarine dynamics.

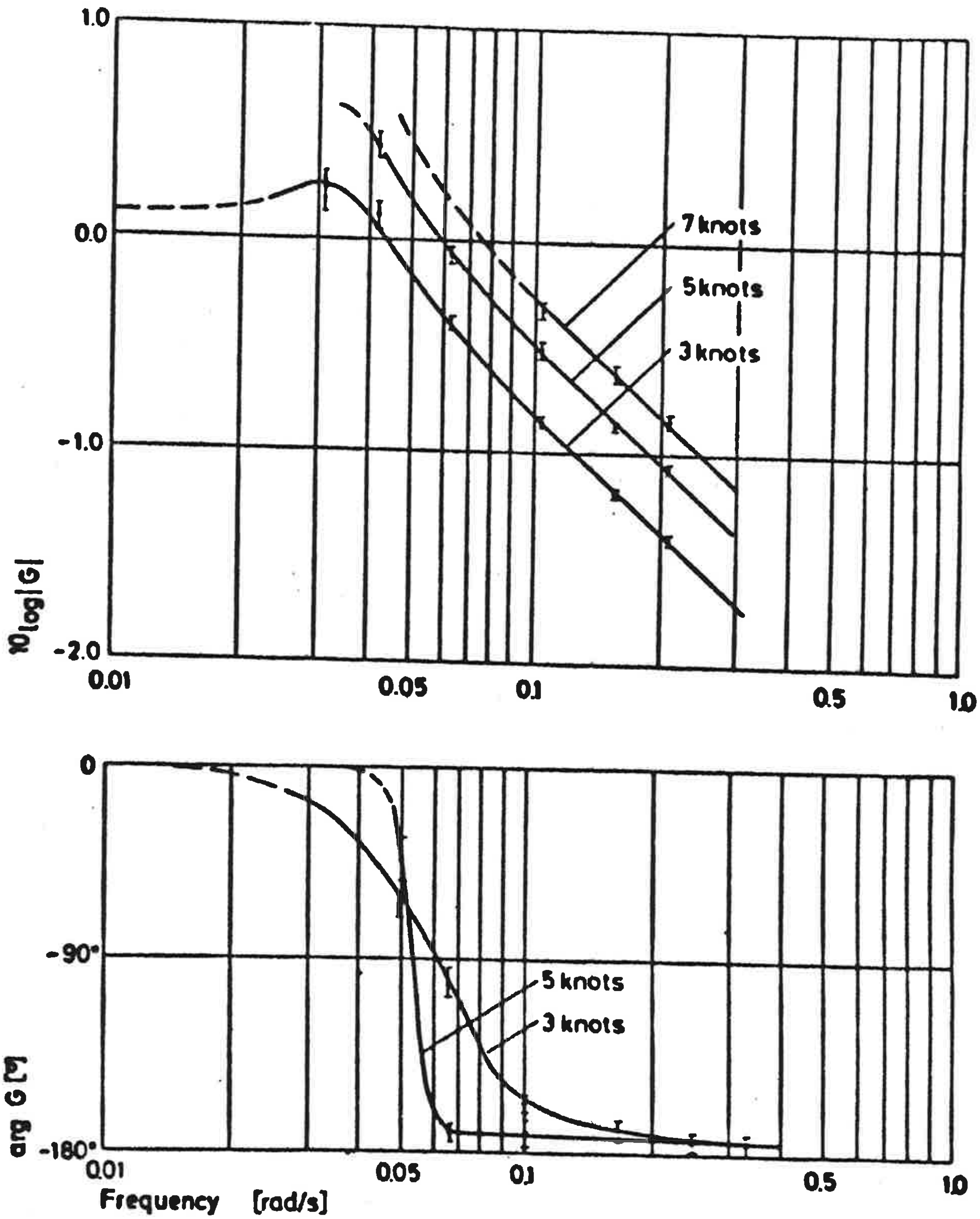
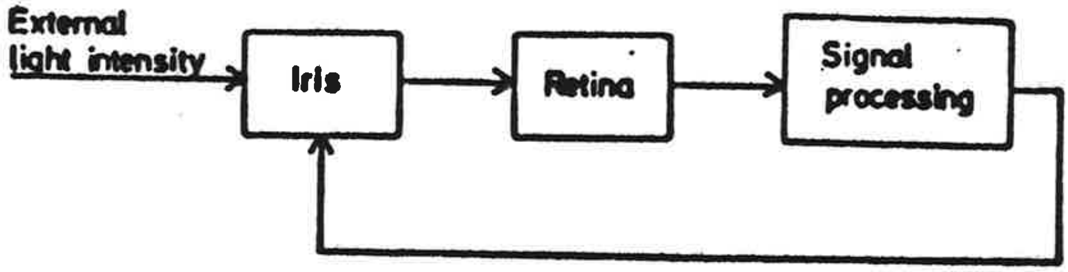
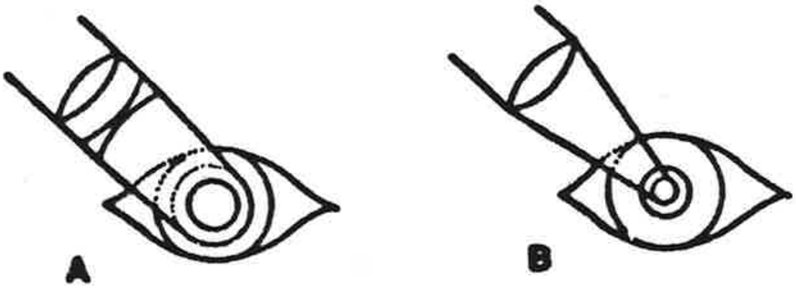


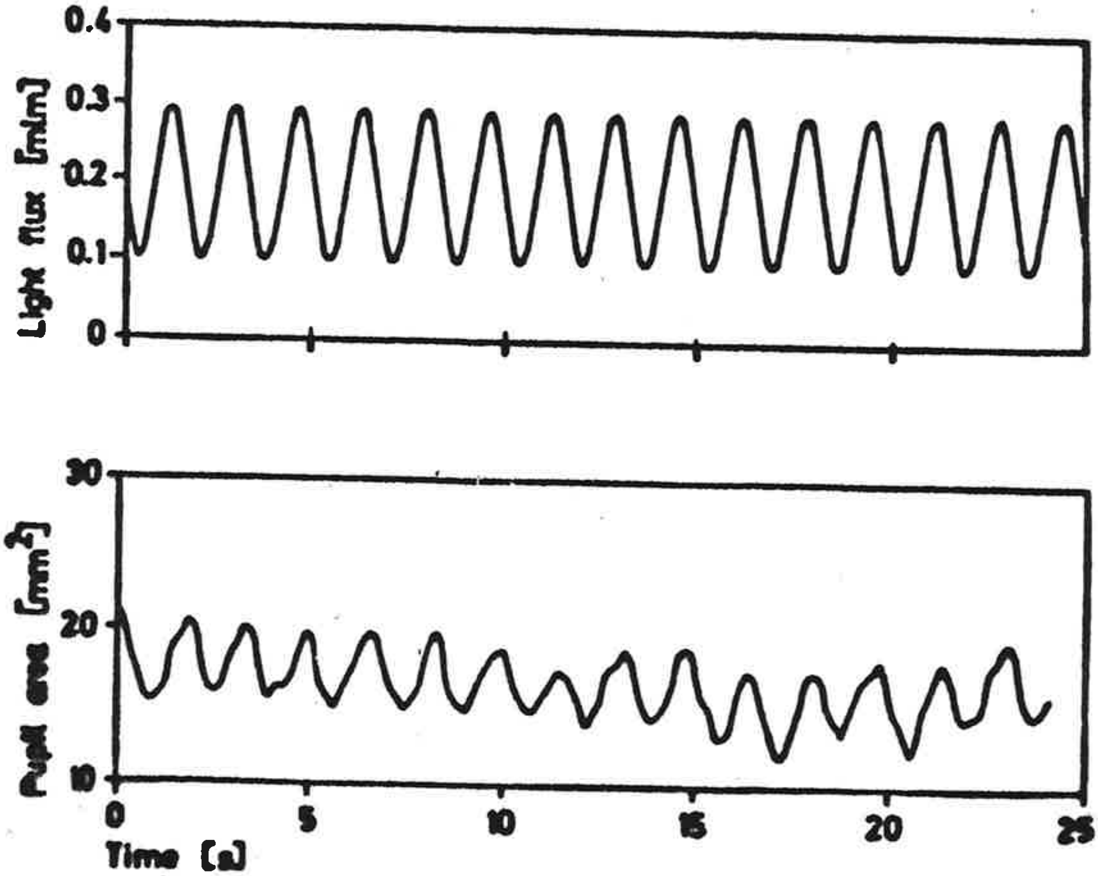
Figure 2.9 - Bode diagram of the transfer function relating pitch angle to rudder angle obtained when applying frequency response to determine submarine dynamics.

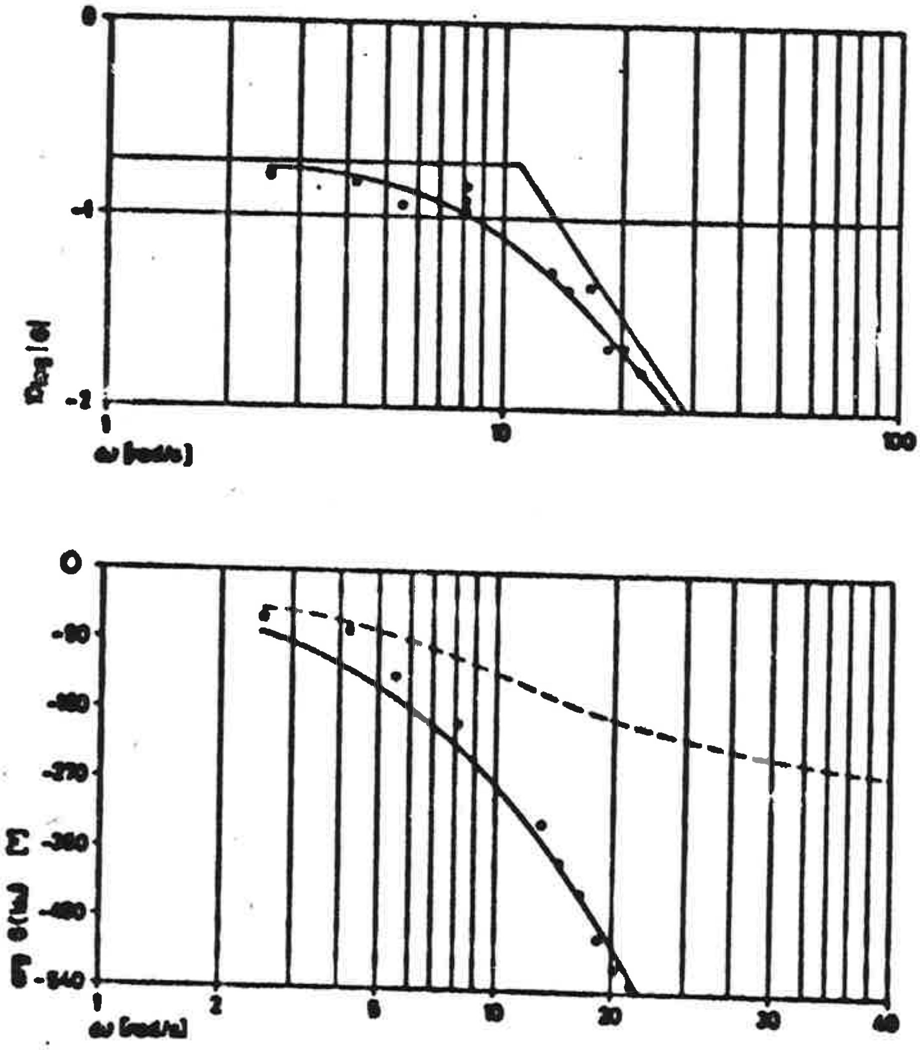
3:8

ÖGAT L.D. STARK

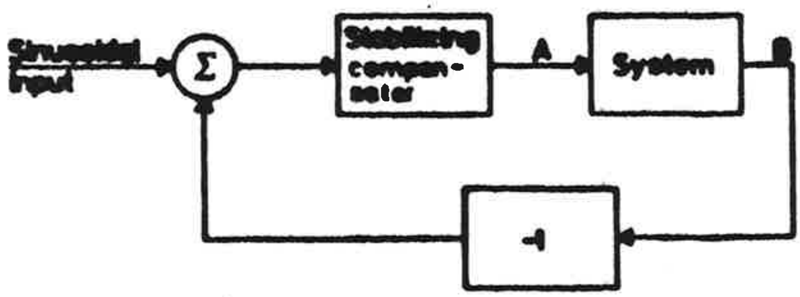


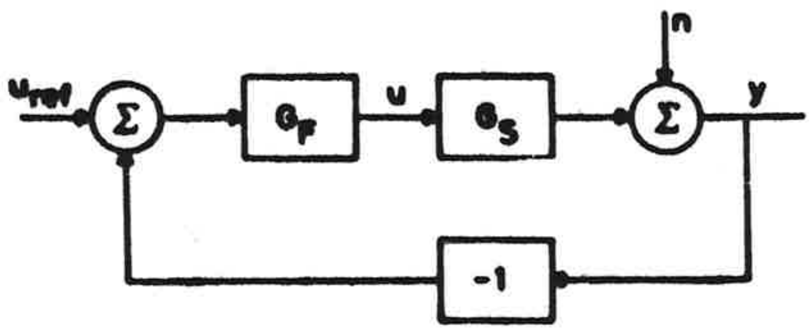





0
-1
-2
1
10
100
ω [rad/s]
0
-90
-180
-270
-360
-450
-540
[°]
1
2
5
10
20
30
40
ω [rad/s]

3:10 UNSTABLE SYSTEMS



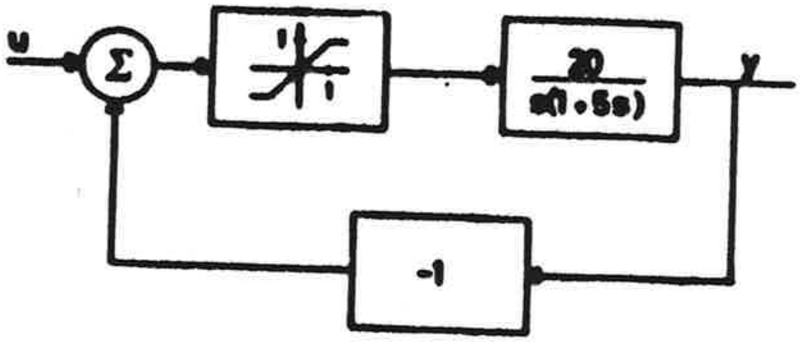


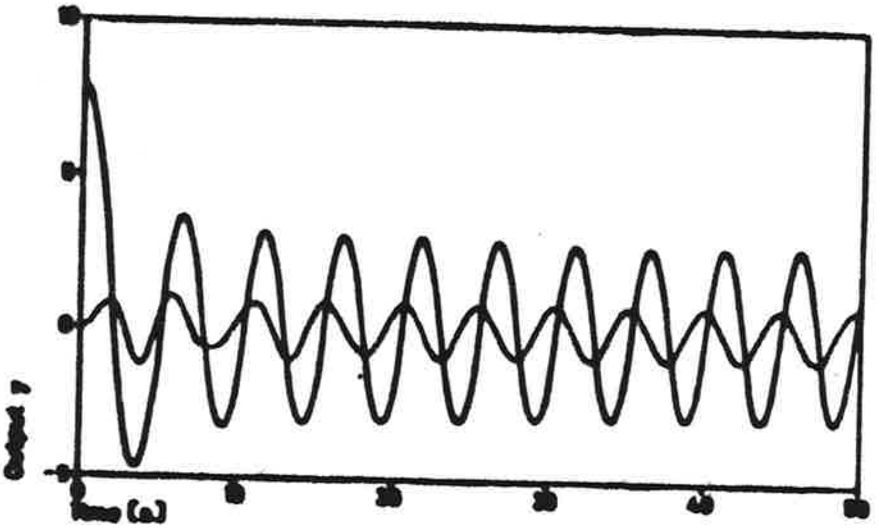
$$Y = \frac{G_S G_F}{1 + G_S G_F} U_{ref} + \frac{1}{1 + G_S G_F} N$$

$$U = \frac{G_F}{1 + G_S G_F} U_{ref} - \frac{G_F}{1 + G_S G_F} N$$

$$\hat{G} = \frac{Y}{U} \; ; \quad U_{REF} = 0 \Rightarrow \hat{G} = -\frac{1}{G_F}$$

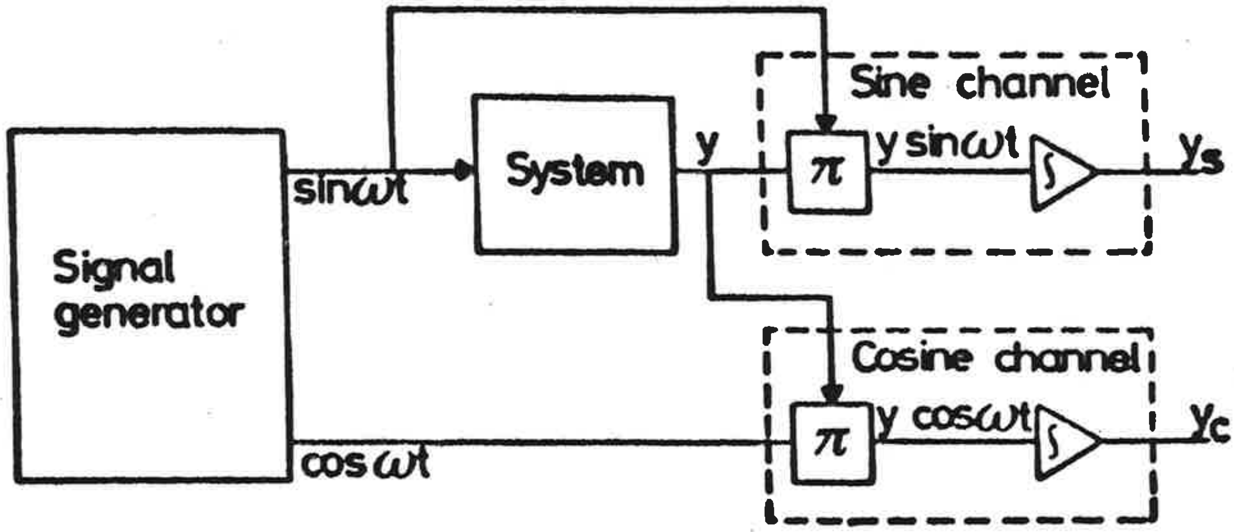
NONLINEAR SYSTEMS





$u(t) = 0.5 \sin 1.3t$

KORRELATIONS METODEN



$$y_s = \int_0^T y(t) \sin \omega t \, dt$$

$$y_c = \int_0^T y(t) \cos \omega t \, dt$$

ANALYSIS

$$\begin{cases} u(t) = u_0 \sin \omega t \\ y(t) = y_0 \sin(\omega t + \varphi) \end{cases}$$

$$y_s(T) = \int_0^T y(t) \sin \omega t \, dt = y_0 \int_0^T \sin \omega t \sin(\omega t + \varphi) \, dt$$

$$= \frac{y_0 T}{2} \cos \varphi - \frac{y_0}{2} \int_0^T \cos(2\omega t + \varphi) \, dt$$

$$y_c(T) = \int_0^T y(t) \cos \omega t \, dt = y_0 \int_0^T \cos \omega t \sin(\omega t + \varphi) \, dt$$

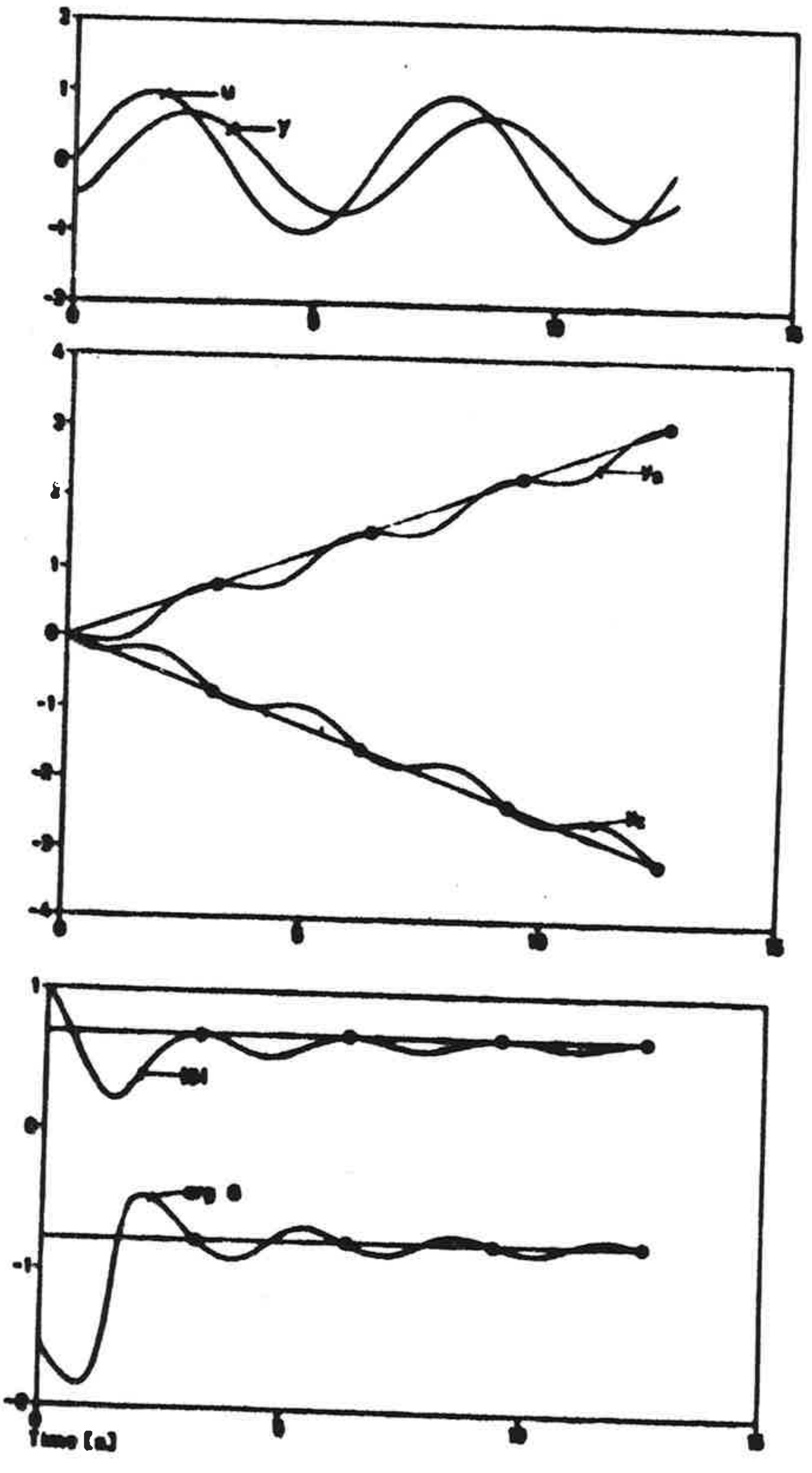
$$= \frac{y_0 T}{2} \sin \varphi + \frac{y_0}{2} \int_0^T \sin(2\omega t + \varphi) \, dt$$

IF $\omega T = \pi, 2\pi, 3\pi, \ldots$

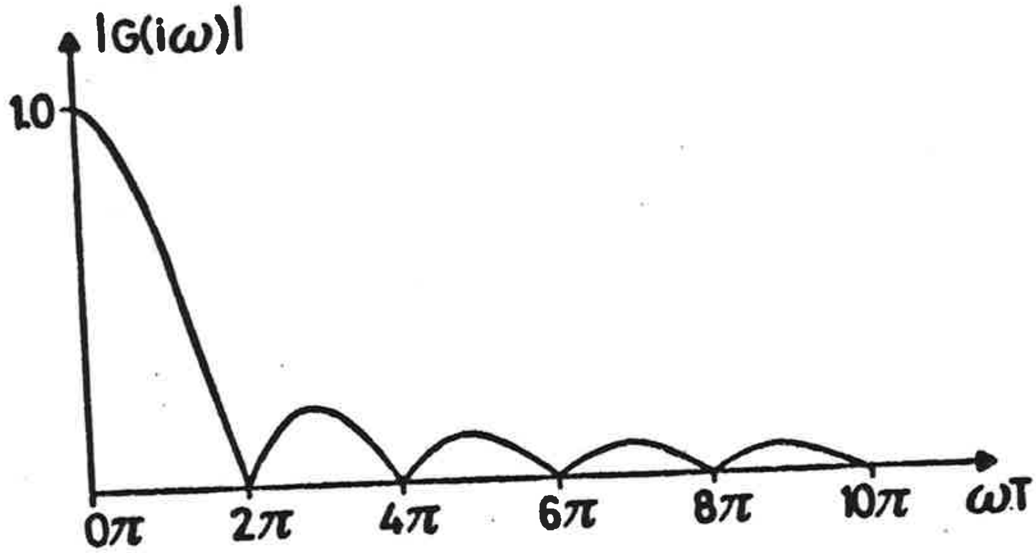
$$y_s(T) = \frac{y_0 T}{2} \cos \varphi = \frac{u_0 T}{2} \operatorname{Re}\{G(i\omega)\}$$

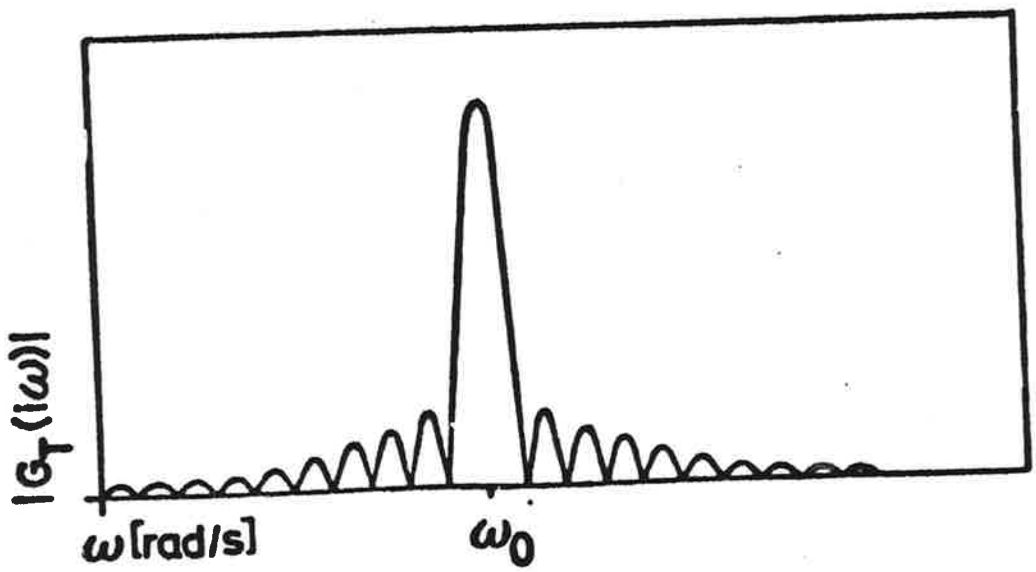
$$y_c(T) = \frac{y_0 T}{2} \sin \varphi = \frac{u_0 T}{2} \operatorname{Im}\{G(i\omega)\}$$

3:14 SIMULERING MED FREKVENSANALYSTOR BASERAD PÅ KORRELATIONSMETODEN



TOLKNING AV KORRELATIONS-KANALERNA SOM BANDPASSFILTER

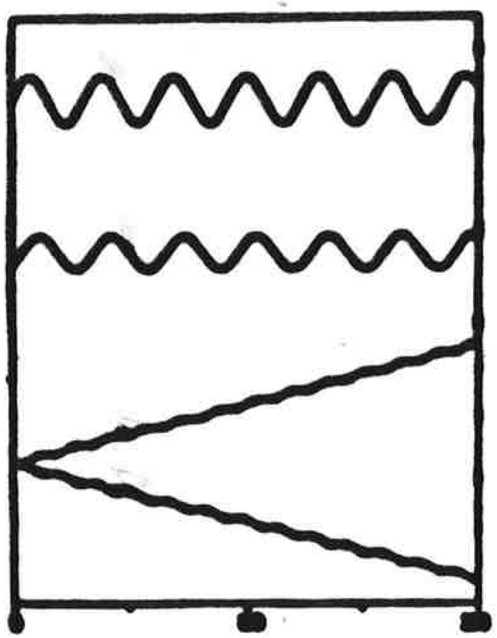




BRUSELIMINERING

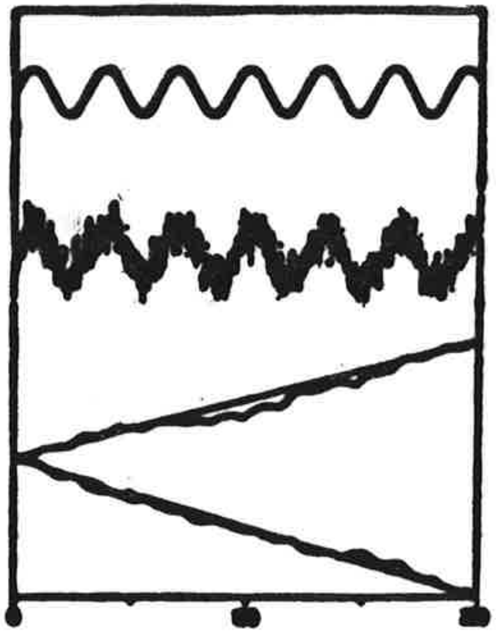
UTAN BRUS N=0

A



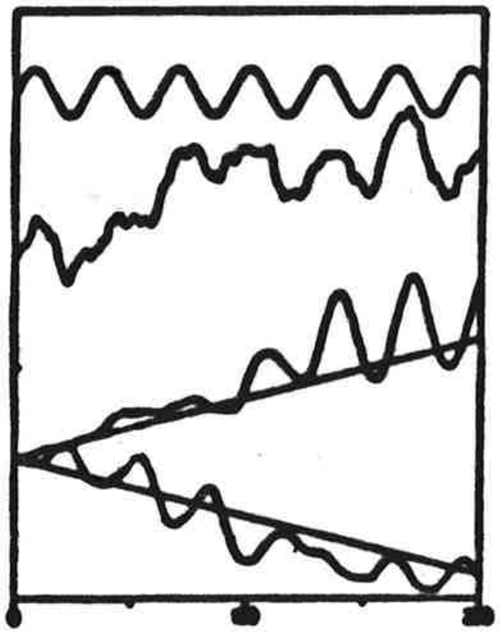
$N = \frac{1}{1+0.1s} E(s)$

B



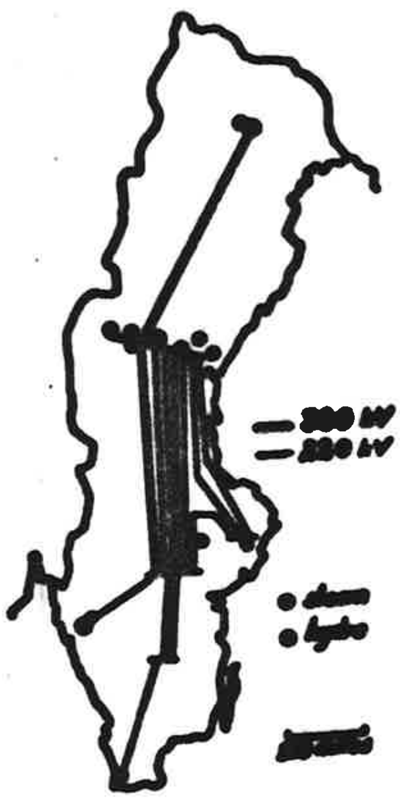
$N = \frac{1}{1+10s} E(s)$

C



KRAFTNÄTSDYNAMIK

(Garde, Oja, Persson 1949)



Modell för frekvensreglering

$$\frac{\Delta f}{\Delta P_C} = \frac{K}{1+sT}$$

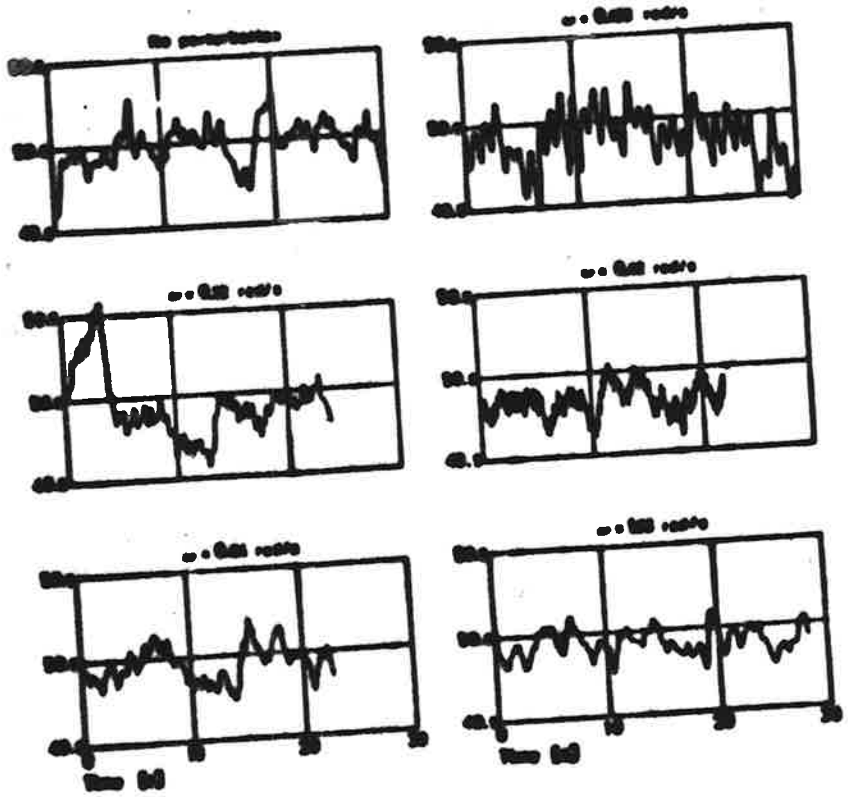
f - frekvens

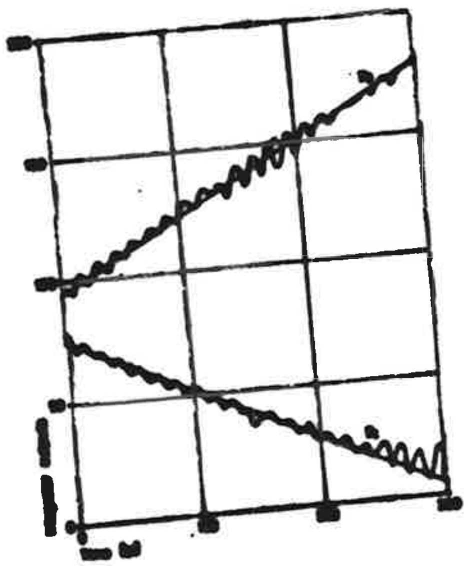
P_C - konsumerad effekt

Experiment

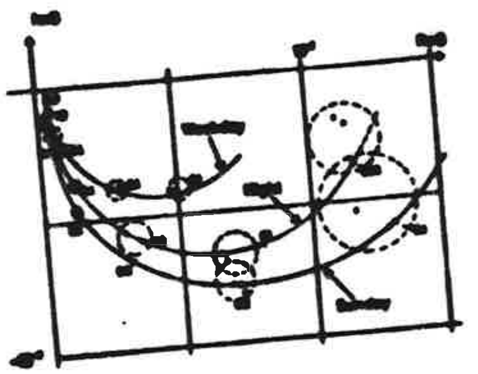
- Alla regulatorer låsta
- Manuell frekvensreglering
- Generator i Midskog gav sinusstörningar 0.05 - 1 rad/s
- Effekt variation ~10 MW
  Totala effekten i nätet ~2200 MW

Mätningar





Databearbetning



Resultat

FREKVENS ANALYS

FÖRDELAR

- ENKEL ATT UTFÖRA
- GER ÖVERFÖRINGSFUNKTION OCH BODEDIAGRAM DIREKT
- STÖRNINGAR KAN ELIMINERAS I UTBYTE MOT EXPERIMENT TID

NACKDELAR

- INSIGNALEN MÅSTE VARA SINUSFORMAD
- MÄTTIDEN KAN BLI LÅNG
- MÄTER BARA PROCESSDYNAMIKEN

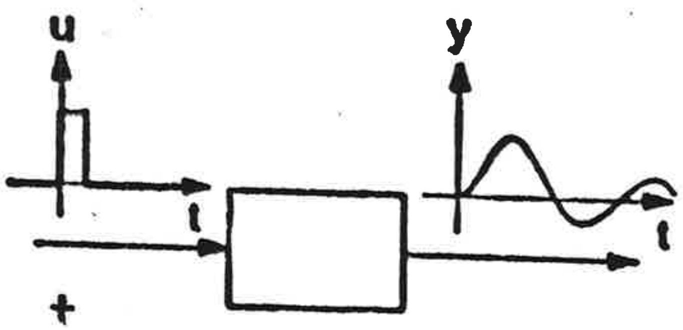
EXTERNA BESKRIVNINGAR FÖR LINJÄRA SYSTEM

DIFFERENTIALSAMBAND

Utsignal y Insignal u

$$\frac{d^n y}{dt^n} + a_1 \frac{d^{n-1} y}{dt^{n-1}} + \ldots\ldots + a_n y = b_1 \frac{d^{n-1} u}{dt^{n-1}} + \ldots + b_n \cdot u \qquad (\dagger)$$

VIKTFUNKTION IMPULSSVAR



BEROR AV INITIAL TILLSTÅND

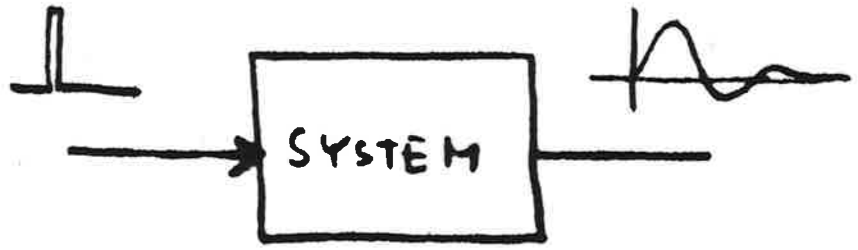
$$y(t) = \int_{-\infty}^{t} h(t-s)\, u(s)\, ds + y_p(t),$$

h impulssvar

JÄMFÖR

$$y(t) = Du(t) + \int_0^t Ce^{A(t-s)} Bu(s)\, ds + Ce^{At} x(0)$$

PRINCIP:



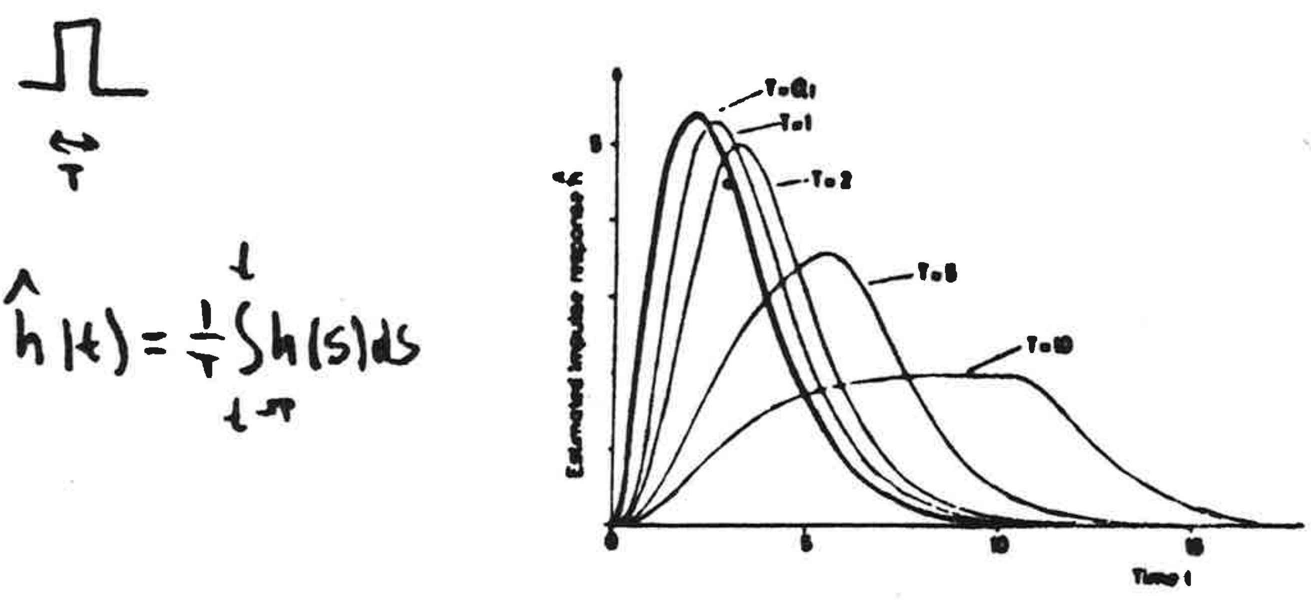
SVÅRIGHET: INITIAL TILLSTÅND

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \qquad u = \delta(t)$$

$$y(t) = \underbrace{Ce^{At}B}_{h(t)} + Ce^{At}x_0$$

POLER OK NOLLSTÄLLEN ÄNDRAS

SVÅRIGHET: EJ PERFEKTA PULSER

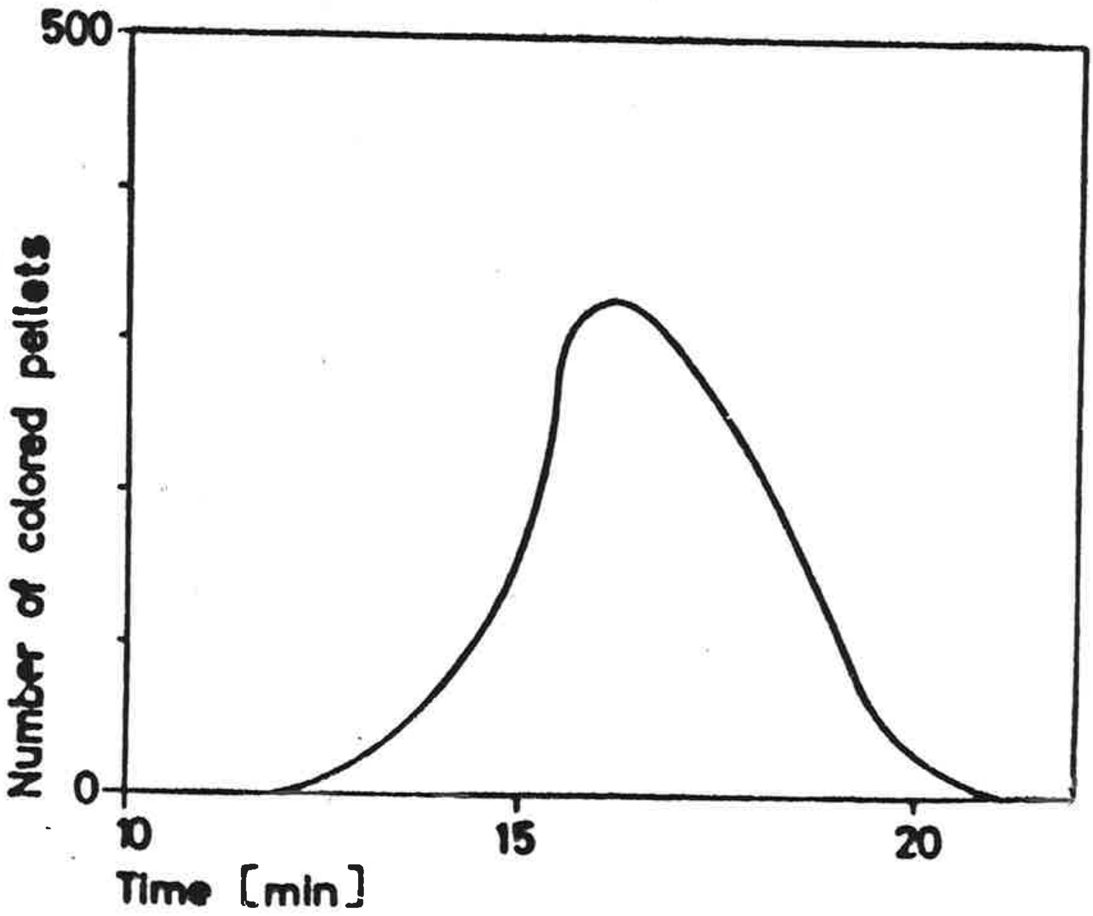


$$\hat{h}(t) = \frac{1}{T}\int_{t-T}^{t} h(s)\,ds$$

EXEMPEL

TORKTUMLARE FÖR KONSTGÖDSEL VID SUPRA

"KASTA I NÅGRA HINKAR MÄRKTA KORN TA PROVER REGELBUNDET OCH RÄKNA!"

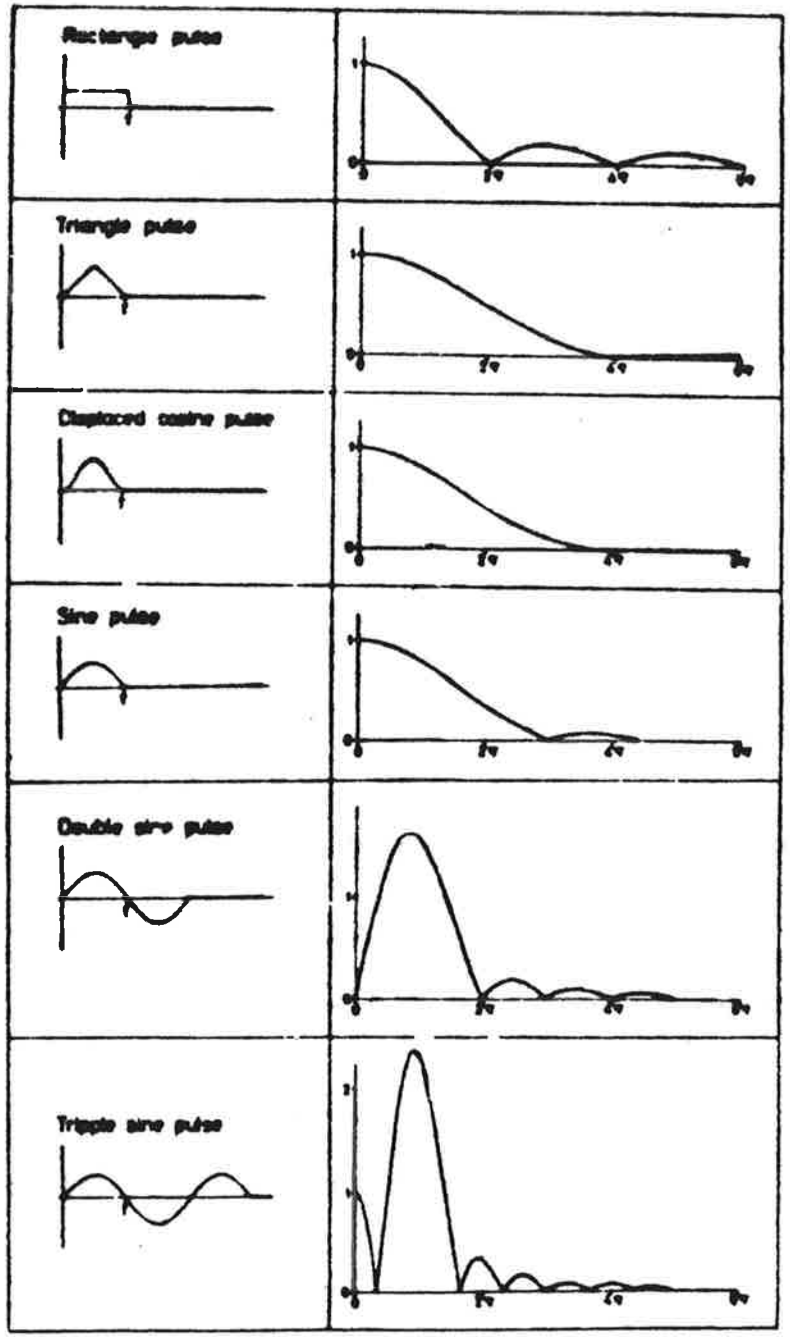


ANDRA INSIGNALER

$$\hat{G}(s) = \frac{\int_0^\infty e^{-st} y(t)\, dt}{\int_0^\infty e^{-st} u(t)\, dt}$$

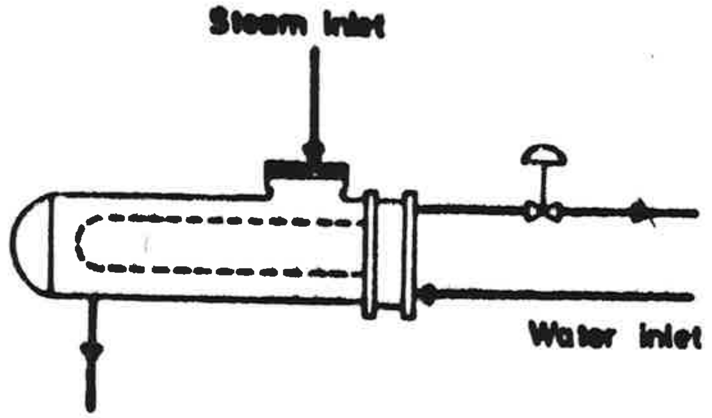
FÖRST FÖR KORREKTION FÖR PULSFORMEN

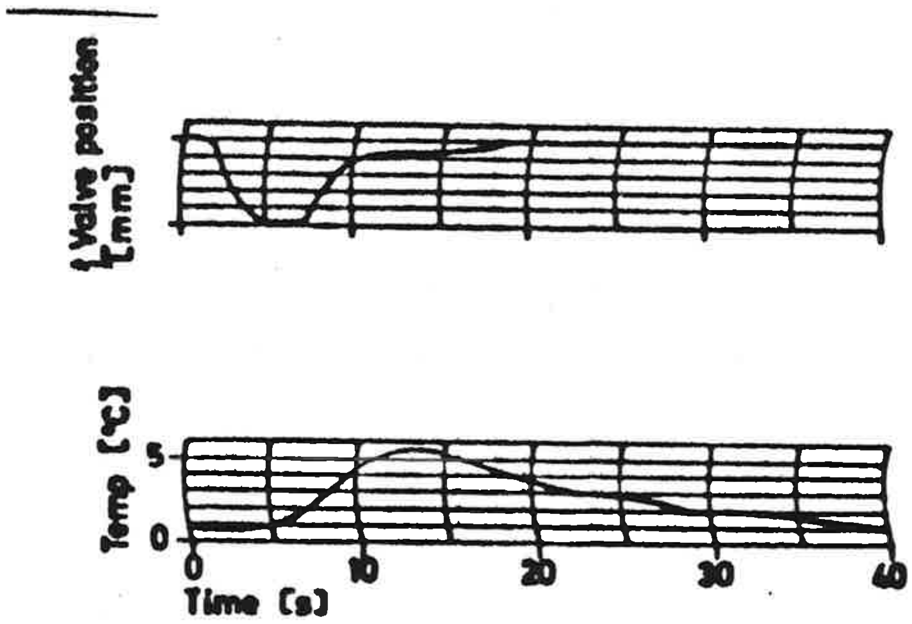
SEDAN FÖR GODTYCKLIGA INSIGNALER

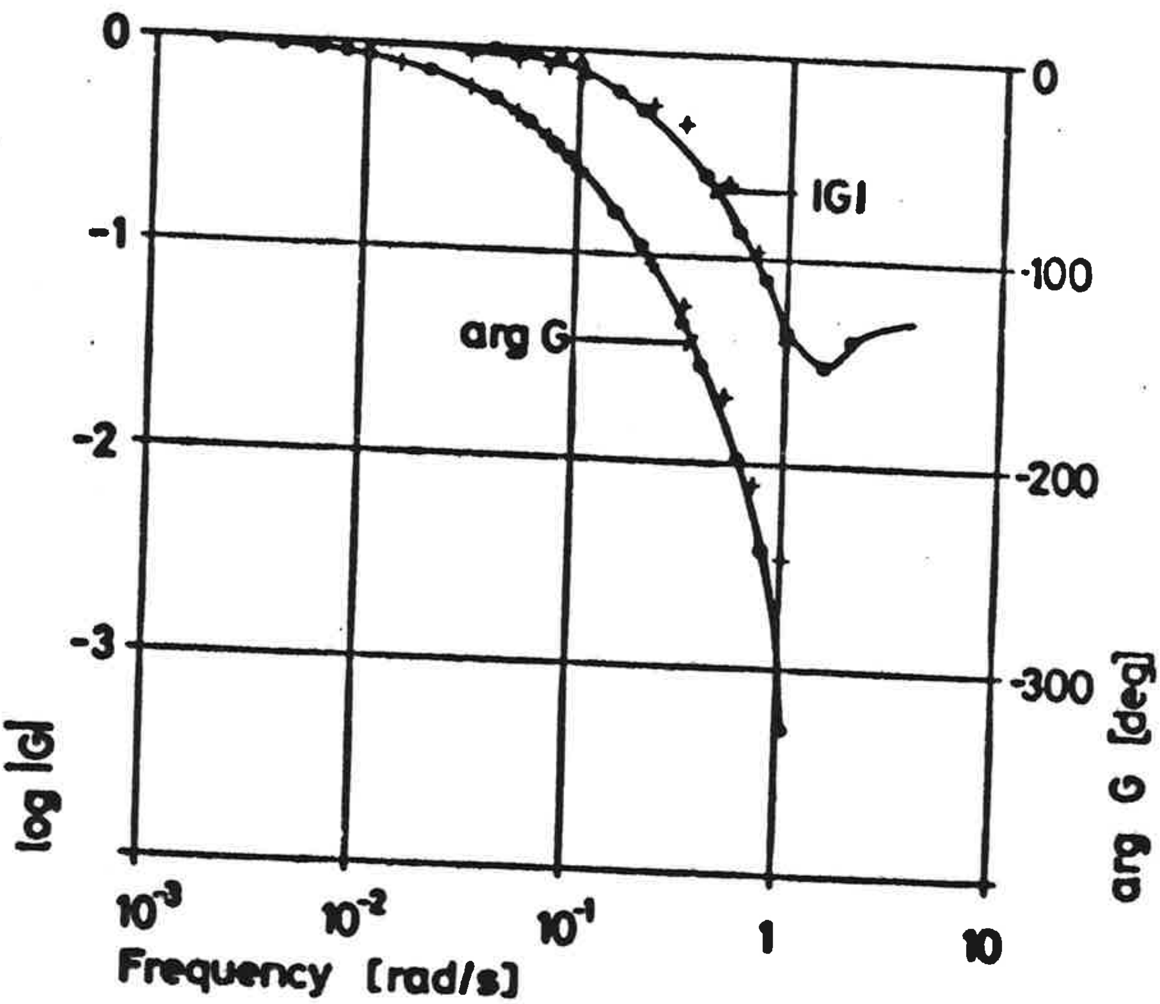


EXEMPEL 1 VÄRMEVÄXLARE

HOUGEN

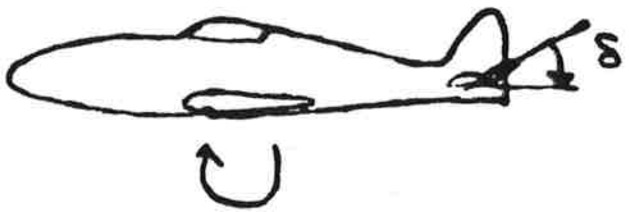


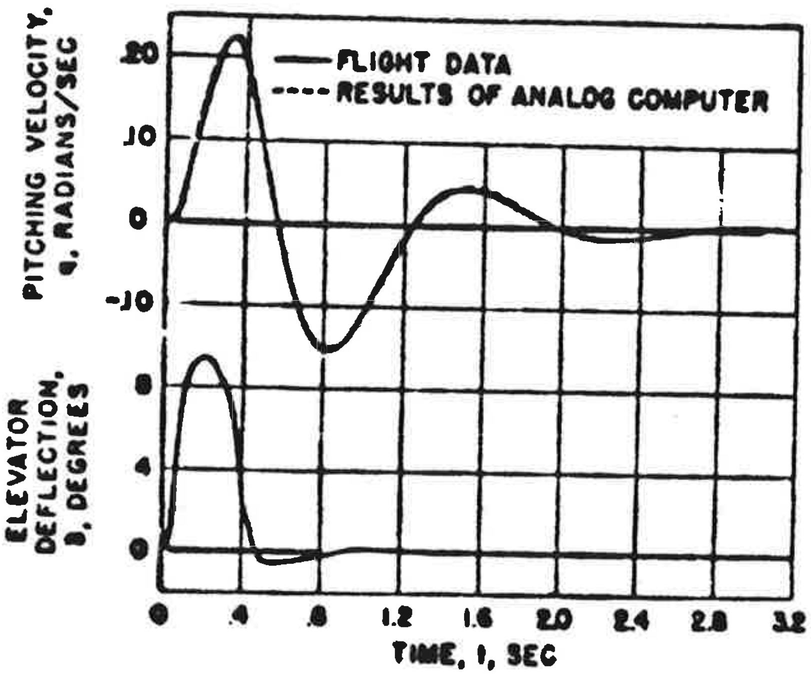


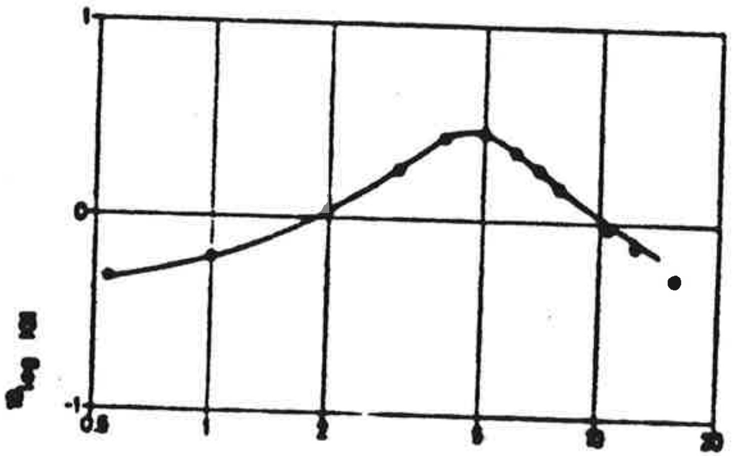


<u>Fig. 2.10</u> - Bode diagram of the transfer function of the heat exchanger. The values computed from the pulse test are marked with o and the results of the direct frequency response measurement with +.

TIPPDYNAMIK FÖR FLYGPLAN







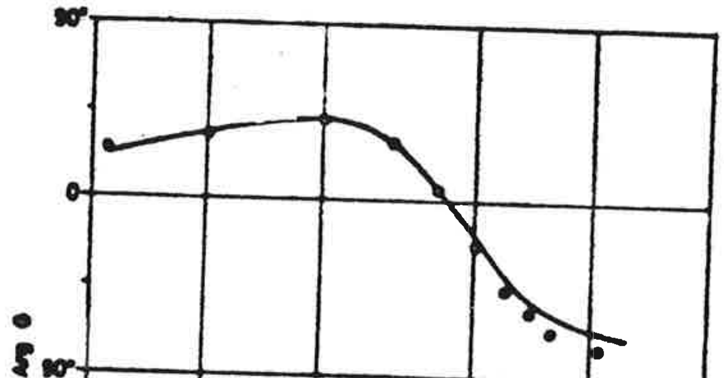
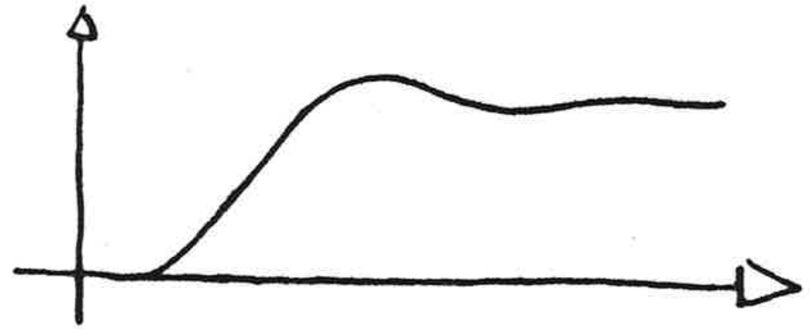
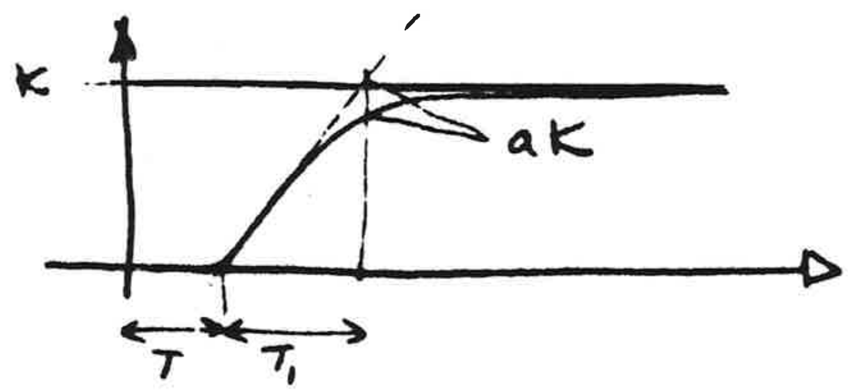


FIG. 5 TYPICAL FLIGHT RECORDS OF TRANSIENT PITCHING-VELOCITY

ANDRA INSIGNALER



STEGSVAR



APPROXIMERA MED

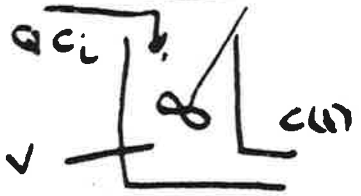
$$G(s) = K \frac{e^{-sT}}{1+sT_1}$$

BESTÄM T, T_1, K

KOLLA $a \approx 0.37$

HÖGRE ORDNINGS TAL OM a EJ STÄMMER.

VOLYMMÄTNING MHA IMPULSSVAR



$$V \frac{dc}{dt} = Q(c_i - c)$$

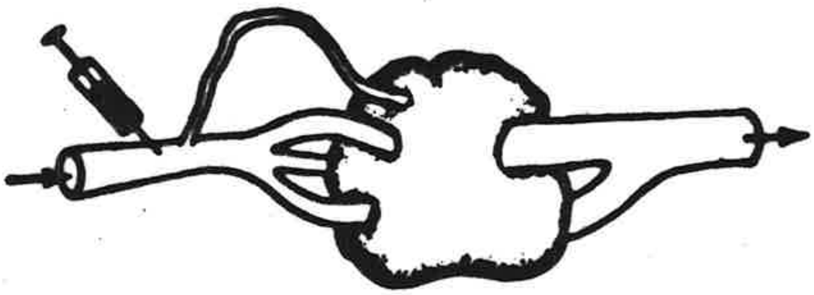
$$\frac{dc}{dt} = -\frac{Q}{V} c + \frac{Q}{V} c_i$$

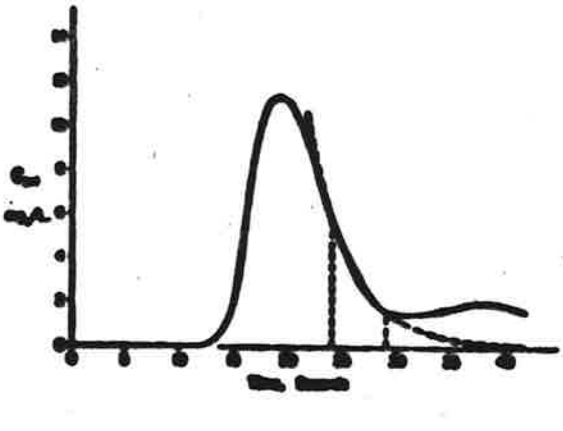
c_i impuls $\Rightarrow$ $c(t) = \frac{Q}{V} e^{-Q/V\,t} = h(t)$

Stewart-Hamiltons ekvation

$$\int_0^t t \cdot h(t) = V/Q$$

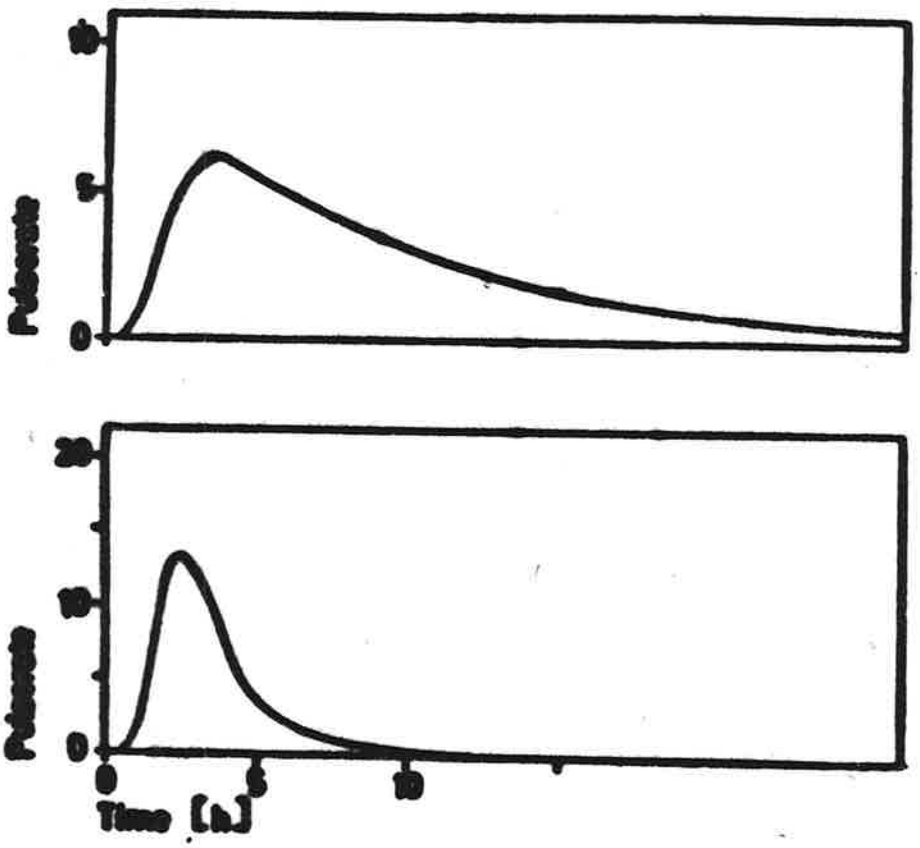
Exempel Blodvolymsmätning





Undersökning av blandning

Massakar



4omr.

2omr

	V/Q
2 omrörare	3.5 tim
4 omrörare	10.6
perf. blandn	11.7

SAMMANFATTNING

* TRANSIENTANALYS ÄR MYCKET ENKEL OCH SNABB ATT UTFÖRA (KRÄVER EJ SPEC INSTRUMENT)
* IMPULSSVARET ERHÅLLES DIREKT
* ENKLA MODELLER KAN LÄTT ANPASSAS FÖR HAND (BRA DÅ MAN ÄR TRÅNGD)
* INITIALVÄRDEN ≠ 0 PROBLEM
* NUMERISK LAPLACE TRANSFORMERING
* SPECIELLA MÄTMETODER BYGGER PÅ TRANSIENTANALYS
* KÄNSLIG FÖR STÖRNINGAR

NÅGRA VIKTIGA IDÉER VI SETT

- EXCITATION
- IDENTIFIERBARHET
- GODTYCKLIGA INSIGNALER
- SPECIELLA MÄTMETODER

4. KORRELATIONS- OCH SPEKTRALANALYS

Ivar Gustavsson

En bra referens för korrelations- och spektralanalys är

Jenkins G M and D G Watts: Spectral Analysis and its Applications. Holden Day 1969.

En del av illustrationerna är hämtade ur denna bok.

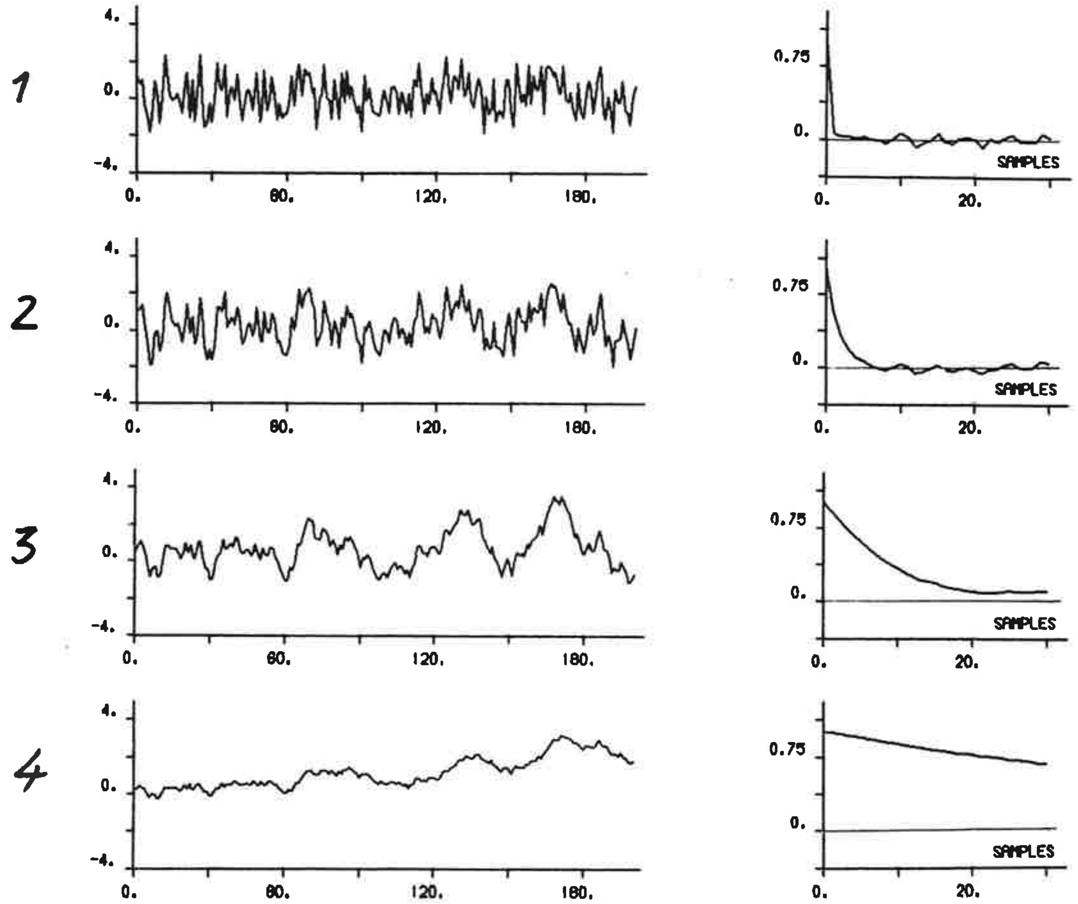
KORRELATIONS- OCH SPEKTRALANALYS

1. KOVARIANSFUNKTIONER
2. SPEKTRALTÄTHET
3. SKATTNING AV ÖVERFÖRINGSFUNKTIONER

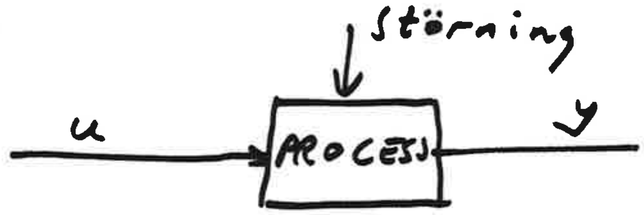
PROBLEM 1: HUR KARAKTERISERA SIGNALER (STOK. PROC)?

<u>TIDSSERIEANALYS</u>

SIGNAL: AUTOKORRELATIONSFUNKTION



PROBLEM 2:



HUR BEROR Y PÅ U? <u>IDENTIFIERING</u>

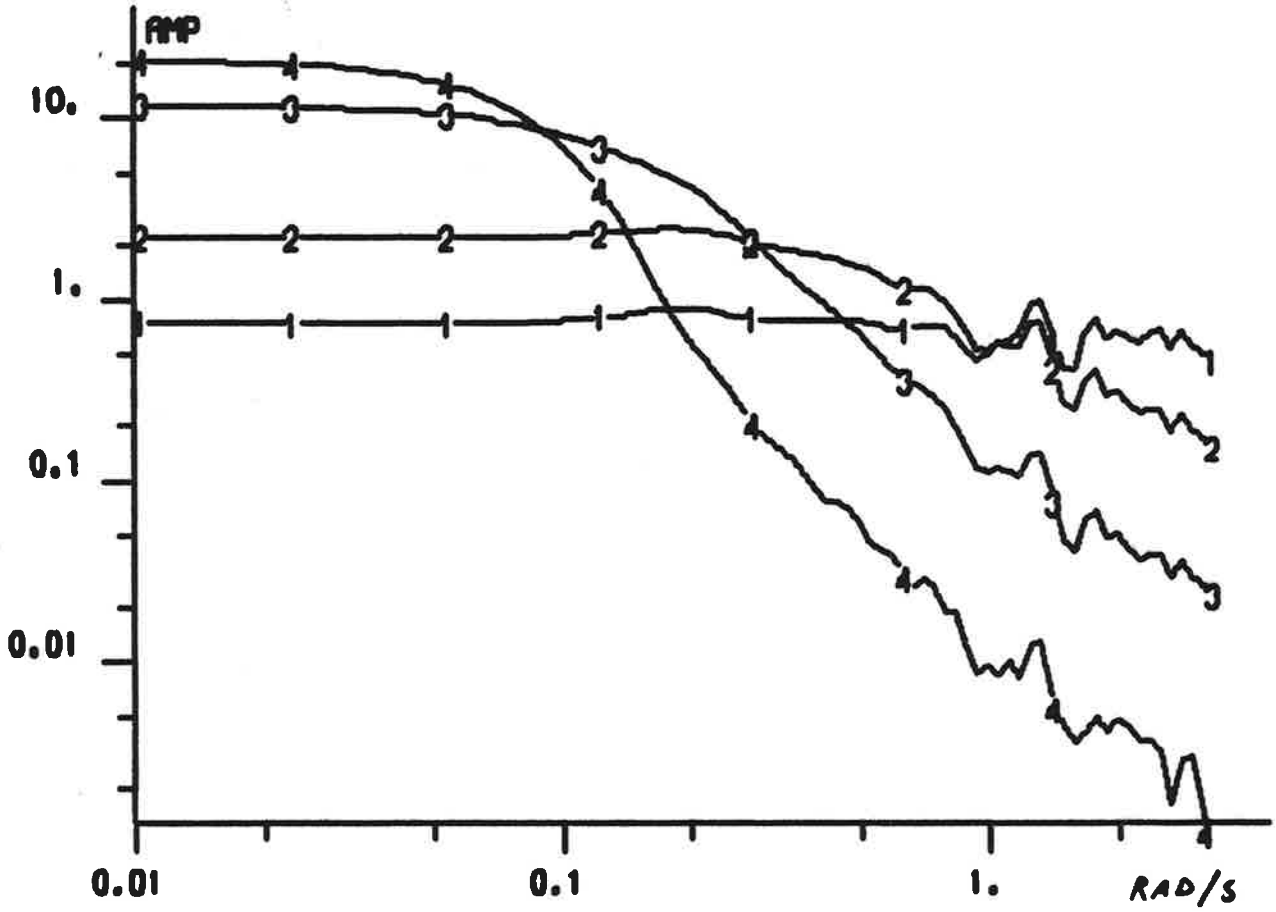
LÖSNING:

(i) KORRELATIONS- OCH SPEKTRALANALYS (*)

(ii) PARAMETRISKA METODER

(*) Jfr TRANSIENT- OCH FREKVENSANALYS

EFFEKTSPEKTRUM:



KOVARIANSFUNKTIONEN

DEFINITION

KORSKOVARIANS: $r_{uy}(\tau) = E\, u(t)\, y(t+\tau)$

AUTOKOVARIANS: $r_{uu}(\tau) = r_u(\tau)$

OBS ALTERNATIV

$$r_{uy}^{*}(\tau) = E\, u(t+\tau)\, y(t) = r_{uy}(-\tau)$$

EGENSKAPER

$r_u(\tau) = r_u(-\tau)$ SYMMETRI

$|r_u(\tau)| \leq r_u(0)$

$\sum z_i z_j\, r(t_i - t_j) \geq 0$ ICKE NEGATIV

KORRELATIONSFUNKTIONEN

KORSKORRELATION $\dfrac{r_{uy}(\tau)}{\sqrt{r_u(0) \cdot r_y(0)}}$

AUTOKORRELATION $\dfrac{r_u(\tau)}{r_u(0)}$

KOVARIANSFUNKTIONEN KONTINUERLIG TID

SKATTNING

$$c(\tau) = \frac{1}{T}\int_0^{T-\tau} u(t)\,u(t+\tau)\,dt$$

MEDELVÄRDE

$$E\,c(\tau) = \left(1-\frac{\tau}{T}\right) r(\tau) \to r(\tau), \quad T \to \infty$$

VARIANS

$$E\left[c(\tau) - r(\tau)\right]^2 = \frac{1}{T^2}\int_{-(T-\tau)}^{T-\tau} (T-\tau-|s|)\left[r^2(s) + r(s+\tau)r(s-\tau)\,ds\right]$$

$$\approx \frac{4}{T^2}\int_0^{\infty} r^2(t)\,dt$$

EXEMPEL:

$$r(t) = e^{-\alpha t}$$

$$Var\left[c(t)\right] = \frac{4}{T}\int_0^{\infty} e^{-2\alpha t}\,dt = \frac{2}{\alpha T}$$

αT	200	20000
σ	0.1	0.01

KOVARIANSFUNKTIONEN

DISKRET TID

SKATTNING

$$c(\tau) = \frac{1}{n} \sum_{1}^{n-\tau} u(t)\, u(t+\tau)$$

MEDELVÄRDE

$$E\, c(\tau) = \left(1 - \frac{\tau}{n}\right) r(\tau) \to r(\tau), \quad n \to \infty$$

KOVARIANS

$$\mathrm{cov}[c(\tau_1), c(\tau_2)] \sim$$

$$\sim \frac{1}{n} \sum_{-\infty}^{\infty} [r(s)\, r(s+\tau_2-\tau_1) + r(s+\tau_2)\, r(s-\tau_2)]$$

$$\mathrm{Var}\, [c(\tau)] \sim \frac{2}{n} \left[r^2(0) + 2 \sum_{1}^{n} r^2(s)\right]$$

SKATTNING AV KORRELATIONS-FUNKTIONER

* $\frac{1}{T}$ vs $\frac{1}{T-\tau}$

* LÅGFREKVENTA STÖRNINGAR

* STARK KORRELATION MELLAN INTILLIGGANDE SKATTNINGAR

* LÅNGA TIDER $\kappa T = 200 \Rightarrow 10\%$

4:8

$$c'(\tau) = \frac{1}{T-\tau} \int_0^{T-\tau} x(t+\tau)\,x(t)\,dt$$

(unbiased)

$$c(\tau) = \frac{1}{T} \int_0^{T-\tau} x(t+\tau)\,x(t)\,dt$$

(asymp. unbiased)

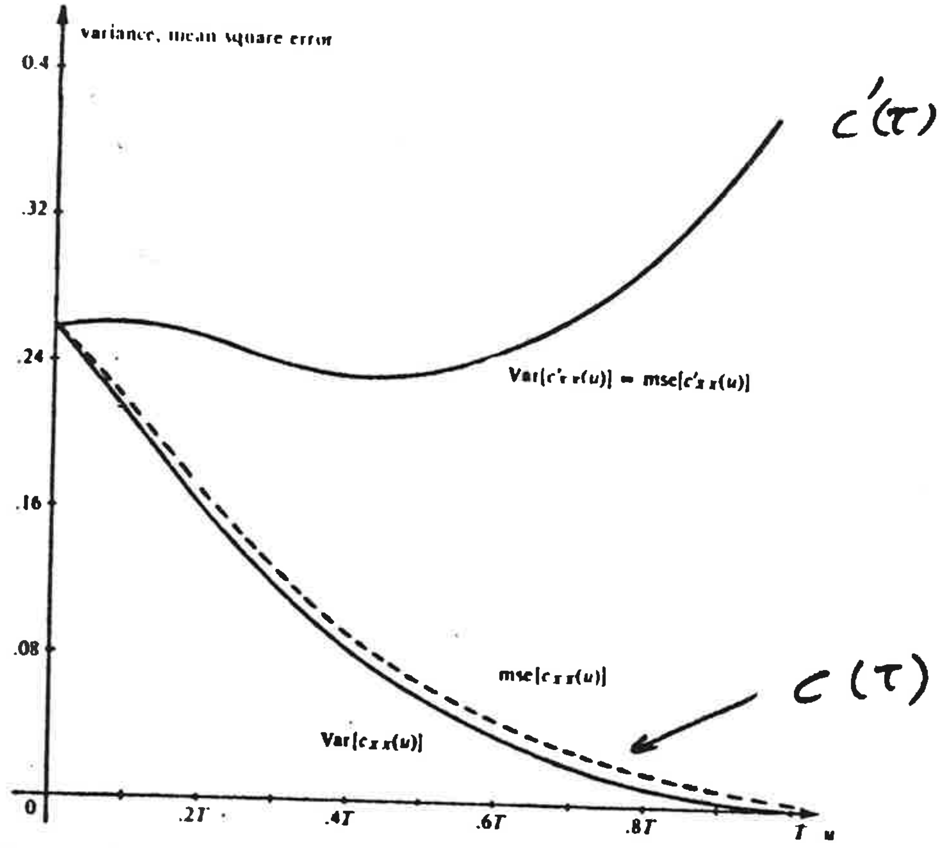
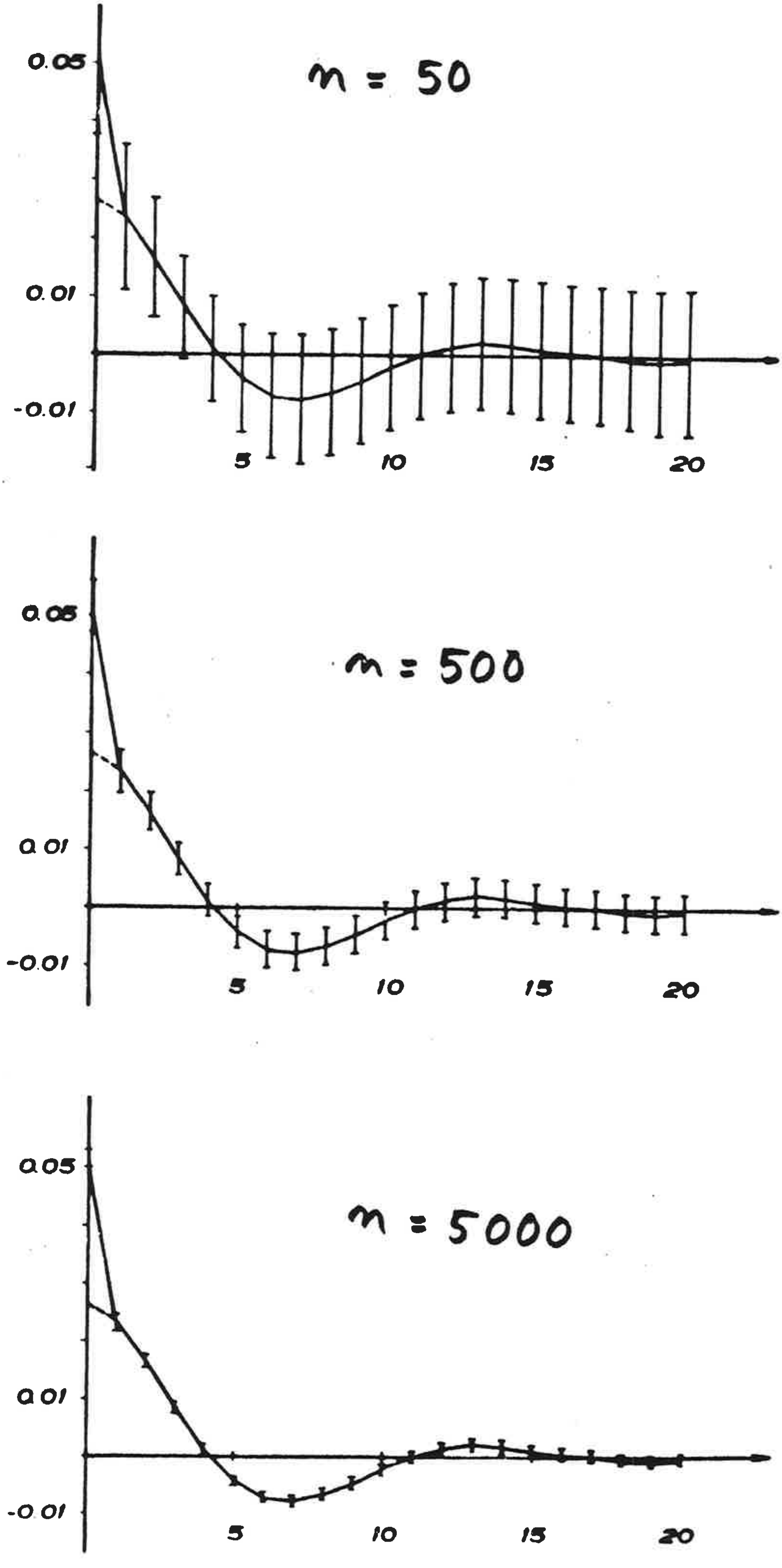


FIG. 5.12: Variances and mean square errors of autocovariance function estimators for a continuous first-order process



NOGGRANNHET AV SKATTNINGEN AV KOVARIANSFUNKTIONEN FÖR OLIKA ANTAL DATA

4:10

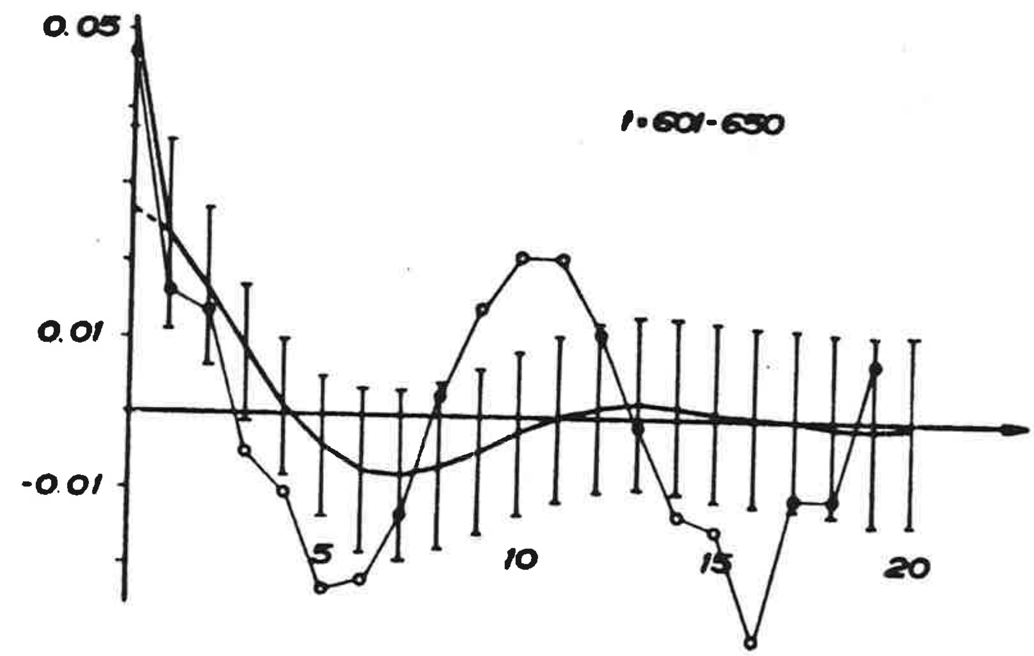


Figure 3.3 B

Sample covariance function calculated from 50 values of the artificial time series. The solid line shows the true covariance function.

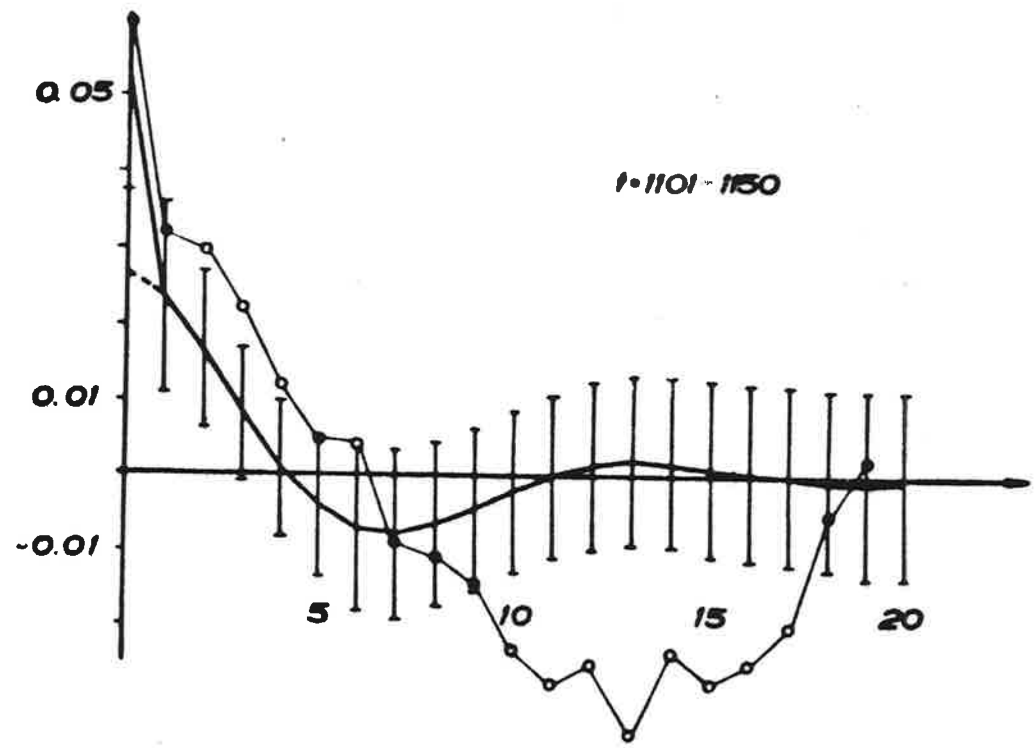


Figure 3.3 C

Sample covariance function calculated from 50 values of the artificial time series. The solid line shows the true covariance function.

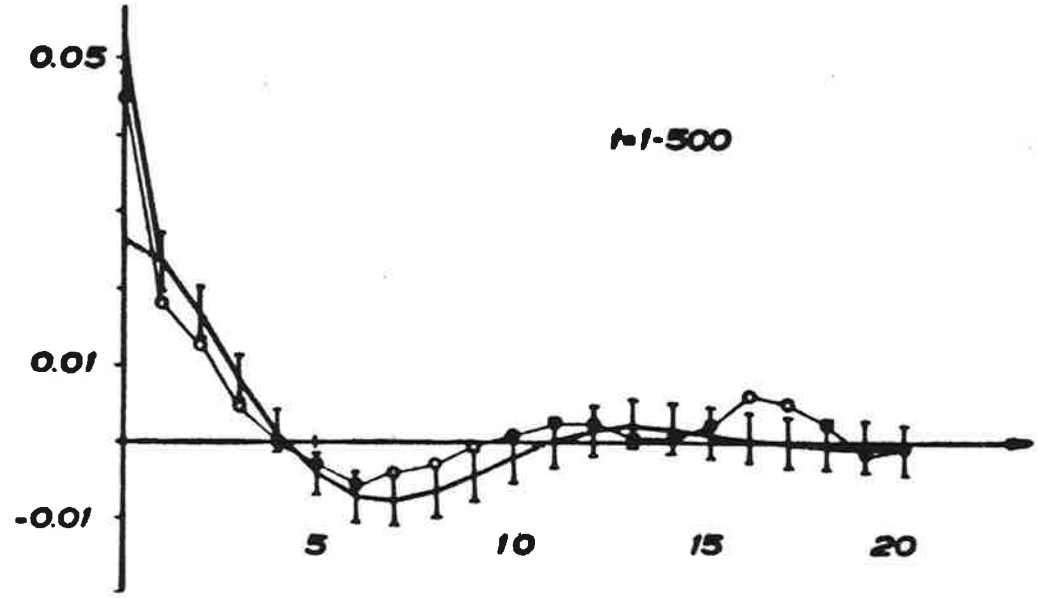


Figure 3.1 A

Sample covariance function calculated from 500 values of the artificial time series. The solid line shows the true covariance function.

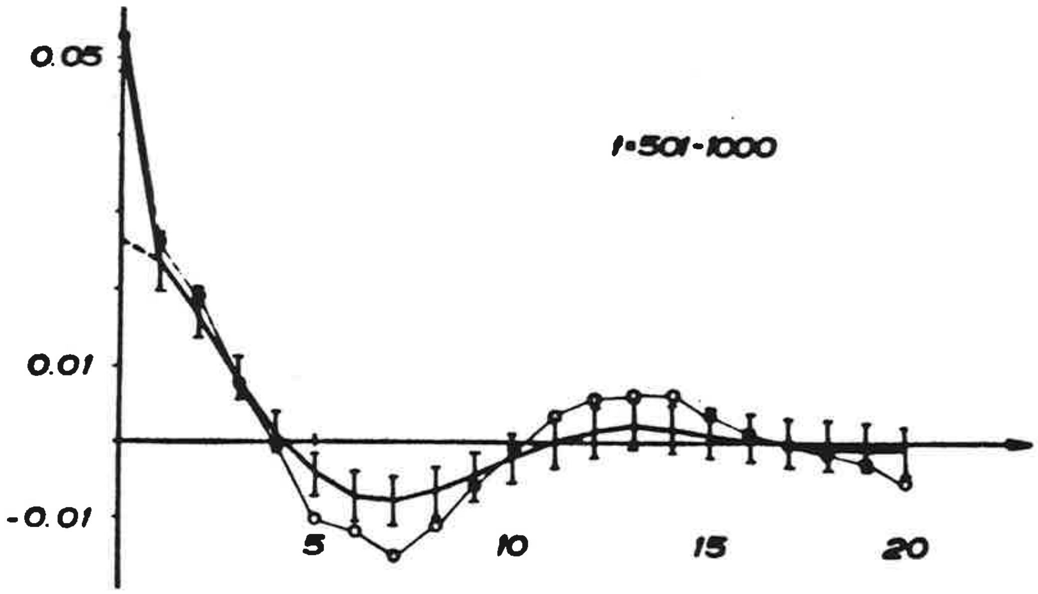


Figure 3.1 B

Sample covariance function calculated from 500 values of the artificial time series. The solid line shows the true covariance function.

SPEKTRAL TÄTHETS SKATTNINGAR

$$\phi(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega t} r(t)\,dt$$

$$\phi(\omega) = \frac{1}{2\pi} \sum_{m=-\infty}^{\infty} e^{-i\omega m} r(m)$$

PERIODOGRAM ELLER SAMPLE SPEKTRUM

$$\hat{\phi} = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega t} c(t)\,dt$$

$$= \frac{2\pi}{T} \left| \frac{1}{2\pi} \int_{-T/2}^{T/2} e^{-i\omega t} x(t)\,dt \right|^2$$

EGENSKAPER

$$\frac{2\hat{\phi}}{\phi} \sim \chi^2(2)$$

$\hat{\phi}(\omega_1) \sim$ OKORRELERAD MED $\hat{\phi}(\omega_2)$

4: 13

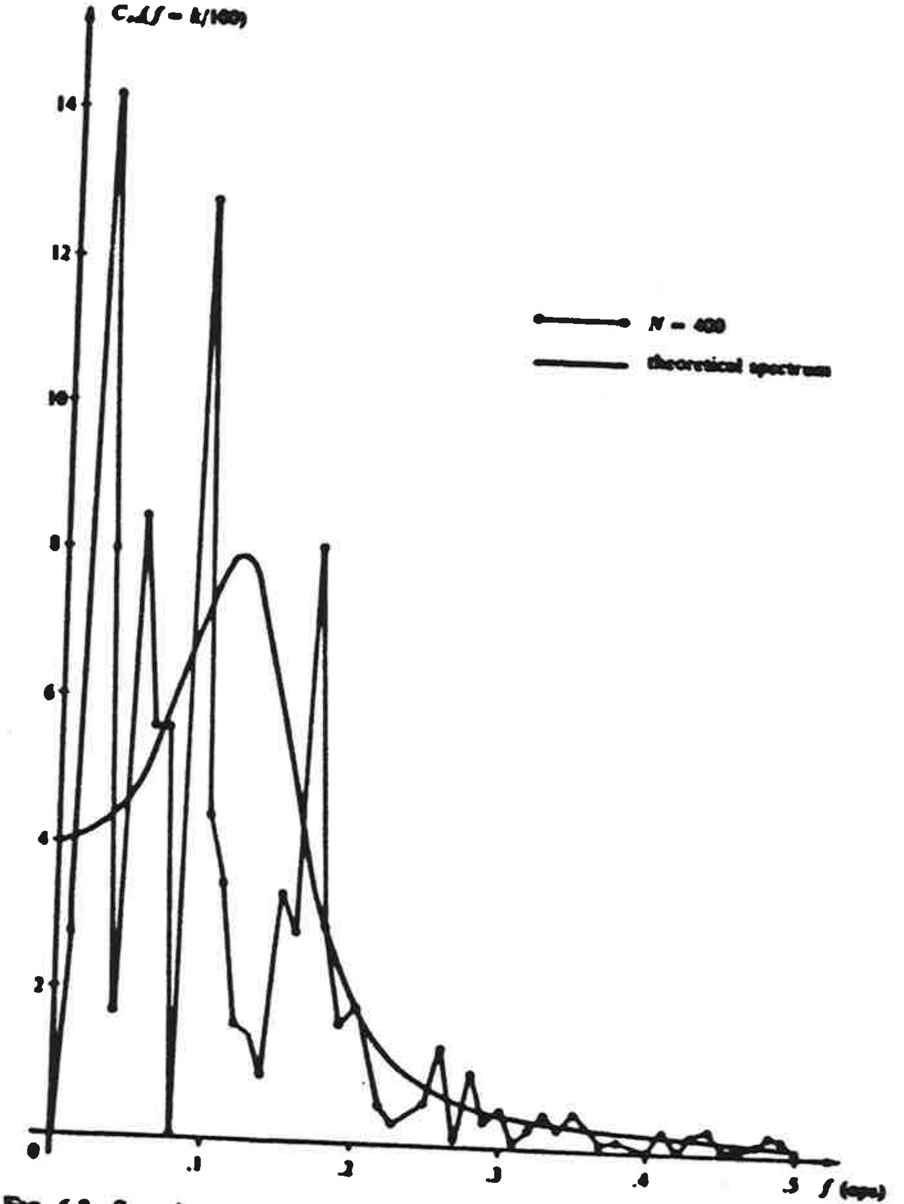


FIG. 6.2: Sample spectrum for a realization of a second-order autoregressive process

FÖNSTER

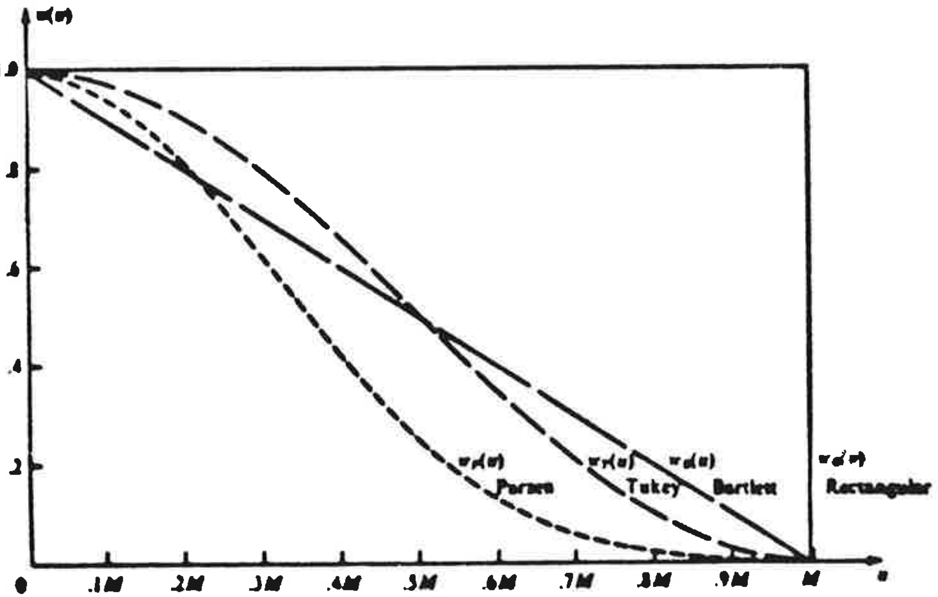


Fig. 6.12: Some common lag windows

TIDS FÖNSTER

LAG WINDOWS

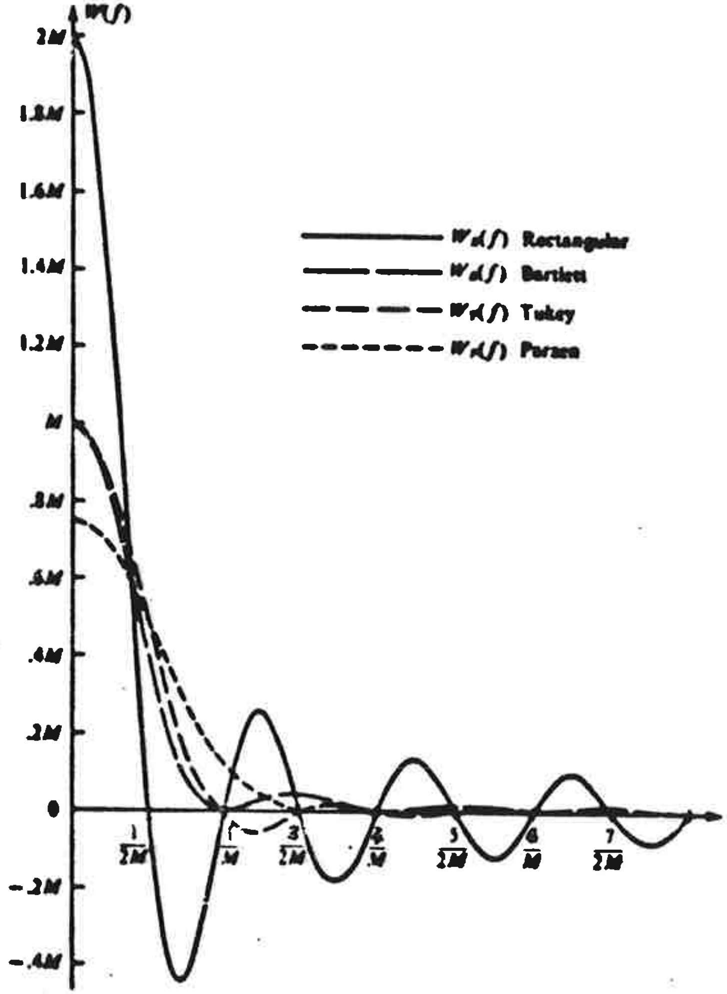


Fig. 6.13: Some common spectral windows

SPEKTRALFÖNSTER

SPECTRAL WINDOW

4:15

TABLE 6.5: Lag and spectral windows

Description	Lag window	Spectral window
rectangular	$w_R(u) = \begin{cases} 1, & \lvert u\rvert \leq M \\ 0, & \lvert u\rvert > M \end{cases}$	$W_R(f) = 2M\left(\frac{\sin 2\pi fM}{2\pi fM}\right), \quad -\infty \leq f \leq \infty$
Bartlett	$w_B(u) = \begin{cases} 1 - \frac{\lvert u\rvert}{M}, & \lvert u\rvert \leq M \\ 0, & \lvert u\rvert > M \end{cases}$	$W_B(f) = M\left(\frac{\sin \pi fM}{\pi fM}\right)^2, \quad -\infty \leq f \leq \infty$
Tukey	$w_T(u) = \begin{cases} \frac{1}{2}\left(1 + \cos\frac{\pi u}{M}\right), & \lvert u\rvert \leq M \\ 0, & \lvert u\rvert > M \end{cases}$	$W_T(f) = M\left\{\frac{\sin 2\pi fM}{2\pi fM} + \frac{1}{2}\frac{\sin 2\pi M(f + \frac{1}{2}M)}{2\pi M(f + \frac{1}{2}M)} + \frac{1}{2}\frac{\sin 2\pi M(f - \frac{1}{2}M)}{2\pi M(f - \frac{1}{2}M)}\right\} = M\left(\frac{\sin 2\pi fM}{2\pi fM}\right)\left(\frac{1}{1 - (2fM)^2}\right), \quad -\infty \leq f \leq \infty$
Parzen	$w_P(u) = \begin{cases} 1 - 6\left(\frac{u}{M}\right)^2 + 6\left(\frac{\lvert u\rvert}{M}\right)^3, & \lvert u\rvert \leq \frac{M}{2} \\ 2\left(1 - \frac{\lvert u\rvert}{M}\right)^3, & \frac{M}{2} < \lvert u\rvert \leq M \\ 0, & \lvert u\rvert > M \end{cases}$	$W_P(f) = \frac{3}{4}M\left(\frac{\sin \pi fM/2}{\pi fM/2}\right)^4, \quad -\infty \leq f \leq \infty$

TABLE 6.6: Properties of spectral windows

Description	Spectral window	Variance ratio I/T	Degrees of freedom	Standardized bandwidth b_1
rectangular	$2M\frac{\sin 2\pi fM}{2\pi fM}$	$2\frac{M}{T}$	$\frac{T}{M}$	0.5
Bartlett	$M\left(\frac{\sin \pi fM}{\pi fM}\right)^2$	$0.667\frac{M}{T}$	$3\frac{T}{M}$	1.5
Tukey	$M\left(\frac{\sin 2\pi fM}{2\pi fM} \times \frac{1}{1 - (2fM)^2}\right)$	$0.75\frac{M}{T}$	$2.667\frac{T}{M}$	1.333
Parzen	$\frac{3}{4}M\left(\frac{\sin(\pi fM/2)}{\pi fM/2}\right)^4$	$0.539\frac{M}{T}$	$3.71\frac{T}{M}$	1.86

4:16

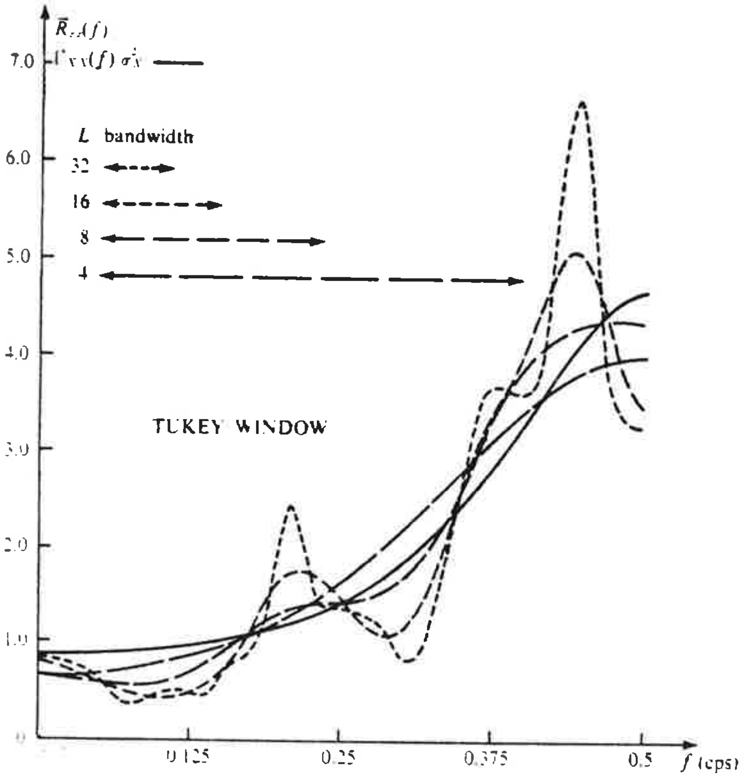


FIG. 7.4: Smoothed spectral density estimates for a first-order ar process ($\alpha_1 = -0.4$; $N = 100$)

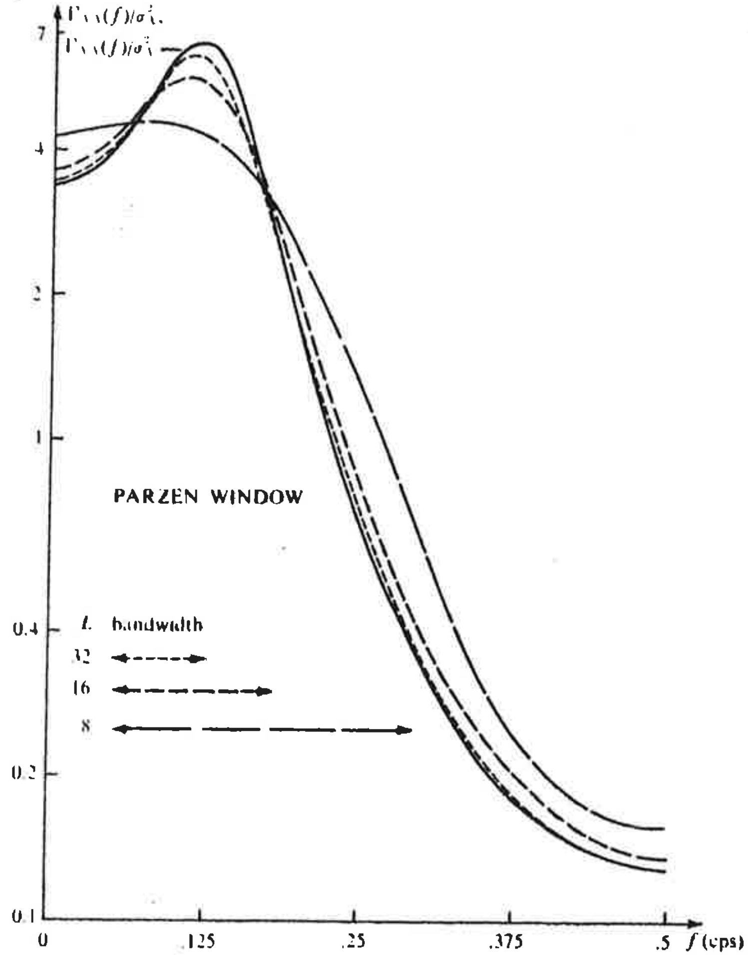


FIG. 7.7: Mean smoothed spectral density functions for a second-order ar process ($\alpha_1 = 1.0$, $\alpha_2 = -0.5$)

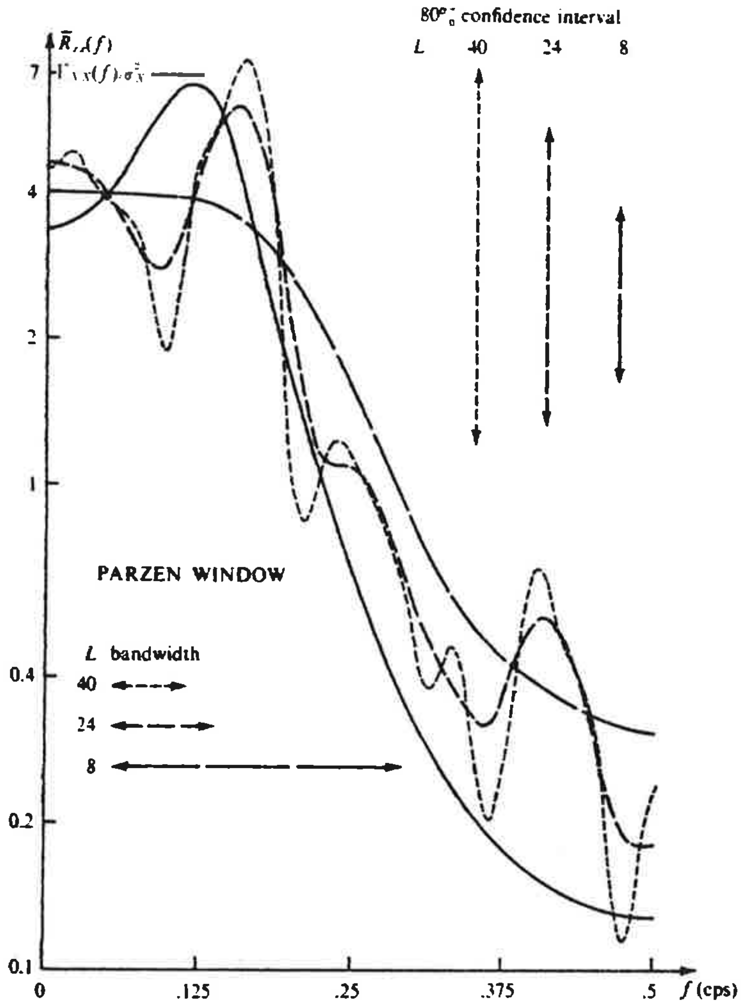


FIG. 7.8: Smoothed spectral density estimates for a second-order ar process ($\alpha_1 = 1.0, \alpha_2 = -0.5; N = 50$)

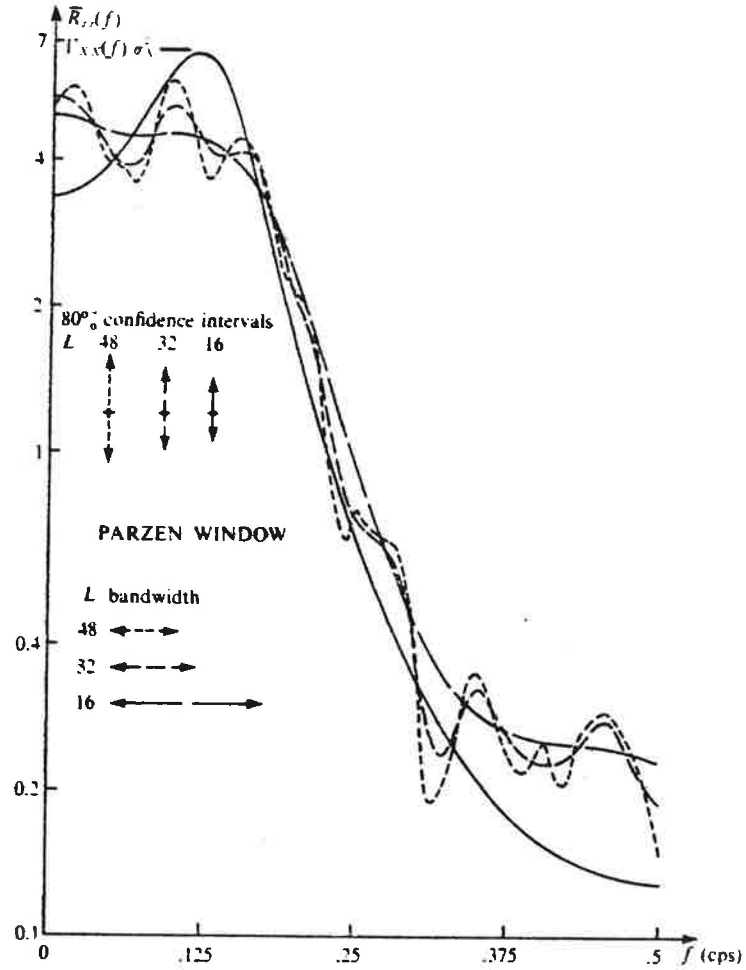


FIG. 7.9: Smoothed spectral density estimates for a second-order ar process ($\alpha_1 = 1.0, \alpha_2 = -0.5; N = 400$)

4/17

STATISTISKA EGENSKAPER

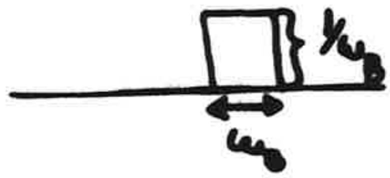
$$E\{\hat{\phi}(\omega)\} \approx \phi(\omega)$$

$$\mathrm{Var}\{\hat{\Phi}(\omega)\} \approx \phi^2(\omega)\,\frac{1}{T}\int_{-\infty}^{\infty} w^2(u)\,du$$

tids fönstret

$$\int_{-\infty}^{\infty} w^2(u)\,du = 2\pi\int_{-\infty}^{\infty} W^2(\omega)\,d\omega \quad \text{Parseval}$$

$$= \frac{2\pi}{\omega_B} = \frac{1}{f_B}$$



$$\mathrm{Var}\{\hat{\Phi}(\omega)\} = \Phi^2(\omega)\cdot\frac{1}{f_B T}$$

VARIANCE × BANDBREDD = $\phi^2(\omega)/T$

$$\frac{\hat{\Phi}(\omega)}{\phi(\omega)} \sim \chi^2(\nu), \quad \nu = 2T f_B$$

Hz

PRAKTISKA SYNPUNKTER

SAMPLINGSINTERVALL

$$h \leq \frac{1}{2f_0}$$

FREKVENSINTERVALL $0 \leq f \leq f_0$

ALIASING

ANTAL LAGS

$$m = \frac{b_1}{ah}$$

"STANDARDISERAD" FÖNSTER-BANDBREDD b_1

SMALAST TOPP AV BETYDELSE a

ANTAL DATA

$$T = \frac{\nu}{2a}$$

$$N = \frac{\nu}{2ah}$$

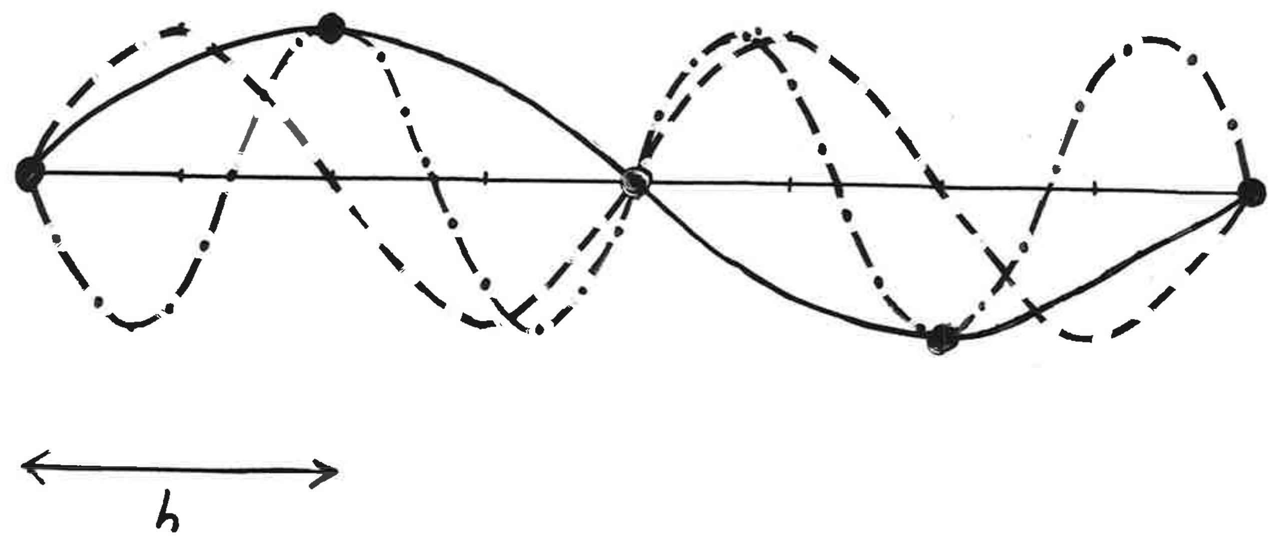
MÄTTID T

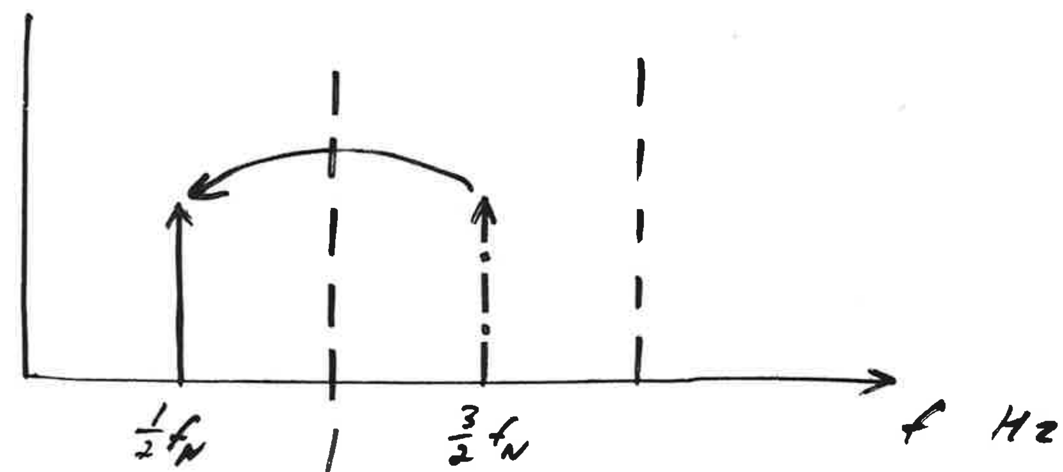
ANTAL FRIHETSGRADER ν

$$\varepsilon^2 = \frac{m}{N}$$

$$\varepsilon = \frac{s.d\,[\hat{\Phi}_{uy}(f)]}{\sqrt{\Phi_u(f) \cdot \Phi_y(f)}}$$

ALIASING





$f_N = \frac{1}{2h}$

Nyquistfrekvensen

NAMN

LAG WINDOW

SPECTRAL WINDOW

FIDELITY (BIAS)

STABILITY (VARIANS)

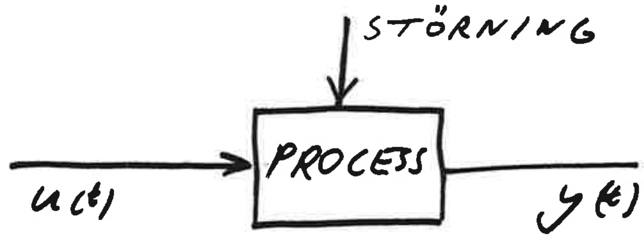
RESOLUTION

WINDOW CLOSING

WINDOW CARPENTRY

4:22

SKATTNING AV ÖVERFÖRINGSFUNKTIO



$h(t)$ viktfunktion

$G(s)$ överföringsfunktion

$$r_{uy}(t) = \int_0^\infty h(s)\, r_u(t-s)\, ds$$

$$\phi_{uy}(i\omega) = G(i\omega)\cdot\phi_u(\omega) \qquad (*)$$

IDÉ: SKATTA ϕ_u OCH ϕ_{uy} OCH BESTÄM G UR (*)

FÖRUTSÄTTNING: $\{u(t)\}$ $\{y(t)\}$ STATIONÄRA

ALTERNATIV:

$$\hat{G}(i\omega) = \frac{\hat{Y}(i\omega)}{\hat{U}(i\omega)}$$

$$\hat{Y}(s) = \int_0^T e^{-st} y(t)\, dt$$

TOLKNING

BESTÄM h SÅ ATT

$$I[h] = \int_0^T \left[y(t) - \int_{-\infty}^{t} h(t-s)\, u(s)\, ds \right]^2 dt \qquad \text{MIN}$$

$$\text{Min } I[h] = T \int_{-\infty}^{\infty} \phi_y(\omega) \left[1 - \gamma_{uy}^2(\omega) \right] d\omega$$

$$\gamma_{uy}^2(\omega) = \frac{|\phi_{uy}(\omega)|^2}{\phi_u(\omega)\, \phi_y(\omega)}$$

KOHERENSFUNKTION

OM ϕ_u, ϕ_y OCH ϕ_{uy} ÄR SAMPLESPEKTRA SÅ ÄR KOHERENSEN 1.

KONSEKVENS 1:

$$\hat{G}(i\omega) = \frac{\hat{\phi}_{uy}(i\omega)}{\hat{\phi}_u(\omega)} = \frac{\hat{Y}(i\omega)}{\hat{U}(i\omega)}$$

KONSEKVENS 2:

SKATTNINGEN MOTSVARAR EJ SÄKERT EN KAUSAL ÖVERFÖRINGS-FUNKTION

VITT BRUS

$$\hat{h}(\tau) = \frac{1}{T} \int_0^{T-\tau} u(t)\, y(t+\tau)\, dt$$

PRBS

VIT INSIGNAL GENOM SPEC. EXPERIMENTVAL

FÖRFILTRERING (PRE-WHITENING)

$$\Phi_{uy}(i\omega) = G(i\omega) \cdot \Phi_u(\omega)$$

INFÖR

$$v = Fu$$

SÅ ATT

$$\Phi_v(\omega) = |F|^2 \Phi_u(\omega) = 1$$

BERÄKNA SEDAN

$$z = Fy$$

$$\Phi_{zv}(i\omega) = |F|^2 \Phi_{uy}(i\omega) = |F|^2 G(i\omega)\, \Phi_u(\omega) =$$

$$= G(i\omega) \cdot 1$$

VIT INSIGNAL GENOM BERÄKNING!

4: 25

SYSTEM: $Y(K) = U(K-2) + 0.1E(K)$, 500 DATA

PREWHITENED

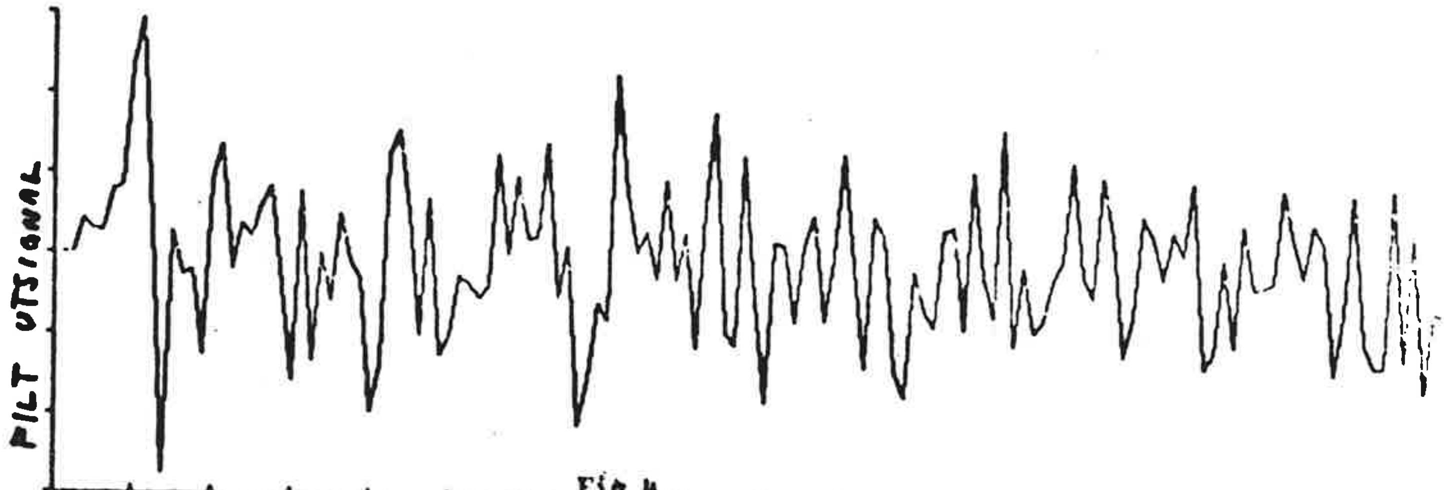


Fig 4.

OUTPUT

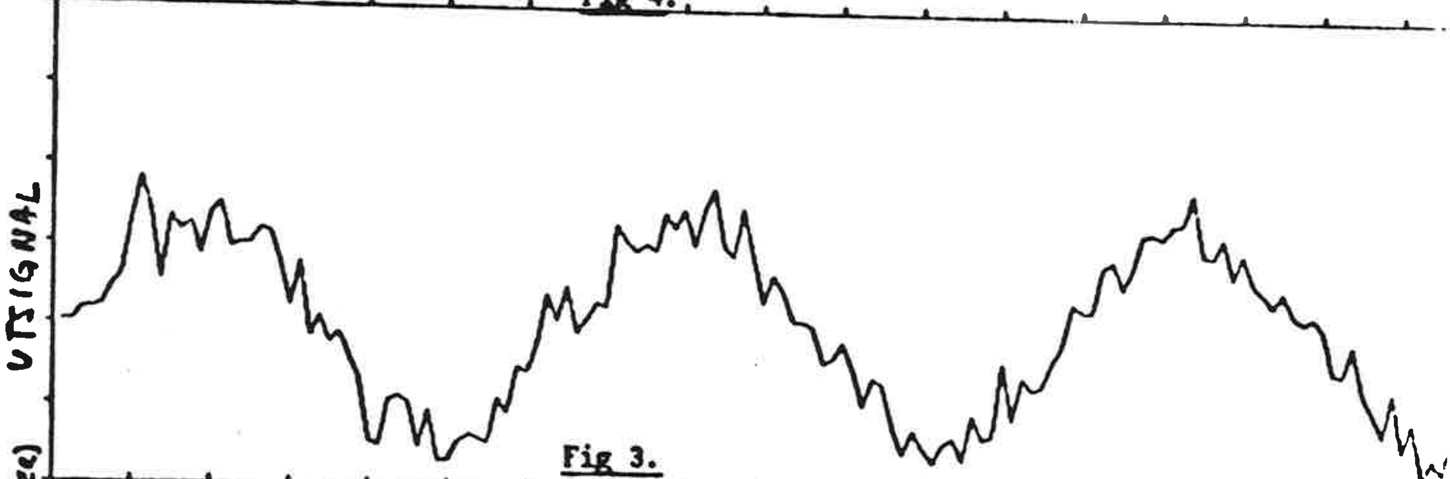


Fig 3.

PREWHITENED

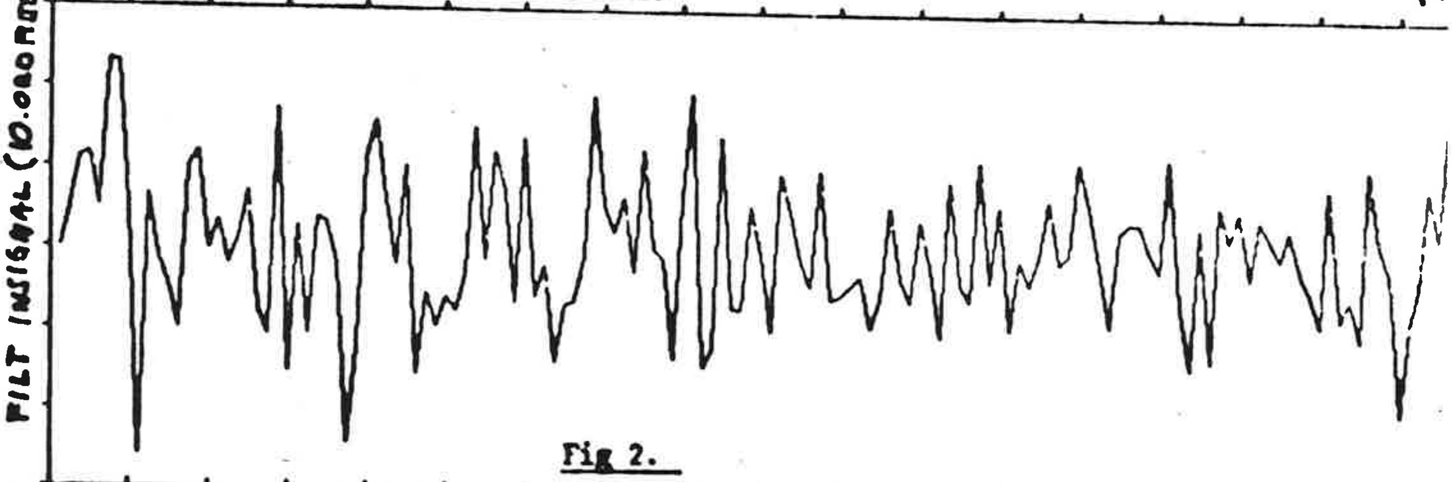


Fig 2.

INPUT

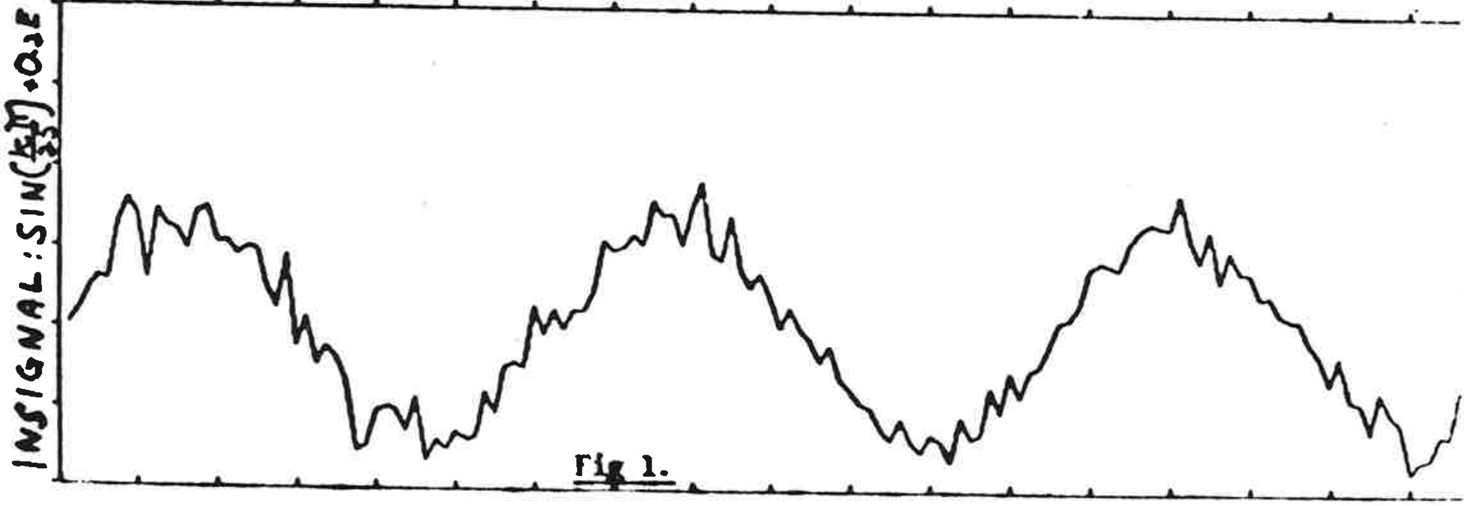
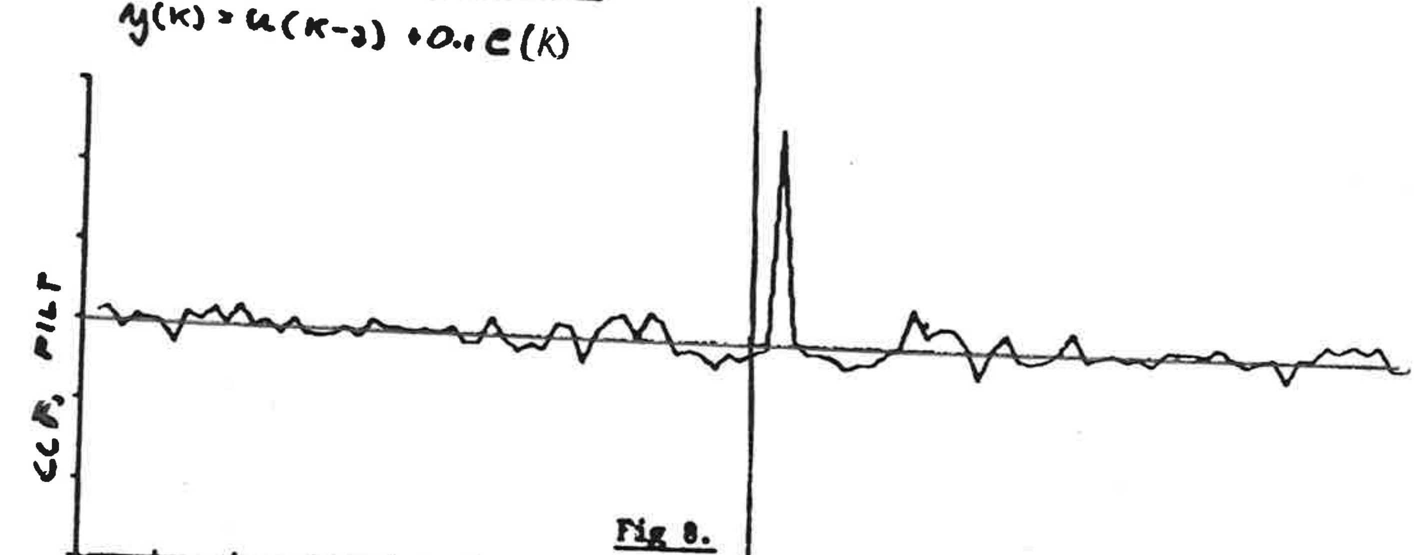


Fig 1.

4:26

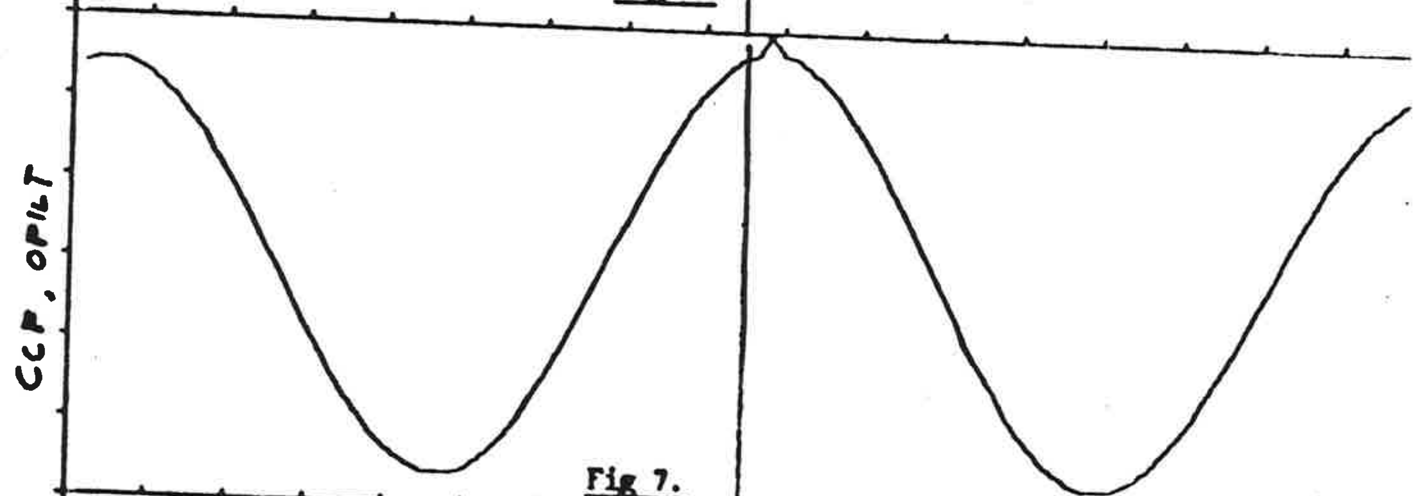
SIMULERAT SYSTEM:

$y(k) = u(k-2) + 0.1\, e(k)$



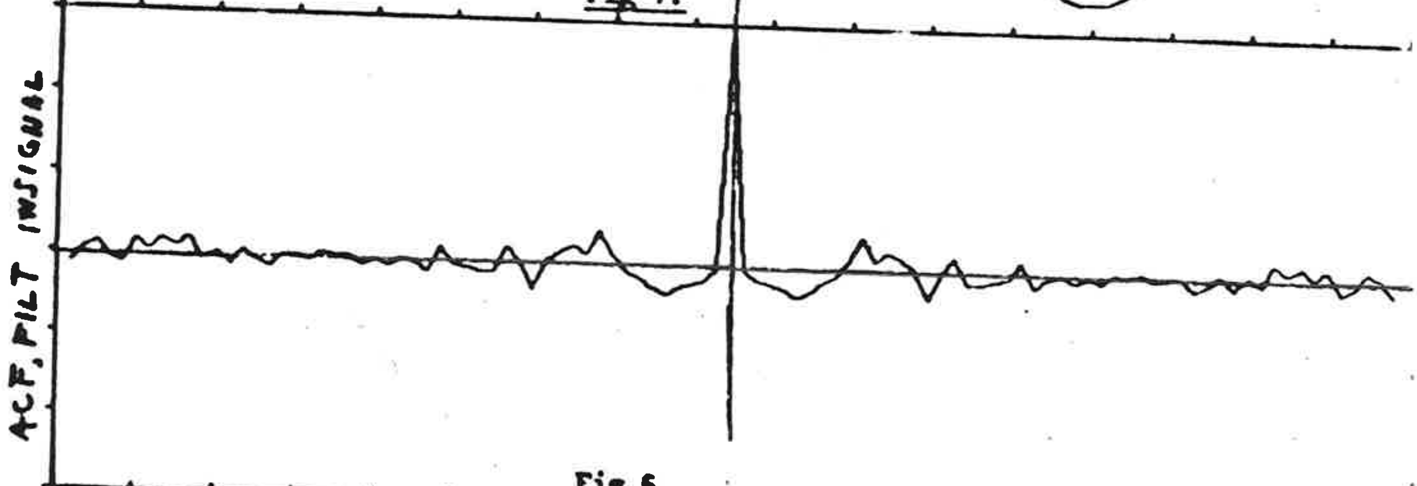
CCF FILTERED

Fig 8.



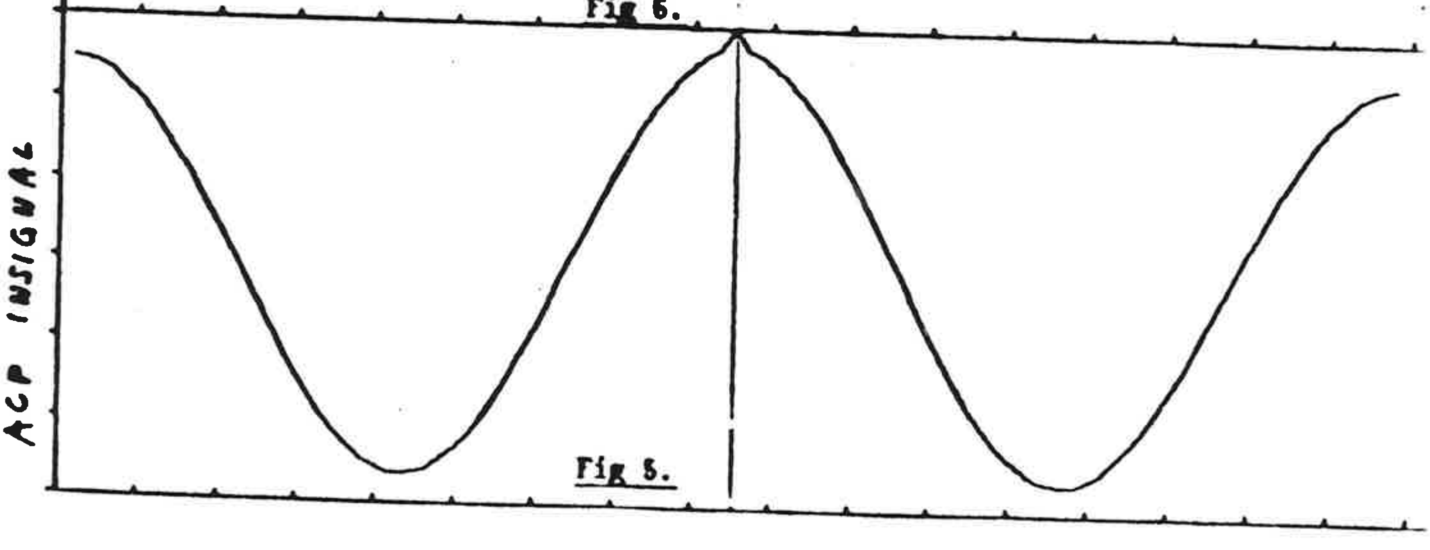
CCF

Fig 7.



ACF FILT

Fig 6.



ACF INPUT

Fig 5.

DISKRET FOURIER TRANSFORM (DFT)

$$X(k) = \sum_{n=0}^{N-1} x(n) \exp\left(-i \frac{2\pi k n}{N}\right)$$

$k = 0, \dots, N-1$

$$x(n) = \frac{1}{N} \sum_{n=0}^{N-1} X(k) \exp\left(i \frac{2\pi n k}{N}\right)$$

$n = 0, \dots, N-1$

N^2 multiplications

SNABB FOURIER TRANSFORM (FFT)

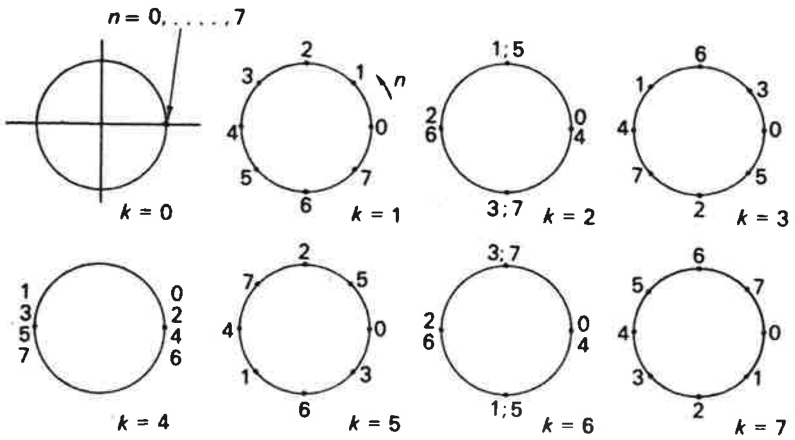
$$\exp\left(-i \frac{2\pi k n}{N}\right)$$

$N = 2^l$ $\quad l$ = heltal

Ex

$N = 8$

$k = 0, \dots, 7$



$N \log_2 N$ multiplications

Ex $N = 1024$ $\quad \frac{N \log_2 N}{N^2} \approx 1\%$

4:28

$y(k) = u(k-2) + \sigma e(k)$

$u(k) = \sin(2\pi k/50) + 0.2\, e(k)$

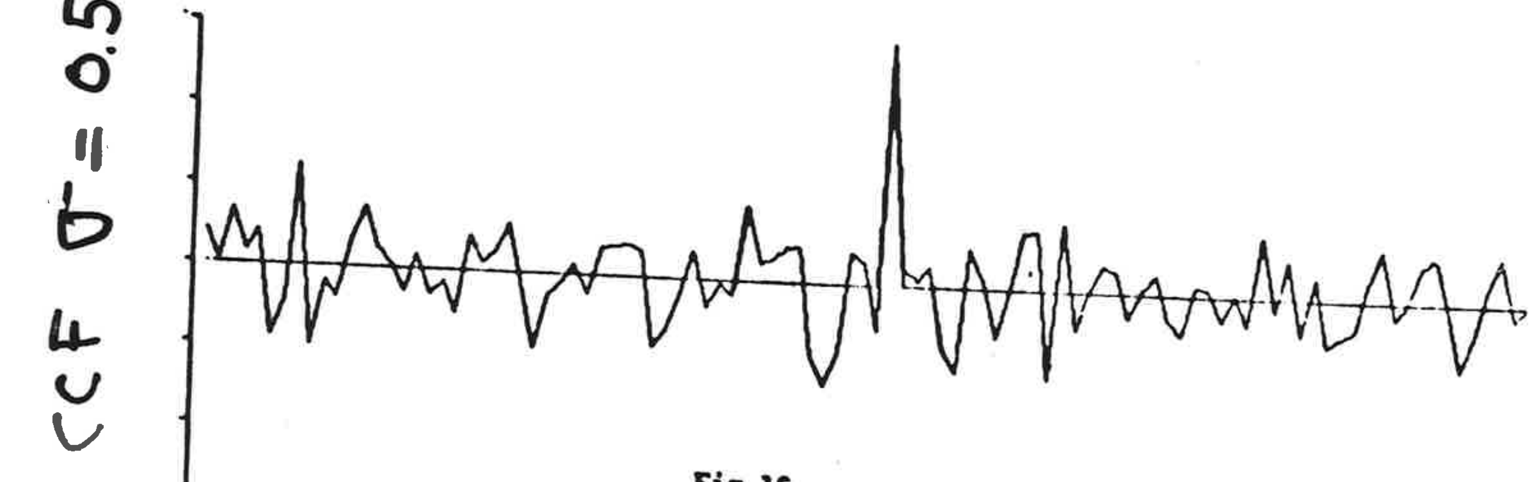


Fig 16.

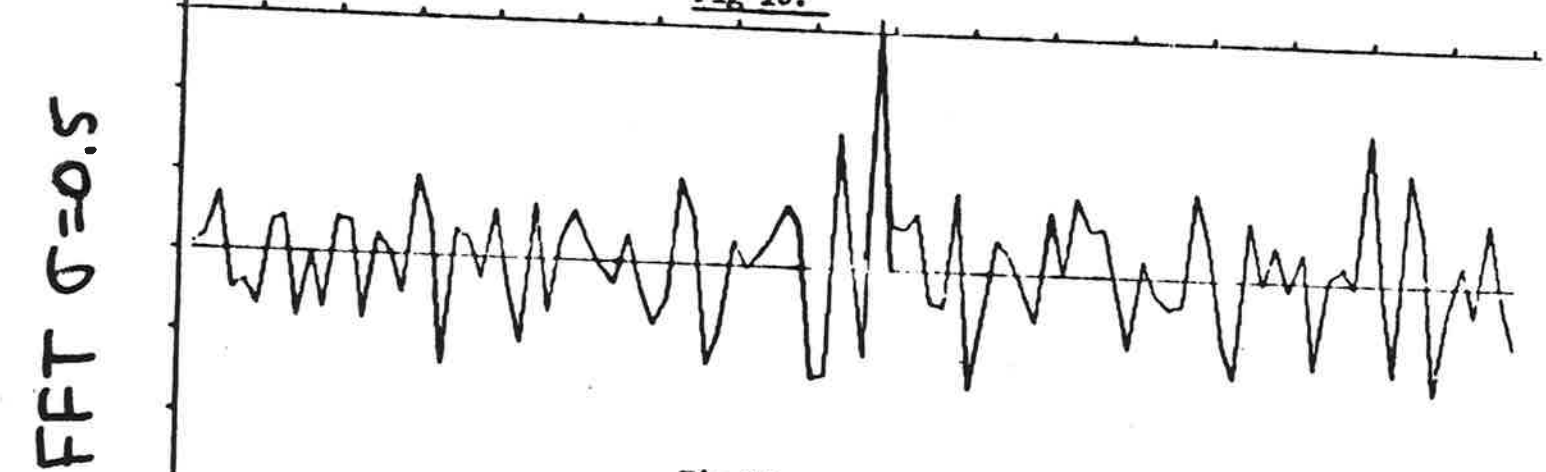


Fig 15.

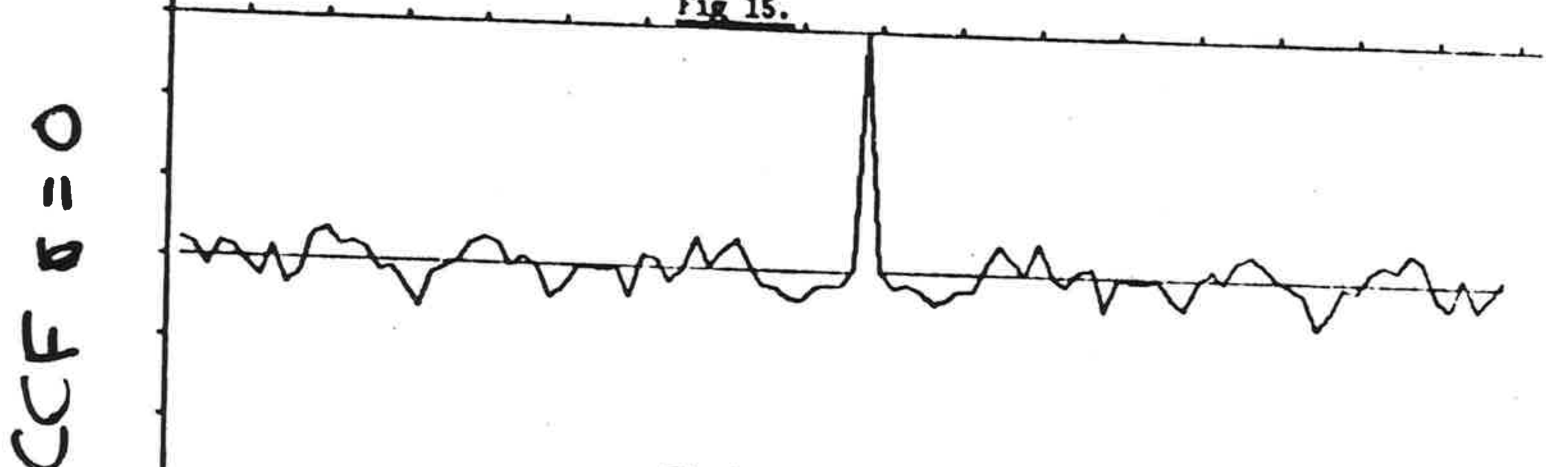


Fig 14.

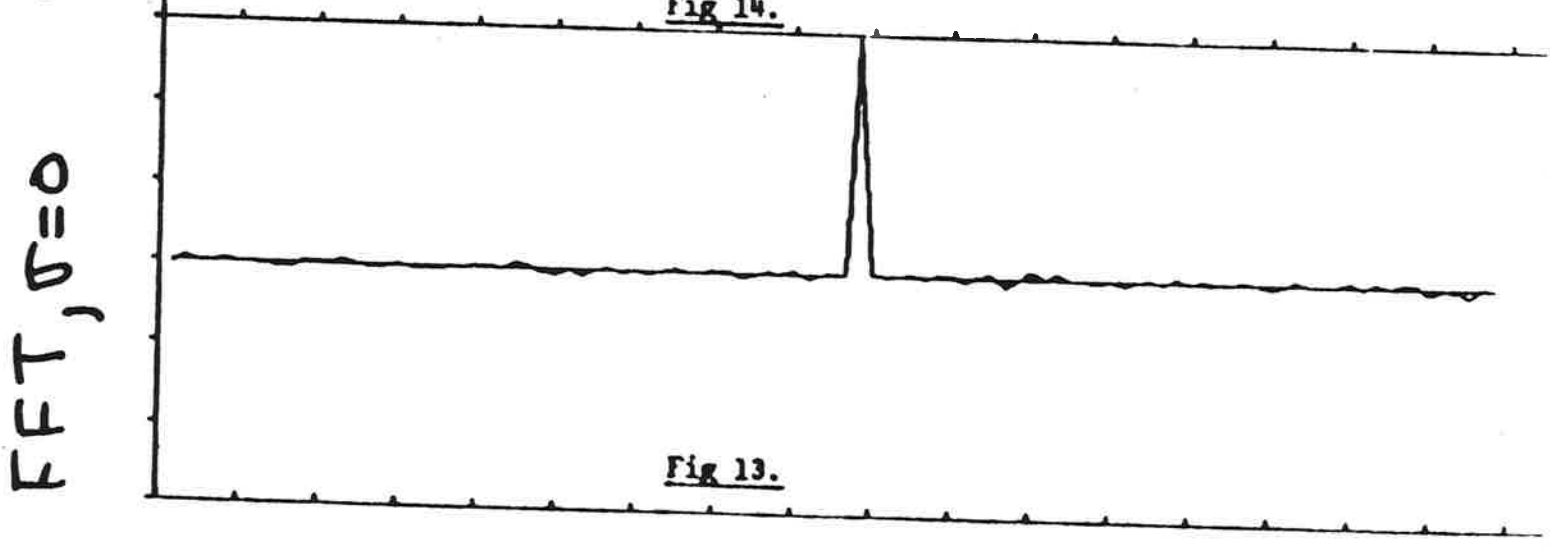


Fig 13.

4:29

A CROSSCORRELATION METHOD FOR MEASURING THE IMPULSE RESPONSE OF REACTOR SYSTEMS

J.D. BALCOMB ET AL NUCLEAR SCI. AND ENG. 11(1961) 159-166

KIWI-A3 IS AN EXPERIMENTAL PROTOTYPE FOR A NUCLEAR ROCKET ENGINE.
REACTOR POWER CAN NOT BE PERTURBED BY MORE THAN 1%. THE NOISE IS ABOUT 0.5%.
TRANSFER FUNCTION FROM CONTROL ROD TO REACTOR POWER IS DESIRED. FULL POWER 10 KW.
STEP TESTS ARE EXCLUDED BECAUSE OF THE OUTPUT NOISE LEVEL AND THE LIMITATION ON ALLOWABLE PERTURBATIONS. OSCILLATION TESTS WOULD TAKE TOO LONG TIME.

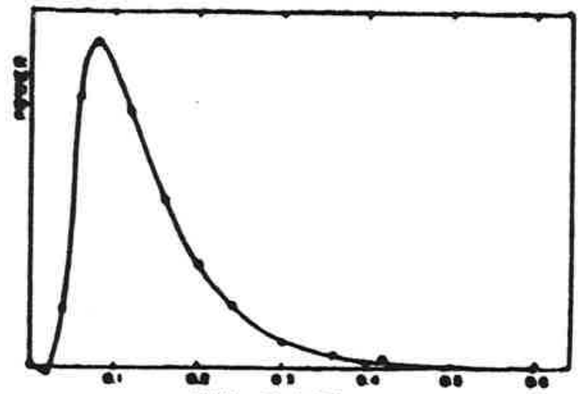


FIG. 7. Crosscorrelation method measurement of the impulse response of the Kiwi A3 reactor at 10 kw in series with an input control rod servo system and ion chamber instrumentation.

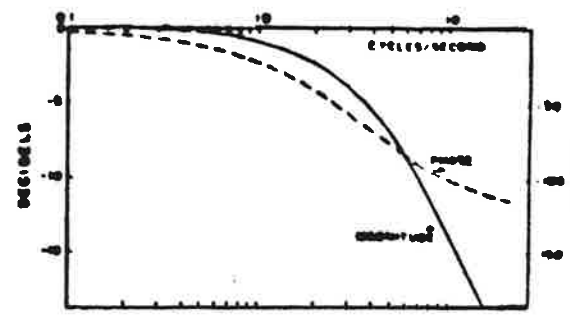


FIG. 8. Transfer function of the Kiwi A3 reactor at 10 kw as calculated from Fig. (13).

CONCLUSIONS

THE VALIDITY OF THE CROSSCORRELATION METHOD FOR THE MEASUREMENT OF THE IMPULSE RESPONSE OF A NUCLEAR REACTOR SYSTEM HAS BEEN DEMONSTRATED EXPERIMENTALLY.
THE ADVANTAGES ARE:

1. IT YIELDS THE ENTIRE INFORMATION ABOUT THE IMPULSE RESPONSE IN THE SHORTEST POSSIBLE TIME.
2. THE METHOD REQUIRES ONLY SMALL AMPLITUDE PERTURBATIONS. CONSEQUENTLY IT IS NOT HAZARDOUS, IT IS NOT LIMITED BY SYSTEM NONLINEARITIES, AND IT DOES NOT INTERFERE WITH NORMAL SYSTEM OPERATION.
3. IT CAN BE USED EVEN IN THE PRESENCE OF STRONG NOISE SOURCES PROVIDED THAT THE CORRELATION TIME IS INCREASED.

CONCLUSIONS

CORRELATION METHOD CAN BE USED TO FIND FREQUENCY-RESPONSE FUNCTIONS WHEN PROPER NUMERICAL TECHNIQUES ARE USED.

A MAJOR LIMITATION IN THE CORRELATION METHOD HAS BEEN FOUND IN THE ESTIMATION OF THE CORRELATION FUNCTIONS FROM FINITE RECORDS. VARIANCES OF THE CORRELATION FUNCTION ESTIMATES CAN BE CALCULATED APPROXIMATIVELY. THESE CALCULATIONS ARE USEFUL WHEN DETERMINING EXPERIMENT LENGTH.

THE INPUT SIGNAL MUST EXCITE THE SYSTEM IN THE IMPORTANT FREQUENCY RANGE.

IT WAS USEFUL TO FILTER THE SIGNALS THROUGH HIGH PASS FILTERS

SLUTSATSER

- △ GER SNABBT RESULTAT
- △ FÄRDIGA PROGRAM OCH KORRELATORER
- △ TILLGÄNGLIGA SYNTESMETODER

- △ GE AKT PÅ ALIASING OCH NOGGRANNHET CONTRA FREKVENSUPPLÖSNING
- △ SVÅRT BESTÄMMA PARAMETRISK MODELL
- △ TVÅ MÄTNINGAR OM STÖRNINGSSPEKTRUM OCKSÅ SKALL BESTÄMMAS.
- △ FÖRFILTRERING (HÖGPASS)
- △ LÅNGA MÄTTIDER
- △ INSIGNALEN MÅSTE EXCITERA PROCESSEN

5. INTERAKTIV DATABEHANDLING

Johan Wieslander

Idéerna bakom konstruktionen av IDPAC och de liknande interaktiva programmen som utvecklats vid Institutionen för reglerteknik, LTH, Lund finns utförligt behandlade i:

Wieslander J: Interaction in Computer Aided Analysis and Design of Control Systems. Report 1019, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

5:1

INTERAKTIV DATABEHANDLING

Johan Wieslander
790328

Interactive User Categories

The users of an interactive program will differ in the relative importance they attach to the facilities offered. They also differ in the frequency with which they utilize these facilities. When designing an interactive program, it is of cource important to realize what the intended users will expect from it. The following is an attempt to summarize a few possible user categories together with their typical needs:

- the batch user
- the experienced user
- the casual user
- the beginner
- the assistant

The batch user can (and must) select in advance a sequence of actions that the program is going to follow, with a specified set of inputs.

It may sound strange that an interactive program might be used in batch mode. It is, however, not at all unnatural. In many cases, a set of similar problems is to be solved. The first two or three may with advantage be solved interactively. After that, the proper way of solving the remaining problems may be known and interaction is no longer valuable. Indeed, it may cause additional costs, as it requires constant human intervention and a more expensive way of running the computer. Therefore an easy and efficient way of running the interactive program in batch mode would be useful.

The *experienced user* is the one with the most exacting demands on the interaction. He has good knowledge and intuitive feeling for the methods he is using and knows and uses the facilities the interaction offers. He might be trying to solve a new and complicated problem exercising his prior knowledge, skill, intuition and common sense combined with the data handling power of the computer. In this situation he wants a maximum of freedom in the choice of solution steps. It is of great importance to him to be able to view the results from the computer as they become available and to be able to communicate his desires promptly. He is likely to be able to spot an erroneous or uninteresting result at an early stage and should have the possibility to abandon such an unpromising

road.

- The casual user could typically be a student solving a laboratory exercise. He would then have to solve a well-defined problem, which is known to be solvable by means of the program in question. Being a casual user, he would not be particularly interested in anything but the facilities necessary for his task. He would consider it an extra burden to be forced to learn a list of commands and command syntax, although this could be considered an advantage from didactic reasons. Rather, a dialogue offering guidance would be preferred, decreasing the risk of serious mistakes and lessening the burden of the supervisor.

- The beginner is initially in the same situation as the casual user, simplicity is important to be able to get started. There is a distinction though; the beginner has a desire to become an advanced user some day. He is interested to learn what facilities are available and to master them. He would want facilities for help and instruction and if possible, a means of gradually growing accustomed to the details of the program.

- The assistant is someone that performs routine investigations, typically designed by the experienced user. The assistant is not required to know the fine details, neither of theory nor of the program. He is primarily engaged in providing the program with proper data and collecting the results. The means by which the experienced user instructs his assistant and whether or not primitives can be constructed is of great importance.

Naturally, the ideal type of interaction is quite different for these users, ranging from no interaction in the batch user case to the heavy demands of the experienced user. It is most important to realize however that the same program may have to meet these varying requirements. A few examples on this situation will be given.

First of all, the ideal situation for the beginner has already been described as a gradual change from interaction with much guidance to the full freedom of the experienced user.

Secondly, let us regard the casual user in the form of a student doing a laboratory exercise. Although he is using an interaction scheme with good guidance, he is likely to get stuck sooner or later. He will then call the help of his supervisor, presumably a more or less experienced one. In the correction of the student's mistake, the supervisor would prefer a more direct form of interaction.

Thirdly and finally, the experienced user may take many shapes. He may turn into a batch user if he finds that the interaction of a part of his job will be entirely predictable like in a Monte-Carlo simulation situation. Or he may be preparing primitives for routine investigations to be done by himself or by his assistant. Or, after a few months of disuse, he may be regarded a fast-learning beginner and will appreciate some of the informative functions created for the beginner.

Summarizing, it may well happen that the desired type of interaction is very varying and in the design of a program, the satisfaction of these demands will pose some problems.

Interaction Needs

We are now going to discuss the needs we have on interaction. The interest will be focused on the type of information that has to be exchanged between the problem solver and the computer. The form of the interaction will be discussed later. The four typical needs are called:

(1) Choices and parameters
(2) Multi level interaction
(3) Computation structure
(4) Interaction structure

Note that the interaction needs listed here are typical not only to automatic control problem solving, but are general to a wide class of situations where a computer is used to help a designer with heavy computations or data handling operations. Typical are also the existence of well structured data objects and the element of human intervention in the operations.

There certainly are disciplines with other interaction needs, e.g. circuit design, computer draughting and inventory control.

(1) Choices and parameters

The most common and most basic information the user of an interactive program will have to pass is what choices he has made and what parameters he wants to use. The choices to be made include the action to be performed and the datasets which are inputs and outputs of a given program module. Parameter values must also be specified. They represent a possibility to influence the operation in a predefined way.

In this type of interaction, the operation is fixed and given by the program code, and the input and parameters are the only freedom left for the user. This situation is by far the most common one. It is typically found in analysis and specification type operations. Hence it also plays an important role in the synthesis and measurement situations.

(2) Multi-level interaction

With multi-level interaction is simply meant that the interaction is split into two or more levels. This situation occurs when proper parameter values, appropriate secondary input or other choices in the applied method are not apparent until some preliminary computations have been performed. It is often possible and most attractive to divide such operations into two or several parts, allowing common analysis tools to be used to determine suitable future steps. If, however, the information to pass between the different parts is special in structure or the analysis needed is not of a general nature, it would be more natural to implement the method in a single program module but allow interaction in several levels. This would also be the case where a number of options exist. If they had to be specified all at the same time, it would be clumsy, difficult to comprehend and remember and would be generally unaestetic.

The solution is to allow interaction in several levels. The first level is used to specify the problem. On the next level the problem is analysed, or details of its solution are entered. In the general case, temporary results could be asked for and allowed to influence the user's actions on the lower levels.

Examples are found in Identification. One applies to the fitting of a transfer function to a frequency response, where the second level of interaction treats details of the curve fitting method. The second example was found in the maximum likelihood identification method, where the lower level of interaction is used to optionally specify the starting point, values of fixed parameters etc.

(3) Computation structure

A need is found in general data analysis operations, in identification of non-linear models and primarily in the simulation of non-linear systems, to be able to interactively specify a series of computations. Unlike the previous needs where the computations are fixed and only data sets and and parameters are changed, we here encounter a need to specify the series of operations themselves. In a computer system, this is a task solved by what is called compilers or interpreters, usually large and expensive programs.

The difference with this interaction need compared to the previous ones is that it involves to parse statements in some arithmetic language, obeying its syntactical rules, and to generate a sequence of (computer) instructions that performs the intended task. It is possible to include such facilities in an interactive program. Here we will not explore this need any further.

(4) Interaction structure

In many places it may be noted that there is a desire of being able to specify an interaction structure. Such a facility would be useful either for temporary or more permanent use.

By interaction structure is understood a fixed sequence of interactions that can be invoked easily, maybe with some planned alterations. The temporary use of such a sequence would be very natural for the interactive user that solves a problem with a partly iterative method.

The more permanent interaction sequence serves to build new functions from other more basic ones. This could be used to implement methods applicable to certain problems, e.g. a synthesis method, or to construct interaction modules aimed at a certain cathegory of users, e.g. students.

Note that this so called 'interaction structure' bears strong resemblance to the earlier 'computation structure'. To a degree, the same desired result could be achieved by this facility, provided suitable basic functions were available and called in proper order.

In fact, this is the key difference between the two concepts. Here, talking of 'interaction structure', a very simple syntax is assumed. The elementary operations in the interactive program are called one at a time, with proper operands. The only rules to obey is in the choosing of operands, something that in any case must be checked, presumably in the code implementing the elementary operations.

Demands on Program Organization

In this section we will try to list some of the demands on interactive programs that arise from the programming point of view. Some are general with no specific influence on this type of programs and will not be discussed to any great length, others have already been encountered.

Portability

The portability of a program means its ability to be run on other computers of comparable or greater size but with other organization. There are some simple rules to follow in order to achieve a high degree of portability. First of all, the programming should be done in the standard dialect of a commonly used programming language. Secondly, parts of a program that have to be computer dependent should be confined in small separate program modules, so as to be easily identified and modifiable. Examples of such computer dependent parts are file I/O, non-standard I/O of textstrings, graphic (display) handling, and numeric test quantities.

Maintainability

Here we are interested in the possibilities to correct/modify portions of a program without affecting the rest of it. The solution lies in the proper structuring of program code, and not least, structuring and storage method of data.

A practical problem arises because programs tend to be large, consisting maybe of several hundred modules. A way out of this situation would be to split the program into several separate parts that, being smaller, would be easier to maintain. In the case of a command dialogue, the parts performing the computations would be a natural choice. The idea would then be to make the main part of the program call the other parts as separate programs when their services are needed. Unlike subroutine (procedure) calls, the existence and way of implementation of this facility is a function of the operating system on a specific computer. However nice, this solution thus violates the demands on portability.

Expandability

The ease of including a new facility may be of importance in many projects. There are a few factors that will promote this quality. One is the frequent use of primitives; i.e. common operations are made available as separate modules forming a pool of ready-made building blocks to glean from. Another one is that the data objects, the program is made to handle, are so structured that different portions of the program can be independent and be able to communicate through them only.

Segmentation

Interactive programs for general use will always be segmented on computers lacking some form of virtual memory system. The reason is either that the primary memory is too small or that there are restrictions on how much that may be used by any one user. The last situation applies to time-shared implementations. The ease with which such a segmentation can be made depends on the internal structure of the program.

Locality

On computers with virtual memory systems, programs need no segmentation, at least if the address space is sufficient. Instead there is a desire to have good locality in the program. This means that the points in address space referenced during a short period of time should be grouped together as well as possible. This will minimize the number of pages to be kept in primary memory as well as the number of page transfers from mass memory.

Modularity

There is a desire that the program code is divided into suitable modules, i.e. subroutines or procedures. Apart from being a result of good programming practice in general, this will be the key to the satisfying of the other demands above.

INTRAC

Intrac is a communication module, used in Idpac and some other interactive programs. Intrac serves as a common means of communication between the user and the program. The facilities offered by Intrac is of prime importance for the interactive possibilities of the programs. These facilities are of a general application independent nature and are described below:

The Macro

Macro commands is a facility supported by Intrac. They are calls to previously defined command sequences on mass memory. Technically, when Intrac recognizes a reference to such a command sequence, it starts reading commands from a mass memory file, rather than from the user's terminal. A macro corresponds to subroutines or procedures in ordinary programming languages.

A macro consists of a sequence of Intrac-statements, macro calls and application commands. They are stored as a text file on mass storage. The first line in the macro should be a MACRO-statement which has the following form 1).

```
MACRO <macro identifier> [<formal argument>/<delimiter>/
                          <termination marker>]*
```

The statement declares the formal arguments of the macro. After the MACRO-statement follows a sequence of Intrac-statements, macro calls and application commands. The last line in the macro should contain an END-statement:

```
END
```

A macro is called by giving its name followed by actual arguments in the same way as a command. If the <termination marker> is not used then the number of actual arguments should be equal to the number of formal arguments in the MACRO-statement. The delimiters appearing among the formal arguments should be given at corresponding positions in the call.

The <termination marker> is used when a variable number of actual arguments is allowed. It indicates that the formal arguments and delimiters appearing following the symbol need not have corresponding actual arguments. If the <termination marker> is used several times in the macro, then it gives alternative places where the call can be terminated. The formal arguments which have no corresponding actual arguments will be 'unassigned'.

An Example

A question & answer dialogue, giving good guidance for the infrequent or one-time user, may be realized in the form of a macro. A simple example using commands from Idpac is shown in Figure 1.

The READ and WRITE general purpose commands of Intrac are used to communicate with the user, presumably in this case a student of stochastic processes. Instead of just 'playing around' with some of the commands in Idpac, requiring some familiarity with specific details, he is taken in an orderly fashion by a macro through a sequence of commands showing the effect of a class of dynamic systems on a white noise input. Some points are worth noting:

a) If an error is detected, the macro will be suspended, i.e. the program goes into command mode. Any Idpac

1) The notation [ ]* denotes that the enclosed item is optional or may be repeated several times.

```
MACRO NOISEDEMO
INSI WNOISE 200
NORM
X
LABEL DESCR
WRITE 'The effect of filtering white noise through'
WRITE 'Butterworth' filters will be demonstrated. You can'
WRITE 'choose filter type, order, and cut-off frequency.'
WRITE 'In the advent of errors, type GOTO DESCR to receive'
WRITE 'this description again, or type GOTO RESTART to'
WRITE 'start from the following.'
LABEL RESTART
WRITE 'Choose filter order and type (LP, BP, HP).'
READ N INT TYPE NAME
WRITE 'Now enter cut-off frequency. Enter two frequencies'
WRITE '(low and high) if you chose BP.'
READ CF REAL; CF2 REAL
FILT FILTR < TYPE N 1. CF CF2
DSIM COLNOISE < FILTR WNOISE
WRITE 'Hit the return key to see 50 samples of gaussian'
WRITE 'noise coloured by your choice of filter.'
READ ; I INT
PLOT 50 COLNOISE  "Plot of coloured noise
WRITE 'Hit return key to see Bode plot of theoretic and'
WRITE 'computed power spectrum.'
READ ; I INT
KILL
ASPEC NSP < COLNOISE 50
SPTRF (POW) FSP < FILTR B/A
BODE FSP NSP
WRITE 'Do you want another run?'
READ ANS YESNO
IF ANS.EQ.YES GOTO RESTART
END
```

Figure 1 A simple question & answer demonstration of coloured noise, implemented via a macro containing informatory text and questions.

command is then legal. The inexperienced user is advised in the description to use GOTO RESTART, which will allow a complete description of filter parameters.

b) The use of the <termination marker> ; in the reading of cut-off frequencies allows input of only one real value. The local variable CF2 will then be 'unassigned', and its appearance in the command FILT will be invisible to the action routine.

c) The dummy read statements READ ; I INT, where the ; allows the user to respond with an empty line, serves to include a pause so that the display is not erased until the user is ready.

Intrac statements

Intrac implements a number of statements of an application independant nature. They provide many of the functions found in any general purpose programming language. They therefore further emphasizes the idea of Intrac as a basis for application oriented problem solving languages.

a) Generation of macros

There are some different ways to generate a macro. Since a macro is implemented as a text file it is possible to generate and modify a macro using a text editor. A macro can also be generated by entering the MACRO-statement from the terminal. This statement was defined in the previous section. In generation mode all correct commands entered from the terminal are stored on a file. This continues until generation mode is left by the END-statement. Whether the commands in the macro should be executed during generation or not is determined by the switch EXEC. If EXEC is OFF then the commands are only checked for formal errors and if correct stored on the file. If EXEC is ON the commands will also be executed.

The FORMAL-statement can be used to extend the list of formal arguments anywhere in the macro. It is placed after the MACRO-statement automatically when the generation is finished.

b) Assigment of variables

Formal arguments are allocated and possibly assigned when a macro is entered. Their values can be changed with the LET-, DEFAULT-, FOR-, and READ-statements. Among the forms allowed is the usual arithmetic statement, and the main form is:

```
LET {<variable>=}*{<number>[{+/-/*//}<number>]
```

The DEFAULT-statement is a conditional assignment statement. Its form is:

```
DEFAULT {<variable>=}* <argument>
```

The assignment is performed only if either
- the named variable is 'unassigned'
- the named variable does not exist.

In the last case a new variable is allocated.

c) Branching

To make macros flexible it is necessary to have a way to change the sequence of commands executed. This may be acieved through branching statements and labels. The labels used in branch statements are declared 'on site' using the LABEL-statement:

```
LABEL <label identifier>
```

```
<label identifier>::=<identifier>/<integer>
```

The unconditional GOTO-statement is:

```
GOTO <label identifier>
```

Since the argument in the GOTO-statement could be a variable whose value is a label identifier it is possible to use the statement as the assigned GOTO of FORTRAN.

The conditional GOTO statement has the form:

```
IF <argument> (EQ/NE/GE/LE/GT/LT) <argument>
  GOTO <label identifier>
```

The effect of this statement is the same as for the GOTO-statement if the relation is true. If it is false the next command in sequence is executed.

d) Looping (FOR, NEXT)

It is possible to introduce loops among the commands in a macro. This is done with the FOR- and NEXT-statements. The FOR-statement begins the loop and has the following form:

```
FOR <variable> = <number> TO <number> [STEP <number>]
```

The NEXT-statement ends the loop and has the form:

```
NEXT <variable>
```

e) Output and input

The macro facility can be used to implement question and answer interactive programs. Questions are written on the terminal with the WRITE-statement and the answers are read using the READ-statement. The WRITE-statement is used to write variables and text strings. Its form is

```
WRITE [( [DIS/TP/LP] [FF/LF] )] [<variable>/<string>]*
```

The READ-statement reads values from the terminal and assigns variables. Its form is:

```
READ { {&lt;variable&gt; {INT/REAL/NUM/NAME/DELIM/YESNO}} /
       &lt;termination marker&gt;}*
```

After each variable a type specification for the expected value is given:

INT	– integer number
REAL	– real number
NUM	– integer or real number
NAME	– identifier
DELIM	– delimiter
YESNO	– identifier YES or NO

When the READ-statement is executed a prompting # is written on the terminal. The <termination marker> has the same function as in the MACRO-statement. It gives alternative places where the answer could be cut off. The variables that are not given any value become 'unassigned'.

There are two means of escape from the READ statement, resulting in the suspending of the macro.

- If the answer is just a > the READ-statement will have no effect and the macro is suspended. If the macro is resumed by the statement RESUME the READ-statement will be re-executed.
- If an acceptable answer is given followed by a > the variables will be properly assigned and the macro suspended. If the macro is resumed with RESUME, the command following the READ will be executed.

f) Suspending a macro

There are cases when the freedom to have formal arguments in a macro is not enough. At generation time it may for example not be known which command is appropriate at some point in a macro. It is then possible to switch to command input from the terminal (i.e. suspend the macro). When the command input from the terminal is finished the macro is resumed. This facility is handled by the statements SUSPEND and RESUME.

A macro is automatically suspended in some cases.

- When an error is detected during the execution of a macro then an error message is printed and the macro is suspended. The user can then e.g. enter a correct form of the erroneous command and then RESUME the macro.
- When the READ-command has been executed in a macro, the user has to input the requested values from the terminal,

or he can enter a special escape character ()) which causes the macro to be suspended.

Idpac Command List

ACOF

Computes the autocovariance for a column in a data file

```
ACOF FNAM1[(C1)] < FNAM2[(C2)] NOL [EXT]
```

ASPEC

Computes the autospectrum for a column in a data file

```
ASPEC FRF[(F)] < FNAM2[(C2)] NOL [FREQ]
```

BODE

Plots amplitude and phase versus angular frequency in a logarithmic diagram on display

```
BODE [(SW)] FRF1[(F11 F12 ..)] [FRF2[(F21 .. )] ..
```

Subcommands:

```
PAGE
KILL
```

CCOF

Computes the cross covariance for a column in a data file

```
CCOF FNAM1[(C1)] < FNAM2(C21 C22) NOL
```

```
CCOF FNAM1[(C1)] < FNAM2[(C2)] FNAM3[(C3)] NOL
```

CONC

Concatenates two data files

```
CONC [DNAM1] < DNAM2 DNAM3
```

CONV

Transfers a free format data file to a binary data file

```
CONV DNAME < FNAM[(C1..)] NCOLX [TSAMP]
```

CSPEC

Computes the cross spectrum for a column in a data file

```
CSPEC FRF[(F)] < FNAM2(C21 C22) NOL [IALIGN] [FREQ]
```

```
CSPEC FRF[(F)] < FNAM2[(C2)] FNAM3[(C3)] NOL [IALIGN]
                 [FREQ]
```

CUT

Picks out a part of a data file

```
CUT [DNAM1] < DNAM2 IB IE
```

DELET

Deletes files from disk

```
DELET FNAM1[(DMODE1)] [FNAM2[(DMODE2)] ... ]
```

DETER

Performs simulation of a multiple input - single output linear discrete dynamic system as an addition of simulations of single input - single output systems

```
DETER DNAM1[(C1)] < SNAME[(NAME)] DNAM2[(C21 ..)]
                    [DNAM3[(C31 ..)] [....]..] [NP]
```

DFT

Performs the Discrete Fourier Transform on a time series

```
DFT [(RES)] [(WND)] SPEC < DATA[(IND)] [START NSAMP]
```

DSIM

Similar to DETER but includes a noise input

```
DSIM DNAM1[(C1)] < SNAME[(NAME)] DNAM2[(C21 ..)]
                   [DNAM3(C31 ..)] [....]..] [NP]
```

EDIT

Symbolic text editor

```
EDIT FNAME
```

FHEAD

Displays file head and enables the user to change its parameters

```
FHEAD [AGGREG:]FILE
```

FILT

Computes a digital low- or high-pass Butterworth filter of given order and with given cut-off frequency. A band pass/stop filter is constructed by combining a high- and a low-pass filter and has the double order.

```
FILT FNAME < FITYP NO DELTAT OML [OMH]
```

FORMAT

Converts a binary data file into a formatted data file

```
FORMAT [FFILE] < BFILE[(C1 C2 .. )] [BEGIN COUNT]
```

FROP

Adds, subtracts, multiplies or divides two frequency response files for frequencies which coincide with an error less than .000001.

```
FROP [FRF1[(F1)]] < FRF2[(F2)] OP FRF3[(F3)]
```

FTEST

Performs a file existence test

```
FTEST FNAME [(DMODE)]
```

GETFIL

Retrieves a file from back-up storage

```
GETFIL PROGFILE FILESPEC [FILESPEC..]
```

IDFT

Performs the Inverse Discrete Fourier Transform on a frequency response

```
IDFT DATA < SPEC[(IND)]
```

INSI

Generates data sequencies

```
INSI FNAME [(C)] NP [TSAMP]
```

Subcommands:

```
PRBS [IBP [NBIT [ISTART [OPT] ] ] ]
NORM [RMEAN SIGMA]
RECT [A B]
SINE [OMEGA FI]
ZERO
STEP
RAMP [A B]
PULSE [LENGTH]
SRTW [PS]
LOOK
KILL
X
```

LIST

Lists on display, line printer or teleprinter the contents of (a part of) a data file, a macro file or a system file - for a data file the columns and the first record and number of records may be specified, for a system file sections of interest may be specified

```
LIST [(DEV)] [(FEED)] [(DMODE)]
     [AGGREG:]FNAME[(A1 A2..)] [IF NUM]
```

LS

Performs Least Squares identification

```
LS [(SW)] SYST[(SECT)] < SFIL [EXT]
```

Subcommands:

```
SAVE STDEV
SAVE COMAT
KILL
X
```

ML

Performs Maximum Likelihood identification

```
ML [(SW)] SYST[(NAME)] < DATA[(C1 .. )] NO [EXT]
```

Subcommands:

```
INVAL 'ABC'/'C' SYST[(NAME)]
FIX A (2) [VA2] (3) [VA3] B (21) [VB21] .....
SAVE [STDEV] [GRAD] [EVALS] [COMAT]
LOOK
KILL
X
```

MOVE

Transfers a data file, a system file, a macro file or specified columns in a data file from one kind of mass storage to another. Can also be used to rearrange the columns of a data file

```
MOVE [(OUTP)] [(DMODE)] [[AGOUT:] FOUT [(C11..)]] [O] <
     [(INP)] [AGIN:] FIN [(C21..)]
```

PICK

Picks out equidistant records from a data file

```
PICK FNAM1 < FNAM2 NR
```

PLMAG

Makes it possible to plot small parts of a data vector and alter data values or remove data points

```
PLMAG DATA [(C)]
```

Subcommands:

```
B[LOCK] NB
P[LBEG] NR
A[LTER] NR [NUM]
PA[GE]
D[ELET] NR [NUM]
KILL
X
```

PLOT

Plots data files on display

```
PLOT [(NP)] [FNAMX[(C1..)] < ] [(OPT1)] FNAM1[(C11..)]
    [[(OPT2)] [FNAM2[(C21..)]] .. ] [YMI YMA]
```

Subcommands:

```
KILL
PAGE
SKIP [N]
```

RANPA

Generates a Gaussian random vector with given covariance matrix and adds it to the parameters in a system description

```
RANPA SNAM1 < SNAM2[(NAME)]
```

RESID

Computes residuals, autocorrelations of residuals, and cross correlations between residuals and input signal(s)

```
RESID RES[(C1)] < SYST[(NAME)] DATA[(C11 C12 .. )]
                  [NOL [NFREE]] [EXT]
```

Subcommands:

```
KILL
PAGE
TABLE
```

SAVFIL

Saves a file on back-up storage

```
SAVFIL PROGFILE FILESPEC [FILESPEC..]
```

SCLOP

Each element in a specified column in a data file is added, subtracted, multiplied or divided by a constant

```
SCLOP [FNAM1[(C1)] < FNAM2[(C2)] OPER CONST
```

SLIDE

Shifts the columns in a data file along each other

```
SLIDE [FNAM1] < FNAM2 K1 K2 K3 ..
```

SPTRF

Computes the power spectrum or the amplitude and phase of a transfer function TPN(Q^-1)/TPD(Q^-1)

```
SPTRF [(SW)] FRF[(F1)] < SYST[(NAME)]
      TPN[(NRN)] / TPD[(NRD)] [FREQ[(F2)]]
```

SQR

Computes square-root matrix for LS identification

```
SQR RFIL < FNAME [(C1 C2 ..)] SFIL
```

STAT

Computes the statistical properties sum, mean value, variance, standard deviation, minimum and maximum value for a specified column in a data file

```
STAT FNAME [(C)] [EXT]
```

STRUC

Creates and updates struc files

```
STRUC SNAM2

STRUC [SNAM2] < SNAM1
```

Subcommands:

```
REVERT
NA   [SW] NVA1
NU   [SW] NVA1
NB   [SW] NV1 ... NVNU
KB   [SW1] NV1 ... NVNU
FIX A(N) [VN] (M) [VM] ...
B NU1 (N1) [V1] (N2) ... B NU2 ...
UNFIX A(..  N .. M ..) B NU1 (N1 ... NN) ..
SW : 'MAX' / 'ACT'
SW1: SW / 'MIN'
KILL
X
```

TREND

Removes polynomial trends from data vectors using least-squares technique

TREND [FNAM1[(C1)]] < FNAM2[(C2)] NO [IF IL]

TURN

Manipulates program switches

TURN SWITCH STATE

VECOP

Adds, subtracts, multiplies or divides two data vectors element by element

VECOP [DNAM1[(C1)] < DNAM2[(C2)] OPER DNAM3[(C3)]

6. MINSTA KVADRAT METODEN (MK)

Jan Sternby

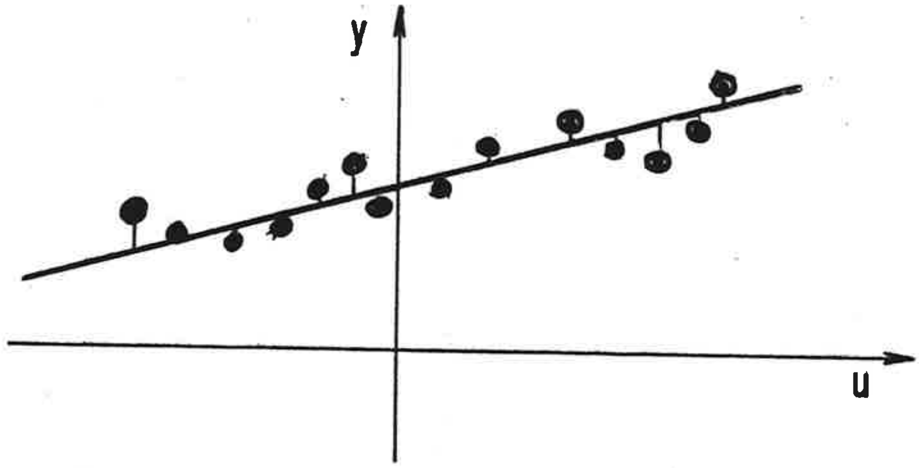
Minsta kvadratmetoden finns beskriven i de flesta böcker om identifiering. Den finns utförligt behandlad i

Åström K J: Lectures on the identification problem - the least squares method. Report 6806, Division of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1968.

IDENTIFIERING MED MINSTA KVADRAT METODEN

* REGRESSIONSANALYS
* MK PÅ DYNAMISKA SYSTEM
* MK I IDPAC
* EGENSKAPER - SVÅRIGHETER
* TEST AV ORDNINGSTAL
* EXEMPEL

REGRESSIONSANALYS



GIVET: MÄTPUNKTER (U, Y)

SÖKT: PARAMETRARNA FÖR EN LINJE, SOM ENLIGT NÅGOT KRITERIUM ANPASSAR LINJEN SÅ BRA SOM MÖJLIGT TILL MÄTPUNKTERNA

MODELL: $Y = BU + C$

KRITERIUM:

MINIMERA $\sum e^2$ $e_i = y_i - Bu_i - C$

DÄR e ÄR AVSTÅNDEN I Y-LED FRÅN PUNKTERNA TILL LINJEN OCH DÄR SUMMERINGEN SKER ÖVER ALLA MÄTPUNKTERNA.

LÖSNINGSMETOD:

DERIVERA MED AVSEENDE PÅ B OCH C

$$V = \frac{1}{2}\sum_i e_i^2 = \frac{1}{2}\sum_i [Y_i - BU_i - C]^2$$

$$\begin{cases} \dfrac{\partial V}{\partial B} = \sum_i [Y_i - BU_i - C]\cdot(-U_i) = 0 \\ \dfrac{\partial V}{\partial C} = \sum_i [Y_i - BU_i - C]\cdot(-1) = 0 \end{cases}$$

$$\begin{cases} \left[\sum_i U_i^2\right]\cdot B + \left[\sum_i U_i\right]\cdot C = \sum_i Y_i U_i \\ \left[\sum_i U_i\right]\cdot B + \left[\sum_i 1\right]\cdot C = \sum Y_i \end{cases}$$

Ger $\hat{B}$ och $\hat{C}$

6:4

Inför $\varphi_i = [U_i \quad 1]^T$

$\Theta = [B \quad C]^T$

Modell:

$$y_i = [U_i \quad 1]\begin{bmatrix} B \\ C \end{bmatrix} = \varphi_i^T \Theta$$

Fler parametrar:

Ex: $\varphi_i = [1 \; U_i \; U_i^2 \; \cdots \; U_i^n]^T$

$\Theta = [a_0 \; a_1 \; a_2 \; \cdots \; a_n]^T$

Vi får

MODELL	MODELLFEL
$y_1 = \varphi_1^T \Theta$	$e_1 = y_1 - \varphi_1^T \Theta$
$y_2 = \varphi_2^T \Theta$	$e_2 = y_2 - \varphi_2^T \Theta$
$\vdots$	$\vdots$
$y_N = \varphi_N^T \Theta$	$e_N = y_N - \varphi_N^T \Theta$

Vektornotation:

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \quad \Phi = \begin{bmatrix} \varphi_1^T \\ \varphi_2^T \\ \vdots \\ \varphi_N^T \end{bmatrix} \quad E = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}$$

MATRIS!

$$E = Y - \Phi\theta$$

Minimera

$$2V = \sum_i e_i^2 = E^T E = (Y - \Phi\theta)^T (Y - \Phi\theta)$$

$$= Y^T Y - Y^T \Phi \theta - \theta^T \Phi^T Y + \theta^T \Phi^T \Phi \theta$$

$$= \left[\theta - (\Phi^T\Phi)^{-1}\Phi^T Y\right]^T \Phi^T \Phi \left[\theta - (\Phi^T\Phi)^{-1}\Phi^T Y\right] +$$

$$+ Y^T Y - Y^T \Phi (\Phi^T \Phi)^{-1} \Phi^T Y$$

Välj $\boxed{\hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^T Y}$

Löser "normalekvationerna":

$$\Phi^T\Phi \quad = \Phi^T Y$$

EGENSKAPER

(Regressionsanalys)

Verkligt system:

$y_i = \varphi_i^T \theta_0 + e_i$

e_i har medelvärde 0

* $E\,\hat{\theta} = \theta_0$ (mvr, unbiased)

Om $E\,e_i e_j = \begin{cases} 0 & i \neq j \\ \sigma^2 & i = j \end{cases}$ så:

* $\text{Var}\,\hat{\theta} = \sigma^2 (\Phi^T \Phi)^{-1}$ (= effektivitet)

* $s^2 = 2V(\hat{\theta})/(N-n)$ mvr skattning av σ^2

$\text{Var}(\hat{\theta})$ avtar som $1/N$ (om $\varphi_i \sim 1$)

Minsta kvadrat identifiering

Modell:

$$y(t) + a \cdot y(t-1) = b \cdot u(t-1) + e(t)$$

Minimera:

$$V = \frac{1}{2}\sum_t e(t)^2 = \frac{1}{2}\sum_t \left[y(t) + a y(t-1) - b u(t-1)\right]^2$$

$$\begin{cases} \dfrac{\partial V}{\partial a} = \sum_t \left[y(t) + a y(t-1) - b u(t-1)\right] \cdot y(t-1) \\ \dfrac{\partial V}{\partial b} = \sum_t \left[y(t) + a y(t-1) - b u(t-1)\right] \cdot \left[-u(t-1)\right] \end{cases}$$

$\frac{\partial V}{\partial a} = \frac{\partial V}{\partial b} = 0$ ger

$$\begin{cases} \left[\sum_t y(t-1)^2\right] \cdot a - \left[\sum u(t-1) y(t-1)\right] \cdot b = -\sum_t y(t) y(t-1) \\ -\left[\sum_t y(t-1) u(t-1)\right] \cdot a + \left[\sum_t u(t-1)^2\right] \cdot b = \sum_t y(t) u(t-1) \end{cases}$$

Relation till korrelationsanalys!

Jämför med regressionsanalys:

$$\varphi(t) = [-y(t-1) \quad u(t-1)]^T$$

$$\theta = [a \quad b]^T$$

Modell: $$y(t) = \varphi(t)^T \theta$$

I matrisform:

$$Y = \Phi \theta$$

Minimera

$$\sum_t e(t)^2 = \sum_t [y(t) - \varphi(t)^T \theta]^2 =$$

$$= E^T E = (Y - \Phi\theta)^T (Y - \Phi\theta)$$

$$\boxed{\hat{\theta} = (\Phi^T \Phi)^{-1} \Phi^T Y}$$ som förut!

* <u>Men</u> Φ nu stokastisk
* Lätt utvidga till fler parametrar.

Relation till Kalman filter

$\varphi(t)^T\hat{\theta}$ predikterar $y(t)$

$e(t) = y(t) - \varphi(t)^T\hat{\theta}$ prediktionsfel

$\hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^T Y$ rekursivt i data:

$$\begin{cases} \hat{\theta}(t) = \hat{\theta}(t-1) + K(t)e(t) \\ e(t) = y(t) - \varphi(t)^T\hat{\theta}(t-1) \\ K(t) = \dfrac{P(t-1)\varphi(t)}{1 + \varphi(t)^T P(t-1)\varphi(t)} \\ P(t) = P(t-1) - \dfrac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{1 + \varphi(t)^T P(t-1)\varphi(t)} \end{cases}$$

Jämför Kalmanfilter för

$$\begin{cases} \theta_0(t+1) = \theta_0(t) & \text{– tillstånd} \\ y(t) = \varphi(t)^T\theta_0(t) + e(t) & \text{– mätning} \end{cases}$$

MINSTA KVADRAT I IDPAC

LSID MODEL<DATA(N_1 N_2) N

INSIGNAL : DATA(N_1)

UTSIGNAL : DATA(N_2)

MODELLORDNING: N

RESULTAT I FILEN MODEL

Egenskaper

(MK identifiering)

Verkliga systemet

$$y(t) = \varphi(t)^T \theta_0 + e(t)$$

- $E\, e(t) = 0$
- $E\, e(t)e(s) = \begin{cases} 0 & t \neq s \\ \sigma^2 & t = s \end{cases}$ (vitt brus)
- $e(t)$ oberoende av $u(s)$, alla t, s
- u tillräckligt oregelbunden

Då:

* $E\hat{\theta} \to \theta_0$ $N \to \infty$ (asymptotiskt mvr)
* $\hat{\theta} \to \theta_0$ $N \to \infty$ (konsistens)
* $\text{cov}(\hat{\theta}) \to \sigma^2(\Phi^T\Phi)^{-1}$ (as. effektiv)
  [ger skattning av noggrannhet]
* $\hat{\sigma}^2 = \frac{2}{N-n} V(\hat{\theta})$ as. mvr

För minsta kvadratmetoden antogs

$$y(t) + ay(t-1) = bu(t-1) + e(t)$$

där $e(t)$ och $e(s)$ oberoende för $t \neq s$

Är detta realistiskt?

Exempel

$$\begin{cases} x(t+1) + ax(t) = bu(t) \\ y(t) = x(t) + e(t) \end{cases} \longleftarrow \text{ober. mätfel}$$

$$y(t+1) + ay(t) = bu(t) + \underbrace{e(t+1) + ae(t)}_{v(t)}$$

VAD HÄNDER OM RESIDUALERNA ÄR KORRELERADE?

EXEMPEL:

$$y(t) - 0.5\, y(t-1) = 1.0\left[e(t) + c\, e(t-1)\right] + 1.0\, u(t-1)$$

c = 0.,0.2, 0.5, 0.7, 0.8, 0.9, 0.99

RESULTAT:

c	$\hat{a} \pm \hat{\sigma}(a)$	$\hat{b} \pm \hat{\sigma}(b)$	λ
0.0	-0.515 ± 0.027	1.023 ± 0.045	1.023
0.2	-0.577 ± 0.026	1.019 ± 0.046	1.037
0.5	-0.642 ± 0.025	1.014 ± 0.050	1.116
0.7	-0.668 ± 0.025	1.011 ± 0.054	1.205
0.8	-0.677 ± 0.025	1.010 ± 0.056	1.260
0.9	-0.684 ± 0.026	1.009 ± 0.059	1.320
0.99	-0.689 ± 0.026	1.008 ± 0.061	1.379
Rätta värden	-0.5	1.0	1.0

Tabell 3. RESULTAT FRÅN MINSTA KVADRATSKATTNING AV SYSTEM MED KORRELERADE RESIDUALER

ANDRA METODER

✓1. GENERALISERAD MINSTA-KVADRAT SKATTNING
✓2. MAXIMUM LIKELIHOOD IDENTIFIERING
✓3. MINSTA KVADRAT SKATTNING AV HÖG ORDNING

MK av hög ordning

Exempel:

$$y(t) - 0.5\,y(t-1) = (1 - 0.5q^{-1})\,y(t) = $$
$$= A(q^{-1})\,y(t)$$

Modell:

$$A(q^{-1})\,y(t) = u(t) + C(q^{-1})\,e(t)$$

Skriv om

$$\frac{A(q^{-1})}{C(q^{-1})}\,y(t) = \frac{1}{C(q^{-1})}\,u(t) + e(t)$$

Men

$$D(q^{-1}) = \frac{1}{C(q^{-1})} = \frac{1}{1 - cq^{-1}} \approx 1 + cq^{-1} + c^2 q^{-2} + \ldots$$

$$\boxed{ADy = Du + e}$$ MK-modell!

- Högre ordningstal
- Rätt överföringsfunktion

Återkoppling

Kan ge problem!

Exempel:

$$\begin{cases} y(t) + a y(t-1) = b u(t-1) + e(t) \\ u(t) = f y(t) \end{cases}$$

$$2V(a,b) = \sum e(t)^2 =$$

$$= \sum [y(t) + \underline{a} y(t-1) - \underline{b} u(t-1)]^2 =$$

$$= \sum [y(t) + a y(t-1) - b u(t-1) + \underbrace{k f y(t-1) - k u(t-1)}_{=0}]^2 =$$

$$= \sum [y(t) + (a+kf) y(t-1) - (b+k) u(t-1)]^2 =$$

$$= 2V(a+kf, b+k)$$

$\therefore V(a,b) = V(a+kf, b+k)$ alla k!

Rät linje i parameterplanet.

6:16

F-test av ordningstal

$V(\hat{\theta})$ minskar då n ökas.

Vad är signifikant?

Antag data genererats av

$$Y = \Phi \theta_0 + e \qquad e \in N(0, \sigma^2)$$

Antal parametrar $= n_0$

Tag $n_0 \leq n_1 \leq n_2$ — Minima V_i

Då gäller (asymptotiskt)

(i) $\frac{2V_2}{\sigma^2} \in \chi^2(N - n_2)$

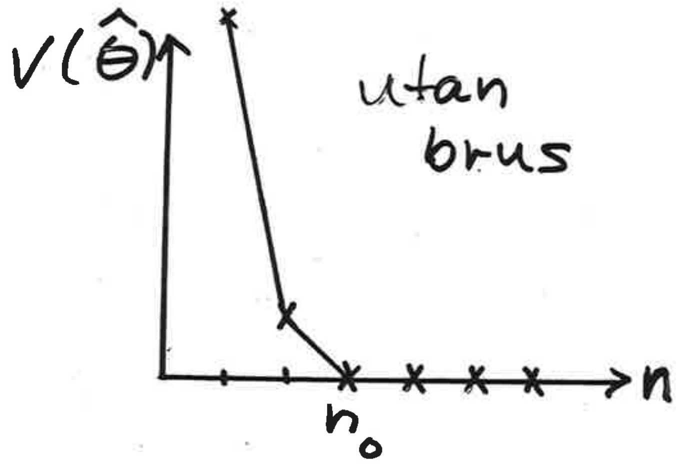
(ii) $\frac{2(V_1 - V_2)}{\sigma^2} \in \chi^2(n_2 - n_1)$

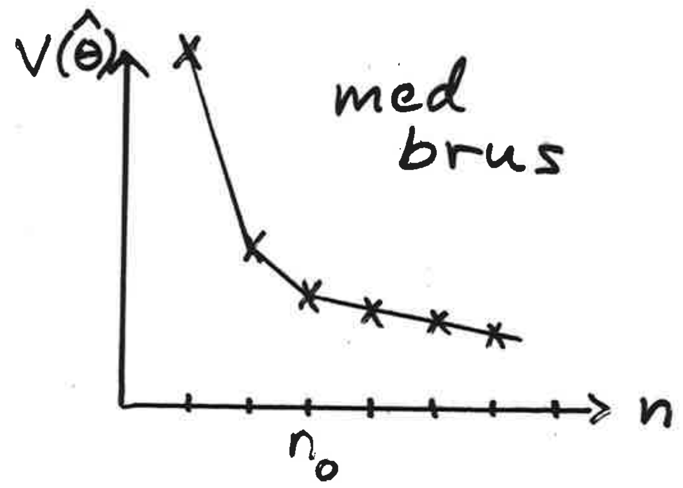
(iii) V_2 och $V_1 - V_2$ oberoende

så att $\tau = \frac{V_1 - V_2}{V_2} \cdot \frac{N - n_2}{n_2 - n_1} \in F(N - n_2, n_2 - n_1)$

Ex: $n_2 - n_1 = 2$, $N \gtrsim 100$: $P(\tau > 3) \approx 0.05$

AIC





Minskar V tillräckligt när n ökas?

<u>Idé</u>: Addera en term som ökar med n.

Informationsteori motiverar kriteriet

$$AIC = N(1 + \ln 2\pi + \ln \hat{\lambda}^2) + 2n$$

$$\left[\hat{\lambda}^2 = \frac{2}{N} V(\hat{\theta}) \right]$$

AIC minimalt för rätt antal parametrar?

6:18

TEST OF ORDER

- Residuals
- Loss function reduction (F - test)
- AIC
- Errors of deterministic output
- Parameter redundancy
- Common factors
- Cross-check
- Simulations
- A priori knowledge

Exempel:

$$y(t) - 1.5\,y(t-1) + 0.7\,y(t-2) =$$
$$= u(t-1) + 0.5\,u(t-2) + e(t)$$

$u(t) = \pm 1$ enl. figur (PRBS)

$e(t) \in N(0, 1)$

$N = 102$

Kommandon för exempel

```
LSID   LS1<U(1 2)  1          }
RESID  RES1<LS1  U            }
PLOT   U(2)/(HP)U(1)/E        }  Ordning 1
DETER  Y1<LS1  U(1)           }
PLOT   Y1/(HP)U(1)/RES1       }

LSID   LS2<U(1 2)  2          }
RESID  RES2<LS2  U            }  Ordning 2
DETER  Y2<LS2  U(1)           }
PLOT   Y2/(HP)U(1)/RES2       }

LSID   LS3<U(1 2)  3             Ordning 3

STOP
```

LS(SC) LSI<LSFIL
79.03.16 - 15:49:24

RESULT OF PARAMETER REDUCTION(S):

DISCARDED PAR.	VLOSS	AIC
NONE	45.908	302.73
A (3)	45.911	298.74
B I(3)	46.450	295.93 ←
A (2)	98.381	368.48
B I(2)	138.03	399.02

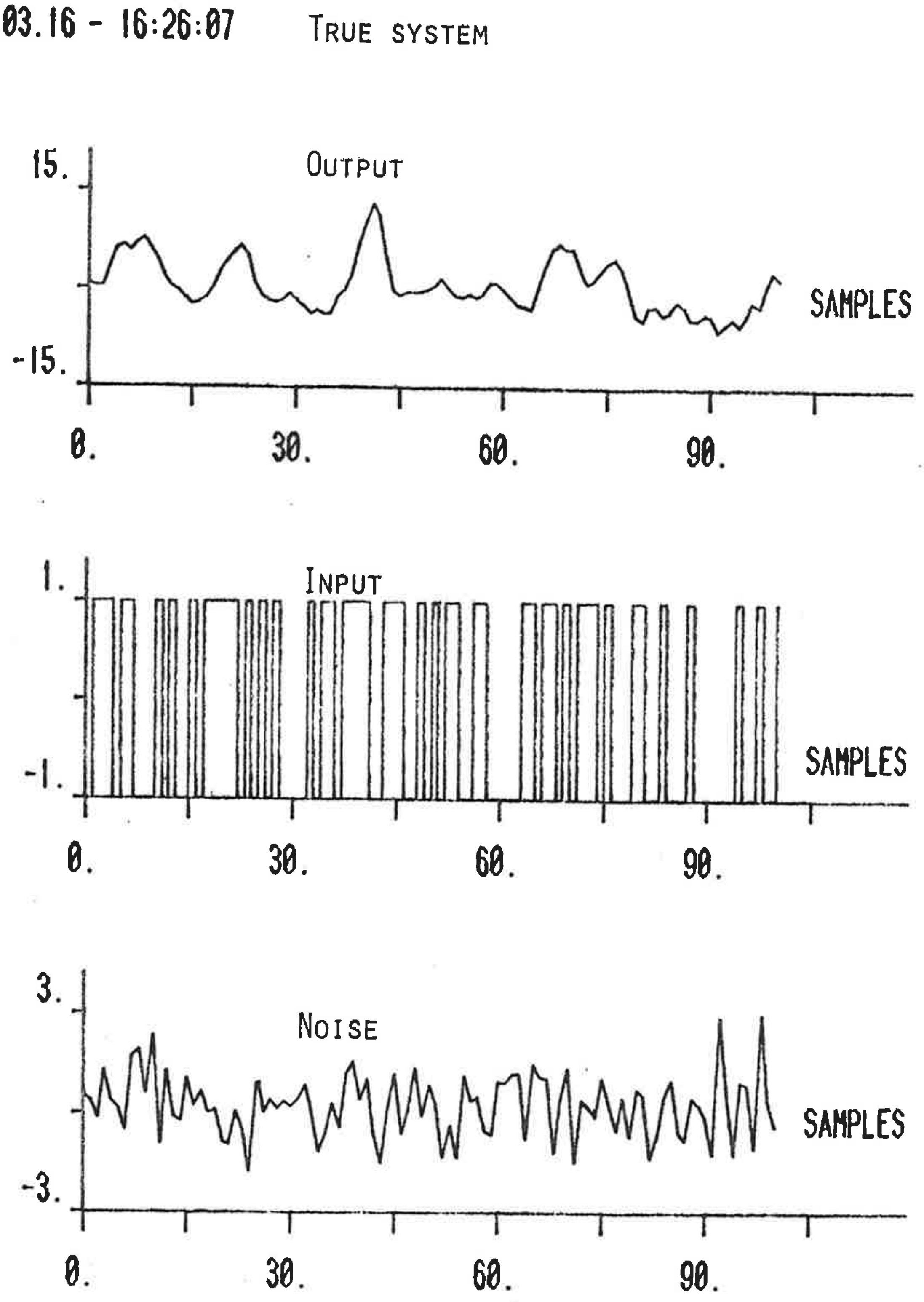
>>

RESULTAT AV F - TEST

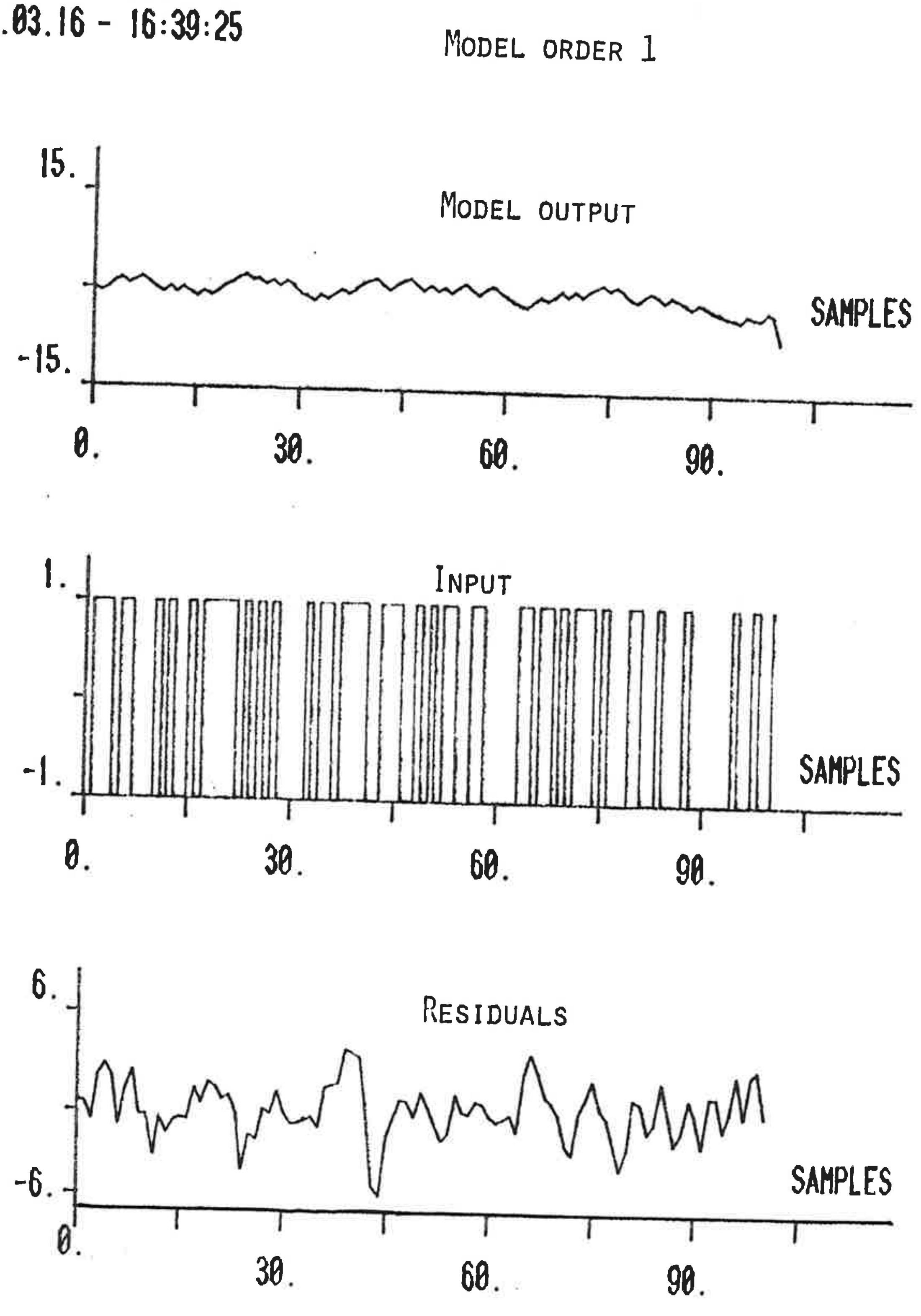
TESTKVANTITET VID ÖVERGÅNG I ORDNINGSTAL

FRÅN \ TILL	2	3
1	97	48
2	-	0.57

PLOT U(2)/(HP)U(1)/E
79.03.16 - 16:26:07



PLOT YI/(HP)U(I)/RESI
79.03.16 - 16:39:25



6:24

RESID RESI<LSI U (PAGE 1)
79.03.16 - 15:54:09

ORDER 1

VARIANCE OF THE RESIDUALS:
2.69185

NUMBER OF CHANGES OF SIGN
OF THE RESIDUALS: 37

5 PERCENT TOLERANCE LIMITS:
40 60

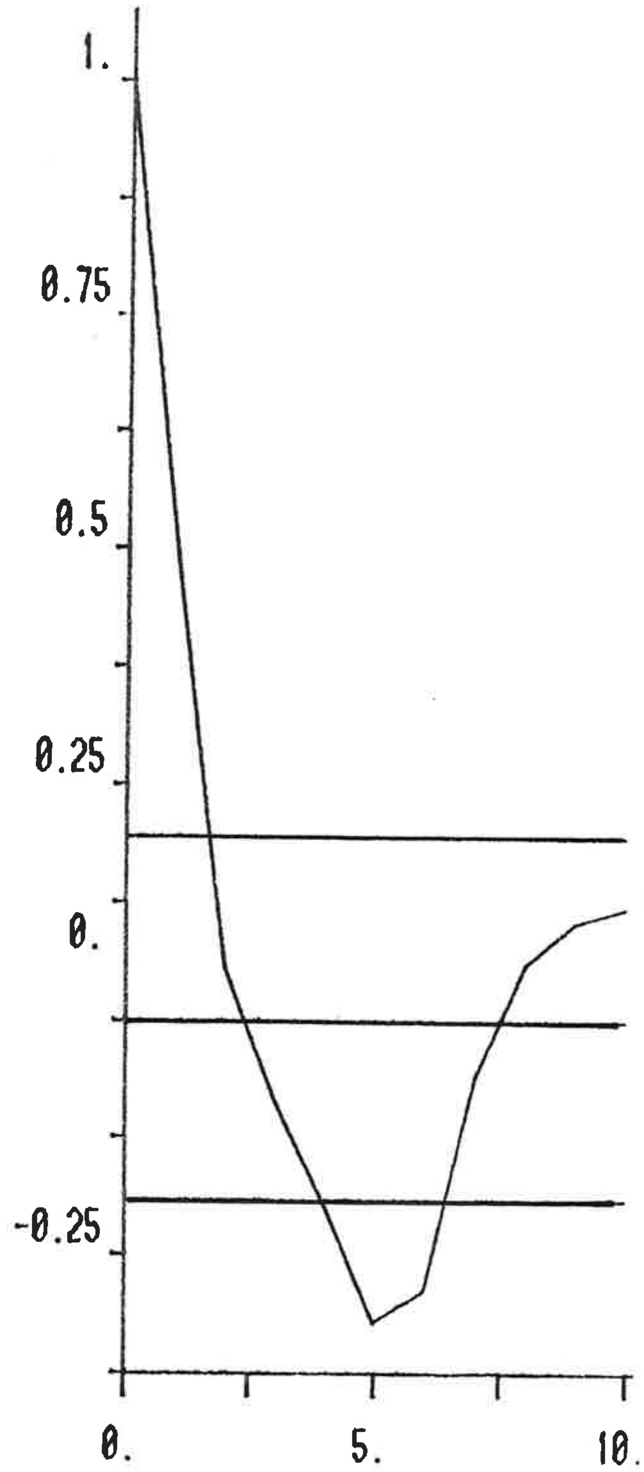
TEST OF INDEPENDENCE OF THE
RESIDUALS

E(RES(T)*RES(T+TAU))
FOR: 0 < TAU < 11

TEST QUANTITY: 51.9869
DEGREES OF FREEDOM: 10

TEST OF NORMALITY

TEST QUANTITY: 14.9345
DEGREES OF FREEDOM: 17



RESID RESI<LSI U (PAGE 2)
79.03.16 - 15:57:52

ORDER 1

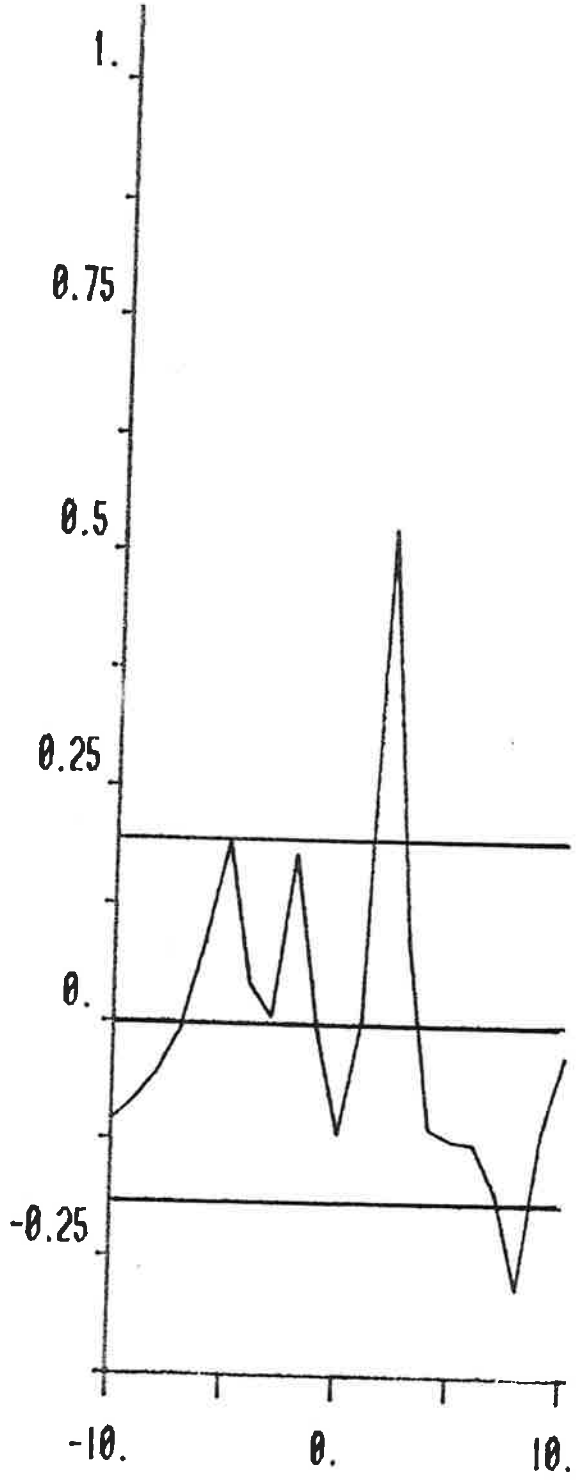
TEST OF INDEPENDENCE BETWEEN
RESIDUALS AND INPUT: 1

E(RES(T)*U(T+TAU))
FOR: 1 < TAU < 12

TEST QUANTITY: 51.2074
DEGREES OF FREEDOM: 10

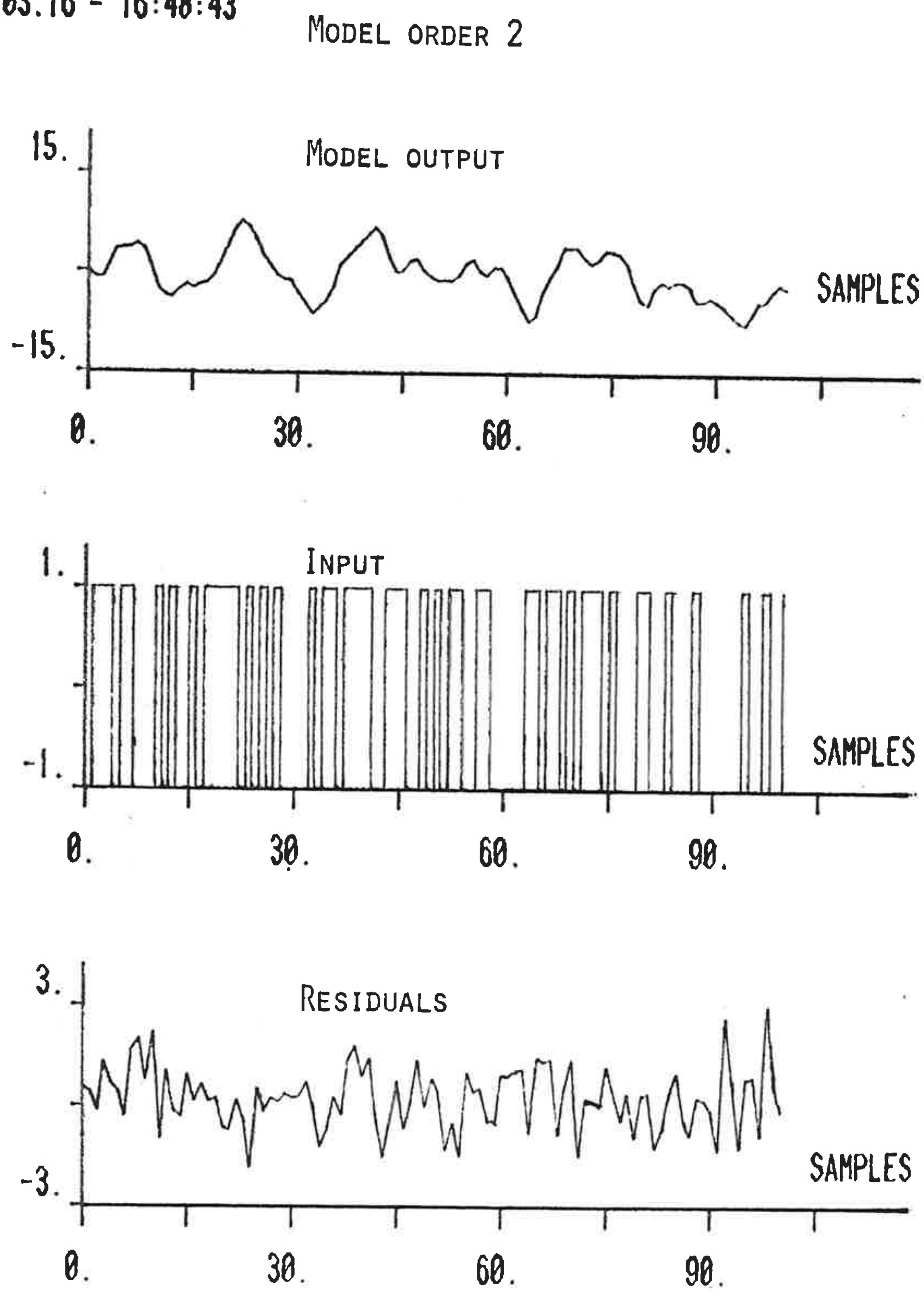
E(RES(T)*U(T+TAU))
FOR: -10 < TAU < 1
FOR: -10 < TAU < 1

TEST QUANTITY: 11.4430
DEGREES OF FREEDOM: 10



6:26

PLOT Y2/(HP)U(1)/RES2
79.03.16 - 16:48:43



RESID RES2<LS2 U (PAGE 1)
79.03.16 - 16:09:09

VARIANCE OF THE RESIDUALS:
.866043

NUMBER OF CHANGES OF SIGN
OF THE RESIDUALS: 52

5 PERCENT TOLERANCE LIMITS:
40 60

TEST OF INDEPENDENCE OF THE
RESIDUALS

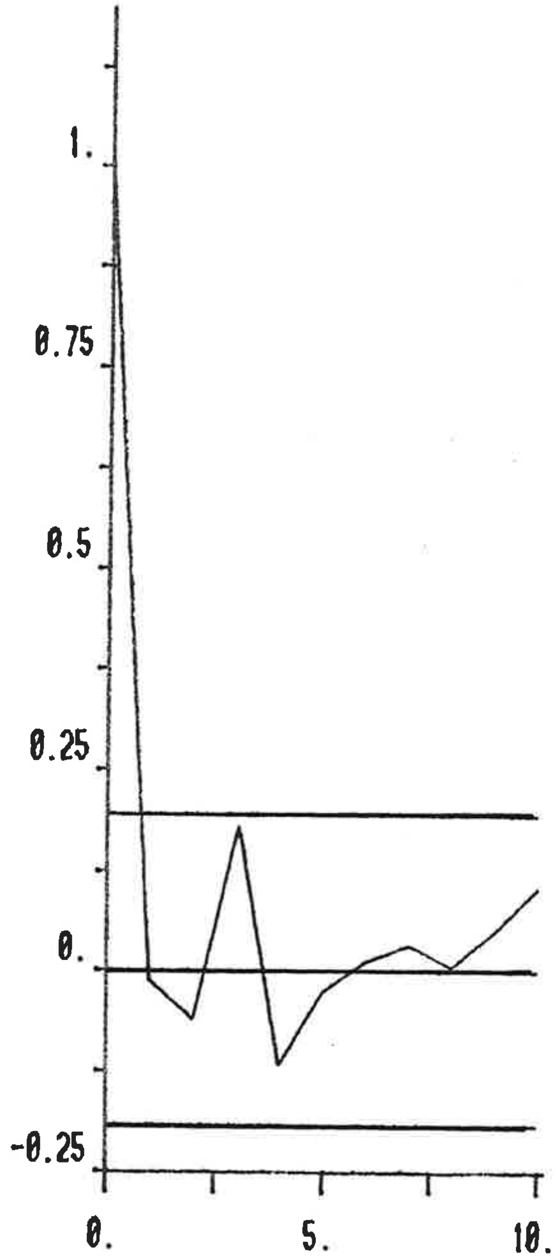
E(RES(T)*RES(T+TAU))
FOR: 0 < TAU < 11

TEST QUANTITY: 6.33243
DEGREES OF FREEDOM: 10

TEST OF NORMALITY

TEST QUANTITY: 10.6958
DEGREES OF FREEDOM: 17

ORDER 2



RESID RES2<LS2 U (PAGE 2)
79.03.16 - 16:12:01

TEST OF INDEPENDENCE BETWEEN
RESIDUALS AND INPUT: 1

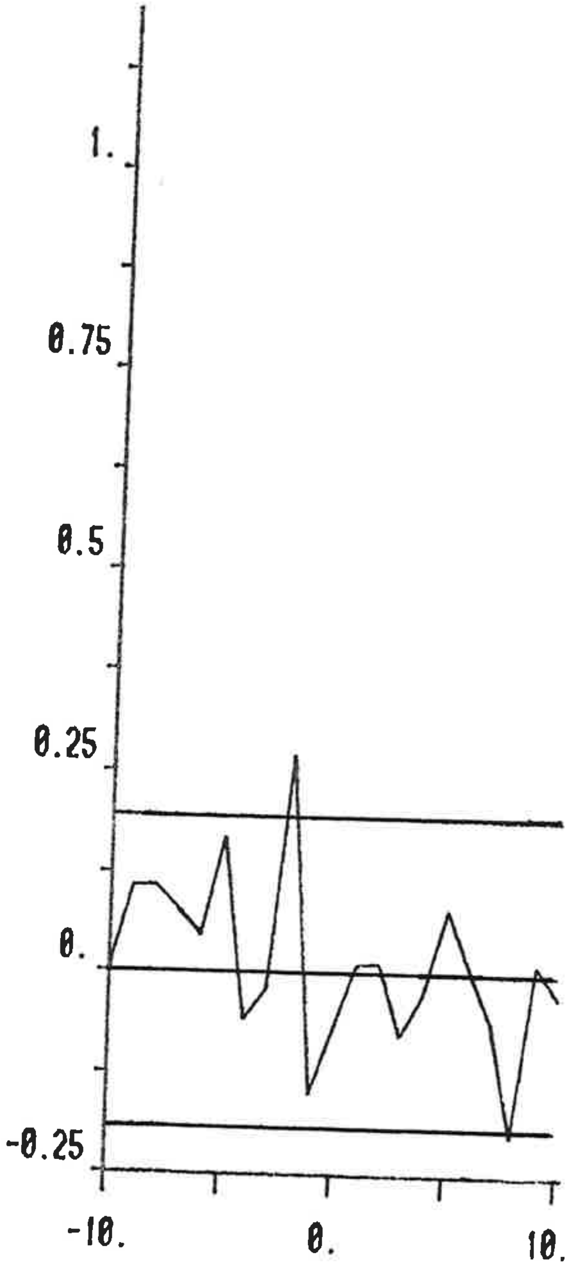
E(RES(T)*U(T+TAU))
FOR: 2 < TAU < 13

TEST QUANTITY: 6.57065
DEGREES OF FREEDOM: 10

E(RES(T)*U(T+TAU))
FOR: -10 < TAU < 1

TEST QUANTITY: 16.4841
DEGREES OF FREEDOM: 10

ORDER 2



MINSTA KVADRAT SKATTNINGAR

	Modellordning			
	1	2	3	RÄTT
$\hat{a}_1$	-0.89 ± 0.04	-1.40 ± 0.06	-1.43 ± 0.10	-1.50
$\hat{a}_2$	----	0.61 ± 0.06	0.63 ± 0.15	0.70
$\hat{a}_3$	----	----	0.007 ± 0.08	0.0
$\hat{b}_1$	0.72 ± 0.16	0.98 ± 0.10	0.98 ± 0.10	1.00
$\hat{b}_2$	----	0.53 ± 0.10	0.50 ± 0.14	0.50
$\hat{b}_3$	----	----	-0.13 ± 0.12	0.0
$\hat{\lambda}$	1.65 ± 0.12	0.95 ± 0.07	0.95 ± 0.07	1.0
V	138.03	46.450	45.908	
AIC	399.02	295.93	302.73	

6:30

Residualer / Modellfel

System:

$$y(t) + a\,y(t-1) = u(t-1) + e(t)$$

1) Residualer $\varepsilon(t)$

$$\boxed{\varepsilon(t) = y(t) + \hat{a}\,y(t-1) - u(t-1)}$$

$\varepsilon(t)$ = skattning av $e(t)$

2) Modellfel $\Delta y(t)$

Simulera skattade systemet:

$$x(t) + \hat{a}\,x(t-1) = u(t-1)$$

$$\boxed{\Delta y(t) = y(t) - x(t)}$$

Samband:

$$\Delta y(t) + \hat{a}\,\Delta y(t-1) = \varepsilon(t)$$

FÖRDELAR:

LÄTT ATT ANVÄNDA

GER DIREKT PULS ÖVERFÖRINGSFUNKTIONEN

MÖJLIGHET ATT TESTA ORDNINGSTAL

RÄKNINGARNA KAN GÖRAS REKURSIVA

NACKDELAR:

KAN GE ONÖDIGT HÖGA ORDNINGSTAL

7. MAXIMUM LIKELIHOOD METODEN

Bo Egardt

Maximum likelihood metoden beskrevs ursprungligen i

Åström K J and T Bohlin: Numerical Identification of Linear Dynamic Systems from Normal Operating Records. IFAC Symposium on Theory of Self-Adaptive Control Systems, Teddington, England, in Theory of Self-Adaptive Control Systems (Ed. P H Hammond), Plenum Press, New York (1966).

MAXIMUM LIKELIHOOD METODEN (ML)

1. MOTIVATION
2. BESKRIVNING
3. DYNAMISKA SYSTEM
4. ML MED IDPAC

1. MOTIVATION

PROBLEM: MK + FÄRGAT BRUS ⇒ BIAS

LÖSNING: ANDRA METODER

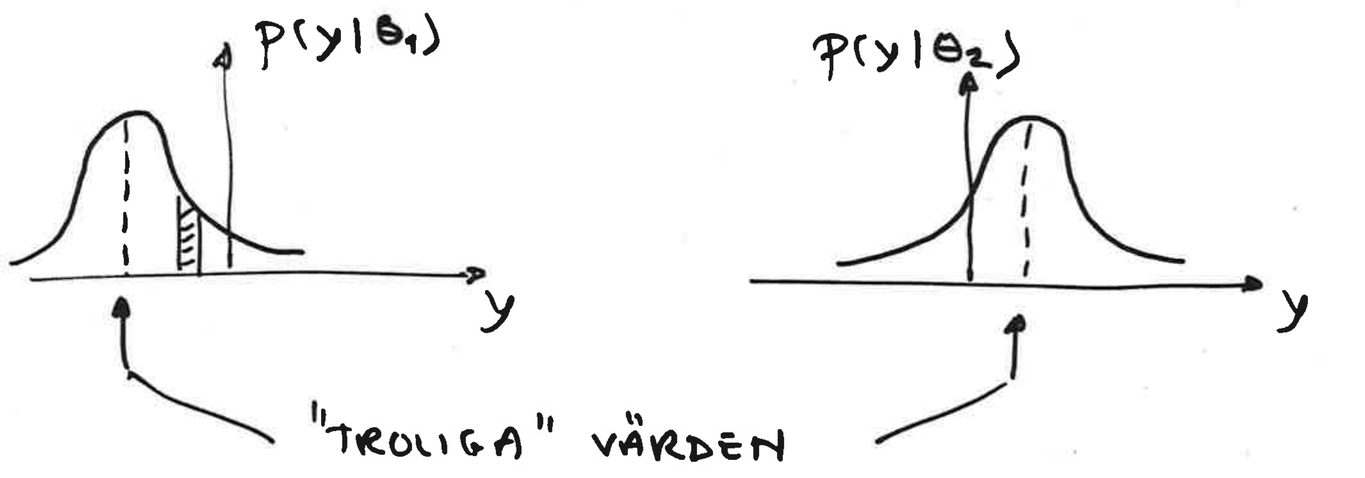
T.EX. MAXIMUM LIKELIHOOD (ML)

2. BESKRIVNING AV ML

- STATISTISK UTGÅNGSPUNKT
- Y STOKASTISK VARIABEL MED SANNOLIKHETSTÄTHET (FREKVENSFUNKTION) $p(y|\theta)$; θ OKÄND PARAMETERVEKTOR

EXEMPEL

Y SKALÄR S.V. ; θ ANTAR VÄRDENA θ_1 ELLER θ_2



MÄTS: $Y = 0$ ⟶ SKATTNING: $\theta = \theta_2$

ALLMÄN PRINCIP

MÄTNING y AV Y

LIKELIHOODFUNKTIONEN $L(y,\theta) = p(y|\theta)$

ML-SKATTNINGEN AV θ:

$$L(y,\hat{\theta}_{ML}) \geq L(y,\theta) \quad \text{alla } \theta$$

- ALLMÄNT
- OLINJÄRT
- EGENSKAPER: KONSISTENS $\hat{\theta}_{ML} \to \theta$
  EFFEKTIVITET

EXEMPEL

$Y = x\theta + w$ SKALÄR S.V. $\quad x$ KÄND, θ OKÄND
w $N(0,\sigma)$ σ KÄND

MÄTNING y

$$L(y,\theta) = p(y|\theta) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2\sigma^2}(y-x\theta)^2}$$

$$\max_\theta L(y,\theta) \Leftrightarrow \min_\theta (y-x\theta)^2 \Rightarrow \hat{\theta}_{ML} = \frac{y}{x}$$

jfr MK!

3. DYNAMISKA SYSTEM

MODELL: $y(t) + a_1 y(t-1) + \dots + a_n y(t-n) =$

$= b_1 u(t-1) + \dots + b_n u(t-n) +$

$+ e(t) + c_1 e(t-1) + \dots + c_n e(t-n)$

$A(q^{-1}) y(t) = B(q^{-1}) u(t) + C(q^{-1}) e(t)$

$\{e(t)\}$ oberoende $N(0, \sigma)$

$\theta^T = [a_1, \dots, a_n, b_1, \dots, b_n, c_1, \dots, c_n]$

OBSERVATIONER $y_1, \dots, y_N$

$L(y_1, \dots, y_N, \theta, \sigma) = p(y_1, \dots, y_N \mid \theta, \sigma) =$

$= \left((2\pi)^N \sigma^{2N}\right)^{-1/2} e^{-\frac{1}{2\sigma^2} \sum_{t=1}^{N} \varepsilon^2(t)}$

där $C(q^{-1}) \varepsilon(t) = A(q^{-1}) y(t) - B(q^{-1}) u(t)$

7:6

ML- ESTIMATET

$$* \begin{cases} \min_{\theta} \sum_{t=1}^{N} \varepsilon^2(t) \longrightarrow \hat{\theta}_{ML} \\ \hat{\sigma}^2_{ML} = \frac{1}{N} \sum_{t=1}^{N} \varepsilon^2_{\hat{\theta}_{ML}}(t) \end{cases}$$

- modell för bruset

- $\varepsilon(t) = \dfrac{A(q^{-1})\,y(t) - B(q^{-1})\,u(t)}{C(q^{-1})} = y(t) - \hat{y}(t)$

$\hat{y}(t)$ bästa prediktionen av $y(t)$

dvs * bra även om ej normalfördelning

BERÄKNINGAR

DATA: $y_1, \dots, y_N, u_1, \dots, u_N$

MODELL: $A(q^{-1})\,y(t) = B(q^{-1})\,u(t) + C(q^{-1})\,\varepsilon(t)$

$$y(t) = \theta^T \varphi(t) + \varepsilon(t)$$

$$\varphi^T(t) = [-y(t-1), \dots, -y(t-n), u(t-1), \dots, u(t-n), \varepsilon(t-1), \dots, \varepsilon(t-n)]$$

ML-SKATTNING:

$$\min V(\theta) = \frac{1}{2} \sum_{t=1}^{N} \varepsilon^2(t) \quad \text{m.a.p.} \quad \theta$$

t.ex. med Newton-Raphson:

$$\hat{\theta}_{k+1} = \hat{\theta}_k - V''^{-1}_{\hat{\theta}\hat{\theta}}(\hat{\theta}_k)\, V'^T_{\hat{\theta}}(\hat{\theta}_k)$$

$$V'_{\hat{\theta}}(\hat{\theta}) = \sum_{t=1}^{N} \varepsilon'_{\hat{\theta}}(\hat{\theta}, t)\, \varepsilon(t)$$

$$V''_{\hat{\theta}\hat{\theta}}(\hat{\theta}) = \sum_{t=1}^{N} \varepsilon'_{\hat{\theta}}(\hat{\theta}, t)^T\, \varepsilon'_{\hat{\theta}}(\hat{\theta}, t)$$

$$\varepsilon'_{\hat{\theta}}(\hat{\theta}, t) = \frac{1}{\hat{C}(q^{-1})}\, \varphi(t)$$

- <u>alla</u> data används i varje steg
- $V(\theta)$ olinjär i C-parametrarna;

  flera lokala minima möjliga

  $\Rightarrow$ olika startvärden!
- $C = 1 \Rightarrow$ MK
- $B = 0 \Rightarrow$ tidsserieanalys
- rekursiv variant approximativ

4. ML MED IDPAC

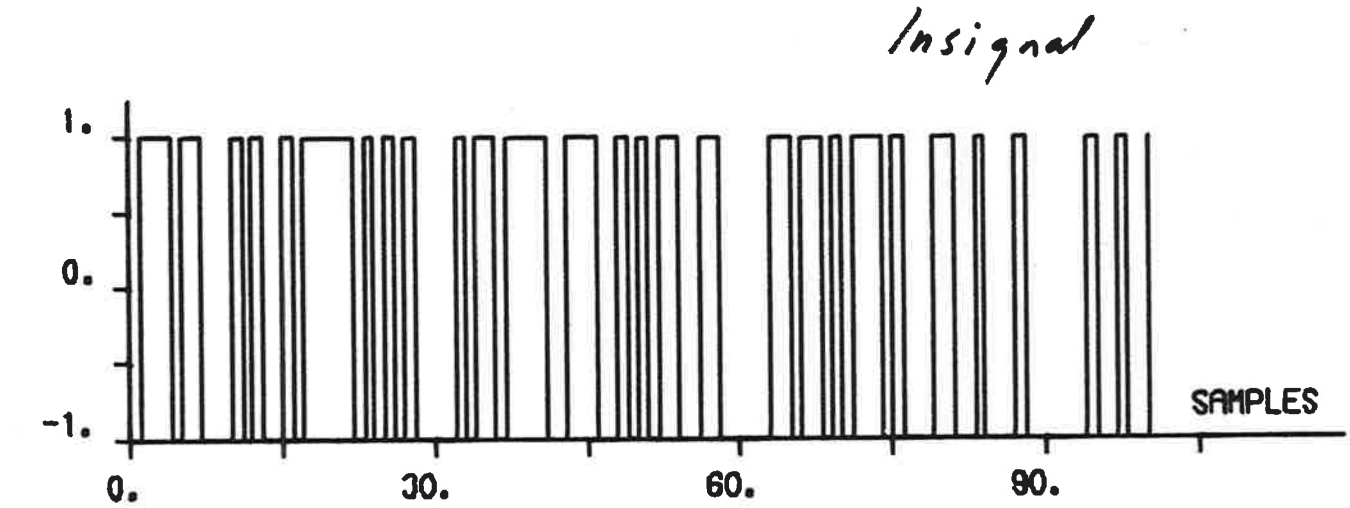
```
ML  SYST ← DATA  NO
                 └ MODELLORDNING
           └ DATAFIL
     └ SYSTEMFIL MED
         SKATTAD MODELL
```

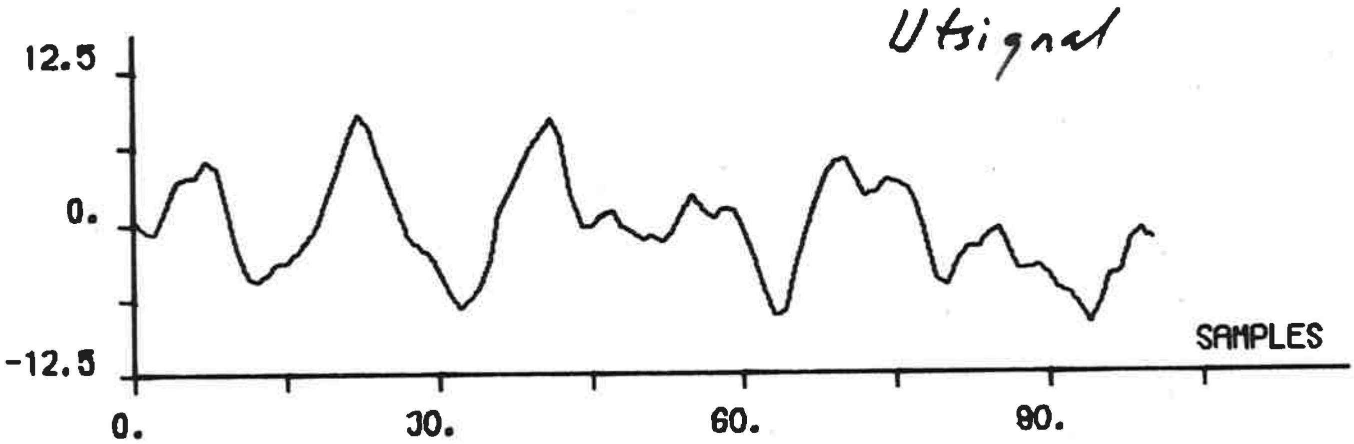
EXEMPEL

```
> PLOT (HP) DATA1(1)/DATA1(2)          " BILD 1
   > KILL
> ML ML1 ← DATA1  1
> RESID RES1 ← ML1 DATA1               " BILD 2
   > PAGE
   > KILL
> ML ML2 ← DATA1  2
> RESID RES2 ← ML2 DATA1               " BILD 3
   > PAGE                              " BILD 4
   > KILL
> DETER YM ← ML2 DATA1(1)
> PLOT YM/DATA1(2)                     " BILD 5
> STOP
```

7:10

1979-03-20 15:09:28 NR 1
HCOPY





1979-03-20 15.16.28 NR 2
HCOPY

Modell no 1

VARIANCE OF THE RESIDUALS:
1.34834

NUMBER OF CHANGES OF SIGN
OF THE RESIDUALS: 75

5 PERCENT TOLERANCE LIMITS:
85 113

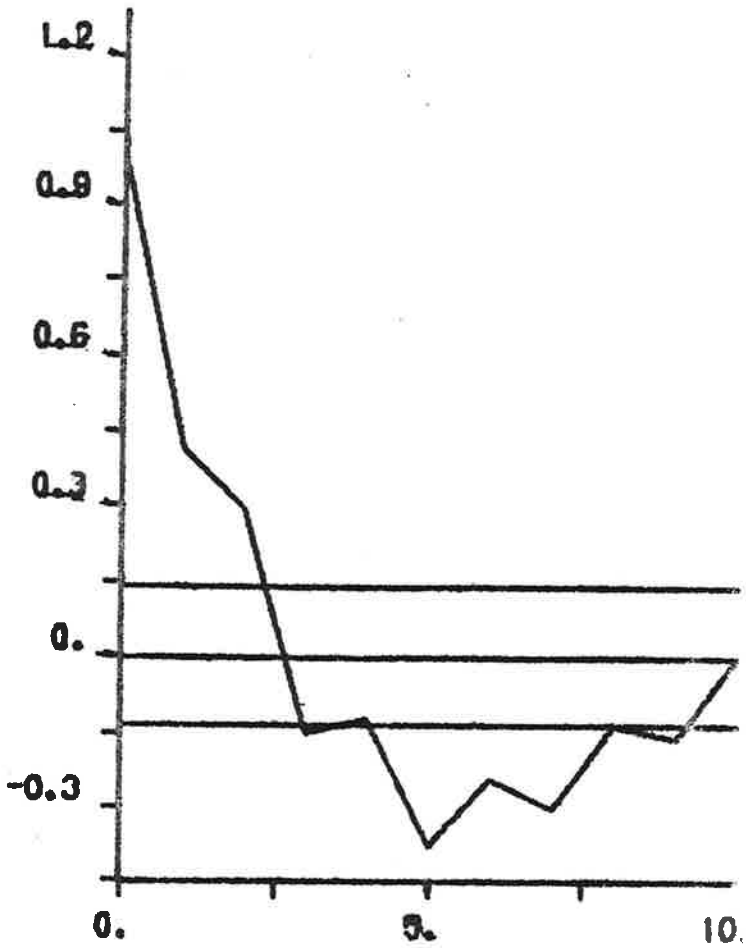
TEST OF INDEPENDENCE OF THE
RESIDUALS

E(RES(T)*RES(T+TAU))
FOR: 0 < TAU < 11

TEST QUANTITY: 127.343
DEGREES OF FREEDOM: 10

TEST OF NORMALITY

TEST QUANTITY: 19.9319
DEGREES OF FREEDOM: 17



7:12

1979-03-20 15:21:22 NR 3
HCOPY

Modell n:2

VARIANCE OF THE RESIDUALS:
5.183488E-02

NUMBER OF CHANGES OF SIGN
OF THE RESIDUALS: 90

5 PERCENT TOLERANCE LIMITS:
83 113

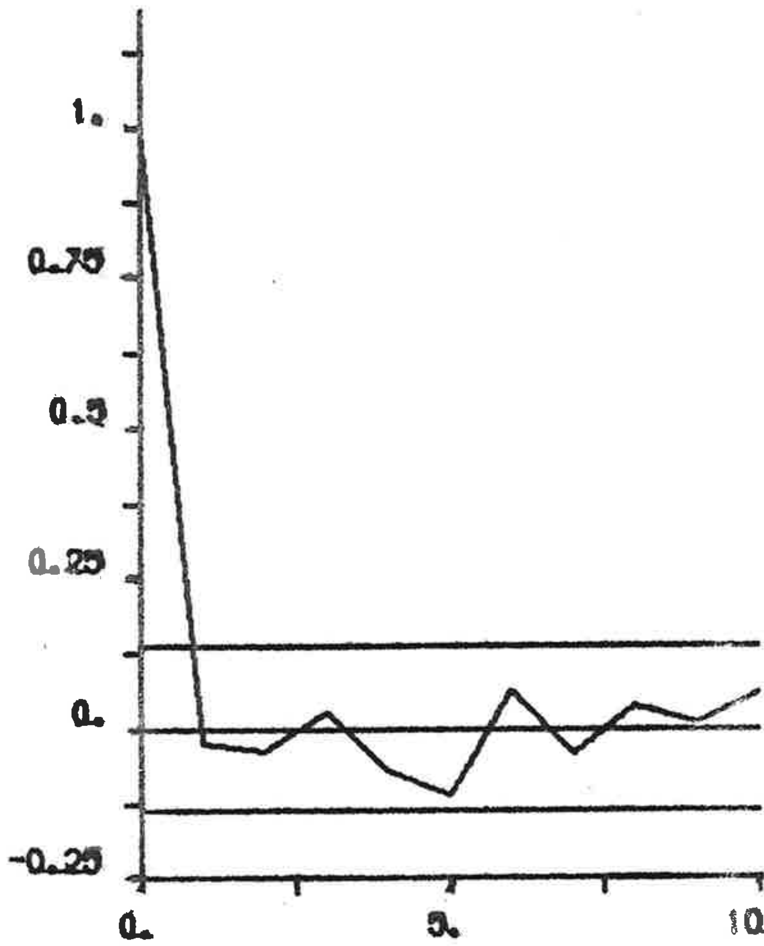
TEST OF INDEPENDENCE OF THE
RESIDUALS

E(RES(T)*RES(T+TAU))
FOR: 0 < TAU < 11

TEST QUANTITY: 6.33141
DEGREES OF FREEDOM: 1

TEST OF NORMALITY

TEST QUANTITY: 12.6135
DEGREES OF FREEDOM: 17



1979-03-20 15.22.08 NR 4
HCOPY

Modell n = 2

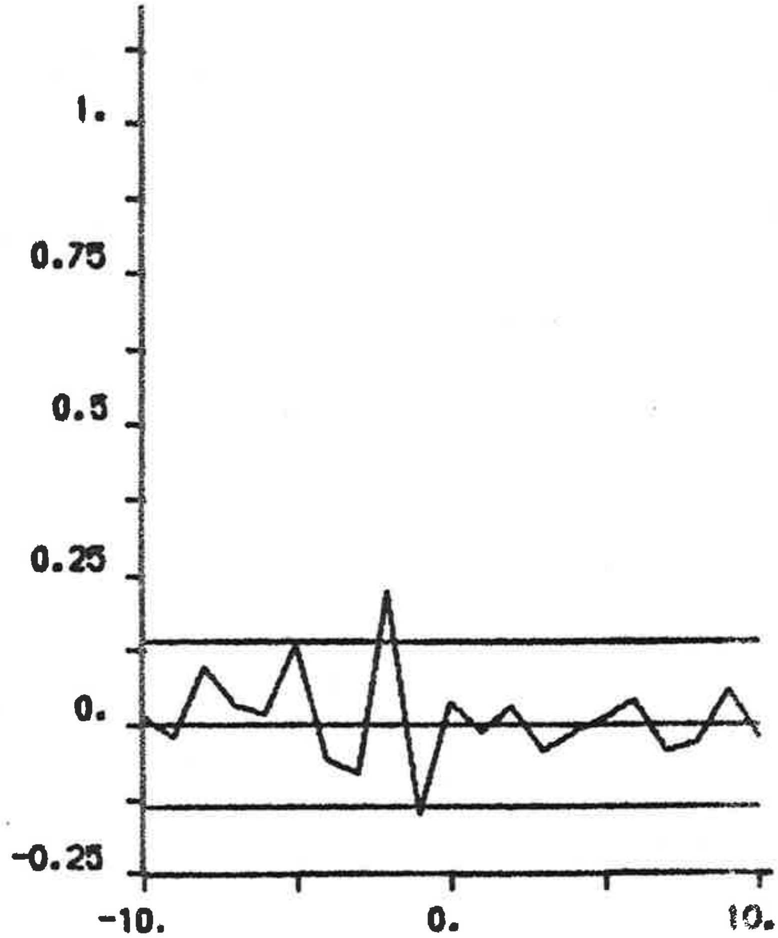
TEST OF INDEPENDENCE BETWEEN
RESIDUALS AND INPUT: 1

E(RES(T)*U(T+TAU))
FOR: 2 < TAU < 13

TEST QUANTITY: 3.91417
DEGREES OF FREEDOM: 10

E(RES(T)*U(T+TAU))
FOR: -10 < TAU < 1

TEST QUANTITY: 22.1189
DEGREES OF FREEDOM: 10



8. EXPERIMENTPLANERING. PRELIMINÄR DATAANALYS. MODELLVERIFIKATION.

Gustaf Olsson och Ivar Gustavsson

En hel del av dessa problem finns diskuterade mera i detalj i kapitel 3 av

I Gustavsson: Survey of applications of identification in chemical and physical processes. Automatica 11, 3-24, 1975.

EXPERIMENT-PLANERING

- SYFTE MED MODELLEN
- FÖRKUNSKAPER
- INSTRUMENT
- STÄLLDON
- ÅTERKOPPLING

E X P E R I M E N T P L A N E R I N G

SYFTET MED MODELLEN

- REGLERING
- ÖKAD PROCESSKUNSKAP
- PROCESSDIAGNOS

FÖRKUNSKAPER OM PROCESSDYNAMIKEN

I N S T R U M E N T

VILKA MÄTSIGNALER SKALL ANVÄNDAS VID REGLERING?

ANDRA RELEVANTA MÄTNINGAR

FINNES MÄTINSTRUMENT?

- LABTESTER
- MANUELLA OBSERVATIONER

NORMALA VARIATIONER PA INSTRUMENTUTSLAG

INSTRUMENTEGENSKAPER

- DYNAMIK (ÄR INSTRUMENTEN SNABBA NOG?)
- BRUS
- DRIFT
- KALIBRERING

KABLAR (LÄNGD, STRÖM, SPÄNNING ?)

ENKLA REGISTRERINGAR

- SKRIVARE OSCILLOSKOP

A/D-OMVANDLARE (KVANTISERINGSFEL?)

S T Ä L L D O N

KAN DE MANÖVRERAS?

- MANUELLT?
- AUTOMATISKT?

STÄLLDONSDYNAMIK

- BEGRÄNSNINGAR I STÄLLDON
- BEGRÄNSNINGAR I STÄLLDONSÄNDRING

NOGGRANNHET I STÄLLDONEN

REGISTRERING

- STÄLLDON
- VERKLIG STYRSIGNAL TILL PROCESSEN

BEGRÄNSNINGAR I TILLATEN STYRSIGNAL TILL PROCESSEN

VAD KOSTAR EN STÖRNING? (SPEC. PROCESSTUDIER)

SYFTAR IDENTIFIERINGEN TILL REGLERING?

- INKLUDERING AV STÄLLDONSDYNAMIK

Å T E R K O P P L I N G

FINNES REGULATORER I PROCESSEN?

KAN REGULATORERNA KOPPLAS UR?

FINNES NATURLIGA ÅTERKOPPLINGAR (T.Ex. RECIRKULATION)?

NATURLIGA STÖRNINGAR GER EJ GOD NOGGRANNHET

IDENTIFIERBARHET

IDENTIFIERING IBLAND BÄTTRE OM ÅTERKOPPLING FINNES

ENKLA EXPERIMENT

- enkla insignaler
- vad kan undersökas?
- vilken kunskap får man?
- kontakt med personal!

ENKLA EXPERIMENT

KOMPLEXITET I EXPERIMENTBETINGELSER MATCHAS MOT
BERÄKNINGSARBETE

LOGGNING UNDER NORMALDRIFT

STEGSTÖRNINGAR

IMPULSSTÖRNINGAR

KORRELATIONSANALYS

ENKLA EXPERIMENT

UNDERSÖK SPECIELLT:

- ENKLA ORSAKS-VERKANSAMBAND
- OLINJÄRITETER
- DOMINERANDE TIDSKONSTANTER
- BRUSNIVÅER
  - BRUM
  - LÅGFREKVENT BRUS
- STATIONARITET
  - I PROCESSEN
  - I INSTRUMENT
- TIDSFÖRDRÖJNINGAR
- TEST AV UTRUSTNINGEN
- FINNS OLIKA DRIFTSFALL?
- KOPPLINGAR I SYSTEMET (FLERVARIABELT?)
- DRIFTSJOURNALER
  - HAR KALIBRERINGAR GJORTS UNDER EXPERIMENTET?
  - MANUELLA INGREPP I PROCESSEN?

E N K L A E X P E R I M E N T (FORTS)

INFÖR NÄSTA EXPERIMENTFAS, KUNSKAP OM:

SAMPLINGSINTERVALL

FILTRERING AV SIGNALER

DRIFTSPROBLEM

- INSTRUMENT
- PROCESS

TILLÅTNA NIVÅER I INSIGNAL

- MED HÄNSYN TILL OLINJÄRITETER
- " " " BRUSNIVÅER
- " " " PROCESSBETINGELSER

TALA MED PERSONALEN!!

SYFTE MED EXPERIMENTEN

ERFARENHETER FRÅN EXPERIMENTEN

OPERATÖRERNAS ERFARENHETER

* VAL AV INSIGNAL
* VAL AV SAMPLINGSHASTIGHET
* VAL AV EXPERIMENTLÄNGD

INSIGNAL

- FORM

FREKVENSANALYS	SINUS
TRANSIENTANALYS	STEG etc
KORRELATIONSANALYS	PRBS
PARAMETRISKA METODER	

- AMPLITUD

NOGGRANNHET $\sigma^2 \sim \frac{1}{\text{INSIGNALEFFEKT}}$

KONFLIKT: GODTAGBAR PRODUKTION
LINJÄRITET

TUMREGEL: INSIGNALENS INVERKAN SKALL ÅTMINSTONE SKÖNJAS I UT-SIGNALEN

- FREKVENSKARAKTERISTIK

INSIGNALEN SKALL HA EFFEKT INOM HELA DET FREKVENSOMRÅDE MAN VILL STUDERA

Jfr $\hat{G}(i\omega) = \dfrac{\hat{\Phi}_{uy}(i\omega)}{\hat{\Phi}_u(\omega)}$

BEGREPPET "PERSISTENTLY EXCITING"

- OPTIMALA INSIGNALER

PRBS (PSEUDO RANDOM BINARY SEQUENCE) 8:13

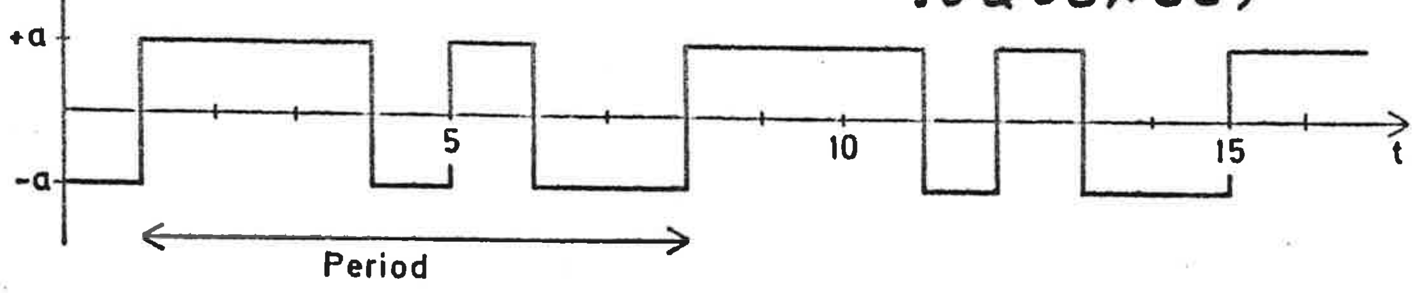


Fig.1. PRBS-signal med N=7

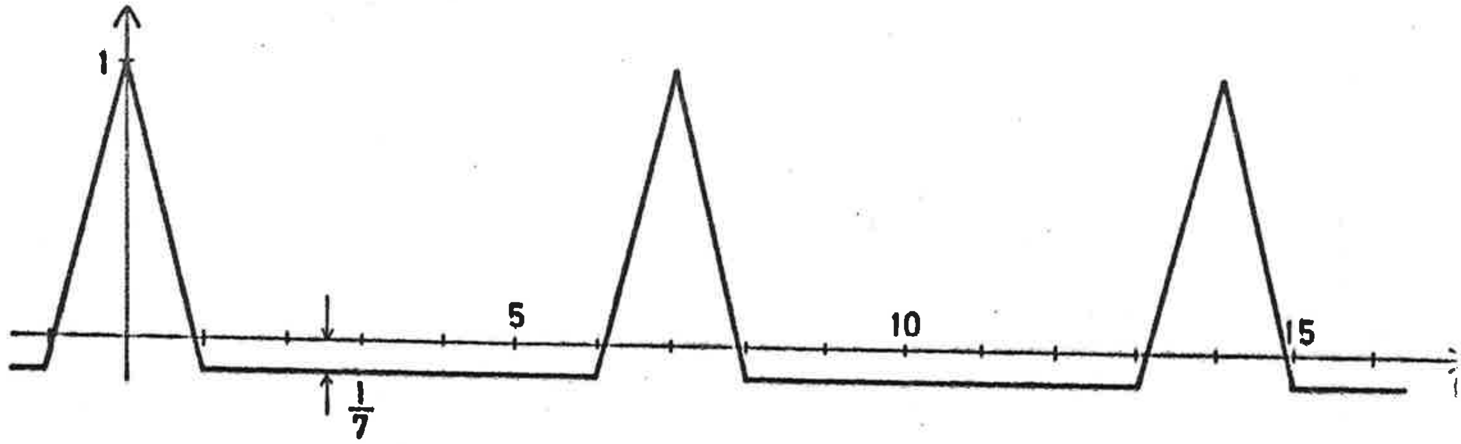


Fig.2. Autokorrelationsfunktionen för PRBS-signalen i fig.1

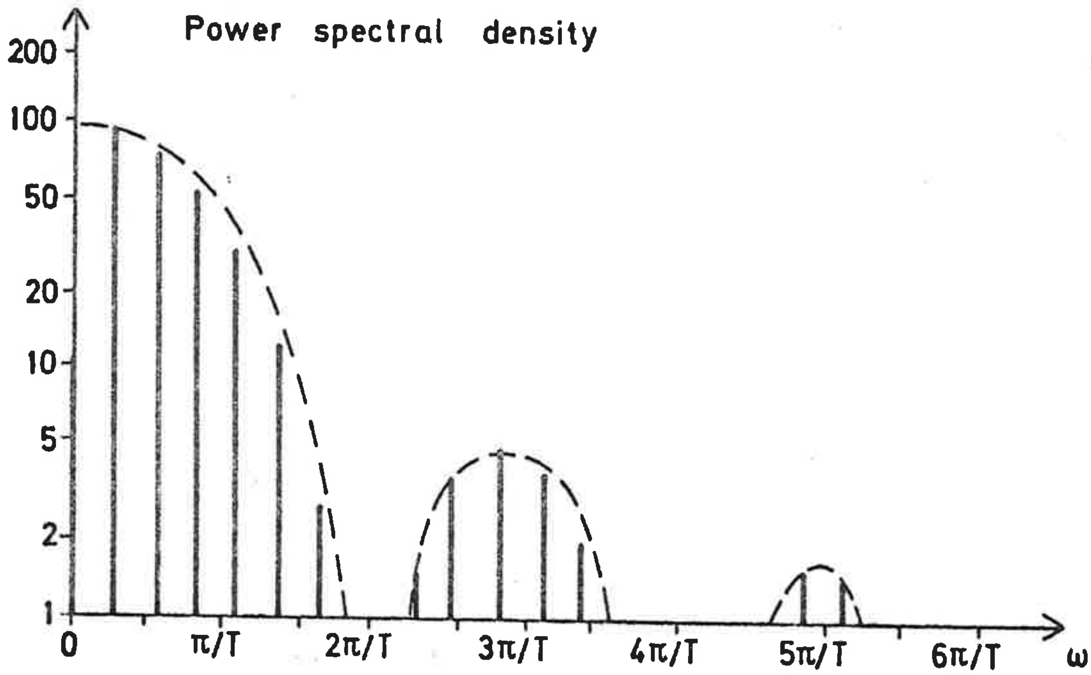


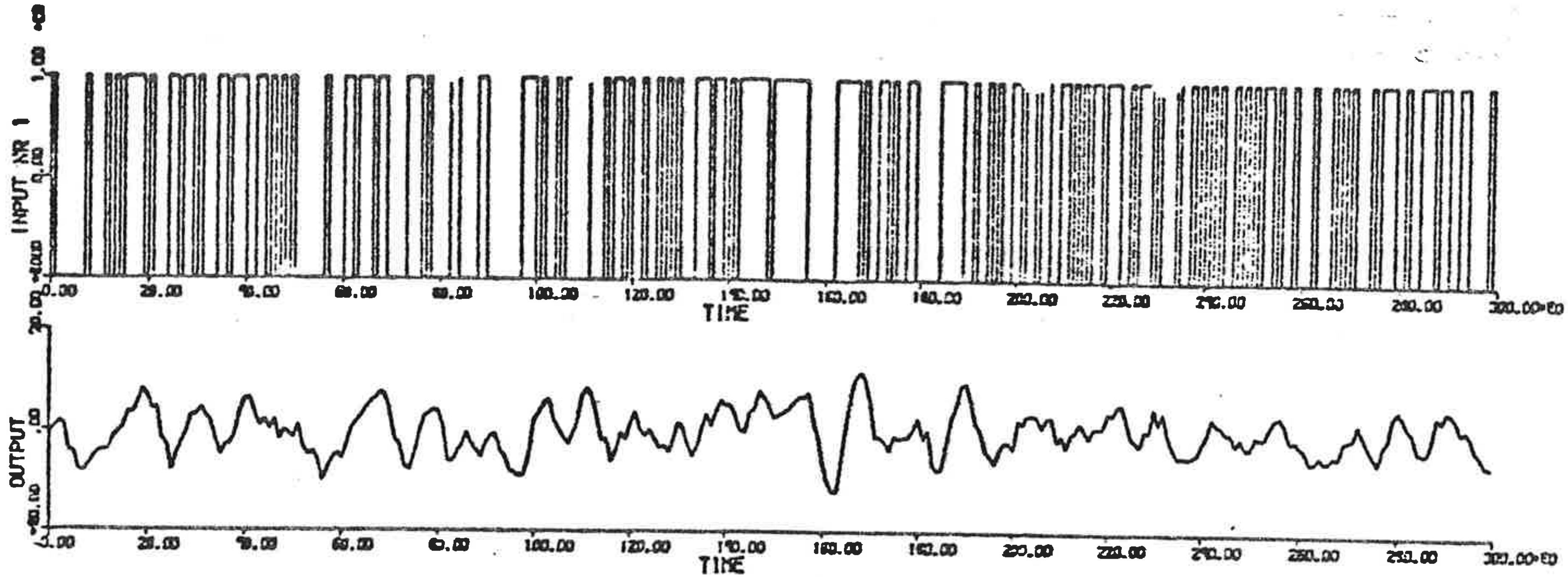
Fig.3. Effektspektrum för PRBS-signalen i fig. 1 (det diskreta linjespektrat). Streckad kurva anger ungefär envelopen till det diskreta linjespektrat för varierande N.
Spektrat ges av formeln

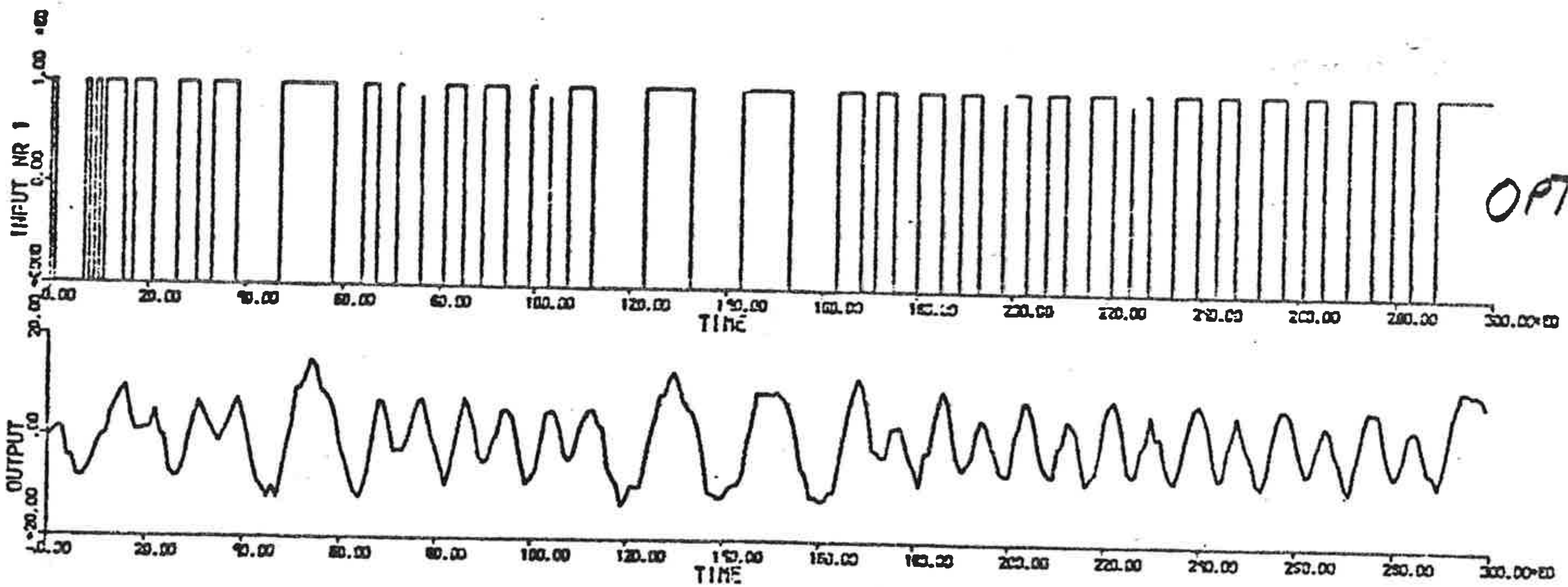
$$\Phi(f) = \frac{1}{N^2}\,\delta(f) + \sum_{r=-\infty}^{\infty} \frac{N+1}{N^2} \left(\frac{\sin\frac{r\pi}{N}}{r\pi} \right)^2 \delta\left(f - \frac{r}{N\cdot T}\right) \quad r=0,1,\ldots$$

EXEMPEL:

$$y(t) = \frac{\overset{b_1}{1.}\, q^{-1} + \overset{b_2}{0.5}\, q^{-2}}{1. - \underset{a_1}{1.5}\, q^{-1} + \underset{a_2}{0.7}\, q^{-2}} + e(t) \qquad N = 500$$

	PRBS	"Optimal"
a_1	-1.495 ± 0.008	-1.500 ± 0.005
a_2	0.698 ± 0.006	0.702 ± 0.004
b_1	0.976 ± 0.043	0.981 ± 0.026
b_2	0.551 ± 0.058	0.525 ± 0.035
c_1	-1.523 ± 0.025	-1.528 ± 0.028
c_2	0.728 ± 0.025	0.728 ± 0.028


INPUT NR 1
OUTPUT
TIME
0.00
20.00
40.00
60.00
80.00
100.00
120.00
140.00
160.00
180.00
200.00
220.00
240.00
260.00
280.00
300.00*E0


INPUT NR 1
OUTPUT
TIME
0.00
20.00
40.00
60.00
80.00
100.00
120.00
140.00
160.00
180.00
200.00
220.00
240.00
260.00
280.00
300.00*E0

OPTIMAL

8:15

EXPERIMENTLÄNGD

NOGGRANNHET $\sigma^2 \sim \frac{1}{\text{EXPERIMENTLÄNGD}}$

PRAKTISKA BEGRÄNSNINGAR:

- ✓ DRIFT
- ✓ PROCESSVARIATIONER
- ✓ MÄT- OCH REGISTRERING UTR.
- ✓ PÅLAGDA STÖRNINGAR SÅ KORT TID SOM MÖJLIGT
- ✓ DATAEKONOMI

TUMREGEL: MINST 5-10 GÅNGER LÄNGSTA TIDSKONSTANTEN

SAMPLINGSHASTIGHET

PROBLEM: ALIASING T.EX.

VAD BESTÄMMER?

- PROCESS-, STÖRNINGSKARAKTERISTIK
- MÄT- OCH REGISTRERINGSUTR
- REGLERÄNDAMÅL
- DATAEKONOMI

ETT EXPERIMENT GER INFORMATION I MAXIMALT 2 DEKADER I FREKVENS-PLANET.

"LÅNGA" TIDSKONSTANTER ~ INTEGRATORER

"KORTA" TIDSKONSTANTER KAN INTE DETEKTERAS

TUMREGEL:

$$h = 0.5 - 1 \; T_{min}$$

T_{min} = KORTASTE TIDSKONSTANTEN

PRELIMINÄR DATAANALYS

PLOTTA DATA		PLOT
TAG BORT OUTLIERS		PLMAG M FL
STATISTIK		STAT
SKALNING		SCLOP
	TILL INGENJÖRSSTORHETER	
	TILL SAMMA STORLEKSORDNING	
SÄTT IHOP SIGNALER	(EX FLÖDE X CONC)	VECOP
DELA UPP TIDSSERIER		CUT
SÄTT IHOP TIDSSERIER		CONC
TAG VART N:TE VÄRDE		PICK
TIDSFÖRDRÖJNINGAR		SLIDE
FILTRERING		
HÖGPASS	DIFFERENTIERING	VECOP SLIDE
	TRENDBORTTAGNING	TREND
	LÅGFREKVENT DRIFT	FILT
LÅGPASS	BRUS	FILT
ORSAKS-VERKAN SAMBAND		
	PREWHITE	ASPEC, ACOF
	KORRELATIONSANALYS	CSPEC, CCOF M M
	SPEKTRA	

FARA: KORRELATION MELLAN TVÅ VARIABLER BETYDER INTE NÖDVÄNDIGTVIS ORSAKSSAMBAND (KAUSALITET)

M O D E L L V E R I F I K A T I O N

SYFTET MED MODELLEN

INSIGNAL-UTSIGNALMODELL

$$A\,Y = \sum B_I\,U_I + \lambda \cdot C\,e$$

JÄMFÖR MED FYSIKALISKA MODELLER

TIDSKONSTANTER

NOLLSTÄLLEN

ORDNINGSTAL

KRITERIUM

PREDIKTIONSFELET e

$$V(e) = \sum_{0}^{N} h(e(t))$$

MODELLVERIFIKATION 2

TEST AV RESIDUALER

RESIDUALER SKALL VARA — OBEROENDE STOK VAR — RESID

NORMALA

MEDELVÄRDE 0, VARIANS 1

RESIDUALER OCH INSIGNAL OBEROENDE

TEST AV MODELLFEL

VERKLIG UTSIGNAL JÄMFÖRES MED MODELLENS UTSIGNAL — DETER, DSIM

TEST AV ORDNINGSTAL

F-TEST

AKAIKE AIC

TEST AV PARAMETRARS NOGGRANNHET

PARAMETRARNAS VARIANS — ML LS

KAN PARAMETRAR SÄTTAS = 0?

MONTE CARLO SIMULERING — RANPA

KÄNSLIGHET FÖR PARAMETETVARIATIONER

GEMENSAMMA FAKTORER I A, B, C ?

MODELLVERIFIKATION 3

TEST AV STATIONARITET

VERIFIERING PÅ OLIKA DATAMÄNGDER — DSIM, DETER

SIMULERING — DSIM

VERKLIGA INDATA — DETER

STEG

PULS

BRA ÖVERENSSTÄMMELSE FÖR KORTA ELLER LÅNGA TIDER?

(JFR MED EXP.LÄNGD OCH SAMPLINGSINTERVALL)

"OVERFITTING"

TESTNING PÅ OLIKA DATAMÄNGDER

TEST I REGLERKRETS

SLUTLIGA TESTET

9. PROCESSIDENTIFIERING. REPETITION OCH UTBLICKAR.

K J Åström

PROCESSIDENTIFIERING

1. INLEDNING – ÖVERSIKT

2. ICKE-PARAMETRISKA METODER

   FREKVENSANALYS
   TRANSIENTANALYS
   KORRELATIONSANALYS

   DATORÖVNING

3. INTERAKTIVA PROGRAM

   ALLMÄNT
   IDPAC

   DATORÖVNING – DATAANALYS

4. PARAMETRISKA METODER

   MINSTA KVADRATMETODEN
   MAXIMUM LIKELIHOOD

   DATORÖVNING

5. PRAKTISKA SYNPUNKTER

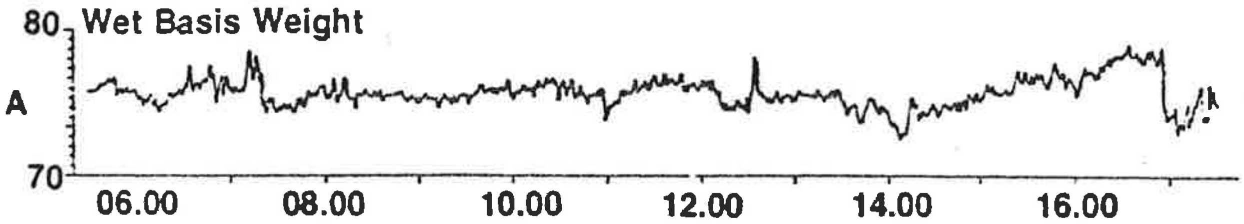
   EXPERIMENTPLANERING
   VALIDERING

6. AVSLUTNING – UTBLICKAR

   ANDRA METODER
   ADAPTIV REGLERING

ASSESSMENT OF BENEFITS OF CONTROL

DATALOGGING:

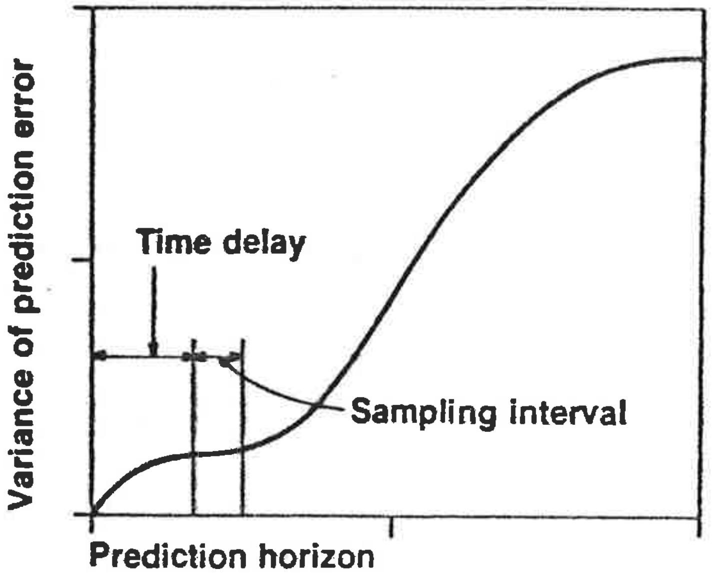


PROCESS IDENTIFICATION: PROCESS MODEL

PREDICTION

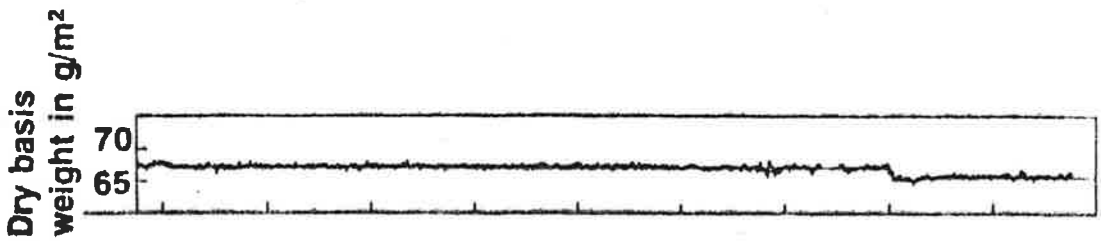
ERROR ANALYSIS

HEDGE

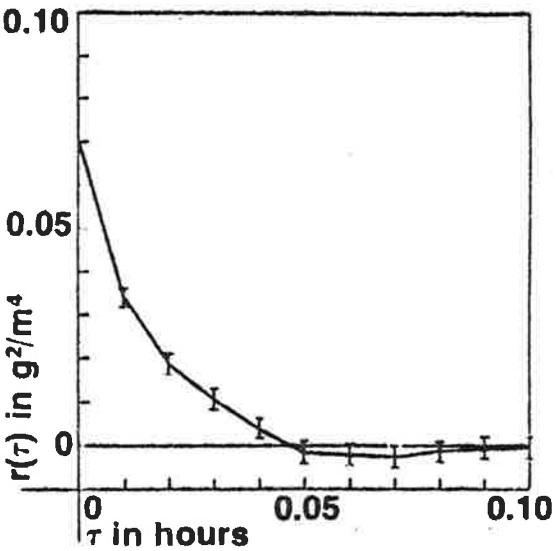


ASSESSMENT OF NEED FOR RETUNING OF MINIMUM VARIANCE CONTROL

LOG CONTROLLED OUTPUT DURING NORMAL OPERATION



CALCULATE COVARIANCE FUNCTION OF OUTPUT
(COV Y)



USE KNOWLEDGE OF PROCESS DEAD TIME AND SAMPLING PERIOD TO TEST IF COVARIANCE FUNCTION SATISFIES MINIMUM VARIANCE CONDITIONS

MODELLBYGGE

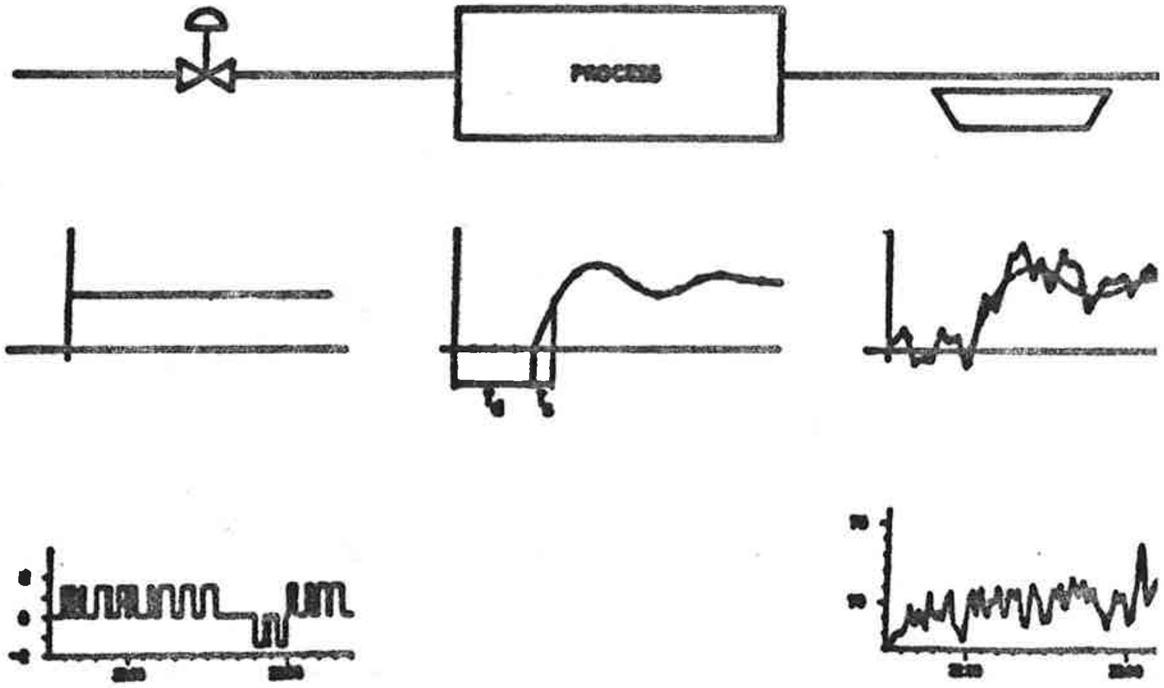
TVÅ VÄGAR

GRUNDEKVATIONER

KONSERVERING AV
MASSA
ENERGI
RÖRELSEMÄNGD

MATERIALEKVATIONER

PROCESSIDENTIFIERING

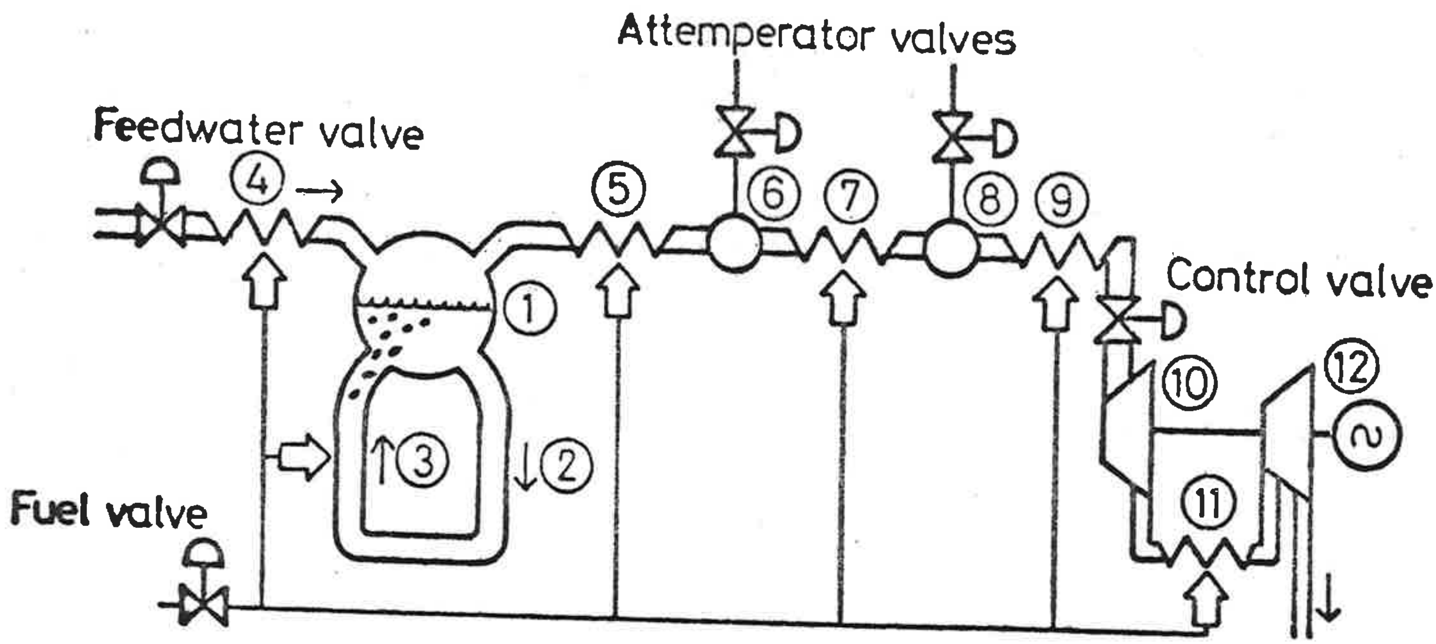


KOMBINATION ?

BOILER MODELING AND CONTROL GOALS

LOCAL BOILER CONTROL

THE BOILER AS A POWER SYSTEM COMPONENT



Inputs
- Fuel flow
- Feed water
- Coolant flows
- Control valve

Outputs
- Dome pressure
- Dome temprature
- Dome level
- Active power
- Steam flow

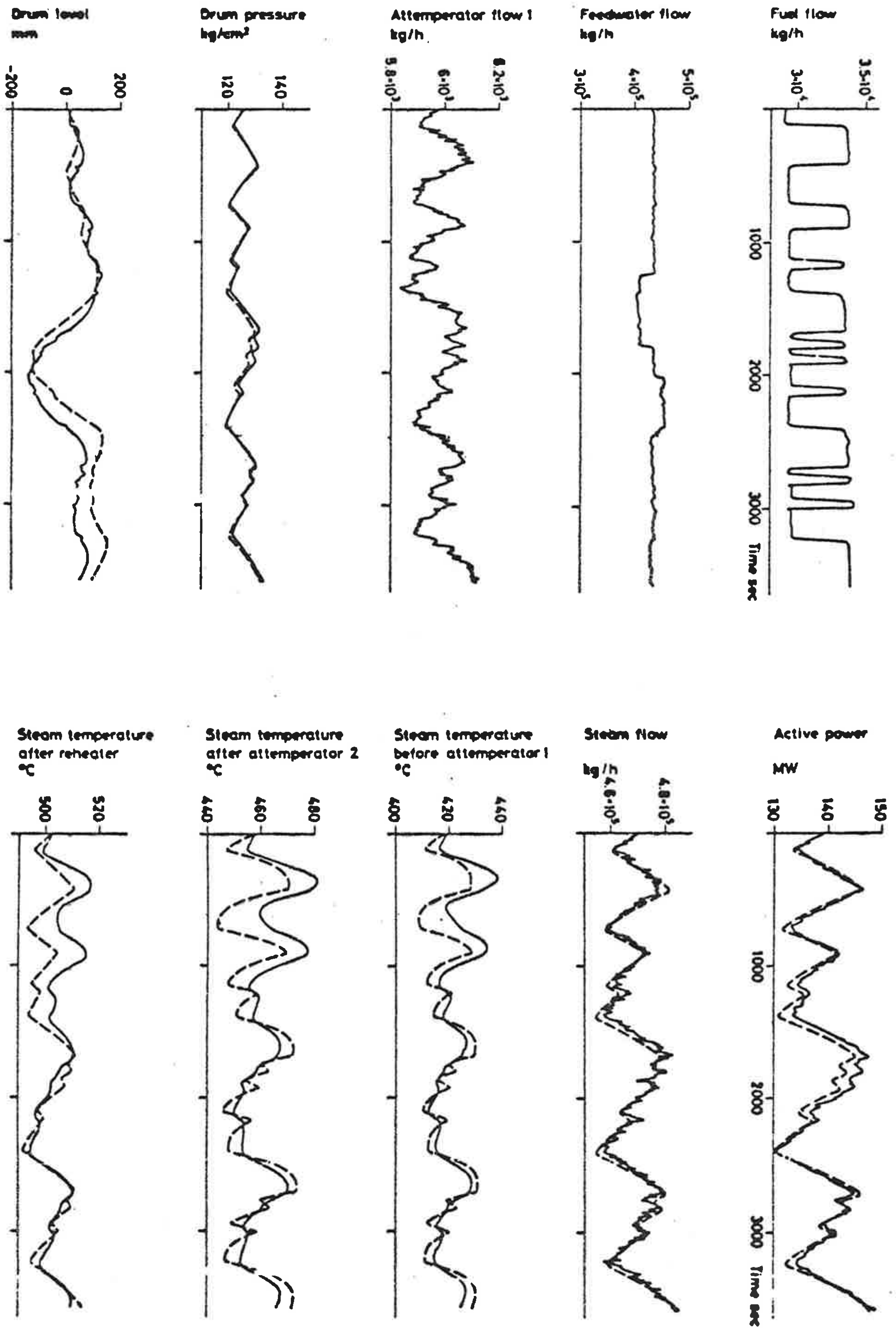
MODELLING EXAMPLE

$$\dot{x} = \begin{bmatrix} a_{11} & 0 & a_{13} & a_{14} & a_{15} \\ a_{21} & 0 & a_{23} & a_{24} & a_{25} \\ a_{31} & 0 & a_{33} & a_{34} & a_{35} \\ a_{41} & 0 & 0 & a_{44} & 0 \\ a_{51} & 0 & a_{53} & a_{54} & a_{55} \end{bmatrix} x + \begin{bmatrix} 0 & b_{12} & b_{13} \\ 0 & b_{22} & b_{23} \\ 0 & b_{32} & b_{33} \\ b_{41} & 0 & 0 \\ 0 & b_{52} & b_{53} \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} x$$

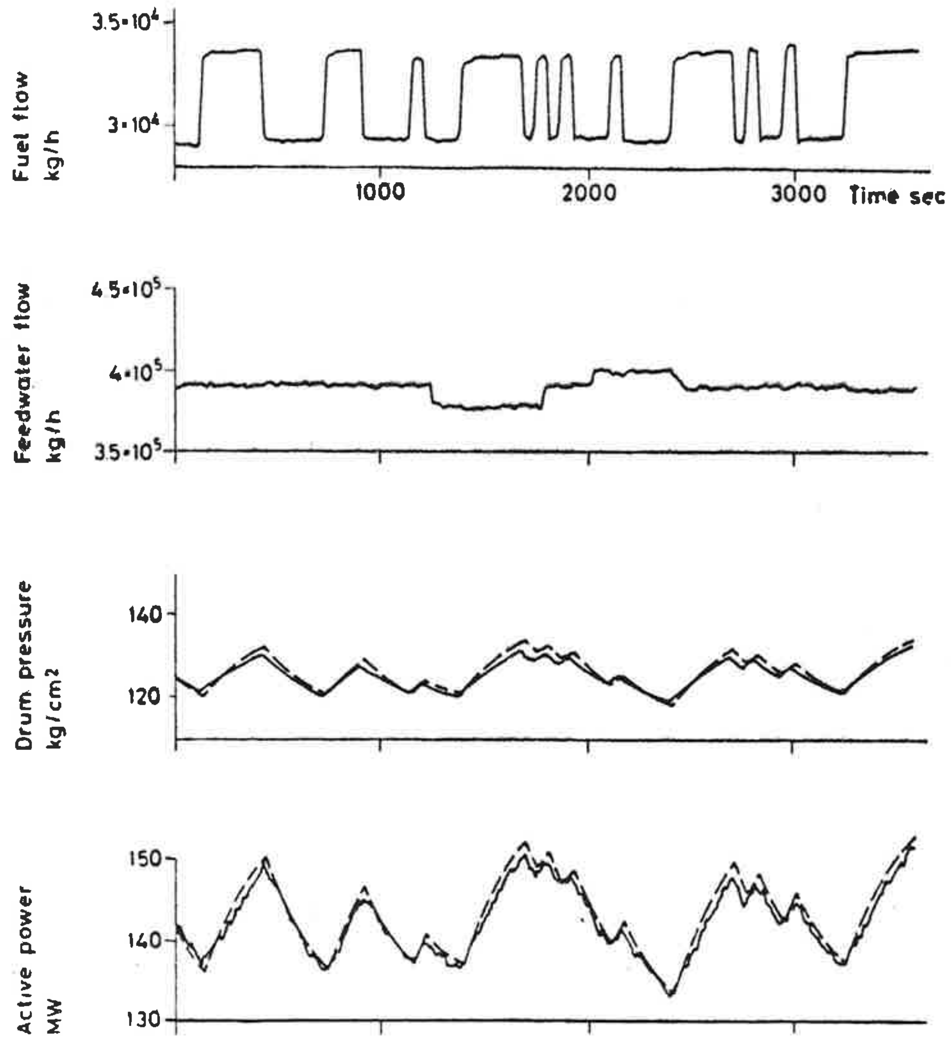
x_1 dome pressure
x_2 dome level
x_3 dome water temp
x_4 riser temp
x_5 steam water ratio

u_1 fuel flow
u_2 feed water flow
u_3 steam flow

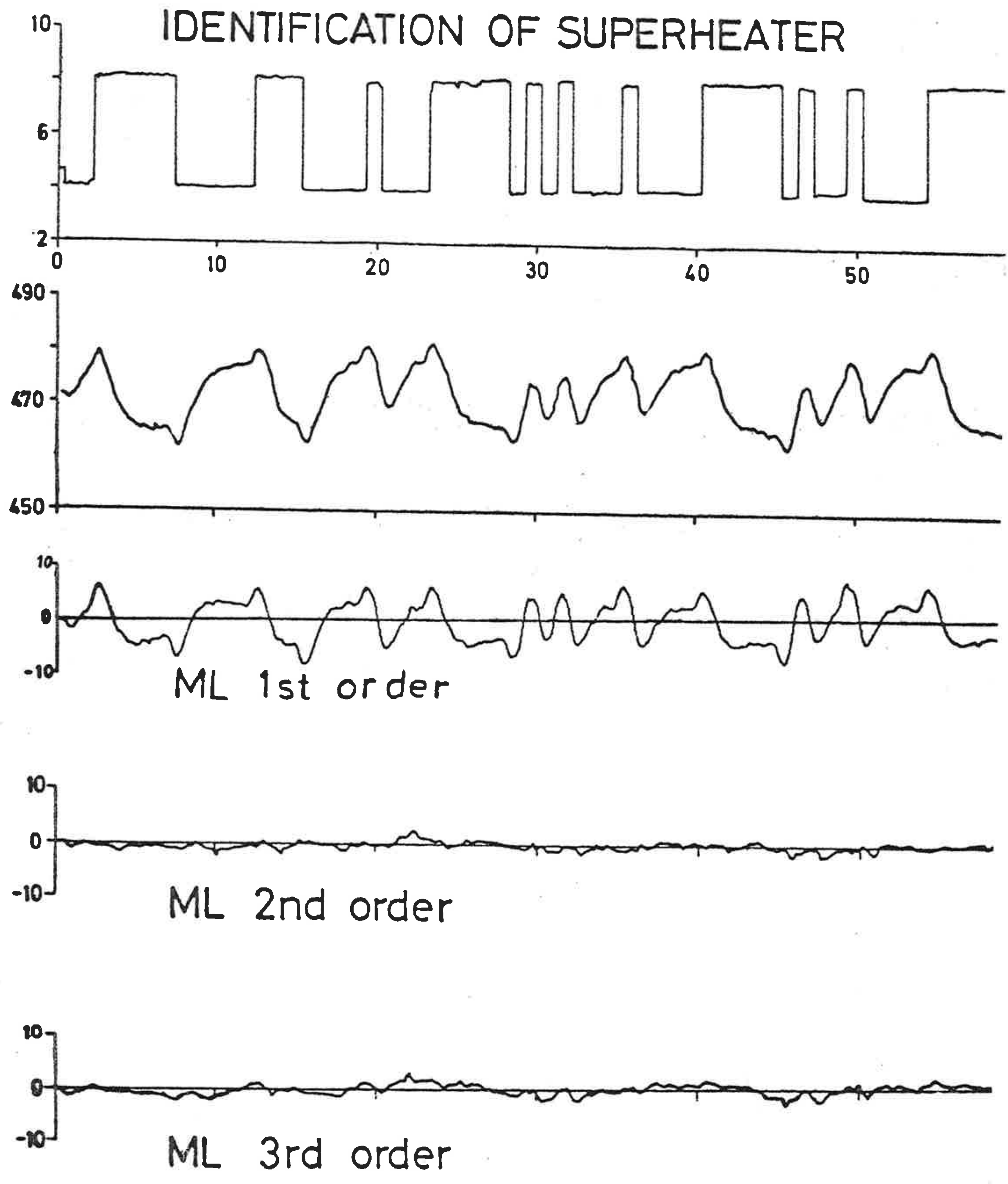

Drum level
mm
-200
0
200
Drum pressure
kg/cm²
120
140
Attemperator flow 1
kg/h
5.8·10³
6·10³
6.2·10³
Feedwater flow
kg/h
3·10⁵
4·10⁵
5·10⁵
Fuel flow
kg/h
3·10⁴
3.5·10⁴
1000
2000
3000
Time sec
Steam temperature
after reheater
°C
500
520
Steam temperature
after attemperator 2
°C
440
460
480
Steam temperature
before attemperator 1
°C
400
420
440
Steam flow
kg/h
4.6·10⁵
4.8·10⁵
Active power
MW
130
140
150
1000
2000
3000
Time sec

9:7

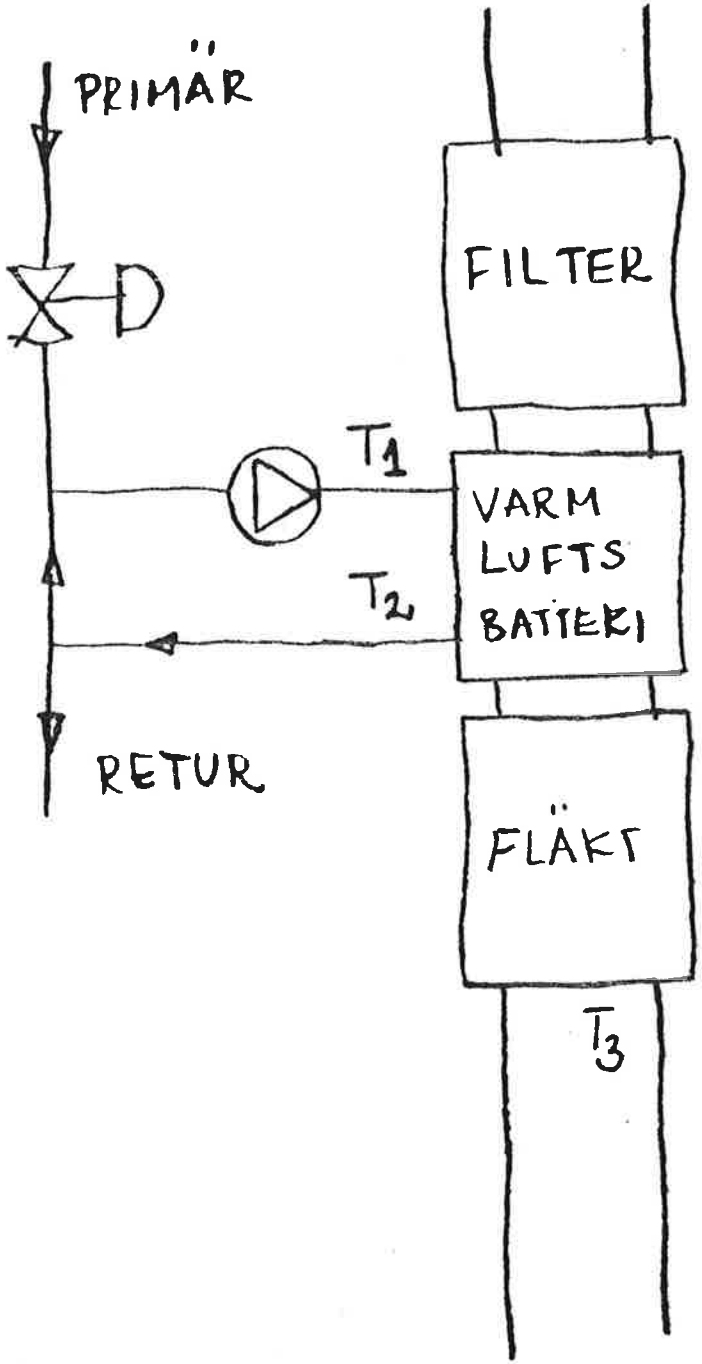
9:8



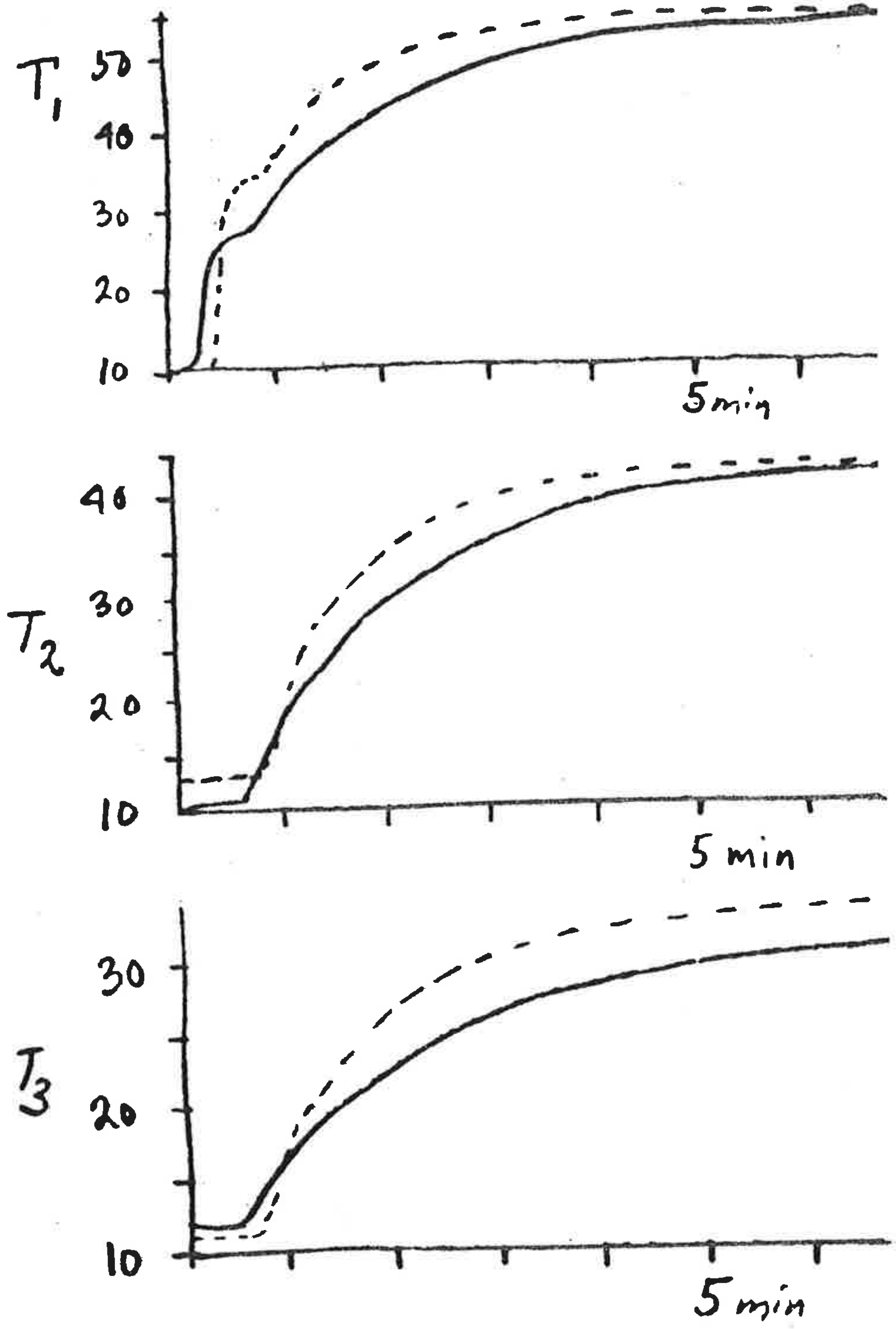
Fuel flow
kg/h
3.5·10^4
3·10^4
1000
2000
3000
Time sec
Feedwater flow
kg/h
4.5·10^5
4·10^5
3.5·10^5
Drum pressure
kg/cm2
140
120
Active power
MW
150
140
130


IDENTIFICATION OF SUPERHEATER
10
6
2
0
10
20
30
40
50
490
470
450
10
0
-10
ML 1st order
10
0
-10
ML 2nd order
10
0
-10
ML 3rd order

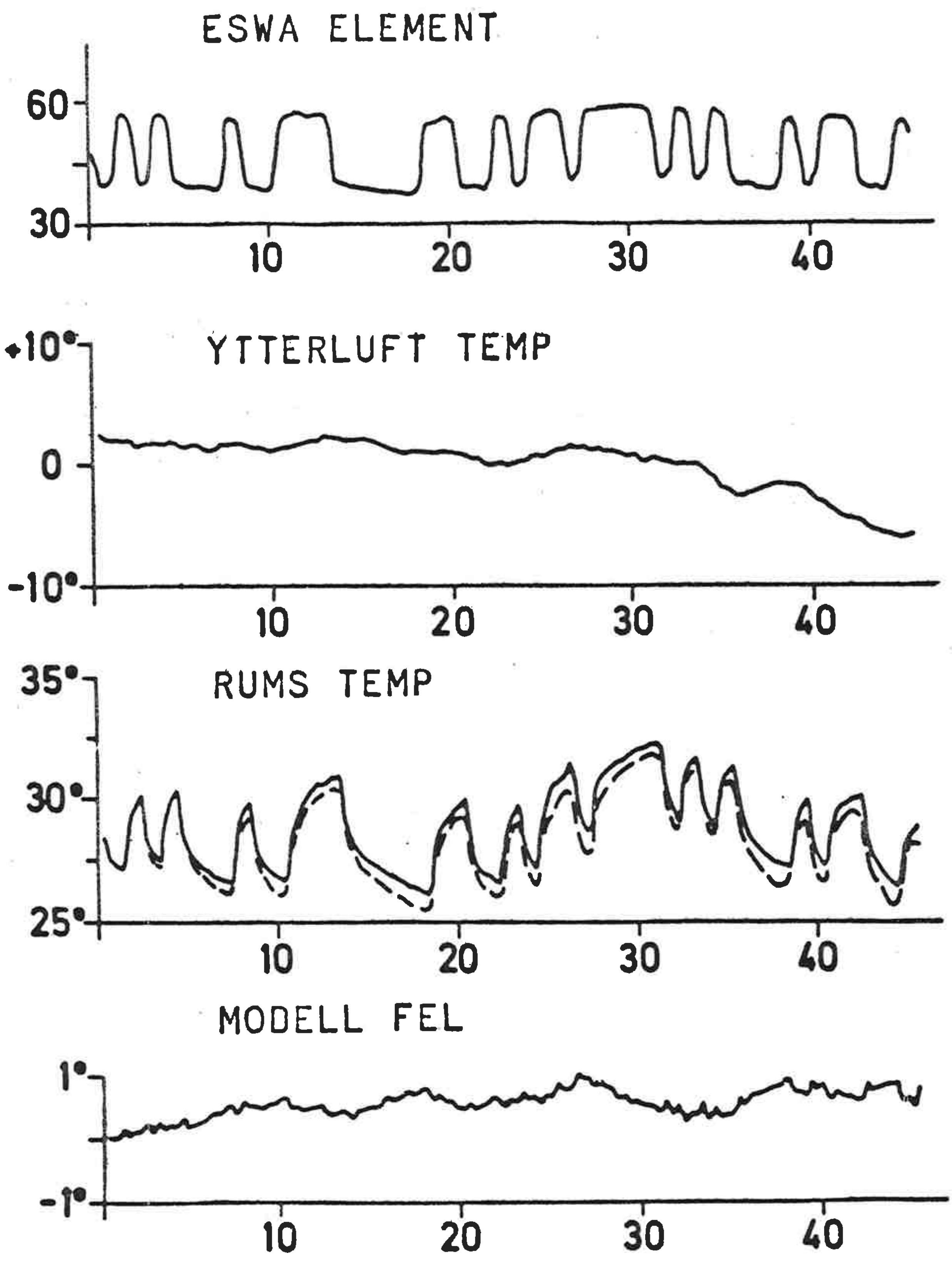
9:10



PRIMÄR
FILTER
T1
VARM
LUFTS
BATTERI
T2
RETUR
FLÄKT
T3



T1
50
40
30
20
10
5 min
T2
40
30
20
10
5 min
T3
30
20
10
5 min


ESWA ELEMENT
60
30
10
20
30
40
YTTERLUFT TEMP
+10°
0
-10°
RUMS TEMP
35°
30°
25°
MODELL FEL
1°
-1°

10. ADAPTIV REGLERING

Bo Egardt

Mera om självinställare kan läsas i

Åström K J, U Borisson, L Ljung and B Wittenmark: Theory and applications of self-tuning regulators. Automatica 13, 457-476, 1977.

ADAPTIV REGLERING

1. INLEDNING
2. ENKELT EXEMPEL
3. SJÄLVINSTÄLLARE – MINIMAL VARIANS
4. TILLÄMPNINGAR
5. ANDRA STRUKTURER

1. INLEDNING

BAKGRUND

- Dynamiken ändras
  - olika arbetspunkter
  - förslitning
  - omgivningen
  - belastning

- Alternativ till
  - manuell inställning
  - identifiering – modellbygge
    ↓
    syntes

2. ENKELT EXEMPEL

$$y(t+1) + ay(t) = bu(t) + e(t+1) + ce(t)$$

$\min Ey^2(t) \Rightarrow$

$$\boxed{u(t) = \frac{a-c}{b} y(t)}$$

Algoritm:

1. Skattning:

Skatta α i $y(t+1) + \alpha y(t) = \beta_0 u(t) + e(t)$

med MK, dvs $\min_{\hat{\alpha}} \sum \left(y(t+1) - \hat{\alpha} y(t) - \beta_0 u(t)\right)^2$

$$\boxed{\hat{\alpha}(t) = \hat{\alpha}(t-1) - \frac{y(t-1)}{\sum_{s=1}^{t-1} y^2(s)} \cdot \left(y(t) - \beta_0 u(t-1) + \hat{\alpha}(t-1) y(t-1) \right)}$$

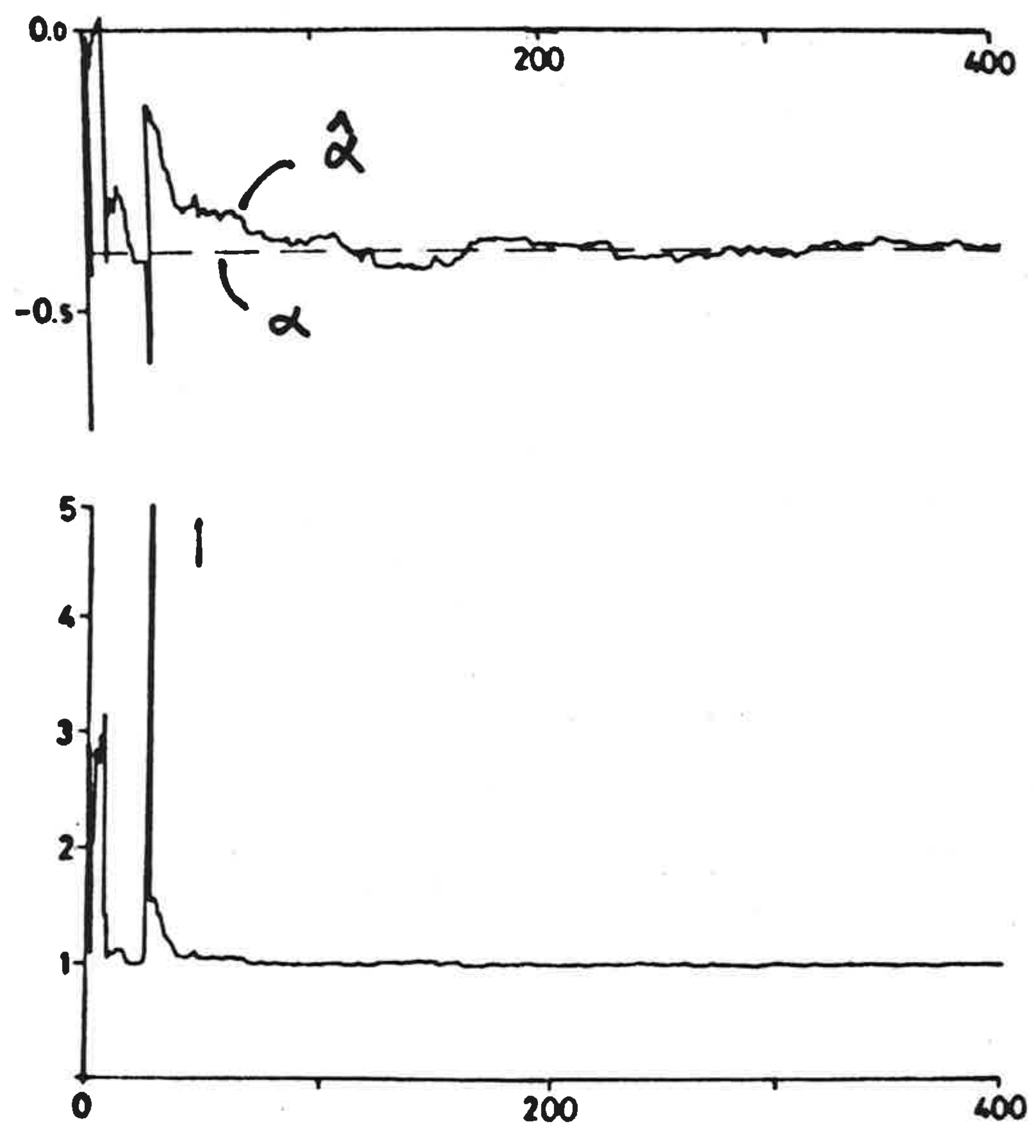
2. Styrning:

$$\boxed{u(t) = \frac{\hat{\alpha}(t)}{\beta_0} y(t)}$$

Simulering

$a = -0.5$ $b = 3$ $c = 0.7$ $B_0 = 1$

$\alpha = -0.4$



3. SJÄLVINSTÄLLARE – MINIMAL VARIANS

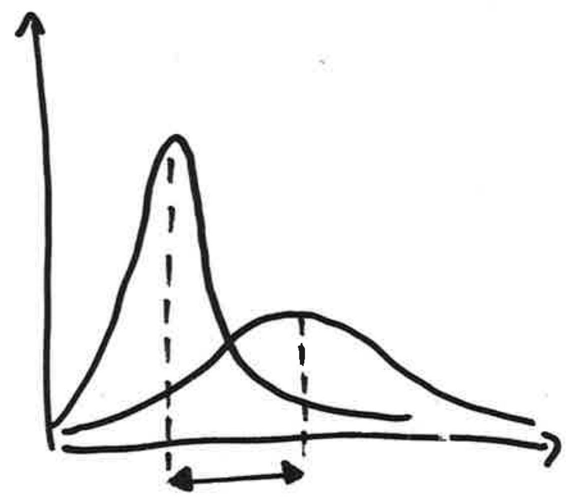
Process: $A(q^{-1})y(t) = B(q^{-1})u(t-k) + C(q^{-1})e(t)$

Minimal varians styrlag:

$$\lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} y_t^2 = Ey^2(t) \quad \text{min.}$$

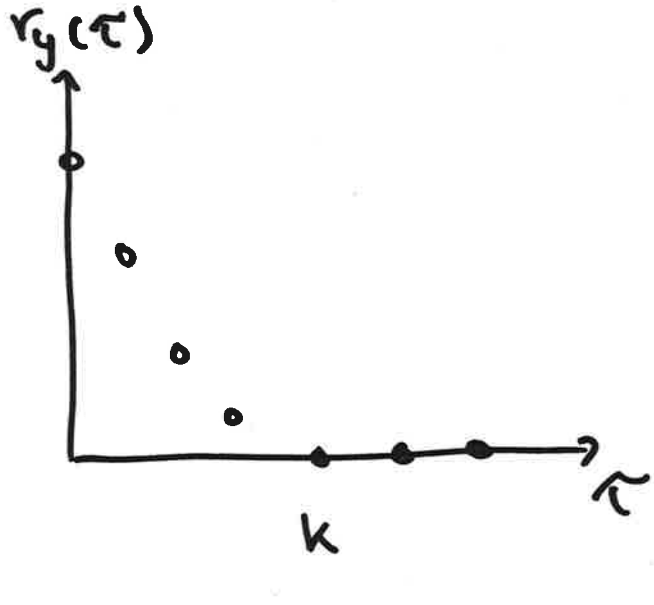
$$C = AF + q^{-k}G$$

$$u(t) = -\frac{G}{BF}y(t)$$



$$y(t) = F(q^{-1})e(t)$$

$$r_y(\tau) = 0\ ,\ \tau \geq k$$



Självinställare:

1. Skattning:

Skatta G och $\mathcal{B}F$ i

$$y(t+k) = G(q^{-1})y(t) + \mathcal{B}(q^{-1})F(q^{-1})u(t) + F(q^{-1})e(t)$$

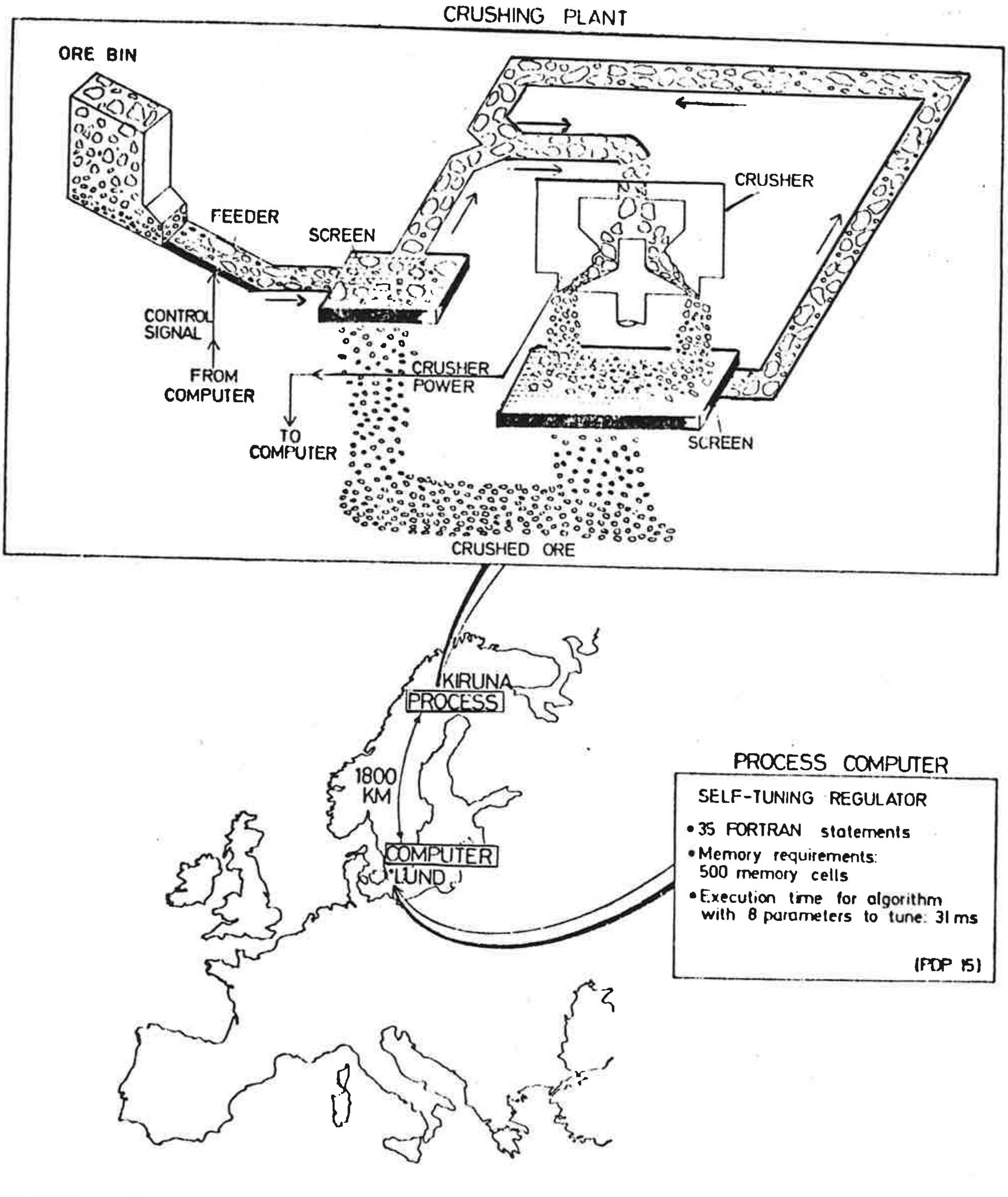
med MK

2. Styrning:

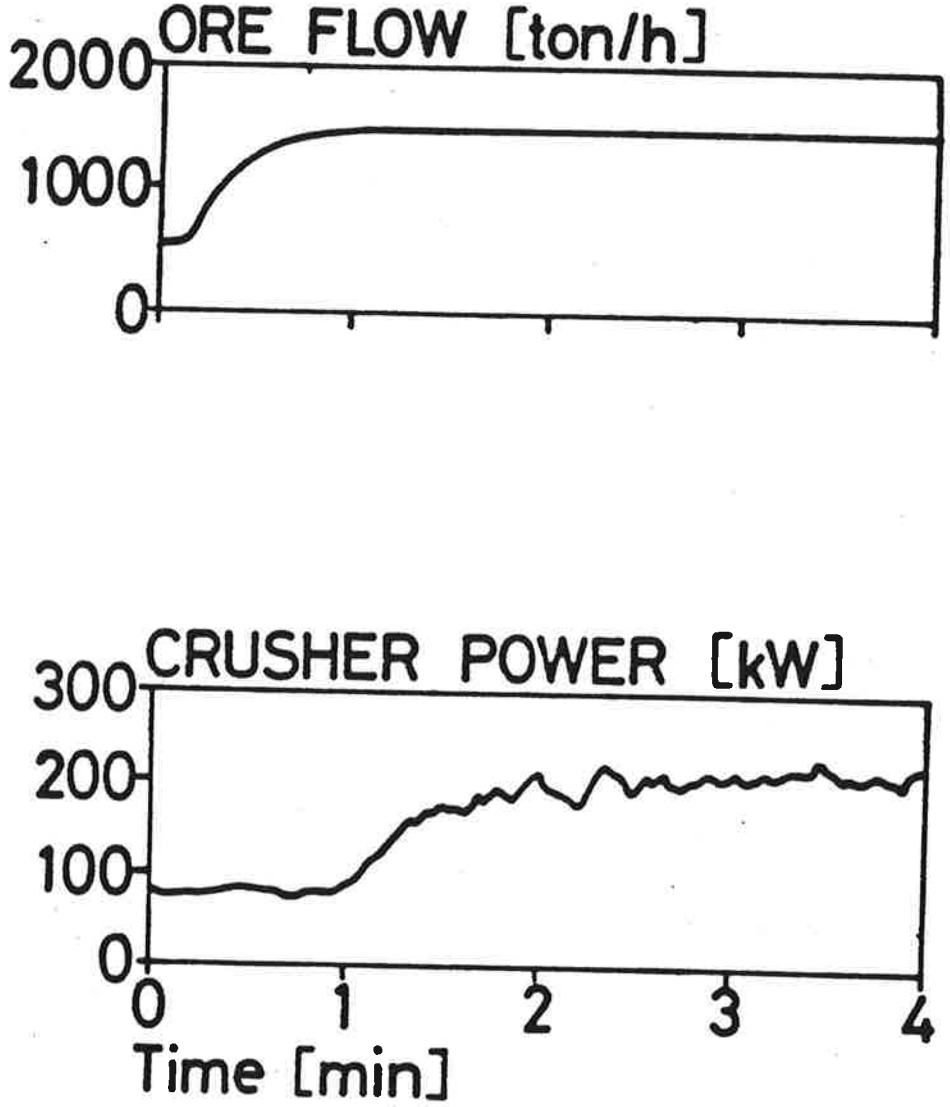
$$u(t) = -\frac{\hat{G}}{\widehat{\mathcal{B}F}}y(t)$$

4. TILLÄMPNINGAR

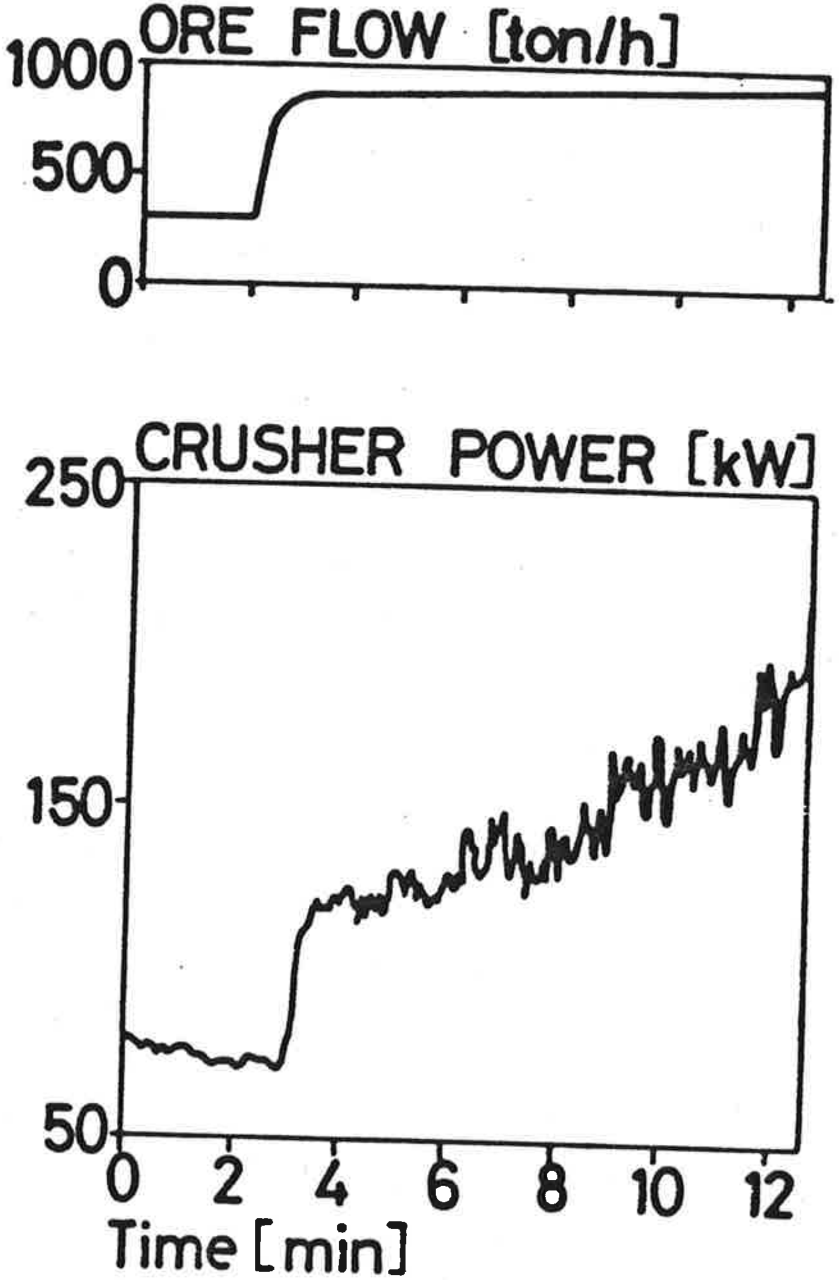
MALMKROSS



10:8

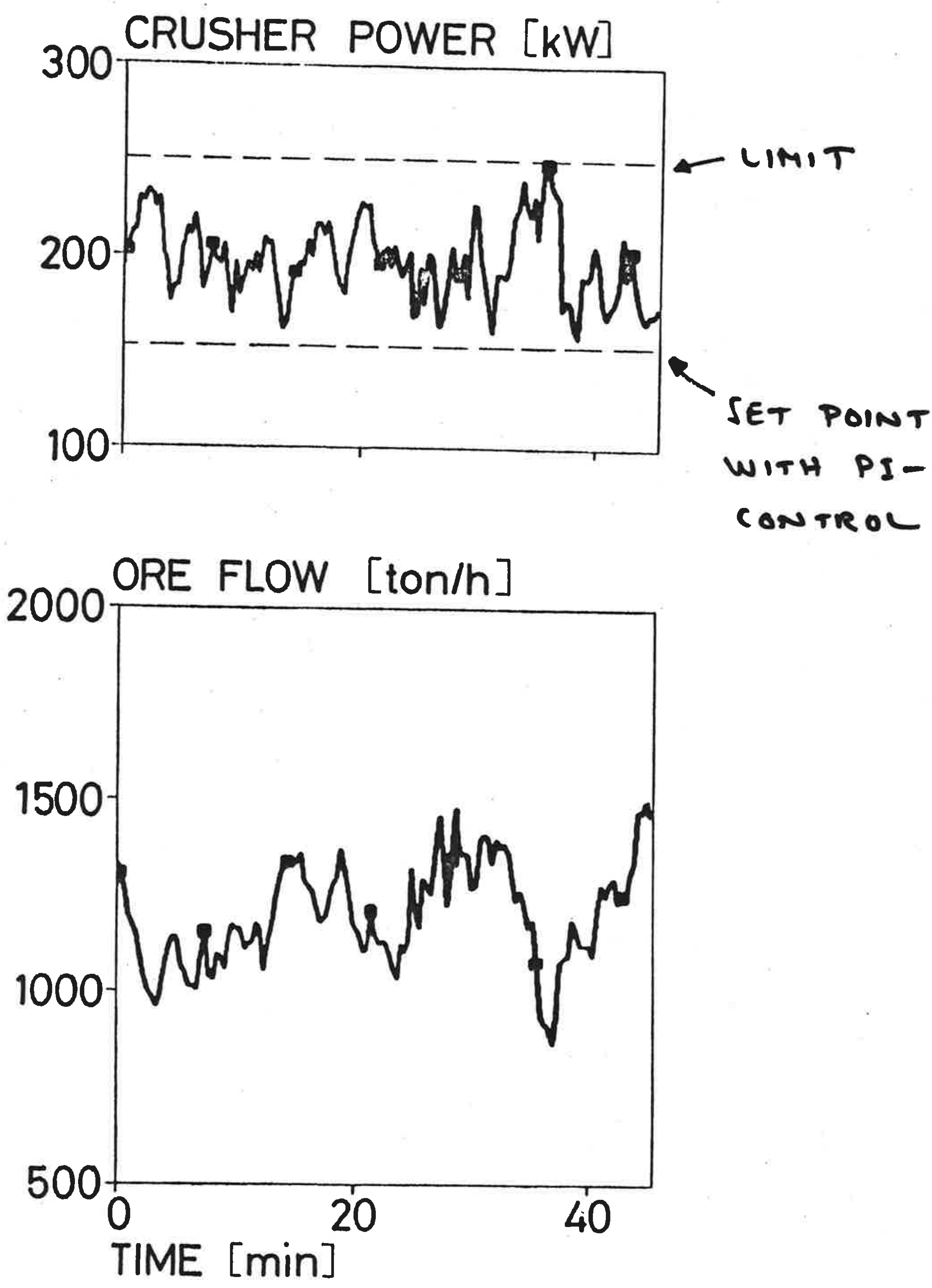


ORE FLOW [ton/h]
2000
1000
0
CRUSHER POWER [kW]
300
200
100
0
0
1
2
3
4
Time [min]



ORE FLOW [ton/h]
1000
500
0
CRUSHER POWER [kW]
250
150
50
0
2
4
6
8
10
12
Time [min]

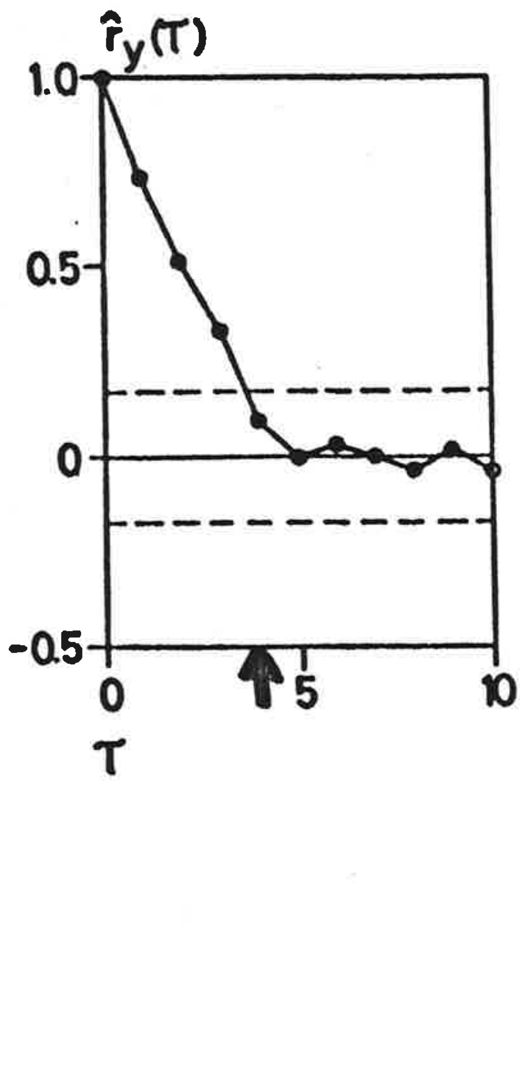
10:9

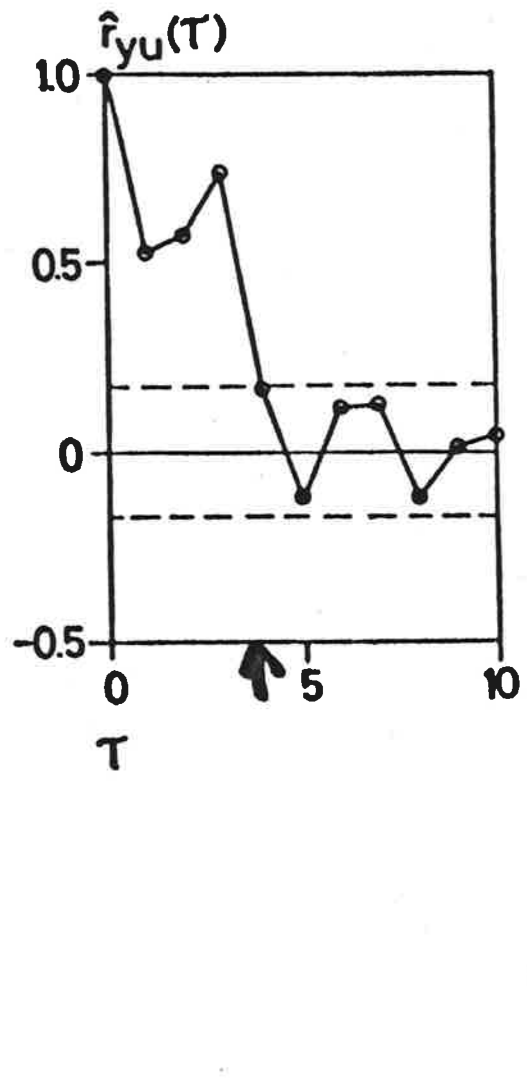

CRUSHER POWER [kW]
300
200
100
LIMIT
SET POINT WITH PI-CONTROL
ORE FLOW [ton/h]
2000
1500
1000
500
0
20
40
TIME [min]

EGENSKAP FÖR MV-REGULATOR:

$$r_y(\tau) = 0 \,, \quad \tau \geq k$$

$$r_{yu}(\tau) = 0 \,, \quad \tau \geq k$$





SUPERTANKER

Sea Swift, 255 000 tdw, full last

Modell:

$$y(t+k+1) + a_1 y(t) + \dots + a_n y(t-n+1) =$$

$$= u(t) + b_1 u(t-1) + \dots + b_m u(t-m) +$$

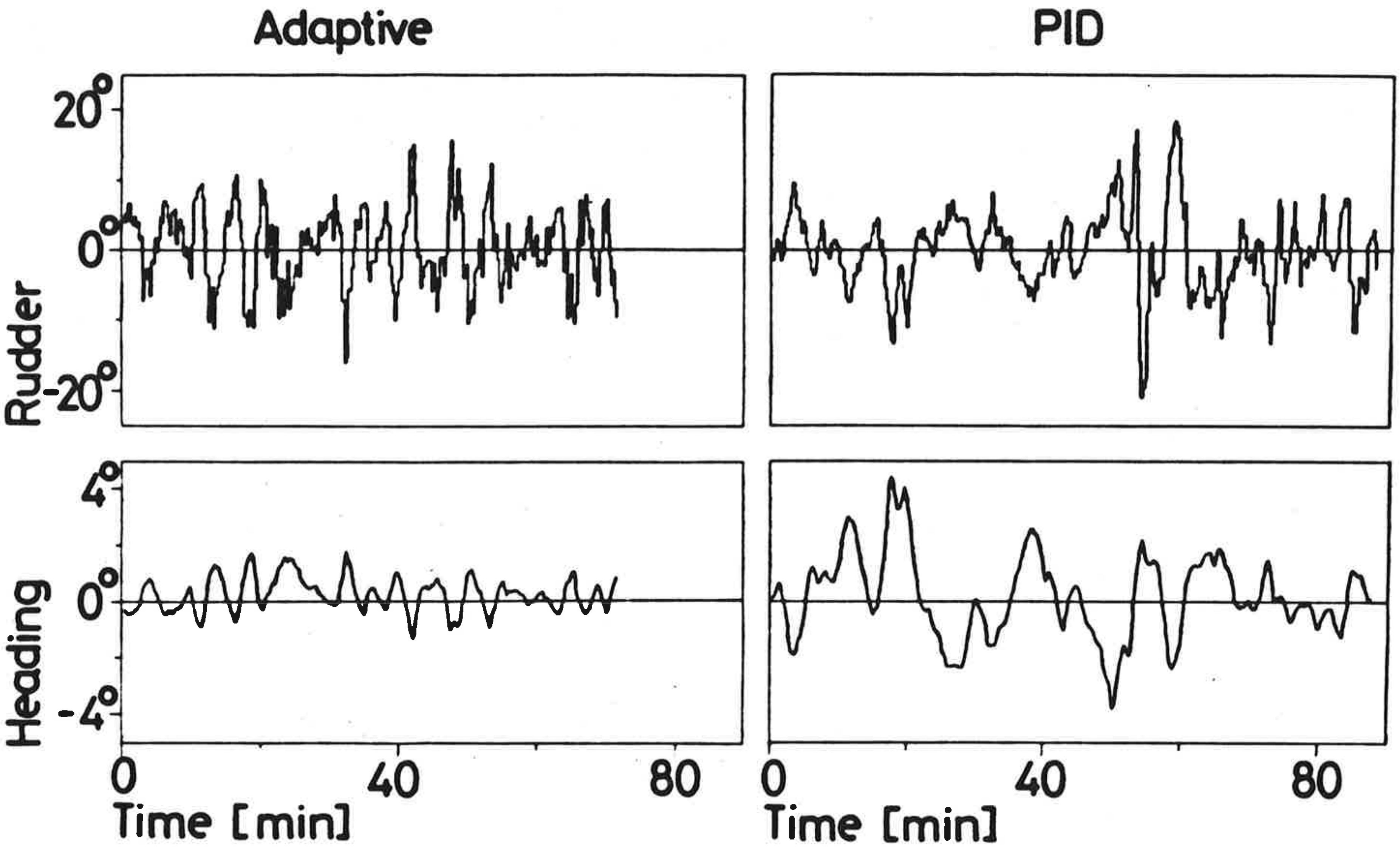
$$+ c_1 w_1(t) + c_2 w_2(t) \qquad n=4, m=2$$

y - kurs

u - roderändring

w_1 - tvärshastighet

w_2 - girvinkelhastighet

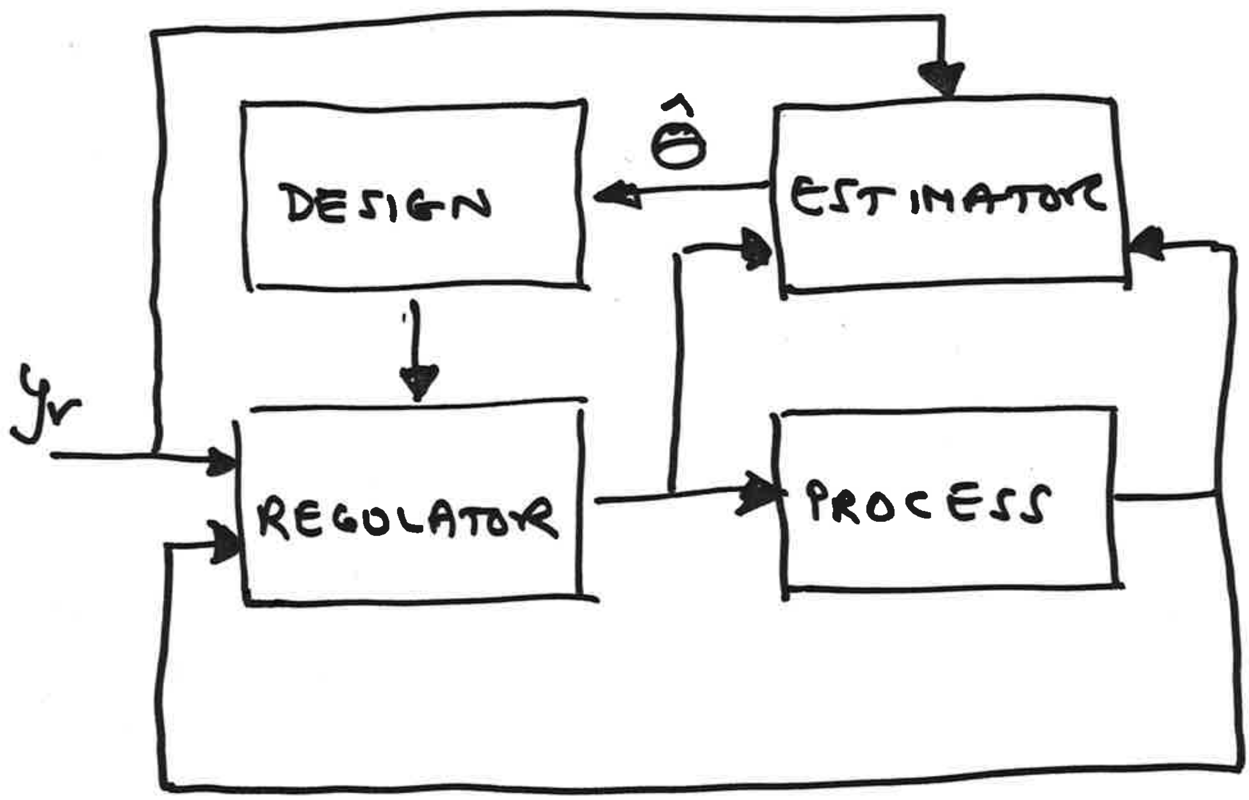

Adaptive
PID
20°
0°
-20°
Rudder
4°
0°
-4°
Heading
0
40
80
Time [min]
0
40
80
Time [min]

5. ANDRA STRUKTURER

Allmän princip:

- design metod
- parameterskattning

kombinera → självinställande reg.



10:14

DESIGN

- minimal varians
- linjärkvadratisk
- frekvenskurvor
- polplacering

etc.

SERVO
↕
REGULATOR

SKATTNING

- minsta kvadrat
- utvidgad MK
- generaliserad MK
- rekursiv ML
- utvidgat KALMAN-filter

etc.

KOMBINATION

EXPLICIT – IMPLICIT

OSÄKERHET

10:15

APPLICATIONS OF SELF-TUNING REGULATORS

Professor K J Åström

Department of Automatic Control
Lund Institute of Technology

1. INTRODUCTION

Many industrial processes can be regulated satisfactorily with PID-regulators. This is not surprising because three term controllers have been used for many years, and many processes have been designed for control by PID-regulators. Since many control loops are not critical three term controllers will undoubtedly be used extensively in the future too. With an increasing demand for efficiency in the use of energy and raw material there are, however, an increasing number of control problems where it is motivated to use regulators which are more complicated than PID-regulators. A typical example of such a control law is the MISO-regulator described by

$$u(t) = \frac{s_0 + s_1 z^{-1} + \ldots + s_n z^{-n}}{1 + r_1 z^{-1} + \ldots + r_n z^{-n}} e(t) + \frac{t_0 + t_1 z^{-1} + \ldots + t_n z^{-n}}{1 + r_1 z^{-1} + \ldots + r_n z^{-n}} v(t), \qquad (1.1)$$

where u is the control variable, e the control error and v a feedforward signal which can be a measurable disturbance or the reference input.

Although there are many rules for tuning a PID-regulator it is not always easy to adjust the parameters of such a regulator for optimal performance, particularly if the process dynamics are slow. For a regulator like (1.1) with $n = 3$ there are 11 parameters to be adjusted. This cannot be done without a systematic procedure. The lack of a suitable tuning procedure has been one of the major reasons why regulators like (1.1) have not been used extensively. One possibility to tune a regulator like (1.1) is to develop a mathematical model for the process, and its disturbances, and to derive the regulator parameters from some control design procedure. The appropriate mathematical models can be obtained from physical modeling or from system identification. The drawback with such a procedure is that it may be fairly time consuming, and that it requires personnel with skills in modeling, system identification and control design. The self-tuning regulator can be regarded as a convenient packaging of a system identification method, and a control design technique which is fairly easy to learn how to apply.

Another reason for using a self-tuning regulator is that the characteristics of the process and its disturbances may change with time. If

---

Lecture notes from Vacation School on Stochastic Processes in Control Systems held at the University of Warwick, England, April 1978.

10:16

the changes are not too rapid a properly designed self-tuning regulator may continuously tune the regulator for close to optimal performance.

The principles for design of self-tuning regulators are discussed in Section 2. The basic idea can be described as follows. Start with a design method that will give adequate results if the parameters of models for the dynamics of the process and its environment are known. When the parameters are unknown, they are replaced by estimates obtained from a recursive parameter estimator. Since the topics of these lectures are in stochastic control theory, the discussion is limited to those classes of self-tuning regulators which are based on stochastic control theory.

Applications to paper machine control are discussed in Section 3. The control problems discussed can be formulated as minimum variance control problems. There was previous experience of using minimum variance control laws based on estimation of parameters of controlled ARMA-processes. The results show that the self-tuning regulators will give good performance after a comparatively short operating time. They give a substantial reduction of the engineering effort compared to the previously used methods for control design based on plant experiment, off-line parameter estimation and control design.

Control of a special type of heat exchanger is discussed in Section 4. The control problem is difficult because there is a substantial change in process gain over the operating range. In this example there is no natural criterion. It is shown, however, that a self-tuning regulator will give a satisfactory closed loop system.

The third example deals with control of an ore crusher. This is also a case where the minimum variance criterion is natural. It is also a case when the process dynamics is poorly known, and thus a good illustration of how the structure of the regulator can be determined by experimentation.

The last example deals with design of an adaptive autopilot for a ship. This case is one of the rare situations when there is a natural quadratic criterion and the weighting on the control actions is given by physical arguments.

2. DESIGN PRINCIPLES

The principles for designing self-tuning regulators will be discussed briefly in this section. For additional material we refer to the lecture by Dr Clarke and to the references [1] through [6].

THE BASIC CONCEPT

The design procedure can be described as follows. Consider a controlled process, characterized by a parametric model of the process and its environment, and a design method. If the parameters are not known their values are substituted by estimates from a recursive parameter estimator and the control design is recalculated when new estimates are obtained. A block diagram of a self-tuning regulator is shown in Fig. 1. Since there are many different recursive parameter estimation procedures and many different methods for control design, there are consequently a large number of different self-tuning regulators. To follow the theme of this lectures the discussion will be limited to self-tuning regulators based on stochastic control theory [8]. In the terminology of stochastic control theory the self-tuning regulator can be classified as a *certainty equivalence* control law [9].

The naive description of the self-tuning regulator is useful as a first evaluation of the suitability of a particular design in the following way. If the design method does not work well when the parameters are known the corresponding self-tuning regulator will most likely perform badly.

SELF-TUNERS BASED ON MINIMUM VARIANCE CONTROL

Some simple self-tuning regulators based on stochastic control theory will now be described. It is thus assumed that the process to be controlled is described by the controlled ARMA process

$$Ay_t = Bu_{t-k} + C\xi_t. \tag{2.1}$$

A particularly simple case is obtained if $C = 1$ because the parameter estimation can then be made by least squares. Other cases are discussed in [4].

The self-tuning regulators can be based on identification of an *explicit* or an *implicite* process model. These distinctions are illustrated by two examples.

EXAMPLE 2.1. (Algorithm with explicite identification)
Consider a process described by (2.1) with k=1 and $C = 1$. Assume that the criterion is to minimize

$$E\, y^2(t). \tag{2.2}$$

Introduce

$$\begin{aligned} \varphi_t &= [-y_{t-1} \dots -y_{t-n} \quad u_{t-1} \dots u_{t-n}]^T \\ \theta_t &= [\hat{a}_1 \dots \hat{a}_n \quad \hat{b}_1 \dots \hat{b}_n]^T \\ \varepsilon_t &= y_t - \varphi_t^T \theta_{t-1}. \end{aligned} \tag{2.3}$$

The estimate is then given by the recursive least squares equations

$$\begin{aligned} \theta_t &= \theta_{t-1} + P_t \varphi_t \varepsilon_t \\ P_t &= \frac{1}{\lambda} \{P_{t-1} - P_{t-1} \varphi_t [\sigma^2 + \varphi_t^T P_{t-1} \varphi_t]^{-1} \varphi_t^T P_{t-1}\}. \end{aligned} \tag{2.4}$$

If the problem is badly conditioned numerically the updating of P_t can be replaced by a square root algorithm [10], [11]. If computational time

10:18

is crucial the updating can be replaced by a fast algorithm [12], [13]. Having obtained the estimate the control law is determined by solving the polynomial equation

$$1 = AF + z^{-k} G \tag{2.5}$$

for F and G. The control law is then given by

$$u = -\frac{G}{BF} y.$$

Because the polynomial B is cancelled when the feedback (2.6) is applied to the process (2.1), it is clear that the control law will work well only if the polynomial B is stable. □

The self-tuning algorithm in Example 2.1 is called an algorithm with *explicite* or *direct identification* of a process model because the parameters of the process model (2.1) are updated explicitely in the algorithm. The corresponding algorithm with *implicite* or *indirect identification* are described in Example 2.2.

EXAMPLE 2.2. (Algorithm with implicite identification)
Consider a process described by (2.1) with $C = 1$ and assume as in Example 2.1 that the criterion is to minimize (2.2). To obtain the algorithm the identity (2.5) is first used to rewrite the process model as follows:

$$y_{t+k} = (AF + z^{-k} G)\, y_{t+k} = Gy_t + BFu_t + FR_t. \tag{2.7}$$

This model is called a *prediction model* because it can be used directly to predict the output k steps ahead. The model (2.7) can be written as

$$y_{t+k} + \alpha_1 y_t + \ldots + \alpha_m y_{t-m+1} = \beta_0 [u_t + \beta_1 u_{t-1} + \ldots + \beta_\ell u_{t-\ell}] + \varepsilon_t. \tag{2.8}$$

The parameters of the prediction model (2.8) are now estimated by least squares. Introduce

$$\varphi_t = [-y_{t-k} \ldots -y_{t-m-k+1} \quad \beta_0 u_{t-k-1} \ldots \beta_0 u_{t-\ell-k}]^T$$

$$\theta_t = [\alpha_1 \ldots \alpha_m \quad \beta_1 \ldots \beta_\ell]^T \tag{2.9}$$

$$\varepsilon_t = y_t - \varphi_t^T \theta_{t-1},$$

then the least squares estimate is given by (2.4). Having obtained $\hat{\theta}_t$ then the control law is given by

$$u_t = -\frac{1}{\beta_0} \varphi_t^T \theta_t.$$

□

The algorithm in Example 2.2 is called an algorithm with *implicite* or *indirect* identification of a process model because the parameters of the prediction model (2.7) are estimated rather than the parameters of the process model (2.1). Notice that the coefficients of the process model are identical to the coefficients of the dead-beat control law associated with (2.1). Notice also that the implicite algorithm is simpler because it is not necessary to solve the identitiy (2.5) in each step. The solution of (2.5) is handled indirectly through the choice of the particular model structure (2.7).

It is easy to include tuning of feedforward in the simple self-tuners. This is achieved simply by replacing the model (2.8) by

$$y_{t+k} + \alpha_1 y_t + \ldots + \alpha_m y_{t-m+1} =$$
$$= \beta_0 [u_t + \beta_1 u_{t-1} + \ldots + \beta_\ell u_{t-\ell}] + \gamma_0 v_t + \ldots + \gamma_r v_{t-r} + \varepsilon_t , \quad (2.11)$$

where v_t is the feedforward signal which may be the reference input or a measurable disturbance. The control law is then given by

$$u_t = \frac{1}{\beta_0} [-\alpha_1 y_t - \ldots - \alpha_m y_{t-m+1} - \gamma_0 v_t - \ldots - \gamma_r v_{t-r}] - \beta_1 u_{t-1} - \ldots - \beta_\ell u_{t-\ell} . \quad (2.12)$$

In this way it is e.g. possible to tune the parameters of the regulator (1.1). It is of course also possible to use several feedforward signals.

SELF-TUNING REGULATORS BASED ON LINEAR QUADRATIC CONTROL THEORY

The self-tuning regulators based on minimum variance control suffers from the same drawbacks as the minimum variance control strategies. The process zeros are cancelled in the closed loop. This is of little importance if the zeros are well inside the unit disc. The cancellation is, however, disastrous if the system is not minimum-phase. It has, however, turned out in practice that nonminimum-phase systems can be dealt with at a loss of efficiency by choosing the parameter k sufficiently large. Another drawback of the minimum variance control is that there is no penalty on the control. This may sometimes lead to excessive control signals. This problem is to some extent overcome by the self-tuning controller [7]. This controller may, however, be unstable for unstable plants. The natural way to introduce a penalty on control is to design a self-tuning regulator based on the linear process model (2.1) and the criterion

$$\lim_{N\to\infty} E \frac{1}{N} \sum_{t=1}^{N} [y^2(t) + \rho u^2(t)] . \quad (2.13)$$

This type of self-tuning regulator is described in [6]. A self-tuning regulator based on this criterion will require more calculations because it is necessary to solve a Riccati equation or equivalently to do a spectral factorization in each step. With the increasing computing power of microprocessors such calculations are, however, feasible in many cases.

There are also many other structures for self-tuning regulators. See [4]. A discussion of convergence properties are given in [7].

3. CONTROL OF BASIS WEIGHT AND MOISTURE CONTENT ON PAPER MACHINES

A schematic drawing of a paper machine is shown in Fig. 2. Basis weight and moisture content are important variables which characterizes the quality of the finished product. Control of these quality variables can be conveniently described by stochastic control theory. Disturbances arise from many different sources. Their net effect on the output can be

10:20

modeled accurately as stochastic processes. The purpose of the control is to reduce the fluctuations in the quality variables. This goal is well described mathematically as to minimize the variances of the process outputs. See Fig. 3. There is no natural way to introduce any penalty on the control variables. The thick stock flow valve is the control variable for the basis wieght loop and the steam pressure is the control variable for the moisture loop. The process dynamics of both loops are characterized by time delays and low order dynamics. There is coupling in the sense that changes in the thick stock flow valve will influence both basis weight and moisture content. Changes in the steam flow to the drying section influences the moisture content only. There are also interactions in the measuring devices because the beta ray gauge measures both basis weight and moisture content. These couplings can, however, easily be eliminated. Simple processing of the measured signals gives dry basis weight and moisture content. The basis weight is then controlled by a feedback from the dry basis weight signal to the thick stock flow gate and the moisture content is controlled by a feedback from the moisture content signal to the steam flow to the drying section. A feedforward signal, e.g. from the thick stock flow measurement to the steam flow to the dryers, is introduced to compensate for the interaction in the process. A more detailed description of the control problem is given in [14] and [15].

It has been verified by experiments on many plants that the dynamics of the plant and the disturbances can be adequately described by controlled ARMA processes of low order. See [14]. By identification of process dynamics and disturbance characteristics based on plant experiments it was demonstrated that substantial improvements over PID control could be achieved by using a basis weight regulator having the form

$$u(t) = -\frac{s_0 + s_1 z^{-1}}{1 + r_1 z^{-1} + r_2 z^{-2} + r_3 z^{-3}} \, y(t) . \qquad (3.1)$$

The parameters of a regulator like (3.1) can not be conveniently tuned by hand. Instead the parameters were determined by system identification based on plant experiments and control design as described in [14]. Such a procedure is comparatively costly because it requires appropriate identification software and skilled personnel. To obtain a reasonably good model it is necessary to experiment on the plant for at least 2 hours. It was therefore attempted to tune the parameters with a self-tuning regulator. Since the criterion was well defined and since the sampling period and the regulator structure could be chosen based on prior experience it was very straightforward to apply the self-tuning regulator. There were several process control computers available at the plant. The code for the simple self-tuning regulator based on least squares identification and minimum variance control was simply introduced as a special control algorithm, and experiments were started. A typical result is illustrated by the simulation shown in Fig. 4. This simulation is based on measured plant disturbances and models for the process dynamics estimated from plant experiments. In Fig. 4 the self-tuning regulator was initialized with all estimates equal to zero. It is seen from Fig. 4 that the fluctuations in basis weight obtained from the self-tuning regulator are larger than those obtained from the minimum variance regulator for the first 30 minutes. After 30 minutes there are, however, very small differences between the outputs of the two regulators. It is perhaps even more instructive to look at the control signals generated by the different regulators. It is seen from Fig. 4 that the self-tuner is fairly sluggish in the initial period. After 30 minutes there are, however, only minor differences in the control signal. Results similar to those shown in Fig. 4 were obtained when controlling the actual plants.

On typical basis weight and moisture control loops the self-tuner will give close to optimal performance after a tuning period of 15 minutes to 2 hours. In the example shown in Fig. 4 no apriori information about the parameters was assumed. In many applications in the paper mill it is possible to start the algorithm with reasonable estimates. The convergence time will then be shorter. The tuning time should be compared with the time required to make a good identification experiment. This time is between 2 and 5 hours. In the paper machine applications there have not been any problems with phenomena like "turn off" of "covariance blow up" which have been reported in literature. One reason is that the disturbances are persistent and fairly stationary.

In the paper mill we have been working with the self-tuning regulator used as a tuning device and not as an adaptive regulator, since the chief instrument engineer does not like the parameters of important control loops to be changed in his absence. The simple self-tuning regulator has been applied to many simple flow and level loops, to basis weight and moisture loops [16] on several different paper machines, and to recovery boilers.

It is much easier to implement a simple self-tuning regulator than to go through the procedure of process experiments, system identification, and control design. It is thus clear that the self-tuning regulator gives a considerable saving of engineering work compared to previously used methods.

4. CONTROL OF A HEAT EXCHANGER

This application is described in [17]. A schematic drawing of the heat exchanger is shown in Fig. 5. This type of heat exchanger is a common component in heating and ventilation systems. The warm air leaving a room gives away some of its enthalpy to the fresh air supplied to the room. The warm and cold air streams pass through the rotor which has axial channels. The rotor is made of a material which can absorb heat and moisture. The enthalpy is exchanged because the rotor segments are alternating between the warm and cold streams when the rotor rotates. The control problem is to adjust the rotor rate in such a way that the room temperature is constant. The disturbances are due to sunshine, heat generated from people, and other heat sources. In the particular case there are also sensor noise because the sensors are thermistors in the air streams. It is desirable to keep temperature fluctuations small but in contrast with the paper machine example there is no natural loss function. For a fixed operating condition it is easy to obtain good control simply by using integrating feedback from room temperature to rotor rate. The major difficulty is that the gain of the process changes drastically with operating conditions. The difficulties were avoided by

10:22

choosing a particular regulator structure combined with a self-tuning regulator. This structure was derived based on detailed knowledge of the physics of the process.

The primary controlled variable is chosen as the thermal efficiency defined by

$$v = \frac{T_{cout} - T_{cin}}{T_{win} - T_{wout}}, \qquad (4.1)$$

where the subscript c stands for cold and w for warm air. The thermal efficiency v is thus first computed based on measurements of T_{cin}, T_{win}, T_{wout}, and knowledge of the desired T_{cout}. The advantage of choosing this variable as the controlled variable rather than the cold outlet temperature, is that the effect on the control loop of some of the process nonlinearities are eliminated at the prize of three extra thermistors. The thermal efficiency is then controlled by a feedback from the computed thermal efficiency to commanded rotor rate.

The dynamics relating thermal efficiency to rotor rate is characterized by a time delay and low order dynamics. The major difficulty in controlling the heat exchanger is that the static gain varies considerably. In steady state the relation between the rotor rate u and thermal efficiency v is approximatively given by

$$v = f(u) = \frac{au}{1 + au}. \qquad (4.2)$$

If the units are chosen in such a way that $0 \leq u \leq 1$ then a may be 20 in a typical case. The static gain thus varies between 20 and 1/20.

The heat exchanger can be controlled by an integrating controller. Due to the gain variations there are, however, difficulties when a regulator with constant gain is used. This is illustrated by the simulation in Fig. 6. It is clearly seen from this figure that the loop gain is too high at low levels and too low at high levels of efficiency. If the relation (4.2) was accurate and did not change with time then equation (4.2) could be used to make the gain of the controller a function of the thermal efficiency. This is unfortunately not possible because the parameter a changes with time. A simple self-tuning regulator was therefore used to eliminate the gain variations. The self-tuning regulator is based on estimation of the parameter b in the model

$$M: v(t+1) - v(t) = b[u(t) - u(t-1)] \qquad (4.3)$$

by least squares. In this case the estimate is particularly simple since only one parameter is estimated. The estimate is given by the following equations:

$$\hat{b}(t+1) = \hat{b}(t) + P(t+1)\ \nabla u(t)\ \varepsilon(t+1)$$

$$\varepsilon(t+1) = \nabla v(t) - \hat{b}(t)\ \nabla u(t)$$

$$P(t+1) = \frac{P(t)\ \sigma^2}{\lambda[\sigma^2 + P(t)\ \nabla u(t)]}.$$

The variable v denotes the thermal efficiency and u denotes the rate of rotation of the rotor. Having obtained the estimate $\hat{b}$ of b the following control law is then used:

$$u(t) = u(t-1) + (k_0/\hat{b})\, v(t), \tag{4.4}$$

where k_0 is an empirical constant. The "cautious" control law

$$u(t) = u(t-1) + \frac{k_0\hat{b}}{\hat{b}^2 + P}\, v(t) \tag{4.5}$$

was also attempted but there was little difference in performance compared to (4.4).

The performance of the self-tuning regulator is illustrated by the simulation results shown in Fig. 7. The models and disturbances were the same as when generating the results shown in Fig. 6. The Figures 6 and 7 are thus directly comparable. It is clear that the self-tuning regulator handles the gain variations very well. The behaviour of the self-tuning regulator on the actual plant is similar to that shown in the simulations. See [17].

5. CONTROL OF AN ORE CRUSHING PLANT

This example is described in detail in [18]. A schematic drawing of the process is shown in Fig. 8. The plant consists of an ore bin, a feeder, two screens, an ore crusher, and conveyor belts. The ore is transported from the bin to a screen, where the small ore lumps are separated from the large lumps. The larger ore lumps are transported to the crusher where the lumps are crushed. After the crusher there is another screen which separates lumps with a diameter of about 2.5 cm. Larger lumps are recirculated to the crusher. The crusher is driven by an electric motor via a slip clutch which releases the motor and stops the line if the torque is too high. The control variable is the amount of ore fed into the line and the controlled variable is the power of the crusher motor. The goal of the control is to keep production as high as possible while avoiding overloading. This can be formulated approximately as to reduce the variance in the crusher power. By the usual argument, illustrated by Fig. 3, the set point of the crusher power can be moved closer to the target and the average production is increased as a consequence. The trade-off between high production and risk for overload is reflected in the choice of set point. The disturbances are due to variations in lump size, crushability, and variations of the crusher characteristics due to wear. The plant dynamics is characterized by a time delay of 40 - 50 s between feeder and crusher, a time delay of 70 - 80 s in the recycle loop and time constants of 10 - 20 s in the crusher itself.

When the experiments were started there was very little apriori knowledge of the characteristics of the process and its environment. Some step-responses were therefore determined initially. An interesting aspect of the experiments was that they were performed using teleprocessing between a plant in Kiruna in northern Sweden and a computer in Lund in southern Sweden. The distance between the two places is about 1800 km.

It was decided to try the simple self-tuning regulator based on minimum variance control. Based on the time delays of the process it was decided

10:24

that a sampling period of 20 s was reasonable. Since the time delay of the process was 40 - 50 s the values 2 and 3 of the parameter k in the self-tuner are then reasonable. It was found experimentally that $k = 3$ worked better than $k = 2$. The complexity of the regulator was determined experimentally by controlling the plant with regulators having different complexity. The sample covariance function and the cross covariance between the output and the control variable were determined. The complexity of the regulator was increased until the conditions

$$r_y(\tau) = 0, \quad \tau \geq k+1$$
$$r_{yu}(\tau) = 0, \quad \tau \geq k+1,$$

which hold for the minimum variance controller, were fulfilled. See [8]. It was found that a simple self-tuning regulator with $m = 4$ and $\ell = 3$ was performing well. The forgetting factor λ was also determined empirically. The value $\lambda = 0.99$ was chosen after some experimentation. The value $\lambda = 0.95$ was found to be slightly better during start-up and during periods with a high variability in ore properties. The results of one experiment are illustrated in Fig. 9, Fig. 10, and Fig. 11.

6. AN AUTOPILOT FOR SHIP STEERING

This application is described in detail in [19] were many additional references are given. An ordinary autopilot for a ship is based on feedback from measurements of heading (and possibly also the rate of change of the heading to the rudder angle). A PID-algorithm is commonly used. An autopilot has different functions. It should be able to maintain the ship at a constant course and it should be able to handle manoeuvers. The dynamics of a ship will change with changes in speed, trim, loading, and water depth. The characteristics of the disturbances will also change considerable with changes in weather and wind. Although, it is in many cases possible to find constant settings of an ordinary autopilot which will guarantee stability over a wide range of operating conditions, there is a considerably advantage in having an adaptive autopilot. It is a common experience on tankers that ordinary autopilots do not work well in bad weather. The reason is partly that the PID-algorithm is too simple to handle the requirements and partly that proper tuning to different weather conditions is required.

The design of an autopilot for straight course keeping can be formulated as a stochastic control problem. The ship dynamics can be described as a linear dynamical system and the disturbances can be characterized as random processes. Fortunately there is also a natural loss function which fits well into the stochastic control formulation. It can be shown by hydrodynamic theory that the average increase in drag due to yawing and rudder motion can be approximately described by

$$\frac{\Delta R}{R} = \mu(\overline{\psi^2} + \rho\overline{\delta^2}),$$

where R is the drag, ψ the heading deviation, δ the rudder angle, and $\overline{\psi^2}$ denotes the quadratic mean value. The values $\mu = 0.014$ deg^{-2} and $\rho = 0.1$ are typical for a tanker. It is thus natural to use the criterion

$$V = \frac{1}{T} \int_0^T \{ [\psi(t) - \psi_{ref}(t)]^2 + \rho\delta^2(t) \} \, dt$$

as a basis for the design and evaluation of autopilots for steady state course keeping. One unit of the loss function would then correspond to an increase of 1.4 % of the average drag or about the same increase in fuel consumption.

Since the design of an autopilot can be formulated as a linear quadratic stochastic control problem an adaptive autopilot can be designed using the corresponding self-tuning regulator. Several such designs have been made, simulated, and field tested. The experiments have been carried out on several large tankers. One of the autopilots has been in operation for well over two years. Available on board computers were used in the experiments. Due to memory constraints several simplifications had to be made. Since the LQG self-tuners require the solution of Riccati equations, which is space consuming, it was also attempted to use the simple self-tuner based on minimum variance control. This was reasonably successful provided that the prediction horizon was chosen appropriately. Extensive comparisons between different regulator structures including well-tuned PID-regulators were made. One comparison is illustrated in Fig. 12. It was found that the adaptive autopilot reduces the drag by 0 - 2 % in comparison with an ordinary autopilot based on PID control. In these comparisons the ordinary autopilot was at all times retuned for best performance. The size of the improvements depends on the operating conditions. The largest improvements are found in bad weather when the ship is fully loaded.

7. CONCLUSIONS

A number of applications of self-tuning regulators have been described. The examples have ranged from situations where very little was known about the characteristics of process dynamics and disturbances to cases where very much was known. It has been demonstrated that good control laws can indeed be obtained. Although the examples chosen for this lecture have been such that the underlying design method is based on stochastic control theory, it should be remembered that there are many other types of self-tuning regulators.

The word self-tuning or adaptive control may lead to the false conclusion that such regulators can be switched on and used blindly without any apriori considerations. This is definitely not true. The self-tuning regulator is a fairly complex control law. A proper design involves the choice of gross features like regulator structure, estimation procedure, and control design, and details like sampling rate, forgetting factors, initialization, and limitation of control output. A proper choice requires insight and knowledge; a bad choice may be disastrous. The application of self-tuning regulators is thus by no means automatic. It is my hope that this lecture may inspire some of you to acquire the appropriate knowledge and try some schemes of your own.

10:26

ACKNOWLEDGEMENTS

Over the years I have had many stimulating discussions on self-tuning regulators and their applications with B Wittenmark, I Gustavsson, U Borisson, L Jensen, L Ljung, C Källström, R Syding, J Sternby, and J Holst. The benefit derived from this is gratefully acknowledged. I would also like to thank the Swedish Board of Technical Development which has supported my research in this field for many years.

8. REFERENCES

[1] V Peterka: "Adaptive digital regulation of noisy systems". Paper 6.2, Preprints 2nd IFAC Symposium on Identification and Process Parameter Estimation, Prague 1970.

[2] K J Åström and B Wittenmark: "On self-tuning regulators". Automatica 9 (1973) 185-199.

[3] B Wittenmark: "Self-tuning regulators". Report TFRT-1003, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1973.

[4] K J Åström, U Borisson, L Ljung, and B Wittenmark: "Theory and applications of self-tuning regulators. Automatica 13 (1977) 457--476.

[5] K J Åström and B Wittenmark: "Analysis of a self-tuning regulator for non-minimum phase systems". Preprints IFAC Symposium on Stochastic Control, Budapest 1974.

[6] D W Clarke and P J Gawthrope: "Self-tuning controller". Proc IEE 122 (1975) 929-934.

[7] L Ljung: "Analysis of recursive stochastic algorithms". Proc IEEE AC-22 (1977) 551-575.

[8] K J Åström: "Introduction to stochastic control theory". Academic Press, New York 1970.

[9] K J Åström: "Stochastic control problems". To appear in A Coppel (ed) "Lectures on Control Theory", Springer Lecture Notes.

[10] C L Lawson and R J Hanson: "Solving least squares problems". Prentice Hall, Englewood Cliffs, N J 1974.

[11] V Peterka: "A square-root filter for real-time multivariable regression". Kybernetica 11 (1975) 53-67.

[12] N Levinson: "The Wiener RMS (root mean square) error in filter design and prediction". J Math Phys 28 (1947) 261-278.

[13] M Morf, L Ljung, and T Kailath: "Fast algorithms for recursive identification". Proc IEEE Conf on Decision and Control, Clean Water Beach, Florida 1976.

[14] K J Åström: "Computer control of a paper machine - an application of linear stochastic control theory". IBM J Res & Dev 11 (1967) 389-405.

[15] K J Åström: "Applications to paper making in curriculum and methods in control and systems engineering, vol II". IBM Corporation 1970.

[16] U Borisson and B Wittenmark: "An industrial application of a self--tuning regulator". Preprints IFAC Symposium on Digital Computer Applications to Process Control, Zürich 1974.

[17] L Jensen and R Hänsel: "Computer control of an enthalpy exchanger". Report TFRT-3081, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1974.

[18] U Borisson and R Syding: "Self-tuning control of an ore crusher". Automatica 12 (1976) 1-7.

[19] C G Källström, K J Åström, N E Thorell, J Eriksson, and L Sten: "Adaptive autopilots for steering of large tankers". To be presented at IFAC World Congress, Helsinki, June 1978.

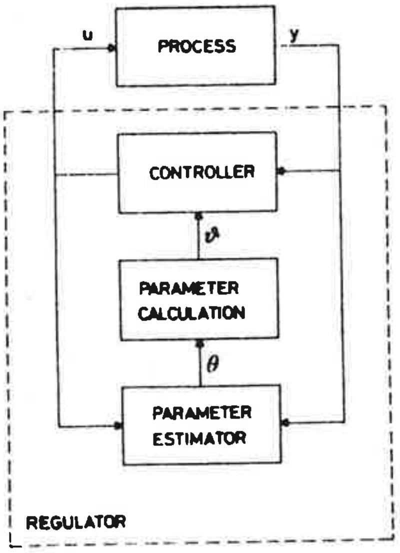


Figure 1. Schematic diagram of a self-tuning regulator.

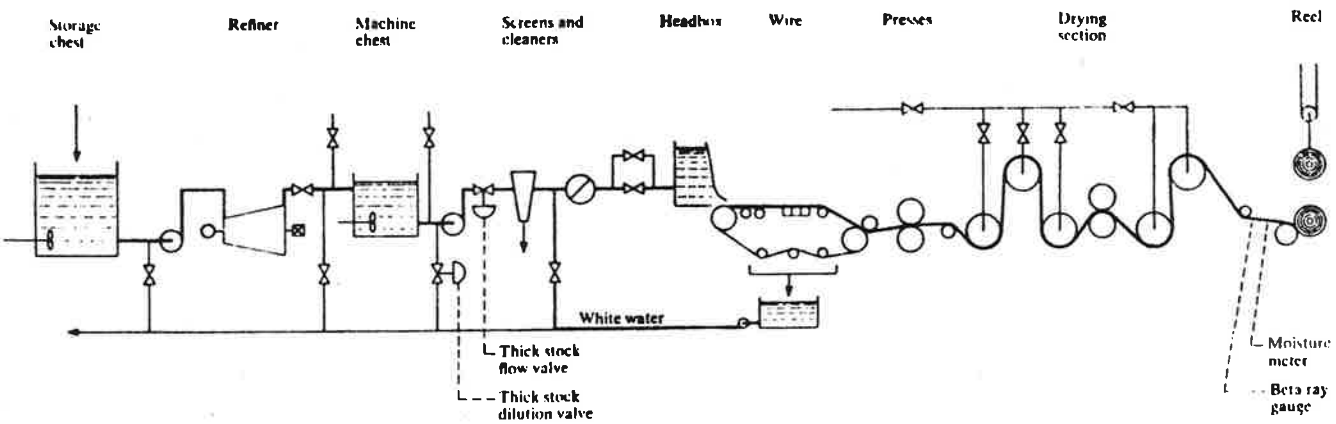


Figure 2. Schematic drawing of a paper machine. From [14].

10:28

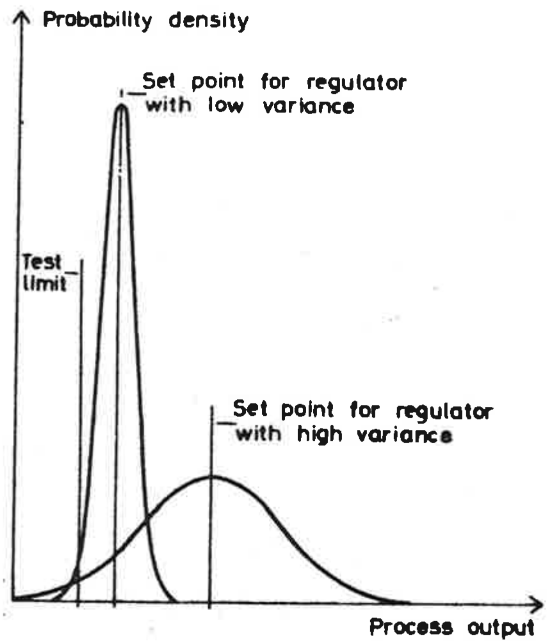


Figure 3. Benefits of minimum variance control. By reducing the variance of the output signal, the mean value can be moved closer to the target. From [8].

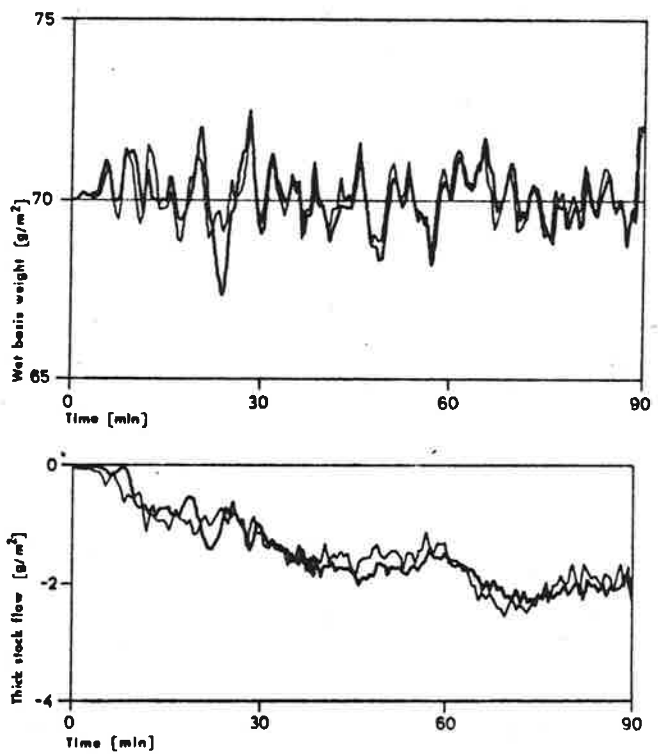


Figure 4. Simulation of minimum variance control (thin lines) and self-tuning control (thick lines) of basis weight of a paper machine. From [2].

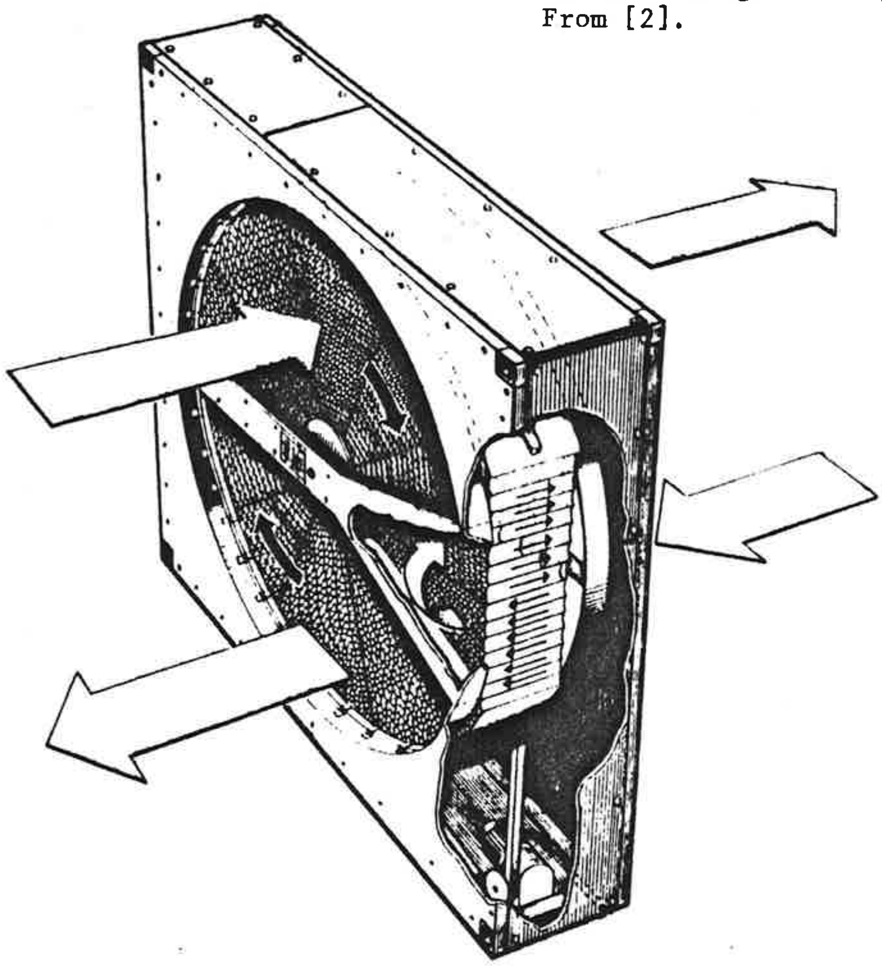


Figure 5. Schematic drawing of an air-to-air heat exchanger.

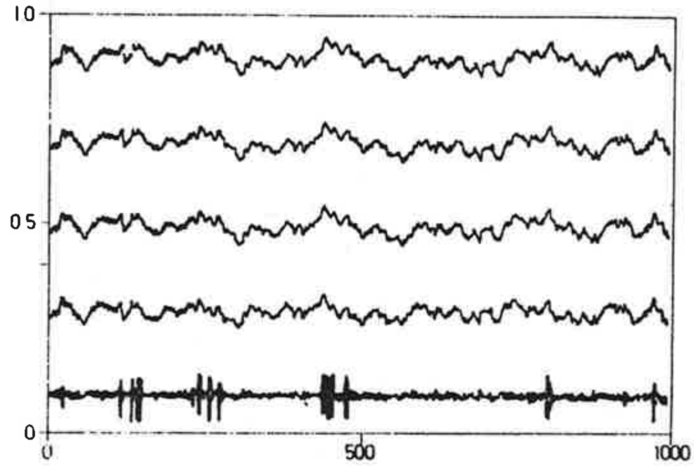


Figure 6. Thermal efficiency of a heat exchanger controlled by an integrating controller having fixed gain.

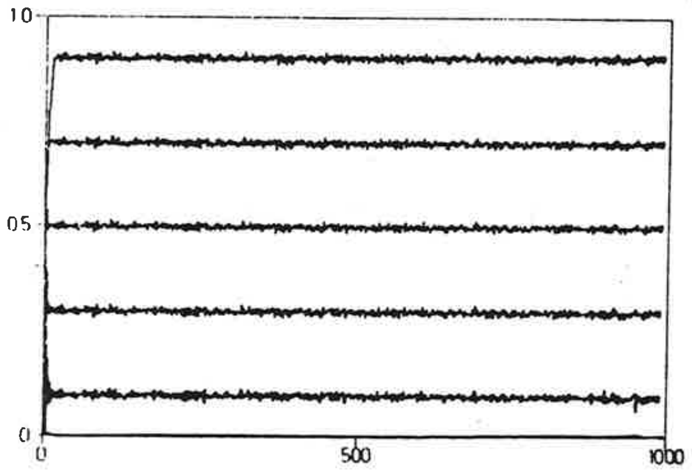


Figure 7. Thermal efficiency of the heat exchanger with a self-tuning integrating regulator.

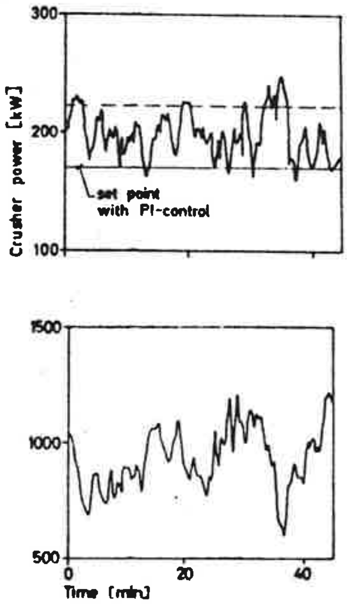


Figure 9. Control variable (ore flow) and controlled variable (crusher power) from experiment on an ore crusher. From [18].

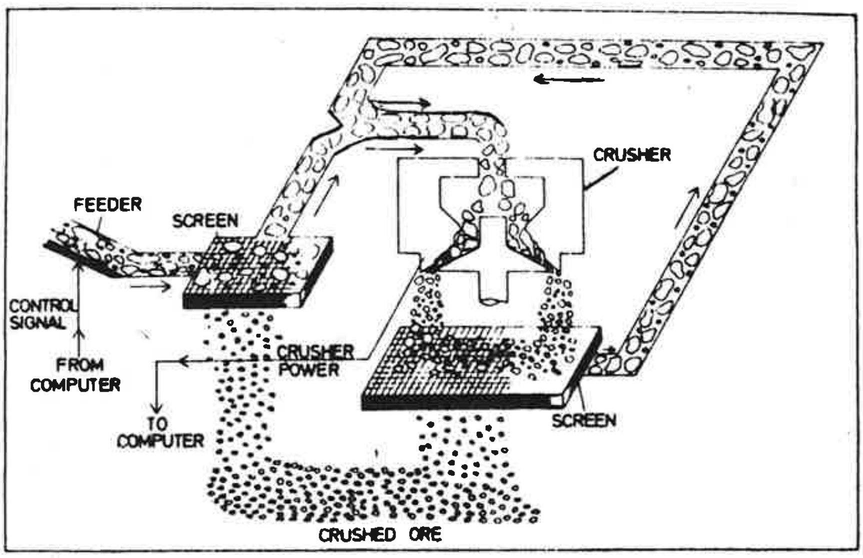


Figure 8. Schematic drawing of an ore crushing plant. From [18].

/0:30

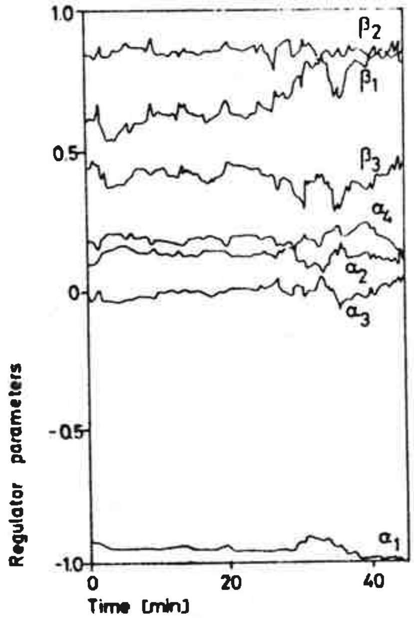


Figure 10. Parameters of the self-tuning regulator obtained in an ore crusher experiment. From [18].

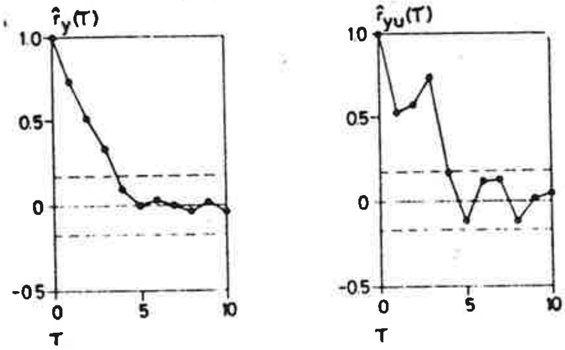


Figure 11. Covariance functions r_y and r_{yu} from an ore crusher experiment. From [18].

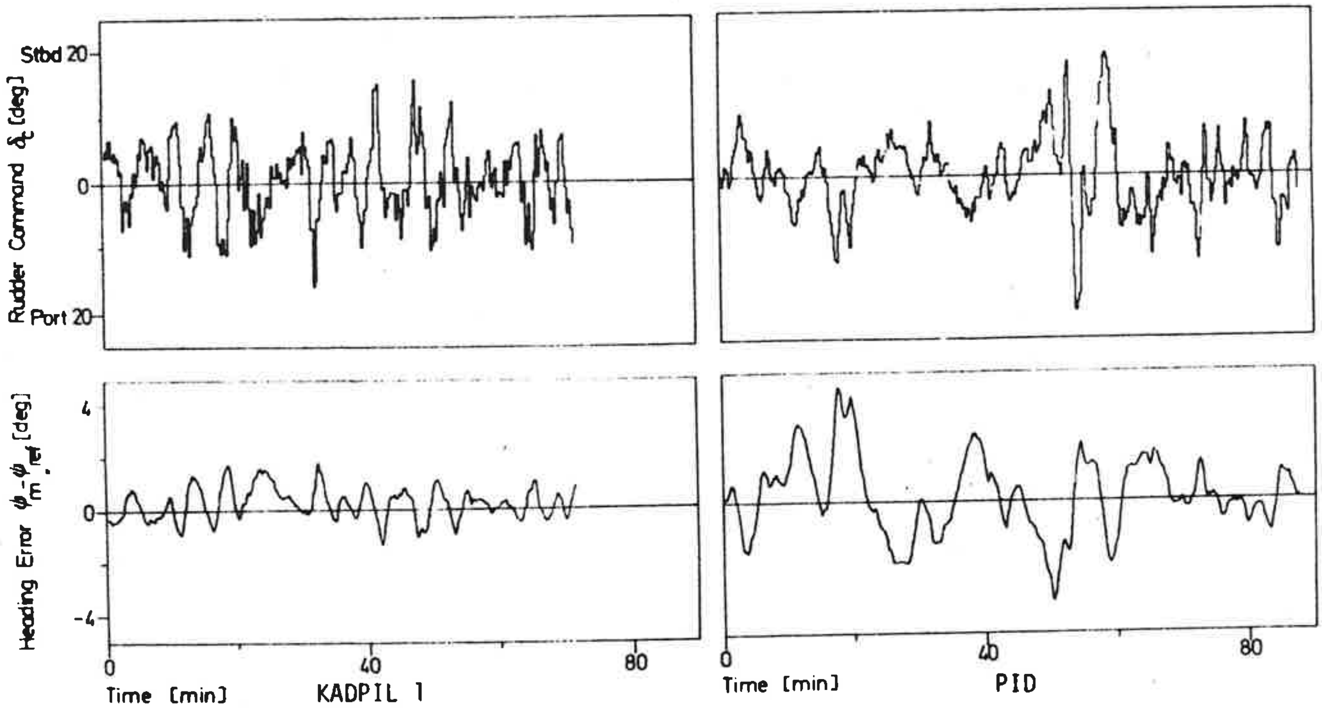


Figure 12. Comparison of a well-tuned PID-regulator and a self-tuning regulator in ship steering experiments. From [19].