SRC TR 88-67

TECHNICAL RESEARCH REPORT

On-Line Optimization of Chemical Plants Using Steady State Models

by

J. Koninckx

SYSTEMS RESEARCH CENTER

UNIVERSITY OF MARYLAND

COLLEGE PARK, MARYLAND 20742

ABSTRACT

Title of Dissertation :

ON-LINE OPTIMIZATION OF CHEMICAL PLANTS USING STEADY STATE MODELS

Jan Koninckx, Doctor of Philosophy, 1988
Dissertation directed by : T.J.Mc Avoy
Professor
Chemical & Nuclear Engineering Dept.

The subject of this dissertation is the on-line optimization of continuous chemical plants that are operated under steady state. For these plants, transient periods are short compared to periods of steady operation, and a steady state optimization (using a steady state nonlinear model) covers the major part of the potential gain that can be made through optimization. Cheaper computer technology and a more competitive market cause an increased industrial interest in this supervisory control technique.

The existing literature on applications of on-line optimization using steady state models is discussed. Publications reflect the wide interest in the optimization based technique, but in general the reports are vague, and they do not answer many fundamental questions. Some are even contradictory on issues such as partitioning of the optimization problem or choice of optimization variables.

In this dissertation, a modular structure for an on-line optimizer is suggested. In this structure, existing algorithms in model updating, data reconciliation and optimization are combined with new applications. The application of sensitivity analysis is the most important new approach that is presented in

this dissertation.

Optimization sensitivity analysis is a computationally cheap tool that provides information about the status of an optimization result. That information can be used in an on-line optimizer for use in a significance test of setpoint changes, and for an on-line accuracy assessment of the on-line optimizer operation. Sensitivity information is therefore combined with statistical information from e.g. the model updating module. Results from a sensitivity analysis can also be used in short-cut feasibility studies.

Also model execution frequency, data reconciliation techniques and particular problems with model updating are discussed, as well as the influence of noise on the performance of on-line optimized plants.

Case study results are provided as illustration. The systems studied in these case studies are a distillation column (propane propylene splitter), a boiler network with common header and a simple heat exchanger network.

ON-LINE OPTIMIZATION OF CHEMICAL PLANTS USING STEADY STATE MODELS

by

Jan Koninckx

Dissertation Submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1988

Advisory Committee :
Professor T.J. Mc Avoy
Professor O.A. Asbjornsen
Professor T.E. Marlin
Associate Professor A.L. Tits
Assistant Professor K.P. Halemane

ACKNOWLEDGMENTS

I gratefully acknowledge my advisor, Dr. Thomas J. McAvoy for his guidance, s ιpport and encouragement during the entire length of my stay at the University of Maryland. His kindness, hospitality and optimism have made my studies towards my Ph.D.degree an enjoyable time. I thank Dr. Thomas E. Marlin for his numerous and invaluable suggestions, as well as for the many interesting hours of discussion we spent together. I am also grateful to Dr. André L. Tits, for his splendid instruction in optimization, and for his accurate criticism.

I am greatly indebted to the Systems Research Center and the Graduate School of the University of Maryland, for supporting me financially during my Ph.D. studies.

I thank the faculty and the staff at Maryland for their services. I also thank my fellow students for their friendship and company. Leo Hsie, Philippe Rinaudo and especially Pete Minderman have been particularly fine office mates. I am also very pleased to thank Ed Peters, and certainly Coreen Lee for being dependable roommates and good friends. Some luck lies in not getting what you thought you wanted but getting what you have, which once you have it you may be smart enough to see is what you would have wanted had you known. (Garrison Keillor,[Lake Wobegon Days])

Last, but not least, I wish to express my gratitude towards my parents. Their continuous support and encouragement since my first little scribble in elementary school, have made this possible.

Jan Koninckx

September 1988, College Park, Maryland

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

A	reaction component, heat exchange area or flow area
B	bottoms flow, reaction component, index set of active inequality constraints, matrix of (intradepartemental flows) reconc. equations
b	model parameter estimate (confidence interval calc.)
C	reaction component
C_p	heat capacity
C^n	set of n times continuously differentiable functions
c	concentration, cost (if not a function of other variables), operation cost function
D	distillate flow, matrix of interdepartemental flows reconc. equations, drum or tube diameter
d	fouling factor
E^n	n-dimensional Euclidean space, calculated enthalpy
F	flow rate (unless specified : mass flow rate), quantile value of the F distribution
f	optimization objective, friction coefficient
G	Choleski triangle of setpoint covariance matrix, mass velocity
g	inequality constraint (possibly vector), gravity
H	enthalpy (on a mass basis)
h	equality constraint (possibly vector), a general vector mapping (in observability theory), heat transfer coefficient

I	index set
i	index
j	index, Chilton-Colburn factor
K	Kalman filter constant
k	reaction kinetic coefficient, heat transfer coefficient, index
L	liquid down flow, Lagrangian function, tube length
l	dimensionality of the measurement space
M	mass (e.g.: tube mass), tangent subspace
N	number of trays
$\mathbb{N}$	the set of natural numbers
n	a given integer, i.e. also number of measurements, number of states
P	general optimization problem, pressure
p	a given integer (also number of parameters) probability ($p(a\|b)$=probability of a condition to b)
Q	(measurement) variance covariance matrix, exchanged heat
q	shift forward operator
q^{-1}	shift backward operator
$\mathbb{R}$	the set of real numbers
R	reflux ratio
R^2	multiple correlation coefficient
Re	Reynolds number
S	subset of $\mathbb{R}^n$
SQP	succesive quadratic programming
s	estimate standard deviation
s^2	estimate variance
T	temperature
t	time, quantile value of the t distribution
U	heavy key differential cost,

	heat transfer coefficient
u	lagrange multiplier for unequality constraints, measurement values (per plant console)
V	reactor, drum or tube volume, vapor boilup
	a set of stochastic processes
v	specific volume, measurement values (interconnecting plant consoles), also measurement noise
W	measurement error variance covariance matrix, weighting matrix, light key differential cost, Wilcoxon test statistic
w	lagrange multiplier for equality constraints
X	independent variables, steam quality
x	fraction,
	independent optimization variables ($\epsilon\mathbb{R}$), system states,
Y	dependent variables
y	independent optimization variables ($\epsilon\mathbb{N}$), dependent optimization variables, model responses
z	measurement, gaussian random number

Greek Symbols

α	relative volatility, test confidence limit
β	compensated test confidence limit, true parameter value (confidence inerval calc.), quantile of the χ^2 distribution
ϵ	element of
ε	small (usually positive) number
δ	change, difference
ζ	measurement adjustments (interdepartemental)
η	(tray)efficiency measurement adjustments (intradepartemental)
λ	average latent heat per unit mass
	lagrange multiplier for the equality constriants

μ	average or mean, lagrange multiplier for the inequality constraints, viscosity
ξ	slack variable
ρ	density
σ	standard deviation
σ^2	(co)variance
τ	mean reactor residence time or time constant
ϕ	heuristic test variable for gross error detection
χ^2	quantile of the χ^2 distribution
ψ	residual variance covariance error

Superscripts

T	transpose
$\star$	refers to optimality or to saturated steam
$\hat{x}$	estimated value
$\tilde{x}$	adjusted, corrected
$\bar{x}$	average value or mean, closed set

Subscripts

0	refers to initial condition, also refers to intradepartemental streams (data rec.)
a	air
c	controller
d	drum and downcomer
e	boiler feedwater
ev	evaporator
F	fire box
f	fuel
g	flue gas
gr	gas in the riser
gs	gas in the superheater
h	header
i	ith element of an array
in	entering the header
nom	nominal
out	leaving the header

r	riser
ref	reference
rt	riser tube
s	steam or superheater
sh	superheater
st	super heater tubes
sw	spray or attemporator water
qss	refers to quasi steady state

Special Symbols

$\forall$	for all
$\exists$	there exists (at least one)
$\exists!$	there exists exactly one
$\subset$	is part or subset of
$\wp$	product of heat transfer coef. and area as single parameter

CHAPTER 1

INTRODUCTION

1.1. SCOPE AND SIGNIFICANCE

The operating conditions of a chemical plant are determined by a control system that, in general, is layered. The lowest layers enforce the setpoints. Higher layers set objectives for individual plants based upon e.g. market situation. A layer in between uses these plant objectives together with a plant model to obtain setpoints that will have to be maintained. This intermediary layer of control is the subject of this dissertation.

The connecting layer is conceived as an optimization based process, and is referred to as an *on-line optimizer*. An optimization routine trades off objectives with various constraints to obtain a set of optimal setpoints. Plant knowledge has to be used in this optimization. Ideally, the optimizer would make use of a detailed dynamic model of the plant. However, such detailed models are not often available or reliable. If a detailed dynamic model exists and if it is reliable then such a model is usually characterized by an extremely large number of variables and equations. If the plant is a continuous plant, and the dynamics of the plant and the loop control as well as the disturbances acting upon the plant are such that the plant is operated under steady state conditions most of the time, then a steady state model of the plant is sufficient for effective on-line optimization. The major economic incentive is determined by the steady state under these circumstances. Steady state models are much more readily

available, are usually more reliable and contain less variables and equations. This dissertation is limited to these plant, i.e. *plants that are operated under steady state conditions, with transient periods that are relatively short between periods of steady state operation.*

The evolution of technology combined with changes in the world-wide market situation increased industrial interest in on-line optimization. Large scale applications of on-line optimization become more and more realistic as computer costs drop. Frequently the necessary hardware for data acquisition and processing is available in modern production faciltities. While computer costs decrease, energy costs increased dramatically in the seventies. This resulted in the heat integration of many plants. The energy efficiency of individual units was often improved significantly. A logical step following this unit by unit heat integration, combines units, optimizing their interactions. Instead of minimizing the energy consumption, the net operating cost can be considered as a more general objective to minimize. The net operating cost contains the energy consumption, but includes the product revenue as well. This optimization can be automated, and can be carried out on-line.

At the same time, many sectors of the petroleum, petrochemical and chemical industries suffer of world-wide production overcapacities. Competition is high and effective production plans are important. The competitive edge cannot be maintained any more by building larger units, and breaking production records. Existing operations have to be increasingly effective and efficient. Automatic setpoint updating in an optimization based way can be a powerful tool to achieve optimal production, while maintaining safety and environmental regulations.

On-line optimization is not optimal control. Optimal control is a common name for a group of techniques that optimizes a control objective in order

to specify an optimal plant trajectory from the current state to the desired state (setpoints). If an analytical solution can be obtained or approximated, then an optimal control law results. Optimal control is a setpoint enforcement technique, and therefore belongs to the setpoint enforcement layer of control techniques that receives setpoints from the on-line optimizer. Optimal control takes a plant to a new setpoint (optimal with respect to the economic objectives) along an optimal trajectory (optimal with respect to the control performance criterion). For the same reason, on-line optimization is not adaptive control, although the steady state model parameters can be updated based upon new data.

1.2. OBJECTIVES

1.2.1. Validation

Various applications of on-line optimization have been reported (see Chapter 2). However, often these reports remain vague, or fail to make general conclusions. A number of questions of general interest remain unanswered. Answers to some of them are given in this work. The literature on the various disciplines that coagulate in an on-line optimizer, does not address problems related specifically to on-line optimizers. These disciplines are statistics, optimization, systems theory, and estimation. The gap between theoretical results and applications is wide. This work attempts to narrow that gap.

1.2.2. Data Reconciliation

The volume of on-line measured variables in a chemical plant is vast. At any time, it is possible that at least one of these measurements is wrong. The quality of this stream of data has to be checked with mass and energy balances

for random and certainly for systematic errors. This is a computational challenge. What are the best ways (CPU effort versus accuracy, methodology) of implementing this "error filter" ? Data reconciliation is followed by model updating in an on-line optimizer. How does this sequence of data manipulations influence each other ? These questions are discussed and Chapter 3, and the results of that discussion are implemented in Chapter 5.

1.2.3. Estimation

The steady state model that is used by the optimization module is of course of extreme importance. The methodology of building models is not studied here. The literature (see Chapter 2) shows that statistical tests can indicate whether or not a model is "good enough" for use in an on-line optimizer. What is the cost of poor model in the system performance ? Chapter 5 illustrates how a poor model can deteriorate the optimizer performance. In Chapter 3, it is described how a failure of the estimation can be detected on-line. In Chapter 5 it is shown how unsteady state data can deteriorate steady state model updating.

1.2.4. Dynamics and Noise

Plant data have different levels and types of noise. This noise can be an important problem for parameter estimation, steady state detection, and for the optimization itself. The level of noise influences the feasibility of on-line optimization, as is explained in Chapter 3. The mechanistics of how noise influences the performance of the on-line optimizer are analyzed. The execution frequency of an on- line optimizer, as well as the approximate cut-off frequency of the low pass filter that should be applied to on-line optimizer input, is discussed.

1.2.5. Sensitivity Analysis

Sensitivity analysis is rarely ever used in the context of an on-line optimizer. Applications of sensitivity analysis are presented in this dissertation. Sensitivity analysis can be conceived as a technique to approximate the solution of the optimization problem in the neighborhood of the current solution ($\simeq$ in the neighborhood of the current operating conditions) with an analytical expression. It will be seen that sensitivity analysis, often combined with the statistical analysis of the model updating result provides a very useful tool for the evaluation of the status of an on-line optimizer. Some of the issues that can be tackled by applying sensitivity analysis are listed below. Detailed descriptions can be found in Chapter 3.

1.2.5.1. On-line Accuracy Assessment

On-line optimization will not be implemented in an industrial environment unless some safety reflexes are built in. Can the expected accuracy of the on-line optimizer be calculated on-line, such that unreliable results can be detected before they are implemented ? Sensitivity analysis can be used to obtain such accuracy results on-line. Furthermore, results of sensitivity analysis can be used to eliminate much of the high frequency interaction between plant and optimizer such that the optimizer can be run at a high frequency.

1.2.5.2. Feasibility Study

The development of an on-line optimizer is an expensive job that requires many man-hours of highly trained people. On-line optimization projects will have to compete for investment capital with several alternative programs for efficiency increase. Therefore, it must be possible to distinguish between interesting and uninteresting candidate processes for on-line optimization early on.

Again sensitivity analysis can be of use here, as is shown in this dissertation.

1.3. OUTLINE

In Chapter 2 the on-line optimizer will be considered in its relation with the hierarchy of the plant control structure. Convergence of the on-line optimizer to the true optimum is considered. Literature on on-line optimization will be discussed.

In Chapter 3 the internal structure of an on-line optimizer will be discussed. In that chapter, a collection of theory from the literature on the different disciplines that are utilized in an on-line optimizer is presented. Chapter 3 shows how the various sections of an on-line optimizer are composed and how they interact with each other. Original approaches and methodologies are proposed as well. A new steady state detection method is proposed in Chapter 3, but most important are the new applications in the field of applications of sensitivity analysis. Data reconciliation, model updating and optimization are reviewed from the point of view of on-line optimization.

In Chapter 4 the case studies that are used in this work are described.

Chapter 5 discusses the results of the case studies. The results are grouped by subject, and not by case study. The cases are not the primary subject of this work, but they serve as illustration. The on-line optimizer and its units stands central in this work, not the case study systems.

Finally, Chapter 6 presents a summary and conclusions as well as future recommendations.

CHAPTER 2

A SUPERVISORY CONTROL TECHNIQUE

2.1. INTRODUCTION

In this chapter, a selection of the literature dealing with on-line optimization is reviewed. An overview of this literature makes clear what the current state of the art is, what issues are treated by academia, and what problems industry solves or addresses. It is not claimed that this chapter discusses *all* literature concerning on-line optimization, but the most important publications in steady state on-line optimization are referenced.

This chapter is only concerned with literature explicitly on the subject of on-line optimization. In the next chapter, the mechanism of an on-line optimizer is discussed. In that chapter, literature is covered that is specifically connected with certain techniques such as data reconciliation or sensitivity analysis that are used within an on-line optimizer scheme, and some new approaches are suggested there. This chapter considers a selection of literature on on-line optimizer applications, and on the relation between the on-line optimizer and the control level above (plant scheduling) and the control level below (loop control) in the control hierarchy.

First, literature that discusses on-line optimization within a global hierarchical control structure is reviewed. Then, on-line optimization application reports are summarized. Finally, some conclusions are presented.

2.2. CONTROL HIERARCHY AROUND THE OPTIMIZER

2.2.1. Introduction

The objective of a control engineer is the control of an integrated process. For a large production unit, such as for instance a fertilizer plant or a refinery, this task can be vast and complex. The top-down approach solves control problems in a hierarchy of control layers of increasing scope and decreasing detail. At the same time, such a hierarchical structure makes it easier to allocate responsibility, account for success and performance of different units and reduce data transmission (Jennergren [1974], Atkins [1973]).

Hierarchy *as such* is a topic of research. A survey by Gijsbrechts [1985], listing 78 references of studies on organization of a hierarchy, brings up various *practical* issues that interact with the implementation and success of hierarchical structures in large scale systems. The most important of these issues are the inherent plant structure and the relation between model complexity and model scope. In the next paragraph, these issues are discussed in relation to the layered control structure.

The goal of a control structure is the safe production of specified products within environmental and operational constraints at optimal economic conditions. *Theoretical* studies on hierarchies often focus on the characteristics of convergence of the layered structure to this goal (Gijsbrechts, [1985]). An illustration of convergence theory of hierarchical control structures is given in the final part of this section on control hierarchy.

2.2.2. A Controlled Plant as a Hierarchical Organization

2.2.2.1. The Impact of the Inherent Plant Structure

The idea of organizing plant control in layers is not new. Bernard [1966] stated already that the most practical way to develop a control system is by examining and controlling the process at increasing levels of scope until the process objectives are met by the hierarchy of control layers. How are these layers determined, and what are their tasks ? Gijsbrechts [1985] reviews practical and theoretical results in the field of hierarchy. He lists the inherent structure of a system (in this case : the plant that has to be controlled) as the most important factor determining the character of the structure of the hierarchy that organizes this system (in this case : the layered control system). The number of meaningful managerial hierarchies is limited. The physical, operational and functional structure of the plant determines the structure of the control hierarchy, which is not all that surprising. In process control, these levels were among the first issues control engineers conceptually agreed upon. A plant is physically divided in a number of units, in which distinct chemical operations are carried out (e.g.: a catalytic cracker, an extraction column, a heat exchanger). Units are grouped in sections marked by some storage of important intermediary products (e.g. sulfuric acid production, ammonia production, ammonia sulfate production), and these sections are grouped in separate plants. A flow control loop, for instance, generally uses only local information (information from within the same unit as the controlled flow). The setpoint of this flow controller could however be determined by an optimizer based upon the global plant behavior. The flow controller belongs to a much lower level than the optimizer.

Bernard [1966] suggests four levels. In the first level, conventional loop control maintains plant setpoints of variables such as flow, temperature, pressure and liquid levels employing relatively simple control schemes. The second level maintains plant heat and material balance, and compensates for important interaction phenomena. Bernard [1966] includes "unconventional feedback

schemes employing nonlinear, logic, or sampled data concepts" in this second control level. The author refers to techniques that are now well known, such as adaptive control, IMC, minimum variance control, dead time compensation, certain applications of artificial intelligence, and many others. The third level of Bernard [1966] buffers the lower two levels and the optimizer, which is located on the fourth level, by changing the plant operating point as dictated by the optimizer. This level could include techniques such as DMC or IDCOM. An important task of such a level is to maintain constraints. More recent authors (e.g. Coombes *et al.* [1983], Jang *et al.* [1987], Lasdon and Baker [1986]) will often group the first three levels of Bernard [1966] together in one single level. This single level will then have the task of enforcing the process control setpoint on the plant while keeping certain constraints satisfied. Bernard and Howard [1970] even add a "zero level" to the four levels of Bernard [1966]. That zero level contains procedures for safe emergency shut down and system failure warnings. Later, this zero level is expanded with the "safety reflexes" of the process, also called "security control" (Isermann, [1981]). These control mechanisms contain early reactions to specific disturbances to stop the process from going into an emergency status.

The fourth level of Bernard [1966], or equivalently the second level of Coombes *et al.* [1983], determines the best operating condition of the plant in an optimization based way. It is this level that is the subject of this dissertation. Bernard [1966] mentions that the optimization level is not necessarily worthwhile, and that the optimization task can be done off-line. Seventeen years later, Cutler and Perry [1983] state that optimization is very useful in maximizing profits, and that doing it on-line has a number of significant advantages over off-line optimization (see section 2.3.2).

On top of the optimization level, Bernard [1966] suggests that other levels

may be added, but no details concerning the task or scope of higher levels are listed. It is clear that this higher level has to provide instructions to the plant optimizer. Coombes *et al.* [1983] refer to this level above the optimization level as the *production scheduling and production control* level. The task of this level is to maximize company or division capacity utilization while spreading the production activity over the smallest possible number of unit processes.

Lasdon and Baker [1986] describe a hierarchical operation for a complete corporation. The lower three levels of the hierarchy described by Lasdon and Baker [1986] correspond to the three levels of Coombes *et al.* [1983]. Lasdon and Baker [1986] suggest six levels in total.

(1.) *Strategic and tactical planning* at the top level decides the long range goals of a corporation. This level includes such tasks as forecasting and risk management, capital budgeting, market expansion, and facility expansion. The horizon of this operation can be several years. The input to this level is a very broad range of economical, commercial, technical and social factors, as well as information from the lower levels, such as quarterly reports. The output of this level is a strategy that will serve as input and guideline to the level below.

(2.) *System Operations Planning* establishes multi-plant logistics system tactics. The time horizon of this level is of the order of magnitude of several months to one year. This level obtains instructions from the higher level and adds e.g. more detailed market information to it. This level produces plans and budgets, evaluates raw material and sales contracts. The level also produces production targets, and passes these on to the lower level.

(3.) The *Plant Operations Planning* determines operating goals and tactics for single plants for one to three months. Linear programming is often used for this task. Yield and quality data are incorporated. Time boundary

conditions determine initial and final inventory as well as production conditions.

(4.), (5.) and *(6.)* correspond to the *scheduling*, the *optimization* and *setpoint enforcement* layers of Coombes *et al.* [1983].

Lasdon and Baker [1986] also point out that the projected changes in technology such as distributed computation and improved communication technology, are expected to keep these functions *separate*. Future enhancements in this structure will probably be caused by improved communications between the different levels in the hierarchy, and by the development of new general decision making or information processing tools that can be used at several levels. The development of more comprehensive models with a larger scope is not immediately expected. The availability of models is another important factor determining the control hierarchy, and is focussed on concisely in the next section.

2.2.2.2. The Trade Off between the Model Complexity and Scope

Gijsbrechts [1985] mentions differences in existing model type and complexity as another important practical characteristic of a system such as a controlled plant, influencing the hierarchical organization of that system. The process control levels that are suggested above are characterized by the different types of models that are available. The loop control level, e.g. flow control, is fast enough to make a model unnecessary. The "plant section" (i.e. the flow loop) is operated on directly. More advanced control techniques use detailed dynamic models. Use of dynamic models can certainly be expected if constrained control has to be applied. That is the case for adaptive control, IMC, etc. An optimizer level uses a nonlinear steady state plant model. The production scheduling level uses less detailed linear plant models. In practical applications, as the scope of the model becomes larger, less details are modeled. States are lumped, others are neglected. Relations are often linearized in models at higher levels. Com-

plex dynamic models of complete plants are in general not available, and if they are, they are useless because the number of variables and equations involved is too large to manipulate. Furthermore, models of that size are difficult to maintain and troubleshoot. As the scope of a model becomes larger, it is not efficient to maintain the same level of detail as is possible on a smaller scope.

A summary of the layered hierarchy and more particularly the position of on-line optimization in it, is given by figure 2.2.1.

2.2.3. Requirements for the Loop Control Layer

In order to perform optimally, a plant will often be driven to its constraints. Because of the plant model mismatch, constraints that are met by the optimization may not be met after the optimal setpoints are implemented in the plant. Therefore it is necessary to install a loop control system that is capable of handling constraints. This point will be repeated and illustrated in the remainder of this chapter. Cutler and Perry [1983] suggest the implementation of Dynamic Matrix Control (DMC), or one of its variants. Garcia and Prett [1986] also suggest the use of DMC, but suggest some alternatives such as IMC as well. References on process control under constraints can be found in Chang and Seborg [1982]. It was pointed out already by Bernard [1966] that the control loop system must be able to change the operating condition from an old state to a new state moving along constraints. Cutler and Perry [1983] point out that operators are not able to deal with the multivariable aspects of such control actions, and typically will move the process away from the constraints to let the PID controllers move more freely. They estimate that the *operators wrap* represents about 3 to 5% of the value a process adds to the raw products. This loss has to be avoided by the ability of the control system to follow constraints tightly.

DESCRIPTION	COMMUNICATION	TASK AND SCOPE
strategic & tactical planning systems operation planning plant operations planning production scheduling	● The combination of higher layers obtain their information from a broad range of sources depending upon the planning level. ● An economic model of the plant is essentially what is given to the optimizer level in the form of objectives and constraints.	● Determination of the economic objective of the plant. ● Determination of the various constraints posed upon the operation. ● Determination of cost of intermediates used, and price of intermediates produced, as well as of final products. ● Scope is from years (top level) to days or weeks.
optimizer model identification plant optimization setpoints	● **Cost of raw materials and intermediates are given from higher levels, as well as the objective and constraints.** ● **Optimal setpoints are passed to lower level.** ● **Plant data are received and analyzed.**	● **Optimization of the setpoints based upon the economic model the higher layer provides and the plant behavior data from the lower layers to which a model is identified.** ● **Check for accuracy with which the task is expected to be completed.** ● **Describe status of the task.** ● **Interrupt when unreliable.**
loop control PT DMC	● Setpoints are set by the on-line optimizer or are fixed. ● The loop control determines the values of the manipulated variables and finally the position of the valve actuators.	● Enforce setpoints (scope=hours to sec.) ● Keep constraints satisfied. ● Override optimizer results if necessary.
plant	● Sensing devices record data and send these through to the loop control and higher levels.	● Production of desired products while maintaining safety and respecting environmental specifications. The plant must include devices to prevent potentially dangerous conditions even if any of the higher levels is cause of such a condition.

Fig.2.2.1. Summary of the Layers of a Plant Control Structure

2.2.4. Convergence (= Stability) of a Layered Control Structure

In this section, we summarize some convergence and stability results for a layered optimizing control structure as published by Brdyś and Roberts [1987]. This publication is certainly not the only one on the subject of stability of layered control systems, but the approach considered here is similar to the on-line optimization configuration we study. Therefore, the results of Brdyś and Roberts [1987] illustrate very well the convergence characteristics of the on-line optimizer scheme that is the subject of this dissertation.

The optimum operating point of a plant can be formulated as the solution of the following problem :

$$\begin{aligned} &\min_{x,y} f(x,y); \\ &\qquad \text{subject to } y = q^{\star}(x); \\ &\qquad\qquad \text{and } g(x) \leq 0. \end{aligned} \tag{2.2.1}$$

in which x is a vector of setpoints, y is a vector of plant outputs, g is a vector of constraints and f is the optimization objective. The *exact* plant behavior is given by $q^{\star}(x)$. However, $q^{\star}$ is unknown. An approximation $q(p,x)$ is used. In this approximation, p is a vector of adjustable model parameters.

The most important differences between the scheme proposed by Brdyś and Roberts [1987] and the scheme that is used in this dissertation and presented in Chapter 3 are that *(1.)* an optimization technique is used that compensates for differences in plant gradients and model gradients (more on such a technique can be found in section 3.5.3.4), and *(2.)* that the setpoints are implemented using :

$$x_{i+1} = (1-K)x_i + Kx^{\star}_{i+1}. \tag{2.2.2}$$

In this equation, x_i is the previously implemented setpoint, $x^{\star}_{i+1}$ is the new optimal setpoint that is the result of the optimization, x_{i+1} is the new implemented setpoint and K is the implementation gain matrix. Brdyś and Roberts

[1987] limit their study to gain matrices of the form $K = kI$, with I the identity matrix and k a scalar. In the scheme that is presented in Chapter 3, the correction for the difference between model and plant gradients is not always applied, since it requires a large amount of steady state plant data close to the current operating point. Also, in the scheme that is studied in the remainder of this dissertation, the implementation gain $K = I$ (or $x_{i+1} = x^*_{i+1}$).

The most important assumptions accepted by Brdyś and Roberts [1987] are the following :

(i) the feasible set is bounded, and the set of model parameters that minimize the estimation criterion is compact ;

(ii) the functions g_j, $\forall j$, are convex and continuous ;

(iii) the function f^* is Fréchet differentiable in the feasible set, and its derivative is Lipschitz continuous with a constant δ, i.e.:

$$||(f^*)'(x+h) - (f^*)'(x)|| \leq \delta||h||; \forall\, x+h, x \text{ in the feasible set ;} \quad (2.2.3)$$

(iv) the functions q and q^* are continuous ;

(v) the derivative of f with respect to the parameters is uniformly monotone.

If these assumptions are valid, then $\exists k$ such that the following holds :

(1.) the series of optimal setpoints will converge, if none of the plant inputs change ;

(2.) every array of optimal setpoints x_i will satisfy the operation constraints, and every new setpoint will perform better than the previous setpoint.

Assumptions and conclusions have been simplified here. More details can be found in Brdyś and Roberts [1987]. They conclude that under mild assumptions, there exists a k such that the on-line optimization scheme will converge to an optimal operation without violating the operating constraints if the plant inputs remain constant. That indicates that plant model mismatch, if no gradient difference compensation is applied, can inhibit convergence and/or result in

violated constraints. It also indicates that application with $k = 1$ can cause the scheme to diverge or violate constraints. Finally, the theory did not address what can happen if the plant inputs vary. As will be shown in Chapter 5, an on-line optimizer can actually amplify certain input frequencies. Noise was not considered in the study by Brdyś and Roberts [1987]. Noise can deteriorate the model parameter estimation quality significantly. It can be concluded that although it can generally be expected that the scheme will converge, checks should be built in to interrupt the optimizer implementation in case potential divergence is detected. A method to do so is presented in Chapter 3 and is illustrated in Chapter 5.

2.3. ON-LINE OPTIMIZER APPLICATIONS

2.3.1. Introduction

In this section, a number of publications on the application of on-line optimization using steady state models are reviewed. First, general publications on on-line optimization are considered. Then reports of specific applications are reviewed. These reports concern applications in boilers, utility systems, olefins production, olefins separation, refineries, paper and pulp mills, ammonia processes and bench scale or pilot scale plant optimization. Finally, some conclusions are presented.

This literature review is limited to relatively recent applications. On-line optimization has been discussed since the fifties.

2.3.2. Optimizing Plants

On-line optimization is discussed by industry as well as academia as an extension of loop control that drives a plant relentlessly to its most profitable operating condition. In this section, some general papers on the topic of on-line optimization are summarized.

Cutler and Perry [1983] prefer the on-line application of plant optimization to off-line optimization because it provides a closer enforcement of the optimal policy. These authors devote attention to the influence of data accuracy and model parameter estimate accuracy on the success of on-line optimization. Cutler and Perry [1983] consider five principal factors involved with on-line process optimization : model parameter updating, process constraint determination, process models for prediction of plant behavior, feed and product values, and interaction with the process control system. The authors suggest that the probability of success of on-line optimization can be set equal to the product of the

relative accuracies of these five factors. A module (e.g. the model updating) may for instance have an expected accuracy of a percent in its prediction. Another module may have an accuracy of b percent. This means that the result of the first module may be $1.0a$ times the true result. Then, in the worst case, the results of both modules combined may be $a + b + (\frac{ab}{100})$ percent off. In the next chapter, an alternative and new approach to assess the potential success (= feasibility) of on-line optimization is presented. It is better to base an estimation of the feasibility of on-line optimization on statistical principles, than to approximate it with a product of accuracies of involved factors. Cutler and Perry [1983] discuss some of the particular reasons causing inaccuracies in the five factors that were mentioned before. Cutler and Perry [1983] suggest an on-line identification of the model parameters. They stress that in model building, not always enough attention is devoted to *mechanical* phenomena such as maximal power of turbines, maximal pressure drop through equipment, etc. According to the authors, these factors influence the operation very significantly, since they are incorporated in many constraints, while model builders often devote attention mainly to kinetic, equilibrium and thermodynamical facets.

Garcia and Prett [1986] review advances in model-predictive control. This review includes on-line optimization. The authors specify a control phase where dynamic process representation is used to optimize a performance objective function which penalizes deviations from the setpoints and potentially includes a price for control actions. On top of this control layer is an optimization layer using a steady state process representation, maximizing profit. The authors state that the operating point is *updated with a frequency comparable to the disturbance frequency*. The division between both phases is justified if the disturbances with important economical impact are much slower than the inherent dynamic disturbances. Even if this is not the case, the split between

a steady state optimization level and a dynamic optimization is made, because it is impossible to solve large scale on-line optimization problems with dynamic models with the current state of the technology. This split confirms the conclusion made in section 2.2.2.2 on the impact of model scope and complexity on the control hierarchy. The optimization phase is easier to comprehend than the multivariable control problem with plant constraints. Therefore, it often is appealing to implement on-line optimization without devoting attention to the underlying control scheme. Nevertheless, one does not make sense without the other. In the reports on on-line optimization that are summarized below, it is often mentioned that implementation of the on-line optimizer scheme goes together with a retrofit or complete reorganization of the existing control system.

Jang *et al.* [1987] study the on-line optimization of constrained multivariable plants. They consider on-line optimization using steady state models as a special case of the technique of dynamic optimization they propose. In case a plant is operating most of the time at a steady state, then a dynamic model is not necessary and steady state methods can be applied. Whether or not a steady state model can be used, depends upon how the frequencies of the input disturbances and process time constants relate to each other. It has to be pointed out, that dynamic models are not always available, or that it is not always feasible, as mentioned above, to solve an on-line optimization problem with a detailed dynamic model. Jang *et al.* [1987] identify the dynamic model on-line using a least squares estimation technique. They stress that one has to avoid identifying models with a data base that has too high a covariance. Using a data base with a high covariance will make the estimated parameters unreliable, and cause oscillations at the optimum (Bhattacharya and Joseph [1982]). This is illustrated in Chapter 5. For this reason, it is suggested in Chapter 3 that new data are checked and compared to data already stored, and

are only used if the new set is sufficiently different from the sets already stored. Furthermore, it is suggested in Chapter 3 that a statistical analysis should be carried out on the estimation result, such that unreliable parameter estimates are detected. Jang *et al.* [1987] illustrate their technique with a simulated two-CSTR example. Chen and Joseph [1987] illustrate the same approach with a simulated ethylene oxide reactor example. Bhattacharya and Joseph [1982] use this dynamic optimization technique as well. They do not use a steady state model technique because they argue that collecting steady state data is too time consuming.

2.3.3. Optimization of the Operation of Steam Boilers

Steam boilers are major energy consumers (see Chapter 4, section 3). Optimization of boiler efficiency is therefore very interesting. The most important factor in keeping boiler efficiency up, is good maintenance. Good maintenance will keep the heat exchange surfaces clean by soot blowing and proper blow down. Applications of on-line optimization are limited to the optimization of excess air, and boiler load allocation if several boilers are operated in parallel. Excess air has a very large effect on boiler efficiency. The optimization of the excess air in the air fuel mixture has become part of the loop control of a boiler. A very good description is given in e.g. Lipták [1987], Payne [1985], Garcia-Borras [1983], Jackson [1987]. The optimum air excess point does not coincide with the point for which the air excess is zero. That is so because no burner can provide ideal mixing. At the minimum loss point, the fluegas losses compensate for the better combustion. Special care has to be taken during transients in the fuel consumption to avoid the danger of explosion. Fired boilers (and furnaces in general) operated with low excess air are fuel efficient but unstable (Gomes, [1985]). As the demand increases , the air flow has to increase immediately,

while if the demand decreases the air flow is decreased only after the fuel flow has been decreased.

Boiler load allocation is an example of on-line optimization at the supervisory level. It is also one of the earliest applications of on-line optimization (Blevins *et al.* [1980]). A steady state model is used to predict the plant performance. Optimal setpoints are calculated based upon the expected plant performance. Five publications on the subject are reviewed here in an increasing order of complexity.

Cho [1978] discusses optimal boiler load allocation. The author considers a network of parallel boilers with a common header that provides utility steam. The optimizer solves the problem of how to allocate steam load among n boilers such that the total steam production cost is minimized. The optimization is constrained by the requirement that steam has to be supplied to fulfill the plant consumption. Cho [1978] does not include minimal and maximal individual boiler loads in his problem description. A quadratic efficiency curve is used as a model for the steady state relation between the boiler load and its efficiency. The coefficients of the model are obtained by fitting this second order polynomial to steady state boiler efficiency data. The operating cost of every boiler is then approximated as proportional to the ratio of load and efficiency. The sum of these individual costs is minimized with respect to the individual boiler loads, and with respect to the steam demand. A penalty function method is applied for the minimization. Cho [1978] does not give any detail about why the second order model is used or how it is identified. It is not mentioned whether or not the identification has to be performed on-line and the estimation algorithm is not specified. The frequency of execution is not discussed. The author, just as any other author on steady state on-line optimization, does not use any statistics to quantify the influence of random data error and model plant mismatch on

the accuracy of the parameter estimates and on the optimization result and solution. Cho [1978] presents a four boiler example. This example is analyzed in depth in Chapter 5.

Cho and Blevins [1980] add a system of turbo generators to the boiler system for power generation (Cho [1978]), as well as throttle and extraction valves. They optimize the total system. The objective is the minimization of :

the cost of steam input into the turbo generator network
+ the cost of electric energy purchased from the utility
company (base cost as well as peak load fee)
− the value (as if purchased from the utility company) of
the electric energy production of every turbo generator.

An example is not given. The authors point out that steam mass flow measurements should balance to within about 3% in order to have sufficient accuracy to apply the technique. A justification for this number is not provided.

Green and Al ai-Shaikh [1980] also discuss on-line optimization of boiler load allocation. These authors use a second order relation between the boiler load and the boiler efficiency as well. Green and Al ai-Shaikh [1980], as opposed to Cho [1978], devote attention to the parameter updating method. First they discuss the determination of the boiler thermal efficiency. A heat balance over a steam boiler includes three terms : the rate of heat input on one side, balanced by the sum of the rate of heat output and the rate of heat loss on the other side. Therefore, boiler thermal efficiency can be calculated in three different ways. The first method calculates the boiler thermal efficiency as the ratio of the rate of heat input to the rate of heat output :

$$\text{efficiency} = \frac{\text{rate of heat output}}{\text{rate of heat input}}. \tag{2.3.1}$$

The second method combines the rate of heat loss and the rate of heat input in

the following way :

$$\text{efficiency} = 1 - \frac{\text{rate of heat loss}}{\text{rate of heat input}}. \tag{2.3.2}$$

Finally, the third method compares the rate of heat loss to the rate of heat output :

$$\text{efficiency} = \frac{1}{1 + \frac{\text{rate of heat loss}}{\text{rate of heat output}}}. \tag{2.3.3}$$

The true the values of the three previous expressions are of course equal. However, the accuracy with which all three can be calculated from plant data, is very different. Heat inputs are mainly determined by the amount of fuel that is burned and the air (including moisture) that is used for combustion. Heat input also includes the feed water and electric power driving forced convection fans, feed water pumps, coal crushers etc. The heat output of a boiler is contained in the steam delivered to the steam header. Blow down and soot blowing steam is not useful output, but should be considered as heat losses. Other important heat losses are accounted for by stack losses and radiation losses. The stack losses can be itemized as due to incomplete combustion, sensible heat of flue gas and combustion water. Green and Al ai-Shaikh [1980] show that for realistic measurement and data accuracies, the method combining heat input and heat loss is the most accurate method to calculate the boiler thermal efficiency. The method combining heat output and heat loss is a close second, but the method combining heat input and output can be expected to be approximately five times more inaccurate for oil and gas fired boilers, and even much more inaccurate for coal fired burners. The reason for this result is that boilers have high thermal efficiencies. The rates of heat input and heat output are approximately an order of magnitude larger than the rate of heat loss. Therefore, although heat loss may not seem easy to measure, the absolute error on the number obtained is smaller than the error in the rates of heat input and output. Therefore, the methods using heat loss with either heat input or output

are more accurate. The authors do not discuss the possibility of using all three rates of heat transport combined with reconciliation. Green and Al ai-Shaikh [1980] run the efficiency calculation every minute.

The authors points out that the coefficients of the quadratic relation between boiler load and efficiency change slowly with time, and therefore have to be updated on-line. A modified Kalman-like filtering of the steady state data is suggested. This method is a recursive regression mechanism with an exponential forgetting factor. Before new data are admitted to the data base, the process computer checks for steady state. Green and Al ai-Shaikh [1980] do not specify how that is done. It is also checked whether or not the data are within a certain envelope in the load versus efficiency space.

The optimization problem that is solved by Green and Al ai-Shaikh [1980] is identical to the one suggested by Cho [1978]. Green and Al ai-Shaikh [1980] report fuel savings of 1 to 3% by boiler load optimization. The authors mention that 2.2% savings in energy were obtained on a site that involved four gas fired boilers with a capacity of 120,000 lb/h each.

Ko [1987] also discusses on-line optimal boiler load allocation. The steam generating system that this author considers is structurally identical to the systems studied by Green and Al ai-Shaikh [1980] and Cho [1978]. A number of parallel boilers feed a common header. Some of the boilers are swing boilers and the others are base load boilers. The fuel load of these base load boilers is directly determined by the on-line optimizer. The optimization problem that is solved by the on-line optimizer is identical to the optimization problems solved by Green and Al ai-Shaikh [1980] and Cho [1978]. The objective function in both publications is the total cost of steam production. The model for boiler efficiency as a function of boiler load is a quadratic relation in both papers. Ko [1987] does incorporate minimum and maximum individual boiler loads. The

author states that the numerical solution of the optimization problem can be cumbersome. Other authors do not mention this point, and no problems were experienced in the optimal boiler load allocation case study that is presented in Chapters 4 and 5. Ko [1987] suggests another optimization method, in which at every optimization incremental load is added to the boiler with least incremental boiler cost, or subtracted from the boiler with the highest incremental boiler cost. The optimizer is run every ten to fifteen minutes, and in this way the boiler load allocation will approach the optimum, at which the incremental steam cost for every boiler is equal. A model (quadratic) can be used, but only two parameters are involved per boiler, since the derivative of the efficiency with respect to the boiler load is used and is assumed to be a linear function of the boiler load. Three efficiency data points allow one to estimate the slope of the cost as a function of the individual boiler load. Moreover, if no model is used, two efficiency observations will give the local incremental load. The algorithm can be applied until these observed local incremental loads of all boilers are equal. In that way, since no model is used, all potential problems with model plant mismatch are avoided. However, the method used is an evolutionary operation type method (the optimizer operates directly on the plant), which is not only very dependent upon the numerical optimization technique but is also generally accepted as a poor method only to be used in case no other possibility is left (Latour, [1980]). Iteration on the plant is a slow process compared to iteration on a model. Ko [1987] illustrates his approach with a two boiler example. He stresses that the system he proposes would results in a much quieter and smoother process behavior than the "traditional" systems (such as Green and Al ai-Shaikh [1980], and Cho [1978]). It is not clear why this is the case. The author does not compare the method Cho [1978] and others use with the method he proposes using simulations or pilot plant results. Boilers

function as low-pass filters, and smoother input changes do not necessarily result in smoother output signals. Ko [1987] reports fuel savings of up to 5%.

Matsko *et al.* [1982] discuss optimal boiler load allocation. They introduce the method of reallocating loads at real time in order to obtain equal incremental costs for every boiler, used by Ko [1987], but describe it in a much more concise manner.

Lipták [1987], in a survey paper on improving boiler efficiency, also discusses boiler load allocation, and suggests a method similar to the one used by Ko [1987]. Lipták [1987] notes that flow meters have to be sufficiently accurate, and that the error contribution of the flow meters should not exceed $\pm\frac{1}{2}\%$ to $\pm\frac{3}{4}\%$. Most important is that Lipták [1987] lists optimal boiler load allocation as the *last* method to increase boiler efficiency. Optimal boiler load allocation, if profitable, can only be interesting if all other methods to increase individual boiler efficiency are properly applied. These methods are, according to Lipták [1987] :

- ◇ good loop control for setpoint enforcement ;
- ◇ excess air or air to fuel ratio optimization ;
- ◇ fluegas composition monitoring ;
- ◇ steam pressure optimization (also : optimization on the consumers side) ;
- ◇ blow down and soot blowing optimization ;
- ◇ feed water pump rpm optimization.

Proper feed water quality will reduce downtime and increase boiler efficiency as well (Wilcox [1978]).

2.3.4. Optimization of Utility Systems

Poje and Smart [1986] discuss on-line utility energy optimization in a chemical complex. They limit themselves to the two most important energy carriers :

electricity and steam. Electrical power is purchased from a utility supplier, and in certain cases it is produced from high pressure steam. Steam is produced from heat generated from burning fuels, or recovered from process heat. Poje and Smart [1986] consider a general energy system network. They consider two groups of boilers, one producing steam at 1600 psi (110 atm.), and the other group at 600 psi (40 atm.). They use four pressure levels of steam consumption : 1600 psi, 600 psi, 200 psi (14 atm) and 50 psi (3atm). The higher pressure steam is usually consumed by turbines, the lower pressure steam is used for heating and is also consumed as process steam (cogeneration of steam and electric power). Electric power is used to drive motors. The steam and electricity energy demands are set by the consumers, and the energy system is self regulating. The steam systems are pressure controlled. The authors present an example that is schematically given on figure 2.3.1. In the example Poje and Smart [1986] present, the cost of meeting the energy demand is determined by (optimization variables) :

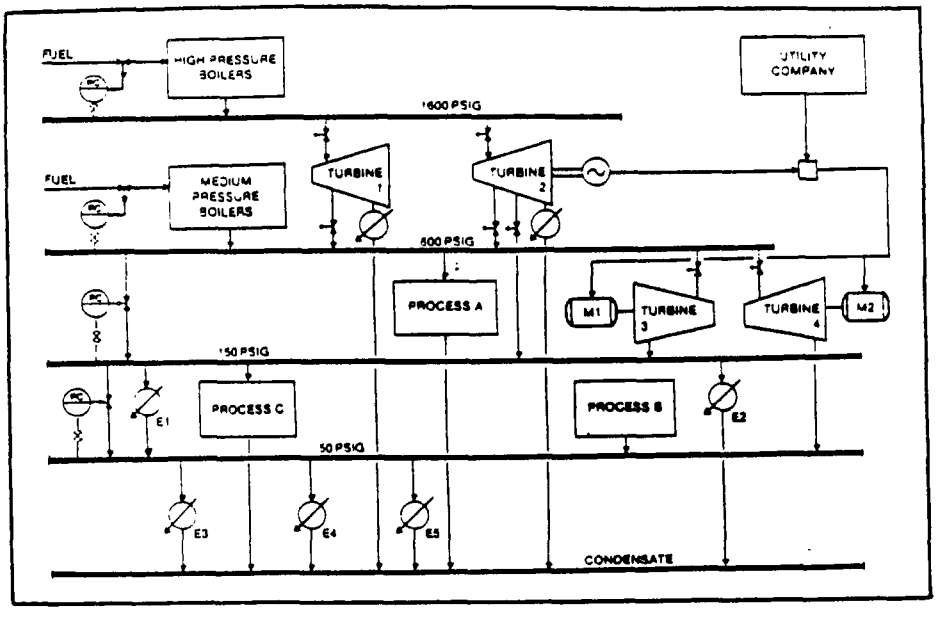

Fig.2.3.1. Utility System Considered by Poje and Smart [1986]

- ◇ extraction flow from turbine 1 and 2 to the 600 psi header, and turbine 2 to the 150 psi ;
- ◇ electrical power generated from turbine 2 ;
- ◇ on/off status of motors 1 and 2
- ◇ pressures of the steam headers
- ◇ temperature of superheated steam produced by high and medium pressure boilers.

Constraints are given by :

- ◇ maximum and minimum loading of boiler groups ;
- ◇ maximum power generating capacity in turbo generator 2 ;
- ◇ maximum header pressures and temperatures ;
- ◇ maximum extraction flows from turbine stages.

The energy requirements of production units are not constant. The efficiencies of boilers and turbines depend upon their load. Different fuels have different costs. Electricity costs are not constant, but are a complex function of various characteristics of the consumption pattern. All these variables indicate that a n optimal energy production policy has to be adjusted frequently. Poje and Smart [1986] distinguish between two classes of plant optimization. The first class of plant optimization methods minimizes the energy cost by varying the load of the different utility units in a continuous space. The second class of plant optimization methods includes on/off status of equipment in the list of optimization variables. Boilers, for instance, may be shut down or started up in order to optimize the energy cost, since boilers only have a limited range of safe operation. Poje and Smart [1986] refer to the first class of plant optimization methods as "constrained optimization", and to the second class as "global optimization". This qualification does not refer to optimization specification constraints, but to a set of optimization variables "constrained" to continuous

variables or "global". The authors suggest that the result of a "constrained" optimization can be implemented on-line by the process computer. Starting and stopping of equipment is a management decision, and is therefore not implemented on-line.

Poje and Smart [1986] state that on-line optimization of an energy system has advantages to off-line optimization. Energy demands fluctuate too frequently to make an off-line optimization effective. The authors list four requirements to make the on-line optimization of an energy system successful :

- ◇ a common database shared by the optimizer with all management reports that involve data on energy usage and cost ;
- ◇ process data validation ;
- ◇ an effective energy system model ;
- ◇ implementation in a supervisory manner.

Poje and Smart [1986] suggest that data validation is carried out in three ways. First, data are compared to high and low values. Then, the measurements are checked and compared with a maximum rate of change. Finally, data and mass balances are checked. The authors point out that a distributed way of implementing optimization allows for more robustness.

Poje and Smart [1986] estimate that a complete energy optimization computer system costs about $ 600,000. This figure includes a computer with 1 megabyte main memory, and a 264 megabyte mass memory, a tape drive and line printer. Furthermore, 2 consoles, 2 alarm printers and software are included. The cost of connecting the computer to the process is estimated to add between $ 50,000 and $ 150,000 for 500 inputs. Costs for additional equipment and retrofitting of the control scheme, which is often necessary if on-line optimization is implemented in an existing plant, are not included. A potential benefit of 2% to 10% is mentioned by Poje and Smart [1986]. This return

would result in a short payout time (months). The authors do not refer to any industrial applications.

Nath *et al.* [1986] suggest the joint optimization of process units and utility systems. The authors introduce a conceptual split of a plant in two distinct parts : the process units and the utility subsystem. The utility system is further partitioned into a steam subsystem, a fuel subsystem, an electric power subsystem and a compressed air subsystem. These subsystems interact with each other. Fuel and electricity are supplied to the plant by outside companies. Fuel is used for the production of steam, and steam can be used for electric power production. Fuel, steam (at various pressures) and electricity are provided to the plant units, and under certain conditions compete with each other (e.g. as a source of heat). Compressed air is provided to the instrumentation and is consumed as process air. Air is available at different pressure levels, and may be dried at the lowest pressure level for instrumentation purposes. The authors point out that typically, the process units are optimized separately from the utility subsystem. The utility subsystem is optimized after the units are optimized and all utility demands are known. Prices are set for the utilities but these prices are not often reconciled. Often the utility subsystem is evenly split up in its four main components and each subsystem is optimized separately. Nath *et al.* [1986] propose to optimize the complete utility system and the production systems combined. They use a small imaginative example plant simulation as illustration. They report an application in a Union Carbide plant. A mixed integer linear model was used, and the authors claim that such a model can represent plant units and utilities satisfactorily. They mention that the technology will be transferred to other plants.

Craw *et al.* [1988] report on the real time optimization of large scale cogeneration systems. They discuss a general purpose power (steam and electricity)

station optimization package. The authors single out cheap computing power and robust and fast converging optimization algorithms as the main technological advances that have made real time optimization possible. A detailed steady state model of a power station is developed. It contains rigorous heat and mass balances. The model is nonlinear. The authors point out that this nonlinearity is very relevant to the problem that is to be solved. Even the best linear models are inaccurate and their use can lead to significant suboptimality of the solution. The authors do not discuss any details of the power plant model equations that are used in the package. An infeasible path approach is used for the optimization module, but the algorithm is not specified. The optimizer is run every few minutes. It can also be used off-line to assist the operating staff in determining what the effect of certain operating policies would be. On/off variables seem to be included in the list of optimization variables. The authors list a cost reduction of 2.5% for the Wilton Power station. This complex power station is located in ICI's production facilities in Teeside, UK. It typically produces 200 MW electricity and exports 820 mtons/h steam. It includes five boilers that produce steam at 118 bar and three boilers that produce steam at 63 bar. Steam is produced at four pressure levels. Added to that are 11 turbo alternators for electricity production. The real time optimizer was commissioned in 1982, so that several years of experience can be looked back upon. Data reconciliation or sensitivity analysis are not discussed in the report.

2.3.5. Optimization of Olefins Producing Units

Sourander *et al.* [1984] report on the control and optimization of olefin cracking heaters in Neste Oy's ethylene plant (Finland). First an improved loop control (setpoint enforcement) is discussed. It is interesting to note that the loop control scheme can carry out constraint control. They provide two

different optimization objectives. The first leads to maximization of ethylene production against plant constraints. The second is a profit optimization including factors for feedstock costs, product values, energy costs etc. Constraints and objectives are linear, and linear programming is used. Use of model updating and techniques to update the model are not discussed.

Näsi *et al.* [1985] present an application of on-line modeling, control and optimization of an acetylene hydrogenation unit in the same Neste Oy ethylene plant (Finland). An acetylene hydrogenation unit is composed of two reactor beds, with cooling (heat exchanger) in between. The reactor is autothermal. The objective of the operation is to maximize acetylene removal from the deethanizer overhead, while keeping the ethylene loss to secondary reactions as small as possible. The net ethylene gain is maximized while keeping the acetylene concentration below a certain value. The optimization variables are bed inlet temperatures and the flow rates of the hydrogen additions. The model (adiabatic bed with plug flow) is updated for catalyst deactivation.

Funk *et al.* [1984] discuss the optimization of an ethylene plant. They state that there is no "best" way to run an ethylene plant. In the past, according to these authors, it was sufficient to set new production records from time to time in order to be regarded as successful. Production is not any more the measure for success but is replaced by efficiency. The authors list a number of misconceptions concerning on-line optimization, in particular of ethylene plants. These are :

(1.) minimum energy does not insure maximum profit ;

(2.) optimizing distinct areas of the plant independently and then combining the results is *not equivalent* to optimizing the total plant ;

(3.) the best feed composition for maximum profit is not necessarily the best ethylene producing feed stock ;

(4.) false economic values lead to incorrect optimum conditions ;

(5.) minimum impurity in the recycle streams is not always good.

Funk *et al.* [1984] create a detailed model of an ethylene plant (furnaces, TLE (quenching heat exchanger on exit of furnaces), compressor, cold box, distillation columns) and then optimize different combinations :

- ◇ the furnaces only ;
- ◇ the fractionators only ;
- ◇ the furnaces and the fractionators combined ;
- ◇ the furnaces and the fractionators and the feed combined.

These four combinations were run for three different objective functions :

- ◇ maximize profit ;
- ◇ minimize production cost ;
- ◇ minimize production cost per pound of olefin production.

Funk *et al.* [1984] made the following conclusions using the maximum profit objective function :

- ◇ as more sections of the plant are optimized, profits increased ;
- ◇ optimization of the fractionating area alone generated much less profit than any of the other areas by themselves ;
- ◇ as more sections were optimized, the ratio of ethylene to propylene production increased. The optimization shifts the propylene production to ethylene production.
- ◇ The solution of the fractionator area alone using the maximum profit objective function does not give a minimum energy answer. The optimizer can make profits by recycling ethane and propane to produce ethylene and propylene.

Using the minimum cost objective function, the authors conclude that :

- ◇ as more sections of the plant were included, cost decreased.

- ◇ The ratio of ethylene to propylene does in most cases increase.
- ◇ The deethanizer seemed to be the key column, with greatest variation in optimal separation values.

If the objective function is the minimization of the cost of production per pound of olefin, the authors pointed out that :

- ◇ also in this case, the gains obtained by on-line optimization increased as the scope of the optimization increased.
- ◇ Optimizer moves can be in opposite directions depending upon whether the whole plant or only the fractionator section is considered.
- ◇ The benefits from optimizing furnaces and fractionator section separately are much lower than the optimization of furnaces and fractionators combined.
- ◇ Optimal operation is sensitive to byproduct prices. Therefore these prices have to be determined carefully.

An important result is that partial optimization without coordination, or optimization of only parts of the plant can be substantially less interesting than a coordinated or global optimization. Optimization of a part of the plant can actually lose money on the global plant.

Funk *et al.* [1984] also stress that optimization and control are two distinct functions in the operation of a plant. This point supports the hierarchy concept that is described in section 2.2.

Cronin *et al.* [1988] present the Advanced Plant Management System, a trademark of Pyrotec N.V. This system contains a ethylene plant model, data reconciliation capability, plant and feedstock optimization. Unfortunately, Cronin *et al.* [1988] limit their publication to an illustrative example of the capabilities of the package. Specifics about the data reconciliation algorithm and/or methodology are not given, the optimization routines are not specified,

and model fitting and updating is not discussed.

2.3.6. Optimization of Olefins Separation Sections or Units

Olefins are produced in mixtures, that are separated after cracking. These separations involve several distillation columns with high reflux ratios and relatively small relative volatilities. In a first step, the reflux ratio of a column can be optimized, trading off recovery values with energy cost. A publication by Martin *et al.* [1981] deals with that application. Distillation of such light components as ethylene, ethane, propylene etc., requires refrigeration for condensation of the light end. Kaya [1983] considers plant chillers and their optimization. Darby and LeJune [1988] discuss plant refrigeration as well, and couple it with distillation column optimization as discussed by Martin *et al.* [1981].

Martin *et al.* [1981] discuss the on-line optimization of distillation energy. They conceive of this optimization as a tradeoff between energy costs and product recovery value. In their publication, they consider a typical olefins plant separation section with a deethanizer, a depropanizer, a debutanizer, a C_2 splitter and a C_3 splitter. Daily savings between \$ 100 and \$ 1000 are listed. For a long time, distillation columns were run at maximum reflux, but higher energy costs provide incentive to lower reflux. The cost for energy (reboiling and condensation) is approximately linear with the column reflux. Product recovery value also increases with the reflux, but with a decreasing slope. In some cases, the increase in the product recovery value will not compensate for the increasing energy costs at a reflux below the maximum reflux. A column is a likely candidate for on-line reflux optimization if the total variable cost curve is more convex than its product recovery value curve. Ethane ethylene splitters are key columns in olefin separation plants. C_2 splitters exhibit very high reflux ratios,

since the relative volatility is low. Condensation demands refrigeration, and is therefore expensive. The optimum reflux can change significantly with the target ethane concentration in the ethylene flow. Savings of more than $ 150 per day are reported by Martin *et al.* [1981]. The reflux optimization of a C_3 splitter is also discussed by Martin *et al.* [1981]. Chapter 4 (section 4.1) covers this example in depth, since it is the base for one of the case studies presented later in this dissertation. The authors characterize the optimization scheme they propose as easy to use, and comparable to a computer control loop. The authors do not specify what models are used in the optimization, and whether or not they are updated. It is also not mentioned how the value of the intermediate products (e.g. propane) is determined. These intermediate product values are very important, and determine directly the position of the optimal reflux. More results on the on-line optimization of distillation columns are presented by Sourander and Gros [1986]. Their presentation is reviewed in the next section. In this section, we will focus more on the olefin plant separations. As mentioned above, refrigeration has to be provided for the condensation of light products.

Kaya *et al.* [1983] report on the on-line optimization of an industrial plant chiller system for minimization of energy consumption. This application report shows a good example of loop control that is simple and minimizes the need for optimization by supervisory control. These authors consider a refrigeration plant. A schematic overview of the plant is given in figure 2.3.2. The global optimization layer realizes a trade off between conflicting objectives. These objectives are the cost of energy used in every part of the plant. The sum of these costs has to be minimized, but this optimization is made easier and more effective by loop control that performs some local optimization and reduces the need for global optimization. The chilled water temperature is maximized while keeping the users provided of chilled water, by keeping the most open valve to

a chilled water user open at about 90%. The 10% percent control margine is left for operational safety and transient load demands. At the same time the chilled water temperature is increased, keeping the user demands satisfied. Such a control policy minimizes the need for optimization. This approach is very similar to applications of valve position controllers as suggested by Shinskey [1978]. The condenser water temperature is optimized similarly by keeping the condenser water as cool as possible without wasting energy supplied to the cooling tower fans. The fan speed is controlled. Hot water temperature is kept as low as possible, and is traded off with its flow rate (pumping energy). There are several parallel pumps and chillers in the plant. Local load allocation optimization is applied. The global optimization then trades off these local optimizations. The authors mentioned robustness as an important advantage of this approach.

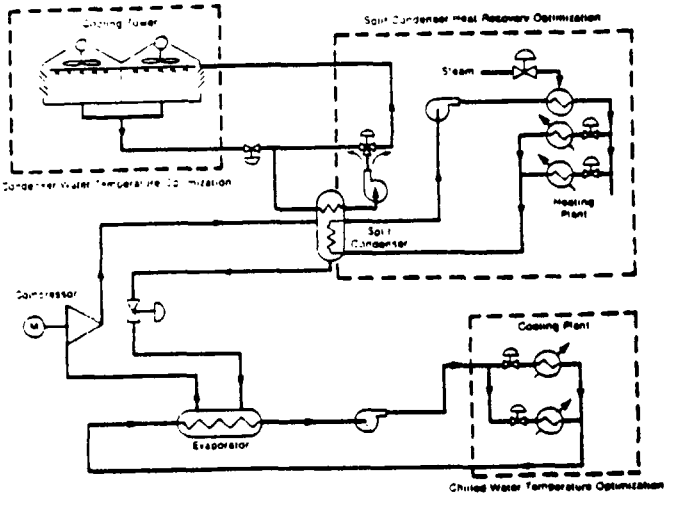

Fig.2.3.2.. System considered by Kaya et al. [1983]

Darby and LeJune [1988] present the optimizing control for an olefins plant refrigeration area. Refrigeration, using propylene and ethylene as refrigerants, is necessary in the distillation section of an olefin production plant to realize condensing in the separation equipment. The refrigeration machines are

usually multistage isenthalpic expansion systems. Compression is provided by centrifugal units, driven by steam. Darby and Lejune [1988] describe a propylene refrigeration system and also describe the loop control. The controllers on the lowest levels are PID controllers, and keep expansion drum levels, compressor suction pressure and reflux flow rate at setpoint. Darby and LeJune [1988] optimize the refrigeration allocation on-line. They divide the optimization into two parts :

- ◇ calculation of refrigeration coordination prices ;
- ◇ calculation of optimal product recovery.

Darby and LeJune [1988] use multiple local optimization, instead of one single global optimization. Every distillation tower in the separation section that is served by the refrigeration machine, is optimized separately, by minimizing the variable operation cost. The reflux flow rate is the optimization variable. For the refrigeration system, the compressor suction and discharge pressures, and the compressor inlet vane positions (if applicable) are optimization variables. A coordinator module combines and directs the local optimization. They list the following advantages local optimizers have to a global optimization :

- ◇ more frequent execution ;
- ◇ model accuracy ;
- ◇ on-line adaptation ;
- ◇ maintenance ;
- ◇ robustness.

The authors do not explain or illustrate the claimed advantages. Easier maintenance and robustness seem justifiable advantages. Model accuracy and on-line optimization do not seem related to decentralization of the optimization. Model parameter estimation should always be decentralized (see section 3.3). More frequent execution of the subset optimization problem is generally possi-

ble in a decentralized optimization problem, because the slowest dynamics of the plant section that is optimized influences the frequency at which this plant section can be optimized. That is certainly so if no check of significant change is made on the setpoints. The optimizer execution frequency will be subject of discussion in Chapter 5. The coordinator will not be able to be executed at a much higher frequency than a global optimizer can be run. This structure is somewhat comparable to a cascade control structure. The coordinator calculates the prices of the intermediate products, while the local optimizers minimize the local cost. The prices of the intermediate streams are determined based on the final product pricing and downstream processing costs. This task is by no means trivial. Darby and Lejune [1988] conclude that the optimization scheme they present has shown to be effective in terms of increased product recovery from the product mixture as decreased energy consumption in various applications. No details are given on these applications. They also stress that on-line implementation provides a direct and automatic link between business objectives and plant operation. Profit gain or investment payback time are not discussed. Optimization and estimation algorithms are not specified, and data reconciliation or sensitivity analysis are not mentioned.

2.3.7. Optimization of Refinery Sections

Latour [1980] reports on closed loop optimization of petroleum refining processes. According to Latour [1980], the successful optimization systems have a substantial impact on the gap between the control room operations, higher level management objectives, planning and environmental restrictions. This conclusion conforms with section 2.2.2.1 on the importance of the plant structure on the control hierarchy. Much attention has to be focused on the information interface between the optimizing computer, the operators, and the reaction of

engineering managers to market information. Latour [1980] lists following requirements for successful application of on-line optimization in oil refineries :

- ◇ selection of the proper independent variables as optimization variables ;
- ◇ formulation of the model accuracy requirements ;
- ◇ formulation of the business objectives ;
- ◇ selection of a proper optimization algorithm (guarantee a feasible solution, handle partial equipment failures) ;
- ◇ specify sensor inputs and manual inputs ;
- ◇ define a human interface for objectives and other problem specifications ;
- ◇ specify the interface with regulatory controls, timing, etc ;
- ◇ specify computer hardware and software ;
- ◇ consider procedures for maintenance (i.e. model plant correspondence).

Latour [1980] stresses that computer optimization competes for investment funds with other methods of process improvements. The benefits of on-line optimization can also be achieved by alternate methods, e.g. by changes in the process flow sheet. An important difference is the fact that an on-line optimizer is much more flexible than, e.g. adding a heat exchanger. This flexibility provides robustness and lowers the investment risk. Applications of on-line optimization are typically the best investment alternatives for small incremental improvements for highly constrained and complex processes subject to varying operating conditions.

Latour [1980] describes in some more detail specific objectives and constraints for various units of a refinery (local optimization). He introduces three fundamental areas of on-line optimization. The first area is *the optimization of the material balance (1.)*. This means the determination of the optimal routing of material through the plant. Flows are not often independent optimization variables, but allocation of load between parallel units, for instance, is frequently

an independent optimization problem. A second area concerns *the energy integration (2.)*. Optimal energy use is very important since often energy is an important or even the most important factor in the cost of a final product. *Optimization of reactor conditions (3.)* is a third fundamental optimization area. Often, it is said that reactor condition optimization (maximization of product yield) is the most important area, and that profits obtained in that area generally exceed savings that can be achieved in the others. Therefore, the third area would be the only important area in an on-line optimization problem that involves one or more reactors. However, it is also true that reactor optimization by itself can be expensive (utilities, wasted heat, etc.), and therefore it is not always affordable to neglect the two first fundamental optimization areas.

Sourander and Gros [1986] discuss the control and on-line optimization of integrated fractionation processes in a petroleum refinery. First, the authors describe the most important characteristics of the Naantali refinery (Neste Technology, Finland). The refinery has two atmospheric crude towers, and a vacuum unit to process the bottoms of the atmospheric columns. Heavy atmospheric gasoil and light and heavy vacuum gasoil are the main feed to the catalytic cracker. Light and medium atmospheric gasoil as well as similar fractions from the cat cracker are hydrotreated to diesel and light gasoil. Hydrotreated and reformed heavy naphta forms a base for the gasoline pool, and light naphta is stabilized in a debutanizer and further treated in a naphta splitter. A complex heat exchanger network is provided for integrated heat recovery. The study presented by Sourander and Gros [1986] is focused on four columns : both atmospheric crude columns, the vacuum unit and the naptha splitter. The installation of an on-line optimizer in this fractionation section was carried out within the frame of a complete modernization of the refinery. A substantial amount of new instrumentation was installed. Control loops were reviewed and

reorganized where necessary. The interaction between loop control and on-line optimization was taken into account while reorganizing the loop control for the fractionation section. The optimization of the fractionation section is only a part of the global refinery optimizer that was installed. However, Sourander and Gros [1986] limit their publication to the fractionator unit. The optimization of this unit is further split into an optimization of the pumparound duty (higher pump duty gives better separation but raises costs), optimization of the column feed temperature, vacuum unit optimization and light ends fractionation minimization. Optimization of energy recovery versus yields coordinates the local (high frequency) optimizations. Sourander and Gros [1986] report a high availability of the system, and a more constant product quality. They mention a payback time of approximately one year.

Sofer *et al.* [1988] discuss the retrofitting of a hydrocracker main fractionator to computer optimization at the Kerr Mc Gee Refinery in Wynnewood, Oklahoma. A hydrocracker fractionator with a design feed flow of 91000 lb/h was switched from analog control to distributed advanced control with setpoints set on-line by an optimizer. The fractionator includes a main column with a side draw for heavy gasoline above the feed tray, a reboiler, a reflux condenser, a reflux accumulator and a gasoil stripper with a separate reboiler. In a first phase, the fractionator is successfully simulated and in that way more information is gathered on unmeasured variables, such as the tray loadings. Simulation studies point out problems with the original control scheme that are not of interest here. A new control scheme is proposed and simulation test runs are carried out. Specifics of this new scheme are not important with respect to on-line optimization, but it is important to note that no optimization scheme can solve the problems that are experienced with a loop control scheme. The original control scheme in this example, causes trays to run dry. Because of that, the scheme

cannot maintain controlled variables at their setpoint values. Implementation of an on-line optimizer does not make sense under such conditions. The new control scheme for the hydrocracker fractionator does maintain setpoint values. The new scheme also guarantees a quieter operation with better compensation for unmeasured disturbances such as ambient temperature. It also has to be noted that some instrumentation (e.g. on-stream analysis) was added. The report on the optimization part of the work is rather concise. Model updating is not mentioned, nor is data reconciliation, and probably neither is implemented. The simulator model is optimized off-line for several operating conditions, and a polynomial curve is fitted through the optimal result. This parameterized optimum is implemented. The optimization variable is the steady state main column reboiler exit temperature. The optimization objective is not specified as financial profit. This example is a case in which the model was accurate enough to make on-line updating redundant. The frequency at which the optimal reboiler exit temperature is updated is not given. Sofer [1988] states that the retrofit resulted in $ 1945 savings a day. Improved control is responsible for approximately 70% of the savings, while the other 30% is saved by the optimization. Investment cost is not given, nor is the payback period. It has to be pointed out that the $ 600 per day saved by the optimizer, can only be saved after the installation of the new control system with more instrumentation than the previous scheme.

Takama *et al.* [1980] present a method for the optimal water allocation in a petroleum refinery. The study combines the optimization of water treating and water polluting systems into a global optimization. The authors state that this combined optimization of water using and water treating systems is much more effective than the optimization of waste water treating systems only. The optimization of water treatment systems only, aiming for a zero discharge result,

is the standard case in current practice, according to Takama *et al.* [1980].

The authors formulate the optimal water allocation problem in a design and an operation stage. They limit themselves to the operation problem, i.e. the repeated optimization of the water allocation in an existing refinery. The optimization only concerns the steady state water allocation. Very simple steady state models are used for the water polluters as well as the water users. It is assumed that every polluter k produces a polluting component l at a rate f_{kl}. Furthermore, it is assumed that every treatment facility i removes a fixed fraction r_{il} of that pollutant l per pass for a cost that is function of r_{il}. The total combined cost for waste water treatment and fresh water purchase is minimized. Environmental regulations constrain the outlet composition. For every water treatment and water polluting unit, the water supply and composition has to be maintained within certain values to ensure proper functioning of the unit. A last set of constraints requires every flow to be positive. Depending upon how extensive the piping network connecting the polluting units with the treating units is, several streams can become zero.

Takama *et al.* [1980] present an example optimization using realistic plant data. A penalty function algorithm is used to solve the constrained optimization. The results of that illustration are not compared with more complex and accurate simulations, nor with plant results. The authors do not indicate how to obtain the unit characteristics f_{kl} and r_{il}. Execution frequency, data reconciliation, sensitivity analysis are not discussed. Economic feasibility is not mentioned.

Martin *et al.* [1988] present a study on the optimization of the reflux systems of a crude unit. A crude tower has multiple reflux streams (pumparounds) and they can be adjusted to influence the column heat exchange. Choosing these reflux stream rates is not an obvious problem, and it lends itself to on-line op-

timization since process behavior and economic models are available. Martin *et al.* [1988] split the effects of determining a crude unit reflux streams in two parts : the product yield effects and the heat recovery effects. They present models for both types of effects. It is important to note that these authors also explicitly stress the importance of good regulatory and constraint control. Meeting process constraints continuously has to be guaranteed. Simulation studies show that, compared to previous operation, the on-line optimization results in increased pumparound duty, with increased yields of diesel and kerosene. When the differential product price outweighs fuel costs, the optimum is constrained by the minimal pumparound operation. If the fuel cost outweighs product prices difference, the optimum is constrained by maximal pumparound. For certain combinations of product prices and fuel cost, there is a region with a solution that is not constrained by pumparound specifications. However, in most cases the solution lies on a constraint.

2.3.8. Optimization of Paper and Pulp Mills

Blevins *et al.* [1980] present a software package for energy monitoring and energy system optimization in the paper and pulp industry. Blevins *et al.* [1980] suggest that there are three basic approaches to on-line optimization : advisory, semi-automatic and automatic. In the first approach the computer merely suggests new setpoints to the operator. The operator has to feed these setpoints manually into the process control computer. In the semi-automatic application only the approval of the operator is necessary for the implementation of the optimization results. The automatic mode implements new results automatically without interacting with the operator. Blevins *et al.* [1980] prefer the automatic mode because it is best in keeping the setpoints near optimum. They developed two software packages for energy management. A first package is

directed towards design of energy balances and performing energy calculations. The second package permits the implementation of the plant on-line optimization, specifically load allocation on a parallel utility. They provide models for the simulation of various units of a pulp and paper plant. They also provide optimization modules, and objectives.

Coombes *et al.* [1983] describe the application of optimization techniques to an integrated Kraft paper mill to allocate material and energy flows throughout the mill in an optimal way. This technique, referred to by the authors as the Setpoint Control System, fills the gap between process loop control with decision time scales on the order of magnitude of an hour or less, and corporate scheduling, with time scales of the order of magnitude of weeks or longer. Each section of the mill is modeled in a level of detail that depends upon the expected variable cost the unit represents. Model parameters are updated using a recursive least squares technique. However, it has to be noted that the recursive technique is run from a history file. Therefore the tracking is off-line. A piecewise linearized model, linearized around the current operating condition is used by the optimizer. The objective is linear, as well as the constraints. The objective function includes most relevant cost terms. A linear optimization algorithm is applied. The complete system has sixty seven optimization variables with forty six equality and twenty two inequality constraints. Some simulation results are presented. A first set is simulated under the assumption that there is no market production limit. A second set is simulated under the assumption that there is a market production limit. The simulation studies show increases in profit of about three to six percent, starting from realistic operations. Although the authors indicate the potential for savings, and the effectiveness of their setpoint control system, an actual application in the Kraft paper mill is not reported. There is also no specification of the model parame-

ter estimation algorithm. Data reconciliation and sensitivity analysis (here LP : postoptimality analysis) are not mentioned.

Bergh [1984] proposes, with less detail, a similar distributed system. This author does not mention any algorithms, but spends more attention on the interface with the user. Also Wen [1983] presents a similar technique but gives more support to the use of dynamic models.

2.3.9. Optimization of a Large Scale Ammonia Process

Tsang and Meixell [1988] discuss the on-line optimization of a large scale ammonia plant in real time. The plant that is the subject of their study is located at the Agricultural Chemical Complex of Esso Chemical Alberta in Redwater, Alberta, Canada. This plant has been subject to extensive heat integration design, and therefore a high number of interactions between the production, the energy consumption and the setpoints exists. Central in this publication is the modeling effort that goes together with the implementation of a real time optimizer. The model that is developed is based on first principles, and is therefore very extensive (5500 equations). A equation management and solving package that is developed by Exxon Chemicals is used to solve the model. The authors do not indicate how the parameters in the model have been obtained and how they are updated. The optimization is still run off-line, but an on-line application is planned. Details on the actual real time optimization are not specified.

2.3.10. Optimization of Bench Scale Units and Pilot Plants

Lee and Lee [1983] report on the on-line optimization of a bench scale fixed bed reactor in which the partial oxidation of n-butane to maleic anhydride is carried out over a vanadium phosphorous catalyst. The reaction is strongly

exothermic. The objective of the optimization is the maximization of profit. An important control variable is the maximum bed temperature. This temperature has to be monitored carefully, since under certain conditions runaway can occur. The following functions are carried out by the control scheme :

(1.) adaptive control of a bed temperature near or behind the temperature peak in the catalyst bed, using the coolant temperature setpoint as a manipulated variable ;

(2.) on-line identification of a dynamic reactor model with the catalyst bed temperature measurement and the residence time as input, and as output the yield of maleic anhydride ;

(3.) steady state optimization using system parameters supplied the by on-line identifier described in *(2.)*.

A dynamic model is identified as suggested by Bamberger and Isermann [1978]. If disturbances are relatively fast such that the plant is not often at steady state, this approach has to be used. However, a dynamic model has to be constructed. In this example, a black box model is used. A recursive parameter identification method is used by Lee and Lee [1983]. The dynamic model is a second order Hammerstein model. This is given by :

$$\begin{aligned} D(q^{-1})y(k) =& E_v(q^{-1})v(k-d_v) + E_r(q^{-1})r(k-d_r) + \\ & E_{v^2}(q^{-1})v(k-d_v)^2 + E_{r^2}(q^{-1})r(k-d_r)^2 + \\ & E_{vr}(q^{-1})v(k-d_v)r(k-d_r) + D_I. \end{aligned} \tag{2.3.4}$$

in which :

$$D(q^{-1})y(k) = y(k) + d^1 y(k-1) + \ldots \tag{2.3.5}$$

$$E_v(q^{-1})v(k) = e_v^0 v(k) + e_v^1 v(k) + \ldots \tag{2.3.6}$$

$$E_{v^2}(q^{-1})v(k)^2 = e_{v^2}^0 v^2(k) + e_{v^2}^1 v^2(k-1) + \ldots \tag{2.3.7}$$

...

and D_I is a bias term. r refers to the temperature measurement deviation variable, v is the reactor residence time deviation variable, and y is the maleic anhydride yield deviation variable. A steady state version of (2.3.1) is used by the on-line optimizer, given by :

$$y_\infty = G_v v_\infty + G_r r_\infty + G_{v^2} v_\infty^2 + G_{r^2} r_\infty^2 + G_{vr} v_\infty r_\infty + G_I; \tag{2.3.8}$$

with :

$$G_v = \frac{E_v(1)}{D(1)}; \tag{2.3.9}$$

$$G_I = \frac{D_I}{D(1)}. \tag{2.3.10}$$

The dynamic identifier is activated with a PRBS signal.

The steady state optimizer maximizes the net profit per unit time of the production of maleic anhydride from n-butane. The temperature of the bed has a maximum constraint, to prevent runaway. The constrained optimization is carried out using a quadratic penalty function method :

$$\max\left(\text{objective}\right) = \max\left(\text{net profit} - \beta \cdot \text{penalty}\right) \tag{2.3.11}$$

with :

$$\text{net profit} = \frac{\left(\alpha(y_\infty + y_{\text{ref}}) - 100\right)}{v_\infty + \tau_{\text{ref}}}, \tag{2.3.12}$$

and

$$\text{penalty} = r_\infty + T_{ref} - T_{crit}. \tag{2.3.13}$$

In this equation, y_∞, v_∞ and r_∞ are the steady state deviation variables for maleic anhydride yield, residence time and the bed temperature. The reference values are added to these deviation variables. y_∞ is replaced by the steady state model given by equation 2.3.5. The optimization variables are the residence time in the reactor and the maleic anhydride yield setpoint. The adaptive control system brings the reactor to the new setpoint by manipulation of the coolant

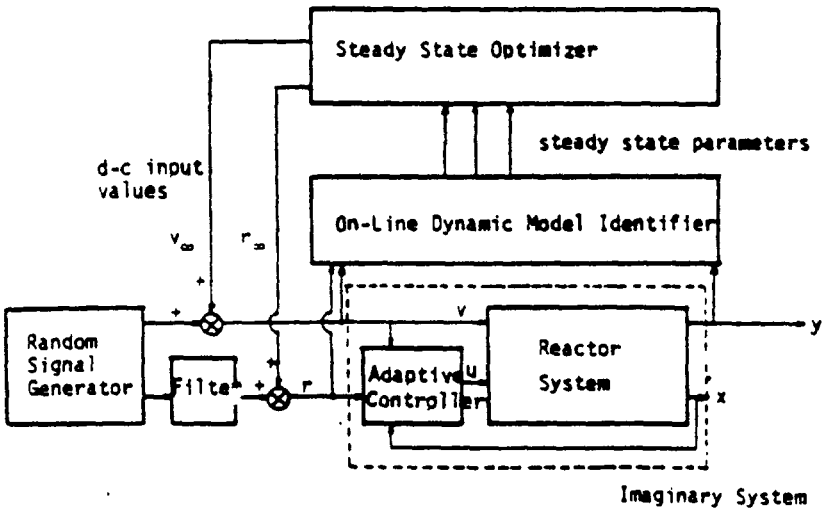

Fig.2.3.3. On-Line Optimizing Control Scheme for a Bench Scale n-C_4 Partial Oxidation Fixed Bed Reactor (from Lee and Lee [1983]).

temperature. The optimizer determines setpoints for air and n-butane mass flow directly every 100 minutes. Figure 2.3.3 gives an overview of the control scheme.

The technique that is briefly described above, is implemented on a bench scale reactor. A sampling time of one minute is used, and can not be decreased because the slow response of the oxygen analyzer. Lee and Lee [1983] find a good correspondence between the predicted profit and the actual profit. The true optimum of the reactor is not known. The cooperation between adaptive control and on-line optimization using the same identification module, seems efficient and effective. The authors expect further improvement if the optimization is executed more frequently than once every 100 minutes. However, they also fear stability problems because the high demands that are put upon the model identifier since an important plant model mismatch exists close to runaway conditions. A more optimal result would probably bring the reactor closer to runaway conditions, and therefore the plant model mismatch would be relatively larger under such conditions, with all the risks that go together with operation

close to runaway conditions.

The study presented by Lee and Lee [1983] does not mention implementation of data reconciliation or sensitivity analysis, nor does it estimate the potential economic benefit that can be obtained from implementing their technique.

The on-line optimization of a bench scale liquid extraction process, separating acetic acid from amyl alcohol using water as solvent, is described by Jackson and Agnew [1980]. Computer control can maintain the mass transfer interface level, the feed and solvent flow rates and the stirrer speed. The last three setpoints are optimization variables. Feed and solvent flow rates, stirrer speed, feed, raffinate and extract compositions, feed temperature and interface position are measured. A very interesting aspect of this study is that four steady state models are used, with different degree of complexity. The first model is a linear empirical model. The second model is a nonlinear empirical model. The third model is a plug flow model. The fourth model superimposes axial diffusion on the plug flow. Details of the model are not given here, but can be found in the publication of Jackson and Agnew [1980]. It has to be noted that all four models are fit to the same set of experimental data. The F-values of the fit of the first two models (the empirical models) indicated that these models do not fit the data well compared to the 95% confidence limit. The models based on physical principles do fit the data well. Based upon the same test, the quality of their fit is comparable, although the fourth model fits the data marginally better. Jackson and Agnew [1980] used four different optimization objectives. The objectives are chosen such that the performance of the optimizer at constrained and unconstrained optima could be studied, and such that the objectives are functions of model responses and hence the model type. From an energy consumption or operation cost point of view, these functions are essen-

tially arbitrary. Minimum and maximum values of the optimization variables constrain the optimization, as well as column flooding. A gradient projection algorithm is used in the optimization. From a numerical point of view, only one of the objectives causes a problem, being "a narrow cliff" along a constraint. The empirical models give optimization results that are biased compared to the more rigorous models. In some cases, the optimization did not converge at all if empirical models were used. The rigorous models predict optima that are not significantly different from the "true optima" as obtained using evolutionary operation (direct search using the plant instead of a model). The physical models performed significantly better in tracking a moving optimum than the empirical models.

An important conclusion of the study is that the on-line optimization scheme can fail because of poor plant behavior predictions. Simple models should only be used with great care. It also seems that the results of a statistical test (such as the F-test) can be good indicators for the possibility of poor model predictions. The frequency of execution is not discussed.

D'Hulster *et al.* [1980] report on an optimizing control system for a pilot scale thermal cracking reactor. The control system is realized in such a way that three levels of the control hierarchy are run by two different computers. A PDP 8E computer serves on-line functions. It logs data, runs A/D conversions, integrates gas chromatograph data and does the loop control (setpoint enforcement) using PI algorithms. On the basic loop control level, only temperatures are controlled. Composition measurements are too slow to be used at this level. A PDP 15, a more powerful machine, supports the higher control level. On the intermediate control level a multivariable inferential control algorithm is used. This inferential control is used to deduce the behavior of the plant in between composition measurement logging, which is approximately every 15 minutes.

The steady state setpoints are results of an optimization. The highest level estimates parameter values of the model used at the intermediate level. The paper reports experiments with black box models and models based upon the physical principles. Both types give satisfactory results. No details are given concerning the numerical optimization or estimation algorithms.

2.3.11. Conclusions

In all publications the importance of the control hierarchy is made clear. The communications between the layers are important. In all industrial applications reported, a new, upgraded or advanced loop control (= setpoint enforcement) is discussed at the same time. This illustrates how important the loop control is, and *(1.)* that on-line optimization puts high requirements on the loop control system and *(2.)* that on-line optimization does not make sense without good setpoint enforcement. Particularly important is the need for a control system that can maintain constraints.

Furthermore, several publications point out that the global plant optimization has to be split into smaller subproblems : a distributed optimization system. The coordination problem is a difficult one to solve, and most of the suggested solutions (i.e. determination of intermediate prices and values) are not entirely satisfactorily. Often the methods require some iteration. Several authors prefer global optimization systems to distributed optimization because of coordination problems. On the other hand, a distributed optimization system is much easier to maintain, reliable, and robust. The optimization problem is easier to formulate and understand. A distributed optimization system is not a combination of partial optimization systems. Partial optimization is a much poorer solution than global or distributed optimization, since there is no coordination between the parts. Combining the results of partial optimization *without*

coordination is in general wrong and can even decrease overall performance.

The arguments for using dynamic models instead of steady state models usually refer to problems with collecting steady state data. On the other hand, the arguments for using steady state models often mention to problems optimizing and processing the dynamic models, as well as to the problems identifying them and building them. Some authors propose a trade off identifying dynamic models, while optimizing a steady state version of that model. In any case, for an optimization problem with a large scope (several units or a plant) dynamic models are very difficult to build and identify. For that reason, we limit ourselves to steady state models in this work.

Another important conclusion concerns the importance of the model quality for optimizer *convergence* and performance. Not only absolute values but also gradient predictions have to be accurate to obtain the correct estimate of the optimum setpoint. The majority of the publications on real-time optimization applications stress the crucial importance of good modeling.

None of the papers referenced, discuss the application of sensitivity analysis in on-line optimization. In the next chapter, it will be shown that sensitivity analysis can be extremely useful in on-line and off-line applications of setpoint optimization. Sensitivity analysis describes the status of the optimization solution. This analysis allows an evaluation of the importance of constraints and parameters, and permits one to estimate the influence of model identification accuracy on the optimizer performance.

Only Poje and Smart [1986] discuss the use of data reconciliation. They don't specify any algorithms, but only mention that data can be compared with each other based upon mass and energy balances. Plant data can always contain systematic errors, and it is very important to detect these. The potential gain that can be made by implementing an on-line optimizer can be lost if erroneous

data are used. Moreover, from fitting the model to wrong data an unexpected plant model mismatch can result. This mismatch can cause instability. Methods for data reconciliation are discussed in the next chapter and their use is illustrated in Chapter 5.

The reports that industry publishes are particularly vague. Algorithms are often not specified. Many successes are reported but it is often not clear what the base case or reference is. Nevertheless, it seems that profits from 2 to 10% are expected and/or achieved. When given, payback times are from some months to a year. Various publications note also that one of the additional advantages of the implementation of an on-line optimizer is the improved monitoring and better understanding of the process.

CHAPTER 3

THE CONCEPT OF AN ON-LINE OPTIMIZER

3.1. INTRODUCTION

This chapter presents the structure and concept of an on-line optimizer in detail. The algorithms used are presented and characterized. The flow and storage of information within the on-line optimizer is described, as well as how the information is processed. Figure 3.1.1 gives a schematic overview of an on-line optimized plant. The plant includes the process units as well as the loop control. Therefore the setpoints are a channel of input to the controlled plant. The other input channel and the output channel are not necessarily input and output in the physical sense, but in the sense of the operation and the modeling. The on-line optimizer block is divided in four layers. The layers are characterized by the typical scope of the data manipulation that is realized there. Data manipulations in the top layer can be done on one measurement at a time. The scope of the data reconciliation and the gross error detection can vary from the order of magnitude of a unit to the complete plant. One model identification processes data from one unit, and finally the optimization routine considers the complete plant again. Since the scope of every layer in figure 3.1.1 are different, the data may have to be regrouped or spread over smaller modules while crossing the borders of a layer.

The plant inputs and outputs are collected and checked for saturation and compared to minimum and maximum realistic values. A module checks for

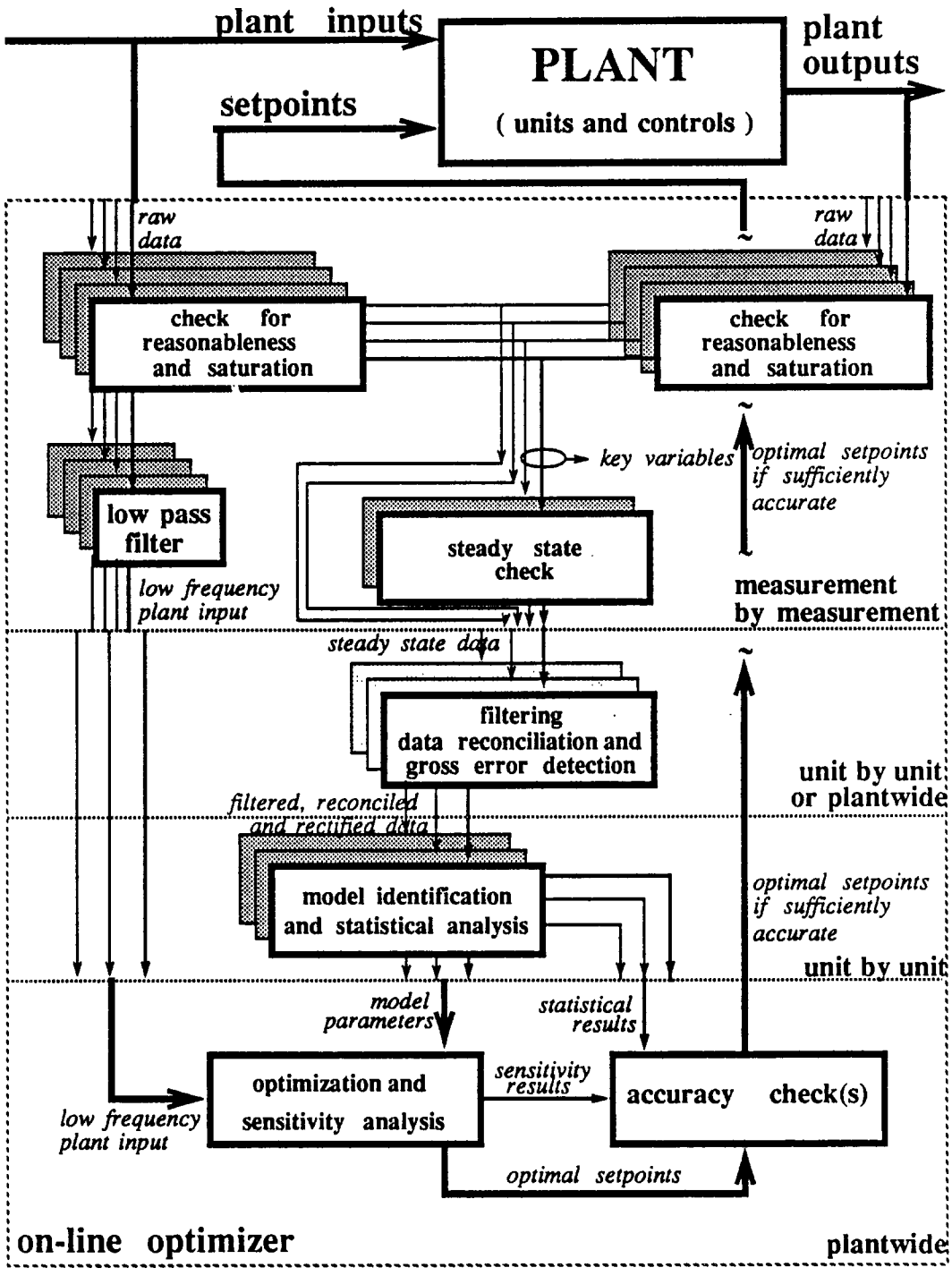

Fig.3.1.1. Schematic Overview of an On-Line Optimizer.

steady state using a number of key variables. The data reconciliation and gross error detection module checks steady state data for inconsistencies (within one steady state) and the model identification module estimates steady state model parameters and checks for outliers (inconsistencies among steady states). The optimization routine uses the updated model parameters, and the plant input to generate an array of optimal setpoints. These setpoints are implemented after the expected accuracy of the combination of model identification and optimization is estimated, and found sufficient.

Main storage of information is in the steady state data base which can be considered as a part of the model identification block(s). The low pass filter contains some past information of the various measurements, and the accuracy check may also contain some information on the conditions of previous accuracy checks.

Checking for reasonableness and saturation is straightforward. In general, if a measurement does not pass the test, it will be flagged as unreliable and the data reconciliation can incorporate this information during its manipulation. If a measurement does not pass the test repeatedly, then the instrument will have to be inspected unless another reason for the failure is known. The low pass filter in figure 3.1.1 is quite important, as will be made clear later. It may be useful to have different cut off frequencies for measurements corresponding to states with significantly different time constants.

This chapter discusses in detail the theory of the remaining manipulation in figure 3.1.1. First steady state detection is discussed. Then, data reconciliation and gross error identification are discussed, followed by model identification. Optimization follows, and finally the calculation and application of sensitivity analysis is discussed.

Some new approaches are presented in this chapter, other sections are se-

lected from the literature on statistics, optimization theory, data reconciliation, model estimation, etc. on the basis of applicability in on-line optimization.

3.2. STEADY STATE DETECTION

3.2.1. Introduction / Key Variables

The steady state detection block has to decide whether or not any part of the plant that has a model that can be identified separately from the rest of the plant (from here on referred to as a unit) is steady or not. Whether or not a model can be identified separately from the rest of the plant is a numerical question. A model is identifiable separately from the rest of the plant if the model parameters and the *calculated* variables that are used in the equations of that model are not used in any other model equations. The calculated variables have to be completely determinable from the unit model equations (i.e. the set of equations that contains the unit model is solvable). An example of a unit can be a single heat exchanger. A heat exchanger model with the heat transfer coefficient as its only parameter can be estimated separately from other units if the C_p values of the hot and the cold medium is known, or if these C_p values can be calculated without having to estimate other parameters. If a certain unit is steady, then the corresponding data can be used for steady state model identification. A unit can be a boiler, a distillation tower, a heat exchanger, a catcracker etc. Of course, if a dynamic parameter estimation technique is being used steady state detection will not be necessary. However, in this work it is assumed that steady state models are used only.

Steady state detection is a problem for which many *ad hoc* solutions are being applied, and too few fundamental answers are being proposed. Usually these practical solutions are tailored to the plant they are applied to, and they perform well. Good steady state detection is important, since data with significant departure from steady state can bias the parameter estimates and deteriorate the performance of an on-line optimizer. The use of unsteady state information

is a systematic error, which is often very difficult to identify as such by any other means than comparing the alleged steady state data with plant output versus time graphs.

Steady state detection is not a problem that is fundamentally discussed here. We limit ourselves in this section to the presentation of three methods that are being used in industrial practice at this time. After that, a statistically based test is given.

All methods that we consider have something in common. The tests focus on a number of key variables, which are a subset of all the measurements. The selection of these key variables is very important. In theory these variables have to be chosen so that, if any (observable) state in the system is not constant or steady, then at least one of the key variables is not steady either.

3.2.2. Practical Tests

These are some of the most common tests. Innumerable variations can be suggested, but the concepts are usually the same.

3.2.2.1. The Band Test

One of the methods that is frequently used considers the maximal variation of a variable over a certain length of time. If the signal can be squeezed into a band of preset width for a given length of time, then that variable is considered steady. There are two tuning variables per signal to this test : the width of the band and the length of the time interval (Contino [1987]).

3.2.2.2. Filter Based Tests

3.2.2.2.1. Low Pass vs. Very Low Pass

A test could conceivably compare the output of two low pass filters with

considerably different cut-off frequencies with each other. If the difference between the outputs of both is below a certain threshold for a certain length of time, then steady state is accepted. Here four tuning variables have to be determined. These are the two cut-off frequencies, the length of the time interval and the size of the accepted difference. From the concept of this test, one can expect that this test will perform better under noisy conditions than the previous test. A disadvantage is the double number of tuning variables. The type of filter could be simple : linear (average) or exponential.

3.2.2.2.2. Moving Average

The rate of change of a moving average can be checked. If this rate is smaller than a certain value, for a certain length of time, then steady state is accepted. Here there are three tuning variables : the length of the time interval, the length of the moving window and the critical rate of change of the moving average (Contino [1987]).

3.2.3. A Test Based upon the Wilcoxon Test

While the previous tests are mainly based upon filtering, this new approach uses a statistical test, working on the raw data. Consider two consecutive series of values of a key plant variable. If this variable is steady, then the values can be written as a constant with (high frequency) noise superimposed. The distribution of the superimposed noise is assumed to be the same in the first series and in the second. Therefore, if the tested variable is steady, then both series can be considered as samples of the same distribution. If there is a trend in the test variable, then both series cannot be considered as samples of the same distribution. A nonparametric order method such as the two sample Wilcoxon test (Wilcoxon, [1945]) allows for testing this hypothesis without any assumptions concerning the distribution of the noise (except that it has to be a

continuous distribution). Therefore the test would also work for non-gaussian noise.

Let the first series be denoted by x_1, x_2, ..., x_{n_1}, and the second series by y_1, y_2, ..., y_{n_2}. Order the combined sample x_1, ..., x_{n_1}, y_1, ..., y_{n_2} in increasing order of magnitude, and assign increasing ranks 1, 2, ..., $n_1 + n_2$ to the values after they have been ordered. W, the Wilcoxon test statistic, is the sum of all the ranks of the y series. It can be shown (Hogg and Tanis, [1983], p.394) that the average (or expected value) of W is given by :

$$\mu_W = \frac{n_2(n_1 + n_2 + 1)}{2}; \tag{3.2.1}$$

and the variance of W by :

$$Var(W) = \frac{n_1 n_2(n_1 + n_2 + 1)}{12}. \tag{3.2.2}$$

If n_1 and n_2 are sufficiently large, based upon the central limit theorem :

$$z = \frac{W - \frac{n_2(n_1+n_2+1)}{2}}{\sqrt{\frac{n_1 n_2(n_1+n_2+1)}{12}}} \tag{3.2.3}$$

is approximately normal with zero mean and unit variance ($N(0,1)$). z can be compared with the α confidence value of the standard normal distribution to obtain an α confidence test for steady state. Tanis and Hogg [1983] mention that if n_1 and n_2 are both larger than 7, then the assumption that z is standard normal is sufficiently accurate to be practical. This test is applied in the boiler case study in this work.

3.3. RANDOM AND GROSS ERROR IDENTIFICATION

3.3.1. Introduction

The input to an on-line optimizer consists mainly of plant data. The model parameter estimation extracts information from this stream of plant data, by fitting a model to these data. In practice, the data never fit the model *exactly*. Residuals are caused by model/plant mismatch and by errors in the measurements or data that are used. The errors in the data can be characterized as noise or as systematic or gross errors. The errors caused by random and systematic phenomena will result in data that not only will not fit the model equations, but also are in conflict with each other. Forcing models to fit certain "parameter-free" or "non-predictive" (Stanley [1982]) equations, such as mass and energy balances, can reduce the level of noise, and can allow for detection of systematic errors in data arrays (Mah *et al.*, [1976]). A systematic error in a data array corresponds to at least one measurement, and it is possible to identify the most probable cause using data reconciliation and gross error detection techniques. If certain measurements correspond to frequently detected gross errors or to gross errors with a suddenly increasing frequency, then it may be an indication that the measuring instrument has to be checked for malfunctioning or has to be recalibrated. On the other hand, no information can be extracted from these unreliable data. Therefore, it is important to detect gross errors.

In this section, the topic of data reconciliation and gross error identification is discussed with special attention to the needs and requirements in an on-line optimization scheme. First some observability and redundancy requirements are discussed. Several concepts of data reconciliation and gross error identification are listed next. Then some more details are given on preferred techniques.

3.3.2. Observability and Redundancy in Steady State Systems

Redundant measurements are needed in order to reconcile data, in other words, to check data with each other. Before we can define redundancy in plant measurements, observability has to be introduced. Observability is a concept that is best known in relation to dynamic systems that can be represented by linear ordinary differential equations. Quoting Ray [1981,p.250] : "(...) a system is observable if it is possible to determine all the state variables at some time t_0 based on the knowledge of the system output (...) and control (...) over some finite time interval". This concept was extended towards systems with partial differential equations and nonlinear systems. A similar concept of observability was defined for steady state systems by Stanley and Mah [1981a]. For steady state systems, observability concerns the possibility to determine whether or not the available measurements allow one to determine the current state of the system. The difference with the observability concept for dynamic systems is that the assumption of steady state is accepted, but on the other hand no "initial state" is known or relevant. Therefore, a system that is dynamically observable is not necessarily steady state observable and vice versa, although they will often go hand in hand. Steady state observability theory as presented here also differs from dynamic observability in the possibility to use steady state observability as a local system behavior characteristic, as well as in relation to single measurements and not only the complete system.

While avoiding to much detail, some of the observability theory that is developed by Stanley and Mah [1981a] will be presented here. These authors define a steady state system very generally as a triplet (S, h, V), with S a subset of $\mathbb{R}^n$ and h a vector function mapping $\bar{S}$ into $\mathbb{R}^l$. The system has n states and l measurements. V is a set that determines the noise. x will represent the states of the system, while z will represent the measurements. Let $x \epsilon S \subset \mathbb{R}^n$,

$z \epsilon \mathbb{R}^l$, $v \epsilon V$ and $z = h(x) + v$. If $v = 0$, the measurement is perfect. There is no noise or systematic error. Then, discrepancies between model and data can only be due to model plant mismatches. In practice v will not be equal to zero. In the simple case of white noise, v will be gaussian. The question that is posed here is the following : "Is it possible to reconstruct the ith state of the steady x_i system from the measurements h ? If so, then the ith state is *observable*. This is a local property if the equations determining S or h are nonlinear. The observability of a state is, in general, function of the complete state x of the nonlinear system. Observability is function of a linear system. For nonlinear systems it is function of the linearization of the system around the actual state.

Stanley and Mah [1981a] base the rigorous development of steady state observability theory on a definition of *unobservability*, which is easier to define. Let I be an index set, and let k represent an element in I. In a system (S, h, V), let $x_0 \epsilon \bar{S}$. The ith state x_i is *locally unobservable* if $\exists$ $\{x^k\}_{k=1}^{\infty}$ such that :

$$\begin{aligned} & x^k \rightarrow x_0 \text{ with } k \rightarrow \infty : \\ & \forall k \ : x^k \epsilon S \ ; \\ & \qquad h(x^k) = h(x_0); \\ & \qquad \delta x_i^k = x_i^k - x_{0i} \neq 0. \end{aligned} \tag{3.3.1}$$

In other words, there exists a sequence of states converging to the steady state under consideration within the set of "possible" or "feasible" states S that all give the same measurement values. The fact that there is a *converging* sequence with that property means that there are states *arbitrarily close to* ("next to") the actual steady state x_0 that result in the same measurements as x_0. That means that, in an arbitrarily close neighborhood around the unobservable state x_i, there is no unique solution to the determination of the state from the measurements.

The following definitions are based on the *unobservability* concept introduced above. If $x_0 \epsilon \bar{S}$ in a system (S, h, V), then x_i is *locally observable* at x_0 if x_i is not unobservable at x_0. Definitions for global observability and global unobservability on a subset of $\bar{S}$ are introduced by Stanley and Mah [1981a]. A state is globally observable or unobservable on a set $S_1 \subset \bar{S}$ if it is locally observable or locally unobservable in every point (state) of that set S_1. The system (S, h, V) is *calculable* on $S_1 \subset \bar{S}$ if h is one-on-one on S_1. This of course implies global observability of all states in S_1. Calculability of a system is a very useful characteristic, because the inverse of h maps measurements on the state space subset S_1.

Several observability classification theorems are formulated (with proof) by Stanley and Mah [1981a] for a rather general "standard form" of S and h. First and second order sufficient conditions for observability and calculability are introduced and shown to hold for the standard steady state system. These theorems are certainly of importance in the frame of steady state estimation, as illustrated by some of the theorems presented by Stanley and Mah [1981a]. In recent publications, rules on classification of variables and measurements in process networks are presented by Stanley and Mah [1981b], Romagnoli and Stephanopoulos [1980]. More specified rules have been worked out, e.g. on multicomponent networks by Kretsovalis and Mah [1987] and Vaclavek and Loucka [1976]. In this section however, the description of observability was only introduced as a base to define redundancy in a rigorous way.

Redundancy is very closely related to observability. A measurement is redundant if the observability of any of the states does not change if the measurement is removed. Definitions of local and global redundancy of a measurement are formulated the same way depending upon the type of observability that is used in the definition. Redundancy of a measurement adds robustness to

information extraction (model identification). If a measurement fails, there is no immediate loss in quantity of information (but possibly in quality). Redundancy can reduce the effects of noise, and allows data arrays to be checked for internal consistency.

Lack of redundancy is often caused by the structure of the network. Consider for instance the simple network that is presented in figure 3.3.1. This network counts 2 nodes. Stream 1 and stream 2 join in node 1 and form stream 3. Stream 3 enters node 2, and stream 4 leaves node 2. If all streams are measured, then it is still impossible to distinguish between a bias in stream 1, a bias in stream 2 or a leak in node 1, if node 1 is the only node that does not balance. One should not attempt to reconcile stream 1 and stream 2 separately, nor should one attempt to make a gross error identification scheme distinguish between a bias in stream 1, a bias in stream 2 or a leak in node 1. Redundancy is usually studied for networks with only one gross error. As soon as more than one gross error is present, the possibility for detection of both errors correctly becomes function of the relative position of both errors. An example is given in section 3.3.5.3.

Note that the concepts introduced in observability theory are not a function of any specification of V, the process noise space. The type and volume of the noise that is superimposed on the ideal data does not influence the observability of a system. Weber and Brosilow [1972] suggest ways to select measurements to avoid the problem that the results of manipulations with these variables are sensitive to noise and modeling errors.

The observability of states is only necessary with respect to data reconciliation. The optimization module of the on-line optimizer only needs observability of system *outputs*. If a system output becomes unobservable, then the on-line optimization cannot be executed. If there is a realistic probability that this

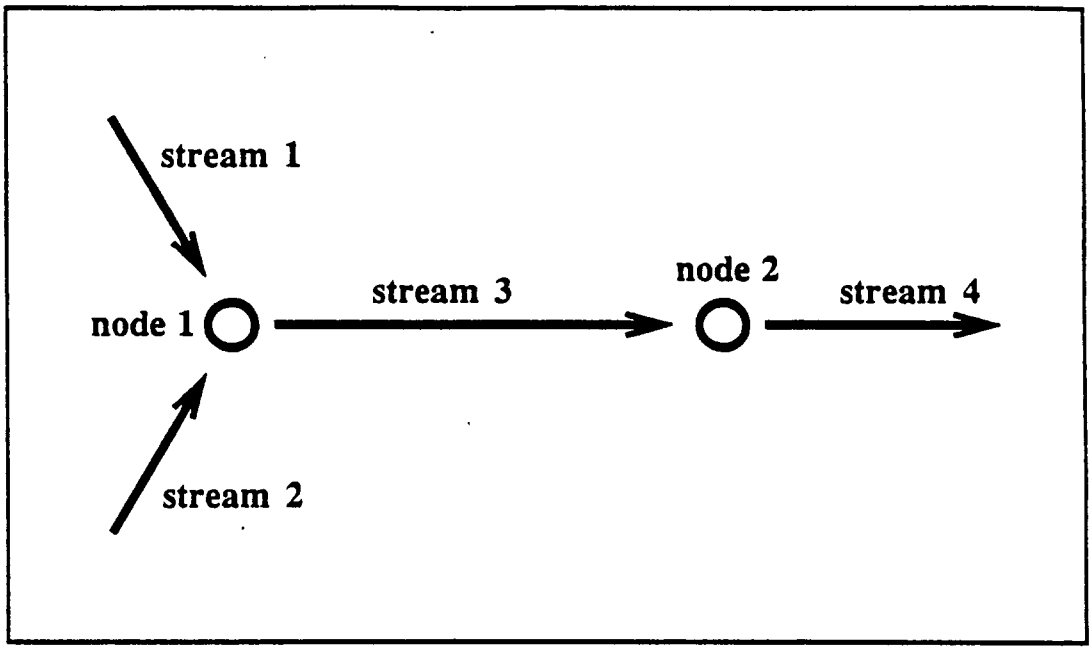

Fig.3.3.1. Simple Network with Lack of Resolution

happens, the on-line optimizer has to check for the output observability before running the optimization module.

Finally, it has to be stressed that data reconciliation and gross error identification can only be realized if sufficient measurement redundancy is available. If not enough redundancy is available, most reconciliation algorithms will nevertheless come up with an (senseless) answer. Checking for sufficient redundancy becomes very difficult if more than one gross error is detected. Some algorithms include a check and will limit the reconciliation to measurements with some redundancy (e.g. Crowe *et al.* [1983]).

3.3.3. Some Concepts and Some Algorithms

3.3.3.1. Overview

Problems with inconsistent measurements are not new. A basic data ad-

justment algorithm was presented by Demming [1946]. Kuehn and Davidson [1961] are among the first to propose a data reconciliation algorithm in the chemical engineering literature. Their algorithm is based upon a minimal least squares adjustment of the measurements subject to a number of reconciliation equations. The data reconciliation algorithm of Kuehn and Davidson [1961] does not even model gross or systematic errors. Ripps [1965] points this out, and is one of the first to incorporate a mechanism to detect gross errors. Since, many algorithms for gross error detection have been suggested. Data reconciliation techniques can be divided in six classes (Himmelblau [1988]).

(1) Techniques based on the least squares principle (maximum likelihood) represent the largest group. These techniques are also called Gauss-Markov estimates if the weights correspond to the inverse of the variance of each response. Techniques based on the least squares principle are most frequently reported on, most thoroughly tested and best known. In simple terms, one could say that they minimize a sum of squares of measurement corrections, usually with some provisions to detect systematic errors and replace them by rectifications or reconstructions. It is very important to detect systematic errors separately or they will be smeared out over the entire data set. Attention will be focussed on this type of technique since it is the most widely used and studied.

(2) Some techniques based on mathematical programming, in a more general way than the least squares application as specified in *(1)*, are proposed as well. For instance, Marro *et al.* [1981] attempt to decompose the complete (sometimes large) reconciliation problem. Their technique is limited to linear sets of reconciliation equations. In some other suggested algorithms, the Lagrange multipliers are crucial.

(3) Techniques based on Kalman filtering are used, especially for quasi steady

state systems. (e.g.: Stanley and Mah [1977], Stanley [1982])

(4) Techniques based on sensitivity coefficients are used, minimizing the sensitivity of the reconciled measurements for potential errors. (e.g.: Václavek *et al.*[1979])

(5) Himmelblau [1988] suggests a method of data reconciliation based on data from which gross errors are removed, using interval analysis. An advantage of the method is that no assumptions have to be made about the statistical distribution of the random measurement errors. A disadvantage is that interval analysis is not very well known among engineers, the major practitioners of data reconciliation. Software for interval analysis is not readily available. Hardware that speeds up the interval analysis calculations is being developed, but is not generally available. Therefore this technique is of no practical use yet, and definitely not one to be used on-line.

(6) Some methods are based on artificial intelligence principles, heuristics etc. These methods only pertain to very specific reconciliation methods, and are of no general interest. References can be found in Himmelblau [1988].

It is important to stress that many of the published reconciliation algorithms do not take constraints into account. This can lead to unrealistic results. The algorithm presented by Narasimhan and Mah [1987] is an example of an algorithm that *does* take leaks > 0 constraints into account. The MIMT method that is discussed below, takes minimum and maximum value constraints on the reconciled measurements into account. Implicit inequality constraints cannot be specified in the MIMT method.

3.3.3.2. Random and Gross Error Identification using LSQ

(LSQ = Least Squares) Let u be a matrix of *raw* measurements. These data will be adjusted with a term η such that the data reconciliation equations $B(u + \eta) = 0$ are met. Such adjustments are often obtained by minimizing

$\eta^T W \eta$ in which W is a weighting matrix, ideally equal to the covariance matrix of η. Section 3.3.5.1 deals with the calculation of this matrix. In the special case that the reconciliation equations are linear, the reconciliation problem has an analytical solution (e.g. Iordache *et al.* [1985]). Routines for solving a linear or nonlinear least squares problem are generally available in several software packages.

Different gross error detection schemes can be combined with the least squares data reconciliation approach. Four of these approaches will be presented here, others can be found in references given in two excellent reviews on data reconciliation and gross error identification by Hlavacek [1977] and by Mah [1981].

3.3.3.2.1. The MIMT method

The Modified Iterative Measurement Test (MIMT) is described by Serth *et al.* [1986] and will be illustrated in Chapter 5. There it is applied to data from a distillation tower. The method will also be described in more detail in Appendix D. Here the description is limited to the concept of the gross error identification. The MIMT method is an improvement of the Measurement Test method that was proposed by Mah and Tamhane [1982]. This test in principle is a simple check on the reconciliation residuals for outliers. The Measurement Test has two drawbacks. The first drawback is that the performance is very poor if multiple gross errors are present in the data set. The second drawback is that the reconciliation can fail and produce very large negative or positive values that are obviously wrong. The MIMT method is an improved version and does not have these drawbacks.

First the measurement corrections η are calculated as indicated above.

Then for every measurement j a criterion z_j is calculated, using :

$$z_j = \frac{\eta_j}{\sqrt{(v_{jj})}} \tag{3.3.2}$$

with :

$$V = QB^T(BQB^T)^{-1}BQ \tag{3.3.3}$$

In this equation, B is the coefficient matrix of the linearized reconciliation equations and Q is the measurement error covariance matrix. Section 3.3.5.1 deals with the calculation of that matrix. Equation 3.3.3 is derived as follows. Let u be a vector of unadjusted measurements, and let $\tilde{u}$ be a vector of adjusted measurements. Then the objective of the reconciliation is to minimize $(\tilde{u} - u)^T Q(\tilde{u} - u)$ subject to $B\tilde{u} = 0$. It can be shown that the solution of this problem can be written as

$$\tilde{u} - u = QB^T(BQB^T)^{-1}By; \tag{3.3.4}$$

The variance of this correction equals V in equation 3.3.3. Therefore z_j in equation 3.3.2 is N(0,1). The values of z_j (equation 3.3.2) are compared with each other, and the measurement that contains a systematic error with larger probability than the other measurements corresponds to the largest absolute value of z_j. This largest value is compared to a limit value to determine whether or not there is indeed a gross error. Serth *et al.* [1986] list as limit value of $|z_{j_{max}}|$, the $(1 - \frac{\beta}{2})$ confidence value of the normal distribution, in which β is defined as :

$$\beta = 1 - (1 - \alpha)^{\frac{1}{n}}; \tag{3.3.5}$$

with α the confidence interval of the test for systematic errors, and n the number of measurements involved in the data reconciliation. If a measurement is found to contain a gross error, then the corresponding stream is eliminated by nodal aggregation. The two nodes that the erroneous stream connect, are merged and

the erroneous stream disappears from the network. The process is then repeated until no more gross errors are found, or no more redundancy is available.

Serth *et al.* [1986] consider the method reliable and effective, and computationally less expensive than other methods with comparable performance. Iordache *et al.* [1985] publish suggestions to increase the power of the Measurement Test in gross error detections. These suggestions also apply to the MIMT method. A disadvantage of the MIMT method (and of all Measurement Test methods) is that only gross errors that influence only one measurement (i.e. measurement bias) can be detected and not errors that involve more than one measurements (e.g. node leak). The following method allows checking this type of gross error.

3.3.3.2.2. A Method Based Upon the Generalized Likelihood Ratio

Narasimhan and Mah [1987] presented a method for the detection of gross errors based upon the generalized likelihood ratio test. This method is illustrated in chapter 5, where it is applied to heat exchanger networks. In this method, a number of gross error models are hypothesized and the most probable is identified, using the generalized likelihood ratio method (see e.g. Hog and Tanis [1983]). Then a check is made as to whether or not this error is systematic. This method is one of the first to differentiate between types of errors, and not limit itself to measurement bias only. The next gross error is identified after compensation for the previous one, instead of using nodal aggregation. This approach allows a more thorough checking of systems with multiple gross errors. A more detailed description of the method can be found in Appendix E.

3.3.3.2.3. Heuristic Techniques for Gross Error Detection

In this type of identification technique, a ratio of the balance residual to the measured average flow of the balanced variable through the node is compared

with a preset value. This ratio for the jth node ϕ_j can be written as :

$$\phi_j = \frac{\sum_i b_{ji} u_i}{\frac{1}{2} \sum_i |b_{ji}| u_i}. \tag{3.3.6}$$

In this equation, b_{ui} is an element from the reconciliation equation matrix B, and u_i is the ith unadjusted measurement. The critical value for the absolute value of this ratio is usually chosen between 0.05 and 0.2, based upon experience. The method is rather inaccurate, and represents one of the earlier attempts at systematic error detection (Hlavácek [1977]).

3.3.3.2.4. Sequential Removal of Measurements

In order to locate measurements that contain gross errors, one can sequentially remove one or more measurements from the adjustment calculations and compare the results obtained with and without these measurements. Substantial differences would support the suggestion that the measurements that were discarded contained gross errors. This method is unattractive because of the computational expense, and the difficulty in interpreting the results (Hlavácek [1977]).

3.3.3.3. A Two-Level Technique

For several reasons it would be interesting to decompose the data reconciliation and gross error detection problem. Suppose that a plant is controlled by more than one computer. Assume that more than one process control computer does the loop control. Usually, only a condensed amount of information is sent to the supervisory machine or program. Furthermore, the computational effort necessary to reconcile a large system can be vast, to say the least, and can easily be too much to be handled by one computer in real time. Added to that are the compatibility and timing problems involved with sizable data transfers (*all* process measurements) between the local and the supervisory process computers

or control applications, if data reconciliation and gross error identification are done in a central unit. These problems constitute an incentive to decompose the large reconciliation problem into smaller, easier to manipulate subproblems. Often the Jacobian of the reconciliation equations is sparse, and this suggests a decomposition technique.

Marro *et al.* [1981] propose a method to handle data reconciliation in a decomposed scheme, *without iteration.* They assume that every plant can be divided into a number of departments, with measurement vector u_i for the ith department. The departments are connected with streams v. Marro *et al.* [1981] assume that the reconciliation equations within the departments as well as for the interconnecting streams, are *linear.* Figure 3.3.2 schematically presents the organization of the problem as suggested by these authors. The data reconciliation problem is then formulated by Marro *et al.* [1981]as :

$$\min_{\eta_i,\zeta} \Big(\sum_{i=1}^{N} \eta_i^T W_i \eta_i + \zeta^T W_0 \zeta \Big), \tag{3.3.7}$$

$$\begin{aligned} \text{subject to} \quad : \quad & D_0(v+\zeta) = 0, \\ & \text{and } B_i(u_i+\eta_i) + D_i(v+\zeta) = 0, \forall i = 1,\ldots,N. \end{aligned} \tag{3.3.8}$$

In this formulation, N is the number of "departments", η_i is the measurement adjustment vector for department i, ζ is the measurement adjustment vector for the connecting streams, W_i is a symmetric weighting matrix for the ith department, ideally the measurement error covariance matrix, W_0 is the same for the connection streams. The first set of reconciliation equations listed concern the connection streams *only* with the departments as nodes, and the remaining group of reconciliation equations are the reconciliation equations within the departments. Note that Marro *et al.* [1981] assume that errors in measurements of different departments, and errors in departmental measurements and measurements in connection streams are uncorrelated. This is a realistic assumption.

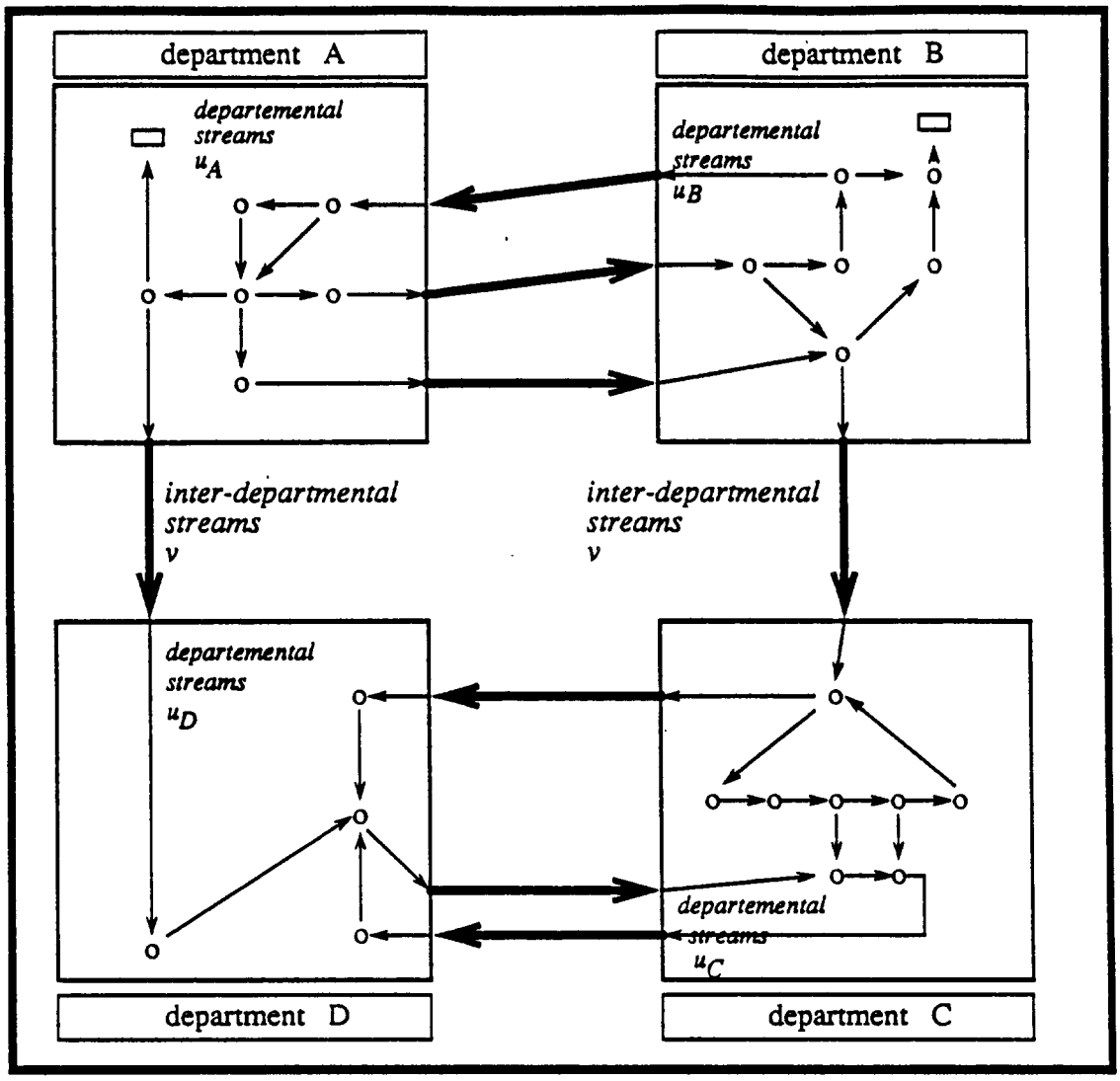

Fig.3.3.2. Schematic Illustration of Marro et al. [1981] Organization of a Plant.

However, correlations in errors can be caused by errors in data highways that are shared by measurements or by errors in multiplexers. Correlation can also be caused if a measurement device, e.g. an analyzer, is shared between a number of streams. Marro *et al.* [1981] consider a level 1 problem, which is the

following :

$$\min_{\eta_i} c_i(\zeta) = \sum_{i=1}^{N} \eta_i^T W_i \eta_i,$$

$$\text{subject to} \quad : \quad \mathrm{B_i(u_i + \eta_i) + D_i(v + \zeta), \forall i = 1, \ldots, N.} \tag{3.3.9}$$

This level 1 problem is solved with ζ as a parameter.

$$\eta_i = -W_i^{-1} B_i^T (B_i W_i^{-1} D_I^T)^{-1} (B_i u_i + D_i v + D_i \zeta) = l_i(\zeta), \tag{3.3.10}$$

The optimal value, *up to a constant in* ζ is :

$$\begin{aligned} c_i(\zeta) =& u_i^T B_i^T (B_i W^{-1} B_i^T)^{-1} D_i \zeta \\ &+ v^T D_i^T (B_i W_i^{-1} B_i^T)^{-1} D_i \zeta \\ &+ \frac{1}{2} \zeta^T D_i^T (B_i W_i^{-1} B_i^T)^{-1} D_i \zeta. \end{aligned} \tag{3.3.11}$$

The level 2 problem coordinates the level 1 problems.

$$\min_{\eta} \sum_{i=1}^{N} c_i(\zeta) + \frac{1}{2} \zeta^T W_0 \zeta, \tag{3.3.12}$$

$$\text{subject to} \quad : \quad D_0(v + \zeta) = 0. \tag{3.3.13}$$

Marro *et al.* [1981] propose the following scheme for plant data reconciliation :

a) The master computer sends raw connection stream data to the slaves.

b) The department solves its local reconciliation problem, and sends its cost as a function of ζ to the master.

c) The master then solves the level 2 problem, and sends the corrected ζ's to the locals.

d) Each departmental computer computes the corrections for the given ζ. The total amount of information that is exchanged between the various computers is rather small. Only an array ζ and a few matrices are being transferred.

Marro *et al.* [1981] do not indicate how to handle gross errors. On the local level, gross errors can for instance be removed by any of the gross error identification methods presented in section 3.3.3.2.1–4. On the connecting level it is much less obvious how to deal with gross errors on the supervisory level. It is also not clear what the influence of gross errors in the connecting stream measurements is on the departmental data reconciliations.

This technique does not reduce the amount of calculations. It suggests a technique for spreading the computational efforts over several computer nodes, and reducing the amount of data that has to be transferred.

Finally, the method will require iteration if it is extended to nonlinear reconciliation problems. Then the frequent communications between various computers that is needed during every iteration is potentially a significant disadvantage, and will slow down the procedure considerably.

3.3.3.4. Unsteady State Data Reconciliation

3.3.3.4.1. Quasi Steady State Data Reconciliation

Stanley [1982] defines a quasi steady state variable as a variable that can change only very slowly, or have an occasional sharp transition in between two steady state values. Instrument bias, which can be considered as a very low frequency error, can be modeled as a quasi steady state variable. Atmospheric conditions (temperature, pressure) can be considered as quasi steady state variables as well. Some more examples are given by Stanley [1982]. Stanley and Mah [1977] and Stanley [1982] propose a discrete Kalman filter to estimate and reconcile systems using these variables. These authors suggest a "cascade estimation". Steady state reconciliation algorithms are able to recognize instrument bias under certain conditions, but are "paralyzed" once it is detected. The steady state data reconciler works with a "snapshot" of the plant, and does not

relate the current picture to the previous one. The approach suggested by Stanley [1982] and Stanley and Mah [1977] would allow to model instrument bias using a *random walk* model. The random walk model assumes that the value of the instrument bias would not change in between two measurement time intervals. Measurements are then used to correct that value of the instrument bias. Instead of over and over again detecting instrument gross error, and neglecting the corresponding measurement, instrument bias could be recognized as a quasi steady state variable and quantified. The biased measurement would then still be useful. The following equations are used by the filter : (let $x_{qss}(t)$ be a quasi steady state variable vector, and let $z(t)$ be the corresponding measurement vector) :

$$\begin{aligned} x_{qss}(t_k) &= x_{qss}(t_{k-1}) + \quad \text{process noise vector with } t_k = t_{k-1} + \Delta t; \\ z(t_k) &= x(t_k) + \quad \text{measurement noise vector.} \end{aligned} \tag{3.3.14}$$

The linear filter equation is :

$$\hat{x}_{qss}(t_k) = (1 - K)\hat{x}_{qss}(t_{k-1}) + Kz(t_k), \tag{3.3.15}$$

with K the Kalman filter gain.

Stanley [1982] presents a simple example that describes the method to estimate a quasi steady state variable and still reconcile. Consider a pipe split in which a stream 1 is split into three streams (2,3 and 4). The split is assumed to be a quasi steady state variable. Physically this could correspond to a fixed split in the piping. No control valves are involved in the determination of the down stream flows 2,3 and 4 as a function of stream 1. The split will change as a function of fluid mechanistic variables. Only flows 1 and 2 are measured. The flows are represented by x_1 and x_2, and the corresponding measurements are written as z_1 and z_2. At every time step, the algorithm goes through the following steps :

(1) calculate a recent approximation of the fraction of the stream 1 that is passed on to stream 2 : $c_{new} = z_2(t_k)/z_1(t_k)$;

(2) update the fraction that will be applied in the date reconciliation using the linear filter equation :$\hat{c}(t_k) = (1 - K)\hat{c}(t_{k-1}) + Kc_{new}$;

(3) reconcile using the reconciliation equation : $\hat{c}(t_k)x_1 + x_2 = 0$.

In this way, the low frequency changes are passed on to the bias estimation. The high frequency changes are reconciled. Similar methods can be applied to a bias that can be considered as a quasi steady state variable.

3.3.3.4.2. Dynamic Data Reconciliation

Under certain conditions (e.g. nodes with significant storage) actual *dynamic reconciliation* may be necessary. This is beyond the scope of this work, and only reference is made to the fact that limited results in dynamical reconciliation exists. For more details see Stanley [1982], who introduced the term "dynamic reconciliation" to chemical engineering. This author proposes three approaches to reconciliation problems that have dynamics that are too slow to reach steady state fast, and too fast to make the variables quasi steady state (and use Kalman filtering). In a first approach the data are filtered, cutting out the high frequencies. Dynamic equations are approximated with algebraic equations. The second approach simply ignores the dynamics, and accepts the resulting inaccuracies - it is obvious that the system must be close to "very fast" or "quasi steady state"-. A third approach attempts to use steady state reconciliation equations to *dynamically synchronized* values. This concept was introduced by Bartman [1981]. For related topics (such as mixtures of static and dynamic constraints, parametric failures,...) see also e.g. Morra *et al.* [1981], Park and Himmelblau [1982], Lancraft and Caglayan [1982], Tulpule and Knapp [1982], Gentler and Almasy [1973].

3.3.4. Data Reconciliation and Model Estimation

In an on-line optimizer algorithm, the results of data reconciliation are used to identify a steady state model. This sequence of manipulations poses important requirements to the data reconciliation. Data reconciliation equations should be limited to equations that do not contain any parameters or relations that are being used by the model estimation routines. This is explained in the following lines.

Complex steady state models will have several responses that have to be fitted to plant data. This fit will be realized by minimizing an estimation criterion. A problem during the parameter estimation of multiresponse models is that responses have to be weighted versus one another and they are correlated with each other. Box and Draper [1965] propose an estimation criterion that overcomes problems with weights and correlations. The criterion that Box and Draper [1965] suggest (to be minimized) is :

$$|V|^{-\frac{n}{2}}, \tag{3.3.16}$$

with

$$V = \{v_{ij}\} \quad \text{and} \quad v_{ij} = \sum_{u=1}^{n} (\tilde{y}_{iu} - \hat{y}_{iu})(\tilde{y}_{ju} - \hat{y}_{ju}); \tag{3.3.17}$$

This criterion will discussed in more detail in section 3.4.2.1.1. The criterion becomes invalid if the model responses are linearly dependent. This can occur if the data become linearly dependent, as pointed out by Box *et al.* [1973]. Therefore, one should avoid at all cost applying this estimation criterion to data arrays that have linearly dependence among the columns. Data reconciliation enforces certain relationships between measurements. Many of these relationships are mass and energy balances. Thus, after data reconciliation, data can be linear dependent. The data reconciliation can *introduce* linear relationships. Therefore, data reconciliation should be limited to equations that will not endanger the parameter identification quality. Often parameter estimation uses

equations describing phenomena within the unit. Data reconciliation usually concerns material and energy balances in which plant units are network nodes, in other words, balances of inter unit streams. Therefore, the scope of the equations used by data reconciliation on one hand and by parameter estimation on the other hand are often different. That would obstruct including equations in the set of reconciliation equations that are also part (maybe implicitly) of the set of parameter estimation equations. Furthermore, the data used by the model parameter estimator could be limited to a nonredundant set, offering the same observability of the unit or plant as the original set. This approach would exclude all the relationships that cause redundancy and are enforced by the reconciler. Finally, one can check the reconciled data fed into the parameter estimation routine by looking for linear relationships among the reconciled data, e.g. by using principal component analysis (Joliffe, [1982]).

It can be concluded that the data reconciliation module and the model fitting module have to be generated in the same effort, and have to be maintained together. Consistency and disjunction between the sets of equations is necessary. The problem can altogether be avoided by combining data reconciliation and model fitting in one step. Mc Donald and Howat [1988] compare data reconciliation and model parameter estimation combined in one single step to data reconciliation and parameter estimation in a sequential approach. They note that the combined procedure is statistically rigorous. They point out that both procedures perform well on their example : a flash operation. The combined procedure performs better overall, and the difference becomes larger as the noise level increases. However, the combined procedure consumes much more computer time, for a marginal improvement. Therefore, combined data reconciliation and model fitting is certainly not suggested for large models.

3.3.5. Problems With Practical Applications

3.3.5.1. Measurement Error Covariance Matrix

Most reconciliation methods (e.g. all least squares based methods) will use the measurement error covariance matrix to weight measurement adjustments. It is difficult to obtain this measurement error covariance matrix. Often one will have to determine the measurement error covariance matrix from a set of measurements at a practical steady state. Often correlations between errors of different measurements are neglected, which does not always correspond to the reality, in particular when common data transmission paths are used. Almasy and Mah [1984] propose a very elegant scheme to estimate measurement error covariances from process data using the data reconciliation equations. The procedure is without any doubt of interest from a theoretical point of view, but it is computationally involved. The method performs well up to a correlation coefficient level to 0.1. This correlation level is rather low. Almasy and Mah [1984] do not recommend this procedure at higher levels of correlation because of numerical accuracy and convergence problems. Furthermore, the method will fail without warning if bias is present in the raw data, which is often the case in process data. Therefore, in practice the measurement error variances still have to be obtained in an efficient way from known measurement accuracy (vendor information) and/or from previous measurements. Since an optimizer only runs at a relatively low frequency, repeated data sets can usually be collected at the same steady state. Statistical information can be calculated from these sets, and can be used as approximation of the true variances.

3.3.5.2. Computational Effort

The computational effort connected with data reconciliation and gross error identification can increase rapidly with the size of the reconciliation problem. For instance, the method Narasimhan and Mah [1987] based upon the general-

ized likelihood ratio method inverts a symmetric matrix with order equal to the number of reconciliation equations. *If* the measurement error covariance matrix is constant, then this inversion can be done off-line, but this specific inverted matrix becomes invalid if more than one gross error is detected. For every gross error found, a new matrix of smaller order (previous order -1) has to be inverted. For large systems, this computational effort becomes potentially cumbersome. Nonlinear reconciliation problems may require iteration and therefore require significantly more number crunching. The decomposition proposed by Morra *et al.* [1981] and described in section 3.3.3.3 does not reduce the volume of computations that have to be completed but only spreads the work over several computers, and limits the stream of information that has to be exchanged.

One way to actually reduce the amount of computation, is by splitting up the complete network of nodes into partial networks. Every network could then be reconciled separately, and gross errors can be identified in every partial network. It is clear that the results will be inferior to the results of a complete data reconciliation and gross error identification. The reconciled streams at the "seams" will not match. However, often gross error identification is more important than the data reconciliation and the data reconciliation can be omitted. This is so because data reconciliation deals with the random errors (= noise). If the noise is within reasonable limits, data reconciliation does not have to be applied. Gross errors can always occur, and therefore it is useful to protect the model fitting module from such systematic errors, regardless of the noise level. Local gross error identification can monitor the plant less accurate than plant-wide gross error identification but with a significantly smaller effort.

Because of the computational effort that goes together with data reconciliation and gross error detection, and because of the particular character of many practical applications (sparsity) parallel computing can be considered, i.e.

application of neural networks.

3.3.5.3. Resolution in Determining Multiple Gross Errors

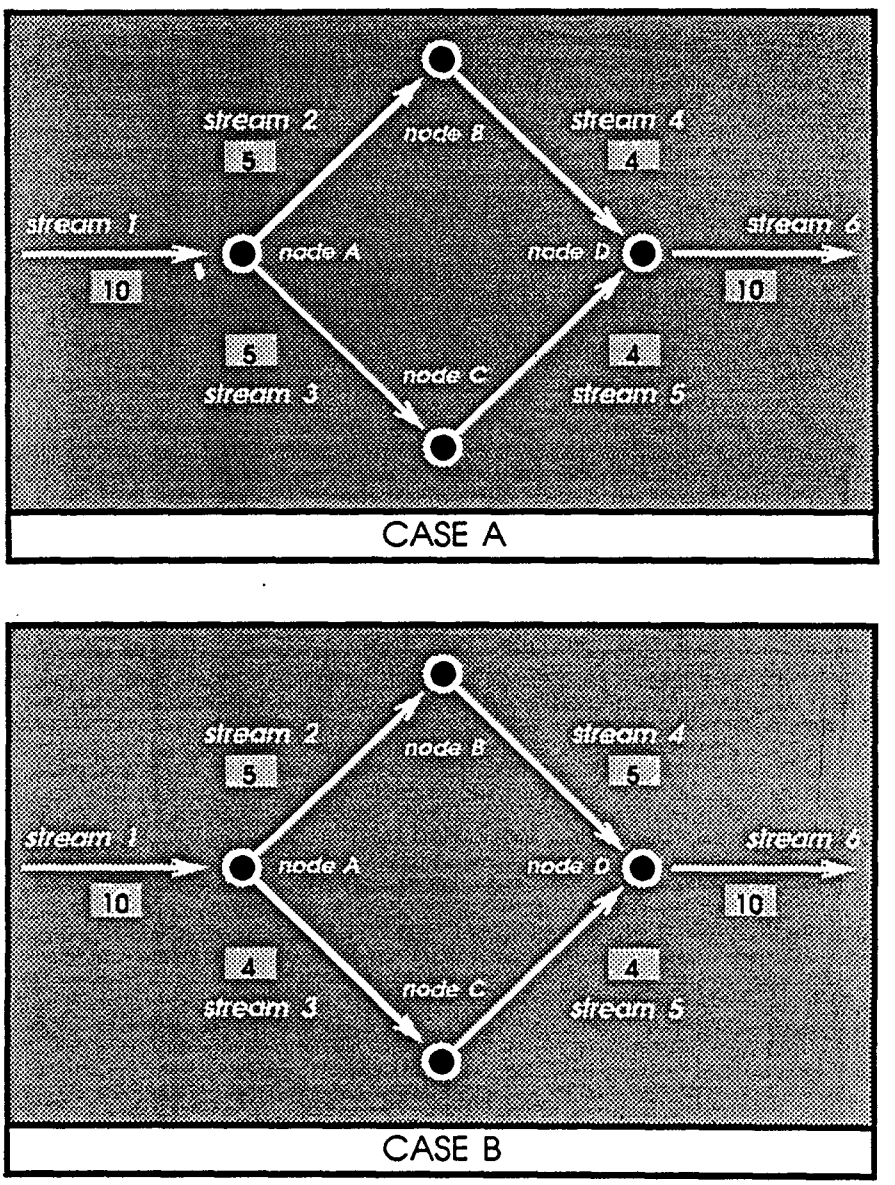

Fig.3.3.3. Network with two measurement errors. In Case A both errors can be detected. In Case B it is impossible to detect both errors unambiguously.

Figure 3.3.3 considers a network with four nodes. Stream 1 is split in two

streams 2 and 3 in node A. Stream 2 enters node B, and stream 3 enters node C. Stream 4 leaves node B, and stream 5 leaves node C. Stream 4 and 5 join to stream 6 in node D. Figure 3.3.3 presents two cases. In both cases, streams 1 and 6 have a flow rate measurement value of 10, stream 2 has a flow rate measurement value of 5, and stream 5 has a flow rate measurement value of 4. However, the measurement values of the flow rates for stream 3 and stream 5 for case A and case B are swapped. In Case A both errors can be detected. It is clear that stream 4 and stream 5 are biased. The only alternative is a leak in node B and node C combined with a bias in stream 6 compensating for both leaks exactly at the same time. That alternative is much less probable than a bias in stream 4 and stream 5. In Case B it is impossible to detect both errors unambiguously. It is possible that the measurements of streams 2 and 4 are biased, and that both measurement values should be 6. It is also possible that stream 3 and stream 5 are wrong, and that the values should be 5. The following remarks can be made :

(1) True lack of resolution often remains undetected. Process and measurement noise will cause the criterion values for the truly undistinguishable gross errors to be slightly different from each other. Therefore one of the unresolvable gross errors will be considered to be slightly more possible than the others. The chance of drawing a wrong conclusion is roughly $\frac{n-1}{n}$ for n unresolvable gross errors. This suggests that there should be a substantial difference in probabilities for two or more gross errors to be resolvable. Introducing this principle would of course make the gross error identification algorithms more complicated, and furnish them with one more tuning variable that has to be set at a "good" value. Potential unresolvable sets of gross errors could be identified off-line by simulation. Knowledge on which combinations of gross errors are potentially unresolv-

able would allow to check more carefully before certain errors are assigned as systematically erroneous. It is not clear at this point how to do this in a general and automated way.

(2) If a set of unresolvable errors is detected, then it becomes very difficult to search for the potential next gross error. One possible way to do this, is by accepting every error separately and look for possible additional errors accepting that one of the unresolvable errors is true. That combinatorial method could increase the computational burden considerably. There is at this moment no good way to continue the search for gross errors after unresolvable sets of gross errors are detected.

3.3.6. Conclusions of Data Reconciliation

Data reconciliation and gross error identification is discussed in this section. In the case studies, it will be demonstrated that data reconciliation and most definitely gross error identification can be very useful in on-line optimization.

Certain problems may make implementation difficult. The measurement error covariance matrix, a quantification of the probability of the occurance of one error or a combination of errors, is difficult to obtain. Multiple, interacting errors can "confuse" the reconciler. Observability and redundancy principles are not clearly defined in case more than one gross errors is present. Most importantly, from a practical point of view, is the large requirement of CPU time. Partitioning reconciliation problem reduces the time requirement considerably, but a loss of performance can be expected. This is demonstrated in Chapter 5.

3.4. PLANT MODELS IN AN ON-LINE OPTIMIZATION

3.4.1. Plant Models : Why Updated ?

Plant models are used in two blocks of the on-line optimizer. During the optimization phase, steady state plant models are used to describe process behavior. Before that, the model parameters are updated by the model updating block. This section on plant models is focussed on model parameter estimation or model fitting more than on model building or identification. The model structure is not changed by the on-line optimizer, *only* the model parameter values are updated. The on-line optimizer uses a steady state model equations as constraints in the optimization. The model reflects the steady state behavior of the plant as a function of several variables, such as the optimization variables. It is in general nonlinear. Such a model contains parameters that need updating for two reasons :

- ◇ The model does not exactly reflect the steady state of the plant. There is some plant model mismatch. In order to obtain a good fit of the model to the plant behavior in the neighborhood of a particular operating condition, parameters will be adjusted to "pull" the model closer to the true plant behavior in the considered range of operation. It is clear that the "poorer" the model structure, the more the model parameters will have to be adjusted for changes in operating conditions. In principle, the affordable magnitude of the measured model plant mismatch is limited by the rate at which new data can be expected to become available during operation. Setpoint moves based upon models with a poor model structure, may have to be limited in range because the model is only valid within a limited range around the current operating condition.
- ◇ Because of slow changes in plant equipment properties, certain model pa-

rameters may have to be updated regularly. Typical examples are catalyst deactivation, heat exchanger fouling, coke buildup. Such characteristics have to be sufficiently slow to be picked up by the model identification. If such variations are "fast", provisions will have to be taken for the loop control to correct for them.

The model equations used to update the model parameters do not have to be the same as the equations that are used as constraints in the optimization, reflecting the steady state plant behavior. The model updating block could, for instance, even fit a dynamic model to the plant data. A steady state version of that model with the parameters that are obtained from the dynamic fit, can then be applied during the optimization. This concept was proposed by Bamberger and Isermann [1978]. The model parameter estimation could also work with a model with approximate or simplified dynamics (e.g. exponential filtering), in order to be able to obtain more information from the data stream, without the need for developing an often expensive and/or inaccurate dynamic model. Frequently, however, the model equations that are fitted to data in order to obtain model parameter estimates, and the steady state equations used in the optimization module, are the same.

3.4.2. Model Parameter Identification

3.4.2.1. Estimation Criterion

Consider a model that is linear in parameters and data. The measurements have zero mean normal errors. Then the likelihood (probability density function) that the set of measurements correspond to the model with parameters b can be written as :

$$p(Y|b,\sigma^2) = (2\pi)^{\frac{1}{2}nN}|A|^{\frac{1}{2}n} \exp\Big(-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left((\sigma^2)^{-1}\right)_{ij}v_{ij}\Big); \qquad (3.4.1)$$

with:

$$v_{ij} = \sum_{u=1}^{n} (y_{iu} - \hat{y}_{iu})(y_{ju} - \hat{y}_{ju}). \tag{3.4.2}$$

In this equation, n is the number of observations, N is the number of responses and $|A|$ is the determinant of the inverse of the covariance matrix $(\sigma^2)^{-1}$. Two important concepts in parameter estimation start from this probability expression. The first concept attempts to find the combination of parameters that maximizes the probability given above, in other words, attempts to maximize the probability that the measurements have the values that were found, given the values of the parameters. The second concept uses the distribution of y given the parameters and σ^2 ($p(Y|b,\sigma)$ as determined above) to calculate the distribution of the parameters, given the measurements ($p(b|y)$). The first approach needs σ^2, and the second doesn't. However, the second approach needs some additional assumptions concerning the unconditional distributions of b and σ^2. Both approaches are described below. References give more detailed information.

3.4.2.1.1. Maximum Likelihood Principle

The maximization of $p(Y|b,\sigma^2)$ with respect to the model parameters can be simplified to a minimization of

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{u=1}^{n} (\sigma^2)^{-1}_{ij} \cdot (y_{iu} - \hat{y}_{iu}) \cdot (y_{ju} - \hat{y}_{ju}). \tag{3.4.3}$$

In the special case of a model with a single response, this becomes the well known least squares criterion $(Y - \hat{Y})^T(Y - \hat{Y})$, if Y is an array $\{y_{iu}\}$. Another special case occurs if there is no correlation between the responses. Then the criterion can be simplified to (Gauss-Markov estimation) :

$$\sum_{i=1}^{N}\sum_{u=1}^{n} \left((\sigma^2)^{-1}\right)_{ii} \cdot (y_{iu} - \hat{y}_{iu})^2; \tag{3.4.4}$$

the weighted least squares criterion. Usually σ^2 is unknown a priori. If σ^2 is unknown, it can be approximated by the mean residual sum of squares after a solution to the estimation criterion is found. That way an iterative procedure can be set up, or σ^2 can be expressed as a function of the model parameters (Christian *et al.* [1974], Meyer [1982]). This iterative approach is not frequently applied though, and often the correlations among responses are neglected and fixed weights are used, which can lead to significant errors (e.g. Wenworth [1965]). Criteria to recognize circumstances to detect cases for which a significant fitting error can be expected, do not exist.

3.4.2.1.1. A Bayesian Estimation Criterion

A formulation of the theorem of Bayes is :

$$p(B|A) = \frac{p(B)p(A|B)}{p(A)}. \tag{3.4.5}$$

Applying this to the parameter estimation problem that is discussed here :

$$\begin{aligned} p(b,\sigma^2|Y) &= \frac{p(b,\sigma^2)p(Y|b,\sigma^2)}{p(Y)}; \\ &= \frac{p(b)p(\sigma^2)p(Y|b,\sigma^2)}{p(Y)}. \end{aligned} \tag{3.4.6}$$

The last step is only allowed if the independence between the *unconditional* distributions of b and σ^2 is assumed. This approach is suggested by Box and Draper [1965]. In a second step, they integrate σ^2 out, to obtain $p(b|y)$. After some manipulations, Box and Draper [1965] formulate :

$$p(b|Y) \propto \left|\{v_{ij}\}\right|^{-\frac{1}{2}n}, \tag{3.4.7}$$

with v_{ij} given by equation 3.4.2. Maximizing the determinant of the matrix V with respect to the model parameters will results in a most probable values of b. Box and Draper [1965] also indicated that the Bayesian and the Maximum Likelihood approach are identical for the special case of models with one response only.

3.4.2.2. Estimation Criterion Minimization

If the model is linear in the parameters, an analytical solution to the problem can be formulated. In the case of nonlinear models, estimation criteria often behave very closely to a quadratic function in the neighborhood of the estimation problem solution. This fact is used in several optimization algorithms that were developed to solve nonlinear least squares problems. Various techniques are available, and most standard statistics computer packages have routines for the minimization of estimation criteria. Most of these routines are based upon Marquardt's [1963] compromise between a Taylor (linearization) series approach, and a Newton Gauss approach. Levenberg suggested a similar approach [1944]. Also Powell [1965] suggested an often applied method for minimization of a sum of squares.

3.4.2.3. Detection of Outliers

The data that are used to identify the parameters of the steady state model, have first passed a steady state detection step, and a data reconciliation and gross error detection step. The gross error detection checked for internal inconsistencies in each steady state data set. However, the gross error detection can only use the available redundancy to check for gross errors. Certain errors can still be completely undetectable. They will result in outliers in the data set. Such points have to be removed from the data set. However, not every outlier is due to erroneous measurements. An outlier, a "data point that is very different from the majority of measurements", can also represent the correct plant behavior and more specifically an aspect of that plant behavior that is not represented in the rest of the data. This type of outlier should not be removed. Barnett and Lewis [1984] describe various methods of detecting outliers in univariate and multivariate samples of different types of distributions.

During the parameter estimation phase the data should be checked for outliers. This can only be done after a parameter estimation. Outlier detection has to be done upon the basis of the residual distribution characteristics. The most probable outlier in the residual distribution has to be detected. If the probability of the most probable outlier is higher than a preset value (e.g. 99%) then the outlier is confirmed. The corresponding data point is removed, and then a new parameter estimation has to be performed. This iterative procedure is not guaranteed to converge. The outlier detection in the residual distribution is a multivariate distribution problem.

More details on outlier detection in the residual distributions can be found in Barnett and Lewis [1984]. "Good" outlier detection is difficult. It depends upon what is known of the residual distribution. That again depends upon the plant model mismatch. A disadvantage of outlier detection is the following. If an upset in the model parameters occurs, e.g. caused by a significant change in operating conditions, the first sets of data are refused as outliers. Therefore, data sets that are identified as outliers should be admitted to the data set again during the next parameter estimation session.

3.4.2.4. Parameter Estimates Confidence Intervals

Let b be the estimate of a vector of parameters. Assume that the "true" values of these parameters are given by the vector β. Accepting the assumptions we listed earlier : model linearity in data and parameters, and zero mean normal errors in the measurements, one can prove that b is a sample of a multivariate (univariate if only one parameter) normal distribution with unknown mean β and unknown covariance V_b. The probability density for the parameter b can then be written :

$$p(b) = (2\pi)^{-\frac{1}{2}p}|V_b|^{-\frac{1}{2}} \exp\Big(-\frac{1}{2}(b-\beta)^T V_b^{-1}(b-\beta)\Big). \qquad (3.4.8)$$

If V_b would be known, then confidence contours with significance α could be calculated from the above formula. However, an approximation of V_b has to be used. This results in the following formula for a $100(1-\alpha)\%$ confidence region for values of β (Draper and Smith [1981], Beck and Arnold [1977]) :

$$(\beta - b)^T X^T X(\beta - b) \leq p s^2 F(p, n-p, 1-\alpha) \tag{3.4.9}$$

with p the number of parameters, s^2 the measurement variance, and $F(p, n-p, 1-\alpha)$ the $1-\alpha$ confidence value in the F distribution with p and $n-p$ degrees of freedom. X are the estimation residuals $(Y - \hat{Y})$. Individual confidence intervals for values of β can be obtained from the formula :

$$b_i \pm \sqrt{s^2 \cdot ((X^T X)^{-1})_{ii}} \cdot t(n-p, 1-\frac{\alpha}{2}) \tag{3.4.10}$$

with $t(n-p, 1-\frac{\alpha}{2})$ the $1-\frac{\alpha}{2}$ confidence value of the t distribution with $n-p$ degrees of freedom. s^2 is the residual variance, and $X^T X$ are the diagonal elements of the parameter covariance matrix. It has to be noted that individual confidence intervals for parameter estimates are very useful, but do not show the sometimes considerable correlation between parameter estimates.

The above formulas are approximate if one of the conditions listed earlier are violated, e.g. the model is not linear. It is also assumed above that no model plant mismatch is present. Model plant mismatch will influence the size of the confidence intervals, and in that way structural model inaccuracies are accounted for in an approximate manner.

3.4.3. Choice of Model

3.4.3.1. Factors that Influence Goodness of Fit

While choosing the steady state model, a tradeoff has to be made between good fit and complexity. Fairly simple models will be computationally cheap

to fit, and result in optimization problems that are easier to solve. But simple models can result in a relatively poor fit, and can cause a poor on-line optimizer performance. Models can be mechanistic or empirical. Mechanistic models often provide a better overall fit, and are in principle safer for extrapolation outside the region covered by available data. Sometimes, empirical models are simpler (e.g. linear regression) while still offering a satisfying fit. Mechanistic models may not be available at all. If more than one model is available, and therefore a choice will have to be made, then this choice has to be made off-line. It is not always clear which model provides a better fit. Statistical hypothesis tests can assist in discriminating between models. In the following paragraphs, some of the most important tests and criteria will be presented that allow to evaluate the quality of the fit a model can provide to a given set of data. Therefore these tests will also allow to compare models in their ability to fit this given set of data. This discussion is only an overview, for more details, see e.g. Hogg and Tanis [1983], Beck and Arnold [1977], Draper and Smith [1981], Box *et al.* [1978], etc. There is no exact definition of "goodness of fit". It is a fuzzy concept that everybody understands intuitively, but no rigid definition of it has ever been formulated. The various tests presented below are different. Slightly different concepts of the quality of a fit are used by the different sets. Therefore tests may not be compatible, although often they will give similar indication about the goodness of a fit. Which test is most appropriate under which condition depends upon the form of the models that are being compared, and the computational effort that can be spent. The three first methods are useful as fit quality trend indicators, but don't allow a refined statistical evaluation. All methods can be used for multiresponse models, as well as in the case that several measurements are taken for the same independent variable value (repeated measurements). The first requires the introduction of a matrix notation, and the second a careful

check of the degrees of freedom of the involved statistics. The matrix notation that goes together with multiresponse models is illustrated in the paragraph on the χ^2 goodness of fit test. The influence of repeated measurements on degrees of freedom of various sums of squares is illustrated in the paragraphs on F tests.

It is worth noting that the data set that is used to discriminate models has to be studied before it is used. Plant data will often be obtained while the plant is under operation. Certain *supervisory* policies may be continuously applied by the operators. "If variable x_1 is large, and x_2 is between z_a and z_b then, set x_3 low." If rules like that are applied, then the data set will of course reflect this, and lack some dimension. The models can only be expected to be valid while this policy is being used. The validity of the models is not guaranteed if this specification is not met. Feedback control will also correlate system outputs and inputs. Box and Mc Gregor [1974] discuss the identification of closed loop dynamic stochastic systems. They point out that open loop identification methodologies will not obtain good fits of closed loop systems. In the steady state case also, the controllers make the system inputs dependent upon the outputs. It is important to take that into account. This correlation has to be modeled in. Open loops have to be flagged and this important information has to be passed on to the identification and optimization modules. Principal component analysis (Joliffe, [1982]) can give a clue to potential *linear* relationships between variables in the database, but insight in the procedures that were used while collecting the data should be interpreted where possible.

Noise (process noise and also some measurement noise) will be correlated under practical conditions. Biases will be detected by the gross error identification. But correlated process noise will cause problems during the data reconciliation as well as in the model parameter estimation. Measuring the disturbances where possible, will improve possible filtering, smoothing and estimation.

3.4.3.2. Some Tests for Goodness of Fit

3.4.3.2.1. The (Square of the) Multiple Correlation Coefficient

The square of the multiple correlation coefficient is defined as :

$$R^2 = \frac{\sum_i (\hat{Y}_i - \bar{Y})^2}{\sum_i (Y_i - \bar{Y})^2}, \tag{3.4.11}$$

and can be interpreted as the ratio between the variance in the data that is explained by the regression, and the total variation in the data. A larger R^2 reflects a better fit. A statistically fundamental assessment of the validity (with confidence α) or comparison between fits of competing models is difficult and very often not done. It can be shown that under assumptions of linearity of the model in parameters and data, and zero mean gaussian error in the measurements, R^2 follows a $\beta(\frac{1}{2}(p-1), \frac{1}{2}(n-p))$ distribution, with p the number of parameters and n the number of data points. Violation of the assumptions reduces this statement to an approximation. Tables of the β distribution are not generally available, and therefore the R^2 statistic is seldomly used as a test statistic. For a discussion of the β distribution, see e.g. Hogg and Tanis [1983]. An advantage of R^2 is that it is dimensionless. The R^2 statistic is usually used as a gross indicator.

3.4.3.2.2. The Adjusted R^2 Statistic

The adjusted R^2 statistic is defined as ;

$$R_a^2 = 1 - (1 - R^2)\Big(\frac{n-1}{n-p}\Big). \tag{3.4.12}$$

It is also known as the C_p statistic. An adjustment has been made with respect to the original square multiple correlation coefficient for the degrees of freedom of the residual sum of squares of prediction errors, and the total sum of squares. More details can be found in Draper and Smith [1981].

3.4.3.2.3. The Residual Mean Square

The residual mean square is defined as a sum of squares of the residuals divided by the degrees of freedom connected with this sum of squares, i.e. the difference between the number of measurements and the number of parameters. The residual mean square can be used to compare the fit provided by two or more similar models with different number of parameters. The test is more specific to check overfitting. The data set has to be typically large for this method to have any significance (at least ten times as many data points as number of parameters). This test is not dimensionless. Since the expected value of the residual mean square equals σ^2 (the measurement error variance), which is not *exactly* known (as opposed to for instance the next test statistic with a χ^2 distribution) the method cannot be used for a statistical evaluation. Evaluation is based upon personal judgement more than anything else.

3.4.3.2.4. χ^2 Goodness of Fit Test

A χ^2 goodness of fit test allows to check for the statistical validity of the model (Beck and Arnold [1977]). Let us consider the multiresponse case. Boldfaced $\mathbf{Y}$ and $\mathbf{X}$ indicate matrices that include all measurements and all dimensions of the variables. Assume that a linear model, $Y = \beta X$, with X a vector of independent variables, and β the relation constants, is fitted to a set of data. Assume that measurement errors are additive and are distributed randomly with zero mean, and that the covariance matrix of the residuals (ψ) is known (or a very good approximation can be calculated based on model and data). Let

$$R = (\mathbf{Y} - \mathbf{X}b)^T \psi^{-1} (\mathbf{Y} - \mathbf{X}b). \tag{3.4.13}$$

It can be shown that R has a χ^2 distribution with degrees of freedom equal to the difference between the number of measurements and the number of parameters. R can be too small or too large to have a certain confidence, and for the model

to be valid with respect to the data with a confidence level α,

$$\chi^2_{1-\frac{\alpha}{2},n-p} \leq R \leq \chi^2_{\frac{\alpha}{2},n-p}. \tag{3.4.14}$$

Better fits correspond to more probable R's. The difference in probability may indicate a significant difference in goodness of fit. If not, both models show equivalent fits to the data. If ψ is not known with sufficient accuracy, an F test should be used, since an F test assumes that ψ is only approximately known.

3.4.3.2.5. F Tests

Various tests use ratios of different sums of squares as test statistics. Under assumptions of linearity of the model in parameters and data, and zero mean normal errors in the measurements, many of these ratio statistics can be shown to have F distributions.

Lack of Fit

Consider a data bank that contains n measurements. The independent variables are denoted as X, and the response variable is Y. The variables are collected at m ($m \leq n$) different combinations of X : X_1, X_2,...,X_m. At X_j, n_j different measurements Y_{ju} are taken. Therefore,

$$n = \sum_{j=1}^{m} n_j. \tag{3.4.15}$$

The model contains p estimated parameters. The *residual sum of squares* is defined as :

$$\sum_{j=1}^{m}\sum_{u=1}^{n_j}(Y_{ju} - \hat{Y}_j)^2 \tag{3.4.16}$$

and has (Draper and Smith, [1981])

$$n_e = \sum_{j=1}^{m}(n_j - 1) = \sum_{j=1}^{m} n_j - m = n - m \tag{3.4.17}$$

degrees of freedom. The statistic

$$F_{GF} = \frac{\sum_{j=1}^{m} n_j(\hat{Y}_j - \bar{Y}_j)^2}{\sum_{j=1}^{m}\sum_{u=1}^{n_j}(Y_{ju} - \bar{Y}_j)^2} \cdot \frac{n_e}{n-p-n_e} = \frac{\sum_{j=1}^{m} n_j(\hat{Y}_j - \bar{Y}_j)^2}{\sum_{j=1}^{m}\sum_{u=1}^{n_j}(Y_{ju} - \bar{Y}_j)^2} \cdot \frac{n-m}{m-p} \tag{3.4.18}$$

follows an F distribution with $n-p-n_e$ and n_e degrees of freedom. Its value can be compared to the F value with identical degrees of freedom and appropriate confidence level. This test allows to check the ratio of lack of fit to pure spread of data. A good ratio corresponds to a good fit. The test can only be done on a data set with repeated measurements.

Overall Regression Equation Test

Using :

$$F_{OR} = \frac{\sum_{j=1}^{m}\sum_{u=1}^{n_j}(Y_{ju}\hat{Y}_j) - \frac{1}{n}\left(\sum_{j=1}^{m}\sum_{u=1}^{n_j} Y_{ju}\right)^2}{\sum_{j=1}^{m}\sum_{u=1}^{n_j}(Y_{ju} - \hat{Y}_j)^2} \cdot \frac{n-p}{p-1}, \tag{3.4.19}$$

one can check statistical significance of the regression. Roughly speaking, this means that the proportion of the variation observed in the data is greater than would be expected by chance only. F_{OR} follows an F distribution with $p-1$ and $n-p$ degrees of freedom. F_{OR} is related to R^2 :

$$R^2 = \frac{(p-1)F_{OR}}{(p-1)F_{OR} + (n-p)} \tag{3.4.20}$$

but has a more generally known distribution.

Extra, Partial and Sequential F Tests

The extra sum of squares F test compares the increase in the ratio between the regression sum of squares to s^2 ($s^2 = \frac{1}{n-p} \cdot \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$). If this ratio is significant (with certain confidence), then the additional term is worth adding. This is a test on terms.

Partial and sequential F tests (also called F to enter and F to remove tests) compare changes in residual sum of squares to s^2 when a parameter is removed

or added to the model. These tests can be very useful in model building. Similar tests can check for hypotheses of (linear) relationships between parameters. These tests are tests on a parameters.

3.4.3.2.6. Optimal Design of Experiments

This section on model updating is written under the assumption that enough excitation is available to collect sufficient steady state results to obtain a reasonably accurate model parameter estimate. If this is not the case, then steady state disturbances may have to be introduced. The chances that the introduction of disturbances will be accepted by the operating plant are often small, and it makes sense to design the disturbances so that a maximal amount of information can be obtained from a minimal number of new steady states. Theory for optimal design of experiments can be applied in order to obtain new setpoints that will result in a new steady state with a maximal amount of new information. Methods for optimal design of experiments can be found in e.g. Beck and Arnold [1977].

3.5. THE OPTIMIZATION PROBLEM AND ITS SOLUTION

3.5.1. Introduction

The optimization routine does the conceptually most important task within the on-line optimizer scheme. During the optimization, information from the model identification, the plant optimization inputs and product market information are combined with the description of the optimization problem to obtain new setpoints that will drive the system performance to its extreme.

The optimization task is of extreme importance, and the optimization problems that have to be solved can be large, complex and can be difficult to solve. Nevertheless, the optimization problem solution is not expected to be the principal problem one has to solve in order to make an application of on-line optimization successful. From the literature referenced in Chapter 2, it can be concluded that the modeling is the most critical factor in most applications. Optimization is one of the most important numerical techniques, and intensive research in optimization techniques resulted in several powerful and robust optimization algorithms.

In this section on the optimization problem and its solution, no new optimization techniques will be presented. Instead, some existing optimization algorithms will be introduced that are more suited to the situation in an on-line optimizer than others. First, some aspects of the formulation of the optimization problem are discussed and first and second order conditions for optimality are given.

3.5.2. The Optimization Problem

3.5.2.1. With or Without Integer Optimization Variables

The general optimization problem (P) is formulated as follows :

$$(P)\begin{cases}\min_{(x,y)\varepsilon E^n\times\mathbb{N}^l} f(x,y,\varepsilon) & \\ \text{subject to} & \begin{cases} g(x,y,\varepsilon)\geq 0 \\ h(x,y,\varepsilon)=0 \end{cases}\end{cases} \tag{3.5.1}$$

with :

$$f : E^n \times \mathbb{N}^l \times E^k \rightarrow E^1; \tag{3.5.2}$$

$$g : E^n \times \mathbb{N}^l \times E^k \rightarrow E^m; \tag{3.5.3}$$

$$h : E^n \times \mathbb{N}^l \times E^k \rightarrow E^p. \tag{3.5.4}$$

Usually, $E = \mathbb{R}$. The objective f is generically the operating cost, or the profit (in that case maximized) of the system that will be optimized. x and y are steady state mean values for the setpoints and simulation states. The vector ε contains parameters such as model parameters, but it also includes the "driving forces" such as load requirements etc. Included in the steady state models are the constants in the control and data acquisition and control that conceivably influences the steady state. A quantity with an extremum that is equal to or approximates the minimum of the operating cost can be used instead. This quantity can for instance be the energy consumption per pound of produced product, if the energy required to generate this pound of product represents the main cost of the production. Constant factors such as fixed costs for equipment and labor can be subtracted from the objective. Only the variable cost is of interest. Constraints are introduced to make the solution meet requirements set by safety, material and equipment limits, product specifications, controllability, environment regulations, etc. Constraints are also used to introduce driving forces such as the required production rate. The steady state model is usually introduced as a set of constraints, but in certain cases it can also be built in the objective function. Values of ε are given and represent plant input, the "driving force" behind an on-line optimizer, market information (product and

raw material prices) and steady state model parameters that are updated by the model identification routine.

In general the problem is a mixed integer nonlinear program (MINLP). However in many cases the optimization problem will be a special case of the general problem (P). Integer optimization variables usually correspond to the on/off status ($y_i = 0$ or 1) of particular equipment. The on/off status of equipment is not always included in a closed loop energy consumption optimization. For instance, Poje and Smart [1986] do not consider equipment on/off status as on-line decision variables in their discussion of closed loop optimization. On the other hand, Nath *et al.* [1986] include the on/off status of boilers, turbines, pumps, compressors, etc. in the list of on-line optimization variables. If the on/off status of a piece of equipment is included in the array of the on-line optimized variables, then one must realize that it can be switched on and back off (or vice versa) by two consecutive optimization runs. The piece of equipment that has been switched on and off must at least reach the on or off state from the other within a time period that is relatively short compared to the time interval between two consecutive optimization runs. If one optimizes a number of fast units, and the on-line optimizer is executed every few minutes, it becomes very hard to allow any on/off variables as optimization variables. Only very fast equipment parts like a small pump or a small electric heating resistance can be switched on and off for a few minutes. Switching in or out of larger and slower equipment pieces can be suggested to the operator, together with sensitivity information (see section 3.6) on the incentive to do so. If, however a number of large and slow units is monitored and optimized at a frequency of once per several hours, then larger equipment pieces like a waste heat reboiler can be turned on and off on-line. Strictly speaking, one could take an integer optimization variable out of the array of optimization variables for a predeter-

mined number of optimizer executions. But then the information supplied to the optimizer routine in the ε array pertaining to this integer variable under consideration, has to be filtered in such a way that the decision concerning the on/off status of that integer variable only corresponds to sufficiently long term information. It is not clear how this type of data processing will influence the rest of the optimization variables. In this study, we will not deal with integer optimization variables.

The optimization problems that are mainly considered here are nonlinear programs (NLP). It is of course possible that the objective as well as the constraints are linear, and in that case the optimization problem is a linear program (LP).

3.5.2.2. Alternatives to (P)

In section 3.5.2.1. the general problem (P) was presented, and this problem contained two types of constraints, g : inequality constraints, and h : equality constraints. Most research in optimization problem formulation is done in engineering optimization-based design, and in the following sections three concepts from design optimization are introduced to on-line optimization.

3.5.2.2.1. Optimization under Uncertainty, Robustness

In practical examples many of the constraints introduced will be "uncertain". That means that they may contain constants that are uncertain. This uncertainty is specifically present in steady state model equations that are used as constraints in the optimization. In on-line optimization, the design variables are already fixed, and only the set of optimization variables is determined. The constraint uncertainty (as well as the objective uncertainty for that matter) is not.

The uncertainty is introduced by incorporating an uncertain right hand

term to the constraints. Of course, these uncertainties are only a very specific class of uncertainties. The inequality constraints are then written as $g_i(x) \leq \varepsilon_i$, and the equality constraints as $h_j(x) = \varepsilon_j$. It is also accepted that the uncertain ε_i varies within an interval T_i, defined by

$$T_i = \{\varepsilon_i | \varepsilon_{mi} \leq \varepsilon \leq \varepsilon_{Mi}\}, \tag{3.5.5}$$

then a multiperiod problem formulation presented by Grossmann *et al.* [1983] and Grossmann and Halemane [1982] guarantees that the solution will meet all constraints at any combination of extreme values for ε_i, $\forall i$. Simplifying, one could say that N values in the intervals T_i are chosen, corresponding to different operating techniques, and the combination of the performance for these chosen operating techniques is optimized. The resulting problem has a very specific structure which allows a very efficient decomposition. Still, this approach results in N subproblems to be solved, resulting possibly in too large a computational burden.

An earlier, and more straightforward technique which is sometimes referred to as the *stochastic approach*, works with the most probable values, and minimizes the *expected* operating cost (Grossmann *et al.* [1983]). Distributions were assumed for the uncertain parameters. It is shown in sections 3.6.4.2 and 3.6.4.3 that based on results from optimization solution sensitivity analysis, one can obtain an approximation of the variance of the expected optimal operating cost. Too large a variance of this optimal expected operating cost indicates that the optimizer does not have accurate enough information to work effectively. Implementation of the results does not guarantee near-optimal performance any more under such conditions. Constraints may be violated significantly ! This means that the loop control, ultimately responsible for maintenance of critical constraints, will have to act to meet those constraints, without guidance of the control. For more details, see sections 3.6.4.2. and 3.6.4.3. If no use is made

of the multiperiod technique, then, in general, due to noise and plant model mismatch, equality constraints that are imposed by operational considerations (e.g. sum of the loads over a number of parallel units equals a preset total load value) will be violated. Therefore, the loop control *must* absolutely be able to keep operational constraints (equality and inequality) from being violated. For instance, the optimization result should be overridden if a maximum pressure or temperature is exceeded.

3.5.2.2.2. One, Many or No Objectives

In the general optimization problem (P), the objective f was described as a function $E^n \rightarrow E^1$, characterizing the optimization objective as a *single* function. However, many practical optimization problems (e.g. optimization-based design) have more than one objective. In that case, a trade-off between these objectives has to be made, while meeting the constraints. How will this trade-off be made ? The literature on this subject is extensive. Good reviews on the subject, including over four hundred sixty references to previous publications, is presented by Hwang *et al.* [1979] and Hwang *et al.* [1980]. The solution to a problem with multiple objectives is a set of noninferior points, or the *pareto optimal* surface of the feasible set. A noninferior point corresponds to a combination of optimization variable values such that none of the objectives can be improved without hurting another. Clarck and Westerberg [1983] note that a weighted norm can discover any noninferior point for a convex feasible set. The multiple objective can then be reduced to a single objective :

$$\left(\sum_i (w_i f_i)^p\right)^{\frac{1}{p}}, \tag{3.5.6}$$

with $1 \geq p \geq +\infty$. The positive weights determine which point in the Pareto optimal set will be found. The weights are not necessarily constants, but may be "score functions". In the nonconvex case, only the infinity norm ($p = +\infty$)

will guarantee that all noninferior points can be found (Clarck and Westerberg [1983]). The problem then becomes a min max problem, in which the objective which has to be minimized, is :

$$\Big(\max_i w_i f_i(x, y, \varepsilon)\Big). \tag{3.5.7}$$

The weights or scaling are of extreme importance because they determine which noninferior point will be found. Nye [1983] addresses the problem of scaling, and proposes a *satisfaction / dissatisfaction* rule. The objectives are scaled based upon a good and a bad value for every objective, in such a way that corresponding good values should provide the same level of "satisfaction", while achieving the same bad values should reflect the same level of "dissatisfaction". The scaled objective f_i' is given by :

$$f_i' = \frac{f_i - good_i}{bad_i - good_i}. \tag{3.5.8}$$

This scaling technique is compared by Nye [1983] with some other scaling techniques, and Nye concluded that these scaling techniques are inferior to the proposed *satisfaction / dissatisfaction* rule. Garcia and Prett [1986] suggested to use this scaling technique in real time optimization.

A number of processes are so difficult to operate, that most satisfaction can be found in merely meeting the product specifications. For instance, the properties of certain polymer materials have to meet strict conditions to be of any value to the customer. The commercial value of the material is virtually zero if the material does not meet all the property requirements. The main task of a control system is to keep the product within these requirements. Optimization within the feasible set becomes less important, or even unimportant. It may even become so unimportant, that an optimization objective does not have to be specified. In that case the optimization routine can be replaced by an algorithm that looks for a feasible point. The rest of the on-line optimizer remains

the same, except for the sensitivity analysis, which cannot be performed, and consequently the accuracy check cannot be performed either.

3.5.2.2.3. Soft and Hard Constraints

The optimization problem (P) has equality and inequality constraints. A solution of (P) meets these constraints. The problem with this concept is that it is very rigid. There are constraints that can be set at exact values, for instance minimal and maximal values for the optimization variables, due to constraints in the control. However, many constraints are *fuzzy* in practice, but based upon engineering experience, insight and sound judgment one has to pick a breakpoint. Under these conditions, a rigid constraint may not be a realistic way to portray the optimization problems. In design, this problem is important, and is addressed by Nye and Tits [1983]. They propose a new methodology in which, instead of two types of specifications (objective(s) and constraints), three types of specifications are used to define an optimization problem. These are : objective(s), *soft* constraints and *hard* constraints. Nye and Tits [1983] define these three categories of specifications as follows :

(i) An objective is a quantity that has to be extremized; maximized or minimized according to specification.

(ii) A soft constraint is a quantity that one would like to keep below or above a preset value. If this is impossible, then the constraint should at least be attempted to be pushed close to this preset value, in a trade-off with other soft constraints and/or objective(s).

(iii) A hard constraint is a quantity that has to be below or above a preset value in order for the result to be of any use at all. A violation is unacceptable.

The soft constraints can be reformulated as a competing objective. The optimization problem then becomes an optimization problem with multiple objectives (see 3.5.2.2.2). The objective contributed by soft constraint g_i will be

formulated as $\min(0, g_i)$. These new objectives have to be scaled, and the fact that scaling is needed, is not surprising since soft constraints may have to be traded-off with the objective(s).

3.5.3. The Optimization Solution

In this section, the general nonlinear program (NLP) is considered.

$$(P)\begin{cases} \min_x f(x) \\ \text{subject to} \quad \begin{cases} g(x) \geq 0 \\ h(x) = 0 \end{cases} \end{cases} \tag{3.5.9}$$

First and second order conditions are listed below without proof. For a proof of these theorems; see e.g. Luenberger, Chapter 10 [1984].

3.5.3.1. First Order Necessary Condition

Let $B(x) = \{i | g_i(x) = 0\}$. A *regular* point of the constraint set of (P) is a point for which all constraints are met (a feasible point), and for which the gradients $\nabla h_i(x)$, $\nabla g_j(x)$ with $j \epsilon B(x)$ are linearly independent.

If $x^\star$ is a regular point for the constraints of (P), and a strict local minimum of (P), then

$$\exists \lambda \epsilon E^m, \ \exists \mu \epsilon E^p, \ \text{with } \mu \geq 0 :$$

$$\nabla f(x^\star) + \lambda^T \nabla h(x^\star) - \mu^T \nabla g(x^\star) = 0^T, \tag{3.5.10}$$

$$\mu^T g(x^\star) = 0^T; \tag{3.5.11}$$

with m the number of equality constraints and p the number of inequality constraints.

3.5.3.2. Second Order Necessary Condition

If f, g, $h \in C^2$, and if $x^\star$ is a regular point of the constraints of (P), as well as a strict local minimum of (P), then there is a set of multipliers

$$\exists \lambda \epsilon E^m, \ \exists \mu \epsilon E^p, \ \text{with } \mu \geq 0 :$$

such that

$$\nabla f(x^\star) + \lambda^T \nabla h(x^\star) - \mu^T \nabla g(x^\star) = 0, \tag{3.5.12}$$

$$\mu^T g(x^\star) = 0, \tag{3.5.13}$$

(in other words, the first order necessary condition is met) and,

$$\nabla^2 L(x^\star) = \nabla^2 f(x^\star) + \lambda^T \nabla^2 h(x^\star) - \mu^T \nabla^2 g(x^\star) \tag{3.5.14}$$

is positive semidefinite on the *tangent subspace* of the equality and active inequality constraints at $x^\star$. λ and μ are Lagrange multipliers.

The tangent subspace is a set defined as follows :

$$M(x^\star) = \{y | \forall i : \nabla h_i(x^\star) y = 0; \text{and } \forall j \epsilon B(x^\star) : \nabla g_j(x^\star) y = 0\}. \tag{3.5.15}$$

This means that every vector that lies within the plane perpendicular to the gradients of all the equality and active inequality constraints, belongs to the tangent subspace. The two dimensional equivalent of the tangent subspace is the tangent line at a point of a curve.

3.5.3.3. Second Order Sufficiency Condition

It is again assumed that f, g and h ϵC^2. If for $x^\star$,

$$\exists \lambda \epsilon E^m, \exists \mu \epsilon E^p, \text{such that :}$$

$$\nabla f(x^\star) + \lambda^T \nabla h(x^\star) - \mu^T \nabla g(x^\star) = 0 \tag{3.5.16}$$

$$\mu^T g(x^\star) = 0 \tag{3.5.17}$$

and the Hessian

$$\nabla^2 L(x^\star) = \nabla^2 f(x^\star) + \lambda^T \nabla^2 h(x^\star) - \mu^T \nabla^2 g(x^\star) \tag{3.5.18}$$

is positive definite on the tangent subspace

$$M(x^*) = \{y | \forall i : \nabla h_i(x^*)y = 0; \text{and } \forall j \epsilon B(x^*) : \nabla g_j(x^*)y = 0\}. \qquad (3.5.15)$$

then x^* is a strict local minimum of (P).

3.5.3.4. Convergence

Consider the solution of an optimization problem with n optimization variables. Excluding the degenerate case, there are two possible types of solutions :

- a first case in which there are n active constraints,
- and a second in which there are less than n active constraints.

In the first case, only absolute values of objective and constraints are necessary. Gradients of the optimization problem specifications are not important, because the solution is entirely determined by values of constraints. Therefore, if the estimation can identify the model such that the absolute values of the model optimization specifications have correct values at the optimization problem solution, then the on-line optimizer will converge to the true solution, even if (small) errors are present in the gradients of the objective. In the second case, there are fewer constraints active than the are degrees of freedom. Therefore there is some sort "hill top" solution. Roberts and Roussias [1980] mention that in this case, not only the absolute values but also the derivatives of the objective function have to be known exactly for unbiased convergence to the true optimum. Roberts and Roussias [1980] suggest a method to overcome this problem. This method adjusts the optimization objective to take discrepancies between the model and plant gradients into account.

Consider the unconstrained optimization problem :

$$\min_x f(x, \varepsilon). \qquad (3.5.19)$$

This problem formulation could correspond to an optimization problem for which at the optimum less constraints are active than there are degrees of free-

dom, and the active constraints are substituted into the optimization objective. In the optimization problem presented above, u is a vector of optimization variables, and p is a vector of optimization parameters. An estimate of the vector p is determined by the estimation module. It is found by minimizing the sum of squares of differences between the vector of model outputs $y(v, \alpha)$ and the vector of measured outputs $\hat{y}(v)$ with respect to α. The parameter estimation solution is given by $p = \alpha^*$. The optimization problem is transformed to :

$$\min_x \Big(f(x, \varepsilon) - \lambda x\Big), \tag{3.5.20}$$

with λ given by :

$$\lambda = \Big(\frac{\partial y}{\partial v} - \frac{\partial \hat{y}}{\partial v}\Big)^T \Big(\frac{\partial y}{\partial \alpha}\Big)^{-1} \frac{\partial F}{\partial \varepsilon}. \tag{3.5.21}$$

This term compensates for the difference between the model gradients and the true plant gradients. In practice, gradients of the true plant behavior with respect to its setpoints are very difficult to obtain. They could be obtained by perturbation of the plant. This does not mean that the described method is equivalent to evolutionary operation, since also the model leads the search for the optimum. Nevertheless, the approach presented by Roberts and Roussias [1980] allows to improve the results if the discrepancy between plant and model gradients are significant. Roberts and Roussias [1980] illustrate the use of this correction term on the rigorous dynamic simulation of a pressurized exotherm chemical reactor system. The steady state model used by the optimizer simplifies the reaction scheme. In the example presented by them, the profit return of the optimization without adjustment is roughly half of the profit return of the optimization with adjustment. A strict condition for the implementation of this adjustment term is that the disturbance for which the on-line optimizer corrects is so slow and the noise level is so low that there is enough time and accuracy to effectively calculate derivatives of the process with respect to the setpoints several times during a convergence to the *true* optimum.

3.5.4. The Optimization Algorithm

3.5.4.1. Choice of the Algorithm

The optimization algorithm should be selected based upon the size and characteristics of the optimization problem. If the optimization problem is linear, then of course a linear program solver should be chosen. In general, nonlinear programs are considered. The optimization problems that can be expected in applications of on-line optimization are large problems, with numerous constraints (inequality and equality) and optimization variables. Therefore the optimization algorithm that is selected should be able to solve large nonlinear constrained problems, and should preferably have good convergence properties. More specifically, a global convergent routine is preferred (convergence to a local minimum from every starting point), and a sufficient rate of convergence close to the optimum should be a requirement also. Very often, the constraint Jacobian will be sparse. The many connectivity equations and constraints pertaining to only very specific parts of the plant are the reason for this. Therefore, it would be advantageous, in large applications, to use an optimization algorithm that uses these sparse Jacobians in an efficient way. Finally, the method should be an *infeasible path* method. The operating staff using an on-line optimizer is not interested in intermediate results, and therefore they don't have to be feasible. By working along an infeasible path, the optimization and the plant model equations can be converged together. One has to keep in mind that "somewhere along" this infeasible path, a physically unrealistic combination of optimization variables can be applied to the constraint equations. These equations should nevertheless be defined for any value of the optimization variables. Therefore, checks have to be built in so that the routine will not attempt to divide by zero, take logarithms and/or square roots of negative numbers, and other undefined

numerical manipulations.

Renfro [1986] analyzed the applications and performance of nonlinear optimization algorithms in the frame of the requirements that were listed above. He compared in detail the generalized reduced gradient (GRG) method, methods based upon an augmented Lagrangian, successive linear programming (SLP) and successive quadratic programming (SQP). Methods based upon the augmented Lagrangian have shown their usefulness in various applications. Biggs [1978] considers the augmented Lagrangian methods under certain conditions preferable to SQP. The SLP methods do not use any second order information at all, and the rate of convergence close to the optimum is therefore slow. Applications of on-line optimization using SLP have been reported (Sourander *et al.* [1984]). However, optimization experts seem to prefer SQP. In general, it can be expected that some of the constraints correspond to model equations, and can be expensive to calculate. Under these conditions SQP is preferable over any of the other methods. Therefore, the remainder of this section on optimization will concentrate more on SQP than on any of the other methods.

In the following paragraphs, the SQP method will be discussed without detail. References indicate where more information can be found. Augmented Lagrangian methods and penalty function methods are touched upon next.

3.5.4.2. Successive Quadratic Programming (SQP)

Solving nonlinear programs using quadratic subprograms has probably first been proposed by Murray [1969] and Biggs [1972]. It soon became clear that the formulation of the quadratic subprogram was a critical issue. Biggs [1975] notes that using second order information (Hessian) from the Lagrangian can be advantageous to using second order information of the objective function only. Biggs [1975] suggested the use of the Broyden Fletcher Shanno (BFS) formula to update the Hessian in the quadratic subproblem. Powell [1977] prefers the

use of the Broyden Fletcher Goldfarb Shanno (BFGS) formula for updating the Hessian of the Lagrangian and shows that a *superlinear* rate of convergence is usually obtained, even when the true second derivative matrix of the Lagrangian is indefinite. Powell [1977] addresses two important issues concerning the SQP method : the step size in the optimal direction of the subproblem and the possibility of a quadratic subproblem with an empty feasible set. Powell [1977] introduces an extra variable ξ to solve the problems with the occurrence of an empty feasible set, but Biegler and Cuthrell [1985] demonstrate that Powell's [1977] proposal does not always work. Biegler and Cuthrell [1985] do not use the line search Powell [1977] suggests because it does not guarantee global nor superlinear convergence, and they propose a search step that does have global and superlinear convergence.

The SQP algorithm used later in this work (boiler network example), was proposed by Biegler and Cuthrell [1985], and will be presented briefly below. Consider the following optimization problem :

$$\begin{cases} \min_x f(x), \\ \text{subject to} \quad \begin{cases} g(x) \geq 0; \\ h(x) = 0. \end{cases} \end{cases} \tag{3.5.22}$$

At every iteration i a quadratic program is formulated :

$$(QP_i(x_i, B)) \begin{cases} \min_d \nabla f(x_i)^T d + \frac{1}{2} d^T B d, \\ \quad \text{subject to} \quad \begin{cases} g(x_i) + \nabla g(x_i)^T d \geq 0; \\ h(x_i) + \nabla h(x_i)^T d = 0. \end{cases} \end{cases} \tag{3.5.23}$$

B is updated using the BFGS updating formula, based upon the Lagrangian. Powell [1977] introduces a subproblem to avoid empty feasible sets: $(\max(0, g(x_i)) = p$, such that $p_k = \max(0, g_k(x_i))$

$$(QP_i^{(\xi)}(x_i, B)) \begin{cases} \min_d \quad \nabla f(x_i)^T d + \frac{1}{2} d^T B d + \lambda \xi, \\ \quad \text{subject to} \quad \begin{cases} \max(0, g_j(x_i))\xi + \min(0, g_j(x_i)) \\ \quad + \nabla g(x_i)^T d \geq 0, \\ h(x_i)\xi + \nabla h(x_i)^T d = 0, \\ 0 \leq \xi \leq 1. \end{cases} \end{cases} \tag{3.5.24}$$

and Powell [1977] suggests a large negative value for λ, for instance -10^6. Biegler and Cuthrell [1985] show that this suggestion is unsatisfactory in some cases, as mentioned above. Instead, they let the QP solver calculate the minimal infeasibility, and then they set the maximum allowable violation of the original constraints higher (1%) than the minimum infeasibility. The QP solver they use, is presented by Gill and Murray [1978]. This QP solver is also very effective for large sparse quadratic optimization problems (Coleman [1984]). A sparse optimization problem is a problem with a sparse Hessian. Setting the allowable violation (or constraint violation allowance), guarantees the existence of a solution to the quadratic program. After solving the QP, a line search is done in the direction of the solution of the QP(d_i^*). The line search procedure proposed by Biegler and Cuthrell [1985] uses as line search function :

$$L^*(x, \lambda, \mu, \alpha) = f(x) - \lambda^T g(x) + \mu^T h(x) + \frac{\alpha}{2} || \max(0, g(x)), h(x)||^2. \quad (3.5.25)$$

3.5.4.3. Augmented Lagrangian

Renfro [1986], in his comparison of various multivariable constrained nonlinear optimization techniques, considered the methods based on the augmented Lagrangian as less attractive than successive quadratic problems for the type of problems that are being considered here. Nevertheless, it is an efficient group of techniques with various successful applications. Biggs [1978] compared the SQP method to methods based on augmented Lagrangians, and concluded that the methods using augmented Lagrangian should be favored if the dimensions are large and the problem functions are fairly *inexpensive* to calculate. SQP methods would be better on problems with functions that are more expensive to calculate, and on moderate to small size problems. There is no unanimous conclusion on this subject. An important disadvantage of the methods based upon augmented Lagrangians is the fact that no second order information on the

constraints is being used, since only second order information of the objective is used. The Hessian of the Lagrangian is used in the sensitivity calculations that are presented in the section on sensitivity analysis. If it is decided that a sensitivity analysis will be made, the Hessian of the Lagrangian will have to be calculated anyway. Then it may be better to use an optimization technique that calculates this second order information effectively. Methods that use matrices that converge to the Hessian of the Lagrangian while converging to the solution of the optimization problem (for instance by using a BFGS updating formula, see SQP) seem particularly attractive. The fact that the Hessian of the Lagrangian has to be calculated in any case if a thorough on-line sensitivity analysis is made, neutralizes the advantage of less computational effort of the methods based upon an augmented Lagrangian.

3.5.4.4. Penalty Function Methods

The potential use of a penalty function method should be mentioned. An important problem with penalty function methods using non-exact penalty functions (twice continuously differentiable penalty functions) is that the Hessian at a boundary solution (suppose there are r active constraints at the solution) has r eigenvalues with an absolute value that grow unlimited ($\rightarrow \infty$). The other eigenvalues approach a finite limit. The fact that the condition number of the Hessian of the Lagrangian at the solution tends to infinity is a problem that should be taken into consideration when an unconstrained optimization method is chosen to minimize the penalized criterion. Fiacco [1983] describes in detail how the penalty function optimization method and a sensitivity analysis can be combined in the most efficient way. Fiacco [1983] uses a combination of a logarithmic barrier function (inequality constraints) and a quadratic penalty function (equality constraints).

3.5.5. Distributed Numerical Effort : Optimization Layers

Findeisen *et al.* [1978] suggested a method for decentralizing plant optimization. In this method, a coordinator communicates with the different parts that are being optimized separately in order to obtain a global optimum. In essence, the coordinator determines lagrange multipliers for the connectivity constraints between the various units. The function of the coordinator requires iteration. In every iteration, the plant must reach a new steady state. Therefore, this method, in a restricted sense, requires *evolutionary operation* of the plant. That is a significant disadvantage. Therefore the method can only be applied if every subsection of the plant that is optimized reaches steady state relatively fast, and if the disturbances for which the on-line optimizer attempts to adjust are relatively slow. The method as it is suggested by Findeisen *et al.* [1978] also requires that duality holds. Tatjewski [1985] suggests an improved method that does not have this requirement. The authors publish simulation examples, but applications are not reported. In Chapter 2, it was concluded that many of the industrial applications of larger systems will apply one or the other technique to distribute the optimization effort. As confidence in real time optimization and more specifically in the plant wide models grows, more centralized approaches to on-line optimization can be expected. Coordination between partial optimizers is difficult and time consuming because it is an iterative process. Finally it has to be noted that the distribution technique that is suggested by Findeisen *et al.* [1978] is very similar to the distribution of the reconciliation effort Marro *et al.* [1981] suggests (section 3.3.3.3).

3.6. STABILITY AND SENSITIVITY ANALYSIS

3.6.1. Justification

An on-line optimizer automates control at the supervisory level, with the intention to keep the plant operation optimal with respect to a certain criterion and subject to a number of equality and/or inequality constraints. The core of this task is fulfilled by the optimization routine that was discussed earlier. By applying a numerical technique, such as e.g. SQP, an estimate of the setpoints that solve the optimization problem, can be found. However, this solution only will very often be insufficient in a practical context. An on-line optimizer should not be pictured as a "*black box*". Instead, it should almost be viewed as a plant unit that has to be operated itself. The operating staff conditions the performance of the on-line optimizer by defining the optimization problem, determining the steady state model used, setting the execution frequency etc. In order to determine these specifications properly, feedback from the on-line optimizer is sought. Sensitivity analysis offers very interesting information which is easily interpretable, with regard to the on-line optimizer performance. Analysis of the results of a sensitivity study reveals the limiting specifications. This section will present in detail how it is suggested that sensitivity analysis is applied in the context of an on-line optimizer.

The results of differential sensitivity analysis characterize the immediate neighborhood around the optimal point. This characterization serves several immediate purposes, as discussed by Mc Keown [1980]. First of all, the results of the optimization routine will usually be only an approximation of the true (local) optimum. Sensitivity analysis can give a clue to the potential consequences of the accuracy of the numerical optimization technique to the optimization result. This influence is of importance in on-line optimization. If insufficient

accuracy is available, the expense of the on-line optimizer development and installation make on-line optimization become non-profitable. A second application listed by Mc Keown [1980] refers to problems with more than one local optimum with comparable objective function values. If the global optimum is too sensitive to certain parameters, then a suboptimal solution with lower sensitivity may be preferred. This idea may have applications in a more advanced on-line optimization set-up and/or in some specific application. In general it is not important in on-line optimization. Finally, the influence of tightening or loosening a constraint, or adjusting a certain parameter in the optimization problem solution can be evaluated by a sensitivity analysis. Apart from these applications, some that are of peculiar interest in on-line optimization will be presented. In these applications sensitivity analysis results are combined with statistical information from the steady state model parameters estimation and/or the plant measurements.

As Fiacco [1983] notes, it seems that nonlinear optimization will not be a widespread tool for multi-variable decision making processes without sensitivity analysis. Sensitivity analysis in *linear* optimization has been widely used for some three decades (e.g. Shetty [1959], Holm and Klein [1984]). Sensitivity analysis has already found some applications in optimization-based design (see e.g. Dixon and Szegö [1980]). In the context of the general on-line optimizer scheme presented here, attention will mainly be devoted to sensitivity analysis in nonlinear programming. In general, a nonlinear steady state model is used, and objective as well as constraints can conceivably be nonlinear. The theory for sensitivity and stability analysis in nonlinear programming is detailed and powerful. Most of the theoretical results presented here were originally presented in the summary by Fiacco [1983] (also Fiacco [1980a]). Sensitivity analysis calculates the first and second order derivatives of the optimal value function with

respect to the problem parameters that are not optimization variables, as well as the first order derivatives of the optimization variables with respect to these problem parameters. Interpretation of these derivatives is straightforward, and will be demonstrated in the remainder of this section.

Important theorems of sensitivity analysis are listed first, followed by a short discussion on how sensitivity results can be obtained. Finally, applications of sensitivity analysis in the environment of an on-line optimizer will be discussed in detail, and its possible applications in combination with statistical results from the steady state model parameter estimation.

3.6.2. Elements of Stability and Sensitivity Analysis Theory

This summary is limited to nonlinear programming. In general, a nonlinear steady state model is used to evaluate the expected plant steady state behavior. Sensitivity analysis in linear programming or post optimal analysis is described in various publications and textbooks, e.g. Reklaitis *et al.* [1983], Cooper and Steinberg [1970] and Luenberger [1984].

3.6.2.1. Stability Analysis

This section deals with properties of stability of the optimization solution. The solution of an optimization problem is stable in ε if the the optimization variables and the lagrange multipliers of the solution are continuous functions of ε. It is not the intention to give a complete review of stability results in nonlinear programming in this section, but merely to characterize the results that are available. The material in this section on stability analysis is mainly obtained from Fiacco [1983]. Fiacco collected stability results in two classes. A first class is based on the analysis of the rate of change of the optimal value function and the solution set (differential stability). The second class is based

upon applications of the implicit function theorem. Some of the important theorems are listed here, without proof. More detail and insight are provided by Fiacco [1983, Sections 3 and 4 of Chapter 2], Zollezzi [1984], Klatte [1984], Fiacco and Hutzler [1982] and Zlobec *et al.* [1982].

Throughout this section, optimization problem (P) will mostly be considered :

$$(P)\begin{cases} \min_{x\epsilon E^n} f(x,\varepsilon), \\ \text{subject to :} \end{cases} \quad \begin{cases} g(x,\varepsilon)[:\, E^n \rightarrow E^m] \geq 0, \\ h(x,\varepsilon)[:\, E^n \rightarrow E^p] = 0. \end{cases} \tag{3.6.1}$$

∇, without subscript, denotes derivation with respect to x in this section. Furthermore, the index set of the active inequality constraints is defined by :

$$B(x) \stackrel{\text{def}}{=} \{i | g_i(x) = 0\}. \tag{3.6.2}$$

No assumptions are made initially concerning continuity or differentiability of the functions f, g and h.

First, the *Mangasarian Fromovitz constraint qualification* should be introduced. This constraint qualification holds at $x^\star \epsilon \mathbb{R}$ if :

- $\diamond$ $\exists z \epsilon E^n$: $\nabla g_i(x^\star) z > 0$, $\forall i \epsilon B(x^\star)$ and $\nabla h_j(x^\star) z = 0$ for $j = 1, \ldots, p$, and
- $\diamond$ $\{\nabla h_j(x^\star), j = 1, \ldots, p\}$ are linearly independent.

The Mangasarian Fromovitz constraint qualification can also be written as :

$$\sum_{i \epsilon B(x^\star)} u_i \nabla g_i(x^\star) + \sum_{j=1}^{p} w_j \nabla h_j(x^\star) = 0 \tag{3.6.3}$$

has no nonzero solution $u_i \geq 0$, w_j. It is interesting to note that the Mangasarian Fromovitz constraint qualification is sufficient for the Kuhn Tucker constraint qualification, which is a requirement for the validity of the Kuhn Tucker optimality conditions.

A first theorem that is presented here, was proven by Gauvin and Tolle [1977] and is a differential stability result. It is restricted to a special case of

(P) :

$$(P') \begin{cases} \min_{x \epsilon E^n} f(x) \\ \text{subject to} \end{cases} \begin{cases} g(x)[: E^n \rightarrow E^m] \geq \varepsilon_1 \\ h(x)[: E^n \rightarrow E^p] = \varepsilon_2. \end{cases} \tag{3.6.4}$$

The following definitions are introduced also :

$$R(\varepsilon) = \{x | g(x) \geq \varepsilon_1, h(x) = \varepsilon_2\}. \tag{3.6.5}$$

Furthermore :

$$f^*(\varepsilon) = \begin{cases} \inf_x \{f(x) | x \epsilon R(\varepsilon)\}, & R(\varepsilon) \neq \emptyset; \\ +\infty, & R(\varepsilon) = \emptyset. \end{cases} \tag{3.6.6}$$

Of course $\varepsilon = [\varepsilon_1 \quad \varepsilon_2]^T$ and $\varepsilon \epsilon E^k$ with $k = m+p$. Also, for $R(\varepsilon) \neq \emptyset$, a solution set is defined as :

$$S(\varepsilon) = \{x \epsilon R(\varepsilon) | f(x) = f^*(\epsilon)\}. \tag{3.6.7}$$

Gauvin and Tolle [1977] showed that if $R(0) \neq \emptyset$, with R *uniformly compact* near $\varepsilon = 0$, and if $\exists x^* \epsilon S(0)$ for which the Mangasarian Fromovitz constraint qualification holds, then f^* is continuous in ε at $\varepsilon = 0$.

Fiacco [1980b,c] showed that the above theorem holds for (P) if the objective and constraint functions have continuous first order derivatives in x as well as in ε. Some special cases are listed in Fiacco [1983]. A similar result was obtained by Robinson [1974], based on the implicit function theorem : if T is a Banach space, $T_0 \subset T$, $M \subset E^n$, with M and T_0 open sets, if f, g and h have second order partial derivatives with respect to x that are jointly continuous on $M \times T_0$, then for ε^* in T_0 and with $[x^* \quad u^* \quad w^*]$ the Karush Kuhn Tucker triple of (P), and assuming linear independence of the gradients of the active constraints, strict complementary slackness and second order sufficiency conditions holding at $[x^* \quad u^* \quad w^*]$, the following three facts are proven to be true. These fact are :

(i) $\exists y(\varepsilon)$, which is continuous, with $y(\varepsilon^*) = [x^* \quad u^* \quad w^*]^T$, $\forall \varepsilon$ in a neighborhood of ε^*, $y(\varepsilon)$ is the unique Karush Kuhn Tucker triple of (P);

(ii) $\forall\varepsilon$ near $\varepsilon^\star$, $x(\varepsilon)$ is an isolated local minimizing point of (P);

(iii) linear independence of the gradients of the active constraints, strict complementary slackness and the second order sufficiency condition hold for ε near $\varepsilon^\star$.

Similar theorems are based on the implicit function theorem, and various results are listed by Fiacco [1983]. The two theorems that were presented above have a local character. For a given problem (P), the stability of the solution is considered for a given ε value. The theory allows to check whether or not the solution of (P) is stable in an arbitrary small neighborhood of the preset value of ε. Results that indicate finite bounds over which ε in between the stability of the solution of (P) is guaranteed are rare, and are restricted to very specific forms of (P) and its functions f, g and h.

Stability can be checked off-line, and it may be useful to do so for values of ϵ that are expected to occur frequently. Sensitivity theorems will not hold at places where the solution is not stable. For some simple optimization problems it is possible to prove that the solution is always stable, or the unstable points can be detected.

3.6.2.2. Sensitivity Analysis

In this summary, the sensitivity theory that will be used later in this work, is listed. First, some basic sensitivity results are given. Then, the general theory that is applied in the on-line optimization scheme will be presented.

3.6.2.2.1. Basic Sensitivity Results

Basic Sensitivity Theorem

Fiacco [1983] formulated a *basic sensitivity theorem*, proving the existence of a once continuously differentiable local solution of (P), as well as unique and

differentiable Lagrange multipliers. It is given here without proof.

If :

(i) f, g and h are twice continuously differentiable functions in x and if their gradients with respect to x and the constraints are once continuously differentiable in ε *in a neighborhood of* $(x^\star, 0)$ (i.e. locally),

(ii) the second order sufficiency conditions hold at $[x^\star \quad u^\star \quad w^\star]$,

(iii) the linear independence assumption holds at $x^\star$, and

(iv) $\forall i | g_i(x^\star) = 0 : u_i^\star > 0$ (strict complementary slackness).

Then :

(a) $x^\star$ is a local isolated solution of (P) with unique Lagrange multipliers $u^\star$ and $w^\star$,

(b) $\forall \varepsilon$ near 0, $\exists!$ continuously differentiable function $y(\varepsilon) = [x(\varepsilon)y(\varepsilon)w(\varepsilon)]^T$ that satisfies the second order sufficiency conditions for (P), with $y(0) = [x^\star \quad u^\star \quad w^\star]^T$, hence $x(\varepsilon)$ is an isolated local minimizing point of (P) with unique Lagrange multipliers $u(\varepsilon)$ and $w(\varepsilon)$,

(c) $\forall \varepsilon$ near to 0, the gradients of the binding constraints are linearly independent, and strict complementary slackness holds for $u(\varepsilon)$ and $g(x, \varepsilon)$.

This theorem is the basis from which the general sensitivity results that are presented in the next section, are developed.

Basic Right Hand Side Optimal Value Function Sensitivity Theorem

Right hand side (RHS) sensitivity is a well known special case. Consider case (P') :

$$(P') \begin{cases} \min_{x \in E^n} f(x), \\ \text{subject to} \quad \begin{cases} g(x)[: E^n \to E^m] \geq \varepsilon_1; \\ h(x)[: E^n \to E^p] = \varepsilon_2. \end{cases} \end{cases} \tag{3.6.8}$$

If f, g and h are twice continuously differentiable, and let for $\varepsilon_1 = 0$ and $\varepsilon_2 = 0$ a local solution be given by $x^\star$ and assume that $x^\star$ is a regular point, and that

it, together with the Lagrange multipliers $u \geq 0$ and w, satisfies the second order sufficiency condition for a strict local minimum, then $\forall\ \varepsilon_1$ and ε_2 in a neighborhood around $\varepsilon_1 = 0$ and $\varepsilon_2 = 0$, there exists an optimization solution $x(\varepsilon_1, \varepsilon_2)$, continuous in $(\varepsilon_1, \varepsilon_2)$ such that $x(0,0) = x^\star$, and such that $x(\varepsilon_1, \varepsilon_2)$ is a local minimum of (P'). Besides that :

$$\nabla_{\varepsilon_1} f^\star(x(\varepsilon_1, \varepsilon_2))\Big|_{(0,0)} = u^T; \tag{3.6.9}$$

$$\nabla_{\varepsilon_2} f^\star(x(\varepsilon_1, \varepsilon_2))\Big|_{(0,0)} = -w^T. \tag{3.6.10}$$

This result specifies that the first derivative of the change in the optimal value function as a result of tightening or releasing a constraint is equal to the Lagrange multiplier (+ or −), also referred to as the *shadow prices*. The sensitivity of the optimal value function for a move in an inactive constraint is of course zero. In the next section, more general results will be presented.

3.6.2.2.2. General Sensitivity Results

Results that are much more general than the RHS results presented above, are listed below, without proof.

First and Second Order Parameter Derivatives of the Optimal Value Function

Consider the general problem (P). Under the conditions of the *basic sensitivity theorem* that was presented above (3.6.2.2.1), and if the problem functions are twice continuously differentiable in (x, ε) near $(x^*, 0)$, (in the basic sensitivity theorem only continuous first derivatives in ε, together with continuous second derivatives in x are required) then, in a neighborhood of $\varepsilon = 0$:

$$\nabla_\varepsilon f^\star = \nabla_\varepsilon L \tag{3.6.11}$$

$$= \nabla_\varepsilon f|_\star - \sum_{i=1}^{m} u_i(\varepsilon) \nabla_\varepsilon g_i|_\star + \sum_{j=1}^{p} w_j(\varepsilon) \nabla_\varepsilon h_j|_\star, \tag{3.6.12}$$

and :

$$\nabla_\varepsilon^2 f^\star = \nabla_\varepsilon(\nabla_\varepsilon^T L) \tag{3.6.13}$$

$$= \nabla_{x\varepsilon}^2 L \nabla_\varepsilon x - \sum_{i=1}^{m} \nabla_\varepsilon g_i^T \nabla_\varepsilon u_i(\varepsilon) + \sum_{j=1}^{p} \nabla_\varepsilon h_j^T \nabla_\varepsilon w_j(\varepsilon) + \nabla_\varepsilon^2 L. \tag{3.6.14}$$

$$\left(L(x,u,w,\varepsilon) \stackrel{\text{def}}{=} f(x_\varepsilon) - \sum_{i=1}^{m} u_i g_i(x,\varepsilon) + \sum_{i=1}^{m} w_i h_i(x,\varepsilon)\right) \tag{3.6.15}$$

First Order Parameter Derivatives of the Kuhn Tucker Triple

Consider the general problem (P). Under the conditions of the *basic sensitivity theorem* that was presented above (3.6.2.2.1) :

$$\nabla_\varepsilon\{[x \quad u \quad w]^T\} = M^{-1}N \tag{3.6.16}$$

with :

$$M = \begin{pmatrix} \nabla_x^2 L & -\nabla_x g_1^T & \cdots & -\nabla_x g_m^T & -\nabla_x h_1^T & \cdots & -\nabla_x h_p^T \\ u_1\nabla_x g_1 & g_1 & & 0 & & & \\ \vdots & & \ddots & & & 0 & \\ u_m\nabla_x g_m & 0 & & g_m & & & \\ \nabla_x h_1 & & & & & & \\ \vdots & & 0 & & & 0 & \\ \nabla_x h_p & & & & & & \end{pmatrix}; \tag{3.6.17}$$

$$N = \left(-\nabla_{\varepsilon x}^2 L^T, -u_1\nabla_\varepsilon g_1^T, \cdots, -u_m\nabla_\varepsilon g_m^T, -\nabla_\varepsilon h_1^T, \cdots, -\nabla_\varepsilon h_p^T\right)^T. \tag{3.6.18}$$

$$\left(L(x,u,w,\varepsilon) \stackrel{\text{def}}{=} f(x_\varepsilon) - \sum_{i=1}^{m} u_i g_i(x,\varepsilon) + \sum_{i=1}^{m} w_i h_i(x,\varepsilon)\right) \tag{3.6.19}$$

Under the conditions of the basic sensitivity theorem, the inverse of M exists. Information on second order characteristics of the Lagrangian can for instance be obtained from the an SQP optimization routine.

Special Results

An interesting special result regards the unconstrained optimization problem ($\min_x f(x,\varepsilon)$). If $f(x,\varepsilon)$ is twice continuously differentiable in x and if $\nabla_x f(x,\varepsilon)$ is once continuously differentiable in ε, in a neighborhood of $(x^\star, 0)$, and if the second order sufficient conditions for an unconstrained local minimum of $f(x,0)$ hold at $x^\star$ then, for ε close to zero :

$$\nabla_\varepsilon f^\star = \nabla_\varepsilon f; \tag{3.6.20}$$

$$\nabla^2_\varepsilon f^\star = \left(\nabla^2_{x\varepsilon} f\right)\left(\nabla^2_x f\right)^{-1}\left(\nabla^2_{x\varepsilon} f\right)^T + \nabla^2_\varepsilon f. \tag{3.6.21}$$

Note that for an unconstrained problem the first derivatives with respect to ε of f and $f^\star$ are equal.

3.6.3. Applications of Sensitivity Analysis

In this section, fields of applications of sensitivity analysis information are presented. First, two off-line uses are specified, and then two on-line applications are formulated. The off-line applications can be used during preliminary studies and initial steps in the decision process concerning the feasibility of an implementation of an on-line optimizer. The on-line applications constitute feedback of the on-line optimizer to the operator staff.

3.6.3.1. Marginal Optimization Gain

Consider optimization problem (P).

$$(P)\begin{cases} \min_{x\epsilon E^n} f(x,\varepsilon), \\ \text{subject to} \quad \begin{cases} g(x,\varepsilon)[:\, E^n \rightarrow E^m] \geq 0; \\ h(x,\varepsilon)[:\, E^n \rightarrow E^p] = 0. \end{cases} \end{cases} \tag{3.6.22}$$

The optimization variables x correspond to setpoints of the plant that is the subject of the optimization, and ε are a set of parameters that influence the operation, but are not optimization variables. Elements of ε can be a feed temperature, the ambient temperature or the minimal concentration of a component in

a mixture, or for instance a model parameter. These variables typically change, and force the optimizer to adjust the setpoints in order to stay optimal in (P). The incentive to provide the necessary hardware and software to do on-line optimization, comes from the expected increase in performance. The increase in performance, in practice a better value of the objective, has to be sufficiently high to compensate for the efforts to install and operate the on-line optimizer. Sensitivity results allow to calculate an upper estimate of the increase in profit an on-line optimizer can bring, relative to a well operated, but not on-line optimized plant. A *necessary condition* for a plant to be an interesting candidate for on-line optimization is that this upper limit should be sufficiently high. This upper limit will be referred to as the *marginal optimization gain.*

Figure 3.6.1. gives a schematic presentation. The horizontal axis represents that operation condition, the vertical axis shows the performance. The on-line optimized performance is identical to the performance of the suboptimal operation policy under nominal conditions. This equality can be expected, and has to be met for a fair comparison between on-line optimized plants and plants without on-line optimization. The top curve gives the on-line optimized performance. It is possible that optimized performance is higher for conditions that are not nominal, since the on-line objective is not necessarily identical to the design objective. The area between both curves gives the incentive for optimization. Sensitivity analysis results are used to approximate that area.

Consider again (P). The first assumption that will be introduced here, is the assumption that (P) reflects the actual plant condition perfectly. In other words, it is accepted that the model is perfect and that the parameter estimation is exact. Furthermore, ε is known exactly, no noise or bias disturb the measurements. The increase in profit that is estimated from (P) is higher than realistic because model plant mismatch, measurement bias and noise, and

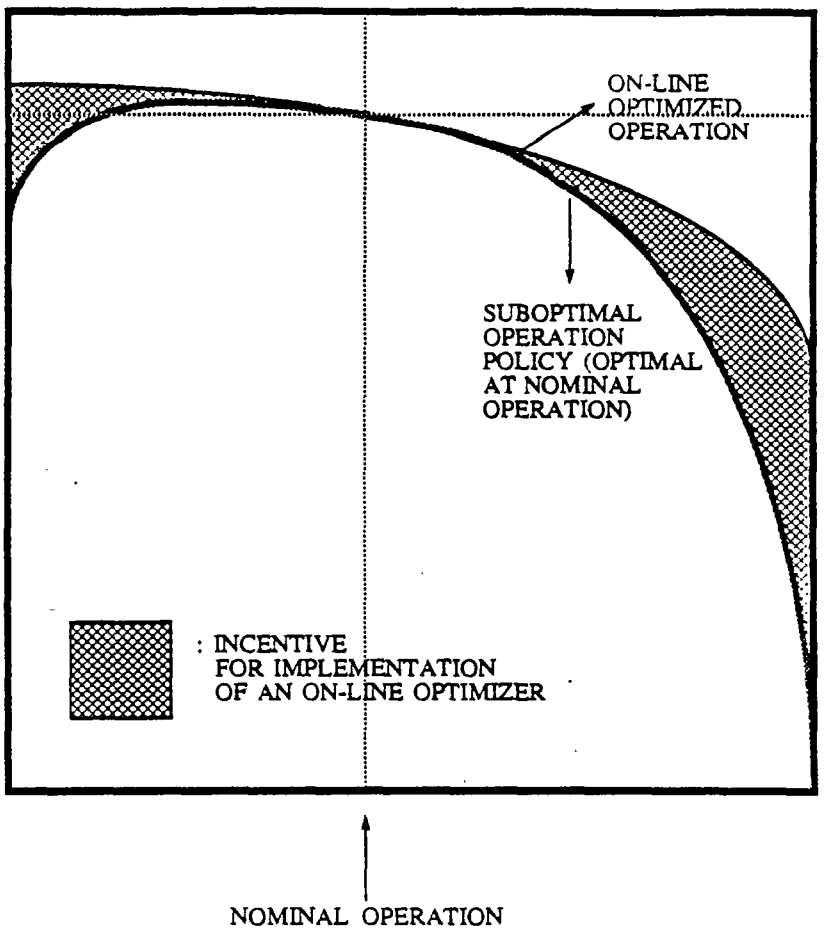

Fig.3.6.1. Schematic Representation of Objective Value Versus Operating Conditions for Optimal and Suboptimal Operation. The Abscissa schematically Represents the Range of Operating Conditions. The Ordinate Shows the Performance (e.g. Profit).

dynamic effects can be expected to reduce the performance of an on-line optimizer. Assume that the nominal operating condition corresponds to ε_0. Let $x_0^\star$ be the solution of (P) for ε_0. Assume also that the conditions of the theorem that gives general results on first and second order parameter derivatives of the optimal value function hold (paragraph 3.6.2.2.2). The theorem that gives general results on first order parameter derivatives of the Kuhn Tucker triple holds

also under these conditions (paragraph 3.6.2.2.2), as well as the basic sensitivity theorem(paragraph 3.6.2.2.1). In particular, assume that :

(i) f, g and h are twice continuously differentiable functions in x and that their gradients with respect to x and the constraints are once continuously differentiable in ε in a neighborhood of $(x^\star, 0)$,

(ii) the second order sufficiency conditions hold at $(x^\star, u^\star, w^\star)$,

(iii) the linear independence assumption holds at $x^\star$ and,

(iv) $\forall i | g_i(x^\star) = 0 : u_i^\star > 0$ (strict complementary slackness) holds.

Then, from the theoretical results that were listed in section 3.6.2, it is known that there exists for ε close to ε_0, a once continuously differentiable function $y(\varepsilon) = (x(\varepsilon), u(\varepsilon), w(\varepsilon))^T$, satisfying the second order sufficient conditions for local optimality for problem (P). The optimal value function can thus be written as $f(x(\varepsilon), \varepsilon) = f^\star(\varepsilon)$, also a once continuously differentiable function for ε close to ε_0. The performance of the on-line optimized plant under the ideal conditions assumed above will be equal to $f^\star(\varepsilon)$. A plant not on-line optimized will not have its setpoints updated. Therefore the performance of the non-optimized plant will be $f(x(\varepsilon_0), \varepsilon)$. The difference between $f^\star(\varepsilon)$ and $f(x(\varepsilon_0), \varepsilon)$ has to be large enough to make on-line optimization interesting. In general, an analytical form of $f^\star(\varepsilon)$ does not exist, but sensitivity calculations provide us with a second order Taylor expansion of the optimal value function $f^\star(\varepsilon)$, again for ε close to ε_0.

$$\begin{aligned} f^\star(\varepsilon) - f(x(\varepsilon_0), \varepsilon) \simeq & (\nabla_\varepsilon f^\star|_0 - \nabla_\varepsilon f|_{\star 0}) \cdot (\varepsilon - \varepsilon_0) \\ & + \frac{1}{2}(\nabla_\varepsilon^2 f^\star|_0 - \nabla_\varepsilon^2 f|_{\star 0}) \cdot (\varepsilon - \varepsilon_0)^2. \end{aligned} \tag{3.6.23}$$

Using results from paragraph 3.6.2 :

$$\nabla_\varepsilon f^\star|_0 - \nabla_\varepsilon f|_{\star 0} = -u^T \nabla_\varepsilon g|_{\star,0} + w^T \nabla_\varepsilon h|_{\star,0}; \tag{3.6.24}$$

$$\begin{aligned} \nabla_\varepsilon^2 f^\star|_0 - \nabla_\varepsilon^2 f|_{\star 0} = & \nabla_{x\varepsilon}^2 L|_{\star,0} \nabla_\varepsilon x|_0 - \nabla_\varepsilon^T g|_{\star,0} \nabla_\varepsilon u|_0 + \nabla_\varepsilon^T h|_{\star,0} \nabla_\varepsilon w|_0 \\ & + \nabla_\varepsilon^2 (-u^T g + w^T h)|_{\star,0}. \end{aligned} \tag{3.6.25}$$

The unconstrained problem is an interesting special case. For the unconstrained case :

$$\nabla_\varepsilon f^\star|_0 - \nabla_\varepsilon f|_{\star 0} = 0, \tag{3.6.26}$$

$$\nabla^2_\varepsilon f^\star|_0 - \nabla^2_\varepsilon f|_{\star 0} = -\left(\nabla^2_{x\varepsilon} f|_{\star 0}\right)\left(\nabla^2_x f|_{\star 0}\right)^{-1}\left(\nabla^2_{x\varepsilon} f|_{\star 0}\right)^T. \tag{3.6.27}$$

Note that in the unconstrained case, $f^\star$ and f have the same derivatives, and therefore that the derivative of their difference, the incentive for on-line optimization, is zero for ε equal to ε_0. In the more general case that no constraints with nonzero derivatives with respect to ε are active, $\nabla_\varepsilon f^\star|_0 - \nabla_\varepsilon f|_{\star 0} = 0$.

The formulas listed above allow one to calculate the marginal optimization gain, an upper limit to the potential increase in performance of a plant due to on-line optimization. As mentioned above, this calculation offers only an upper limit. Measurement noise and bias, plant dynamics, process loop control malfunctions, plant model mismatches and lack of steady state data can reduce the potential gain of an on-line optimization procedure. The condition determining a minimal value for the marginal optimization gain is a *necessary*, but not a sufficient condition for a cost-efficient on-line optimization application. It establishes an upper bound on the benefit.

This method can be refined by instead of simply comparing to the objective function, also taking into consideration that in the unoptimized case ("the reference line") the constraints still have to be met. This can be done by substituting a constraint in the objective, or by analytical techniques. McKeown [1980] discusses the behavior of optimization objectives in the neighborhood of an optimal point while meeting the constraints. Results from that study could be applied in the calculation of the marginal gain in optimization. A central issue is the definition of the steady state results of the loop control.

3.6.3.2. Off-Line Accuracy of the Expected Optimal Value

A large amount of information is used by the optimization. The optimization problem (P) combines all these sources and defines a solution which corresponds to an optimum value of the objective. However, all this information contains some uncertainty. There always is plant model mismatch, and measurements contain noise. These errors will result in a possible error in the solution $x^\star$, and in the optimum value $f^\star$. The higher the uncertainty is in $f^\star$ compared to the range over which the optimal objective function value is expected to vary as operating conditions change, the less attractive it becomes to implement on-line optimization. This principle can be used in a preliminary feasibility study. The combination of sensitivity analysis and statistical information offers a possibility to estimate the accuracy with which $f^\star$ approximates the true value of the objective. If the accuracy of $f^\star$ is low, it means that the quality of the data and/or the model are such that a good prediction of the optimal value function cannot be given. The confidence region around the estimate of the optimal value function should be small compared to the range over which the optimal value function can vary, in order to have an interesting candidate for on-line optimization. Reasonable accuracy in $f^\star$ compared to the operational envelope of $f^\star$ is a *necessary* condition, but not a sufficient condition for a good on-line optimization application. Figures 3.6.2–3 illustrate this.

In figures 3.6.2–3 the vertical axis shows the optimal value function. The vertical axis schematically presents operating conditions in both figures. The size of the rectangle symbolizes the range over which the objective function varies as operating conditions change from one extreme to the other. The accuracy with which the optimal value function is known, is given by the shaded band. In fig 3.6.2 this accuracy is fair, in figure 3.6.3 it is poor. Indicated is the minimal change in operating conditions or expected performance, to have a

significant change of performance with high probability. In figure 3.6.3, the operating conditions have to change over more than $\frac{3}{4}$ of the range for a significant change in performance. That means that for a change of operating conditions of less than three quarts of the total range, the optimized performance does (statistically) not change. Why optimize in such cases ?

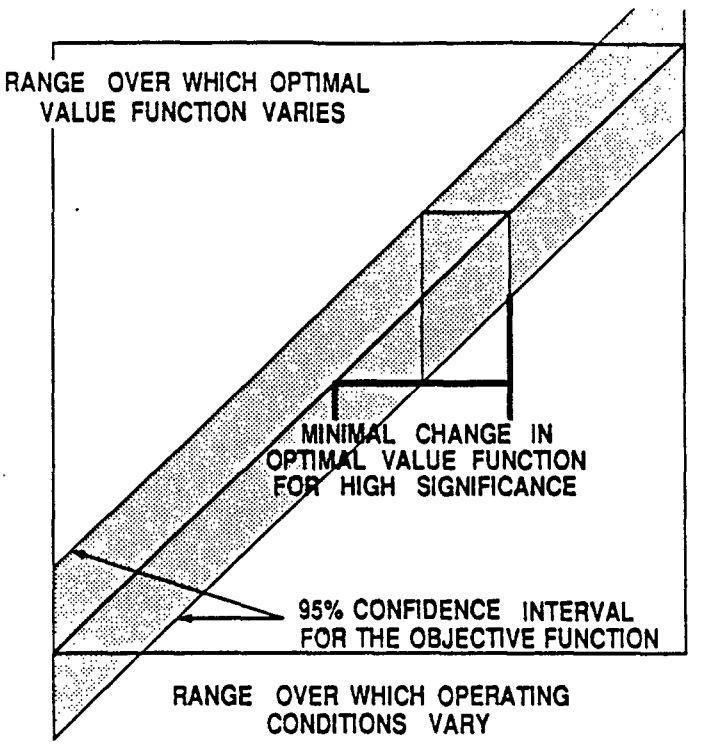

Fig.3.6.2. Performance (with 95% Confidence Limits) Versus Operating Conditions for Fair Accuracy.

The method that is proposed here for the calculation of the accuracy with which f^* is determinated, is based on the concept that the optimization result, the optimization variable and the variables determinating the optimization problem (such as model parameters and measured variables) are statistical variables. Cutler and Perry [1983] suggest to calculate the expected inaccuracy by simply multiplying the fractional inaccuracies of every operation (model fitting, measurement, etc.) that is involved in the on-line optimization operation. The

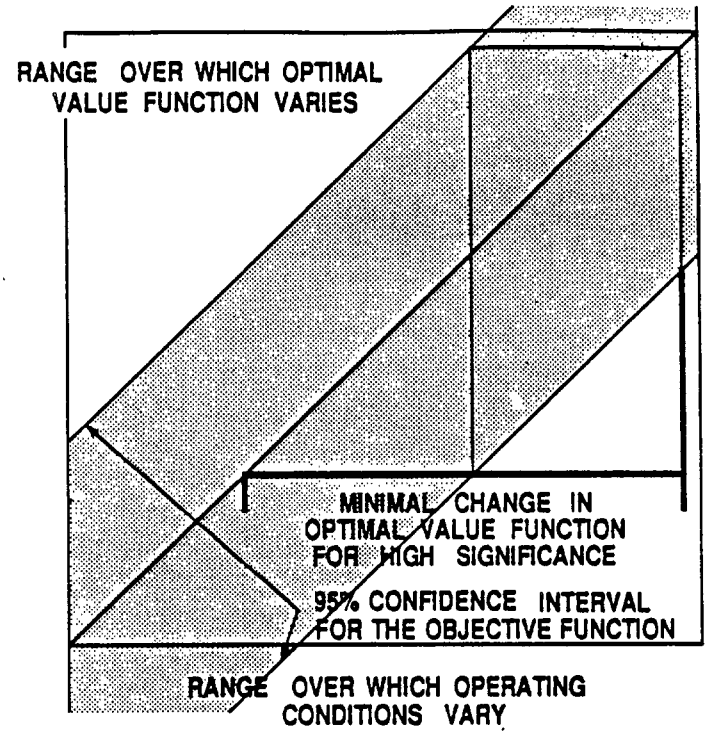

Fig.3.6.3. Performance (with 95% Confidence Limits) Versus Operating Conditions for Poor Accuracy.

approach that is suggested here is much more accurate, and will add variances instead of fractional inaccuracies. It also provides the possibility to take correlations between various variables determining the solution of the optimization solution into account.

Consider the objective function of problem (P). Let the nominal operating conditions correspond to ε_0. A first order approximation to the analytical solution of (P) is given by :

$$f^*(\varepsilon) \simeq f_0^* + \left(\nabla_\varepsilon f^*|_0\right)^T(\varepsilon - \varepsilon_0). \tag{3.6.28}$$

If the conditions of the general sensitivity theorem (paragraph 3.6.2.2.2) hold, then $\nabla_\varepsilon f^*|_0$ can be calculated using the formulas presented above, and the approximation is valid in a neighborhood around ε. The variables that are included in this vector ε are chosen by the user, and are of course known. But

as indicated earlier, the values that make up ε, which is a vector of the variables to which the optimization problem is more or less sensitive, are statistically not known exactly. Model parameters could be represented in the vector ε. Their values are estimates, known with a certain accuracy. The same is true for physical or market information that is included in the vector ε. Therefore, ε can be considered as a random variable, with a certain distribution. It is common to assume that ε has a multivariate normal distribution, with variance covariance matrix Q_ε. In an analogous way, $f^\star(\varepsilon)$ can be considered as the average of a normally distributed variable. The variance of this variable $f^\star$ is then given by :

$$\sigma^2_{f^\star} = \left(\nabla_\varepsilon f^\star|_0\right)^T Q_\varepsilon \left(\nabla_\varepsilon f^\star|_0\right). \tag{3.6.29}$$

It was mentioned earlier that the vector ε can contain various variables such as model parameters, load measurements etc. How to obtain $Q_{\varepsilon,ij}$ depends upon the character of the variables ε_i and ε_j. Many of the covariances (off diagonal terms) will be zero, and Q_ε can be expected to be a block diagonal matrix. The covariances of the model parameters can be obtained from the statistical analysis on the model parameter estimation. The variances of measurements have to be obtained from a series of data taken at the same steady state or from vendor information. Since Q_ε is not exactly known and therefore an approximation is used, it could be argued that $f^\star$ should be considered as following a t distribution. However, that is only the case if $\sigma^2_{f^\star}$ is obtained as an average sum of squares of differences. This is in general not the case. The number of degrees of freedom of the t distribution cannot be determined here. Therefore, the only alternative is to accept the assumption of a normal distribution. A 95% confidence interval is given by $1.96 \cdot \sqrt{\sigma^2_{f^\star}}$. This interval gives a measure of the accuracy with which the actual optimal value function is known. The information required to obtain this confidence interval for $f^\star_0$,

can be obtained from one run of the nominal case study and information on the accuracy with which ε_0 is known.

3.6.3.3. On-Line Accuracy of the Expected Optimal Value

The calculation of the accuracy of $f^\star$ as presented in the previous section does not require a large computational effort. The calculation can be done on-line, at every optimizer execution. The functioning of an on-line optimizer is only guaranteed under many assumptions. Most of these assumptions are idealizations and are therefore more or less violated in every practical example. It was indicated earlier that this may considerably reduce the expected gain of on-line optimization under ideal conditions. Important discrepancies between idealizations and practice are plant model mismatch, unsteady state data ("true" steady state does not exist), measurement noise, measurement bias that passes the gross error detector unnoticed (type II hypothesis test error) and data covariance wind-up (insufficient spread of steady state data). Plant model mismatch, departure from steady state and measurement noise and undetected bias result in an increased confidence interval around parameter estimate values. Data covariance wind-up combined with measurement noise and bias, and unsteady states can increase confidence intervals of model parameter estimates even more. Data covariance wind-up can also lead to model parameter values that are not realistic. This in turn results in optimization results that are very sensitive to the model parameter values. The accuracy of $f^\star$ will under those circumstances decrease substantially. If a sudden increase in the confidence interval of $f^\star$ occurs, or $\sigma^2_{f^\star}$ reaches a preset value, then implementation of the optimization results should be stopped. In other words, the on-line optimizer loop has to be broken, until $\sigma^2_{f^\star}$ is back below the preset value. The values of

the individual terms of

$$\sigma_{f^\star}^2 = \left(\nabla_\varepsilon f^\star|_0\right)^T Q_\varepsilon \left(\nabla_\varepsilon f^\star|_0\right) \tag{3.6.30}$$

can provide an indication as to the cause of the interruption of the on-line optimization. This *accuracy check* adds safety to an on-line optimizer.

This accuracy check is a very global check, and therefore easy to implement. However, it may be somewhat coarse. Consider for instance a plant that consists of n parallel units. Assume that at a given moment the model for unit 1 cannot be identified properly. The remaining $n-1$ unit models are still well identified. The impact on the confidence of $f^\star$ can be such that the optimization is completely stopped. A partial optimization of the $n-1$ units disregarding connections with 1 (a suboptimal result) would still have been possible, but is not realized. If, on the other hand, the impact of the weak model identification of unit 1 on $f^\star$ is too small to cause an interruption in the implementation of the optimization results, then unit 2 through n will still be optimized with each other, but as a result of the erroneous model of unit 1, the setpoints for this model could be changing very fast, or absolutely irrational. Therefore, for cases with well defined separate units with a limited interaction, this test does seem insufficiently refined. An accuracy test on the setpoints instead of the optimal value function is suggested under those circumstances.

3.6.3.4. On-Line Accuracy of the Expected Optimal Setpoints

Confidence intervals can also be calculated for solution of the optimization problem, the optimal setpoints. These confidence intervals can be calculated in the same way the confidence interval for the optimization objective is obtained :

$$\sigma_x^2 = \left(\nabla_\varepsilon x^\star|_0\right)^T Q_\varepsilon \left(\nabla_\varepsilon x^\star|_0\right). \tag{3.6.31}$$

In this formula, x represents the setpoints, and therefore the sensitivity of the optimal setpoints to ε has to be known.

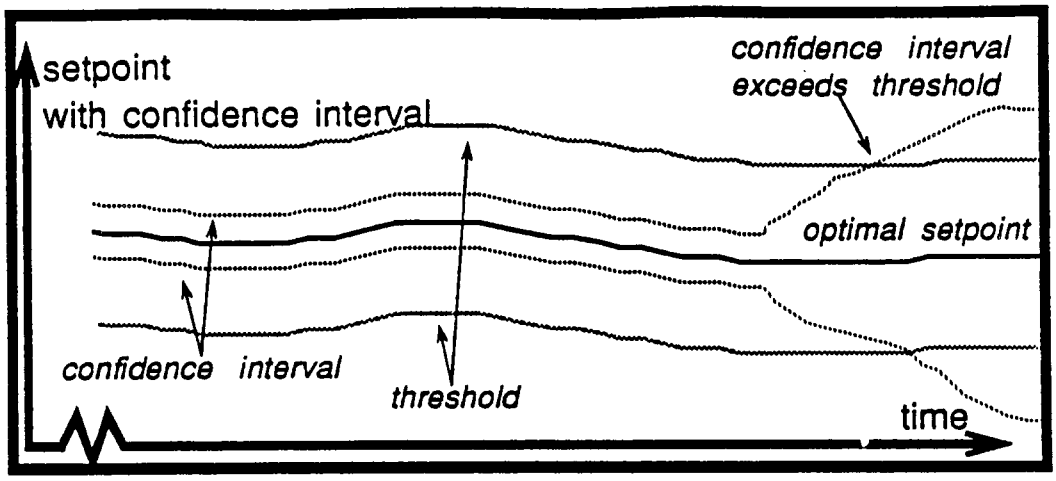

Fig.3.6.4. Use of Confidence Regions of Optimal Setpoints

A first application of these confidence intervals is identical to the application for the optimal value function that was suggested in the previous section. This point is illustrated in figure 3.6.4. The full black line schematically represents the result of the optimization, the optimal setpoint. The narrow dotted line above and below the optimal setpoint indicates the confidence interval (say 95%) of the setpoint that is represented by the full black line. The somewhat thicker dotted line indicates a preset threshold with which the confidence interval is compared. At a given point in time, it is suggested in the figure that the confidence region about the setpoint starts increasing suddenly. At the point that is indicated with the arrow, the size of confidence region becomes larger than a preset threshold. The ratio of the confidence region to the setpoint value could be used also. As soon as the confidence region interval becomes larger than a value that is agreed upon as too inaccurate, the optimization result becomes too uncertain for this setpoint. The result should not be implemented. If one or more optimal setpoints are considered inaccurate, three alternatives are open.

(1.) None of the optimization results (optimal setpoints) is implemented. This of course brings us more or less back to a more conservative version of the

previous accuracy test, which considered the confidence region around the optimal value function.

(2.) All the setpoints that belong to a certain section and unit of the plant that the inaccurate setpoint belongs to, are frozen. Or, similarly, all the downstream setpoints (if there is no large recycle) remain fixed. A potential risk is that the setpoints that are inaccurate do not change and therefore that information will never come in to provide enough information about the plant behavior to make the on-line optimizer certain about changes in that setpoint.

(3.) The inaccurate setpoints are not updated. The optimization is rerun (if enough time is available) with fixed values for the inaccurate setpoints.

A second application of the confidence intervals concerns the reduction of the number of setpoint changes. The optimization routine of the on-line optimizer is executed at a certain frequency. New optimal setpoints are obtained at every execution. If nothing changes significantly, models nor plant input, the new optimal setpoints will not be significantly different from the previous array of setpoints. Small changes in various setpoints in the plant may however cause upsets, and setpoint changes should be avoided if these changes are not significant. This concept is presented in figure 3.6.5.

In figure 3.6.5 a full black line and narrow dotted lines indicate the optimal setpoint surrounded by a confidence region. The implemented setpoint is only adjusted if a *significant settled change* is detected. An advantage of this method is that the plant behavior is more quiet, with the indirect advantage that steady state data collection is made easier. The significant difference is indicated in figure 3.6.5 as the point where the confidence regions of the implemented and the optimal setpoint become disjoint. This approximation is for most practical applications correct if optimal setpoints are uncorrelated. The

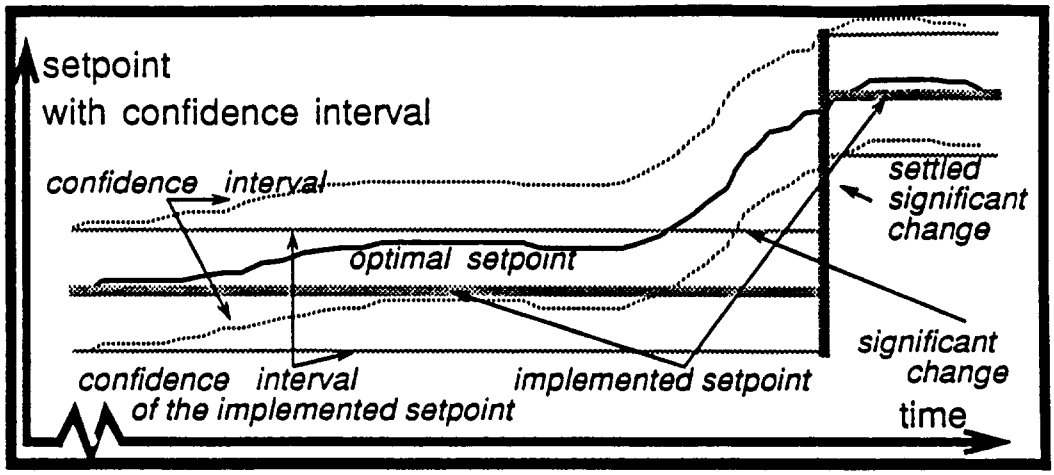

Fig.3.6.5. Use of Confidence Regions of Optimal Setpoints

technique for detecting significant changes in optimal setpoints does not require any tuning constants, except for a possible settling time. Some form of settling has to be applied. At the moment a significant change is detected, the setpoint may still change within the new significance envelope. This is clear from figure 3.6.5. A significant change was detected at the point indicated by the arrow "significant change". The setpoint continues to move after that detection of significant change, until it settled. If the setpoint that was the actual optimal setpoint at the time of the detection of the significant change would have been implemented, then a biased optimal setpoint would have been used. Another reason for applying a settling time, is the possibility of a nonlinear effect that could be characterized as an inverse response. An inverse response is illustrated in Chapter 5. In that case, due to nonlinearities in the model, the estimate of the optimal setpoint under certain circumstances, first moves in the opposite direction of the settled or "steady" change. Because of the large loss of performance during such transient phenomena, a settling time is very useful for system with this type of behavior. More details can be found in Chapter 5. If no "inverse response" behavior is expected, other methods requiring no tuning constants can be implemented, as illustrated in Chapter 5. The settling

time has to be selected with care, since large settling times may cause excessive delays.

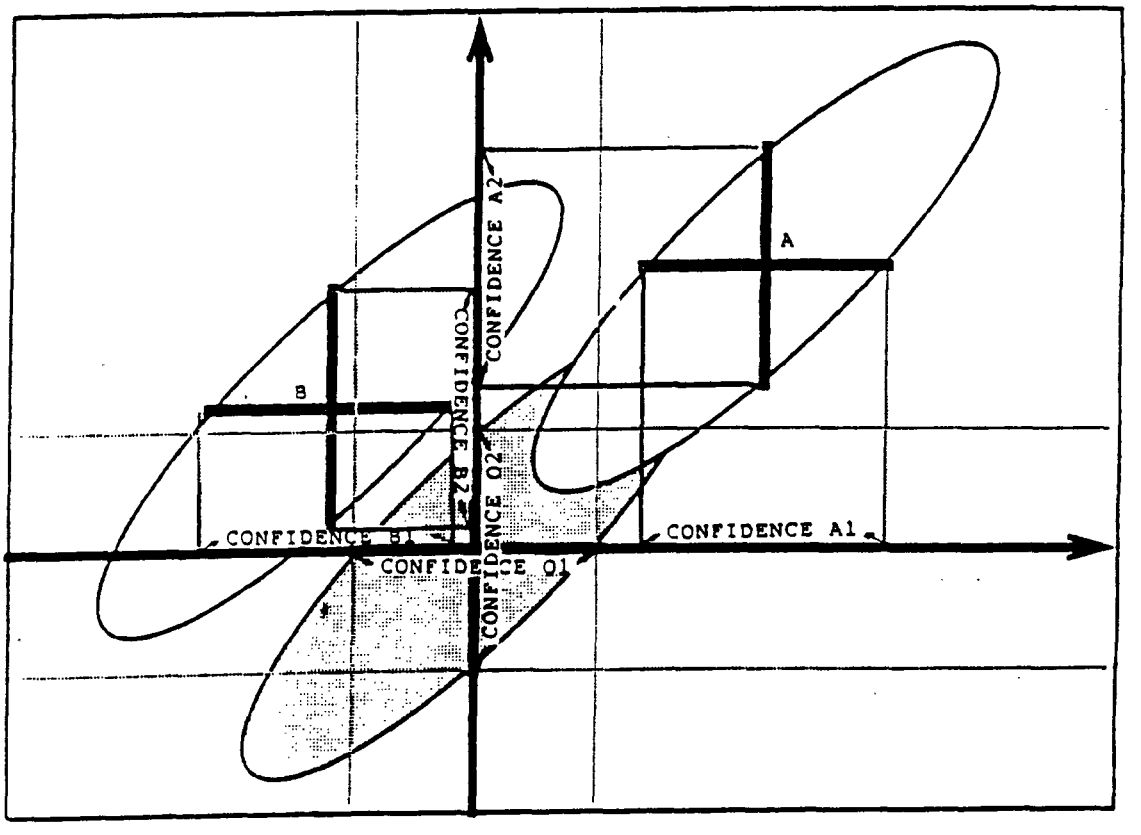

Fig.3.6.6. Significant and Insignificant Differences of Correlated Setpoints

Figure 3.6.6 shows a plane of setpoint pairs, with the implemented pair as the origin. Two alternative setpoint couples are given, together with their confidence ellipses. While discussing this figure, we will refer to the confidence interval of a setpoint (as opposed to the confidence ellipse of a pair of setpoints) as the line, through the setpoint pair, parallel to the axis of the setpoint within the ellipse. Using the criterion that was used in figure 3.6.5, A is not significantly different from the implemented setpoint, but B is. However, the confidence region for A and the implemented setpoint do not overlap. Therefore, they are significantly different. The confidence region of B and the implemented pair

do overlap. They are not significantly different. The conclusions based upon confidence intervals only, can therefore be wrong if there is strong correlation between the optimal setpoints. This correlation is a function of the steady state model, and therefore has the gains of process cross interaction as physical background. In higher dimensions the situation is analogous to the one described in figure 3.6.6. Before a criterion is presented to test for differences in arrays of setpoints, it has to be pointed out that the criterion suggested in figure 3.6.5, namely a disjoint confidence region, is not correct in the statistical sense. However, it is very close to the statistical solution, and in the one dimensional case it definitely is easier to use, since one only has to consider lines. In higher dimensional cases, the more correct statistical test is actually easier to use, than checking whether or not the intersection of two quadratic hypervolumes is empty or not. Therefore, the test is not based upon disjoint confidence regions. Instead it is based upon the distribution of the difference between the setpoints. The following paragraph will point this out.

The optimization routine provides optimal setpoints. Because of inaccuracies in the model and the data that are obtained from the plant, and because of the inaccuracies of the numerical techniques that are being used, these setpoints are not the "true" optimal setpoints for the plant under the given operating conditions, but merely an approximation of these "true " optimal setpoints. The confidence intervals around the setpoints, or more precisely the confidence regions around the array of optimal setpoints provided by the optimizer and the sensitivity analysis, quantify the degree of uncertainty of these "estimates" of the optimal setpoints that can be expected. Assume that the array of setpoints x_0 represents the current values of implemented setpoints. Assume also that the true optimal setpoint $x_0^\star$ at the moment the implemented setpoint was obtained from the optimization routine, and which is unknown, can be considered as a

sample of a multivariate normal distribution with mean x_0 and variance (matrix) Q_0. In the same way, let the current result of the optimization problem be represented by x_1, and let the true optimal setpoint $x_1^\star$, which is unknown, be a sample of a multivariate normal distribution with mean x_1 and variance Q_1. Let in the remainder of this section i equal 0 or 1. Assume that Q_i is symmetric and positive definite. In reality Q_i is semi-positive definite. We will come back to the special case that Q_i is singular. It can be shown (see e.g. Golub and Van Loan [1983], p.88) that the *Choleski decomposition* of Q_i is given by :

$$Q_i = G_i^T G_i \tag{3.6.32}$$

in which G is a lower triangular matrix that is referred to as the *Choleski triangle*. Equation 3.6.32 can be rewritten as :

$$(G_i^T)^{-1} Q_i G_i^{-1} = I_n \tag{3.6.33}$$

in which I_n is the identity matrix of order n, which is the number of setpoints. Consider a linear transformation $y_i = G_i x_i$. Then the transformed true optimal setpoint $y_i^\star$ can be considered as a sample of a multivariate normal transformation with mean $G_i x_i$ and variance $(G_i^T)^{-1} Q_i G_i^{-1}$. This variance matrix is the identity matrix since G_i is the Choleski triangle for Q_i. Under these circumstances $y_1^\star - y_0^\star$ can be considered as a sample of a multivariate normal distribution with mean $y_1 - y_0$ and variance twice identity. The two setpoints will be virtually equal (not significantly different) with α confidence if the origin (0) is within the α percentile sphere around $y_1 - y_0$. Therefore, if $||G_1 x_1 - G_0 x_0||_2 \leq 2 z_\alpha$ then both arrays of setpoints are not significantly different.

An efficient and very stable algorithm for the calculation of the Choleski triangle is given in Golub and Van Loan [1983], p.89. It requires $n^3/6$ flops,

with n the number of setpoints. A flop is roughly the amount of work associated with a floating point multiplication and a floating point addition. The complete operation requires approximately $n^3/3$ flops. The simple comparison of confidence intervals (not regions) on a setpoint by setpoint basis as in figure 3.6.5 requires approximately $6n$ flops. Therefore, the procedure should remain restricted to subsets of the array of setpoints among which strong correlation can be expected.

It was assumed that Q_i was positive definite. That is not always so. Assume for instance that a certain setpoint has a maximum value constraint which is active. If the maximum value is well known, for instance if it is an instrumentation limit, then the variance of that setpoint is zero. The cross variances of this setpoint with any other setpoint is also zero. Therefore, setpoints with active box constraints have to be deleted from the analysis.

More generally speaking, a constraint with only exactly known constants, except for setpoints, introduces an exact relationship and will render the Q_i matrix singular. An exactly known constraint is a constraint that does not contain any variable in the vector ε that is used in equation 3.6.31. An exact relationship reduces the degrees of freedom by one. Therefore, the presented method will not work if exact constraints are active, unless the number of setpoints is reduced, by deleting a setpoint that is involved in the exact relationship.

3.6.3.5. Sensitivity Information for On-Line Monitoring of the Process Operation

A last area of application of sensitivity analysis in the field of on-line optimization that is presented here, concerns displaying the most important sensitivity information on-line. Sensitivity information is relatively easy to interpret. The information is given in the form of derivatives, which can be considered as proportionality factors in an area close to the current operation point. In in-

dustrial on-line applications, one can expect the optimum to lie at a constraint or an intersection of constraints. Sensitivity analysis could display immediately which constraint(s) are the most limiting constraints, and how much the performance can be expected to improve by loosening a constraint or deteriorate by tightening a constraint. A large scale on-line optimization application can be expected to work around an optimization problem with a very large number of constraints. These constraints result from numerous reasons such as safety, equipment limits etc. Many constraints will contain constants that are obtained from "engineering practice" and "sound engineering judgement" using "reasonable values" and "expected limits". Assume for instance that the maximum inlet temperature for a multitubular ortho-xylene oxidation reactor is limited to 350^oC to prevent runaway. During the operation the constraint becomes active and shows to be "expensive". In other words, sensitivity analysis results indicate that loosening the constraint potentially will increase the profit significantly. At that point, a more detailed analysis can be done on the runaway characteristics of this reactor, and while monitoring possible early indications for runaway, the constraint can be loosened carefully until it becomes inactive, or until the risk for runaway becomes too large for the potential gain in performance. Similar scenarios can be presented in which sensitivity plays a crucial role in fine-tuning and monitoring the process.

3.6.4. Summary and Conclusions

The use of sensitivity analysis is clearly attractive in an on-line optimization environment. The results of sensitivity analysis can be interpreted as an approximation of the analytical solution in a neighborhood around the problem parameter values that are under consideration. This interpretation and the notion of the usefulness of an analytical solution by itself indicate the potential of

sensitivity analysis results.

The most important theoretical results on stability and sensitivity are presented without proof. Most results require rather strict conditions with respect to the differentiability of the problem functions and the second order optimality condition. It has to be kept in mind that sensitivity results are only valid under those conditions. Some counterexamples (the conditions are not met and the conclusions/results are wrong) in stability and sensitivity analysis are given by Gauvin and Dubeau [1984]. It is also underlined that general nonlinear programming stability results for *finite* differences are not published. Such results would allow one to estimate for which changes in ϵ active constraints become inactive and vice versa.

Four important types of application of sensitivity analysis in on-line optimization are described. These are *(i)* marginal optimization gain calculation, *(ii)* off-line and *(iii)* on-line accuracy assessment of the expected optimal value function and *(iv)* setpoints, and *(v)* monitoring and interpretation of the operation of the on-line optimized system.

(i) Sensitivity results allow one to estimate the marginal optimization gain, i.e. the difference in performance (e.g. profit or operating cost) of the on-line optimized system and a *good* reference state. This reference state is optimized at nominal operating conditions. Often in on-line optimization application examples, the optimization result is compared to a very poor operation, and the on-line optimizer has to come out very well. The marginal optimization gain method gives a "lean" estimate of the advantage of on-line optimization for a given system with a minimum of information. The method is a good necessary condition to be evaluated early in the decision process concerning an on-line optimizer.

(ii) By combining statistical and sensitivity information, on can estimate the

accuracy with which the expected optimal value is obtained. If the accuracy is low compared to the range over which this optimum value performance can vary in the course of expected operating condition variations, then the on- line optimizer operation can be characterized as "shooting in the dark". This presents another necessary condition for a system to be an interesting candidate for on-line optimization.

(iii) If the accuracy assessment is obtained on-line, using on-line statistical information from the model parameter identification, then a number of malfunctions of the on-line optimization system can be detected from an increase in the variance of the optimal value. Checking this accuracy before implementation of the optimization results and breaking the optimization loop if accuracy is to low, adds robustness and safety to the closed-loop optimization.

(iv) In the same way, confidence intervals on the optimal setpoints can be used. Furthermore, confidence information on the setpoints can be used to prevent unnecessary setpoint changes. In this way the number of upsets introduced by the on-line optimizer is minimized.

(v) Finally, sensitivity analysis creates a picture of the optimal value function in the neighborhood of the active operating condition. This allows the operating staff to understand the limitations of their operation, where most money can be made and how the operation can be improved. The sensitivity analysis of a the optimization problem (P) allows one to evaluate how well (P) is posed and how it can be improved. This suggests an iteration from problem formulation to solution and back. This type of interaction is known to exist in design problems as well.

CHAPTER 4

CASE STUDY DESCRIPTIONS

4.1. INTRODUCTION

In this chapter, the systems used in the case studies are described. The results from the case studies are presented in the next chapter. Case studies are used as an illustration of the theory, to gain insight in the characteristics of on-line optimizers, to explore problems related to on-line steady state optimization and to recognize tendencies and trends. On-line optimization using steady state models is an interdisciplinary subject that employs procedures from other areas, among them statistics, optimization theory, filtering and estimation theory. The applications are often large scale, and the estimation and optimization problems posed require a numerical solution in almost all cases. Theoretical study only would not guarantee that problems are addressed that are realistic, nor that all difficulties have been taken into account. The case studies are selected such that they represent systems with a broad range of application, and at the same time exhibit typical characteristics of many processes in the process industry. The case studies are relatively simple examples to allow for a complete analysis of the results in an academic context. All case studies are computer simulation studies.

Three case study systems are described in the remainder of this chapter :

* Reflux Ratio and Top Composition Optimization for a Propane Propylene Splitter ;

⋆ Optimal Load Allocation in a Boiler Network ;

⋆ Optimal Load Allocation in a Heat Exchanger Network.

The propane propylene splitter and the heat exchanger networks are steady state studies, while the boiler network is simulated dynamically.

4.2. A PROPANE PROPYLENE SPLITTER

4.2.1. Distillation : Good Candidate for On-Line Optimization

The first case study that is described here, considers the on-line optimization of a distillation tower. Distillation towers are very interesting candidates for on-line optimization. Being very widespread physical separation systems and large energy consumers, they account for a relatively large share of the national energy consumption [Shinskey,1979] [Radian Corp.,1980]. Relatively small reductions of the energy consumption by distillation could result in relatively large improvements in overall plant operation cost.

The steady states of many distillation columns can often be simulated using simple short-cut models in an efficient, yet accurate way. An easily accessible and tractable steady state model makes on-line optimization more feasible.

According to Martin *et al.* [1981] the following characteristics of distillation columns strongly favor the application of reflux optimization :

◇ high reflux ratios

◇ high differential product values

◇ high utility costs

◇ low relative volatility

◇ feed light key far from 50%

According to Martin *et al.* [1981] the following characteristics less strongly favor on-line optimization of the reflux ratio but may still indicate that the method

should be considered :

- ◇ feed composition varying over a wide range
- ◇ hard target specification variations
- ◇ utility cost fluctuations
- ◇ variable differential product values

The first four factors make the variable operating cost more sensitive to the reflux ratio. The last series of four factors seem to contribute to the variability of the operating cost. If the operating cost can vary over a large interval as a function of its operation, then the probability that on-line optimization is useful become larger. If the operating cost of a column were constant under all possible operation schemes, then there would be nothing to optimize. The factors listed above tend to make the range over which the operating cost can vary, large. Therefore they are characteristics of columns that are interesting candidates for on-line optimization. The column that will be studied here is a propane - propylene splitter. These columns have high reflux ratios and a low relative volatility, two characteristics that strongly favor reflux ratio on-line optimization. Usually these columns are part of the separation section of an ethylene plant. Therefore, optimizing the reflux ratio of the propane - propylene splitter without communicating the optimization results to other units in the ethylene plant comes down to a partial optimization. Funk *et al.* [1984] warn against partial optimization of specific units in ethylene plants. Each unit interacts with and influences the other parts of the plant. An efficient operation of a specific unit may force another unit into a much less efficient operation. However, the optimization variable used here is the reflux ratio, an internal stream of the unit. These variables were selected because Martin *et al.* [1981] selected them. Other combinations of independent variables can be selected as optimization variables, but in this way, comparison between this case and the

referenced publication becomes possible. The influence of the optimization on the streams exiting the unit is much more moderate than on the reflux stream in the column itself. The composition of the top stream (the main and most expensive product) is fixed, and the bottom stream, mostly propane, is usually recycled to the cracking furnace. The impact of this bottom stream on the total plant economics is not necessarily small. But for the illustrative purpose of this case study, it is assumed that this impact can be neglected. A very important factor in the plant-wide validity of the optimization of one single column, is the accuracy of the intermediate product values used. These intermediate values are a function of many factors such as the operation of the other units in the plant and the current market condition. However, if these intermediate product values are accurate and correct for the optimum operation, then the partial optimization of this single unit is absolutely valid in the total plant context. The intermediate product values replace the connectivity constraints that would have to be introduced in a formulation of an optimization of the complete ethylene plant.

4.2.2. Distillation : a Good Case Study

This case study is primarily aimed at the study of on-line optimization, not at the study of distillation equipment or distillation control. The situation studied is therefore somewhat simplified, and may not be recognized as state of the art by distillation experts. However, the simplifications introduced allow a reduction of the effects that would make the unambiguous observation of the characteristics of an on-line optimizer difficult. Nevertheless, many of the typical properties of a distillation column are maintained, the most important of which is the nonlinearity of a number of unit responses. By considering only one unit, the dimensions of the problem in this study are reduced. The example

chosen allows a study of many of the interesting topics of steady state on-line optimization on a small and hence clear and efficient scale.

This case study shows a good application of short-cut models. A steady state on-line optimizer utilizes a steady state model as a constraint in the optimization. In choosing the steady state model that will be used, one has to trade off complexity and accuracy with computational simplicity and approximation. A very accurate model often requires a lot of computational effort, but will fit the plant data over a wider range. Larger set point changes can be made without concern about the accuracy of the prediction made using the model. A simple model will require less computation, but will only fit a narrow range well. More effort is required from the model updating, and steps in setpoints may be limited in size. On-line optimization is intended to be applied to large systems, and therefore even a "short-cut" model will be considerable in size. It is quite often not practical to use highly accurate models, since they result in too large computational tasks. A good engineering application of on-line optimization of large systems will therefore often make use of short-cut models. The use of short-cut models is illustrated here. While a simple model is being used by the steady-state optimizer, a more exact model will be used as the plant. Since the short-cut model does not fit exactly the results of the model that represents the plant, parameters of the short cut model have to be updated. This estimation phase is very important and often the critical step in the steady state on-line optimization process. The influence of estimation techniques is illustrated in this example by comparing estimation techniques. The use of data consistency checking techniques such as data reconciliation and outlier detection is discussed and illustrated with examples.

This case study also illustrates how statistical results from the parameter estimation and sensitivity information can be combined. Parameter estimate

confidence information and the sensitivity analysis of optimization results offer a perspective on the status of the results of estimation and optimization. Combined they allow for a reliability analysis of the on-line optimization. This is illustrated in the discussion on steady state on-line optimization feasibility of the distillation column being studied in the case study.

In plant-wide applications, an on-line optimizer will have to process a large amount of raw plant data. These data are not always reliable. They may contain gross errors. Data reconciliation and outlier detection are very effective tools to identify and possibly correct these errors. This is illustrated by this case study.

The closed loop performance of the column with an on-line optimizer is discussed also. Only steady-state simulations were done, and the possible interference of the on-line optimizer operations with plant or disturbance dynamics is illustrated in another case study.

4.2.3. A Propane Propylene Splitter : Description

Martin *et al.* [1981] describe the steady state on-line optimization of a propane - propylene splitter. The objective of the optimization is the reduction of the daily operation cost of the column. The optimization variables are the reflux ratio and the top product purity. The constraints of the optimization are the steady state model of the distillation column and the top purity specification. A 95% propylene weight fraction is required in the distillate. In the range of operating conditions that are of interest here, this constraint will always be active. The steady state model parameters are frequently updated, according to the authors. For this single tower, Martin *et al.* [1981] report savings of over $ 300,000 per year. This is one of the reasons for which this is an excellent case study example.

The column described by Martin *et al.* [1981] is used here. The column separates a mixture that mainly consists of propane and propylene, and is a unit of an ethylene plant separation section. The column has 125 trays, and Martin *et al.* [1981] give an average tray efficiency of 75% . This efficiency means that the column has approximately 94 ideal trays. The column is operated at a high reflux ratio. The operating point given by Martin *et al.* [1981], here after referred to and used as the *nominal state*, has an optimal reflux ratio of 17.2, for a feed rate of 1.2 million lb per day with a 70% weight fraction propylene in the feed. The feed is saturated liquid.

As mentioned before, the objective function for the optimization is the daily operation profit. This profit is itemized as follows :

(propylene sales + propane sales - utility cost - raw material cost)

If the following notation is used :

- c_d : price of propylene in the distillate ($/lb)
- c'_d : price of propane in the distillate ($/lb)
- c_b : price of propylene in the bottom product ($/lb)
- c'_b : price of propane in the bottom product ($/lb)
- x_b : weight fraction of propylene in the bottom product
- x_f : weight fraction of propylene in the feed flow
- x_d : weight fraction of propylene in the distillate
- B : bottom flow (lb/day)
- F : feed flow (lb/day)
- D : distillate flow (lb/day)
- Q_r : reboiler heat load (BTU/day)
- Q_c : condenser heat load (BTU/day)
- c_1 : heating cost ($/BTU)
- c_2 : cooling cost ($/BTU)

then the daily operating cost can be written as :

$$f = \{c_d x_d D + c_b x_b B\} + \{c'_d(1 - x_d)D + c'_b(1 - x_b)B\} -$$

$$\{c_1 Q_r + c_2 Q_c\} - \{c_f x_f F - c'_f(1 - x_f)F\}. \qquad (4.2.1)$$

The terms in the braces correspond to the propylene sales, the propane sales, the utility cost and the raw material cost respectively. By using $W = c_d - c_b$ (light key differential value) and $U = c'_b - c'_d$ (heavy key differential value), this equation can be rearranged to :

$$\begin{aligned} f =& c_d x_f F + c_b(1 - x_f)F - c_f x_f F - c_f(1 - x_f)F - \\ & c_1 Q_r - c_2 Q_C - W x_b B - U(1 - x_d)D \end{aligned} \qquad (4.2.2)$$

The feed composition and flow rate are given, and cannot be changed by adjusting the manipulated variables of the optimization : the reflux ratio and the top purity. Once x_f and F are given, the on-line optimizer will estimate an optimal reflux ratio and top composition subject to the objective, the constraints, and the plant input. Therefore the first four terms in this equation are constant, and are of no importance during the optimization. These terms can be omitted in the optimization objective. The remaining terms will be called $-f_1$. Changing the sign makes the objective a cost that has to be minimized.

$$f_1 = c_1 Q_r + c_2 Q_c + W x_b B + U(1 - x_d)D. \qquad (4.2.3)$$

One has to keep in mind however that f_1 is not the complete operating cost, but only the fraction of the operating cost that is variable with the reflux ratio and the top composition. Finally the following assumption is sufficiently accurate for the purposes of this case study as well as for modeling many other towers :

$$Q_r = Q_c = \lambda V. \qquad (4.2.4 - 5)$$

In this equation λ equals the average latent heat per unit of mass, V the vapor boilup in mass [Edgar & Himmelblau,1988]. This assumption is valid if the enthalpy input of the feed flow to the column approximates the enthalpy output of top and bottom stream combined. This assumption is reasonable if a column is highly refluxed. Then f_1 reduces to :

$$f_1 = (c_1 + c_2)\lambda V + W x_b B + U(1 - x_d)D. \tag{4.2.6}$$

The values for various constants listed by Martin *et al.* [1981] are also applied in this case study and are listed in table 4.2.1. As already mentioned, the operating conditions given by Martin *et al.* [1981] are referred to in this study as the nominal state, and are listed for an optimal reflux ratio and top stream composition in table 4.2.2. An optimal reflux ratio will always result in a top purity of 95% in the range of operation considered, and that is the minimum required. In other words : the $x_{d,min} \leq x_d$ constraint can be expected to be always active.

Figure 4.2.1 gives f_1 as a function of the reflux ratio along the $x_d = .95$ constraint, when using Eduljee's and fenske's equations which are discussed in the next section.

The physical cause of this minimal operating cost for a certain reflux ratio will be explained next [Martin *et al.*, 1981]. The variable operating cost has been split into energy cost and the sales profits. Therefore, for a fixed top composition :

$$\text{variable operating cost} = f = g(R) - p(R). \tag{4.2.7}$$

with R being the reflux ratio, $g(R)$ the energy cost as a function of the reflux ratio and $p(R)$ the value of the product streams, also as a function of reflux ratio. The top composition is constant. The cost $g(R)$ is approximately linear in the reflux ratio R.

$$g(R) \simeq a + bR. \tag{4.2.8}$$

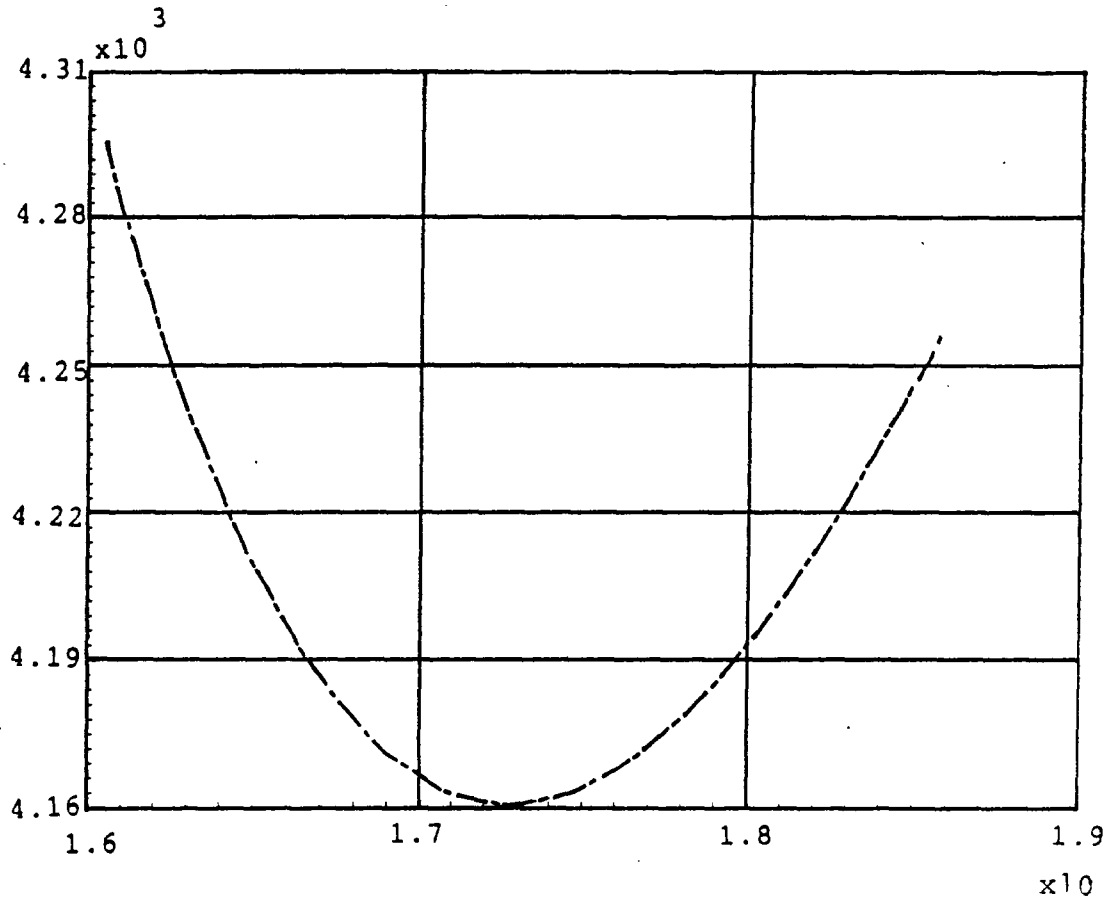

Fig.4.2.1. f_1 *(variable cost) as a function of the reflux ratio for* $x_d = 0.95$

Therefore, the minimal operating cost can only be reached for a reflux ratio such that :

$$b = \frac{df(R)}{dR}. \tag{4.2.9}$$

Figure 4.2.2 presents three cases of different energy costs, together with the corresponding variable operating cost (VC=f). The broken lines represent the cost of energy, used for heating and cooling. The curve labeled "recovery value" indicates the product sales value. Three different levels of energy cost correspond to lines 1,2 and 3. These levels in the three variable profit curves that are

Symbol	Description	Value (dim)
c_1	reboiler heat cost	$3 \cdot 10^{-6}$(\$/BTU)
c_2	condenser cooler cost	0 (\$/BTU)
N_{true}	number of trays	125
η	average tray efficiency	0.75
N	number of ideal trays	94
U	heavy key diff. value	−0.08 (\$/lb)
W	light key diff. value	0.10 (\$/lb)
λ	average latent heat	130 (BTU/lb)
α	relative volatility	1.105

Table 4.2.1.: propane/propylene objective specifications

Symbol	Description	Value (dim)
F	feed mass flow	$1.2 \cdot 10^6$(lb /day)
x_F	propylene weight fract. in feed	0.7
$x_{D,m}$	required top purity	0.95
R	reflux ratio	17.2
R_m	minimal reflux ratio	11.3
B	bottoms mass flow	$331 \cdot 10^3$ (lb /day)
x_B	bottoms propylene weight fract.	0.044
D	top mass flow	$869 \cdot 10^3$ (lb /day)
x_D	top purity	0.95

Table 4.2.2.: propane/propylene splitter nominal state

indicated -VC1, -VC2 and -VC3. The maximum of these curves indicates the optimum reflux. At the lowest level of energy cost, the optimum policy maximizes the reflux. Higher energy costs result in lower reflux rates. In general a column can be a candidate for on-line optimization if the operating cost curve is more convex than the sales value curve as a function of the reflux ratio.

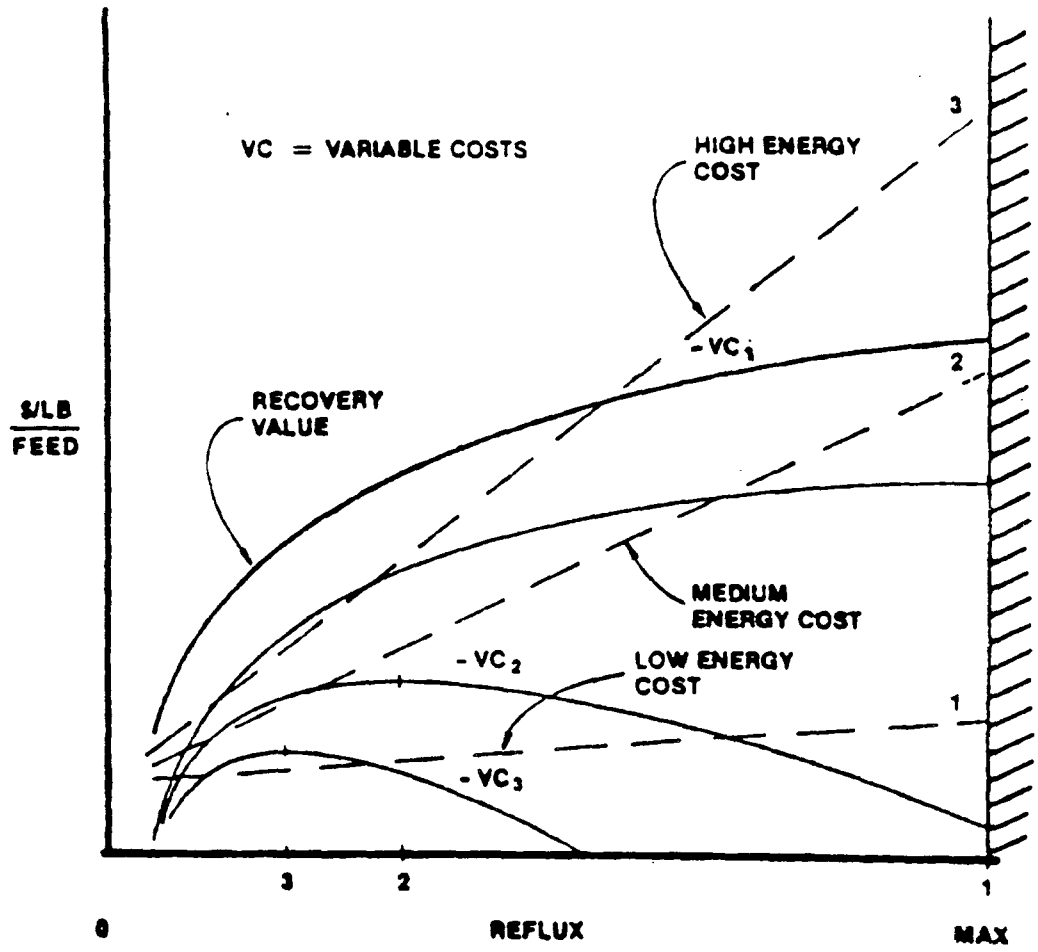

Fig.4.2.2. Variable Operating Cost (VC) as a Function of the Reflux Ratio for Different Energy Costs (from Martin et al., [1981]).

Martin *et al.* [1981] present a base case with a variable cost of $ 5239.74

per day. After optimizing the reflux ratio (increased from 14.12 to 17.76) the variable cost was $ 4401.38 per day, a gain of $ 838.80 per day ($ 306,000 per year).

The fact that the reflux ratio and the top composition were chosen as optimization variables does not imply any assumption concerning the control structure of the column. The actual setpoints of the control system do not necessarily have to be the top composition and the reflux ratio. They can be the top composition and the bottom composition as predicted for the optimal reflux ratio by the on-line optimizer model. It is only assumed that the steady state of the column will be reached at the reflux ratio and top composition dictated by the on-line optimizer. Dual composition control is shown to result in minimal energy consumption in a distillation tower by Luyben [1975]. In this example the bottom composition is also controlled but its setpoint is variable as a function of unit economics. It is only assumed that the control (preferably smooth and reasonably fast) finally achieves this setpoint, no matter how. The bottom composition is not fixed at a constant value, but is determined by the minimization of the objective in which the vapor boilup and the bottom composition is taken into account. Therefore, this scheme should perform better than regular dual composition control with two fixed setpoints.

4.2.4. Models Used

Two steady state distillation column models are presented in this case study. The first model that is discussed is a short-cut steady state model used in the optimization phase. The parameters of this model are updated. In an actual application, this is the only model needed. The second model that is described, will function as the plant on which the on-line optimizer is applied in a number of simulations that are performed in this case study. The model

has a different structure, and is more accurate than the short-cut model used by the optimizer. A realistic model/plant mismatch is in this way introduced.

4.2.4.1. Steady State Model Used by the On-line Optimizer

Short-cut models are necessary to keep on-line optimization of larger systems feasible. A good fit has to be traded off against model complexity and hence a greater computation effort. A poor fit will undoubtly falsify the results, and too complex a model may make the parameter estimation difficult, and it will make the optimization more complex. The on-line optimization concept is aimed at plant-wide application. This implies processing of numerous measurements, and estimating many parameters. A good engineering solution will often fall back on short-cut models. This case study illustrates the use of such models.

The on-line optimizer makes use of the Eduljee correlation, which is based upon a fit of a graph by Gilliland [1940]. The Fenske equation for total reflux [1932] is used as well. The use of these equations for short cut modeling of binary distillation towers has been described and illustrated by many authors, e.g. Mc Avoy [1983] and Edgar and Himmelblau [1988].

First, using x_D, x_F and the average relative volatility (α), calculate the minimal reflux at which the desired top composition can be obtained.

$$R_m = \frac{1}{\alpha - 1}\left(\frac{x_D}{x_F} - \frac{1 - x_D}{1 - x_F}\right). \tag{4.2.10}$$

The value obtained for the minimal reflux ratio R_m and the number of ideal trays are substituted into the Eduljee correlation to find the minimal number of ideal trays N_m necessary to realize the split at total reflux.

$$\frac{N - N_m}{N + 1} = \frac{3}{4}\left(1 - \left(\frac{R - R_m}{R + 1}\right)^{0.5668}\right). \tag{4.2.11}$$

The equation of Fenske can be used to calculate the bottom composition :

$$N_m = \frac{\ln\left(\frac{x_D}{1-x_D} \cdot \frac{1-x_B}{x_B}\right)}{\ln(\alpha)}. \tag{4.2.12}$$

Finally, solving the mass balances provides us with the top and bottom flow rates. Assuming equal molal overflow, the column liquid and vapor streams in the column become known.

$$F = D + B \tag{4.2.13}$$

$$Fx_F = Dx_D + Bx_B \tag{4.2.14}$$

$$L = RD \tag{4.2.15}$$

$$V = (R+1)D \tag{4.2.16}$$

These equations provide the information needed to calculate f_1, the optimization objective.

4.2.4.2. A More Rigorous Steady State Model

4.2.4.2.1. Description

In this case study a simulation takes the place of the actual plant. A steady state simulation is used here. In some of the simulations, Fenske's model was used for the "plant" *as well as* by the optimization routine. In the runs were Fenske's model is used for both the plant and the steady state model, the plant model mismatch is excluded. In case a plant model mismatch is desired, a more rigorous steady state distillation tower model is used. The effects of the plant dynamics on the performance of the on-line optimizer are not considered. Later studies will investigate the interactions between the plant dynamics and the on-line optimizer. The advantage of using only a steady state simulation is that dynamic effects do not contribute to making the results more difficult to interpret.

The more rigorous steady state model used here makes use of Underwoods' equations. Underwoods' equations have been used in tower design and have been shown to be reliable for towers that are nearly ideal. It is true that more rigorous models exist and could have been applied here, but it has been mentioned before that the primary subject for this case study is on-line optimization. Underwoods' equations are given by e.g. King [1979], and Van Winkle [1967]. Mc Avoy [1988] describes and illustrates how the model can be fitted to existing columns and how it can be applied to on-line optimization.

Underwoods' equations are derived from mass balances over the stripping and rectifying sections of the distillation column. It is assumed that the relative volatility is constant but not necessarily equal in both tower sections. In the case study being described here, these Underwood equations are used to obtain the top and bottom streams and the bottom composition given the feed stream and its composition, the top composition, the reflux ratio, the number of ideal trays in each section and the relative volatilities in the rectifying and the stripping sections. Constant molal overflow is assumed. The model solution requires iteration in a single loop (only one variable).

4.2.4.2.1. Underwoods' Equations and Their Solution

In this section, the equations of Underwood and their use in modeling distillation towers is presented. A longer description can be found in King [1979]. Underwoods' model is a tray by tray model. The following assumptions are listed by King [1979] and Van Winkle[1967] : constant molal overflow, constant relative volatility in the section considered by the equation. These assumptions are significantly in error for highly nonideal systems. For systems that are slightly nonideal but with a large difference in volatility, the assumptions are violated. However, in that case the separation is usually completed in a few stages, and therefore the error will not be very large. For relatively ideal

systems that are hard to separate, the assumptions are good, and the error is usually small. Here, the relative volatilities are assumed constant in the rectifying section as well as in the stripping section. Then the equations collapse to the ones given below. In the context in which these equations are used, the bottom composition and the top and bottom flow is obtained from feed flow and composition data together with the top composition. Relative volatilities and number of ideal trays in rectifying and stripping section are assumed to be known.

Define two factors Φ_1 and Φ_2 as the roots of :

$$V = \frac{\alpha_A D x_{A,D}}{\alpha_A - \Phi} + \frac{\alpha_B D x_{B,D}}{\alpha_B - \Phi}. \tag{4.2.17}$$

In the equation above, α_A and α_B are the relative volatilities of the components A and B, in this case propylene and propane respectively. If the volatility is referenced to B, then $\alpha_A = \alpha$ and $\alpha_B = 1$. Further, D is the distillate molal flow, $x_{A,D}$ is the light component mole fraction in the distillate, and V is the vapor boilup.

$$V = \frac{\alpha D x_{A,D}}{\alpha - \Phi} + \frac{D(1 - x_{A,D})}{1 - \Phi}. \tag{4.2.18}$$

It can easily be shown that $\alpha \geq \Phi_1 \geq 1$ and $1 \geq \Phi_2 \geq 0$. King [1979] showed that from the mass balances over the rectifying section and stripping section the following equation can be derived :

$$\left(\frac{\Phi_2}{\Phi_1}\right)^{N_R} = \frac{\left(\frac{\alpha x_{A,F}}{(\alpha - \Phi_1)}\right) + \left(\frac{(1 - x_{A,F})}{(\alpha - \Phi_1)}\right)}{\left(\frac{\alpha x_{A,F}}{(\alpha - \Phi_2)}\right) + \left(\frac{(1 - x_{A,F})}{(\alpha - \Phi_2)}\right)}. \tag{4.2.19}$$

In this equation $x_{A,F}$ is the fraction of light component on the feed plate, which is not necessarily equal to the fraction of light component in the feed flow, and N_R is the number of ideal trays in the rectifier section of the column.

Simultaneously, King [1979] shows for the stripping section :

$$-V' = \frac{\alpha' B x_{A,B}}{\alpha' - \Phi'} + \frac{B(1 - x_{A,B})}{1 - \Phi'}. \tag{4.2.20}$$

In this equation V' stands for the vapor boilup in the stripping section, and B stands for the bottoms stream. $x_{A,B}$ is the fraction of light component in the bottoms stream. It can be shown that $\Phi_1' \geq \alpha'$ and $\alpha' \geq \Phi_2' \geq 1$. From a mass balance over the stripping section, it can be shown that :

$$\left(\frac{\Phi_2'}{\Phi_1'}\right)^{N_S} = \frac{\left(\frac{\alpha' x_{A,F}}{(\alpha'-\Phi_2')}\right) + \left(\frac{(1-x_{A,F})}{(\alpha-\Phi_2')}\right)}{\left(\frac{\alpha' x_{A,F}}{(\alpha-\Phi_1')}\right) + \left(\frac{(1-x_{A,F})}{(\alpha-\Phi_1')}\right)}. \tag{4.2.21}$$

In this equation, N_S stands for the number of trays in the stripping section. α' is the relative volatility in the stripper section. Using the simple relation $V = (R+1)D$, the expression for V is simplified to :

$$(R+1) = \frac{\alpha x_{A,D}}{\alpha - \Phi} + \frac{1 - x_{A,D}}{1 - \Phi}. \tag{4.2.22}$$

This equation can be solved for Φ_1 and Φ_2 since all the other variables are given. Using Φ_1, Φ_2 and α, equation 4.2.19 can be solved for $x_{A,F}$. This equation is linear in $x_{A,F}$. At that point $x_{A,B}$ has to be guessed. Using a similar technique and the mass balances over the tower, a corresponding $x'_{A,F}$ can be obtained. $x_{A,B}$, the bottom mole fraction has to be adjusted until $x_{A,F} = x'_{A,F}$. The expression for V' can be replaced by $(R+1)D + F$. Furthermore, D and B can be obtained as a function of the estimate $x_{A,B}$. A simple line search will converge very fast to the value of $x_{A,B}$ for which $x_{A,F} = x'_{A,F}$. Therefore the model can be solved and the simulation results are available.

In the simulation results that are presented in Chapter 5, the following constants are used : number of rectifier trays = 52, number of stripping trays = 44, α_{top} = 1.10, α_{bottom}=1.11. A more accurate approach would calculate the relative volatilities as a function of feed tray, top and bottom compositions. A geometric average of the relative volatility of the top and the feed tray would be used for the rectifying section, and a geometric average of the relative volatility of the bottom and the feed tray would be used for the stripping section.

Simulations showed that the difference between results using the constant relative volatilities for every section differ with less than 1% from the results using variable volatilities.

The mismatch or gap between the model that was first discussed which uses the equations of Eduljee and Fenske, and the model that is based upon the equations of Underwood will be discussed together with the case study results.

4.3. BOILER NETWORKS

4.3.1. Boiler Networks and On-line Optimization

Load allocation between parallel units is one of the first applications of on-line optimization. The optimization of load allocation between boilers connected to a common steam header has been the subject of theoretical studies, and industrial applications have been reported (e.g.: Cho [1978], Green and Al ai-Shaikh [1980], Nath *et al.* [1986], Ko [1987], Lipták [1987]). Boilers are important energy consumers. In 1981, 17% of the United States national energy was consumed by industrial boilers. Boilers for the production of electricity consume about 30% of the national energy. In 1981, the global U.S. consumption of energy for the production of steam corresponded to approximately 30 quadrillion ($3 \cdot 10^{16}$) BTU per year. It would take more than 5000 supertankers to haul the equivalent amount of energy in crude oil (ref.: Garcia-Borras, [1983]). This means that all boilers in the U.S. burn a supertanker of oil (or its equivalent in coal, bark, natural gas, recovered waste heat) every hour and forty-five minutes. Small improvements in the efficiency of this operation can have very important effects on the national energy spending. Therefore, it is worthwhile to investigate all factors that can increase the efficiency of the operation of steam boilers. Load allocation optimization is one of these.

4.3.2. Boiler Networks as a Case Study

The production of steam is one of the first industrial processes, and also, as illustrated above, remains one of the most important processes. It is well documented, and good models are available. This case set-up allows the study of various dynamic effects. This case study also offers the possibility to illustrate the applications of on-line sensitivity and statistical analysis. Furthermore,

short-cut feasibility studies can be compared to published results. In many aspects, boiler load allocation is a very interesting case study with many direct practical implications.

4.3.3. Network Arrangement

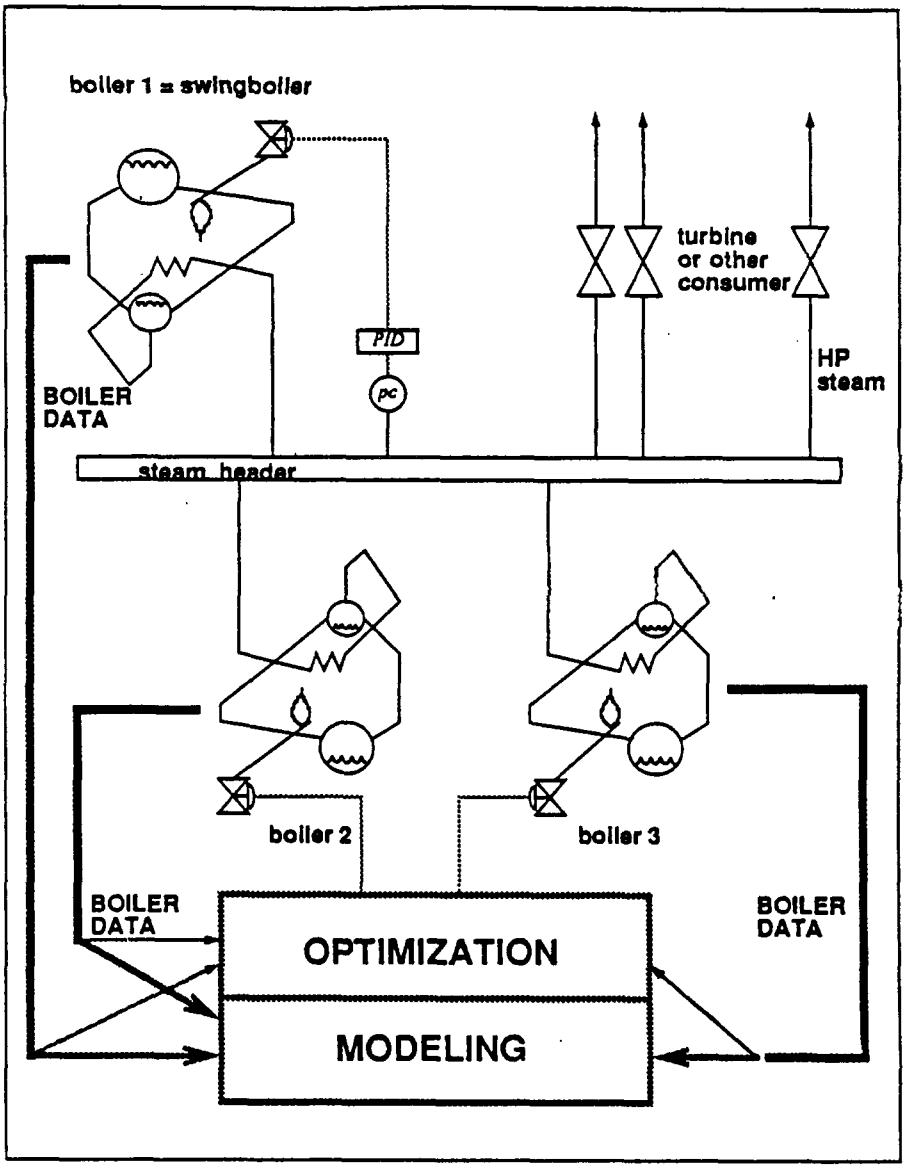

Fig.4.3.1. Boiler Network Arrangement Used in the Simulations

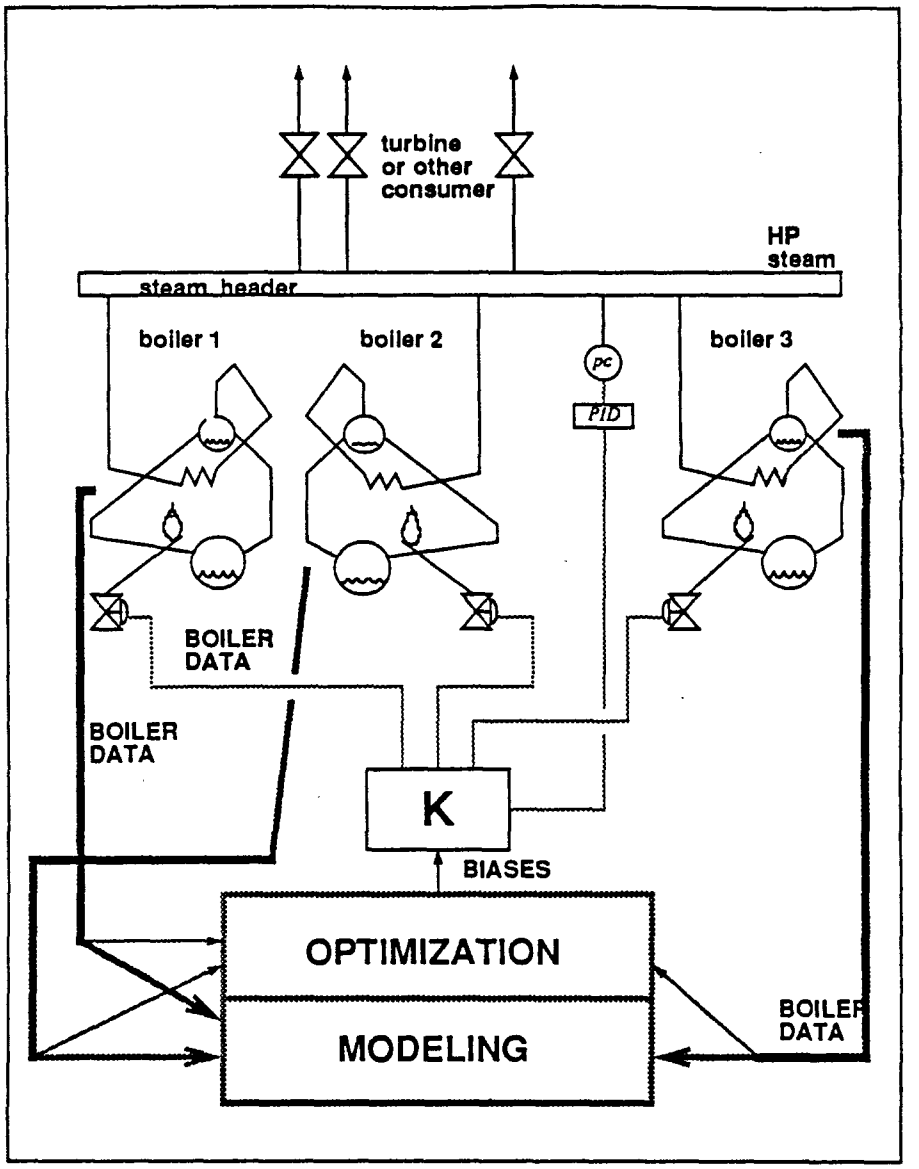

Fig.4.3.2. Robuster Boiler Network Arrangement

The boiler network arrangement is illustrated by figure 4.3.1. One of the boilers is a swing boiler. The fuel supply to the swing boiler is manipulated by the PI control of the header pressure. The fuel load of the other boilers is determined by the on-line optimizer. A more realistic scheme is presented in figure 4.3.2. There a PI controller controls the common header pressure by adjusting

the loads to all boilers. The optimizer sets biases to the allocation made by the controller. Not only provides this last method a faster response, more importantly it is more robust. If in the first scheme (figure 4.3.1) the swing boiler trips, the header pressure is not controlled any more. The scheme of figure 4.3.2 guarantees header pressure control independent of which boiler fails. The scheme of figure 4.3.1 is implemented because of it was simple without losing generality of the conclusions. The objective of the optimization is a maximization of the overall boiler network efficiency. The variables of the optimization are the individual boiler loads (optimal boiler load allocation). The swing boiler load is included in the list of optimization variables, yet is not implemented. Leaving this degree of freedom to the process loop control allows the pressure to be controlled at a constant value. The swing boiler therefore is manipulated for fast changes in steam consumption, whereas the complete set of boilers is manipulated by the on-line optimizer. If the relation between fuel consumption and steam production is known with great precision, and no fast disturbances in steam consumption changes occur, then the PI pressure control will set the swing boiler load to the same value as the optimization routine would suggest. The air to fuel ratios of every boiler can be optimized separately without interaction to the load allocation optimization. It is assumed, for simplicity, that the air to fuel ratios are constant or do not vary significantly. An equality constraint is provided by the requirement that the sum of the steam productions (or loads) of all individual boilers has to equal the total steam consumption. Inequality constraints are given by maximum and minimum boiler loads, and in some of the steady state studies constraints are formed by a maximum allowed change in load per optimization implementation.

The steady state model used by the on-line optimizer and identified by the model updater, is restricted to a very simple empirical model. For every boiler

a second order polynomial represents the relation between load and efficiency. A linear model between the fuel demand and the product of steam load and efficiency is applied. This method is standard in industrial applications, as can be found in the literature (Cho [1978], Green and Al ai-Shaikh [1980], Ko [1987]). Therefore there are five parameters per model. In the dynamic studies, three boilers are used, and therefore there are fifteen parameters that have to be updated regularly. Another short-cut method for modeling boiler efficiency as a function of load is presented in Appendix A.

The remaining sections of this chapter describe the rigorous model that is used as plant, as well as the organization of the various modules of the on-line optimizer. Finally, the interactive menu driven program that was used to conduct the case studies is described concisely.

4.3.4. Rigorous Model : The "Plant With Loop Control"

In this section, the model of every individual boiler is discussed. Then the model of the header and turbines (consumers) is given. This discussion is followed by a description of the loop controls for every boiler. Details are avoided in this presentation. The model was developed by Bertrand [1986], and more details can be found there.

4.3.4.1. The Rigorous Lumped Parameter Dynamic Model

4.3.4.1.1. Assumptions

A schematic energy and mass flow diagram of a boiler is given in figure 4.3.3. The following assumptions are made :

(1.) Energy storage is lumped and limited to *(a.)* the riser section, *(b.)* the steam drum and *(c.)* the superheater. Attemporator water (or spray water) is modeled as entering the steam at the end of the superheater, where in

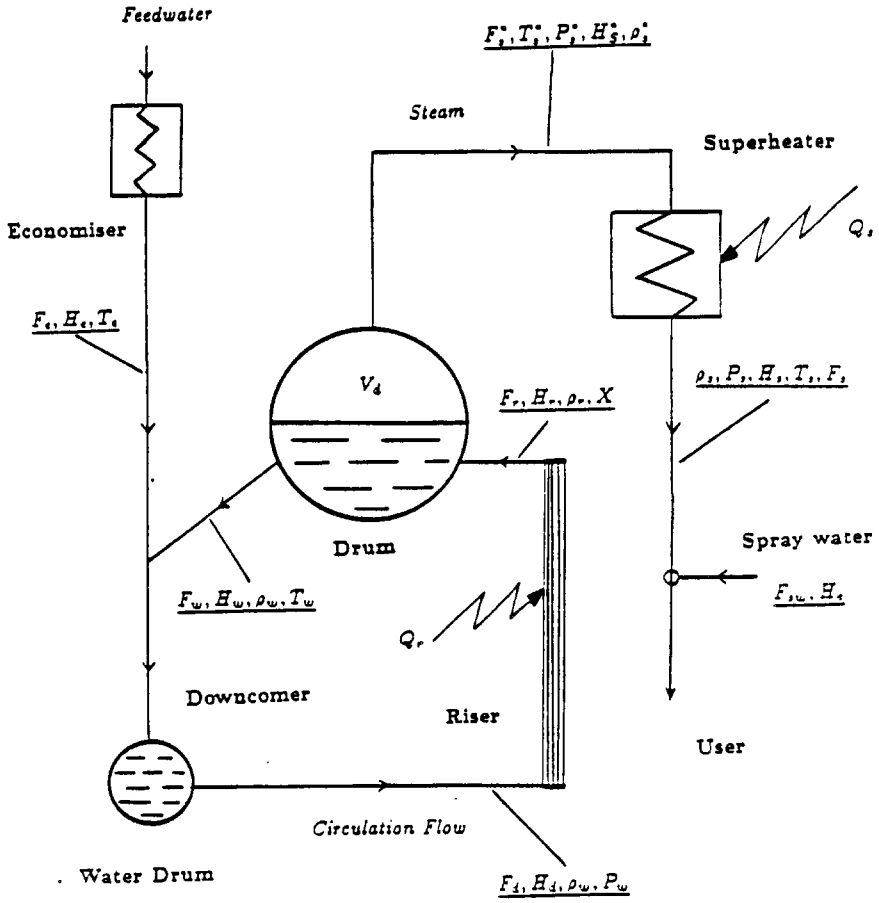

Fig.4.3.3. A Simplified Energy Mass Flow Diagram (from Bertrand et al. [1986])

reality this water is injected more upstream.

(2.) Propagation and evolution of disturbances in the mass, energy and momentum balances of the various sections of a boiler, determine the dynamic behavior.

(3.) The boiler model inputs are the fuel load (W_f), the feedwater input (F_e), the feedwater temperature (T_e).

(4.) The boiler model outputs or dependent variables are the steam states (pressure, density, temperature, enthalpy) in the steam drum, the water drum

and the superheater, and the flow conditions in every boiler section.

(5.) Liquid and vapor are in thermodynamic equilibrium in the riser and in the drum.

(6.) All feedwater enters the down comer.

(7.) Heat transfer along the down comer is negligible.

(8.) Heat and mass storage in the down comer can be neglected.

(9.) Lumping offers a sufficiently accurate approximation to the physical reality. No partial differential equations are needed.

(10.) Furnace heat losses are neglected.

(11.) Radiation heat losses are neglected by Bertrand [1986], but are introduced in this work. Details are given later. Thirty six differential and algebraic equations are developed based upon these assumptions.

4.3.4.1.2. Boiler Equations

The derivation of the boiler equations is described in full detail by Bertrand [1986], and is omitted here. We limit ourselves to a more condensed listing of the equations in the order in which they are solved. The thirty one differential and algebraic solutions for every boiler are solved in three steps.

Step 1 : Fire Box Section

The fuel flow W_f is the input to this section, and is given by the fuel controller (pressure controlling PI controller or on-line optimizer). The fire box gas temperature T_g is calculated from a simple linear relationship :

$$T_g = T_{g,ref} + k_f(W_f - W_{f,ref}). \tag{4.3.1}$$

The suffix ref refers to a reference steady state, given by Bertrand [1986]. Ten seconds first order time lag is incorporated to account for transients in the furnace temperature. Then the following eight equations are solved simultane-

ously :

$$Q_{gr} = Q_r + M_r C_{rt} \frac{dT_{rt}}{dt}; \tag{4.3.2}$$

$$Q_r = k_r (T_{rt} - T_s^{\star})^3; \tag{4.3.3}$$

$$Q_{gr} = k_{gr} W_f^{0.6} (\bar{T_{gr}} - T_{rt}); \tag{4.3.4}$$

in which $\bar{T_{gr}} = (T_{gr} + T_g)/2$ and $\star$ refers to saturated steam,

$$T_{gr} = T_g - \frac{Q_r}{C_g(1 + \frac{W_a}{W_f})W_f}. \tag{4.3.5}$$

$$Q_{gs} = Q_s + M_s C_{st} \frac{dT_{st}}{dt}; \tag{4.3.6}$$

$$Q_s = k_s F_s^{\star 0.8} (T_{st} - \bar{T_s}); \tag{4.3.7}$$

in which $\bar{T_s} = (T_s + T_s^{\star})/2$,

$$Q_{gs} = k_{gs} W_f^{0.6} (\bar{T_{gs}} - T_{st}); \tag{4.3.8}$$

in which $\bar{T_{gs}} = (T_{gs} + T_{gr})/2$,

$$T_{gs} = T_{gr} - \frac{Q_{gs}}{C_g(1 + \frac{W_a}{W_f})W_f}. \tag{4.3.9}$$

The air to fuel ratio W_a/W_f is assumed to be constant. Equation 4.3.2 is a heat balance over the riser tube. Q_{gr} is the transfer from the flue gas to the tube. Q_r is the transfer from the tube to the water in the riser. The second term in the right hand side represents the heat accumulation in the mass of the riser tubes. Equation 4.3.3 expresses the heat transfer from the warm riser tubes to the boiling water in terms of the driving force : the temperature difference. Equation 4.3.4 does the same for the flue gas side. The flue gas temperature against the riser tube wall is expressed by equation 4.3.5. That temperature difference between the bulk of the flue gas and the gas against the riser tube wall is caused by the heat transfer resistance encountered in the layer close

to the riser tubes. Equation 4.3.6 is the equivalent of equation 4.3.2 for the superheater (instead of the riser) section. In this energy balance, Q_{gs} stands for the heat transferred from the flue gas to the superheater tube. Q_s is the heat transfer from tube to steam. The heat transfer from tube to steam is written in terms of the driving force and the flow rate in equation 4.3.7. The dependency of the heat transfer coefficient to the flow velocity is taken into account. On the flue gas side, the heat transfer is described by equation 4.3.8, and the flue gas temperature close to the superheater tube wall is specified by equation 4.3.9. These eight equations (4.3.2–9) are linearized around the current state, and solved simultaneously. The differential equations are solved using an implicit Euler method. All variables are scaled using a reference steady state (given in Bertrand [1986]). In that manner, all variables have values in the same order of magnitude, which enhances numerical stability and precision.

Step 2 : Drum Section

The steam drum section is modeled by eighteen equations (4.3.10–27). Here also the equations are linearized around the current state and solved simultaneously. The first two equations are simple mass and energy balances :

$$F_e = F_d + F_w; \tag{4.3.10}$$

$$F_d H_d = F_e H_e + F_w H_w. \tag{4.3.11}$$

A mass balance and an energy balance over the riser provide two more equations :

$$F_d - F_r = V_r \frac{d\rho_r}{dt}; \tag{4.3.12}$$

$$F_d(H_r - H_d) + V_r \rho_r \frac{dH_r}{dt} = Q_r; \tag{4.3.13}$$

with the enthalpy of the steam water mixture in the riser is given by :

$$H_r = \chi H_s^* + (1 - \chi) H_{wr}. \tag{4.3.14}$$

The steam quality X is given by the weight fraction of steam in the mixture. The mixture density is given by :

$$\frac{1}{\rho_r} = \frac{X}{\rho_r^\star} + \frac{(1-X)}{\rho_w}. \tag{4.3.15}$$

Separate mass balances for steam and water in the drum give the following two equations :

$$\frac{d(V_s\rho_s^\star)}{dt} = XF_r - F_s^\star + F_{ev}; \tag{4.3.16}$$

$$\frac{d(V_w\rho_w)}{dt} = (1-X)F_r - F_w - F_{ev}. \tag{4.3.17}$$

The energy balance for the liquid can be written as :

$$\begin{aligned} V_w\rho_w\frac{dH_w}{dt} =& (1-X)F_r(H_{wr} - H_w) - F_{ev}(H_s^\star - H_w) \\ &+ \frac{P_s^\star}{\rho_w}\big((1-X)F_r - F_w - F_{ev}\big). \end{aligned} \tag{4.3.18}$$

A boiler natural circulation equation is derived by Bertrand [1986] based upon a balance of forces caused by gravity and friction. The equation is included in the drum section, and is given by :

$$\begin{aligned} &\frac{F_d^2}{\rho_w}\Big(\frac{f_dL_d}{A_d^2D_dg_c} + \frac{1}{2g_cA_d^2} - \frac{1}{g_cA_r^2}\Big) + \frac{F_r^2}{\rho_r}\Big(\frac{f_rL_r}{A_r^2D_rg_c} + \frac{3}{2g_cA_r^2}\Big) \\ &\quad + \frac{L_d}{g_cA_d}\Big(\frac{dF_d}{dt}\Big) + \frac{L_r}{g_cA_r}\Big(\frac{dF_r}{dt}\Big) + \rho_rL_r - \rho_wL_d = 0. \end{aligned} \tag{4.3.19}$$

The flashing effects in the drum can be expressed by two equations :

$$H_w = (1-y)H_{wr} + yH_s^\star; \tag{4.3.20}$$

$$F_{ev} = yF_d. \tag{4.3.21}$$

State relationships in the drum result in five equations :

$$P_s^\star = P_{s,ref}^\star + k_1(T_s^\star - T_{s,ref}^\star); \tag{4.3.22}$$

$$\rho_s^\star = \rho_{s,ref}^\star + k_2(T_s^\star - T_{s,ref}^\star); \tag{4.3.23}$$

$$\rho_w = \rho_{w,ref} + k_3(T_s^\star - T_{s,ref}^\star); \tag{4.3.24}$$

$$H_s^\star = H_{s,ref}^\star - k_4(P_s^\star - P_{s,ref}^\star); \tag{4.3.25}$$

$$H_{wr}^\star = H_{wr,ref}^\star + C_{pw}(T_s^\star - T_{s,ref}^\star). \tag{4.3.26}$$

Finally, a momentum balance is added to the list of equations :

$$P_s^\star - P_s = f_s \frac{F_s^{\star 2}}{\rho_s^\star}. \tag{4.3.27}$$

The friction factor for the superheater f_s is assumed to be constant in the operating range of our interest. Similarly to the previous section, the 18 equations presented above are scaled by a reference state before being solved simultaneously.

Step 3 : Superheater Section

From the first step, Q_s, the heat exchanged over the superheater, is known. The enthalpy of the steam in the superheater is calculated using :

$$V_s h \rho_s \left(\frac{dH_s}{dt}\right) = \frac{1}{1 - b - 2cH_s}\left(F_s^\star(H_s^\star - H_s) + \frac{P_s}{\rho_s}(F_s^\star - F_s) + Q_s\right). \tag{4.3.28}$$

This equation is the energy balance for the steam in the superheater. A state relationship is given by :

$$\frac{P_s}{\rho_s} = a + bH_s + cH_s^2. \tag{4.3.29}$$

a, b and c are constants from the state equation that was developed by Edwin Rabin [1982] :

$$Pv = a + bH + cH^2, \tag{4.3.29bis}$$

and are given in table 4.3.1. All variables except ρ_s are known in equation (4.3.29), and therefore the superheater steam density can be obtained from it. F_s is computed using the mass balance over the superheater :

$$F_s^\star - F_s = V_{sh} \frac{d\rho_s}{dt}, \tag{4.3.30}$$

and T_s can be derived from the following state relationship :

$$T_s = T_{s,ref} + \frac{(H_s - H_{s,ref}) + 0.04 \cdot (P_s - P_{s,ref})}{0.5525}. \tag{4.3.31}$$

After solving the 31 equations for every boiler, the steam header is simulated.

4.3.4.1.3. Header Equations

The mass balance :

$$V_h \frac{d\rho}{dt} = F_{in} - F_{out}, \tag{4.3.32}$$

and the energy balance :

$$V_h \rho \frac{dH}{dt} = \frac{1}{1 - b - 2cH} \Big(\sum_{i=1}^{n} F_{s_i}(H_{s_i} - H) + \frac{P_s}{\rho}(F_{in} - F_{out}) \Big), \tag{4.3.33}$$

are solved using a simple Euler method. The header pressure P_s is calculated from the state equation :

$$\frac{P_s}{\rho} = a + b + cH^2. \tag{4.3.29) $\simeq$ (4.3.34}$$

The PI controller will compare this pressure to the setpoint header pressure and manipulate W_f, the fuel load, in order to maintain the header pressure as close as possible to the setpoint.

4.3.4.1.4. Radiation

The model proposed by Bertrand [1986] does not include radiation losses. Loss of energy due to radiation does not have any important effects on the dynamics or operation of a boiler, but it does have important influence on the boiler efficiency. The absolute temperatures of a boiler do not change proportionally with the load of the boiler, but much less. The loss in energy due to radiation is therefore quite constant compared to other phenomena such as loss through the stack. Therefore, radiation losses become relatively more impor-

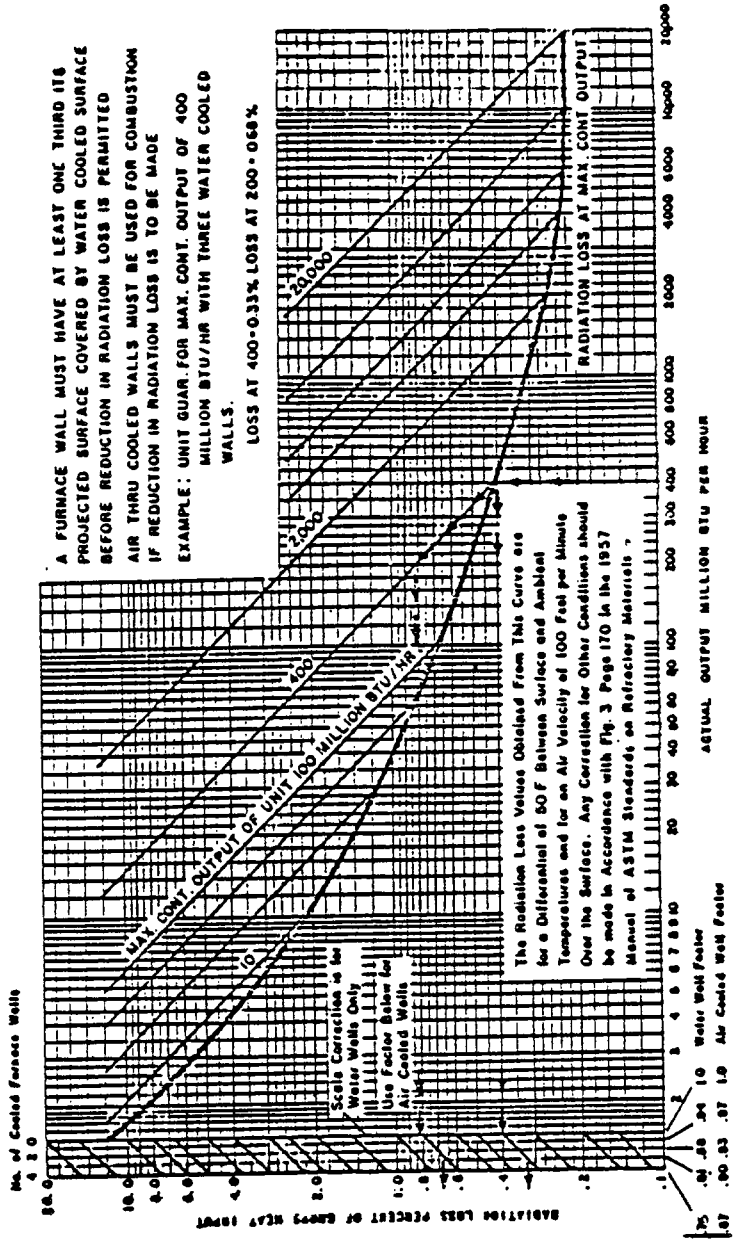

Fig.4.3.4. ABMA Standard Radiation Loss Chart (from Payne [1985])

tant as the load decreases. Radiation losses were included in the simulation that is used for this case study.

Radiation losses are charted in the ABMA Standard Radiation Loss Chart, given in figure 4.3.4. The following equation was derived from this chart, for

the type and size of boilers that is used in this case study :

$$\text{LOSS} = 4.44 \cdot W_f^{-1.25} \tag{4.3.35}$$

with LOSS in % efficiency, and W_f in lb/s. The term LOSS represents the percentage efficiency lost due to radiation. The heat generation in the fire box is reduced by that factor.

4.3.4.1.5. Consumers - Turbines

The consumers (e.g. turbines) are simulated without dynamics. The assumption is that the turbine dynamics are sufficiently fast to be neglected. Therefore, the mass flow leaving the steam header can be written as a sum over all consumers :

$$F_{out} = \Big(\sum_{i=1}^{N_{turb}} C_{turb,i} \Big) P_s. \tag{4.3.36}$$

It is assumed that the conditions of sonic flow are satisfied (Bertrand, [1986]).

4.3.4.2. Loop Control

◇ *Drum Level Control*

The water drum level is controlled, by manipulation of the feedwater flow. A simple steam to feedwater balance is implemented.

◇ *Superheater Temperature Control*

Feedwater is injected in the superheater steam to control the temperature of the steam leaving the superheater. This is often referred to as attemporator water. The effect on the pressure is negligible. It is considered as perfect control here. The required amount of attemporator water for cooling the superheated steam to the set temperature is calculated using a simple mass and energy balance, and added to the stream leaving the superheater.

◇ *Header Pressure Control*

Header pressure control was the subject of a study by Bertrand [1986]. One boiler is manipulated to keep the header pressure constant. The fuel loads of the other boilers are set directly by the on-line optimizer. Following the rules proposed by Bertrand [1986], a PI controller is used, with a controller gain of 0.055 and a reset time of $57s$ for a boiler with design specifications as listed in table 4.3.2 under *boiler 1*. The fuel load is expressed in lb_m/s and the header pressure in *psi*. This controller tuning is rather conservat've and robust, the latter of which is important to this application. The operating condition of the swing boiler can vary substantially, depending upon the action of the on-line optimizer; and a steam boiler becomes substantially slower as the load decreases.

Symbol	Value	Units
a (Rabin eq.)	-225.7667	BTU/lb_m
b (Rabin eq.)	0.28312	-
c (Rabin eq.)	$-0.2063 \cdot 10^{-4}$	lb_m/BTU
C_{pw}	1.118	$BTU/(lb_m F)$
k_1	6.05	psi/F
k_2	1/75	$lb_m/(F ft^3)$
k_3	-0.0576744	$lb_m/(F ft^3)$
k_4	0.017	$BTU/(lb_m psi)$
k_f	244.0	$(F sec)/lb_m$

Table 4.3.1. : State Relation Constants in the Rigorous Steam Model

4.3.4.3. Constants

Table 4.3.1 lists the constants used in the state relation equations listed above. Table 4.3.2 lists the design specifications of the three boilers and the header that were used in the case study. More details on the sources of these constants as well as on their ranges of validity can be found in Bertrand [1986].

Symb.	Description	Boiler 1	Boiler 2	Boiler 3	Units
		STEAM DRUM			
l	Length	26.0	26.0	26.0	ft
$2R$	Diameter	4.02	4.02	4.02	ft
V_d	Volume	330	330	330	ft^3
		SUPERHEATER			
L_{sh}	Length	134	134	134	ft
A_{sh}	Flow Area	0.89	0.89	0.89	ft^2
V_{sh}	Volume	120	120	120	ft^3
f_s	Friction Coef	6.88	6.88	6.88	$\frac{s^2 lb_f}{ft^5 lb_m}$
M_s	Tube Mass	22900	22900	22900	lb_m
C_{st}	Tube Heat Ca.	0.115	0.115	0.115	$\frac{BTU}{lb_m F}$
k_s	T/S Heat Tr.	10.43	11.43	12.43	$\frac{BTU}{lb_m F}$
k_{gs}	T/G Heat Tr.	8.13	9.13	9.53	$\frac{BTU}{(sF(\frac{lb}{s})^{0.6})}$
		HEADER V_h Volume : 1500 ft^3			

Table 4.3.2. -part2: Boilers Design Data

4.3.5. On-Line Optimizer Organization

Symb.	Description	Boiler 1	Boiler 2	Boiler 3	Units
		STEAM DRUM			
l	Length	26.0	26.0	26.0	ft
$2R$	Diameter	4.02	4.02	4.02	ft
V_d	Volume	330	330	330	ft^3
		down comer			
L_d	Tube Length	24	24	24	ft
A_d	Flow Area	5.04	5.04	5.04	ft^3
D_d	Tube Diam.	0.1396	0.1396	0.1396	ft
f_d	Friction Coef.	0.0443	0.0443	0.0443	
H_e	Feedw. Enth.	241.4	241.4	241.4	$\frac{BTU}{lb_m}$
		RISER			
L_r	Length	24	24	24	ft
A_r	Flow Area	9.61	9.61	9.61	ft^2
D_r	Tube Diam.	0.1122	0.1122	0.1122	ft
f_r	Friction Coef.	0.0443	0.0443	0.0443	
V_r	Volume	230.64	230.64	230.64	ft^2
M_r	Tube Mass	49600	49600	49600	lb_m
C_{rt}	Tube Heat Ca.	0.115	0.115	0.115	$\frac{BTU}{lb_m F}$
k_r	T/L Heat Tr.	0.0722	0.0822	0.0822	$\frac{BTU}{sF^3}$
k_{gr}	T/G Heat Tr.	7.621	8.621	6.621	$\frac{BTU}{(sF(\frac{lb}{s})^{0.6})}$
C_g	Gas Heat Ca.	0.25	0.25	0.25	$\frac{BTU}{lb_m F}$
$\frac{W_a}{W_f}$	Air/Fuel	16.58	16.58	16.58	

Table 4.3.2.-part1: Boilers Design Data

The on-line optimizer functions as presented in Chapter 3 are split up

into two asynchronously executed phases. The data processing phase checks continuously (or, more precisely, at a high frequency) for a steady state. If a steady state is detected, then it is checked whether or not the steady state is significantly different from the ones already stored. If so, the new steady state is stored. When a new steady state is stored, the model is reidentified. Therefore, the estimation phase is not activated at a fixed frequency, but whenever new data becomes available. Both equations of the simple steady state model the on-line optimizer are linear in the parameters. Therefore, the five parameters per boiler can be identified without iteration.

The optimization routine is activated at a fixed frequency. This frequency is very slow compared to the frequency at which the process is checked for steady state. The total network load is estimated from the total measurement production. The optimal boiler load is calculated, and a sensitivity analysis is added to this. If required, an optimization objective or setpoint accuracy check is performed and the results are only implemented if the accuracy is sufficient.

4.3.6. A Simulated Experimental Set-Up : "STEAM"

A program called "STEAM" was written to improve the organization of the numerous and sometimes lengthy simulations that have to be performed while studying this case. The program can run in 3 different modes : TEST, BATCH and INTERACTIVE. A VMS macro communicates to the program which mode it has to run in. Figure 4.3.5. schematically presents the organization of the program. The program is written in FORTRAN, but uses VAX-VMS dependent routines. The code consists of more than 14000 lines ($\frac{1}{2}$ Mbyte), and makes also extensive use of library software for estimation, optimization, matrix manipulations and screen output. The executable (MicroVMS, Version 4.2) measures about 300 kbyte.

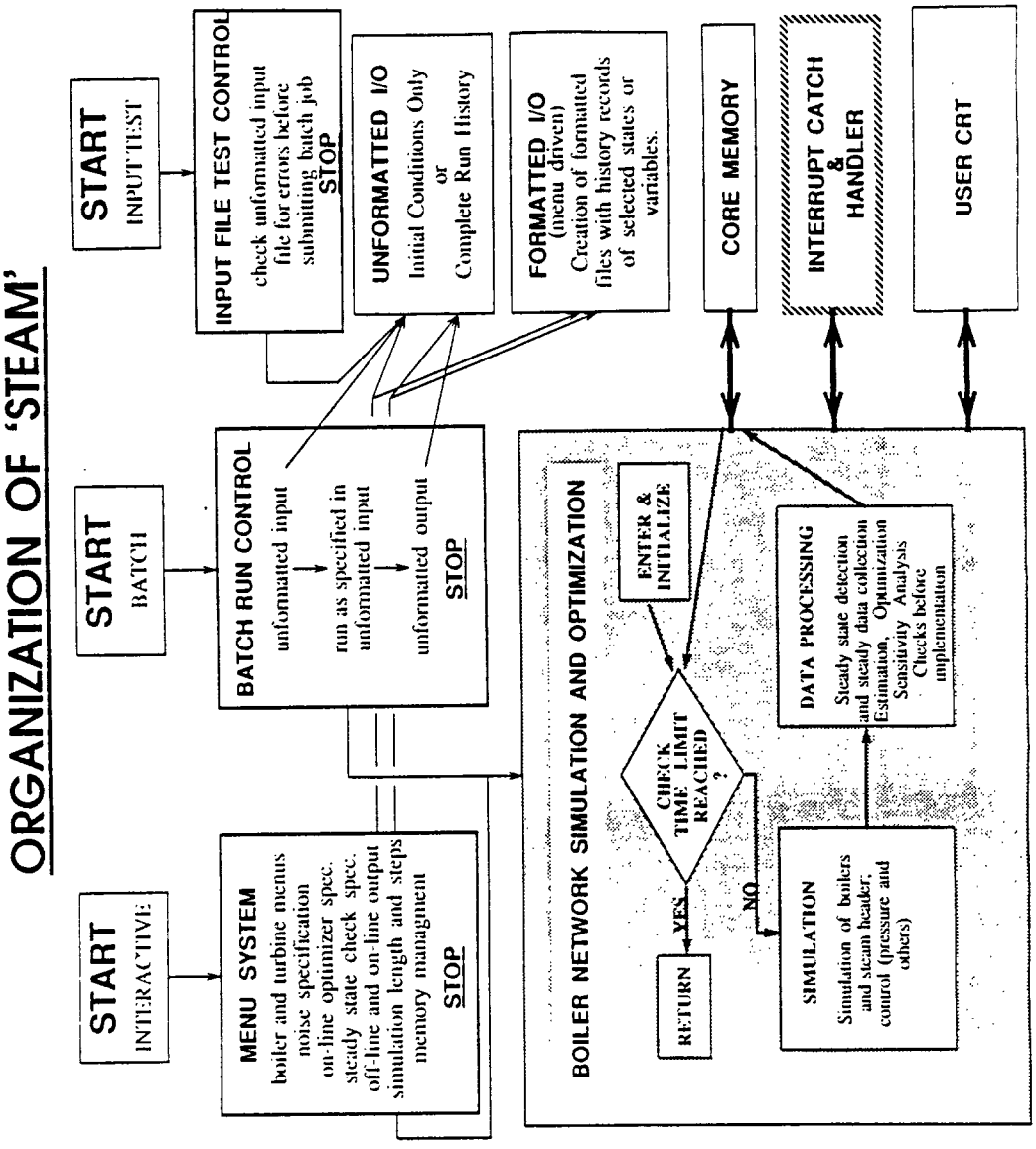

Fig.4.3.5. Schematic Representation of the Organization of the Program "STEAM"

4.3.6.1. The Interactive Mode

The interactive mode is menu driven, and makes extensive use of the VAX Screen Management Facility. The organization of the menus is schematically represented by figure 4.3.6. The main menu offers seven choices. The last

two choices return the user to the operating system. The last choice stops the program; this is a definite exit of the program after user acknowledgement. The next to last choice allows the user to suspend the program, and create a subprocess in the operating system. In that way the user can execute all system commands in that subprocess while the program is actually in suspension. By logging out of the subprocess, control is returned to program "STEAM", and the main menu is restored. Option five starts the simulation, and in the interactive mode a frequently updated display will be generated on the user CRT. At any moment in time, the simulation can be interrupted. The simulation will be suspended, and the user is given the options to restart the simulation with or without switching display, or to stop the simulation. More than one frequently updated display is kept in memory. The one of choice can be brought to the foreground. Other options are concisely described in figure 4.3.6.

The disk I/O deserves some more attention (options 3.3 and 3.4 on figure 4.3.6). The menu system can be used to set up an "experiment". This set-up only, or the set-up and run results can be stored in an unformatted file. The program itself can read those files on a later date. In that way condensed records can be maintained of run set-ups and results. Formatted output can be extracted from the unformatted files using the program itself (the menus only, no simulation is started up). Formatted output can be used for user reference or as intermediate towards graphic output. The unformatted storage is also used in batch mode.

4.3.6.2. The Batch Mode

Longer runs, or a sequence of runs, can be executed in the background, for instance at night, or during the day at a lower priority than interactive processes. In advance, the user has to prepare one or more unformatted set-up files, using the interactive mode and the menus. After that, he has to start the

Fig.4.3.6. Menu System Summary

VMS macro that controls the operation of the program STEAM in the "bg" or "background" mode. The macro will ask the user the name of the unformatted

set-up file. Immediately the file will be tested (the third mode of "STEAM") by a short test reading the macro orders for the program to carry out. If successful, the user is prompted for the name of the output file. Then a job is submitted, and started immediately or held until the time the user specifies. The "STEAM" control macro itself writes a command macro for every job.

The batch job macro starts up "STEAM" in the batch mode. In that mode, no output is generated during the run. The program reads the unformatted set-up file, and starts the simulation immediately. All information is kept in core, and no disk storage is used during the simulation. This is also true for the interactive operation mode. Disk operations only occur before or after the simulation and on explicit request of the user. That way, no unnecessary files are created, and the program is time and disk space efficient. The CPU time requirement for simulating N time units of operating time, is approximately equal to $0.58 \cdot N$ time units on a MicroVax II under MicroVMS Version 4.2. If no interactive users or other high priority processes are on the system (e.g. at night) this will take approximately $0.63 \cdot N$ true time units on a MicroVax II under MicroVMS Version 4.2. After completing of the simulation, "STEAM" in batch mode will dump all results of the run into an unformatted result file. A formatted log file will contain some statistics and start/stop time information of the run.

Later, the user can check the batch queue, and the job log file. If the run was completed successfully, the user can use "STEAM" in the interactive mode to open and read the unformatted result file, and select variables for formatted output, or even continue the run after adjusting certain settings.

4.3.7. Boiler Efficiency

4.3.7.1. Factors Influencing Boiler Efficiency

Within the limits of "normal operation", the boiler load is *not* the most important factor influencing boiler efficiency. A very important factor is without any doubt the steam generating unit design. Boilers that are equipped air preheater and economizer can be 5 to 8% more efficient than units without these heat recovery systems. This certainly applies for larger units (Payne, [1985]). Losses due to incomplete combustion can be considerable. Excess air can improve this combustion efficiency, but will increase losses through the stack. It was mentioned before that the fuel to air ratio was held constant in the simulations. Excess air optimizations can be done without interaction with the boiler load allocation optimization. Excess air optimization can be solved on a boiler by boiler basis, and thoroughly tested procedures are available (Payne [1985], Jackson [1987], Lipták [1987]). A potential improvement of 5% through excess air optimization is listed by Garcia-Borras [1983]. Lipták [1987] lists 2%. Green and Al ai-Shaikh [1980], in a publication on improving boiler complex control, report efficiency improvements of 4 to 9% through improved digital control. Only 1 to 3% is reported by the same authors as due to optimal boiler load allocation. Adequate soot removal can improve boiler efficiency. Jackson [1987] mentions that a $\frac{1}{8}$ inch thick layer of soot on the heat exchange surfaces in the boiler can increase fuel consumption with up to 13% for an identical boiler load. Garcia-Borras [1983] lists potential savings of 2 to 9% due to proper soot blowing. Garcia-Borras [1983] also mentions that presence of iron and silica in the soot make the soot layer an even larger resistance to heat transfer. The same author also suggests fuel additives. Other important factors are blow down, header pressure optimization, as well as monitoring of the work of the feedwater pump of the condensate return system and potential forced convection fans (Lipták [1987]).

All this indicates that boiler load allocation, in the best possible case, will

only make sense if all other factors determining boiler efficiency are optimal. More gains can be made by optimizing the excess air and by careful operation including proper soot blowing and blow down, than can be expected to be made by optimum boiler load allocation. This point will get much attention in the discussion of the short-cut feasibility study method as applied to boilers, in the next chapter.

4.3.7.2. Efficiency Curves for 3 Boilers

Figure 4.3.7 gives the steady state efficiency as a function of individual boiler load for the three boilers of which the design variables are listed in table 4.3.2. The curves have a similar shape. Towards higher loads, the boiler has to exchange more and more heat over the same heat exchange surfaces. Therefore, the mean temperature differences over the heat exchange surfaces of the riser and the superheater have to be increased. This results in a higher stack temperature. Not only in absolute figures, but also in a relative sense more energy is lost to the environment. Therefore the efficiency decreases towards high loads.

Losses due to radiation do not decrease proportionally with the boiler load. Therefore, they become relatively more and more important towards lower loads. This is illustrated in figure 4.3.8. Finally, towards low loads, the relative influence of radiation becomes the main cause for efficiency loss (Payne, [1985]).

Not all boilers have the same shape of load to efficiency curve. It is for instance possible that the minimum load is higher than the load for which stack loss and radiation effects compensate for each others. In that case there will not be a load with maximal efficiency, but the efficiency will decrease monotonically with increasing load. The shape of the curve that expresses efficiency versus boiler load, is however mainly function of the type of fuel that is being used.

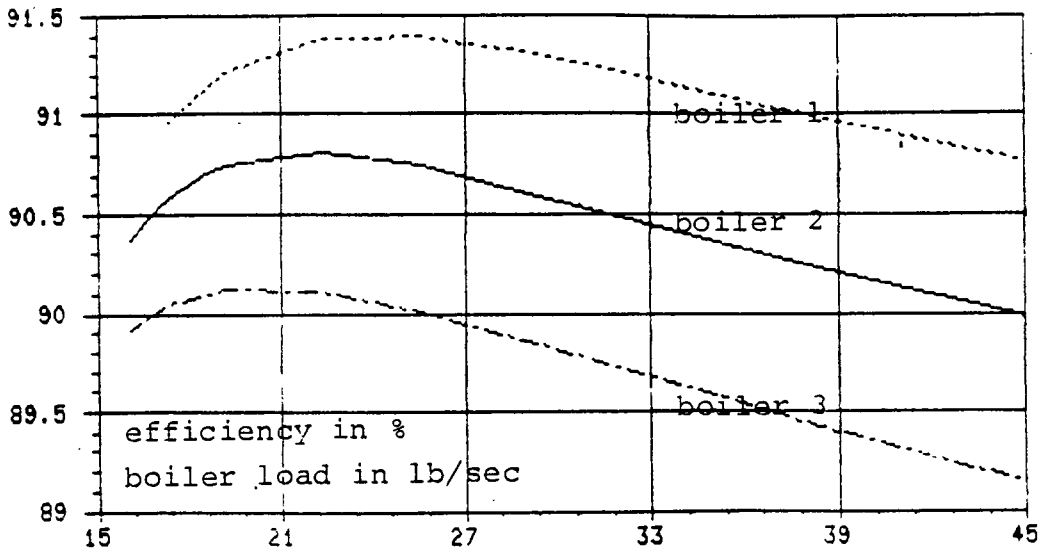

Fig.4.3.7. Individual Boiler Efficiency as a Function of Individual Boiler Load

Figure 4.3.9 illustrates this point.

4.3.7.3. Fit of Steady State Model to Plant

Steady state model fit will be discussed in Chapter 5 in more detail. The on-line optimizer uses a second order relation between boiler efficiency and boiler load as steady state model. Here, we limit ourselves to pointing out that it is clear that the curvature (the second derivative) of the curves on figure 4.3.7 is not constant, as it is for a second order polynomial. Figure 4.3.10 shows the fit of second order polynomials to five points of the curves on figure 4.3.7, representing true efficiency versus boiler load data points. The *systematic* mismatch can be seen to be repeated in all three curves. If more high load data are present in the data set, then typically low load efficiency predictions will be significantly over predicted. This point will be clear in case study results listed in the next chapter. A less arbitrary way to model boiler efficiency as a function of boiler load is suggested in Appendix A.

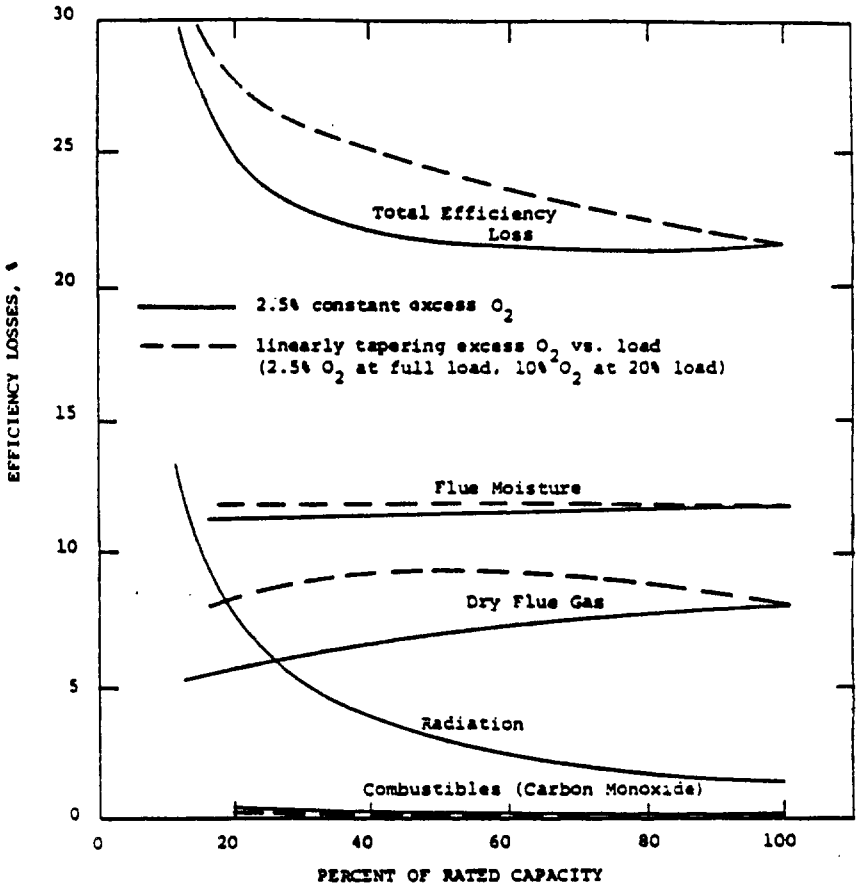

Fig.4.3.8. Variation in Boiler Efficiency Loss (in Particular Radiation) with Changes in Boiler Load (from Payne [1985]).

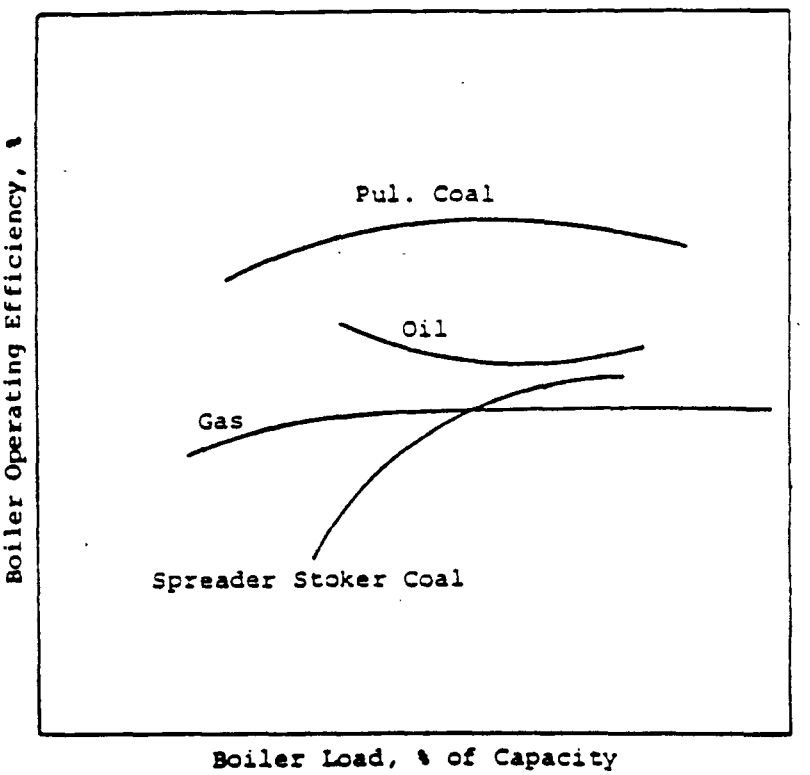

Fig.4.3.9. Typical Efficiency versus Load Dependence (from Payne [1985])

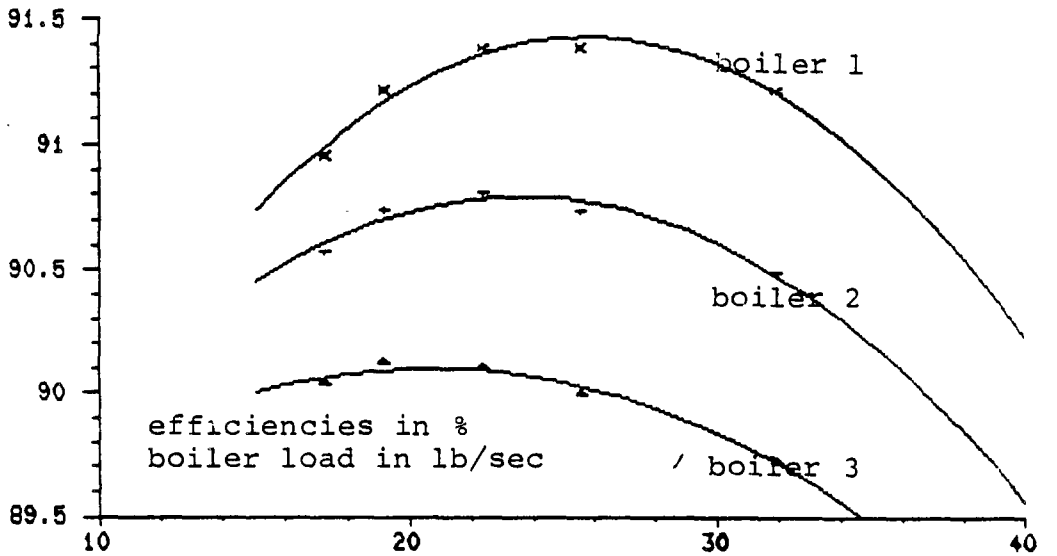

Fig.4.3.10. Fit of Second Order Polynomials to Boiler Efficiency Data.

4.4. HEAT EXCHANGER NETWORK LOAD ALLOCATION

4.4.1. A Candidate for On-Line Optimization

Optimal heat exchanger load allocation is the third and final system that is used for a case study in this work. Often various sources of heat are available in a process. These warm streams can change in temperature and flow in time, and this is also the case for cold streams. Therefore, obtaining an optimal combination of heat and cold streams can improve the efficiency of the heat integration of a plant. This is not only an important issue during the design of the plant (e.g. Duran and Grossmann [1986]), but also during the operation of an existing plant.

4.4.2. A Case Study

Heat exchangers are probably the most widespread units in chemical processes. Literature on heat transport and heat exchanging equipment is extensive. Models with varying degree of complexity are available, and therefore a simulation study can be set up with a more rigorous model that functions as "plant", and a simpler model to be used by the on-line optimizer. Heat exchangers are relatively simple units. The physical phenomena that take place in many heat exchangers are well understood. A simple network example allows to set up a realistic case study of which the behavior can easily be interpreted.

4.4.3. Example Description

4.4.3.1. Arrangement

The example that is described in this section, is inspired by a heat exchanger that is described in example 7.3 of Kern [[1950] (p.151). A crude oil

flow is heated before it enters the crude distillation tower. The heat in the warm kerosene flow leaving the bottom of the distilling column is used for preheating the crude oil flow before it enters the furnace. A nominal flow of 298000 lb/h of 34° API mid-continent crude oil is the cold stream entering at 100F. There are two warm streams of 42° API kerosene, with the same nominal conditions. Both have a nominal flow of 43800 lb/h, and enter at a nominal temperature of 390F. The crude oil stream is split over two heat exchangers, with the same design but not necessarily the same degree of fouling, and in each heat exchanger the cold crude oil exchanges heat with a warm kerosene stream. The nominal exit temperature of the crude oil is 170F. For the kerosene, the nominal exit temperature is 200F. Figure 4.4.1 shows a scheme of the heat exchanger combination.

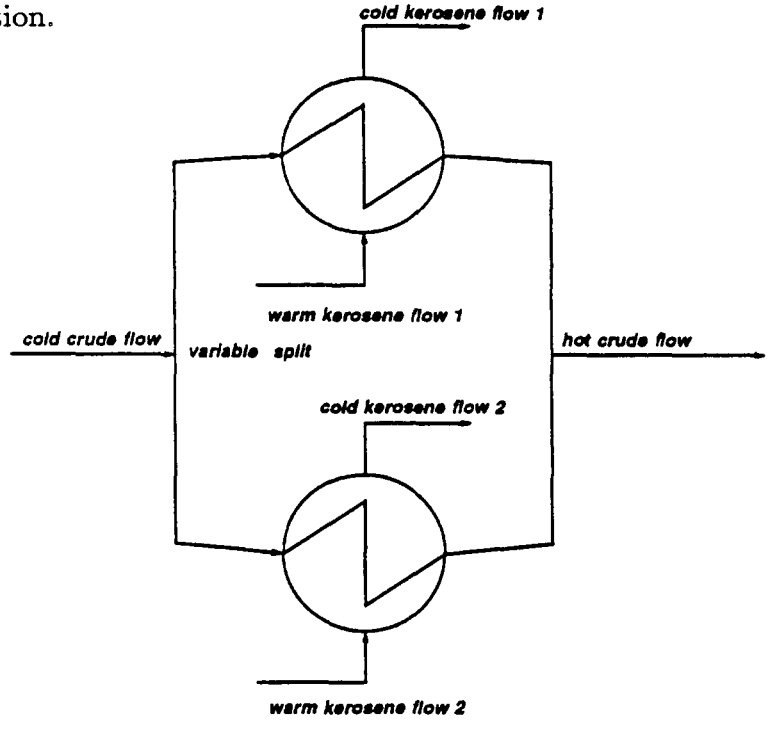

Fig.4.4.1. Heat Exchanger Network

During the operation of the plant, the temperature and flows of the warm streams and the cold stream will deviate from the nominal values. In order

to use the heat source optimally, the split of crude oil flow between both heat exchangers will have to be adjusted as flow and temperature conditions change. Optimal use of the heat source is the objective of the optimization in this case study.

4.4.3.2. Description of the Heat Exchangers

It was mentioned before that both heat exchangers have the same design. The heat exchangers are shell and tube heat exchangers. The internal diameter of the shell is 21 $\frac{1}{4}$ inch. The tube bundle consists of 158 16 inch tubes with 1 inch internal diameter. The tubes are arranged in a 1 $\frac{1}{4}$ inch square pitch, and the shell side stream makes four passes across the bundle with baffles 5 inches apart. The tube side stream passes only once. The kerosene passes through the shell, and the crude oil passes through the tubes.

4.4.4. Models Used

4.4.4.1. Rigorous Model : "Plant"

In this section, the equations that are used in a rigorous steady state model that will serve as the plant in this simulation case study, are presented in the order they are solved. For more details on the validity of these equations, the reader is referred to the extensive literature on heat transport, i.e.: Kern [1950], Bird *et al.* [1960].

◇*Step 1 : Obtain the Mass Velocity*

$$\text{Shell Side} : G_S = \frac{F_S}{A_S} \qquad \text{with} : A_S = 0.1475 ft^2 \tag{4.4.1}$$

$$\text{Tube Side} : G_W = \frac{F_W}{A_W} \qquad \text{with} : A_W = 0.1410 ft^2 \tag{4.4.2}$$

◇*Step 2 : Obtain the Reynolds Numbers*

$$\text{Shell Side} : Re_S = \frac{D_S G_S}{\mu_S} \qquad \text{with} : D_S = 0.0825 ft \qquad \text{with} : \mu_S = 0.97 \frac{lb}{ft \cdot hr} \tag{4.4.3}$$

$$\text{Tube Side} : Re_T = \frac{D_T G_T}{\mu_T} \qquad \text{with} : D_T = 0.0675 ft \qquad \text{with} : \mu_T = 8.70 \frac{lb}{ft \cdot hr} \tag{4.4.4}$$

The viscosities μ_S and μ_T are assumed to be constant in the range of operation considered in this study.

◇*Step 3 : Obtain the Chilton-Colburn j-Factors*

$$\text{Shell Side} : j_{HS} = 0.0473 \cdot Re_S^{0.748} \tag{4.4.5}$$

$$\text{Tube Side} : j_{HS} = 2.359 \cdot 10^{-26} \cdot 10^{12.823 \log Re_T - 1.5060 (\log Re_T)^2} \tag{4.4.6}$$

◇*Step 4 : Obtain the Heat Transfer Coefficient*

$$\text{Shell Side} : \frac{h_S}{\Phi_S} = j_{HS} \frac{k}{D_S} \left(\frac{c_{\mu S}}{k} \right)^{\frac{1}{3}} \Rightarrow h_S = 1.736 j_{HS} \tag{4.4.7}$$

$$\text{Tube Side} : \frac{h_T}{\Phi_T} = j_{HT} \frac{k}{D_T} \left(\frac{c_{\mu T}}{k} \right)^{\frac{1}{3}} \Rightarrow h_T = 3.908 j_{HT} \tag{4.4.8}$$

◇*Step 5 : Calculate the Overall Heat Transfer Coefficient*

$$U_o = \frac{h_S h_T}{h_S + h_T} \tag{4.4.9}$$

◇*Step 6 : Correction for Shell - Tube*

$$\text{for the given configuration } : U = 0.905 * U_0 \tag{4.4.10}$$

◇*Step 7 : Heat Transferred*

$$Q = U \cdot A \cdot d \cdot \Delta T_{ln} \tag{4.4.11}$$

The total heat transfer area equals 662 ft^2. The fouling factor d has a value 0.8052 at nominal conditions.

The simulation of a heat exchanger using these equations includes an interaction. The logarithmic mean temperature difference is estimated first. Based upon that Q is calculated, which results in a corrected logarithmic mean temperature difference estimate. The calculation is completed when successive approximations of the logarithmic mean temperature difference are sufficiently close together.

4.4.4.2. Simplified Model

A much simpler model is used by the on-line optimizer. In this model the product of the total heat transfer U, the total heat exchange surface A and the fouling factor d is lumped into one parameter. This parameter will be noted $\wp$. The simple model can therefore be written as :

$$Q = \wp \Delta T_{ln}. \tag{4.4.12}$$

The parameter $\wp$ has to be obtained by the model identification routine.

CHAPTER 5

DESCRIPTION AND DISCUSSION OF RESULTS

5.1. INTRODUCTION

In this chapter, results from the case studies are described and discussed. Since on-line optimization is central in this work, the results are handled primarily by subject, and not by case study.

First, results relating to data reconciliation are given (section 5.2). Results on model updating are discussed next (section 5.3). The results on sensitivity analysis follow in section 5.4. The discussion of these results is important, since an important contribution of this work is contained in applications of sensitivity. In a study of the dynamic interaction between an on-line optimizer and the plant, a simple CSTR reactor with two consecutive reactions is used as example. Since the CSTR system is used only for that result, the system is described there and not in Chapter 4. Also the boiler case study is used to obtain results on the dynamic interaction between plant and on-line optimizer. These results are given in section 5.5, together with results on the influence of noise on the on-line optimizer performance.

5.2. DATA RECONCILIATION

In this section, data reconciliation techniques are illustrated. The first set of results that is presented, is obtained from the distillation case study. These results illustrate how data reconciliation influences model parameter estimation. The second set of results is obtained from the heat exchanger case study. That set of results considers the possibility of splitting the data reconciliation of a system into reconciliation of smaller systems. The data reconciliation algorithms that are used in the distillation case study and in the heat exchanger example are described in detail in appendices D and E. Some conclusions on data reconciliation follow the results obtained from the heat exchanger case study.

5.2.1. Distillation

5.2.1.1. Introduction

The parameter estimation of the Eduljee model (estimation of α and N) only needs the compositions of the feed, the bottom and the top stream. The stream flow rates are not required for the estimation. The top, bottom and feed purities are assumed to be available. The feed stream is the product of another column, and is therefore also assumed known. However, measurements contain inaccuracies such as noise and bias, that in turn will result in inaccuracies in the parameter estimates and in the results deduced from the parameter estimates. Therefore, if reliable but redundant information can be applied to reduce the inconsistencies due to inaccuracies in measurements, then, in principle, the precision of the estimates and of the calculations that rely upon these estimates can be improved. Systematic errors can be detected as well. This opportunity is realized in this case study through the use of data reconciliation and gross

error identification.

The performance of data reconciliation with respect to the parameter estimation and the optimization is illustrated in three examples. In the first example it is illustrated that data reconciliation can improve data quality even if the data do not contain any gross errors and if the noise levels are fair. The next example shows that data reconciliation is not without risk. Gross errors in redundant data can leak through to non-redundant data. A last example shows that gross error identification can be very useful and improve estimation performance considerably.

Let F be the mass feed flow, B the bottoms mass flow, D the distillate mass flow, x_F the mole fraction of light component (propylene) in the feed flow, x_B the mole fraction of light component in the bottoms flow and x_D the mole fraction of light component in the distillate. The following equations are used by the data reconciler in this case study :

$$F = D + B \tag{4.2.13}$$

$$Fx_F = Dx_D + Bx_B \tag{4.2.14}$$

In section 3.3.4 it is pointed out that the equations that are used in the data reconciliation could introduce relationships in the data that potentially make the parameter estimation criterion singular. The equations 4.2.13–14 do not introduce a fixed relationships between the feed, top and bottom composition that are only used in the fitted model. It was mentioned above that the parameter estimation does not even use any flow information. Therefore the mass balances can be used in the data reconciliation step without jeopardizing the performance of the parameter estimation.

The data reconciliation technique applied in this case study is presented by Serth *et al.* [1986] as the Modified Iterative Measurement Test (MIMT). Serth

et al. [1986] characterize this reconciliation method as effective and reliable, and computationally less expensive than other methods (such as the screened combinatorial test) they compared with the MIMT method. A measurement test method allows gross errors to be directly identified without a separate identification procedure (Iordache *et al.* [1985]). The data reconciliation algorithm is presented in Appendix D. The MIMT method as presented by Serth *et al.* [1986] is slightly adjusted for the case under consideration. In Appendix D it is also described how this algorithm identifies gross errors, and how it reconstructs data that are rejected because of gross error from redundant measurements. Can reconstructed data be used as a base for model updating ? Gross error identification techniques typically check for one gross error at a time. If a gross error is detected, then it is known that this measurement is unreliable. In all techniques it is assumed that only one gross error is present or that if more than one gross error is present, that they do not influence each others detection. This cannot be guaranteed in reality. While a gross error identification technique looks for the most unreliable measurement, it does not say that, if a gross error is found, the other measurements are reliable. Only the measurements that are *most* improbable with respect to the other measurements and the reconciliation equations are identified. So, it is true that nothing is known about the reliability of a measurement that has not been determined to be unreliable unless no gross errors are detected. A measurement that is detected by a gross error, is replaced by a reconstruction. The reconstruction is of course a strict function of the remaining data. The relation is given by the reconciliation equation. The quality of the reconstructed value depends upon the reliability of the data that was used to reconstruct it. If no more redundancy remains, the quality of the reconstructed data cannot be checked. Therefore reconstructed data occasionally can be unreliable. If a certain measurement is inaccurate, then this

very often has an unknown cause. Somewhere in the long route from sensing to the process computer something may have gone wrong. If the data from which this inaccurate measurement is reconstructed are channeled partially over the same route as the inaccurate measurement itself (transmission line, amplifiers, multiplexers) then chances increase that the data used for reconstruction are faulty too. Sometimes it also happens that due to an outlier in the measurement noise, the data reconciliation algorithm determines a measurement that is correct and that is intimately related to the actual wrong measurement, as containing a gross error. In general, reconstructed output from a measurement test reconciliation algorithm should be used with care.

5.2.1.2. Examples of the Performance of the Data Reconciliation

5.2.1.2.1. Random Errors

In this example and in the following two examples the previously discussed estimation mode operating in the $\ln\alpha$, $\frac{1}{N}$ space is used. Random errors with a standard deviation of five percent of the nominal value for the streams (F, B and D) and the bottom propylene fraction (x_B) and five percent of the top and feed (x_F, x_D) propane fraction (1. - propylene fraction) is added to one hundred data points, with inputs randomly chosen around the nominal steady state, and with corresponding plant outputs simulated using the Eduljee and Fenske equations. Figure 5.2.1. gives a schematic representation of how the data are generated. More details on this run can be found in Appendix G (Run # 1). The parameter estimation finds :

$$\begin{cases} 1.1058 \leq \alpha \leq 1.1064 & \text{(center:1.1061)} \\ 99.0 \leq N \leq 107.4 & \text{(center:103.0)} \end{cases} \tag{5.2.1}$$

If data reconciliation is applied, and all data sets for which the reconciler determined a gross error are discarded, then the result of the parameter estimation

becomes :

$$\begin{cases} 1.1141 \leq \alpha \leq 1.1144 & \text{(center:1.1143)} \\ 79.7 \leq N \leq 91.1 & \text{(center:85.0)} \end{cases} \tag{5.2.2}$$

Both results are very comparable, but from figure 5.2.2 it seems as if the result from the estimation based upon the reconciled data is better than the other results. In figure 5.2.2 the point marked U corresponds to the results obtained with unreconciled data. R refers to reconciled data. R is closest to the equivalence curve for nominal conditions which is discussed in section 5.3.1.3.1 and in Appendix B, and can therefore be considered as the best performing result. This equivalence curve collects all combination of parameters that will give the same simulation result. More details on equivalence curves can be found in the next section on results in model identification. T refers to the parameters the data were generated with.

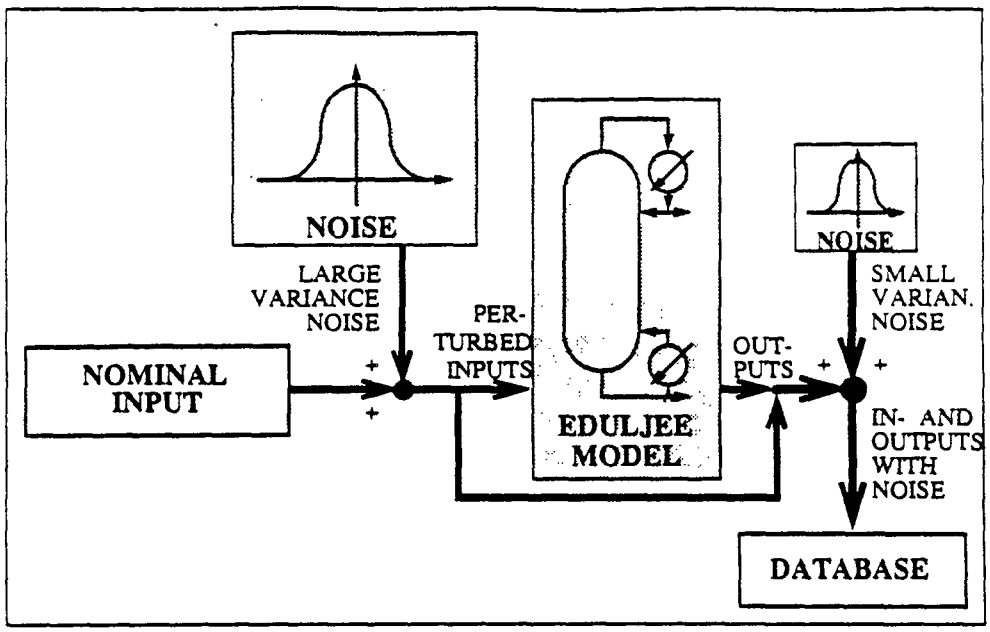

Fig.5.2.1. Schematic Presentation of Data Generation

The better parameter estimation result that is obtained using the reconciled data set, results in lower variable operating cost after optimization. If

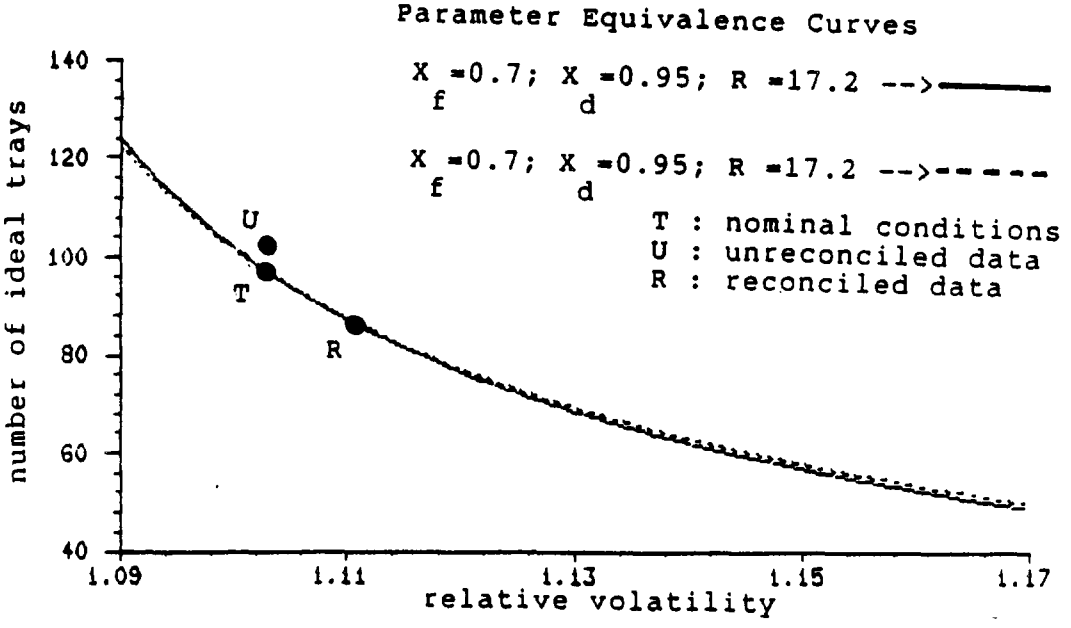

Fig.5.2.2. Performance Comparison for Parameter Estimation Techniques

no data reconciliation is used (results in 5.2.1), the optimizer determines that for nominal feed the optimal reflux ratio is 15.67. That results in a variable operating cost of the actual column (with $\alpha = 1.105$ and $N = 94$) of \$ 5270 per day for nominal feed. If the reconciled data are used (results in 5.2.2), then the optimizer determines that for a nominal feed the optimal reflux ratio is 16.05. That results in a variable operating cost of the actual column of \$ 5170 per day. The true optimum operating cost is \$ 5100 per day at a reflux ratio of 16.93. The application of data reconciliation results in a gain of \$ 100 per day. The data reconciliation reduces the offset in optimized variable cost by more than 60%.

Another difference between the results in set 5.2.1 and the results in set 5.2.2 is that the parameter estimates of the reconciled results have confidence intervals that are much more narrow than the confidence intervals of the param-

eter estimates that are based upon raw data. This is of course a direct effect of the data reconciliation, which effectively reduced the noise on the data. If data reconciliation is used before the parameter estimation, then in the process of obtaining parameter estimates more plant knowledge and more plant data are used than if no data reconciliation is used. In general, one can expect that if this plant knowledge and these plant data are correct, their use will reduce the uncertainty of the parameter estimates. The correct confidence intervals can only be obtained if data reconciliation and parameter estimation are carried out combined. However, combined data reconciliation and parameter estimation demands a large amount of computer time. The obtained confidence intervals are approximations.

Data sets with one or more gross error are discarded in this example. The number of data points that is discarded by the data reconciliation is 9. The confidence limit used was 95% which is rather small for detecting unreliable conditions. A 5% chance exists that a correct condition is recognized as wrong (type 1 hypothesis test error). It is often better to use a 99% confidence. The probability for a type 1 testing error is then only 1%. In most practical situations the increase in probability of recognizing incorrect conditions as correct (type 2 hypothesis testing error) by using a 99% confidence instead of a 95% confidence, is not expected to be very large. For consistency 95% was used in the next two examples also.

5.2.1.2.2. Systematic Error

This example shows that it is possible that data reconciliation is responsible for a loss in quality of the parameter estimates. The parameter estimation only uses the three compositions of the feed stream, the top or product stream and the bottom stream. Data reconciliation checks these composition data, using flow measurements. The total set of data is then reformed to the most probable

one with respect to the raw data and the consistency with the data reconciliation equations. This technique is intended to reduce errors, but it can also cause the introduction of errors. For instance, if the flow measurements have some error that is not always detected, then these will result in incorrect reconciled composition data, and hence the parameter estimates will deteriorate. In this example the same set of raw data is used. However in approximately half of the data sets (randomly chosen) a 20% error (positive or negative) was added to the feed flow data. More details on this example can be found in Appendix G (Run # 2). Of course the estimation result that uses the raw (unreconciled) data is the same with or without the added error in the redundant flow data. The estimation does not use the flow data. The results from unreconciled data are :

$$\begin{cases} 1.1058 \leq \alpha \leq 1.1064 & \text{(center:1.1061)} \\ 99.0 \leq N \leq 107.4 & \text{(center:103.0)} \end{cases} \tag{5.2.1}$$

If the data reconciliation technique is used, then the errors in F that slip through the gross error detection are spread over the composition data, reducing its quality. The result is definitely poorer than the one based upon raw data. The use of data reconciliation would result in a loss of more than $ 100 per day for a nominal flow load in composition compared to not using data reconciliation. In this example also, no data with detected gross error were accepted. Sixty data sets remained after the reconciliation. The results are :

$$\begin{cases} 1.114 \leq \alpha \leq 1.116 & \text{(center:1.115)} \\ 79.0 \leq N \leq 94.4 & \text{(center:86.0)} \end{cases} \tag{5.2.3}$$

if data sets with at most one reconstructed data point are allowed, then 89 data sets from the original 100 are accepted. The results improved only slightly.

This example shows that data reconciliation is not without any risk. Although the conditions here are somewhat exceptional (good composition data and one poor flow measurement), they are not impossible. The decrease in accuracy of the estimates is small but nevertheless significant.

5.2.1.2.3. Failing Sensor

This example shows the advantage of data reconciliation. In this case a failing sensor is simulated. Occasionally ($\frac{1}{4}$ of the cases, randomly chosen) the bottom composition sensor fails and produces a clearly erroneous value that is nevertheless not too obvious (no saturation). In the rest of the cases it performs somewhat weaker than normal (noise variance = 2.5· the variance in the previous examples). More details on this example can be found in Appendix G (Run # 3). If no data reconciliation is applied, the estimation becomes impossible. The 95% confidence interval of N includes for instance 0 and $+\infty$. The estimation collapses and the on-line optimizer has to be suspended, unless it remains in operation with the old parameter values.

If data reconciliation is applied, the data points with false bottom composition values can easily be recognized and rejected or reconstructed. Accepting up to one reconstruction per data set, and based upon reconciled data, the results are (86 data sets are accepted) :

$$\begin{cases} 1.1141 \leq \alpha \leq 1.1144 & \text{(center:1.1142)} \\ 78.3 \leq N \leq 90.3 & \text{(center:84.5)} \end{cases} \tag{5.2.4}$$

The results are as good as the ones in example 1 (section 5.2.1.2.1.) with a good sensor and data reconciliation. The variable operating cost is the same to within $ 3 per day. The offset from the true optimal variable operating cost is approximately $ 100 per day. Without data reconciliation and gross error identification, it would be impossible to implement on-line optimization in this example.

5.2.2. Heat Exchanger Network

5.2.2.1. Introduction

Data reconciliation and gross error identification of a process network is a

time consuming task. In section 3.3.5.2, it is suggested that a way to reduce the computational effort is by splitting up the complete network into partial networks of which the data are processed separately. The problem with this approach is that the reconciled streams would not match at the seams of the partition. In the case that noise levels are reasonably low, and only gross error identification is used, this problem does of course not exist. In this section it is illustrated that the combination of results of gross error identifications on partial process networks provides results that are inferior to the results obtained by carrying out a gross error identification on the complete process network. However, at the same time the CPU effort decreases significantly with the introduction of partitioning.

5.2.2.2. Description of the Reconciliation Problem

The network that is considered in this case study is composed of two parallel heat exchangers. Figure 5.2.3 gives a symbolic representation of the flow sheet and of the measurements. For the sake of the example the network is provided with flow and temperature measurements on every flow. The equations that are being used are mass and energy balances around every node of the network represented on figure 5.2.3. Energy balance equations are bilinear in flow and temperature. In order to get a linear set of equations the enthalpy flow is calculated as a product of mass flow, temperature and heat capacity of the flow. A disadvantage of using calculated variables is that these variables usually violate the white noise assumptions of most reconciliation equations. A more important disadvantage is that by using calculated measurements, correlations between the measurement errors are introduced which have to be carefully analyzed and introduced in Q, the measurement error covariance matrix. This is shown in the following lines.

Let F_i be the value of mass flow measurement i, and let T_i be the corre-

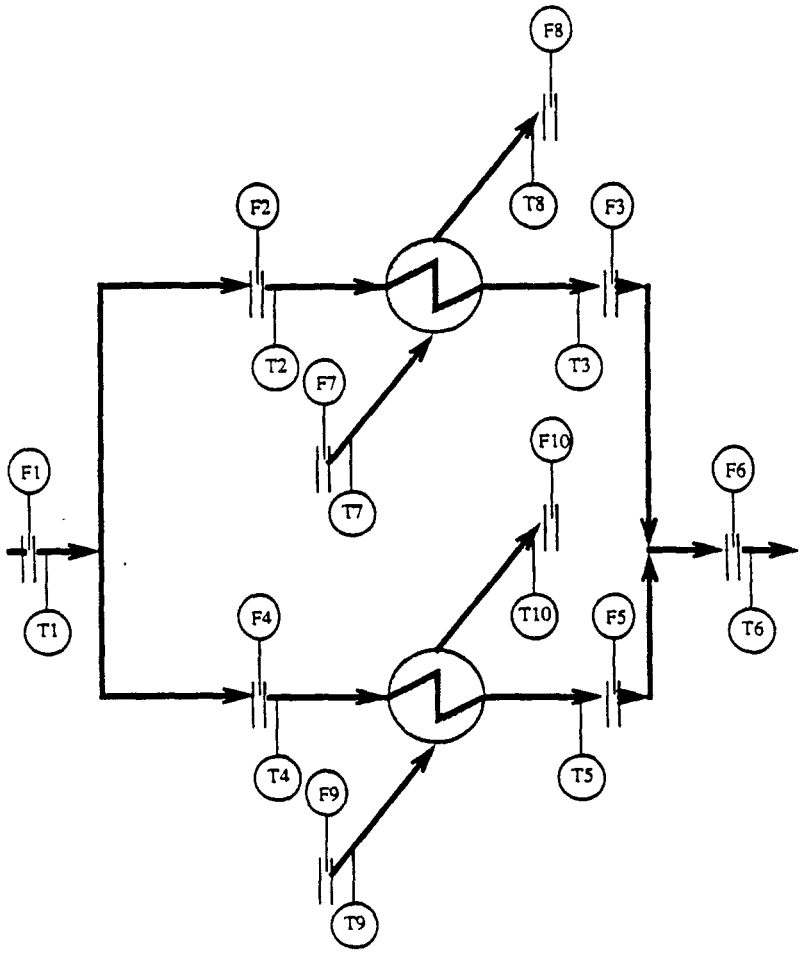

Fig.5.2.3. Instrumentation of the Considered Heat Exchanger Network

sponding temperature measurement. The correct values are given by $F_i^\star$ and $T_i^\star$. It is assumed that the variances of F_i and T_i are known as $\sigma_{F_i}^2$ and $\sigma_{T_i}^2$ respectively. It is accepted that the various measurement errors are uncorrelated. This means in particular that the correlation between the error in F_i and the error in T_i is zero :

$$\sigma_{F_i T_i}^2 = 0. \tag{5.2.5}$$

The calculated enthalpy flow is obtained as follows :

$$E_i = C_{pi} F_i T_i. \tag{5.2.6}$$

This expression can be approximated by a linear expression in the immediate neighborhood of the current true values of F_i and T_i :

$$E_i = C_{pi}F_i^{\star}T_i^{\star} + C_{pi}F_i^{\star}(T_i - T_i^{\star}) + C_{pi}T_i^{\star}(F_i - F_i^{\star}). \tag{5.2.7}$$

This equation will be used to determine the variance of E_i and the covariance of E_i and F_i as a function of $\sigma_{F_i}^2$ and $\sigma_{T_i}^2$:

$$\sigma_{E_i}^2 = E\big((E_i - E_i^{\star})(E_i - E_i^{\star})\big); \tag{5.2.8}$$

in which $E(\cdot)$ represents the expected value. Then :

$$\sigma_{E_i}^2 = E\Big(\big(C_{pi}F_i^{\star}(T_i - T_i^{\star}) + C_{pi}T_i^{\star}(F_i - F_i^{\star})\big)^2\Big) \tag{5.2.9}$$

$$\begin{aligned} = E\Big(\big(C_{pi}F_i^{\star}(T_i - T_i^{\star})\big)^2\Big) + E\Big(\big(C_{pi}T_i^{\star}(F_i - F_i^{\star})\big)^2\Big) \\ + 2 \cdot E\Big(\big(C_{pi}\big)^2\big(T_i^{\star}(F_i - F_i^{\star})\big)\big(F_i^{\star}(T_i - T_i^{\star})\big)\Big). \end{aligned} \tag{5.2.10}$$

Since $\sigma_{F_iT_i}^2 = 0$, the last term will be equal to zero. Therefore :

$$\sigma_{E_i}^2 = C_{pi}^2(F_i^{\star 2}\sigma_{T_i}^2 + T_i^{\star 2}\sigma_{F_i}^2). \tag{5.2.11}$$

In a similar way, the covariance between E_i and F_i can be calculated :

$$\sigma_{E_iF_i}^2 = E\big((F_i - F_i^{\star})(E_i - E_i^{\star})\big) \tag{5.2.12}$$

$$= E\Big(C_{pi}T_i^{\star}(F_i - F_i^{\star})^2\Big) + E\Big(C_{pi}F_i^{\star}(F_i - F_i^{\star})(T_i - T_i^{\star})\Big). \tag{5.2.13}$$

Again the last term is zero because $\sigma_{F_iT_i}^2 = 0$. Therefore,

$$\sigma_{E_iF_i}^2 = C_{pi}T_i^{\star}\sigma_{F_i}^2. \tag{5.2.14}$$

The values of $F_i^{\star}$ and $T_i^{\star}$ are not known. Therefore, the measured values F_i and T_i are used as approximation :

$$\sigma_{E_i}^2 \simeq C_{pi}^2(F_i^2\sigma_{T_i}^2 + T_i^2\sigma_{F_i}^2); \tag{5.2.15}$$

$$\sigma^2_{E_i F_i} \simeq C_{pi} T_i \sigma^2_{F_i}. \tag{5.2.16}$$

Finally,

$$\sigma^2_{E_i F_j} = \sigma^2_{F_i F_j} = \sigma^2_{E_i E_j} = 0, \forall i \neq j. \tag{5.2.17}$$

Equations 5.2.18–27 are used in this data reconciliation example. The symbols correspond to the symbols in figure 5.2.3. The first set of equations are mass balances :

$$F_1 - F_2 - F_4 = 0; \tag{5.2.18}$$

$$F_3 + F_5 = F_6 = 0; \tag{5.2.19}$$

$$F_2 - F_3 = 0; \tag{5.2.20}$$

$$F_4 - F_5 = 0; \tag{5.2.21}$$

$$F_7 - F_8 = 0; \tag{5.2.22}$$

$$F_9 - F_{10} = 0. \tag{5.2.23}$$

The remaining equations are energy balances :

$$E_1 - E_2 - E_4 = 0; \tag{5.2.24}$$

$$E_3 + E_5 - E_6 = 0; \tag{5.2.25}$$

$$E_2 + E_7 - E_3 - E_8 = 0; \tag{5.2.26}$$

$$E_4 + E_9 - E_5 - E_{10} = 0. \tag{5.2.27}$$

All reconciliation equations are used in the complete reconciliation problem. The alternative method will split the system in four separate systems. The first system centers around the splitting node. Only two equations are used : equation 5.2.18 and equation 5.2.24. The second subsystem centers around the node on the exit of the system. Here equations 5.2.19 and 5.2.25 are used. The third and the fourth partial network include one heat exchanger each. Equations

5.2.20, 5.2.22, and 5.2.26 are used for the first exchanger, and equations 5.2.21, 5.2.23, and 5.2.27 are used for the other exchanger.

The algorithm that is used here for the detection of gross errors is proposed by Narasimhan and Mah [1987] and is based on the generalized likelihood ratio test. A detailed description of the algorithm is given in Appendix E. The algorithm by Narasimhan and Mah [1987] because it allows for the detection of gross errors without carrying out the data reconciliation.

5.2.2.3. Results and Discussion

More details on the specifics of these examples can be found in Appendix G (Run # 4).

5.2.2.3.1. Type 1 Hypothesis Test Error

The statistical significance of the test that is applied in this case study is set equal to 99 % . That means that there is a 1 % probability that a gross error will be detected in a system while there is no systematic error. If the complete system is reconciled at once, then there is a total chance of 1% for a type 1 hypothesis test error. If the flow sheet is divided in 4 parts, as suggested in this case, the chance for type 1 hypothesis test error increases to approximately the fourfold. In general, the chance for the incidence of at least one type 1 hypothesis test error for n partial systems tested with significance α equals $1-(1-\alpha)^n$. This points out a first important disadvantage of using a partitioned gross error detection scheme. The probability of detecting a gross error while there is only random noise increases significantly. If the gross error identification of an entire plant would be partitioned into forty subproblems, and they are all checked with a 99 % significance, the chance for at least one type 1 gross error identification mistake is *one out of three* ! This increase in type 1 test error probability is illustrated in this case study. Two hundred cases

without gross error but with random white zero mean noise are presented to the gross error detection algorithm. The white noise has a standard deviation of two precent of the noise free value to which it is added. It should be stressed that the white noise is added to the flow and the temperature measurements. Therefore the calculated enthalpy has colored noise. If the complete system is checked for gross errors, one gross error was detected. If the system is divided in four parts, nine gross errors were detected. These numbers are close to the predicted number of type 1 errors, which are two out of two hundred for the case in which the entire system is checked at once, and eight out of two hundred for the partitioned system.

Often flows are measured as a differential pressure. Then it is more realistic to assume that the noise on the square of the flow measurement is normal. Implementing that noise on the flow measurement did not significantly change the type 1 hypothesis test error. The method that does consider the complete system at once detected no gross errors. The partitioned method indicated 3 gross errors on one hundred data sets. These results are not significantly different from the expected results under the assumption of normal noise.

5.2.2.3.2. Pure Heat Loss in Heat Exchanger 1

In this and the following examples, data sets are generated in a way which is schematically represented in figure 5.2.4. A nominal data set is taken, to which a gross error is added. Then white noise is added, and the measurements are combined to flows and enthalpies before being presented to the gross error identifiers.

One hundred data sets were generated by adding white zero mean noise to temperature and flow measurement with a standard deviation of two percent of the noise free values to which the noise was added. The noise free data set contained a pure heat leak in heat exchanger 1. The gross error detection

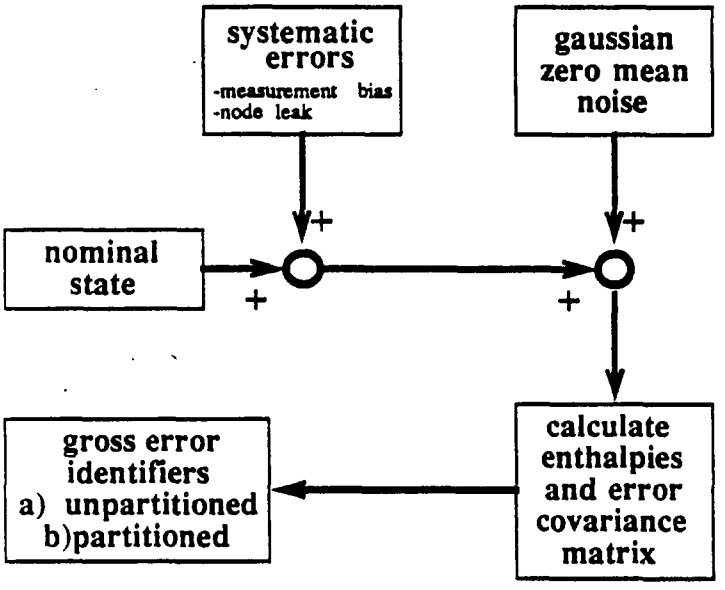

Fig.5.2.4. Schematic Representation of Data Generation

scheme that considers the complete system at once, did detect the type and the size of the gross error correctly in all one hundred cases. If the system is split up into four subsystems, the pure heat leak is detected also in every case, but in five cases one more (liquid) leak before or after the heat exchanger is detected on the crude oil side. This means that the hypothesis of a fluid leak under certain random circumstances becomes relative probable if the network is divided into parts. In case the complete network is considered, the (erroneous) hypothesis of a fluid leak can be checked through the nodes before and after the heat exchanger and then becomes quite improbable, and is rejected. Furthermore, in the partitioned case, two more errors were detected around the splitting and the combining nodes. These are type 1 hypothesis test errors.

5.2.2.3.3. Flow 2 Measurement Bias

Similar to the previous example, sets with a measurement bias in flow 2 were presented to both gross error identifiers. The bias was detected by both identifiers in all cases. The identifier that considers the complete network detected one pure heat leak in the first heat exchanger, which can be considered as a type 1 test error. The partitioned approach had six type 1 test errors such as a pure heat leak in the combining node, etc. One pure heat leak in the splitting node was detected also, which is a type of gross error that can be ruled out much easier if the complete network is considered in one effort.

5.2.2.3.4. Fluid Leak on the Kerosene Side after Heat Exchanger 2

Again, a data base with one hundred data sets is generated by adding white noise to a noise free set with a fluid leak on the kerosene side after heat exchanger 2. In this case the approach that looks at the complete flow sheet does not make any wrong conclusions. The partitioned technique detects the error in every case, but again adds a pure heat leak to the list of errors in 2 cases. This type of error can be ruled out easier using the approach that works with the complete network, by checking along other nodes.

5.2.2.3.5. Unrelated Multiple Errors

Two errors are introduced in a set of one hundred data points. Again white noise is added with a standard deviation of two percent of the true measurement value. The errors are unrelated, i.e. no node residue is influenced by both systematic errors. A bias is introduced in flow 3 and a pure heat loss is introduced in heat exchanger 2. The approach dealing with the complete network, did not have any significant problems detecting both errors effectively. Two type 1 test errors occurred. The partitioned approach detects both errors also, but had more problems with the detection of pure heat leaks in the combining node,

and with type 1 test errors.

5.2.2.3.6. Related Multiple Errors

In this case, biases are introduced in flows 2 and 3. The unpartitioned approach detects both errors correctly. Note that in the development of the algorithm it is assumed that the gross errors are unrelated. Therefore, success with related errors is not guaranteed. In this example, no type 1 hypothesis test errors occur with the the unpartitioned approach. The partitioned approach is virtually unable to deal with this situation, and in 26 out of 100 cases wrong errors are detected. The part of the partitioned network that considers heat exchanger 1 has no effective means to distinguish between the possibility of two biases and many other combinations of errors. Then, it becomes very difficult to interpret the results of the partitioned gross error identification, since the partitions are not in agreement. It is not clear how to resolve this problem.

5.2.2.3.7. Exception

An interesting case is the following. If two equally large biases are introduced in flow measurements 1 and 6, the partitioned approach is quite effective in detecting both errors. However, in this case the approach that considers the entire network completely fails. The influence of both errors is equally large and is symmetrical. This confuses the gross error identification routine and up to ten gross errors are erroneously detected for certain runs. Of course, the chances of an equal bias occurring in flows 1 and 6 at the same time are negligible. But it has to be noted that under certain circumstances the gross error identification can fail completely in the detection of multiple gross errors.

5.2.2.3.8. CPU Time Consumption

The partitioned approach consumes much less CPU time than the approach that splits up the considered network. Table 5.2.1 shows CPU time consumption

	No Systematic Errors	One Systematic Error
No Partition	2.0s	3.5s
Partition	0.6s	0.8s

Table 5.2.1.: CPU Time Consumption for Gross Error Detection

for every single gross error detection run for both approaches.

5.2.3. Conclusions On Data Reconciliation

The distillation case study examples have shown that data reconciliation can be used effectively against systematic errors in data by checking consistency with the reconciliation equations. However improvement in the result is not guaranteed. More specifically, if the data are accurate, except for a reasonable level of noise, it seems that data reconciliation can offer some improvement in this case study. This improvement comes at a CPU cost that could be extremely high for larger systems. Under special conditions it is possible that data reconciliation corrupts the estimation results by spreading errors over the different measurements. However, if an error as drastic as a broken sensor occurs (gross error), then data reconciliation and especially gross error detection proves to be a powerful data analysis tool. Reconstructed data can be applied or rejected. Using reconstructed data means full use of data redundancy, but it is difficult and the reliability of the reconstructed data may be unknown. If reconstructed data are not used, then gross error identification can be considered

as an error detection system with some plant knowledge.

The heat exchanger case study examples address the question of the high CPU time demand of data reconciliation and gross error identification. An alternative to processing data from a complete system, would split up the system in n subsystem, and process them separately. This approach is not very useful for data reconciliation since the streams at the seams will not match. However, if the measurement noise levels are low, the data reconciliation can be omitted and only gross error identification can be applied. One example clearly shows that large difference in CPU effort between both approaches. As the number of gross errors increases, the difference in CPU effort increases also. This difference in CPU time may make a partitioned approach the only feasible alternative for a large system. However, the partitioned approach suffers from several drawbacks that are illustrated in the examples. The partitioned approach is characterized by an increased incidence of statistical test errors, because more tests are performed. If the data are not partitioned, a better assessment of the probability of a candidate gross error is obtained, and candidate gross errors are ranked in probability before tests are performed. Second, the partitioned approach gives a much lower resolution of gross errors, i.e. less gross errors can be modeled and distinguished. The partitioned approach is overall less accurate since, near the partition boundaries, it lacks the depth the overall approach has. This point is illustrated in e.g. example 5.2.2.3.2. If the data of a complete system is processed, then multiple errors are distinguished and identified more effectively than if the data is partitioned. This performance of the unpartitioned approached seems to be even more superior if the influence of multiple errors interacts. Only in exceptional circumstances will the performance of the partitioned gross error identification exceed the unpartitioned approach. In conclusions, it can be said that partitioning of the system should

be avoided were possible, since it deteriorates the performance of the error identification significantly. However, if the high CPU demand makes an unpartitioned approach infeasible, then the system can be partitioned in a number of subsystems until the CPU demand becomes reasonable. This will be traded off with a loss in performance.

5.3. MODEL FITTING

This section illustrates problems with model fitting in an on-line optimization environment. Results obtained for the distillation case study are listed first, followed by boiler case study results. The systems that are studied in the case studies as well as the models that are used to simulate the plant behavior and the models that are used internally by the on-line optimizer, are described in detail in Chapter 4.

5.3.1. Distillation

5.3.1.1. Introduction

The results obtained from the distillation case study serve the following purposes. First they demonstrate that even for as simple a model as Eduljee's distillation tower model, parameter estimation can be difficult. It is pointed out that the choice of estimation scheme is not always obvious. It is also shown that poor parameter estimation can result in loss of profit. In the last part of this section on results related to model fitting in the distillation case study, it is demonstrated that the assumptions for calculating approximate confidence intervals for the nonlinear parameter estimation are satisfactorily met. This result is important since these confidence intervals are used in section 5.4 as input to a short cut study of the economic feasibility of the application of a real time reflux ratio optimizer to the C_3 splitter that is considered here.

5.3.1.2. Correlation between Parameter Estimates

Before discussing estimation techniques for the parameters in the model that makes use of the Eduljee and Fenske equations, it is important to investigate which data are needed and also whether enough information is available

to successfully obtain stable estimates. Fenske's and Eduljee's equation use two parameters that are not measured directly and have to be updated periodically. These parameters are the relative volatility (α), and the number of ideal trays (N). The publication of Martin *et al.* [1981] lists for the number of ideal trays :

$$N = \text{tray efficiency} \cdot \text{number of trays} = 0.75 \cdot 125. \simeq 94. \tag{5.3.1}$$

Edgar and Himmelblau [1988] give a value of 1.105 for the relative volatility. This value is obtained by fitting Eduljee's model to the operating conditions given by Martin *et al.* [1981]. These values are used as a reference in the nominal case. However, due to the model plant mismatch these parameters have to be updated in order to make the model represent the plant as closely as possible.

It is impossible to estimate α *and* N from one operating point. At least two different operating points are needed. Even then one can expect a strong correlation between the parameters. It can be shown (see Appendix B) that the following approximation holds :

$$\frac{1}{N} \simeq \frac{1. - \frac{3}{4}c_i}{b_i} \ln \alpha. \tag{5.3.2}$$

In equation (5.3.2), i is an index that indicates the data record that is considered, and :

$$c_i = \left(1 - \left(1 - \frac{1 + \frac{a_i}{\alpha - 1}}{1 + R_i}\right)^{0.5668}\right), \tag{5.3.3}$$

with :

$$a_i = \frac{x_{D,i}}{x_{F,i}} - \frac{1 - x_{D,i}}{1 - x_{F,i}}, \tag{5.3.4}$$

and :

$$b_i = \ln\left(\frac{x_{D,i}}{1 - x_{D,i}} \cdot \frac{1 - x_{B,i}}{x_{B,i}}\right). \tag{5.3.5}$$

It is shown in Appendix B that for databases that contain consistent data (data from the same distillation column, and a column that meets the requirements

to be modeled by the Eduljee and Fenske equations) these c_i's will be relatively constant. The b_i term can be expected to vary only over a narrow range with i. In plant data, the top purity $x_{D,i}$ can expected to be close to 0.95 since the top purity requirement constraint will be active in all cases we consider . $x_{B,i}$ is not expected to vary over a very wide range either. This means that significant correlation can be expected between the parameter estimates for N and for α. The importance of this correlation can be reduced by the fact that the correlation between both parameter estimates does not always result in a significant error in the optimization process. The correlation is due to the fact that for given combination of parameter estimates, a similar fit of the model to the data is obtained. Therefore, the model can still perform properly during the optimization phase. However, a problem is that it is preferable to have uncorrelated errors in parameter estimates. If errors in parameter estimates are correlated significantly it will be necessary to carry the covariances over to the accuracy calculations that combine statistical results with the sensitivity analysis (see sections 3.6 and 5.4). In that case parameter estimate *co*variances have to be calculated *on-line*. In the case that the parameters are not significantly correlated, then only the variances have to be calculated on-line. The calculation as well as the processing of the covariance matrix soon becomes infeasible to do in real time as model size increases.

The conclusion of this section is that it may be doubtful that two parameters can always be estimated successfully in the Eduljee Fenske distillation tower model. Certain databases may not provide enough information to obtain two parameter estimates. This conclusions depends upon the range over which values of c_i and b_i are spread.

Similar situations in which parameters with an apparently very different physical meaning are closely related are common. In the case study this problem

is easily recognizable. Because of the reduced size of the problem and the simplicity of the model, a few simple algebraic manipulations can characterize the relation. Therefore it is always useful to calculate not only the variances of the parameter model estimates, but also the covariances. Large covariance terms indicate correlations between parameter estimates. As a rule of thumb, covariances that are of the order of magnitude or larger than the product of the corresponding individual parameter variances, are of importance. This check should be carried out during the model building phase. Models should be rebuilt to avoid parameter estimate correlation, or, if this is impossible or undesirable, the most important correlations should be identified so that only these have to be calculated on-line. Also in the boiler case study, the model parameters are correlated. That point is considered again in section 5.3.2.

5.3.1.3. The Cost of Poor Estimation

Four different estimation schemes are proposed here. These schemes will be compared against each other as estimation schemes for on-line optimization. In the *(1.)* first scheme only parameter α is estimated, while N remains fixed. The *(2.)* second scheme estimates N while keeping α constant. The *(3.)* third scheme estimates α and N, while the *(4.)* last scheme estimates $\ln \alpha$ and $\frac{1}{N}$. The four schemes are discussed separately. Two examples will compare the performance of the schemes.

5.3.1.3.1. Four Estimation Schemes

Instead of estimating both parameters, α *and* N, the estimation could be limited to one parameter. This seems a quite realistic alternative, because (*1.*) the parameters are strongly correlated and a certain lack of dimensionality seems to exist in the model as explained in the previous section, and (*2.*) it is possible that operating conditions are such that one of the parameters is only

varying over a narrow range. It is, for instance, possible that load variations are expected in the column. A fixed value for the relative volatility may then be adequate while the number of ideal trays would be adjusted for changes in the tray efficiency. And correspondingly, a rather constant feed flow with a varying composition could make the opposite arrangement (constant number of ideal trays, varying relative volatility parameter) attractive. Keeping one parameter fixed has the trivial advantage that the estimation is simple. The estimation criterion for both one parameter techniques is the minimization of the sum of squares of the differences between the calculated and the measured bottom purity.

The third technique estimates both the relative volatility (α) and the number of ideal trays (N). Here also, the estimation criterion is the minimization of the sum of squares of the differences between the calculated and the measured bottom purity. The advantage of estimating both parameters instead of only one is that a better model fit becomes possible. However this technique suffers from the correlation between the parameter estimates. That correlation was the subject of the previous section. It is mentioned earlier that it is impossible to obtain estimates of the relative volatility as well as the number of ideal trays from one operating point. For instance, for the nominal operating conditions, any point on the curve given by :

$$\frac{6.016}{N} \simeq \ln\alpha\Big(1 - \frac{3}{4}\big(1 - (1 - [\frac{1 + \frac{1.1905}{\alpha - 1}}{18.2}])^{0.5668}\big)\Big) \qquad (5.3.6)$$

would fit the Eduljee and Fenske model to the column data. This relation is derived in Appendix B, and is represented in figure 5.3.1 for nominal operating conditions (full line). The curve can be obtained from the nominal operating point data and the model equations only. The relation is very nonlinear.

As soon as the data points from two different operating conditions are available, estimates for both parameters can be obtained. A parameter esti-

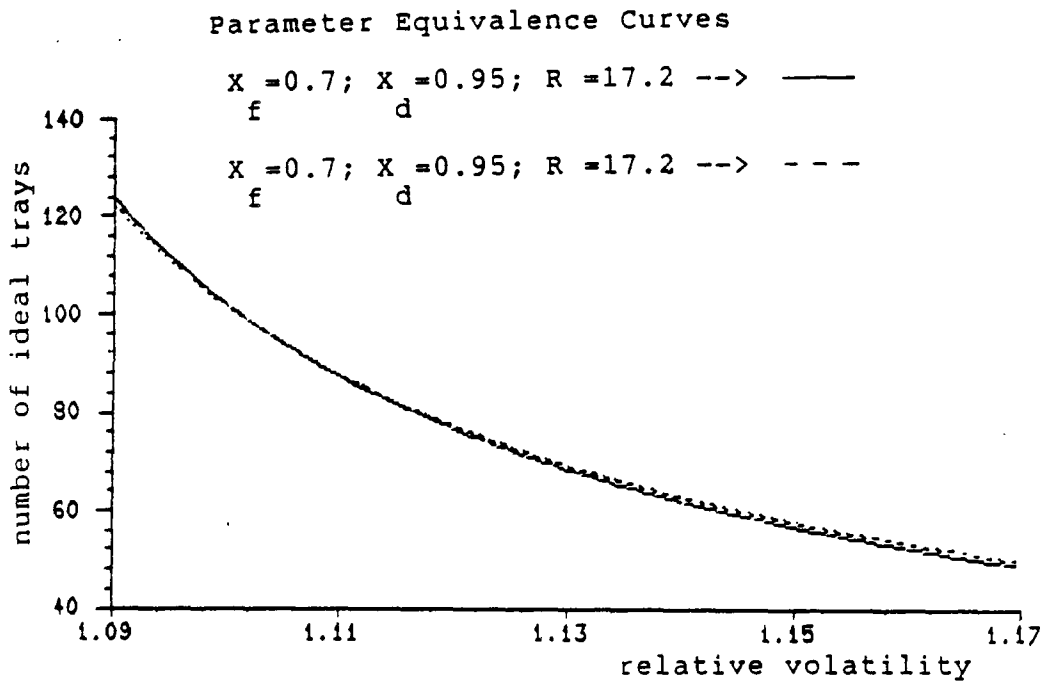

Fig.5.3.1. Parameter Equivalence Curve for the Nominal Case (Full Line) and for Increased Reflux (Dotted Line)

mation could then be conceived as the determination of the intersection of two curves like the ones in figure 5.3.1. However, the two curves intersect at a very small angle, since these "parameter equivalence curves" for similar operating conditions are very close. Small inaccuracies in the data can make the intersection point move up or down the curves considerably. Parameter estimation can be based upon more than two data points. Then the parameter estimation is equivalent to a determination of the intersection of all equivalence curves. Such intersection only exists if all data are correct (i.e. without any noise) and if the model would be correct. Therefore in practice, all curves intersect at various points, and the parameter estimation comes down to a determination of the "most probable" intersection if it would be a single point. "Most probable" in the sense of : resulting in the most probable set of model to data mismatches in the bottom propylene fraction. Again it is clear that inaccuracies in the data

can perturb the position of this most probable point in the neighborhood of the curves on figure 5.3.1. That is the reason for the strong correlation between the values of the estimates of both parameters. The ill-conditioned estimation criterion combined with the nonlinearity of the model can cause incomplete convergence of the estimation scheme.

A three dimensional impression of the estimation criterion can be visualized as a narrow cut along the curves in figure 5.3.1. The non-linearity can deter the performance of the least squares routine. However, it seems that the nonlinearity of the problem can be restrained significantly by a transformation of the problem space. The estimation criterion can be minimized with respect to $\ln \alpha$ and $\frac{1}{N}$. A justification for this transformation can be found in Appendix C. The parameter equivalence curves that are presented in figure 5.3.1. are almost linear in the $\ln \alpha, \frac{1}{N}$ space. This is illustrated in figure 5.3.2. Linearization will make the estimation module more able to deal with the ill-conditioning of the estimation criterion.

Four estimation techniques have been described. The first two techniques estimate α and N respectively, with the sum of squares of the differences between the estimated and the measured bottom propylene mole fraction as the estimation criterion. The third technique estimates both α *and* N with the same criterion. This technique will be referred to as estimating in the α,N plane or space. The fourth technique estimates $\ln \alpha$ *and* $\frac{1}{N}$. As a shorthand, this last technique will be referred to as estimating in the $\ln \alpha$, $\frac{1}{N}$ plane or space.

Two simple examples show that the last approach can give better results than the third (estimating α and N), and therefore is a potentially superior estimation technique. In the first example, the estimation techniques are used to fit Eduljee's model to data obtained using Eduljee's equations. In the second example, Eduljee's model is fit to data obtained from Underwoods model.

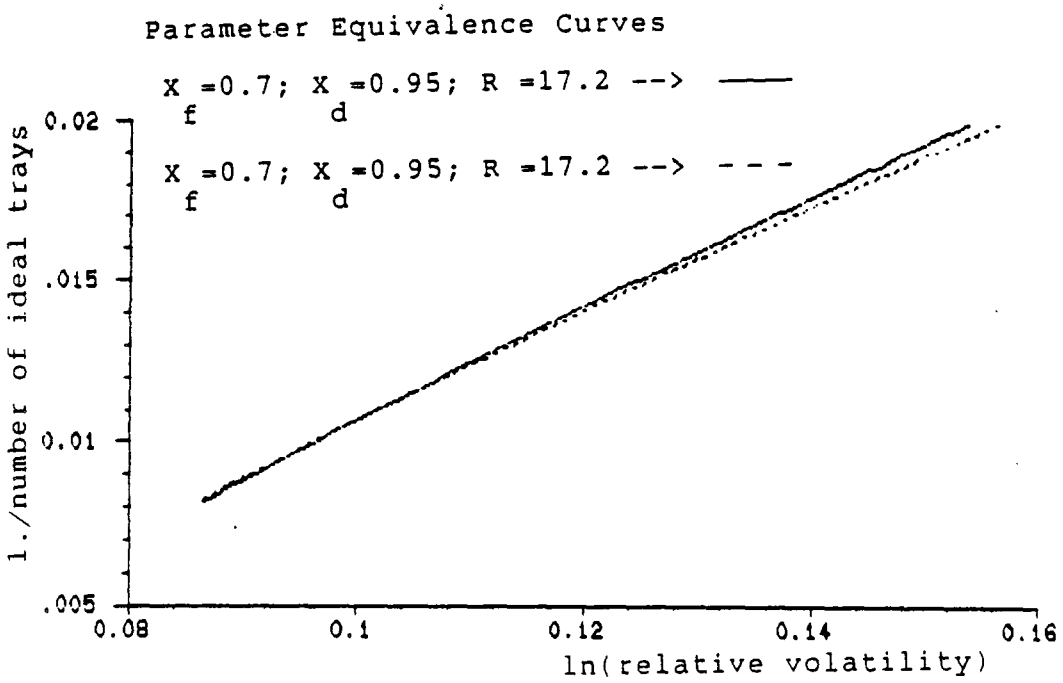

Fig.5.3.2. Parameter Equivalence Curve for the Nominal Case (Full Line) and for Increased Reflux (Dotted Line).

After these two examples that compare the fitting performance of the separate estimation techniques, a third example illustrates an undesirable transient phenomenon that can occur in an on- line optimizer. Finally, some conclusions are given to summarize the results of this section on model fitting that were obtained from the distillation case study.

5.3.1.3.2. Example 1 : Eduljee's Model as "Plant"

In the first example, a data set is used that contains data with inputs randomly spread around the nominal inputs. The Eduljee model with nominal parameters ($\alpha = 1.105$, $N = 94.$) is used to generate outputs corresponding to the these inputs. Then noise (zero mean, gaussian) is added to inputs and outputs. This data base is described in more detail in section 5.4.1.2. Sixteen samples are used, randomly chosen from the extensive data set. The standard deviation of x_F, x_B and x_D is $1.6 \cdot 10^{-3}$. Only these mole fractions and the

reflux ratio ($\sigma_R^2 = 7.1$) are needed to obtain the parameter estimates. More details on this example can be found in Appendix G (Run # 5). Using a Marquardt-Levenspiel (CMLIB) algorithm to minimize the objective function, the parameter estimation in the α, N space obtained as results are :

$$\begin{cases} \hat{\alpha} = 1.1716 \pm 0.0046 & (95\% \text{ confidence}); \\ \hat{N} = 47.59 \pm 1.41 & (95\% \text{ confidence}). \end{cases} \tag{5.3.7}$$

These results are close to the curves in figure 5.3.1. or figure 5.3.2 the method working in the $\ln\alpha$, N^{-1} space resulted in :

$$\begin{cases} \ln\hat{\alpha} = 0.099944 \pm 0.00004 & (95\% \text{ confidence}); \\ \frac{1}{\hat{N}} = 0.010658 \pm 0.00005 & (95\% \text{ confidence}). \end{cases} \tag{5.3.8}$$

and this results in :

$$\begin{cases} \hat{\alpha} = 1.1051; \\ \hat{N} = 93.8. \end{cases} \tag{5.3.9}$$

Due to the combination of ill-conditioning and nonlinearity, the method in the α, N plane stalls before converging to the correct combination of parameters. Using the unconverged combination 5.3.7 instead of the estimates given in 5.3.9 would cause a loss in performance of over $ 1700 per day. It should also be noted that the method in ($\ln\alpha$, N^{-1}) space is much less sensitive to the initial guess than the α, N alternative. Furthermore, the ($\ln\alpha$, N^{-1}) method seems to be more robust. Starting from some very poor guess such as $\alpha = 1.24$ and $N = 200$ the (α,N) method would fail, whereas the ($\ln\alpha$, N^{-1}) method still converges to the same result listed above. It should be pointed out that in order to increase the robustness of the method considerably, a constraint is added to the criterion under the form of a penalty function. The model, and more particularly the Eduljee correlation, contains a term that is raised to the power 0.5668. During the search for the minimum of the estimation criterion, it could occasionally happen that this term became negative, and numerical problems are encountered. In order to prevent this problem, a constraint is

added to the estimation problem formulation. The constraint is an inequality constraint, and specifies that the parameters should be such that the term that is raised to the power 0.5668 should never become smaller than zero. This constraint is never found to be active at the final result, and is introduced for all estimation techniques. The parameter estimation is of course started in the feasible section.

5.3.1.3.3. Example 2 : Underwood's Model as "Plant"

In a second example, data generated using Underwood s equations are used as a database. The constants used in the Underwood equations are given in Chapter 4. In this case data are generated systematically from a grid. The grid is developed in the (F,x_D,R) space. A first group of thirty results is generated using ten values of x_F between 0.65 and 0.75 for $F = 1.2 \cdot 10^6 \frac{lb}{day}$ and $R = 17.5$, for $F = 1.0 \cdot 10^6 \frac{lb}{day}$ and $R = 15$, for $F = 1.4 \cdot 10^6 \frac{lb}{day}$ and $R = 20$. A second group of thirty results is generated using ten values of F between $10^6 \frac{lb}{day}$ and $1.4 \cdot 10^6 \frac{lb}{day}$ for $x_F = 0.7$ and $R = 17.5$, for $x_F = 0.65$ and $R = 20$ and for $x_F = 0.75$ and $R = 15$. A last set of thirty results is generated using ten values of R between 15 and 20, for $F = 1.2 \cdot 10^6 \frac{lb}{day}$ and $x_F = 0.7$,for $F = 1.4 \cdot 10^6 \frac{lb}{day}$ and $x_F = 0.75$,and for $F = 1.0 \cdot 10^6 \frac{lb}{day}$ and $x_F = 0.65$. More details on the specifics of this example can be found Appendix G (Run # 6). The results are shown in figures 5.3.3–6 and in table 5.3.2. Table 5.3.2 shows the estimation results for all four techniques. In the first result only α is estimated, and the number of ideal trays is kept constant at 94, the number indicated by Martin *et al.* [1981]. The regression coefficient R^2 is defined (in all four cases) as :

$$R^2 = \rho^2_{x_B \hat{x}_B} \tag{5.3.10}$$

$$= \frac{\sigma^2_{x_B \hat{x}_B}}{\sigma^2_{x_B} \sigma^2_{\hat{x}_B}} \tag{5.3.11}$$

$$= \frac{\left(\sum_{i=1}^{I}\left((x_{Bi} - \overline{x_{Bi}}) \cdot (\hat{x}_{Bi} - \overline{\hat{x}_{Bi}})\right)\right)^2}{\sum_{i=1}^{I}(x_{Bi} - \hat{x}_{Bi})^2 \cdot \sum_{i=1}^{I}(x_{Bi} - \hat{x}_{Bi})^2} \tag{5.3.12}$$

$$= \frac{\left(\sum_{i=1}^{I} x_{Bi}\hat{x}_{Bi} - I \cdot \overline{x_{Bi}\hat{x}_{Bi}}\right)^2}{\sum_{i=1}^{I}(x_{Bi} - \hat{x}_{Bi})^2 \cdot \sum_{i=1}^{I}(x_{Bi} - \hat{x}_{Bi})^2}. \tag{5.3.13}$$

with I the number of data sets that are available. R^2 can be used as a measure of the fit of a model with certain given parameter values to a data set. A larger R^2 corresponds to a better fit.

# exp.	F	x_F	R
10 exp.	$1.4 \cdot 10^6 \frac{lb}{day}$	0.65 – 0.75	20.0
10 exp.	$1.2 \cdot 10^6 \frac{lb}{day}$	0.65 – 0.75	17.5
10 exp.	$1.0 \cdot 10^6 \frac{lb}{day}$	0.65 – 0.75	15.0
10 exp.	$1.0 - 1.4 \cdot 10^6 \frac{lb}{day}$	0.65	20.0
10 exp.	$1.0 - 1.4 \cdot 10^6 \frac{lb}{day}$	0.70	17.5
10 exp.	$1.0 - 1.4 \cdot 10^6 \frac{lb}{day}$	0.75	15.0
10 exp.	$1.4 \cdot 10^6 \frac{lb}{day}$	0.75	15.0 – 20.0
10 exp.	$1.2 \cdot 10^6 \frac{lb}{day}$	0.70	15.0 – 20.0
10 exp.	$1.0 \cdot 10^6 \frac{lb}{day}$	0.65	15.0 – 20.0

Table 5.3.1.: Input Grid

The second set of results concerns an estimation of only the number of ideal trays, keeping the value of the relative volatility constant to 1.105. This value is listed by Edgar and Himmelblau [1988] as the value for which Eduljee's model fits the data in Martin *et al.*[1981]. A third result is achieved by estimating parameters in the α, N space. The first three results are rather close, and

Technique	Solution (95% confid.int.)	R^2
α only est'd.	$\hat{\alpha} = 1.1037(.0003), N_{fix} = 94.$	0.9982
N only est'd.	$\hat{N} = 93.5(.3), \alpha_{fix} = 1.105$	0.9976
$\alpha\&N$ est'd.	$\hat{\alpha} = 1.1042(.0003), \hat{N} = 95.03(.005)$	0.9980
$\ln\alpha\&\frac{1}{N}$	$\hat{\alpha} = 1.1119(3 \cdot 10^{-6}), \hat{N} = 83.7(.1)$	0.9991

Table 5.3.2.: Estimation Results of Fitting Eduljee's Model on Simulation Results Obtained with Underwood's Model

a comparable quality of the fit. A fourth result is obtained in the $\ln\alpha$, N^{-1} space. The fit is better (higher R^2), but not very much. The values of α and N (the confidence interval width is approximate) are not close to the first three results that are more or less grouped together. Apparently the data are better represented in that way. The fact that the third method converges close to where the two one-dimensional methods converge, indicates that the third estimation technique does not continue its search as soon as it hits the "narrow valley" in which the best estimates can be found. In this example, as opposed to the previous example, the lack of convergence of the third method does not result in an important deterioration of the performance, as will be seen below.

The following figures show the better fit of the result of the estimation in the $\ln\alpha$, N^{-1} space in a graphical way. In the first figure (5.3.3) the bottom composition as a function of the feed composition is given for a mass feed rate of $1.2 \cdot 10^6 \frac{lb}{day}$ (the nominal value) and a reflux of 16.9. This reflux ratio is optimal for the Underwood model with top composition 95% propylene and feed purity 70% propylene. In this figure as well as in the four following figures, the

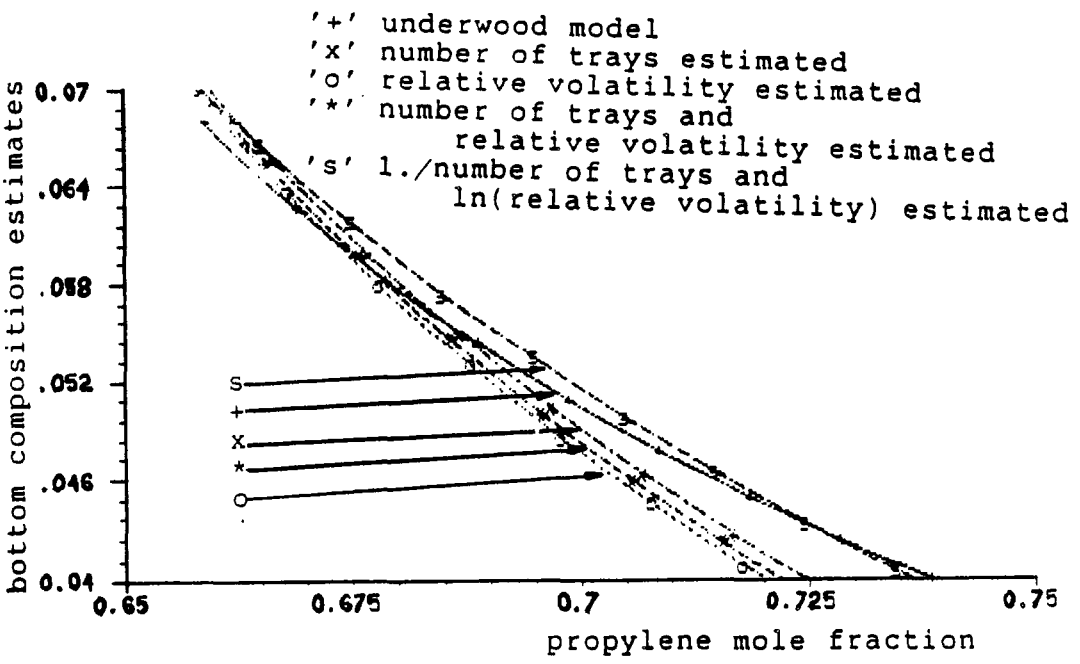

Fig.5.3.3. Bottom Composition Estimates as a Function of Propylene mole fraction in the Feed Flow. Comparison of Results obtained from Eduljee's Model using Parameters Estimates from Different Estimation Schemes and Results obtained from Eduljee's Equations.

curve indicated '+' refers to the Underwood model, or the "data". The curve indicated 'o' refers to the estimation method that only searches the relative volatility and uses a constant number of ideal trays. The curve with the '×' sign is a representation of estimates obtained using results from the estimation in which the number of ideal trays is a variable while the relative volatility remained constant. This is the second technique in table 5.3.2. The curve with the '*' corresponds to the third technique in table 5.3.2., estimating α and N in the α, N space. Finally, the curve marked 's' corresponds to the last technique in table 5.3.2. in which the estimation is performed in the $\ln \alpha$, N^{-1} space. A mismatch between the Underwood results and the fitted results is apparent. It is also clear that the first three techniques (in table 5.3.2.) are very similar. The curves corresponding to these three first estimation techniques are

grouped together, while curve 's' for the fourth estimation technique is definitely different. Figure 5.3.3 represents the bottom fraction of propylene as a function of the feed composition for a feed mass flow rate of $1.2 \cdot 10^6 \frac{lb}{day}$ and a reflux ratio of 16.9. Figure 5.3.4 represents the bottom fraction of propylene as a function of the reflux ratio for a feed mass flow rate that is $1.2 \cdot 10^6 \frac{lb}{day}$ and a feed purity of 70%. In both figures, it seems that technique 's' (estimating in the $\ln \alpha, \frac{1}{N}$ space) produces better fits.

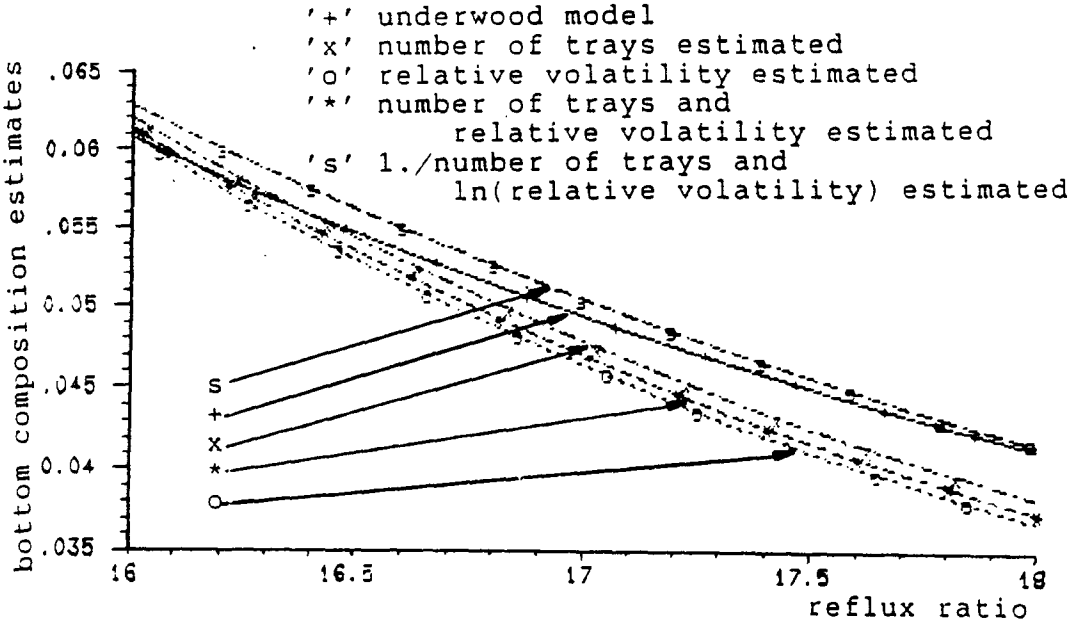

Fig.5.3.4. Bottom Composition Estimates as a Function of Reflux Ratio. Comparison of Results obtained from Eduljee's Model using Parameters Estimates from Different Estimation Schemes and Results obtained from Eduljee's Equations.

Studying the results of the different estimation techniques and their comparison with the Underwood results in the bottom purity values is important since the bottom purity values are the basis of their fit. The bottom purity values are the responses of the models. However, from an on-line optimization point of view, it is also necessary to look at the objective function, the operat-

ing cost. Figure 5.3.5 presents the operating cost as a function of the propylene fraction in the feed.

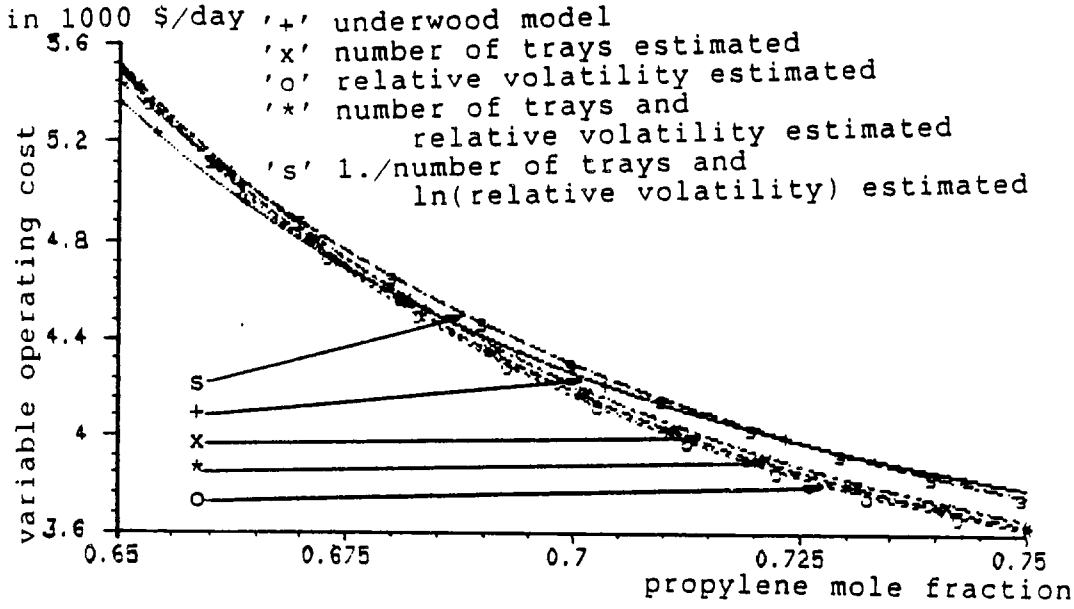

Fig.5.3.5. Variable Operating Cost Estimates as a Function of Propylene mole fraction in the Feed Flow. Comparison of Results obtained from Eduljee's Model using Parameters Estimates from Different Estimation Schemes and Results obtained from Eduljee's Equations.

Although the '*s*' curve does not look much better than the three other estimation curves, it must be pointed out that it seems to have a derivative that is much more comparable with the Underwood curve. More important is the comparison of the operating costs as a function of the reflux ratio, since that is the optimization variable. This comparison is shown in figure 5.3.6.

The fourth technique, estimating in the $\ln\alpha$, N^{-1} space shows a significant overestimation of the operating costs for lower reflux ratios. But still the result labeled '*s*' seems better than the other three techniques. This difference is shown clearly in figure 5.3.7, where the derivative of the operating costs with respect to the reflux ratio is shown as a function of the reflux ratio.

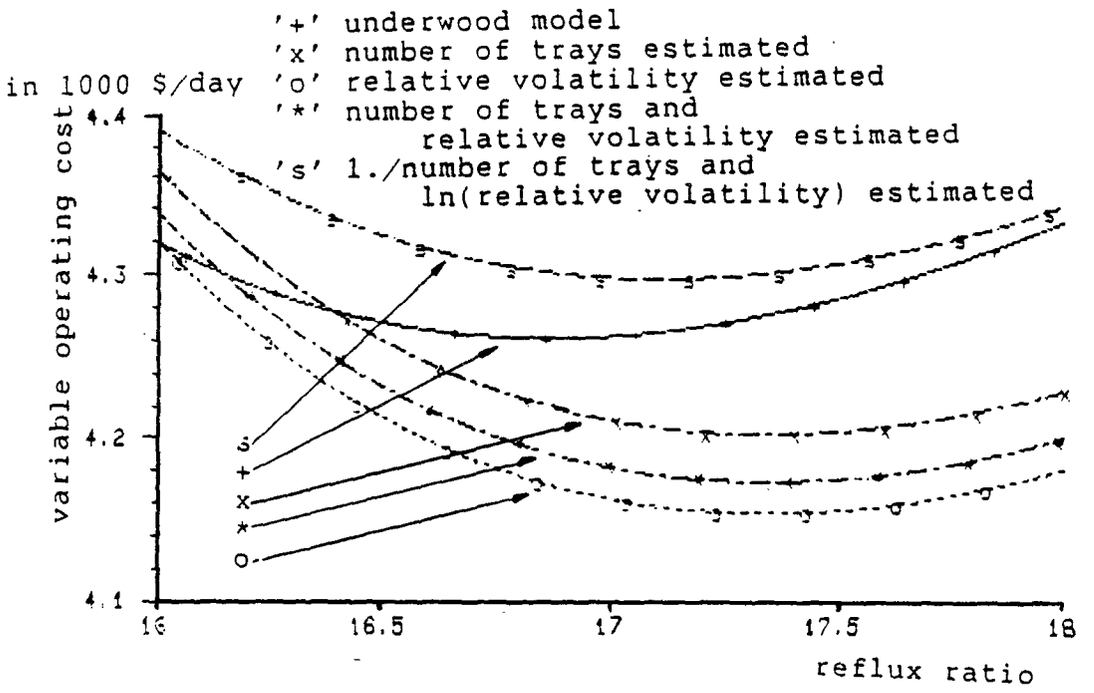

Fig.5.3.6. Variable Operating Cost Estimates as a Function of Reflux Ratio. Comparison of Results obtained from Eduljee's Model using Parameters Estimates from Different Estimation Schemes and Results obtained from Eduljee's Equations.

The reflux ratio for which the derivative would be zero (the optimal reflux ratio) is estimated much better by the fourth technique then by any of the others. The error made by the three first techniques (the curves almost fall together) is approximately twice the error made by the fourth technique.

The model plant mismatch can cause an error in the estimation of the optimal reflux ratio, and in general it will. In figure 5.3.7 it can be seen that the errors are small in this example. The methods that are labeled 'o', '$\star$' and '$\times$' predict an optimal reflux ratio around 17.3. The true optimum reflux ratio is 16.85 as obtained from the Underwood model. The error in the reflux ratio would result in a loss of about $ 30 per hour, which can be neglected. The error of the method labeled 's' is about a quarter of that of the other methods. Therefore, using this last method instead of one of the other estimation

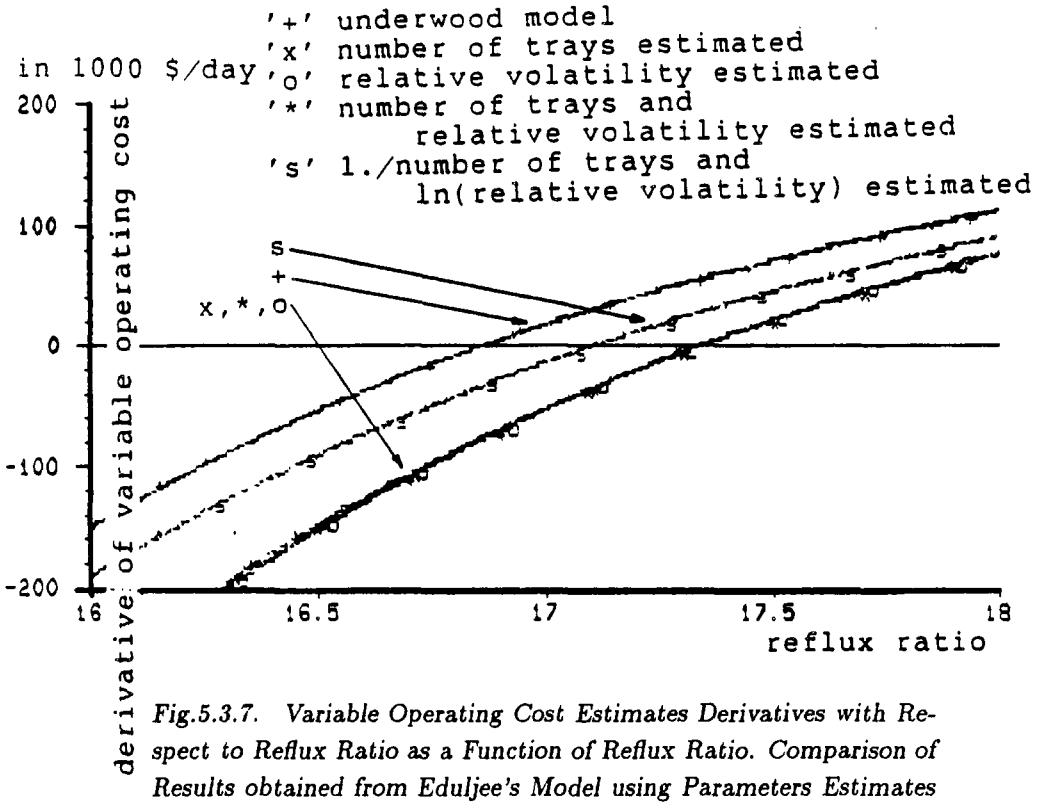

Fig.5.3.7. Variable Operating Cost Estimates Derivatives with Respect to Reflux Ratio as a Function of Reflux Ratio. Comparison of Results obtained from Eduljee's Model using Parameters Estimates from Different Estimation Schemes and Results obtained from Eduljee's Equations.

techniques would only result in about $ 20 per hour loss, which again can be neglected. Mismatch can also result in errors in the calculated sensitivities of the objective function (operating cost) and the optimization variables (reflux ratio and top purity). This is the reason why much effort has to be spent in model identification in almost every on-line optimization application (i.e. in this case study).

5.3.1.3.4. Transient Behavior

The next example illustrates an important transient characteristic of estimation with respect to on-line optimization. It is clear that a parameter estimation technique should be able to achieve a good fit of a short-cut model to a set of steady state data covering a relatively small range. But on the other

hand, in an on-line optimization application, the parameter estimation should also be able to achieve reasonable fits for a wide range of data. In the following example, the relative volatility and the number of ideal trays are made a strong function of the feed composition. This can be considered as a change in the model that originally is presented with constant α and N. Eduljee's model with the variable relative volatility and number of trays is used as "plant". Eduljee's model is selected here to serve as the plant as well as the model in the optimization module, because in that way structural plant model mismatch is excluded as cause of the transient that is discussed below. Eduljee's model with constant relative volatility and number of trays is fitted to "plant" data. The plant model/mismatch is due to the fact that the optimizer works with a constant α and N, while in the data α and N is not constant. While the feed propylene fraction varied from 0.65 to 0.75, the relative volatility went from 1.105 to 1.110 and the number of ideal trays went from 94 to 99. The data base reflects a large shift in steady state operating conditions. This shift will induce a change in the estimated model parameters. Moderate noise is added to drive the estimation. The data base length is thirty. More details on this example can be found in Appendix G (Run # 7). Results are given in figure 5.3.8, 5.3.9 and 5.3.10 where every unit in the x-axis refers to an update of the database. The oldest data set is replaced by a new one. In all three figures the dash-dot line represents results obtained with the estimation technique operating in the $\ln \alpha$, $\frac{1}{N}$ space, and the dotted line corresponds to results from the estimation technique estimating in the α,N space. Figure 5.3.8 gives estimates of the relative volatility. The estimated number of ideal trays are given in figure 5.3.9. Optimal reflux ratios corresponding to the parameter estimates and the nominal operating conditions (presented by Martin *et al.* [1981]) are given in figure 5.3.10. The optimization results are not implemented, therefore this is

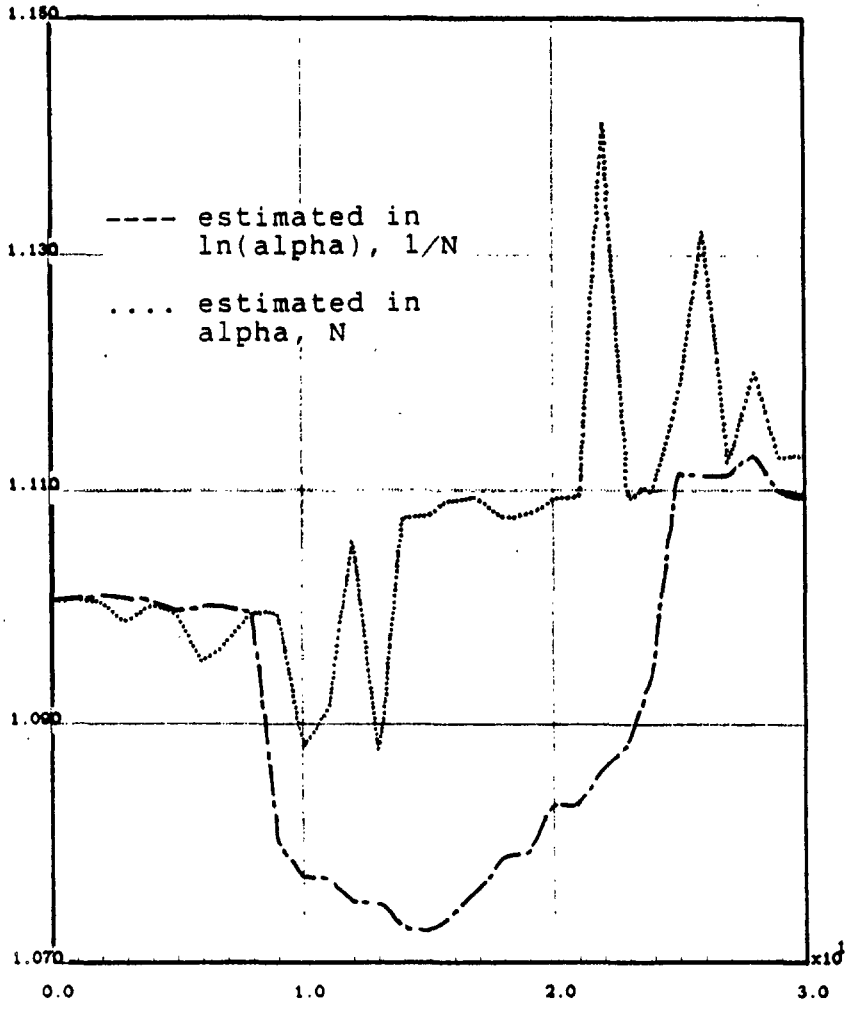

Fig.5.3.8. $\hat{\alpha}$ while Updating of the Data Base

not a closed loop simulation.

Results obtained by the α,N technique (dotted line) indicate some estimation problems, but the $\ln \alpha$, $\frac{1}{N}$ technique results have an obvious systematic error during the transient period. This result illustrates that although a method can be a more powerful estimation technique for narrow range data (a close fit

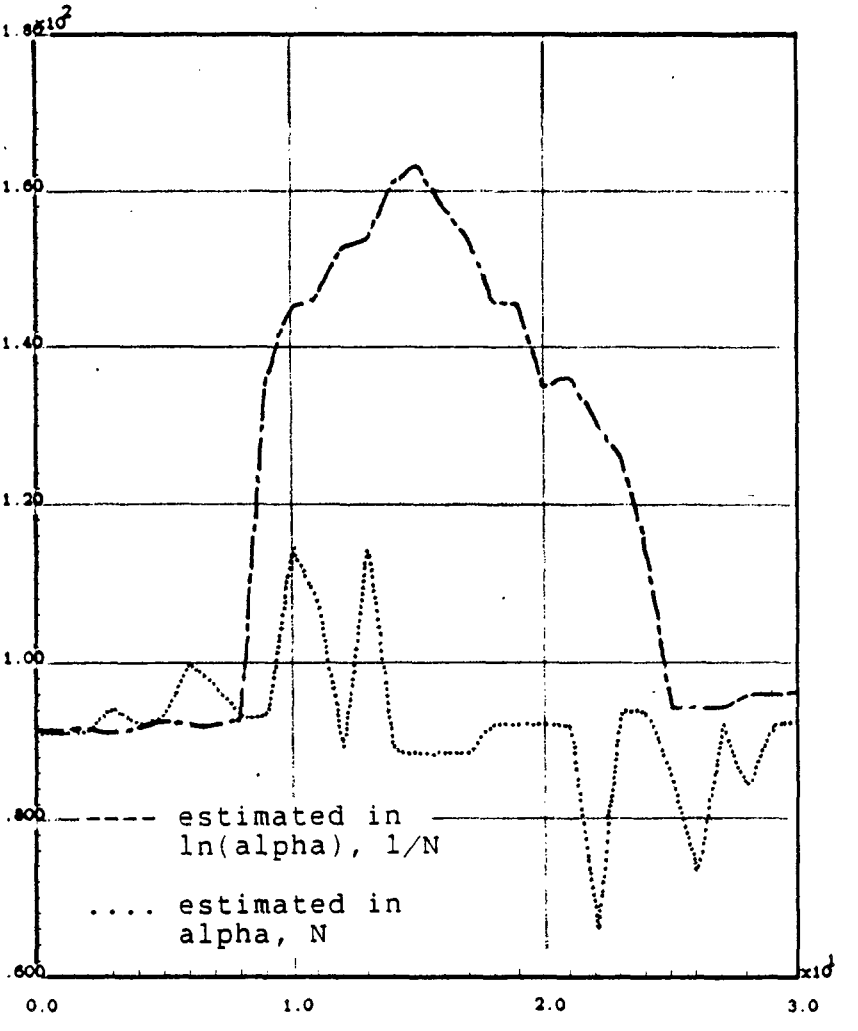

Fig.5.3.9. $\hat{N}$ *while Updating of the Data Base*

is possible), it is not necessarily for a wider range of data (no close fit is possible). For data covering a wider scope, it may not "average out" information so well. If there is no systematic model error, the estimation criterion is related to the probability that a combination of parameters predicts the plant behavior as given in the data base. In this example, the data base corresponds to one

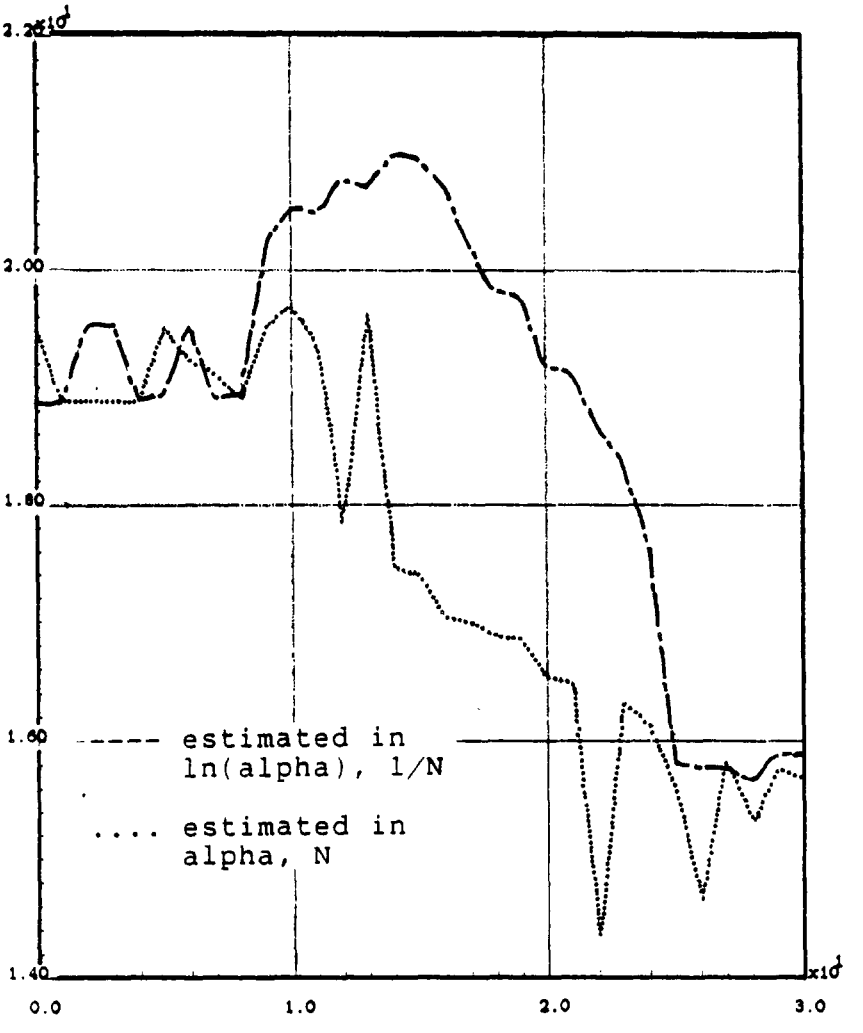

Fig.5.3.10. Optimal Reflux Ratio Corresponding to Parameter Estimates in Previous Two Figures and the Nominal Operating Conditions

steady state at the start of the experiment, and to another steady state at the end of the experiment. In between, two steady states are represented in the data base. In the beginning of the experiment, the estimation criterion is related to the probability that a combination of parameters predicts the behavior of the

plant for the initial steady state. At the end of the experiment, the estimation criterion is related to the probability that a combination of parameters predicts the behavior of the plant for the final steady state. It would be ideal if the most probable combination of parameters would shift in a smooth way from the initial to the final combination of most probable model parameters, without excursions. The results obtained in the α, N space (dotted line) show such a desirable transient, in spite of some spikes due to problems with parameter correlation. But the results obtained in the $\ln \alpha$, $\frac{1}{N}$ do not show the desirable behavior. The transformation of the parameter space made the estimation more sensitive to mixed data bases. In order to avoid implementation of such a "inverse response", sensitivity results can be combined with a statistical analysis as is shown in Chapter 3.

5.3.1.3.5. Conclusions

In this section some problems with model fitting in an on-line optimizer were illustrated by results obtained from the distillation tower case study. First, the correlation among estimated model parameters is demonstrated. It is pointed out that important correlation among parameters has to be checked for and avoided if possible. The nonlinearity of the correlation causes convergence problems in this case study. A simple nonlinear transformation solves these problems, which are demonstrated potentially to cause significant losses of performance. However, it was also illustrated that the same nonlinear transformation can cause transient problems.

In the next section, a justification if presented for the calculation method for obtaining confidence intervals and other statistical characterizations form the $\ln \alpha$, $\frac{1}{N}$ estimation technique results.

5.3.1.4. Confidence and Significance

Results of a statistical analysis allow one to evaluate the status of the parameter estimation result. The results are combined in a later section (5.4) with results of the sensitivity analysis that is carried out on the optimization result. More important statistical characterizations of the estimation results are contained in the confidence intervals and the parameter value significance. The theory on which the calculation of these statistical characterizations is based, is summarized in Chapter 3. These theorems start out from a number of assumptions. How important are these assumptions, and how crucial are they in this case study ? Are they met ?

The most important assumptions are listed here. It is assumed that the errors in the data are normally distributed with a zero mean and a constant variance. The model is assumed to be linear in the parameters. If these assumptions are met, then the estimation residuals are also normally distributed with zero mean and constant variance.

In general, one could argue that the requirement of linearity of the model in the model parameters only has a modest importance. If the confidence intervals are small because data and model are good (the mismatch is small), then the model is approximately linear in the model parameters for these small perturbations. This reasoning assumes of course continuity of the model with respect to the model parameters. The confidence intervals found by assuming that the model is linear in the model parameters are good approximations of the true confidence intervals for the non-linear model. The larger the confidence intervals are, the worse is their approximation. However, if the confidence intervals are large, accuracy becomes less important. If these confidence intervals are used in e.g. significant tests, large errors in confidence intervals become undesirable.

The requirement of normality of the distribution of the errors in the raw

data is quite important. In practice, bias is very often present in raw data. Under certain conditions, data reconciliation can improve this condition considerably. The assumption of normal noise is often incorrect, but far more frequently the only possible assumption. In the simulations in this case study, normal noise is applied. It is important to take into account the correlations between the errors if they are significant. Correlations can for instance be due to a modeling error, or to feedback control.

If the noise in the raw data is normal with zero mean, and the model responses are linear in the data, then the model - plant error values (residuals) are normally distributed with zero mean too. However, the estimation models used in this example are nonlinear in the data as has been discussed before. Normal noise is added to data generated using the Eduljee and Fenske model (Appendix G - Run # 8). The following graph (5.3.11) shows the resulting distribution of residuals of the parameter estimation using the estimation method in the $\ln \alpha$, N^{-1} space. The other techniques give comparable results. The distribution shows a tendency to be skewed, but also tends to be symmetric.

5.3.2. Boiler Study

5.3.2.1. Introduction

5.3.2.2. Correlation between Parameter Estimates

In the boiler case study, the model used by the on-line optimizer is a simple second order polynomial relation between boiler load and efficiency. Furthermore, a linear relation is used between the product of fuel load and efficiency, and the boiler load. This linear relation is accurate and very easy to fit, and the model parameters do not change significantly with the operating conditions. For

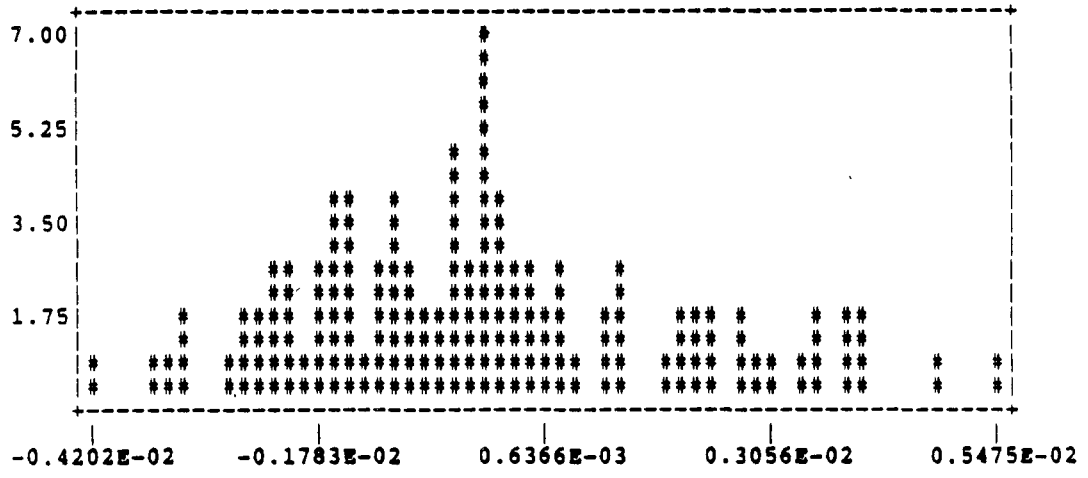

Fig.5.3.11. Histogram of Distribution of Residuals of Parameter Estimation of the Eduljee Model

every boiler, five parameters are updated. Three of these parameters determine the second order relation between boiler load and efficiency. The parameter estimates are correlated significantly. At nominal network load $l_{tot} = 72.8$lb/s with optimal allocation $l_1 = 25.5$lb/s, $l_1 = 30.3$lb/s and $l_1 = 17.0$lb/s (more details in Appendix G, Run # 9), the model parameter estimates are given in table 5.4.3. The model parameter estimate covariance matrix is block diagonal :

$$Q = \begin{pmatrix} Q_1 & 0 & 0 \\ 0 & Q_2 & 0 \\ 0 & 0 & Q_3 \end{pmatrix}; \tag{5.3.14}$$

with Q_i the covariance matrix for boiler i. The off-diagonal sections are zero. The model parameters of the different boilers are not correlated. The matrices Q_i are given below :

$$Q_1 = \begin{array}{c} \\ a_1 \\ b_1 \\ c_1 \end{array} \begin{array}{c} \begin{array}{ccc} a_1 & b_1 & c_1 \end{array} \\ \begin{pmatrix} 3.734 \cdot 10^{-2} & -2.978 \cdot 10^{-3} & 5.760 \cdot 10^{-4} \\ -2.978 \cdot 10^{-3} & 2.391 \cdot 10^{-4} & -4.652 \cdot 10^{-6} \\ 5.760 \cdot 10^{-4} & -4.652 \cdot 10^{-6} & 9.090 \cdot 10^{-8} \end{pmatrix} \end{array}; \tag{5.3.15}$$

$$Q_2 = \begin{array}{c} \\ a_2 \\ b_2 \\ c_2 \end{array} \begin{array}{c} \begin{array}{ccc} a_2 & b_2 & c_2 \end{array} \\ \begin{pmatrix} 1.920 \cdot 10^{-1} & -1.535 \cdot 10^{-2} & 2.970 \cdot 10^{-4} \\ -1.535 \cdot 10^{-2} & 1.233 \cdot 10^{-3} & -2.399 \cdot 10^{-5} \\ 2.970 \cdot 10^{-4} & -2.399 \cdot 10^{-5} & 4.694 \cdot 10^{-7} \end{pmatrix} \end{array}; \qquad (5.3.16)$$

$$Q_3 = \begin{array}{c} \\ a_3 \\ b_3 \\ c_3 \end{array} \begin{array}{c} \begin{array}{ccc} a_3 & b_3 & c_3 \end{array} \\ \begin{pmatrix} 5.042 \cdot 10^{-2} & -4.022 \cdot 10^{-3} & 7.778 \cdot 10^{-5} \\ -4.022 \cdot 10^{-3} & 3.229 \cdot 10^{-4} & -6.281 \cdot 10^{-6} \\ 7.778 \cdot 10^{-5} & -6.281 \cdot 10^{-6} & 1.228 \cdot 10^{-7} \end{pmatrix} \end{array}. \qquad (5.3.17)$$

The boiler efficiency is given as $\eta_i = a_i + b_i l_i + c_i l_i^2$. In the numbers given, the boiler load is given in lb/s, and the boiler efficiency is in %. The covariances are significant, and therefore they have to be included in the accuracy calculations as well as the checks for significant changes in set points, as can be seen in section 5.4. Since the model is linear in the parameters, the fitting does not experience any problems from the parameter correlation. The parameter estimation is a simple least squares problem. The distribution of the residuals is approximately normal and therefore the variances can be used to calculate to confidence intervals.

5.3.2.3. Data Spread

The steady states that are collected in the steady state data base, represent the behavior of the steady state system with respect to changes in the input. Therefore the selection of these points is very important. If too many points are collected, the forgetting process is too slow. If not enough points are collected, the process behavior is not represented properly. It is also important to collect significantly different steady state data. If the same data are collected over and over again, oscillations around the optimum can be expected if the plant inputs do not change. As the plant settles around the optimum, similar data are

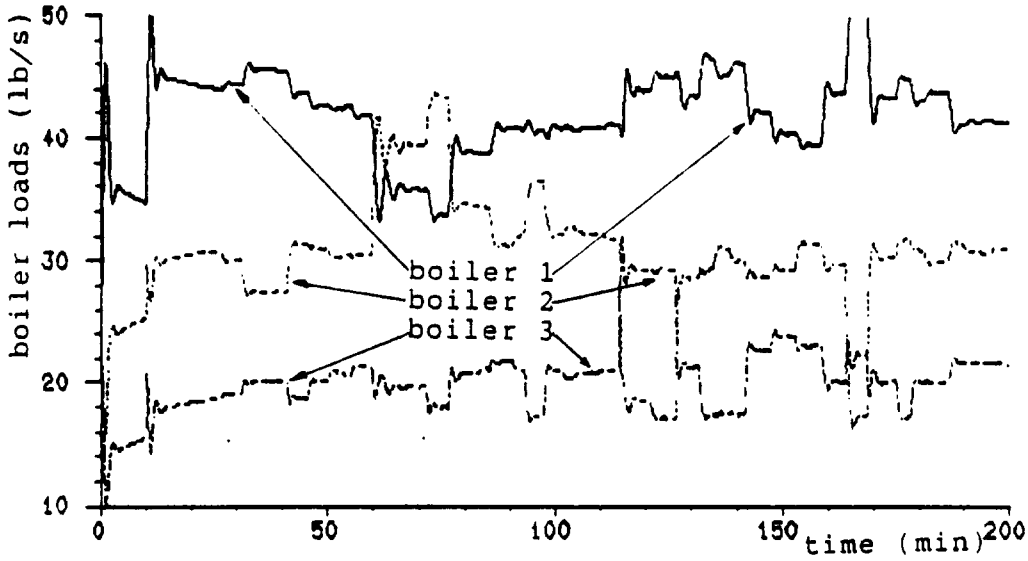

Fig.5.3.12. Individual Boiler Loads as a Function of Time with Constant Steam Load. New Steady State Data Are Not Checked for Significant Difference with Data Already Stored.

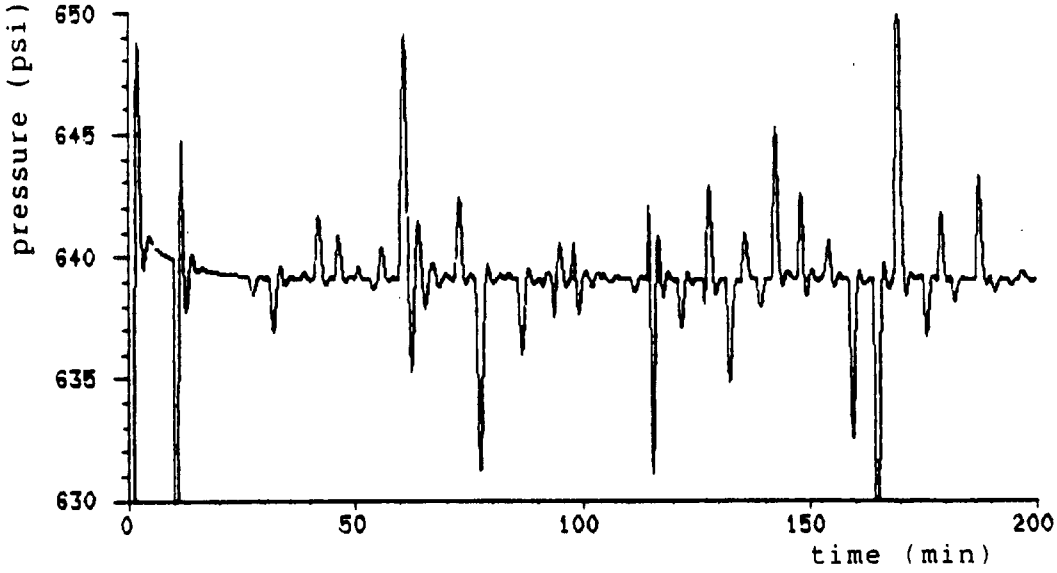

Fig.5.3.13. Pressure as a Function of Time with Constant Steam Load. New Steady State Data Are Not Checked for Significant Difference with Data Already Stored.

collected and the estimated parameters become inaccurate. Then, the optimizer obtains an erroneous result and sends the plant to a new steady state, which allows new data to be collected. This process is repeated, which causes the process to oscillate around the optimum. This oscillation is illustrated here. In figure 5.3.12 the individual boiler loads are given as a function of time for a network load setpoint for a simulation run for which no checks are made to make certain that only new data are allowed that are significantly different from the ones stored before. It is clear that the boiler load allocation oscillates around the optimum. The oscillation frequency is function of the execution rate of the optimizer, and here the optimizer is executed every 12 seconds. If the optimizer would be executed at lower frequencies, which would be more realistic, the oscillations will still occur, but at a proportionally lower frequency.

These oscillations are of course also present in the loop control. In figure 5.3.13 the header pressure is given for the same simulation run. During the upset periods, the network efficiency goes down by approximately 0.1 to 0.05 %. The check that is normally implemented here will compare the new steady state data with the confidence intervals around the already existing data. If the new data point is within the confidence envelope of one of the existing steady state data points, then the new data is not accepted. To save time, the candidate steady state data point is compared with the newest accepted steady state first, since the probability is highest that the test will detect no significant difference there. If that test is applied, the curves in figures 5.3.12–13 are straight lines from the time that the optimal allocation is reached. In the figures 5.3.12–13 the check for significant difference between the new steady state and the old steady states are omitted for every boiler. If they are only omitted for one boiler, the results are less confusing. Figures 5.3.14 and 5.3.15 give examples.

5.3.2.4. Unsteady State Data

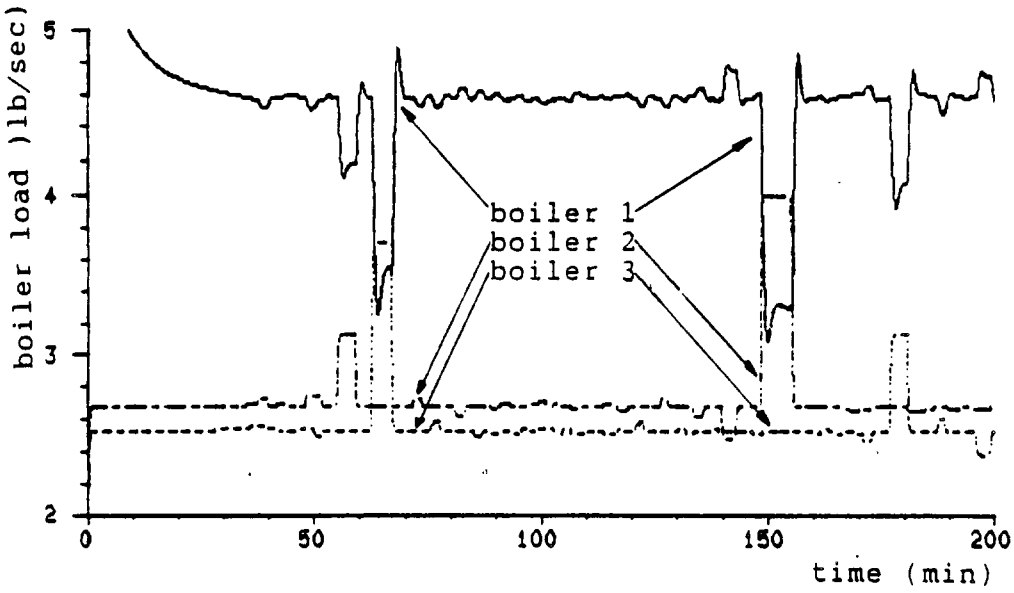

Fig.5.3.14. Individual Boiler Loads as a Function of Time with Constant Steam Load. New Steady State Data Are Not Checked for Significant Difference with Data Already Stored for Boiler 1.

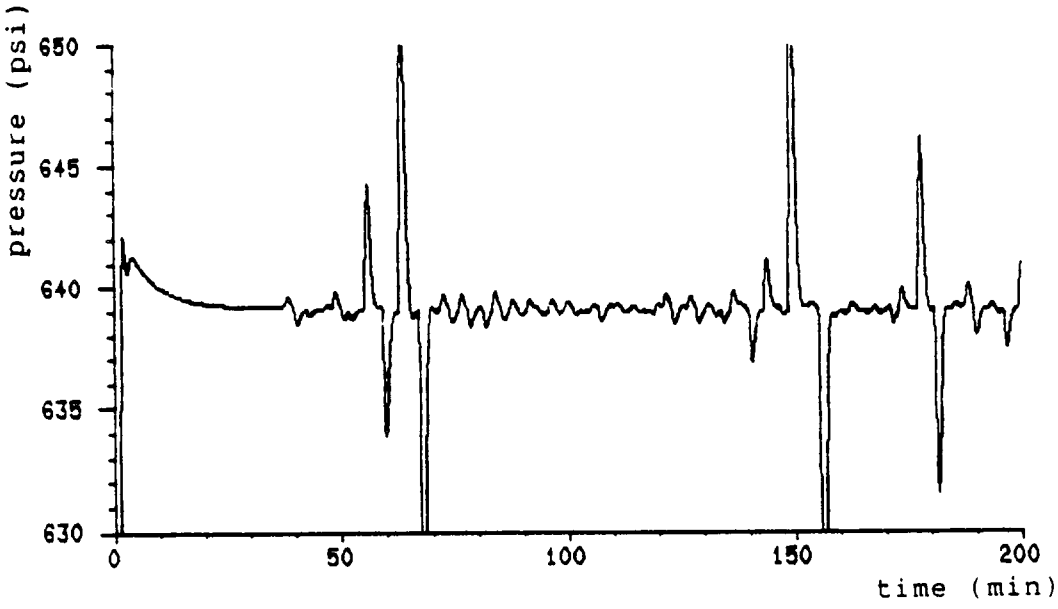

Fig.5.3.15. Pressure as a Function of Time with Constant Steam Load. New Steady State Data Are Not Checked for Significant Difference with Data Already Stored for Boiler 1.

Boiler dynamics are function of the boiler load. Hot flue gasses are generated in the fire box, and travel to the stack while exchanging heat to produce steam. The volume of a boiler on the gas side is constant. If the production of flue gas is increased by a higher firing rate, the residence time for the flue gas becomes shorter. The same happens on the water side of the boiler. As the boiler load increases, the boiler time constants decrease. A boiler is significantly slower at low loads then at high loads. Changes in boiler dynamics can cause problems in the steady state detection. Steady state detection criteria that work for an average boiler load, may detect steady state too early at low loads. Unsteady state data will deteriorate the quality of the model updating. In figure 5.3.16 three efficiency curves are given. One curve gives the true efficiency to load relationship for boiler 2. The initial model that is used by the on-line optimizer is very close to this curve. Two more curves are given, marked "Fit 2" and "Fit 3". These are updated fits that were obtained after a significantly new data point is detected. The network load is increased in time as listed in Table 5.4.4. It is clear that the updated curves fit the true behavior poorly at low loads. More details can be found in Appendix G (Run # 10).

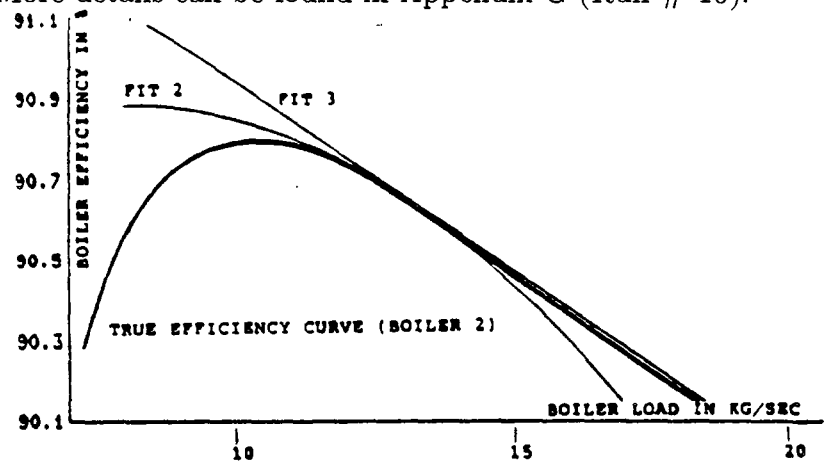

Fig.5.3.16. True Boiler 2 Efficiencies and Second Order Polynomial Fits Used by the On-Line Optimizer as New Steady State Results Becomes Available.

5.4. SENSITIVITY

In the sensitivity results that are presented here, the vector ε combines a number of the variables that describe the optimization problem. Optimization variables cannot be included in ε. Model parameters, load measurements, reactor bed temperatures, on-line analysis results and any other important variable with influence on the optimization problem and its solution can be included. First, sensitivity results obtained in the distillation case study are presented. Then an overview of the sensitivity results obtained in the boiler case study is given.

5.4.1. Distillation

5.4.1.1. Introduction

This section discusses applications of sensitivity analysis in the distillation case study. Some sensitivity results obtained under nominal operating conditions are presented first. Then, the results of a short cut feasibility study are given. Other applications are illustrated in the boiler case study in section 5.4.2.

At this point, it should be emphasized that the objective function of the column optimization as presented in Chapter 4,

$$f_1 = (c_1 + c_2)\lambda V + W x_b B + U(1 - x_d)D, \qquad (4.2.6)$$

is not the total operating cost but only that part of the operating cost which depends upon the reflux ratio and the top composition. The remaining part of the operating cost can still be function of the elements of ε. The derivatives of the remaining part of the operating cost with respect to variables in ε have to be calculated separately and have to be added to the derivatives of the optimal variable operating cost that are calculated in the sensitivity analysis. Results for

Symbol	**Sensitivity** (unit)
c_1	$0.206 \cdot 10^{10}$ (BTU/day)
c_2	$0.206 \cdot 10^{10}$ (BTU/day)
N_{true}	-91.7 (\$/day)
U	43400. (lb/day)
W	14700. (lb/day)
λ	47.4 $(\frac{\$ \cdot BTU}{lb \cdot day})$
α	$-0.133 \cdot 10^6$ (\$/day)
$x_{D,min}$	$0.124 \cdot 10^6$ (\$/day)
x_F	$-14000.$ (\$/day)
F	$8.5 \cdot 10^{-3}$ (\$/lb)

Table 5.4.1: Sensitivity of the Optimal Value Function, the Optimal Variable Operating Cost, for Various Variables (A Legend for the Symbols can be Found in Tables 4.2.1-2.)

the nominal case are listed in Table 5.4.1. More details on how these numbers are obtained can be found in Appendix G (Run # 11).

The results for the sensitivity of $[-(f - f_1)]$, the part of the operating cost that does *not* depend upon the optimization variables, can be obtained by differentiating $[-(f - f_1)]$, which can be written as :

$$[-(f - f_1)] = c_d x_f F + c'_b(1 - x_f)F - c_f x_f F - c'_f(1 - x_f)F, \qquad (5.4.1)$$

and by using the nominal values and the definitions for the light and heavy differential values :

$$W = c_d - c_b \qquad (5.4.2)$$

$$U = c'_b - c'_d \qquad (5.4.3)$$

and their derivatives.

Sensitivities are hard to compare. They have different dimensions. It is easier to look at small equally likely changes of the operating conditions, and compare the consequences for the optimal value function. For instance, an increase in the reboiler heating cost of 10 cents per million BTU (on $ 3 per million BTU) makes the optimal operating cost increase with about 200 $\frac{\$}{day}$ (205.56). A loss of 2% of the tray efficiency, resulting in a decrease in the number of ideal trays, increases the operating cost of the column about 250 $\frac{\$}{day}$ (247.71). It has to be stressed that these sensitivities are derivatives of the *optimal variable* operating cost at the nominal operating conditions, and are not derivatives of the objective function. A derivative of merely the objective function does not include a compensation for the optimization or maintenance of the constraints.

The influence of the variables grouped in ε on the solution (=the optimal setpoints as opposed to the optimal value function) of the optimization problem. In this case only the influence on the reflux ratio is important. The top purity sensitivity is zero for all variables, since the top purity constraint is constant over the complete range of operation that is considered here.

The following Table (5.4.2) lists the sensitivities of the optimal reflux ratio for various variables. Also for these results, it is easier to compare sensitivities by investigating the results of small changes in the variables on the optimal reflux ratio. For instance, an increase of 10 cents per million BTU in the heating cost (nominal value) would have almost no influence on the optimal reflux ratio ($+10^{-18}$), and a decrease of 2 % tray efficiency would increase the optimal reflux ratio 0.05, also relatively small. The feed mass flow rate has a stronger influence on R, but the strongest influence probably comes from the required top purity. Sensitivity analysis allows one to identify the most important parameters in the

Symbol	Sensitivity
c_1	$-0.108 \cdot 10^{-10}$ (BTU/\$)
c_2	$-0.108 \cdot 10^{-10}$ (BTU/\$)
N_{true}	$-0.171 \cdot 10^{-1}$
U	$-0.172 \cdot 10^{-8}$ (lb/\$)
W	$0.320 \cdot 10^{-7}$ (lb/\$)
λ	$-0.125 \cdot 10^{2}$ (lb/BTU)
α	$-0.163 \cdot 10^{3}$
$x_{D,min}$	$0.117 \cdot 10^{4}$
F	$0.859 \cdot 10^{-2}$(day/lb)
x_F	$-0.359 \cdot 10^{2}$

Table 5.4.2: Sensitivity of the Optimal Reflux Ratio for Various Variables Determining the Column Operation (A Legend for the Symbols can be Found in Tables 4.2.1-2.)

problem description.

5.4.1.2. Feasibility Study

The results of a feasibility study are presented in this section. The technique that is presented in section 3.6.3.2, is applied here. Whether or not the application of an on-line optimizer is feasible from the point of view of plant economics, depends upon many factors. The cost of the installation of an on-line optimizer depends upon the level of instrumentation already available in the plant. The operation of an on-line optimizer on a distillation column requires more instruments than the minimum required for feedback control of both compositions. The on-line optimization algorithm requires some computer

time. For the models and estimation techniques used in this case study CPU time on the order of magnitude of a few seconds on a MicroVax II in MicroVMS is needed for optimization and sensitivity analysis depending upon initial guess and operating conditions. This CPU time may be available at marginal cost on an existing plant computer, or a new machine may have to be purchased and hooked up to the plant. Finally, the structure for the implementation of the optimization results has to be provided. This may be not more than a disk access or a computer communication, but communication protocols may be expensive to develop. In a publication by Poje and Smart [1986] (see Chapter 2, section 2.2.4) the cost for purchasing a small computer for running the optimization task in real time and for attaching this computer to the process, is estimated at approximately $ 700,000. Operators have to be trained to work with the optimizer. Furthermore, the on-line optimizer has to be maintained. Optimizer maintenance includes model structure updating and optimization problem formulation updating.

The potential profit of the on-line optimizer is difficult to estimate. One could compare the expected performance of the real-time optimized system with a system that is optimized at nominal operation. The reference control policy is very important, and it should correspond to the best possible result that can be obtained without investing in a real-time optimization system. That method is illustrated on the boiler case study in the next section. Another critical issue is the comparison of the expected accuracy of the on-line optimizer (how close is the estimated optimum expected to be to the true minimum) to the range of operation. If the accuracy is small compared to the range, then on-line optimization is not likely to be attractive, and vice versa. Therefore an initial feasibility study should include an accuracy study. This study can be based upon relatively few data, and requires few calculations. An accuracy study will

allow one to distinguish between processes that are definitely interesting candidates for on-line optimization, and processes that are definitely not. In between, some processes will be found for which an accuracy study seems indecisive and for which more data are needed. An accuracy test will detect processes that are clearly uninteresting subjects for on-line optimization and why they are not interesting in an early phase. The only information used in this rough accuracy estimate is :

- ◇ The steady state plant model used by the optimization algorithm as equality constraint.
- ◇ The first order optimal function sensitivity results obtained for the nominal steady state (a base case).
- ◇ Expected accuracy for various data.

The following assumptions are made :

- ◇ Except for the estimated model parameters, inaccuracies in all data used are uncorrelated.
- ◇ The model is perfect.
- ◇ All errors are random and normal.
- ◇ Accuracy of the measurement device is a good indication of the 95% confidence interval.
- ◇ It is accurate to assume that the models behave linearly within the range of inaccuracies around the nominal state.

In Chapter 4 as well as in the previous section, the total operating cost of the C_3 splitter, f, is split into a part f_1, grouping all the terms that depend upon the optimization variables and a remainder, which from here on will be denoted f_2, that is independent of the optimization variables and therefore is not included in the optimization problem. The optimal value of f_1 is referred

to as $f_1^\star$. Then,

$$f^\star = f_1^\star + f_2, \tag{5.4.4}$$

with $f^\star$ the optimal total operating cost. $f^\star$, $f_1^\star$ and f_2 can be considered as random variables, and their values are samples of certain distributions. It will be assumed here that these distributions are normal. In this accuracy study, f_2 will not be considered. Changes in the variables included in ε can influence the value of f_2. Therefore, although the sensitivity results for $f_1^\star$ may indicate that e.g. a positive change in ε_k for a certain k would increase the performance that is included in f_1, this is not necessarily true for the overall performance f. From Chapter 3 :

$$\sigma^2_{f_1^\star} = \left(\nabla_\varepsilon f_1^\star|_0\right)^T Q_\varepsilon \left(\nabla_\varepsilon f_1^\star|_0\right). \tag{3.6.30}$$

If the covariances between the various variables in the ε are zero or can be neglected, this equation can be simplified as :

$$\sigma^2_{f_1^\star} = \sum_i \left(\left[\frac{\partial f_1^\star}{\partial \varepsilon_i}\right]^2 \sigma^2_{\varepsilon_i} \right) \tag{5.4.5}$$

In these equations, ε_i are variables in the vector ε. In this example these variables are the reboiler heat cost, the condenser cooling cost, the average latent heat, the feed mass flow rate, the feed purity and the model parameters (α and N). The light and heavy key differential values are considered constant. The terms $\frac{\partial f_1^\star}{\partial \varepsilon_i}$ are the first order optimal function sensitivities that were calculated and listed before (Table 5.4.1).

First we look at the accuracy of the parameter estimation and how this influences the accuracy of the optimization. The parameter estimates are strongly correlated, and therefore a covariance factor has to be included. Because it is difficult to determine the covariance between both parameter estimates in an analytical way, the covariance will be obtained using a Monte Carlo simulation. Model parameters α and N^{-1} are estimated from a "good" data base,

i.e. a database spread evenly around the nominal operation with a reasonable amount of noise. Using data generated with Eduljee's model, with hundred randomly perturbed inputs around the nominal input and adding 5% noise to the feed mass flow, bottom propylene fraction, bottom and top flow, and 5% noise to the propane fraction in feed and top, parameters are estimated in the $\ln \alpha$, N^{-1} space. Figure 5.4.1 presents the generation of these data schematically. Data reconciliation and outlier detection are applied. More details on this Monte Carlo simulation can be found in Appendix G (Run # 12). The following results are obtained :

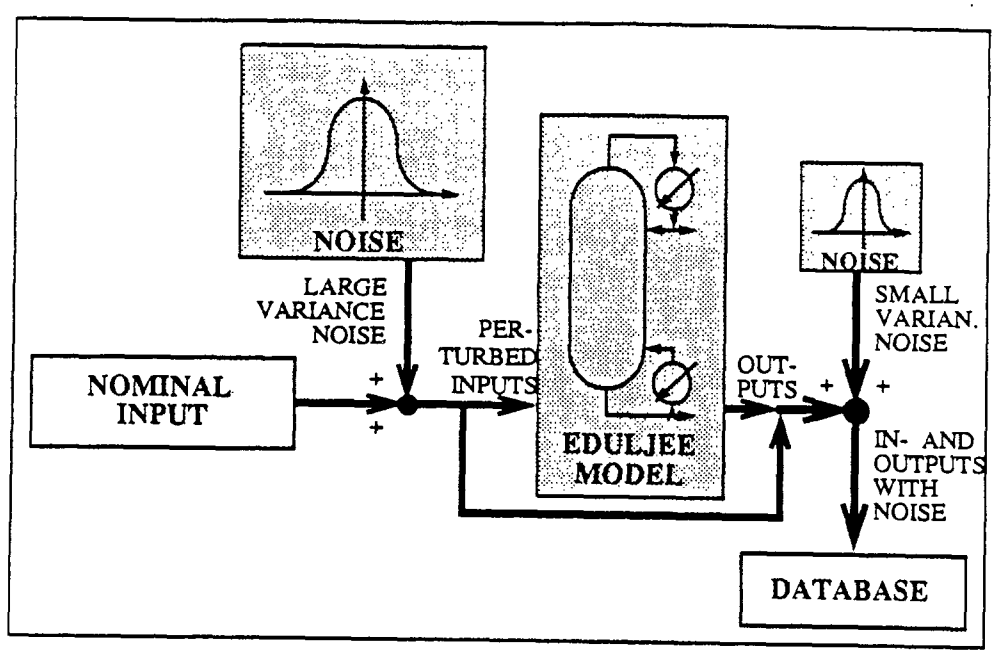

Fig.5.4.1. Schematic Presentation of the Generation of a Data Base for Monte Carlo simulation of parameter covariances.

$$\alpha = 1.105$$

$$N = 94$$

$$\sigma^2_{\ln \alpha} = 0.3311 \cdot 10^{-8} \tag{5.4.6}$$

$$\sigma^2_{N-1} = 0.1022 \cdot 10^{-6}$$

$$\sigma^2_{\ln \alpha, N-1} = -0.1829 \cdot 10^{-7}$$

Applying :

$$\sigma^2_{f_1^\star} = \left(\frac{\partial f_1}{\partial \alpha} \cdot \alpha\right)^2 \sigma^2_{\ln \alpha} + \left(\frac{\partial f_1}{\partial N} \cdot N^2\right)^2 \sigma^2_{N-1} + 2 \cdot \left(-\frac{\partial f_1}{\partial N} \cdot N^2\right)\left(\frac{\partial f_1}{\partial \alpha} \cdot \alpha\right) \sigma^2_{\ln \alpha, N-1} \tag{5.4.7}$$

and (from Table 5.4.1) :

$$\begin{aligned} \frac{\partial f_1}{\partial \alpha} &= -0.132 \cdot 10^6 \frac{\$}{day} \\ \frac{\partial f_1}{\partial N} &= 83.7 \frac{\$}{day} \end{aligned} \tag{5.4.8}$$

then :

$$\sigma^2_{f_1^\star} \simeq 52 \cdot 10^3 \left(\frac{\$}{day}\right)^2 \tag{5.4.9}$$

The inaccuracy of the on-line optimization operations could be expressed as a confidence interval. Then the 95% confidence interval of $f_1^\star$ is about 450 $\frac{\$}{day}$. That is about 8.5% of the variable cost of the tower listed by Martin *et al.* [1981]. It is about half of the profit listed by the same author. However, in the example of Martin *et al.* [1981], both states compared are relatively close. Even if this would correspond to the actual operating range, then still a 97.5% chance would exist that the profit is larger than approximately 390 $\frac{\$}{day}$ ($+140,000 \frac{\$}{year}$) and a 50% probability that the profit is over 840 $\frac{\$}{day}$. These figures indicate that this application is interesting.

The results that were just obtained quantify the accuracy of the predicted optimal value function that one can expect for the specified noise in the system

inputs and outputs if one assumes that the inaccuracies corresponding to this noise are transmitted to the optimal value function only through model parameter inaccuracies. This calculation assumed that all other factors, such as reboiler heat cost, latent average heat etc. were all known and exact. If inaccuracies in the feed purity, the latent heat and the reboiler heat cost are considered, using $\sigma^2_{\text{feedpurity}} = 10^{-4}$, $\sigma^2_{\text{avglatheat}} = 2\,(\frac{BTU}{lb})^2$, $\sigma^2_{\text{reboilercost}} = 2.5 \cdot 10^{-15}\,(\frac{\$}{BTU})^2$, then the global variance for $f_1^\star$ becomes :

$$\sigma^2_{f_1^\star} \simeq 86 \cdot 10^3 (\frac{\$}{day})^2. \tag{5.4.10}$$

This variance corresponds to a 95% confidence interval of 580 $\frac{\$}{day}$ in the optimal value function. If a feed purity between 0.65 and 0.75 can be expected, the reboiler heat cost varies between $2.7 \cdot 10{-}6$ and $3.3 \cdot 10^{-6}$ $\frac{\$}{BTU}$, and the latent heat ranges between 110 and 150$\frac{BTU}{lb}$ then the optimal variable cost is expected to vary over a range of 4500$\frac{\$}{day}$. The ratio between two times the confidence interval and the expected range of the expected profit equals :

$$\frac{2 \times 580 \quad \$/day}{4500 \quad \$/day} \simeq \frac{1}{4}. \tag{5.4.11}$$

This indicates that the variable operating cost would have to change over about a third of it expected range before a significant change in profit with 95 % certainty can be detected, *if* errors are uncorrelated in time. It has to be stressed that this criterion is very conservative, and not a bad result. It means that if the expected profit changes over about one third of its expected range, a significant change in the true profit is certain. If the change in the expected profit is smaller than one third of its expected range, then the probability for a true profit change is still considerable.

If feed mass flow fluctuations are included in the calculation also, the picture changes. If the flow mass flow rate measurement is expected to be approximately correct up to 24000$\frac{lb}{day}$, then the variance of the optimal variable cost

increases to :

$$\sigma^2_{f_1^\star} = 96.8 \cdot 10^3 (\frac{lb}{day})^2, \tag{5.4.12}$$

corresponding to a 95% confidence interval of $600\frac{\$}{day}$. If a feed purity between 0.65 and 0.75 can be expected, the reboiler heat cost varies between $2.7 \cdot 10^{-6}$ and $3.3 \cdot 10^{-6}$ $\frac{\$}{BTU}$, the latent heat ranges between 110 and $150\frac{BTU}{lb}$ and the feed mass flow can vary between $10^6 \frac{lb}{day}$ and $1.4 \cdot 10^6 \frac{lb}{day}$, then the optimal variable cost is expected to vary over a range of $5900\frac{\$}{day}$. The accuracy still is approximately a fifth of that value.

Some results of this feasibility study are presented in figure 5.4.2. The conclusion of this study is that the on-line optimization of the reflux ratio of a C_3 splitter can be interesting and the expected optimal variable cost can be determined significantly compared to its range. The uncertainty on the parameter estimates, and on the feed purity are the largest contributions to the optimal value function uncertainty. Sensitivity analysis allowed the detection of this "information bottleneck" in an unambiguous way.

Finally it has to be noted that the results of this accuracy study are function of the statistical data that is plugged into the equations. Accuracy information on measurements has to be obtained from instrument vendors or from experience. Model parameter accuracy has to be obtained from a statistical analysis on the estimation results.

5.4.2. Boiler Case Study

5.4.2.1. Sensitivities

As an introduction to this section on sensitivity results in the boiler case study, some sensitivities for the optimal value function (optimal boiler network efficiency) for changes in the various model parameters, the total load and the

SYMBOL	DESCRIPTION (UNITS)	σ^2	$\Delta\sigma^2_{f_1^*}$	RANGE	95% CONF.
$\ln\alpha$	MODEL PARAMETER relative volatility ( )	1.3311 $\times 10^{-8}$ | -0.1829 $\times 10^{-7}$	52×10^3 ($/day)	*Monte Carlo*	1.13×10^{-4}
1/N	MODEL PARAMETER number of ideal trays ( )	0.1829 $\times 10^{-7}$ | 0.1022 $\times 10^{-6}$		*Monte Carlo*	6.27×10^{-4}
$f_1^*(\alpha,N)$	OPTIMAL VALUE FUNCTION as function of α and N ($/day)	52×10^3 ($/day)			450
c_1	reboiler heat cost ($/BTU)	2.5×10^{-15}	10.6×10^3	$2.7 - 3.3 \times 10^{-6}$	$1c/10^6$BTU
λ	average latent heat (BTU/lb)	2.	4.5×10^3	110 - 150	3
x_f	feed purity ( )	10^{-4}	19.6×10^3	0.65 - 0.75	0.02
$f_1^*(\alpha,N,\ldots c_1,\lambda,x_f)$	OPTIMAL VALUE FUNCTION as a funtion of α, N, c_1, λ, x_f ($/day)	86×10^3		4500	580
F	feed mass flow (lb/day)	150×10^6	10.8×10^3	$10^6 - 1.4 \times 10^6$	24000
$f_1^*(\alpha,N,\ldots c_1,\lambda,x_f,F)$	OPTIMAL VALUE FUNCTION as a funtion of α, N, c_1, λ, x_f, F ($/day)	97×10^3		5900	600

Fig.5.4.2. Summary of Results of Short-Cut Feasibility Study. Table Columns 1 and 2 Specify the Variables. All Numerical Results are Given for $\ln\alpha$, $1/N$. The Third Column Gives the Variance per Variable. The Fourth Column Gives the Contribution of each Variable to the Variance of the Optimal Value Function. The Fifth Column Gives the Range over which the Variable is Expected to Vary. The Sixth Column Gives a 95 % Confidence Interval for the Variable.

minimal load requirement are presented. In the remaining parts of this section on sensitivity results for boilers, on- and off-line applications are discussed.

Consider three boilers feeding a common header as presented in section 4.3. The boiler specifications are the ones given in Table 4.3.2, and efficiency curves are given in figures 4.3.7 and 4.3.10. The objective of the optimization is to maximize the average network efficiency, while meeting the total steam demand. Minimal (17 lb/sec) and maximal (43 lb/sec) individual boiler loads are set, and are equal for all boilers. The individual boiler efficiency is given as $\eta_i = a_i + b_i l_i + c_i l_i^2$ with l_i the individual load for boiler i. Coefficients are listed in Table 5.4.3.

#	a_i coefficient	b_i coefficient	c_i coefficient
Boiler 1	88.20	0.2197	$-4.639 \cdot 10^{-3}$
Boiler 2	87.43	0.3101	$-5.997 \cdot 10^{-3}$
Boiler 3	88.78	0.1275	$-3.075 \cdot 10^{-3}$

Table 5.4.3. : Coefficients for Boiler Efficiency 2nd Order Polynomials
($\eta_i = a_i + b_i l_i + c_i l_i^2$)
l = Boiler Load in lb/sec, η = Boiler Efficiency in %

The optimal boiler load allocation curves for the efficiency parameters in table 5.4.3, are given in figure 5.4.3. as a function of *total* steam demand. These curves correspond to boiler 2, boiler 1 and boiler 3 respectively from highest to lowest individual load at low total steam demand level. The minimal individual boiler load constraint for boiler 3 is active for total steam demands lower than approximately 71 lb/sec. Figure 5.4.4 gives the individual boiler

and global network efficiencies at optimal boiler load allocation as a function of total steam demand. More details on how figures 5.4.3–10 are obtained can be found in Appendix G (Run # 13).

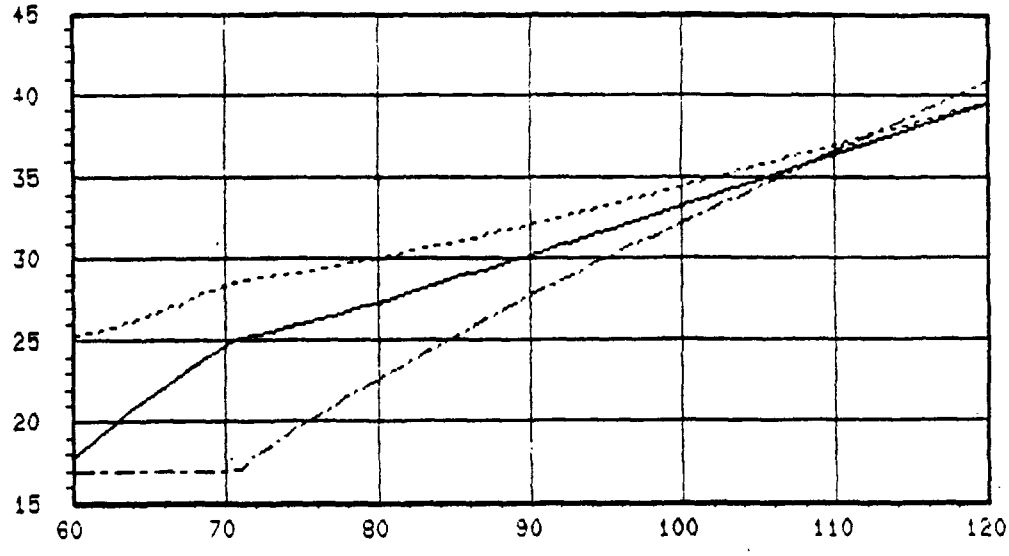

Fig.5.4.3. Optimal Boiler Load Allocations as a Function of Total Steam Demand. Boiler Loads are given in lb/sec.

Boiler1 ----------

Boiler2 ——————

Boiler3 - - - - - - -

Figure 5.4.5 gives the sensitivity of the network efficiency as a function of the total load. This sensitivity is the derivative of the corresponding curve in figure 5.4.4. It is indicated that the network in its totality has a total load for which it becomes most efficient. It should be noted that the curve is not differentiable at the point where the minimal load requirement constraint for boiler 3 becomes active. All sensitivity results are invalid at the border point where the minimum load requirement constraint for boiler 3 becomes active / inactive. At such points the optimization problem solution becomes undifferentiable, and therefore the sensitivity theorems that are presented in Chapter 3 do not hold

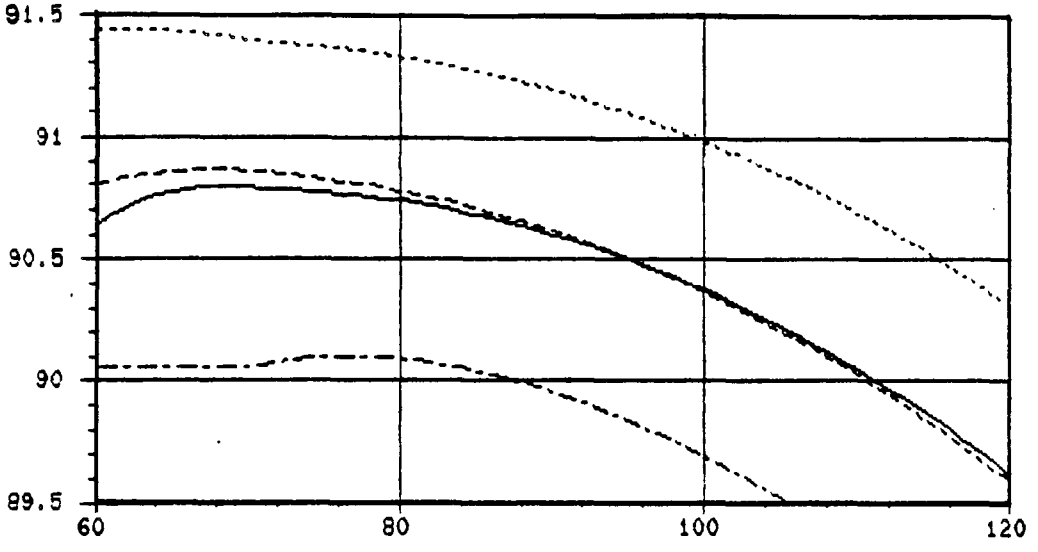

Fig.5.4.4. Individual Boiler Efficiencies and Efficiency of the Total Network for Optimal Boiler Load Allocations as given in the Previous Figure as a Function of Total Steam Demand. Efficiencies are given in %, and Boiler Loads are given in lb/sec.

Boiler1 ----------
Boiler2 - - - - -
Boiler3 --------
Total Network Average ——————

at that point.

Figures 5.4.6–8 give the sensitivity of the optimal overall efficiency for the coefficients of the second order efficiency model. Figure 5.4.6 gives sensitivities of the optimal overall efficiency (dimensionless) for the a_i coefficients, figure 5.4.7 (in lb/sec) for the b_i coefficients and figure 5.4.8 (in $(\text{lb/sec})^2$) for the c_i coefficients.

Note that also in these results the change in the active set of constraints caused a discontinuity in the derivatives of the sensitivities. Again, at that point, the sensitivity results are not valid. Sensitivity results are derivatives. For instance, $\frac{d\eta_{tot}}{da_1}$ gives the change in η_{tot} for an arbitrary small change in a_1. In all sensitivity results, it is assumed that this perturbation (i.e. da_1) is

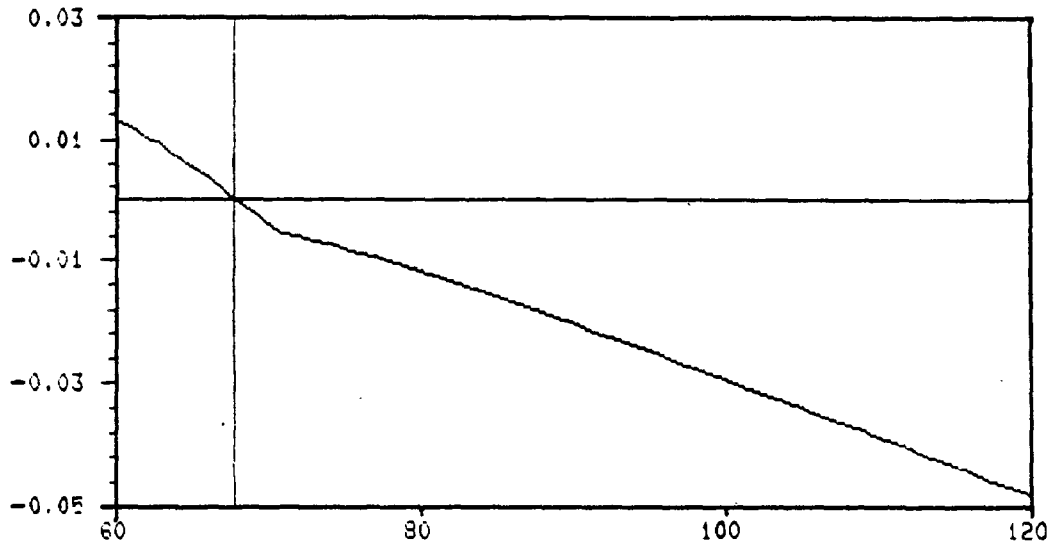

Fig.5.4.5. Sensitivity of the Global Optimal Efficiency for the total Steam Demand as a Function of the Total Steam Demand. Loads are given in lb/sec, and the Sensitivity in sec%/lb.

so small that the active set does not change. That assumption cannot hold at values of a_1 for which *any* smaller value of l_{tot} results in an active minimum load constraint and at the same time *any* larger l_{tot} results in an inactive minimum load constraint.

It should also be said that the sensitivity of the total efficiency for the model parameters of the boiler with an minimum load constraint that is active is not zero or constant, although its optimal load does not change with l_{tot}. It is clear that the global or overall efficiency will increase or decrease with the efficiency of boiler 3, even if its load is constant. The sensitivity of the overall efficiency for the model parameters of a boiler with a constant load is not constant since the *fraction* of the total steam demand that boiler 3 produces is variable with the total load even if its individual load is constant. Finally, note that the larger the individual boiler load is, the larger the sensitivity is of the overall efficiency for the model parameters of that boiler.

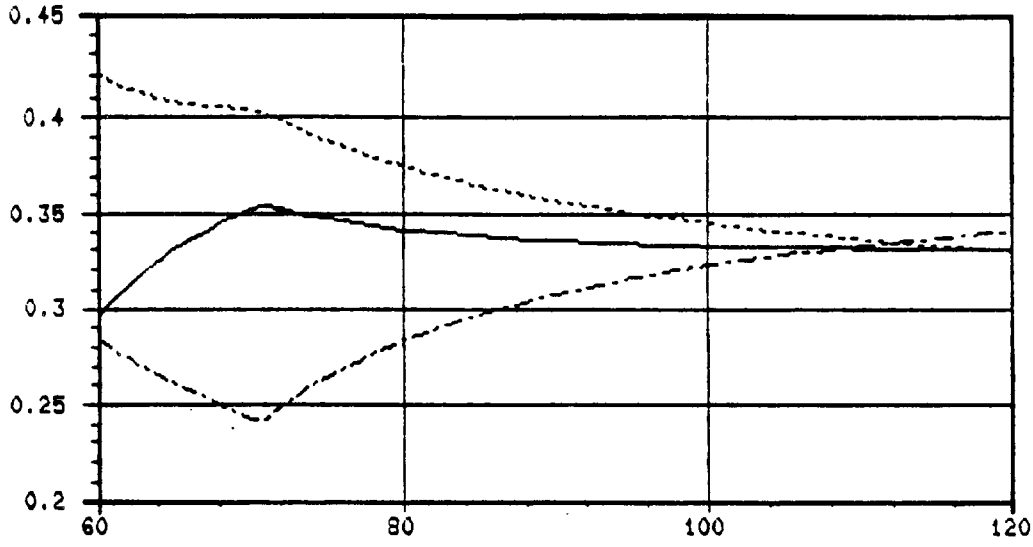

Fig.5.4.6. Sensitivity of the Optimal Overall Efficiency for the a_i Coefficients as a Function of the Total Steam Demand. The Total Steam Demand is in lb/sec. The Sensitivities are Dimensionless.

a_1 ----------

a_2 ——————

a_3 -·-·-·-·-

Figure 5.4.9 gives the sensitivity of the overall efficiency for the lower load requirement for boiler 3 as a function of the total steam demand. For loads higher than approximately 71 lb/sec the sensitivity is zero since the constraint is inactive. The magnitude of the sensitivity becomes smaller as the total steam demand rises towards 71 lb/sec. This indicates that as the total steam demand grows, the difference between the constrained and the unconstrained solution diminishes. The sensitivity given in figure 5.4.9 is negative. This is so because an *increase* in the minimal load requirement for boiler 3 would *decrease* the plant performance (i.e. the overall efficiency).

From this introductory section on applications of sensitivity analysis in the boiler case study, it is clear that sensitivity analysis provides a large amount of information that assists in evaluating the performance of an on-line opti-

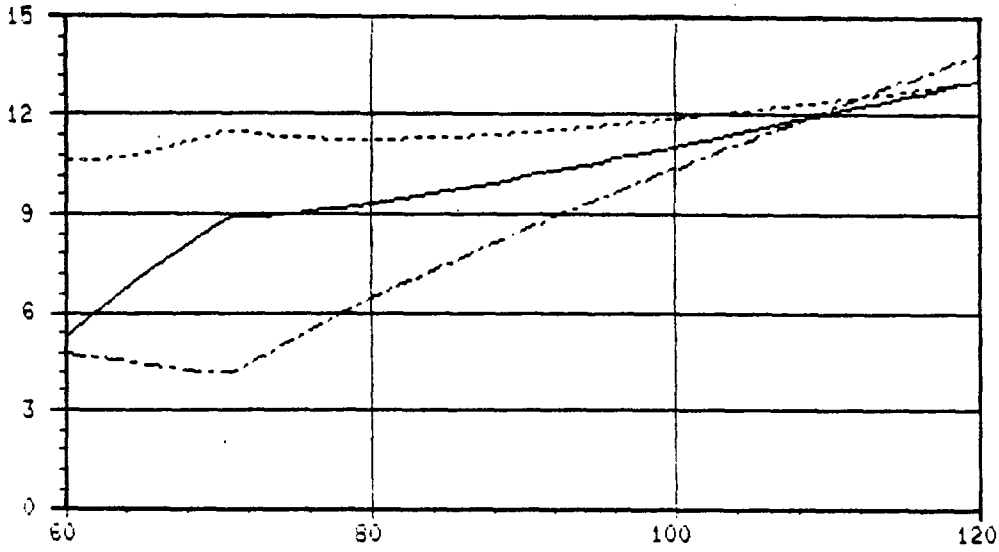

Fig.5.4.7. Sensitivity of the Optimal Overall Efficiency for the b_i Coefficients as a Function of the Total Steam Demand. The Total Steam Demand is in lb/sec. The Sensitivities are in lb/sec.

b_1 ----------
b_2 ——————
b_3 – – – – – –

mizer. In the next section some on-line applications of sensitivity analysis are presented. That is followed by a section on off-line applications.

5.4.2.2. On-Line Applications of Sensitivity Analysis

In Chapter 3 (3.6.3.3–5) methods for on-line application of sensitivity analysis are proposed. In this section the on-line applications of sensitivity analysis are illustrated.

5.4.2.2.1. On-Line Accuracy of the Expected Optimal Value

The sensitivities that are shown above (5.4.2.1) for the nominal operating conditions, can be calculated repeatedly on-line. They can be combined with statistical information on the model parameter estimates, using equation

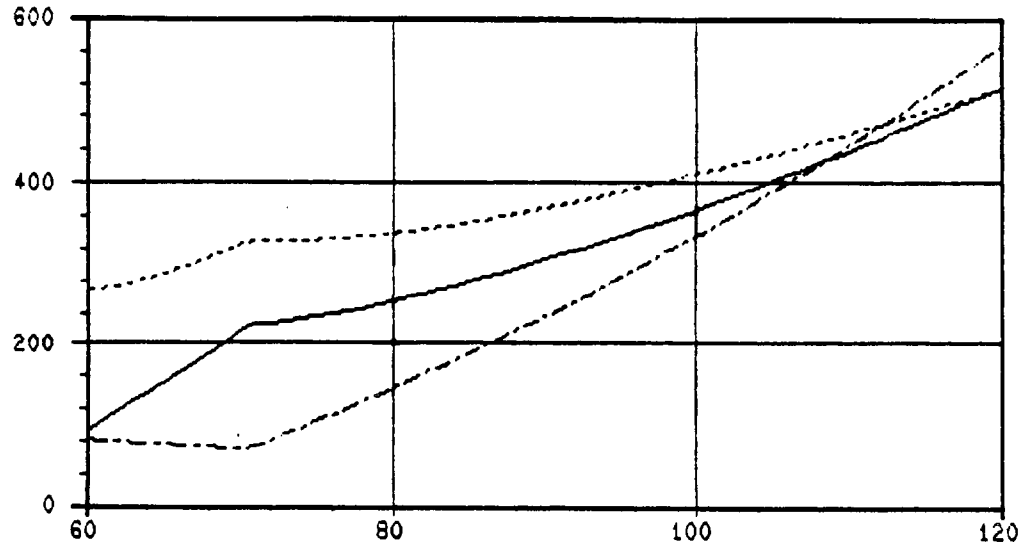

Fig.5.4.8. Sensitivity of the Optimal Overall Efficiency for the c_i Coefficients as a Function of the Total Steam Demand. The Total Steam Demand is in lb/sec. The Sensitivities are in (lb/sec)^2.

c_1 ----------

c_2 ————————

c_3 — — — — — —

(3.6.29) :

$$\sigma_{f^*}^2 = \left(\nabla_\varepsilon f^*|_0\right)^T Q_\varepsilon \left(\nabla_\varepsilon f^*|_0\right). \tag{3.6.29}$$

In this boiler case study, ε is the vector of the model parameters and the boiler network load, and Q_ε is the covariance matrix of the model parameters and the boiler load. For three boilers, this means that Q_ε is a 10×10 matrix. The variance of the optimal efficiency as a function of time is given in figure 5.4.10. In this example the total boiler network load is increasing stepwise, as listed in table 5.4.4

The on-line optimizer is run every 5 minutes. The steady state data base has a length of five data sets for every boiler. No noise is added to the run. A three percent accuracy in the total steam demand measurement was accepted as a realistic value.

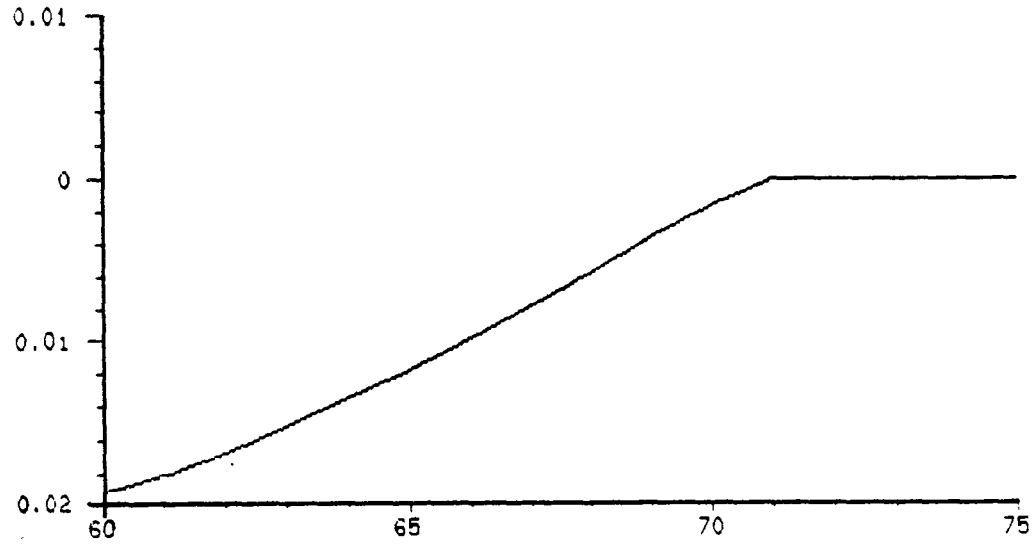

Fig.5.4.9. Sensitivity of the Optimal Overall Efficiency for the Lower Boiler 3 Load Constraint versus Total Steam Demand. Sensitivity is given in % sec/lb. Load is given in lb/sec.

In figure 5.4.11 the contribution of the model parameter covariances in the total expected optimal efficiency accuracy is given. The contribution of the model parameter covariances is roughly half of the total value. In this example, the influence of the parameter inaccuracy and the total steam consumption inaccuracy to the expected optimal total efficiency inaccuracy is approximately equal. For that reason, both factors will be considered. The contribution of a variable to the variation of the objective function is function of two factors : the accuracy of the variable, *and* the sensitivity of the optimal value function for the considered variable. The steam demand is measured much more accurate than the model parameters. But, the sensitivity of the optimal value function for the steam demand is relatively larger than for the model parameters. Therefore both contribute comparable to the variance of the optimal value function. Values for the first 5 minutes are not given because they are not calculated until five minutes have passed. Figure 5.4.12 gives an approximation of the 95

Network Load	Start Time	Stop Time
28.7 lb/h	00h00m	00h20m
29.9 lb/h	00h20m	00h33m
31.1 lb/h	00h33m	00h50m
32.3 lb/h	00h50m	01h07m
33.5 lb/h	01h07m	01h23m
35.9 lb/h	01h23m	01h40m
38.2 lb/h	01h40m	01h57m
39.4 lb/h	01h57m	02h13m
40.2 lb/h	02h13m	02h30m

Table 5.4.4.: Boiler Network Load Versus Time

% confidence values of the expected optimal efficiency for the same simulation as figures 5.4.10–11. In this example, the confidence level remains rather narrow. The small width of the confidence interval indicates sufficient confidence in the optimization result for implementation. (More details in Appendix G (Run # 14).) Another example is shown in figure 5.4.13. In that example the optimization is run every minute. The confidence results are printed around the expected optimal global network efficiency. The measured network efficiency is given as a dotted line. Five full lines are given. The middle line represents the expected network efficiency as given by the optimizer. The two outer lines give the expected optimal efficiency plus and minus the 95% confidence interval. The lines in between indicate the contribution of parameter estimates inaccuracy to that confidence interval. The remaining part is caused by the uncertainty in the total steam demand. The fraction of the confidence interval that is contributed

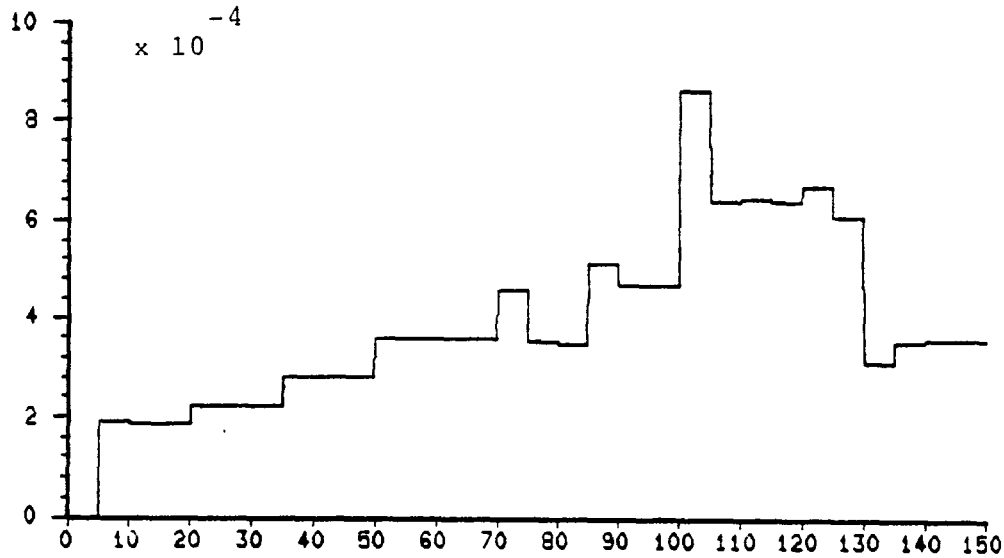

Fig.5.4.10. Variance of the Optimal Efficiency as a Function of Time for Increasing Boiler Network Loads with an Optimizer Execution Frequency of 1/5min. Time is given in minutes, and variance in (100 × useful BTU's / consumed BTU's)^2.

by parameter uncertainty varies. For instance, at time 128 the parameter estimates improved significantly due to new data collected. The implemented allocation did not change significantly but the confidence in the optimization result increased. The steady state value of measured optimal efficiency is in all cases within the confidence limits.

The size of the confidence interval of the expected optimal network efficiency can be compared to a preset value. If the confidence interval width exceeds this preset value, implementation of the results has to be interrupted since too much uncertainty is connected with it. (More details in Appendix G (Run # 15).)

5.4.2.2.2. On-Line Accuracy of the Expected Optimal Setpoints

In the previous section, an application of the on-line calculation of the accuracy of the objective was presented. In this section, an application of the

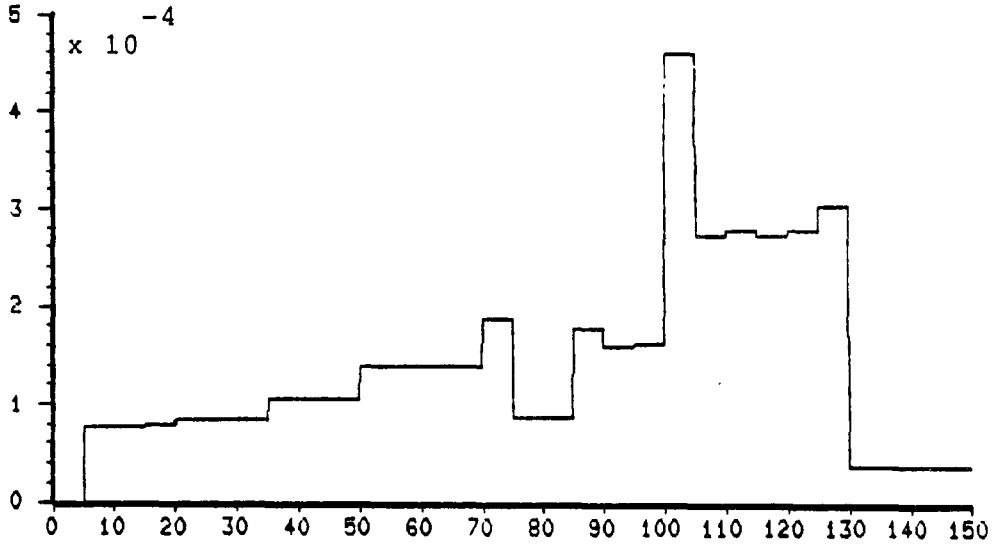

Fig.5.4.11. Contribution of the Model Parameter Covariances to the Variance of the Optimal Efficiency as a Function of Time for Increasing Boiler Network Loads with an Optimizer Execution Frequency of 1/5min. Time is given in minutes, and variance in (useful BTU's / consumed BTU's)^2.

on-line assessment of the accuracy of the setpoints that are calculated by the optimizer, is presented. The sensitivities of the setpoints are more difficult to obtain than the sensitivity of the objective. The theoretical derivation is given in section 3.6.2.2.2, resulting in equations 3.6.16–19. The sensitivities of the optimal setpoints are combined with the variances of the model parameter estimates and the measured total steam consumption, using equation (3.6.31) for a vector of setpoints :

$$Q_l = \left(\nabla_\varepsilon l^*|_0\right)^T Q_\varepsilon \left(\nabla_\varepsilon l^*|_0\right). \tag{3.6.31}$$

In this equation, l represents the vector of individual boiler load setpoints l_i, $\nabla_\varepsilon l^*|_0$ is the matrix of sensitivities of l with respect to ε and Q_l covariance matrix of l. As in the previous section, the vector of the model parameters and the boiler network load is given by ε, and Q_ε is the covariance matrix of the

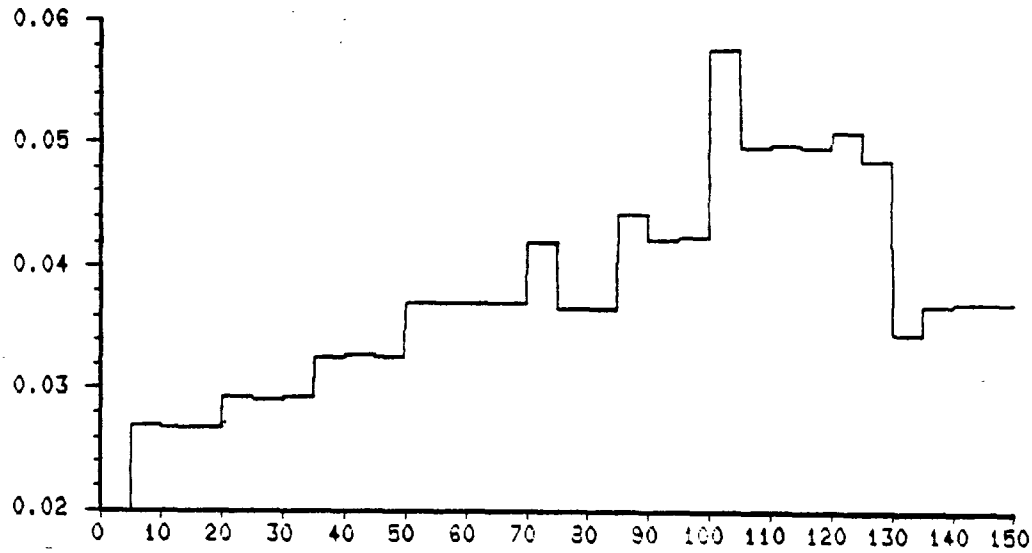

Fig.5.4.12. Confidence (95 %) of the Optimal Efficiency as a Function of Time for Increasing Boiler Network Loads with an Optimizer Execution Frequency of 1/5min. Time is given in minutes, and variance in (useful BTU's / consumed BTU's).

model parameters and the boiler network load. For three boilers, this means that Q_ε is a 10×10 matrix.

As an example, sensitivities are given for the three boilers that are the subject of this case study, for a total load of 72.81 lb/s. The following load allocations are optimal :

$$\begin{cases} l_1 = 25.47\ lb/sec \\ l_2 = 30.34\ lb/sec \\ l_3 = 17.00\ lb/sec. \end{cases} \tag{5.4.13}$$

It is important to note that the third boiler load is constrained by the minimum boiler load. In that case, the setpoint sensitivities can be found using equations 3.6.16–19. For boiler efficiency given as $\eta_i = a_i + b_i l_i + c_i l_i^2$ and with $\varepsilon_1 = a_1$, $\varepsilon_2 = b_1$, $\varepsilon_3 = c_1$, $\varepsilon_4 = a_2$, $\varepsilon_5 = b_2$, $\varepsilon_6 = c_2$, $\varepsilon_7 = a_3$, $\varepsilon_8 = b_3$, $\varepsilon_9 = c_3$ and $\varepsilon_{10} = l_{tot}$, the setpoint sensitivities can be found to be :

$$\nabla_\varepsilon l^* = \Big(A_1 \big| A_2\Big), \tag{5.4.14}$$

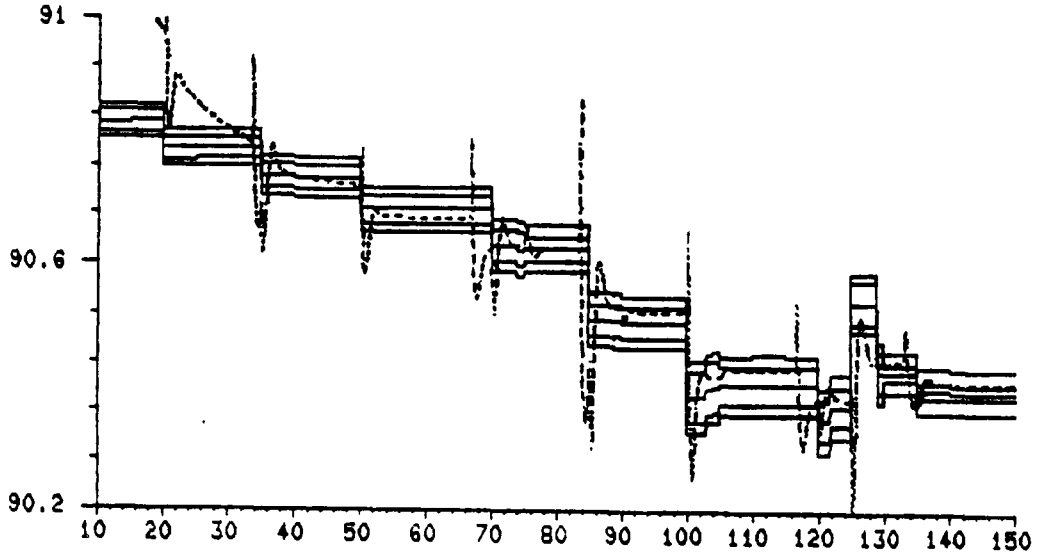

Fig.5.4.13. Confidence (95 %) of the Optimal Efficiency as a Function of Time for Increasing Boiler Network Loads with an Optimizer Execution Frequency of 1/1min. Time is given in minutes, and variance in (useful BTU's / consumed BTU's). Outer limits are the Confidence Interval. Inner Limits are the Contribution by Model Parameter Uncertainty. Center Line is Expected Optimal Efficiency.

with :

$$A_1 = \begin{pmatrix} 2.11 & 107. & 4108. & -2.11 & -128. & -5833. \\ -2.11 & -107. & -4108. & 2.11 & 128. & 5833. \\ 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}; \qquad (5.4.15)$$

$$A_2 = \begin{pmatrix} 0. & 0. & 0. & 0.6534 \\ 0. & 0. & 0. & 0.3465 \\ 0. & 0. & 0. & 0. \end{pmatrix}. \qquad (5.4.16)$$

The units for load are lb/sec, and efficiencies are given in %. The zeros in this matrix are caused by the active minimum load constraint. Since the minimum load constraint for the third boiler is active, it is clear that no infinitesimal small changes in any of the boiler model parameters or the total boiler network load will change the load to the third boiler. For the same reason, small deviations in the parameters of the third boiler will not change the load allocation for boiler 1 and 2. Note also that the derivatives $\nabla_{a_i} l_i^\star$, $\nabla_{b_i} l_i^\star$ and $\nabla_{c_i} l_i^\star$ for $i = 1, 2$ are

positive, while $\nabla_{a_i} l_j^\star$, $\nabla_{b_i} l_j^\star$ and $\nabla_{c_i} l_j^\star$ for $(i,j) = (1,2)$ and $(2,1)$ are negative. Furthermore, the sum of the derivative of the loads of boiler 1 and 2 with respect to the same coefficient are zero. The fact that for instance $\nabla_{a_1} l_1^\star$ is positive, indicates that if the boiler 1 efficiency goes up (because a_1 increases), the load allocated to boiler 1 increases. Similarly, the load for boiler 2 decreases by the same amount, since the total boiler network load did not change. For that reason $\nabla_{a_1} l_1^\star + \nabla_{a_1} l_2^\star = 0$. Consider the last column of the matrix given above. Because the equality constraint determining that the sum of the individual boiler loads has to equal the total steam demand, $\sum_i \nabla_{l_{tot}} l_i = 1$. This condition is met. Using (3.6.31), Q_l can be determined.

$$Q_l = \begin{pmatrix} 0.3493 & -0.1533 & 0. \\ -0.1533 & 0.2573 & 0. \\ 0. & 0. & 0. \end{pmatrix} \tag{5.4.17}$$

The rank of this 3×3 matrix is 2. This is so because there is one constraint active that does not contain any element from the ε matrix. This constraint is the low load inequality constraint. In the boiler example, the only constraints that don't contain any element from the ε matrix are the minimum and maximum load constraints. These constraints pertain only to one optimization variable each, i.e. the corresponding boiler load. Therefore an active minimum or maximum load constraint will always result in a corresponding row and column of zeros.

Once the covariance matrix of the optimal setpoint is obtained, the new setpoint has to be compared with the implemented setpoint. If a significant difference between the new and the implemented setpoint is detected using the multivariable method given in 3.6.3.4, then the new setpoint is implemented. If no significant difference is detected, then a possible way to adjust the implemented setpoint uses the following equation, which is implemented in this case :

$$l_{i+1} = \frac{n}{n+1} l_i + \frac{1}{n+1} l_i^\star. \tag{5.4.18}$$

In this expression l_{i+1} is the new setpoint, l_i is the previous setpoint, and l^*_{i+1} is the vector of optimization results. The number n is increased with one, every time no significant difference is detected between the new and the implemented setpoint. The number n is set to zero whenever a significant change is detected. In that way, no settling time has to be determined. This method can be used here, since no "inverse response" such as shown in the distillation case study, occurs here. So while initially the optimizer result are followed, the implemented results soon become virtually fixed. In the boiler example, the driving force is the steam demand. If the setpoints do not change significantly, then it can be expected that the steam demand did not change significantly either. Therefore, l_{i+1} as given in 5.4.18 will not violate any constraints significantly.

Some results are shown in figures 5.4.14–17. These three figures refer to the same conditions. The total steam demand was increased along the same schedule as above. White gaussian noise was added to the following measurements :

- ◇ the mass flow of steam exiting the superheater (σ = .01 (lb/sec)^2);
- ◇ the temperature of the steam exiting the superheater (σ=1. F^2);
- ◇ the mass flow of feedwater (σ=.01 (lb/sec)^2);
- ◇ the temperature of the feedwater (σ=1.F^2);
- ◇ the mass flow of attemporator water (σ=.0001(lb/sec) ^{2});
- ◇ the temperature of the attemporator water (σ=.0001 F^2);
- ◇ the mass flow of air (σ=.01 (lb/sec)^2);
- ◇ the temperature of the air (σ=1. F^2);
- ◇ the mass flow of fuel (σ=.01 (lb/sec)^2);
- ◇ the temperature of the fuel (σ=.01 F^2).

Added noise is the same for all three boilers. Figure 5.4.14 presents the distance between transformed optimal and transformed implemented setpoints as

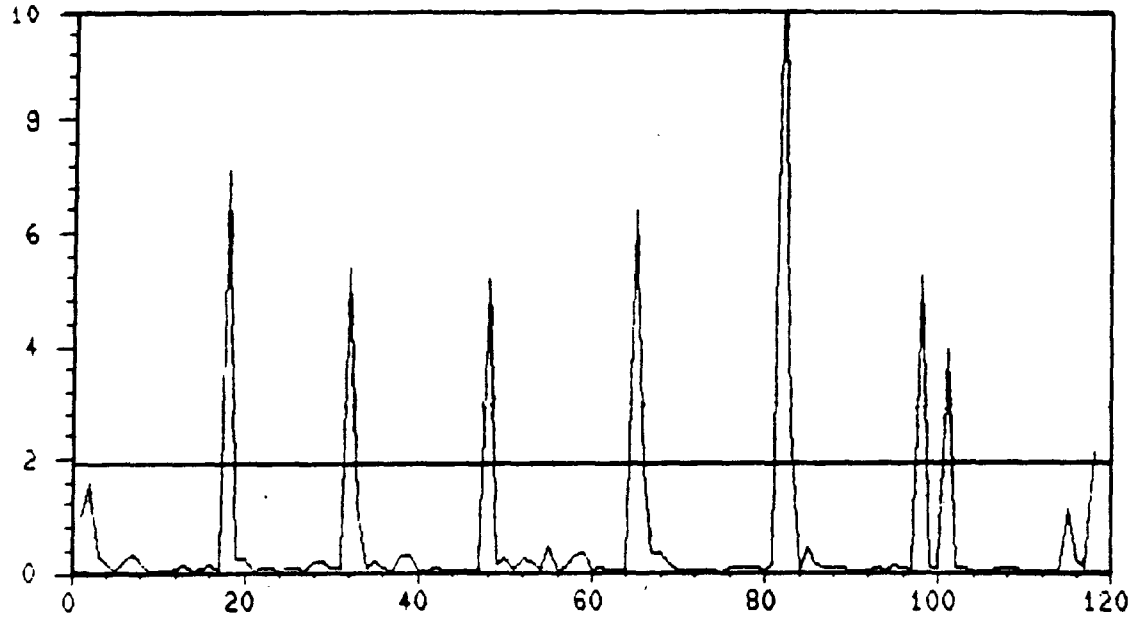

Fig.5.4.14. The Distance between Transformed Optimal and Transformed Implemented Setpoints as a Function of Time. The Full Line indicates a Distance of 1.96 Between Transformed Setpoints, the 95% Confidence Limit for Significant Different Setpoints. The Distance is Dimensionless, the Time is in Minutes.

a function of time. A solid straight line indicates a distance equal to 1.96, which is the 95% threshold for significant difference. Since every peak in the curve corresponds to a change in total steam demand load, the figure shows that the criterion is unambiguous in indicating significant setpoint changes. Therefore this method is not overly sensitive too *its only tuning variable*, i.e. the test significance. Between the big steam demand steps, the distance between the transformed optimal and transformed implemented setpoint usually remains below 0.5. At the load changes, the distance becomes large enough for more than 99.9% significance. Figure 5.4.15 shows the fuel setpoint for boiler 2 as a function of time for the first half of the run. Boiler 1 is a swing boiler, and therefore small changes are implemented there. Note that the fuel load setpoint for boiler 1 is determined by the pressure control and not directly by the optimizer. The dotted line represents the noisy optimal setpoint, the full line

gives the implemented setpoint. "Big" changes are implemented immediately, but small changes are neglected. The loop control has to deal with those fast and small disturbances. It is clear that the implemented setpoint becomes more and more constant as the time since the last significant setpoint change becomes larger.

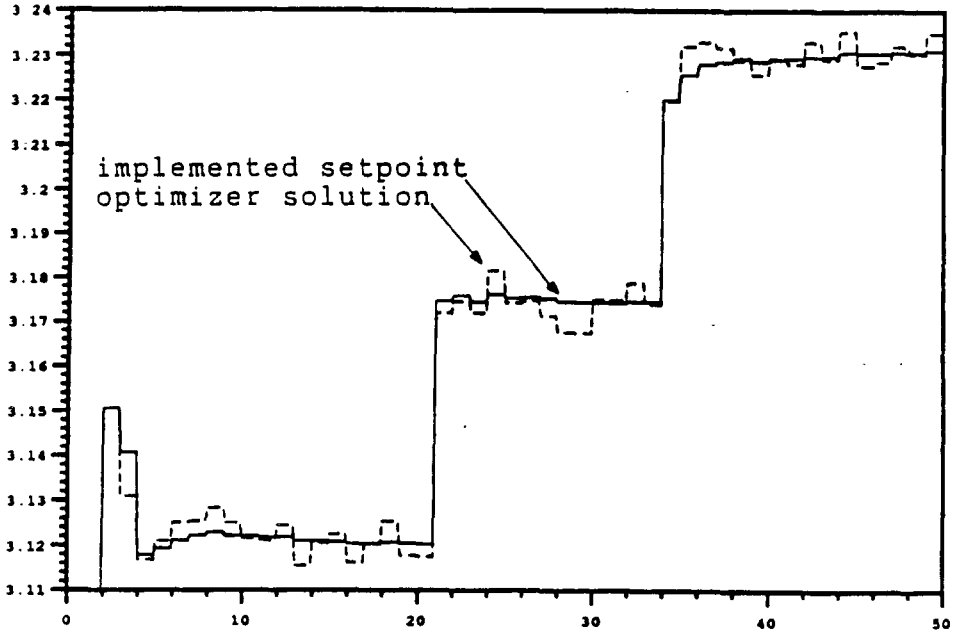

Fig.5.4.15. The Fuel Setpoint for Boiler 2 as a Function of Time for the first 50 Minutes of the Run. For the Remaining Time the Behavior was Similar. Fuel Load is in lb/sec, Time is in Minutes.

The direct advantage of the exclusive implementation of significant setpoint changes is that the operation becomes much quieter. As was indicated earlier, an on-line optimizer can actually amplify noise if no precautions are taken. Figure 5.4.16 shows the header pressure as a function of time. The dotted line gives the header pressure during operation without check for significant change in setpoints. The full line gives header pressures obtained while implementing only significant changes in setpoints. It is clear that the process operation is much quieter if only significant changes in fuel load setpoints are implemented.

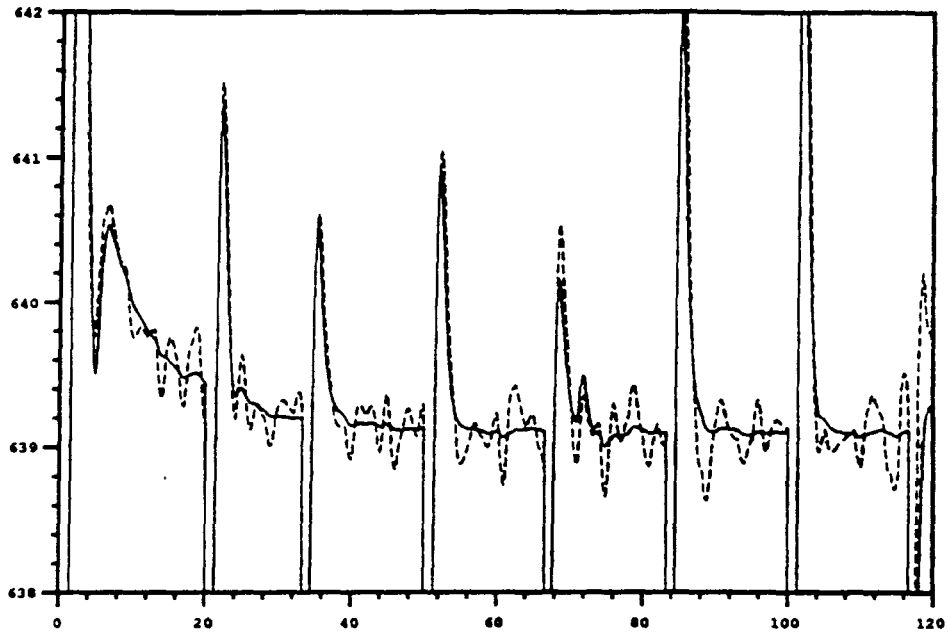

Fig.5.4.16. The Header Pressure as a Function of Time. The Dotted Line gives operation Without Check for Significant Changes in Setpoints. The Full Line gives Results in Case only Significant Changes in Setpoints are Implemented. Header Pressure is in psi, Time is in Minutes.

An indirect advantage of the exclusive implementation of significant setpoint changes is performance improvement. Often quiet behavior will result in better plant performance. In this boiler example, the average network efficiency did go up with the implementation of the significance check. However, the difference was smaller than the confidence interval of the overall efficiency, and therefore the increase in performance is insignificant in this example.

5.4.2.3. Off-Line Application of Sensitivity Analysis : Feasibility

In this section, off-line applications of sensitivity analysis in the boiler case study are illustrated. In Chapter 3 (3.6.3.1–2), these off-line applications are presented in detail. In this case study, only the *marginal optimization gain* (see 3.6.3.1) is studied.

Cho [1978] also studies boiler load allocation. This author considers four boilers with a common header. He uses the same second order polynomial model and optimization problem as is specified for this case study, as was indicated in Chapter 4 (section 4.3). Cho's [1978] publication is discussed in Chapter 2 (section 2.3.3). The coefficients that Cho uses to model the efficiencies of his boilers are given in Table 5.4.5. Note that Cho [1978] expresses steam load in lb/h, whereas in Table 5.4.3 loads were given in lb/sec, following Bertrand [1986]. Figure 5.4.17 gives the boiler efficiency curves versus individual boiler load.

#	a_i coefficient	b_i coefficient	c_i coefficient
Boiler 1	86.	$1.733 \cdot 10^{-5}$	$-3.618 \cdot 10^{-10}$
Boiler 2	85.	$9.187 \cdot 10^{-6}$	$-3.656 \cdot 10^{-10}$
Boiler 3	85.	$9.125 \cdot 10^{-5}$	$-8.14 \cdot 10^{-10}$
Boiler 4	84.	$9.5 \cdot 10^{-5}$	$-6.0 \cdot 10^{-10}$

Table 5.4.5 : Coefficients for Boiler Efficiency 2nd Order Polynomials from Cho [1978]

$(\eta_i = a_i + b_i l_i + c_i l_i^2)$

l = Boiler Load in lb/h, η = Boiler Efficiency in %

Cho [1978] considers a nominal total steam demand of 250,000 lb/h. The optimal individual boiler loads allocated are $l_1 = 51,698$ lb/h, $l_2 = 21,486$ lb/h, $l_3 = 75,722$ lb/h and $l_4 = 101,094$ lb/h. In this marginal optimization gain study, we will consider a suboptimal control policy as reference base line. This suboptimal control policy will keep the ratios of the individual boiler loads constant as the total steam demand is met. This control can be applied in a

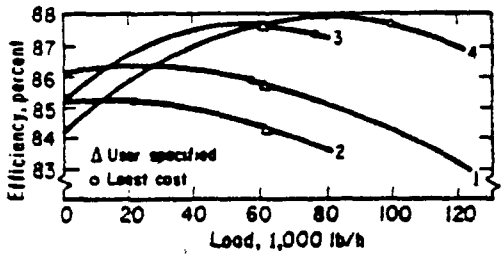

Table: Comparison of Load Allocation Methods

Boiler	Least cost	User specified
No. 1*	51,698.1	62,500
No. 2*	21,485.7	62,500
No. 3*	75,722.2	62,500
No. 4*	101,094.0	62,500
Totals	250,000.0	250,000
Cost**	575.802	579.983
Savings***	4.38135	

* Lb/h steam
** $/h
***1/h, (User specified)-(Least cost)

Fig.5.4.17. Boiler Efficiency versus Boiler Load for Boiler Discussed by Cho [1978]. Boiler Efficiency is in %, Boiler Load in lb/h.

structure such as the one given in figure 4.3.2. The ratio that is maintained is the ratio of the optimal load allocations for the nominal total steam demand. As can be calculated from the numbers above, the ratios $l_1 : l_2 : l_3 : l_4$ at the nominal total steam demand of 250,000 lb/h are 20.7 : 8.6 : 30.3 : 40.4. The suboptimal control policy, which corresponds to an off-line optimization at nominal load, will keep the ratios of individual boiler loads at the ratios specified above. This suboptimal policy is compared with the on-line optimization scheme that will optimize the load allocations after every change in total steam demand.

For this purpose, the optimization problem is reformulated. New optimization variables are introduced to replace the optimization variables used until now. The old optimization variables are the individual boiler loads l_i. The new optimization variables are the individual load fractions $\frac{l_i}{l_{tot}}$. Therefore, the suboptimal control policy after this introduction of new optimization variables, corresponds to *no optimization at all*, keeping the optimization variables (the load fractions) constant. The equality constraint that specifies that the total steam demand must be met ($\sum_i l_i = l_{tot}$) is substituted in the optimization problem by elimination of one of the new optimization variables, using

$\frac{l_N}{l_{tot}} = \left(1 - \sum_{i=1}^{N-1} \frac{l_i}{l_{tot}}\right)$. Since Cho [1978] does not introduce any minimum or maximum individual boiler load requirements, the remaining optimization problem seems to be unconstrained after substituting the equality constraint in the objective. This is not entirely true. Load fractions have to be between 0 and 1. In the total steam demand ranges that are considered here, these requirements never become active. Therefore, they are not included in the new optimization problem. It is also accepted that there are no minimal and maximal load change constraints.

Using the methods given in 3.6.3.1, the "marginal optimization gain" can be calculated. This is half of the absolute value of the second order derivative of the difference in the performance objective between the on-line optimized and the unoptimized case (in this case the outlined suboptimal control policy that keeps the individual boiler loads constant) with respect to the total steam demand at the nominal total steam demand (here 250,000 lb/h). The first order derivative is zero, since the rewritten optimization problem is unconstrained (see 3.6.3.1 for proof). The result is :

$$\frac{1}{2}\frac{d^2\Delta\eta^\star}{dl_{tot}^2}\Big|_{l_{tot}=250{,}000 lb/h} = 6.43 \cdot 10^{-11} \quad \% \cdot \left(\frac{hr}{lb}\right)^2. \tag{5.4.19}$$

Consider a load increase of 25,000 lb/h, which is 10% of the nominal total steam demand. In that case, the on-line optimized boiler arrangement would operate 0.04% more efficient than the suboptimally operated boiler cluster with constant ratios of the individual boiler loads. Using data of Cho [1978], this would result in 2200 $ savings per year in 1978. This profit is a best case estimate, since it was accepted that the model is perfect, and that no noise or losses because of transient phenomena occur. Clearly, 2200$ per year does not justify the use of an on-line optimizer. The suboptimal control policy should be used instead. One could even attempt to find simple suboptimal control that gives better results.

Cho [1978] compared the performance of his on-line optimizer at the nominal total steam demand with a load allocation of 62,500 lb/h for every boiler. If one looks at the figure 5.4.17, it is immediately clear that this must be a very poorly performing reference case. Cho [1978] reports that the scheme he proposes is very useful and efficient, and he predicts significant gains (up to $ 38,000 per year). The reference case is however almost arbitrary, and therefore his conclusions are too optimistic.

The procedure applied to Cho's [1978] boilers can also be applied to the boilers that are used in this boiler load allocation simulation study. The model parameters for the boilers used in this case study are given in table 5.4.3. For a total network load of 80 lb/sec, the optimal load allocation is the following : $l_1 = 27.19$ lb/sec, $l_2 = 30.04$ lb/sec and $l_3 = 22.81$ lb/sec. Also here the second derivative of the difference in efficiency between the on-line optimized group of boilers and the boilers operated suboptimally can be calculated. The result is :

$$\begin{aligned} \frac{1}{2}\frac{d^2\Delta\eta^*}{dl_{tot}^2}\Big|_{l_{tot}=80lb/sec} &= 2.18\cdot 10^{-4} \quad \% \cdot (\frac{sec}{lb})^2; \\ &= 1.68\cdot 10^{-11} \quad \% \cdot (\frac{hr}{lb})^2. \end{aligned} \tag{5.4.20}$$

The second value is given for comparison with the value found for Cho's [1978] boilers. This marginal optimization gain is only one quart of the marginal optimization gain found in Cho's [1978] example.

The boilers that are considered in this boiler load allocation study as well as the boiler network considered by Cho [1987] do not seem to meet the necessary condition of a sufficiently high marginal optimization gain at some nominal operating condition. The question arises whether or not for some other operating condition, the marginal optimization gain ($\frac{1}{2}\frac{d^2\Delta\eta^*}{dl_{tot}^2}$) becomes large enough to make the on-line optimization of the load allocation of a number of parallel boilers feeding a common header interesting. We therefore consider the three boilers for which the model parameters are listed in table 5.4.3. The range over

which the total steam demand can vary, is accepted to extend from 60 to 120 lb/s. For total steam demands below approximately 71 lb/sec and above approximately 118 lb/sec individual boiler load constraints become active (more details in Appendix G (Run # 16)). Under those conditions, one of the optimization variables, namely the load fraction corresponding to the constrained boiler, is removed from the set of optimization variables during the sensitivity analysis, and the analysis is completed under the assumption that this variable is merely a constant. Therefore, the marginal optimization gain will show a discontinuity at points where the active set changes. Figure 5.4.18 shows the marginal optimization gain as a function of total steam demand. It is clear that the on-line optimization is not an interesting alternative for any point in the operating range chosen as nominal total steam demand. A similar result was obtained using the boilers presented by Cho [1978], in figure 5.4.19 (see Appendix G (Run # 17). Also there it is visible when boiler load constraints become active.

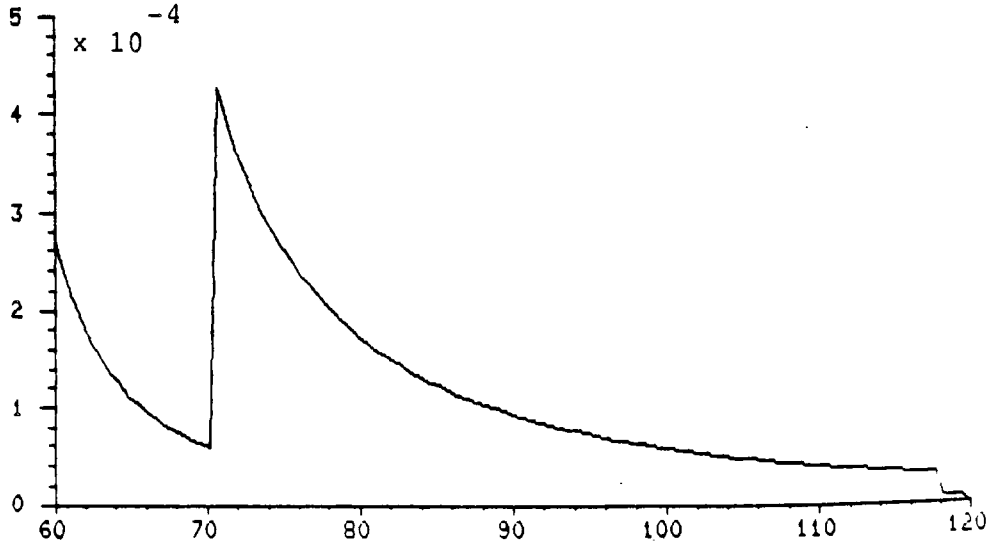

Fig.5.4.18. The Marginal Optimization Gain as a Function of Total Steam Demand. MOG is in % (sec/lb)^2, Steam Demand is in lb/sec.

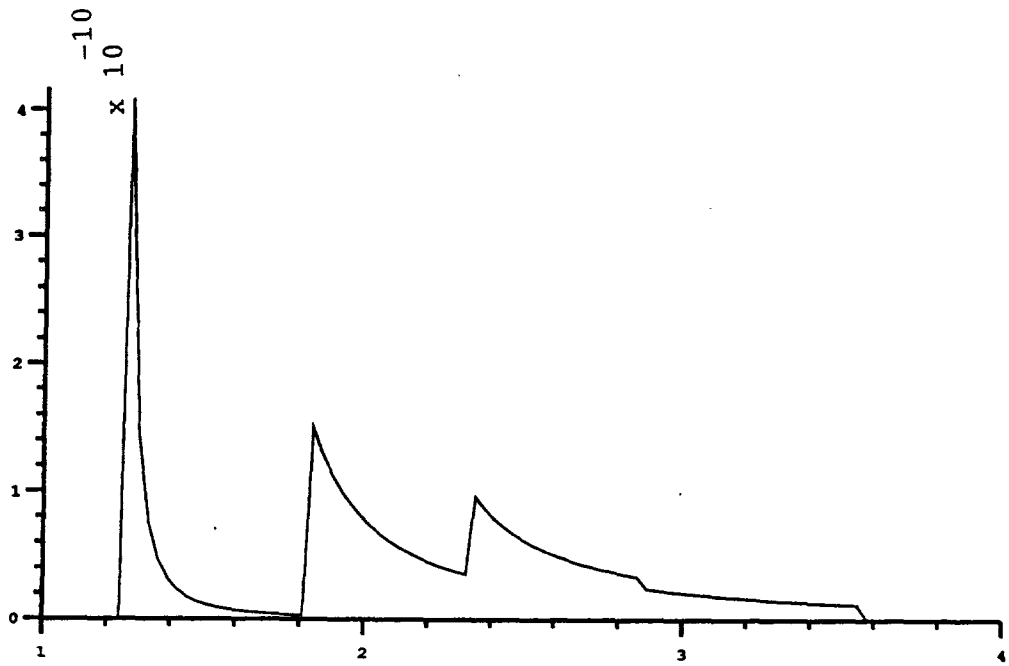

Fig.5.4.19. The Marginal Optimization Gain as a Function of Total Steam Demand. MOG is in % (h/lb)^2, Steam Demand is in lb/h.

The influence of the relative positions of the quadratic efficiency curves on the marginal optimization gain is illustrated next. First consider figure 5.4.20. In that figure the x-axis is the total steam demand, and the y-axis represents difference between the value of the a coefficient of the first boiler (1) and the value listed in table 5.4.3. In other words, $y = 0$ corresponds to a a_1 value that is equal to the value listed in 5.4.1 (88.20), while $y = 1$ corresponds to $a_1 = 89.20$. The z-axis corresponds to the number of degrees of freedom in the boiler load allocation optimization problem. Since the three boiler loads always have to add up to the total steam demand, the largest number of degrees of freedom is two.

In figure 5.4.21, the x and the y-axis have the same meaning as in the previous figure. In this case, the marginal optimization gain is given in the z direction.

Note that the marginal optimization gain is zero if the degrees of freedom are zero. This is the trivial case : the optimization problem has no more variables, and hence no gain can be found in on-line optimization. As the deviation

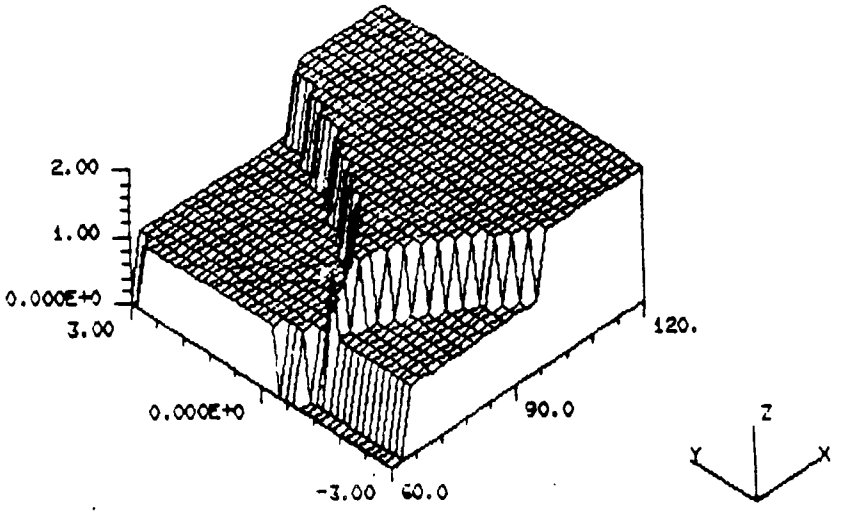

Fig.5.4.20. Number of Degrees of Freedom in the Boiler Load Allocation Problem. The x-axis is the Total Steam Demand in lb/sec, the y-axis is the Deviation of the a_1 Value from the Value listed in Table 5.4.3 in % , and the z Value corresponds to the Degrees of Freedom.

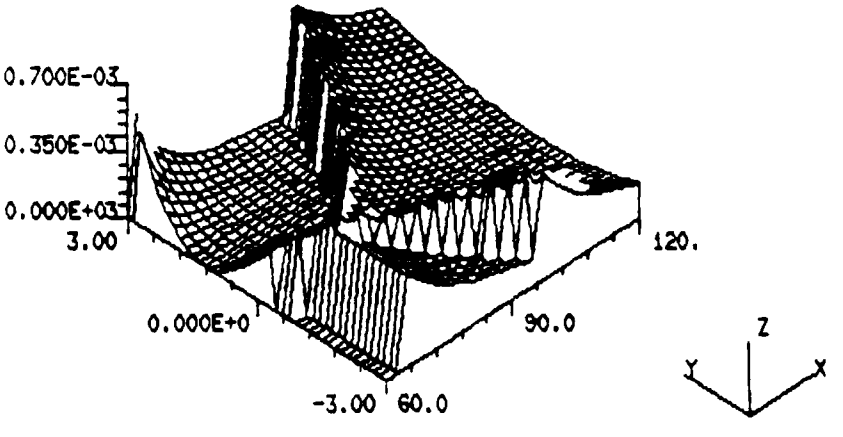

Fig.5.4.21. Marginal Optimization Gain in the Boiler Load Allocation Problem. The x-axis is the Total Steam Demand in lb/sec, the y-axis is the Deviation of the a_1 Value from the Value listed in Table 5.4.3 in % , and the z Value corresponds to the Marginal Optimization Gain, and is given in % $(sec/lb)^2$.

in a_1 becomes larger and larger, the marginal optimization gain increases. Even at the ends of the plot, the marginal optimization does not exceed $7 \cdot 10^{-4}$ %

$(\text{sec/lb})^2$. That value is still insufficient to justify the installation of an on-line optimizer. A marginal optimization gain of $7 \cdot 10^{-4}$ % $(\text{sec/lb})^2$ would correspond to *at most* \$ 3,000 per year savings for a continuous deviation of 10 lb/sec in the total steam demand, based on numbers published by Cho [1978]. As the deviation in a_1 becomes larger, the range over which two degrees of freedom are available becomes smaller. The marginal optimization gain is smaller as the degrees of freedom shrinks.

An analytical study of the marginal boiler optimization gain aimed at finding conditions for an interesting candidate for on-line optimization is theoretically possible, but soon the results become so lengthy and complicated, that it is virtually impossible to analyse the characteristics of the result. Only the case for two boilers is manageable. Some results are presented here. Consider therefore two boilers. Three different suboptimal control policies will be considered here, and therefore three different marginal optimization gains (with different basis) will be found. The first suboptimal control policy keeps the ratio of the two boiler loads constant at the optimal ratio at nominal steam demand. The second suboptimal control policy keeps the load for boiler 1 constant at the optimal load for nominal steam demand. The third and last suboptimal control policy keeps the load for boiler 2 constant at the optimal load for nominal steam demand.

Consider first the case in which the suboptimal control policy keeps the ratio between the loads of both boilers constant. In that case, the objective that is to be minimized can be written as :

$$f = a_1\alpha + a_2(1-\alpha) + b_1 l_{tot}\alpha^2 + b_2 l_{tot}(1-\alpha)^2 + c_1 l_{tot}^2\alpha^3 + c_2 l_{tot}^2(1-\alpha)^3. \quad (5.4.21)$$

The marginal optimization gain (see section 3.6.3.1) can be calculated as :

$$\left| \frac{1}{2}\left(\frac{\partial^2 f}{\partial\alpha\partial l}\right)^2 \left(\frac{\partial^2 f}{\partial^2\alpha}\right)^{-1} \right| \quad (5.4.22).$$

In this case, the marginal optimization gain equals :

$$\mathrm{MOG}_1 = \frac{1}{2} \cdot \frac{\left(b_1\alpha^\star - b_2(1-\alpha^\star) + 3c_1 l\alpha^{\star 2} - 3c_2 l(1-\alpha^\star)^2\right)^2}{b_1 l + b_2 l + 3c_1 l^2\alpha^\star + 3c_2 l^2(1-\alpha^\star)}. \qquad (5.4.23)$$

Note that the marginal optimization gain seems to be independent of a_1 and a_2. That is not really true since $\alpha^\star$ is function of a_1 and a_2. The solution of the optimization problem, $\alpha^\star$, is a root between 0 and 1 of :

$$\frac{df}{d\alpha} = a_1 - a_2 + 2b_1 l\alpha - 2b_2(1-\alpha) + 3c_1 l^2\alpha^2 - 3c_2 l^2(1-\alpha)^2 = 0. \qquad (5.4.24)$$

Consider the second case, in which the load to boiler 1 is kept constant at the optimal value for a nominal steam load demand. In that case, the objective can be written as :

$$f = a_1 l_1 + a_2(l_{tot} - l_1) + b_1 l_1^2 + b_2(l_{tot} - l_1)^2 + c_1 l_1^3 + c_2(l_{tot} - l_1)^3. \qquad (5.4.25)$$

In that case, the marginal optimization gain can be written as :

$$\mathrm{MOG}_2 = \frac{1}{2} \cdot \frac{\left(2b_2 + 6c_2(l_{tot} - l_1^\star)\right)^2}{2b_1 + 2b_2 + 6c_1 l_1^\star + 6c_2(l_{tot} - l_1^\star)}. \qquad (5.4.26)$$

Here again, the a_1 and a_2 coefficients are of importance, since $l_1^\star$ is the solution of :

$$\frac{df_1}{dl_1} = a_1 - a_2 + 2b_1 l_1 - 2b_2(l_{tot} - l_1) + 3c_1 l_1^2 - 3c_2(l_{tot} - l_1)^2 = 0. \qquad (5.4.27)$$

In the third case, in which the load for boiler 2 is kept constant at it optimal value at a nominal total steam demand, the result is similar to the one given above, if every subscript 1 is replaced by subscript 2 and vice versa. That would result in MOG_3.

A *good* suboptimal control policy is a control policy for which the marginal optimization gain is small. Indeed, if the incentive for on-line optimization is small, then the suboptimal control policy under consideration can be applied

instead of on-line optimization without substantial loss. If for all possible combinations of a_i, b_i and c_i, a suboptimal control policy can be found that results in a small incentive for on-line optimization, then on-line optimization of parallel boilers feeding a common header is not interesting. Note that the curvature of the efficiency to load relationship is determined by the b and c coefficients, and not by the a coefficient. If the efficiency curve of boiler 2 is very "flat" (i.e. b_2 and c_2 are rather small) compared to the curve of boiler 1, then MOG_2 becomes small. Analogously, if the efficiency of boiler 1 is rather constant, MOG_3 will be small. This seems to indicate that a good suboptimal policy in case one of the boilers has a relatively constant efficiency, is to keep the load on the *other* boiler relatively constant, while adjusting for changes in the total steam demand using the boiler with relatively constant efficiency. This is intuitively clear. The boiler with more variable efficiency will be kept close to its highest efficiency, while moving the load of the other boiler around does not influence the combined efficiency very much. If both boilers have comparable b and c coefficients, then there are two possibilities. In the first possibility the efficiency versus individual load curves of both boilers are close. In that case, MOG_1 can be expected to be small since boiler loads will be comparable, and the terms in the numerator of the expression for MOG_1 for boiler 1 and 2 will compensate. If the efficiencies of both boilers are *not* comparable, then it is more likely that a minimum or maximum load constraint will become active, leaving the optimizer without degrees of freedom and hence no marginal gain in the on-line optimization. Therefore, if b and c coefficients are comparable, constant load ratios seem to be an interesting alternative to on-line optimization.

Finally, if the efficiency for neither boilers is particularly constant, but the b and c coefficients are not comparable, then one boiler must have a relatively small curvature (c), and a not too small b coefficient (otherwise the efficiency is

relatively constant). Consider as a special case, the case in which c_1 is zero, and therefore the efficiency curve of boiler 1 becomes a straight line. The efficiency curve of boiler 2 is not straight, and may or may not have a maximum in it. The optimum load allocation will correspond to loads for which the derivatives of the efficiency curves with respect to the individual boiler loads are equal. Therefore, at all times, the optimal load allocation for boiler 2 will be at the point for which the derivative of the efficiency curve of boiler 2 with respect to the individual boiler load for boiler 2 is equal to b_1, if such point exists between the minimum and maximum boiler 2 load constraint. Therefore, in general, if the efficiency of one boiler is almost a straight line, a good suboptimal policy will be to keep the load on the other boiler constant.

It seems as if for two boilers, a good suboptimal control policy can always be found if it is accepted that the relation between boiler load and boiler efficiency can be accurately represented by a second order relation. Simulation results with three and four boilers seem to suggest that this can be extended to more than two boilers.

Until now, this section on applications of the marginal optimization gain focused on changes in total steam demand as the main reason for adjusting the boiler load allocation. The results obtained indicate that on-line optimization of the boiler load allocation of parallel boilers feeding a common header is not sufficiently interesting compared to alternative simple control policies if changes in the total steam demand are the main disturbance. Another possibility is that not the total steam demand, but the individual boiler efficiency changes frequently. In that case, none of the suboptimal control policies presented above will perform satisfactorily since they are based on an off-line optimization at the nominal total steam demand. An on-line optimizer with on-line model identification could compensate for changes in the efficiencies in boilers. However,

it must be underlined that under such circumstances probably more profit can be made by properly maintaining the boiler, so its efficiency does not decrease substantially. Next an order of magnitude study is presented. The object of this study is to determine whether or not an on-line optimizer can be expected to improve the global system efficiency of a boiler network if the relationships between boiler efficiency load changes in time.

First consider a variable a coefficient. Constant boiler load ratios are taken as the alternative suboptimal control policy. In that case :

$$\begin{aligned} \mathrm{MOG}_{a_i} &= \frac{1}{2}\left|\left(\frac{\partial^2 f}{\partial\alpha\partial a_i}\right)^2\left(\frac{\partial^2 f}{\partial^2\alpha}\right)^{-1}\right| \\ &= \frac{1}{2}\left|\left(\frac{\partial^2 f}{\partial^2\alpha_i}\right)^{-1}\right| \\ &= \left|\frac{1}{2(b_i l + c_i l_{tot}^2\alpha_i)}\right|. \end{aligned} \tag{5.4.28}$$

In the following, let load be in lb/sec, and efficiency in %. Then, from table 5.4.3, it can be seen that the typical order of magnitude for b_i is -1, and for c_i is -2. The order of magnitude of l_{tot} is 2. Therefore, the order of magnitude of MOG_{a_i} is about -2. The gain by implementing an on-line optimizer if a change Δa_i can be expected in a_i is estimated as $\mathrm{MOG}_{a_i} \cdot (\Delta a_i)^2$. A large change in efficiency (and therefore in a_i) would be 5%. The order of magnitude of $\mathrm{MOG}_{a_i} \cdot (\Delta a_i)^2$ would then be around zero. This means that potentially up to 1% in overall efficiency could be gained. That is substantial. For example, MOG_{a_1} for the boilers listed in table 5.4.3, for a total steam demand of 80 lb/sec, equals $1.16 \cdot 10^{-2}$. For a 5% change in a_1, the difference between the suboptimal and the on-line optimized performance could be as high as 0.29%. This could for large boiler arrangements justify the implementation of the on-line optimizer.

In the same way, results can be obtained for the b and c coefficient. Order of magnitude of MOG_{b_i} can be expected to be around 2, and for MOG_{c_i} an

order of magnitude of 6 can be expected. Specifically, MOG_{b_1} for the boilers in table 5.4.3 with a steam load of 80 lb/sec equals 34.4, and MOG_{c_1} for the boilers in table 5.4.3 with a steam load of 80 lb/sec equals 230,000. A Δb_1 of 0.18 corresponds to a change of 5% in the efficiency in boiler 1 under those conditions. The incentive for optimization could be as high as 1% under those circumstances. A Δc_1 of $6.8 \cdot 10^{-3}$ corresponds to a change of 5% in the efficiency in boiler 1 under those conditions. The incentive for optimization could be as high as 10% under those circumstances. The last two results are definitely large enough to justify considering the application of an on-line optimizer. Note that the performance improvements are best case numbers that can be reduced by noise, plant model mismatch and transient effects. But are such large changes in the b and c coefficients realistic ? If instead of selecting Δb_1 and Δc_1 based upon 5% change in efficiency, they would be chosen as 5% of $|b_i|$ and $|c_i|$ respectively as listed in table 5.4.3, then the expected gain in on-line optimization becomes approximately 0.01% in both cases.

The results on the application of the marginal optimization gain, an off-line application of sensitivity analysis, indicate that parallel boilers feeding a common header are, in general, not interesting candidates for on-line optimization of the individual boiler load allocation. It seems that an easily applicable suboptimal control policy such as keeping ratios among boiler loads constant, can be found. The marginal performance increase on-line optimization offers with such a suboptimal policy as reference is in general too small to justify the application of an on-line optimizer. This is certainly true if one keeps in mind that boiler efficiencies are often rather flat curves of the individual boiler load, and that boiler efficiencies usually can only be obtained with an accuracy of at best 1% (Green and Al ai-Shaikh [1980]).

5.5. INFLUENCE OF DYNAMICS ON OPTIMIZATION

5.5.1. Introduction

In this section the interaction between the on-line optimizer (a discrete time system) and the plant (a continuous time system) is studied. First a frequency analysis shows how an on-line optimizer can change the frequency behavior of a plant. An important conclusion is that on-line optimizers can amplify certain frequencies in the input. Then a time average performance study gives insight into the effects of optimizer execution frequency. This is illustrated with a CSTR example, as well as with results from the boiler case study. Effects of noise on the optimizer performance are shown next. That is followed by some conclusions on the dynamic interaction between plant and on-line optimizer.

5.5.2. Frequency Analysis

5.5.2.1. Description

Consider a CSTR in which two consecutive reactions take place. The rate constant for both reactions is equal. The reactions are given by :

$$A \rightarrow B \rightarrow C. \tag{5.5.1}$$

The feed mass flow is constant. The feed is composed of the components A and B, and the composition varies. Product C is considered waste. The objective is maximal production of B. Since the feed mass flow is constant, the objective is equivalent to a maximization of the concentration of B in the product stream. It is assumed that the continuous process is time-invariant, and that the process is reflected exactly by the model. The temperature control is assumed to be immediate and perfect. The on-line optimizer adjusts the temperature period-

ically for changes in the feed composition. A steady state model which in this case does not need any updating, is used by the on-line optimizer. The Laplace domain model is given by :

$$X_B(s) = \frac{X_{Bf}(s) \cdot \theta^{-1}}{(s + \theta^{-1} + k)} + \frac{(\frac{1}{s} - X_{Bf}(s))k}{(s + \theta^{-1} + k)^2}. \tag{5.5.2}$$

In this equation, θ is the residence time, $X_B(s)$ is the Laplace transform of the mole fraction of B in the reaction mixture, $X_{Bf}(s)$ is the Laplace transform of the feed mole fraction of B and k is the reaction rate constant. The reactions are first order. Since the transients resulting from the initial conditions of the reactor are not of interest here, the (decaying) terms that are determined by the initial conditions are omitted here. The steady state optimum rate constant is given by

$$k_{opt} = \theta^{-1} \cdot (1 - 2x_{Bf}); \qquad \text{for } x_{Bf} < \frac{1}{2}. \tag{5.5.3}$$

Only feed stocks meeting the last condition are considered here. Appendix G (Run # 18) gives more detail on this example.

5.5.2.2. Gain and Phase Lag Characteristics

Let the superscript ss indicate steady state values. The linearization of equation 5.5.2 around $x_{Bf}^{ss} = 0.2$, $k^{ss} = \theta^{-1} \cdot 0.6$, $x_A^{ss} = \frac{1}{2}$ and $x_B^{ss} = \frac{5}{16}$ can be written with deviation variables as follows :

$$\begin{aligned} X_B(s) = & \underbrace{\left(\frac{\theta^{-1}}{s + \theta^{-1} + k^{ss}} - \frac{k^{ss}\theta^{-1}}{(s + \theta^{-1} + k^{ss})^2}\right)}_{g_1(s)} \cdot X_{Bf}(s) \\ & + \underbrace{\left(\frac{x_A^{ss} - x_B^{ss}}{s + \theta^{-1} + k^{ss}} - \frac{k^{ss}x_A^{ss}}{(s + \theta^{-1} + k^{ss})^2}\right)}_{g_2(s)} \cdot k(s). \end{aligned} \tag{5.5.4}$$

Let $x_{Bf}(t) = x_{Ba} \cdot \sin(\omega t + \varphi)$, again in deviation variables. The approximate values for the amplitude and the phase lag as a function of ω can be based upon

the following expression (Åström and Wittenmark, [1984]) :

$$\begin{cases} g_1(j\omega) + g_2(j\omega)\frac{\omega_s}{\pi}H(j\omega)F(j\omega), & \omega \neq k\frac{\omega_s}{2}; \\ g_1(j\omega) + g_2(j\omega)\frac{\omega_s}{\pi}H(j\omega)F(j\omega)e^{j(\frac{\pi}{2}-\varphi)}\sin\varphi, & \omega = k\frac{\omega_s}{2}; \end{cases} \tag{5.5.5}$$

in which ω_s is the sampling frequency, k is an integer, $H(s) = -2\theta^{-1}$ and $F(s) = \frac{1}{s}(1-e^{\frac{-2\pi s}{\omega_s}})$. The on-line optimizer is executed at the sampling frequency ω_s. In equation 5.5.5, g_1 and g_2 is specified in 5.5.4.

Figures 5.5.1–4 summarize the results of the frequency analysis. The frequency in the axis of the figures is dimensionless. It is the frequency ω multiplied with the residence time θ. Figures 5.5.1–2 apply to the plant without an on-line optimizer. Figure 5.5.1 represents the plant gain, and figure 5.5.2 represents the phase lag. These figures indicate no unusual frequency behavior of the plant without an optimizer. Figures 5.5.3–4 are equivalent to figures 5.5.1–2 *with* an on-line optimizer. A bold black vertical line in both figures 5.5.3 and 5.5.4 indicates the sampling frequency. In the neighborhood of the sampling frequency, with accent on the frequencies above the sampling frequency, the gain increased significantly with the use of an on-line optimizer. At higher frequencies the gains of the optimized and the not optimized plant are approximately equal. The resolution of the plots does not allow seeing every detail clearly. At the sidebands $(k\omega_s + \omega)$ the frequency response becomes dependent on the synchronization of the optimizer execution frequency and the input signal oscillations. There the frequency response curve becomes discontinuous, but the discontinuities are too small to be visible here. The on-line optimizer causes a positive phase lag φ at low frequencies. Furthermore, the phase lag is not a monotonous function any more of the frequency. It should be stressed that no low pass filters were applied here to the input to the on-line optimizer. Therefore aliasing occurs. Aliasing and the synchronization effect of the sidebands cause nonmonotonous behavior.

Note that an on-line optimizer does not necessarily decrease the gain. One

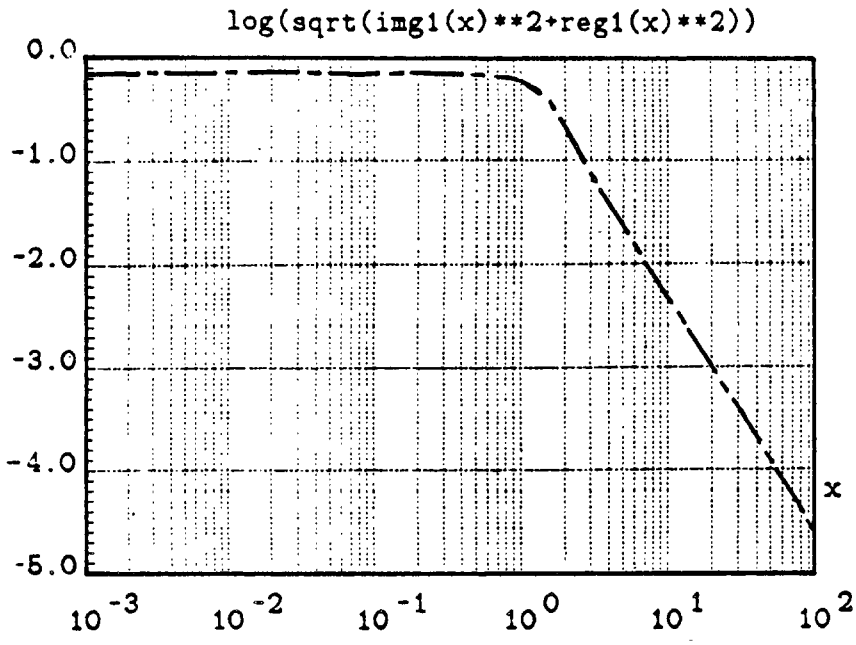

Fig.5.5.1. System Gain without On-line Optimizer

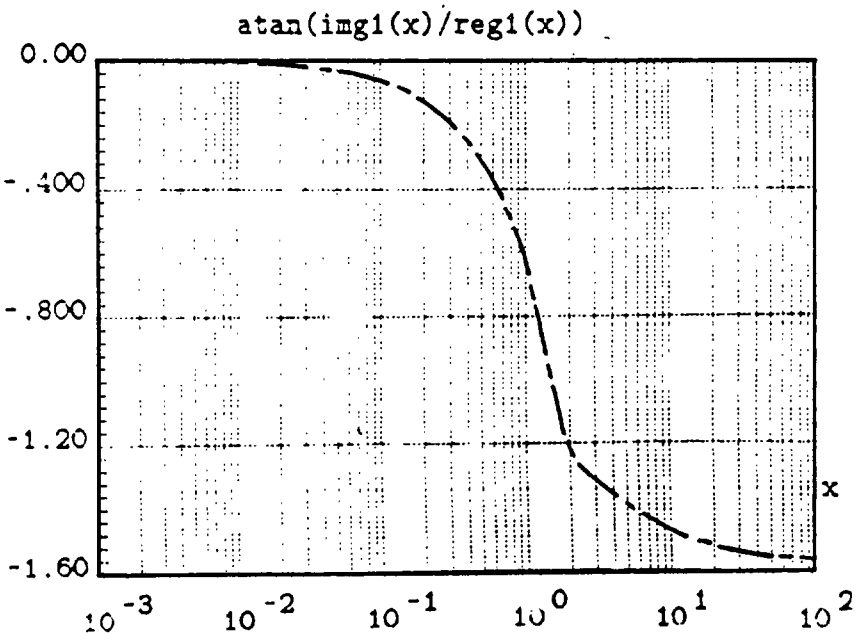

Fig.5.5.2. System Phase Lag without On-line Optimizer

might be tempted to think that the on-line optimizer adjusts the control settings to maintain "the" optimally performing operation. However, this optimally performing operation is function of the plant inputs, such as the feed composition.

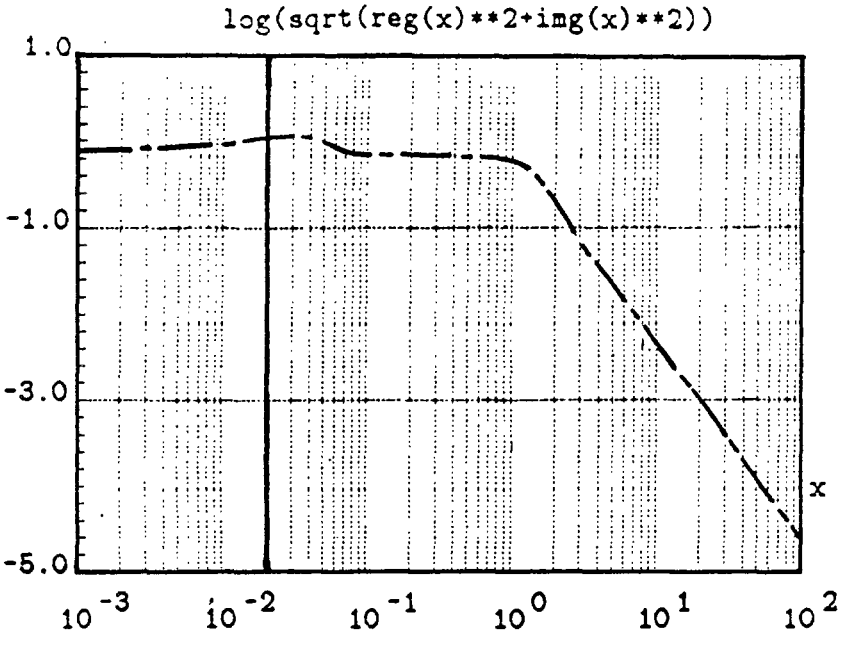

Fig.5.5.3. System Gain with On-line Optimizer

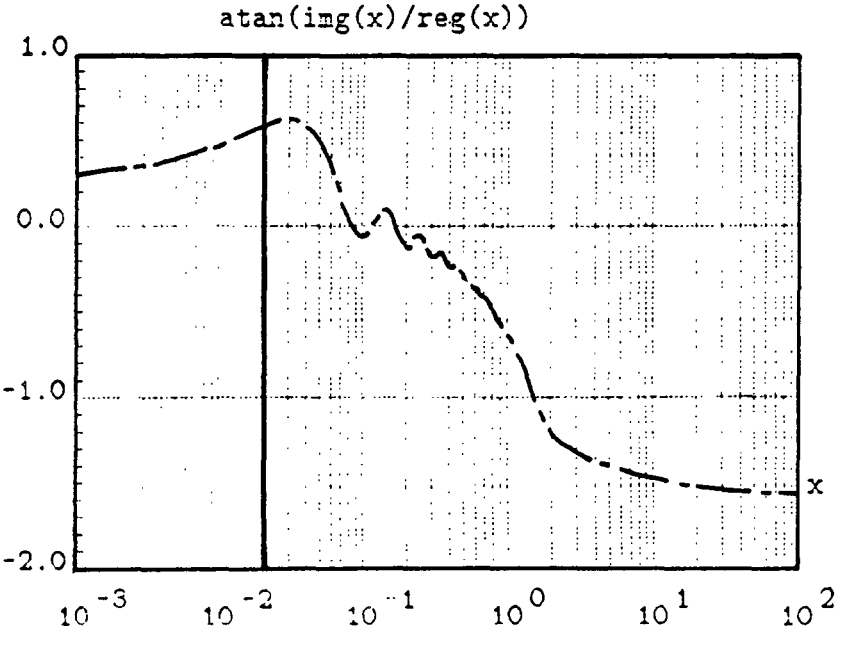

Fig.5.5.4. System Phase Lag with On-line Optimizer

Oscillations in the input are not necessarily damped. In this example the gain *increases.*

5.5.3. Time Average Frequency Analysis

5.5.3.1. Introduction

One can expect that the ratio of the frequency of the input deviations and the execution frequency influences the time average performance of the on-line optimized system.

Consider a right continuous square wave as input for the process. Assume the optimizer (in particular the steady state plant model) to be perfect, assume also that model updating is unnecessary,and that the implementation of the optimization result (reactor mixture temperature) is implemented perfectly (without offset) and immediately. The plant is time invariant. The square wave presents only two distinct inputs to the on-line optimization. Let v_1 and v_2 be the two input values, and let $x_1^\star$ and $x_2^\star$ be the two corresponding optimal setpoints. All together four possibilities occur. These are the following combinations ; $x_1^\star$ combined with v_1, $x_1^\star$ combined with v_2, $x_2^\star$ combined with v_1, and finally $x_2^\star$ combined with v_2. Some definitions are introduced here. Let the plant be described by $y(t) = L\{x, v, w_0\}(t)$. L is an operator on the plant input x, the setpoints v, and w_0. The initial conditions are given by w_0. Let $c(x, y, v)$ be the performance criterion, which has to be minimized. Let $\hat{c}(t) = c\big(x(t), L\{x, v, w_0\}(t), v(t)\big)$. The time average of $\hat{c}$ is given by the following expression :

$$\bar{\hat{c}}(t) = \lim_{t\to\infty} \Big(\frac{1}{t}\int_0^t \hat{c}(t)dt\Big). \tag{5.5.6}$$

Finally, let N_v be the number of time units per half period in the input signal, and let N_s be the number of time units between two optimizer runs.

5.5.3.1.1. Case 1 : $N_v = \infty$

In this case, the frequency of the input oscillations is zero. The on-line

optimizer performs ideally, since the period between $t = 0$ and the first on-line optimization can be neglected. The time average performance of course depends upon of the value of the plant input, be it v_1 or v_2. Since this case will be the reference case, the average of performances with input v_1 and v_2 is taken :

$$\bar{\hat{c}} = \frac{1}{2} \cdot \big(c(x_1, y_1(\infty), v_1) + c(x_2, y_2(\infty), v_2)\big). \tag{5.5.7}$$

5.5.3.1.2. Case 2 : $N_v >> N_s$

It is assumed that for a large difference between N_v and N_s the transients are unimportant and can be neglected. Furthermore, it is assumed that there is no integer k for which $kN_s = N_v$. Then the probability that the optimizer is executed between time units N and $N + dN$ after the step change in v is dN/N_s. Integrating from 0 to N_s time units gives :

$$\begin{aligned} \bar{\hat{c}} =& \frac{N_s^2}{4N_vN_s}\Big(c\big(x_2, y_{21}(\infty), v_1\big) + c\big(x_1, y_{12}(\infty), v_2\big)\Big) \\ &+ \frac{2N_vN_s - N_s^2}{4N_vN_s}\Big(c\big(x_1, y_{11}(\infty), v_1\big) + c\big(x_2, y_{22}(\infty), v_2\big).\Big) \end{aligned} \tag{5.5.8}$$

In this equation, $y_{ij}(\infty) = L\{x_i, v_j, w_0\}(\infty)$. As N_s increases, $\bar{\hat{c}}$ will become larger, and hence the performance will decrease. Notice that as N_v/N_s increases, the influence of the first term disappears. In the limit $\frac{N_s}{N_v} \to \infty$, the results of the previous case are found.

What happens around the points that were excluded, namely the points for which $kN_s = N_v$ for an integer k ? The value at these points depends upon the synchronization of the input oscillation and the optimizer execution frequency. Nothing in expression 5.5.8 indicates that the limit from the left at these points would be any different from the limit at the right from the excluded points. However, at a frequency close to but higher than the excluded points (sidebands), the average length of the time intervals during which no changes

in setpoints and input occur, is shorter then at a frequency close but below the sidebands. This is a result of interference close to the sidebands.

At the beginning of the description of this Case 2, it was assumed that transients were considered unimportant. Let us replace that assumption now by the assumption that the performance during transients is always poorer than during steady state. Then frequent steps would results in poorer results. The values of $\bar{\hat{c}}$ will jump at a sideband if this assumption is accepted. These discontinuities are not important in this discussion. The general trend of the performance as a function of the ratio of the frequencies of execution and oscillation of the input is of importance here.

5.5.3.1.3. Case 3 : $N_s >> N_v$

In this case a large number of step changes in the input takes place between every optimization run. Assume that at the moment of the optimization the input consisted of v_1. Therefore x_1 was chosen as optimal setpoint. This will be maintained for at least N_s time units. In the mean time a number of step changes take place in the input. If we maintain the assumption that transients perform poorer as steady states, then as N_v decreases, the steady state is less and less well approximated. Therefore, it can be expected that the performance decreases with N_v for a constant N_s.

5.5.3.2. The Lost Profit Curve

The conclusions of this discussion are presented schematically in figures 5.5.5–6. In figure 5.5.5, the performance of the on-line optimized plant is compared to the performance of the plant without an on-line optimizer. The latter serves as reference. It is clear that as $N_s \rightarrow \infty$ both curves will come together. The shape of the reference curve is difficult to predict. If the system does not contain discontinuities then the reference curve can be expected continuous. If

it is assumed that the system performs poorer during transients than under steady operation, then one can expect that the reference curve will decrease as N_v/N_s increases, as is shown in figure 5.5.5. Both curves are subtracted, and the result is schematically presented in figure 5.5.6. It is expressed as the *lost profit* due to implementing an on-line optimizer. This presentation is chosen because it resembles a Bode gain plot with resonance peak.

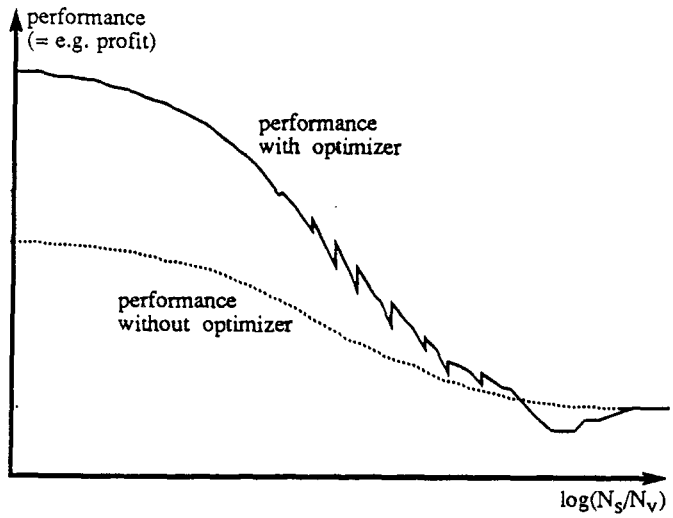

Fig.5.5.5. Sketch of the System Performance with and without On-line Optimization as a Function of the Ratio of the Frequencies of the Optimizer Execution and the Input Oscillations

Let N_s be a fixed number. Then the frequency of execution of the on-line optimizer is fixed. What spectrum of input oscillations can be dealt with effectively ? The *lost profit* curve is divided in four parts, shown in figure 5.5.7. If only very slow oscillations in the system inputs (deep in part A) are present, then they can be handled off-line. Frequencies in the range B are the most appropriate frequencies for input to an on-line optimizer. Frequencies in range C are resonance frequencies. The point $\omega_s = \omega_v$ lies somewhere in this range. It is possible, but not necessary that the optimized system does worse in this region of frequencies than a not optimized system, as shown in figure 5.5.7. Signals in this range should be filtered out. The loop control has to correct

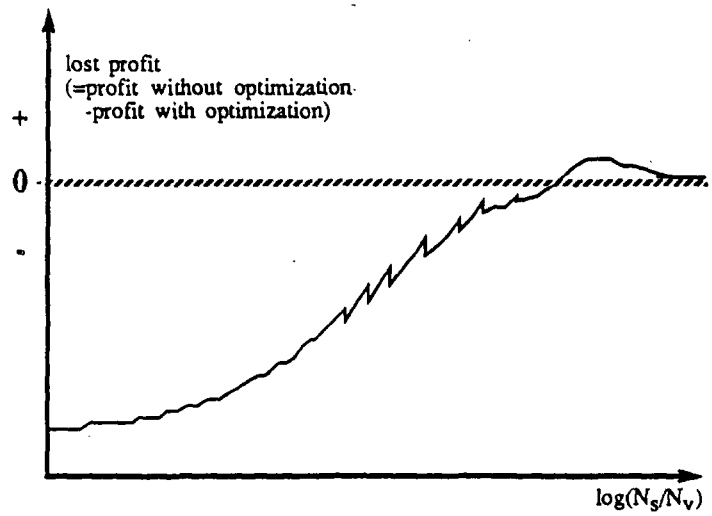

Fig.5.5.6. Sketch of the System Lost Profit by Using an On-line Optimizer as Compared to not Optimizing in Real Time as a Function of the Ratio of the Frequencies of the Optimizer Execution and the Input Oscillations

for these input frequencies. The on-line optimizer is statistically ineffective at frequencies in the D range. That means that on the average the performance of an on-line optimized system is not expected to be better than a system without on-line optimizer. However, the performance for one specific case is unpredictable. Therefore also these frequencies have to be filtered out.

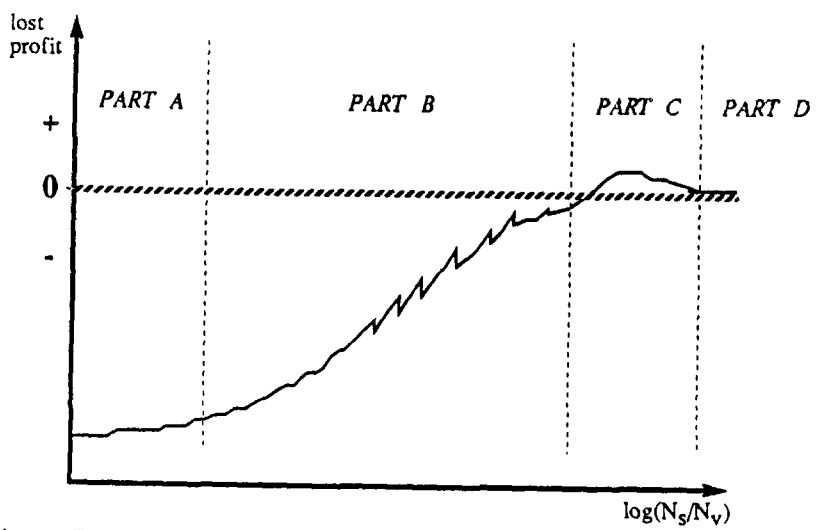

Fig.5.5.7. Partitioning of Lost Profit by Using an On-line Optimizer as Compared to not Optimizing in Real Time Based upon the Ratio of the Frequencies of the Optimizer Execution and the Input Oscillations

Let N_v be a fixed number. In other words, the frequency spectrum of the input signal is known. At what rate should the optimizer be executed ? *If* there is no interaction with the plant dynamics, the optimizer should be run at as high as possible a rate, as long as the cost for the optimization does not increase sharply.

The plant dynamics have been neglected so far. It was only assumed that transient performance was inferior to steady state performance. The interaction between plant dynamics and optimizer is a complicated one. Plant dynamics influence the steady state data collecting. If the plant setpoints are updated frequently compared to the plant dynamics, even over small amounts, the steady state data collection may be jeopardized. Implementation of a significance check as illustrated in section 5.4 reduces this interaction. The following paragraph explains this point.

Let the plant have a number of time constants, the largest of which is ω_p^{-1}. If the optimizer input contains only frequencies significantly below ω_p, then the on-line optimizer can be executed at a high rate. The output will only change when a statistically significant change in the optimization results is found. This can only happen if the inputs change, which is at a low frequency. The frequently running optimizer, not changing the setpoints at every execution, can be considered *stand-by* until it detects a change in its output that is significant.

If the optimizer input does not only contain frequencies significantly below ω_p, then the higher frequencies have to filtered out. The loop control should be able to maintain a practical steady state. If not, steady state optimization should not be applied. The plant setpoints will only be updated if the optimizer results change significantly. Therefore interaction with the plant dynamics is virtually impossible, and the on-line optimizer can be run at a rate that is significantly faster than the top input frequencies. The optimizer execution

frequency can be smaller than the plant frequencies with a check for significance in the setpoint changes. For instance, the optimizer in the boiler case study was run as often as every 12 seconds.

5.5.3.3. Example 1 : CSTR Reactor

Consider the CSTR example that was used in the frequency analysis in section 5.5.2. Let the input to the system be a square wave with variable frequency in the feed composition with $v_1 = \frac{x_{Af}}{x_{BF}}|_1 = \frac{1}{0}$ and $v_2 = \frac{x_{Af}}{x_{BF}}|_2 = \frac{0.6}{0.4}$. Results are shown in figures 5.5.8–9.

The curve in figure 5.5.8 shows the lost profit for the optimizer defined in equation 5.5.3 :

$$k_{opt} = \theta^{-1} \cdot (1 - 2x_{Bf}). \tag{5.5.3}$$

The optimizer execution frequency is indicated by a bold black vertical line. Figure 5.5.9 shows lost profit curves for a corrupted optimizer. The performance of the optimized plant is *lower* than the unoptimized plant for high input frequencies (= PART D). This example clearly shows a resonance peak in the area for which disturbance frequency and optimizer execution frequency are close (in order of magnitude).

5.5.3.4. Example 2 : Boiler Case Study

A second example is provided by the boiler case study. The steam demand posed to the boiler network is increased stepwise by 1.2 lb/sec each step. The initial steam demand was approximately 29 lb/sec, and the final demand 41 lb/sec. This stepping is done at three different rates. A step is taken after every 5 minutes (12 h^{-1}) at the high frequency, after every 10 minutes (6 h^{-1}) at the middle frequency, and after every 30 minutes (2 h^{-1}) at the low frequency. For every rate of steam demand increase, the optimizer is run at six different execution frequencies. Appendix G (Run # 19) gives more details on this

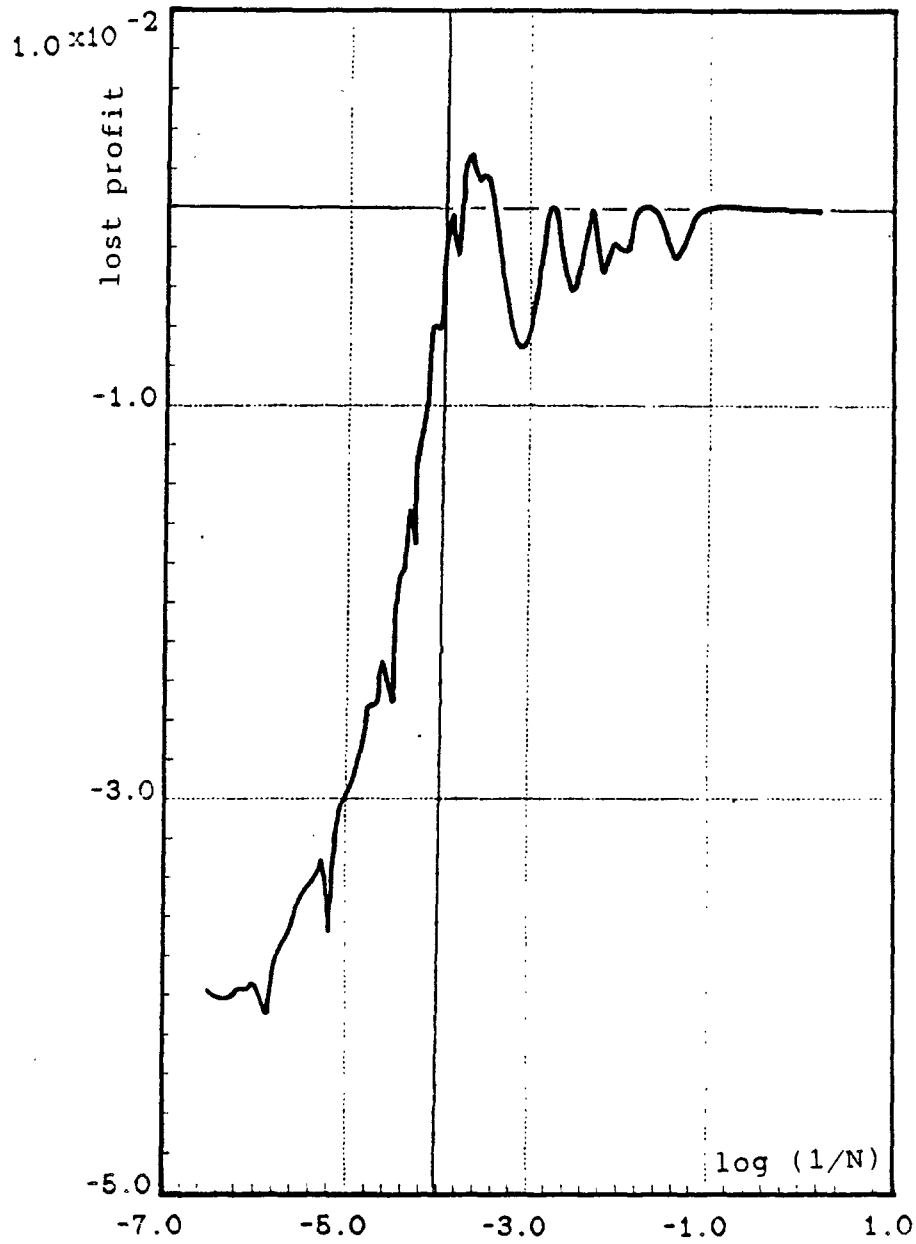

Fig.5.5.8. Lost Profit Curve for a CSTR with Consecutive Reactions with an Ideal Optimizer

examples. Results are presented in figures 5.5.11. Of course these curves are not as detailed as in the lost profit curves in the previous example. The CPU demand for every simulation of this network of three boilers is considerable, as can be seen on figure 5.5.10. Approximately 0.63 seconds CPU time are consumed on a MicroVAX under VMS for every run of the on-line optimizer.

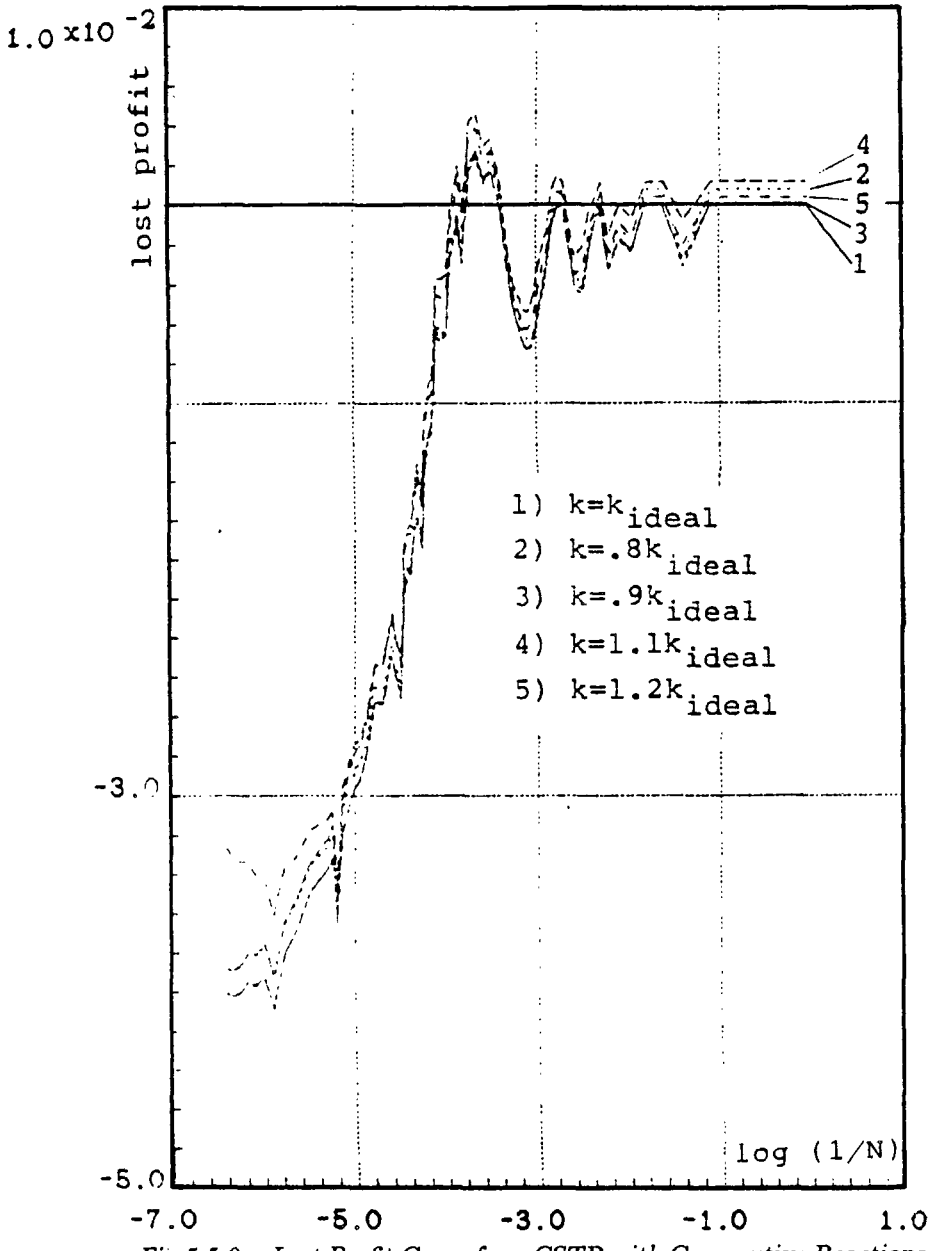

Fig.5.5.9. Lost Profit Curve for a CSTR with Consecutive Reactions with a Corrupted Optimizers

This includes optimization, sensitivity analysis and accuracy checking.

Figure 5.5.11 shows the loss of average plant performance as a function of optimizer execution frequency. The three curves represent different frequencies in the steam demand steps. If the optimization execution frequency is slowed down, more performance is lost for increasing steam demand stepping rate.

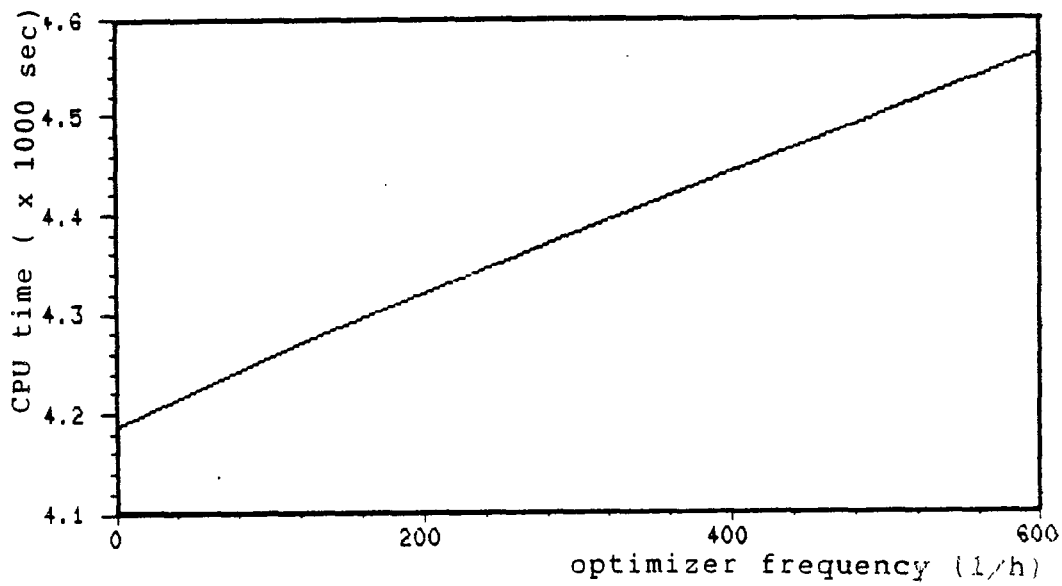

Fig.5.5.10. CPU Time Demand for Two Hours Simulated Time as a Function of Optimizer Execution Frequency on a MicroVAX under VMS

This result is as expected. The curves that are shown here correspond to part B in figure 5.5.7. In figure 5.5.7, the loss of profit is given, which should be minimal, while on figure 5.5.11 the plant efficiency is presented, which should be maximized. In figure 5.5.7 the loss of profit is given versus the ratio of $\log(N_s/N_v)$. As the optimizer execution frequency increases, the value of this expression will decrease. In figure 5.5.11 the frequency of optimizer execution is used. Therefore both axes in figures 5.5.11 should be reversed to compare to figure 5.5.7. It is clear that lowering the optimizer execution frequency can cause a significant loss in overall average plant performance. The results are given in % efficiency, and a loss of 0.05% can be significant in boiler networks. Therefore, it makes sense to keep the on-line optimizer frequency high. The significance check on the setpoint changes makes fast optimizer runs possible, without much concern about interactions between on-line optimizer and plant dynamics.

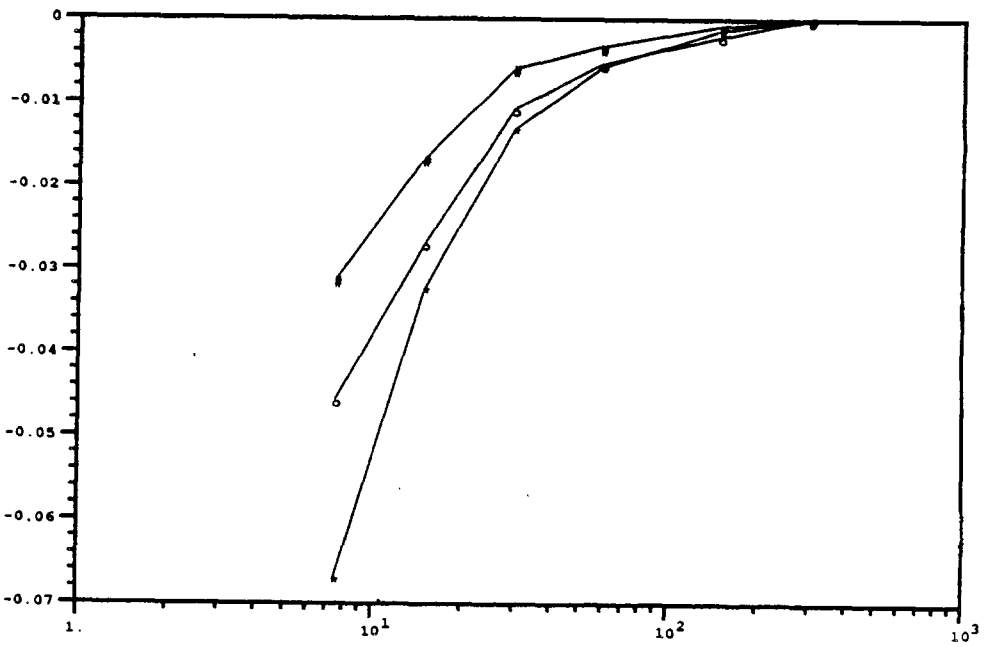

Fig.5.5.11. Plant Performance Changes in % Efficiency as a Function of Optimizer Execution Frequency for Different Steam Demand Load Step Frequencies. Notations : $\star$ $2h^{-1}$; o $6h^{-1}$; $\#$ $12h^{-1}$.

5.5.4. Effects of Noise

Noise of various amplitudes influences the performance of the on- line optimizer in different ways. Results obtained from the case studies are presented. They illustrate different ways in which noise influences the data processing that takes place in the optimizer. First some results from the heat exchanger case study are listed. The heat exchanger results are preceded by an analysis of the parameter estimation problem. Finally, conclusions on this topic are presented. Next results that are obtained from the boiler case study are presented.

5.5.4.1. Heat Exchanger Case Study

5.5.4.1.1. Model Updating

The models that are used in this example are described in Chapter 4. A rigorous model acts as plant, while a simplified model is used in the on-line optimizer. Both models are steady state models. The most important difference between these models concerns the calculation of the heat transfer in the heat exchanger. In the rigorous model ($\simeq$ plant), the flow rates influence the heat transfer coefficient. In the simplified model, the heat transfer coefficient is a constant that has to be updated. Two data sets are generated using the rigorous heat exchanger model. Only one heat exchanger is considered in this comparison. Both sets contain two thousand data points. The difference between both sets is that the first set covers a range of kerosene mass flow rates while the temperatures are maintained constant, while the second set is characterized by constant flows but different kerosene inlet temperatures. The first set was generated for a crude mass flow of 149000 lb/h, a crude inlet temperature of 100 F and a kerosene inlet temperature of 390 F. The kerosene mass flow varied from 33800 lb/h to 53800 lb/h. To that result, noise was added with zero mean and a standard deviation of 3% of the nominal value of the variable to which it is added. The second set is generated in the same way, but with a constant kerosene mass flow of 43800 lb/h and a kerosene inlet temperature varying between 290 and 490 F. Again, noise is added to all variables. More details are given in Appendix G (Run # 20). Table 5.5.1 shows statistical results obtained by fitting the simplified model to both data sets. It is clear that the simplified model fits the data with variable temperatures but constant flows very well, but performs poorly on the set with constant temperatures but variable flows. Figures 5.5.12 and 5.5.13 compares estimated exchanged heat with the calculated exchanged heat for both data sets and illustrate that this

lack of fit is due to a systematic error. For an ideal result, the estimated and calculated exchanged heat would be equal. The residuals from the fit to the second data set (constant flow rates) have a distribution (not shown) that is closer to the normal distribution. These results are obtained while using a least squares estimation technique, i.e. the estimation criterion is the sum of squares of the differences between the estimated and the calculated ($\simeq$ measured) heat transfer.

Test Statistic	**Changing Kerosene Flow**	**Constant Kerosene Flow**
R^2	0.2110	0.9915
Res. Mean Sq.	$0.614 \cdot 10^{11}$	$0.162 \cdot 10^{11}$
F	1158	120000

Table 5.5.1.: Statistical Results HX-Fitting

The on-line optimizer that will optimize the load allocation between n (in our case study 2) heat exchangers, will influence the flows over the heat exchangers. Therefore, the on-line optimizer can invalidate its own model fairly easily. This is very important, since it can conceivably cause instable oscillations. However, probably there will exist a k in equation 2.2.2 (section 2.2.4) to stabilize the on-line optimized plant. Simulation results show that although oscillations may occur, they decay fast.

An important question concerns the length of the data base under noisy conditions. The parameter φ (see equation 4.3.12 in section 4.4.4.2) can be

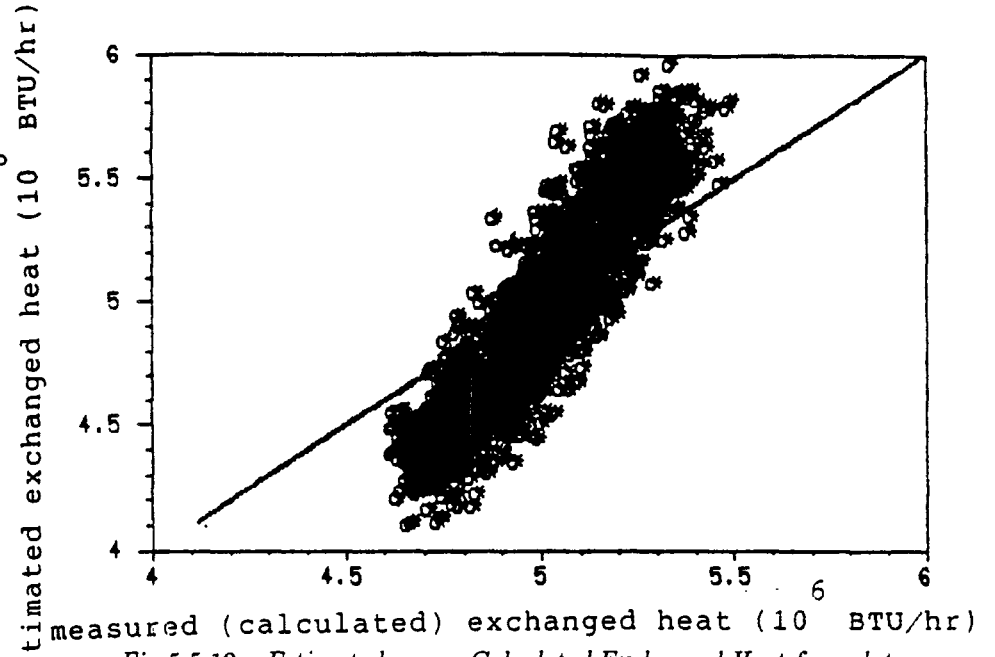

Fig.5.5.12. Estimated versus Calculated Exchanged Heat for a data set with Variable Kerosene Flow Rates

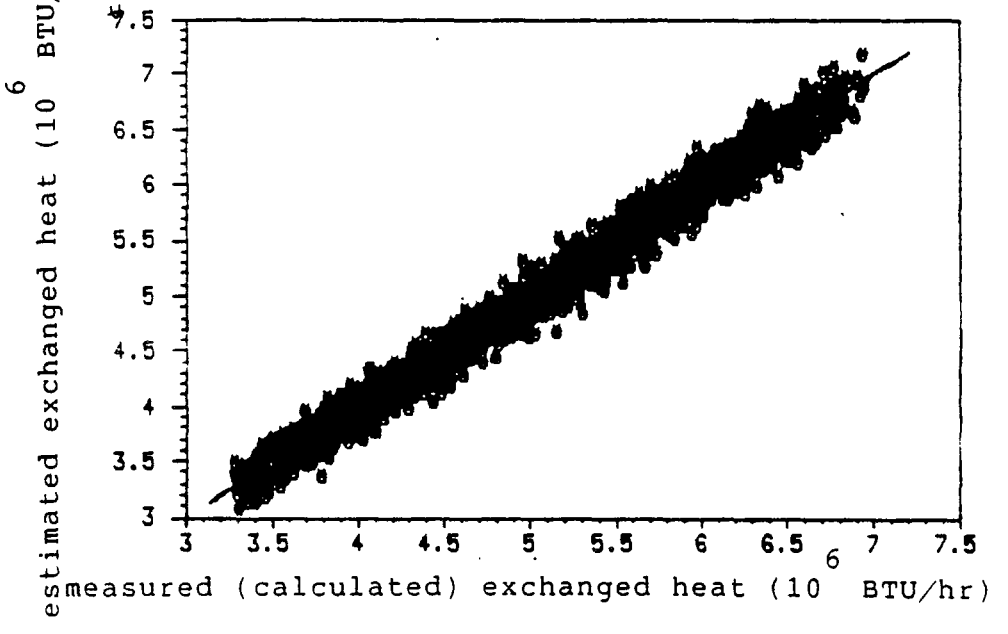

Fig.5.5.13. Estimated versus Calculated Exchanged Heat for a data set with Variable Kerosene Inlet Temperatures

obtained from one data point. Clearly, if φ does not vary, the estimation of the parameter will improve with the number of data points used under noisy

conditions. On the other hand, if a trend in φ exists, then use of a large number of data points will bias the estimate of the parameter. Therefore a trade off exists between minimizing the variance of the parameter estimate (increasing number of data points) and minimizing the bias on the parameter (decreasing number of data points). In Appendix F it is shown that the data base length that minimizes the magnitude of the *expected error in the parameter estimate* is n such that the integer n minimizes :

$$\sqrt{\frac{2}{\pi n}} \cdot \frac{1}{D} \cdot \cosh\Big(\frac{(n-1)^2 n^{\frac{1}{2}} D^2}{8}\Big) + \frac{n-1}{2} \cdot \Big(P\big(\frac{(n-1)\sqrt{n}}{2} \cdot D\big) - P\big(-\frac{(n-1)\sqrt{n}}{2} \cdot D\big)\Big). \tag{5.5.4}$$

In equation 5.5.4, D is the normalized linear drift of the parameter φ, given by :

$$D = \frac{\Delta\varphi}{\sigma_\varphi}, \tag{5.5.5}$$

with $\Delta\varphi$ the change of the true value of φ in between data points, and σ_φ the standard deviation of the estimation of φ if only one data set is used. $P(\cdot)$ is the probability that a random number in the normal distribution with zero mean and unit variance is smaller than the argument of $P(\cdot)$.

A variable length of database is used in the case study. A straight line is fitted through the last ten estimates of φ, in order to obtain D. Then n is calculated so that expression 5.5.4 is minimal, with a maximal value of 10. This maximal data base length is chosen arbitrarily. The application is simplified by using a moving average of n parameter values that fit one data point instead of a least square fit of n data points. Using a moving average instead of a least square fit does not change the results significantly. A comparison between both techniques on different data bases showed results that were usually undistinguishable.

A comparison is made between a run using the variable data base length and a run using a constant data base length. Since the simulation are steady

state, this is equivalent to assuming that the changes in operating conditions as well as the optimizer execution frequency are slow compared to the dynamics of the heat exchangers. A simulation is run over 100 optimizer executions. Between time units 20 and 40, the kerosene inlet temperature of one of the heat exchangers is ramped from 390 to 290 F. Two percent standard deviation random zero mean noise is added to the data before processing. Figure 5.5.14 gives recuperated heat as a function of time for variable data base length (full line) and for constant data base length (dotted line). The difference between both curves is given in figure 5.5.15. after time unit 20. Figure 5.5.16 gives the estimated φ for variable data base length, and the data base length as well. It has to be noted that a test for significant change in setpoints is applied here, as described in Chapter 3. More details are given in Appendix G (Run # 21).

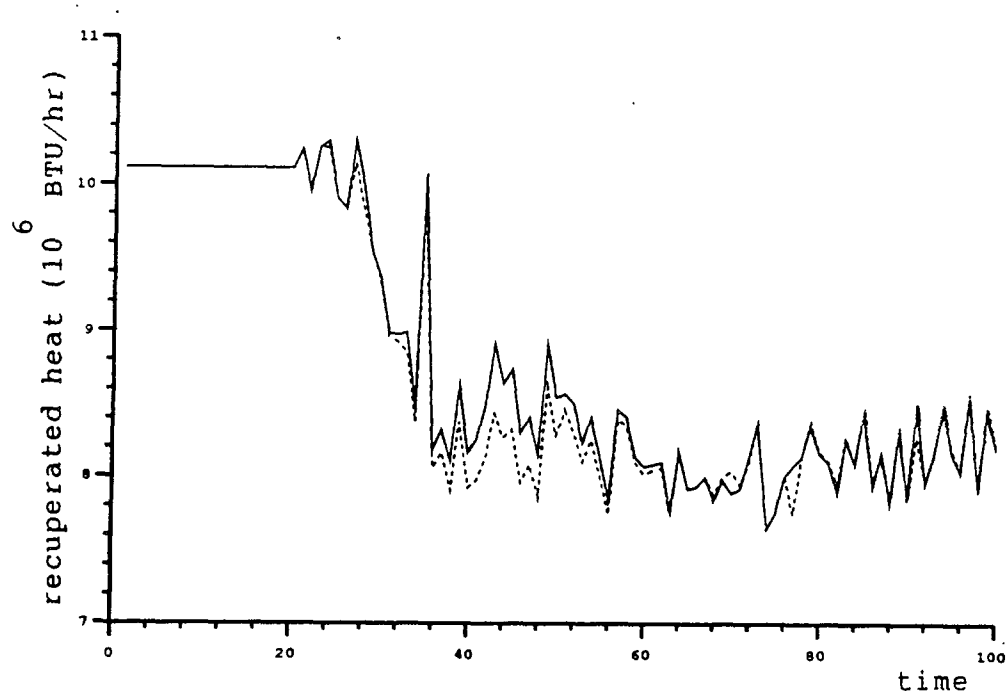

Fig.5.5.14. Recuperated Calculated Heat Variable (Full Line) and Constant (Dotted Line) as a Function of Time for a Ramp in HX-1 Kerosene Inlet Temperature.

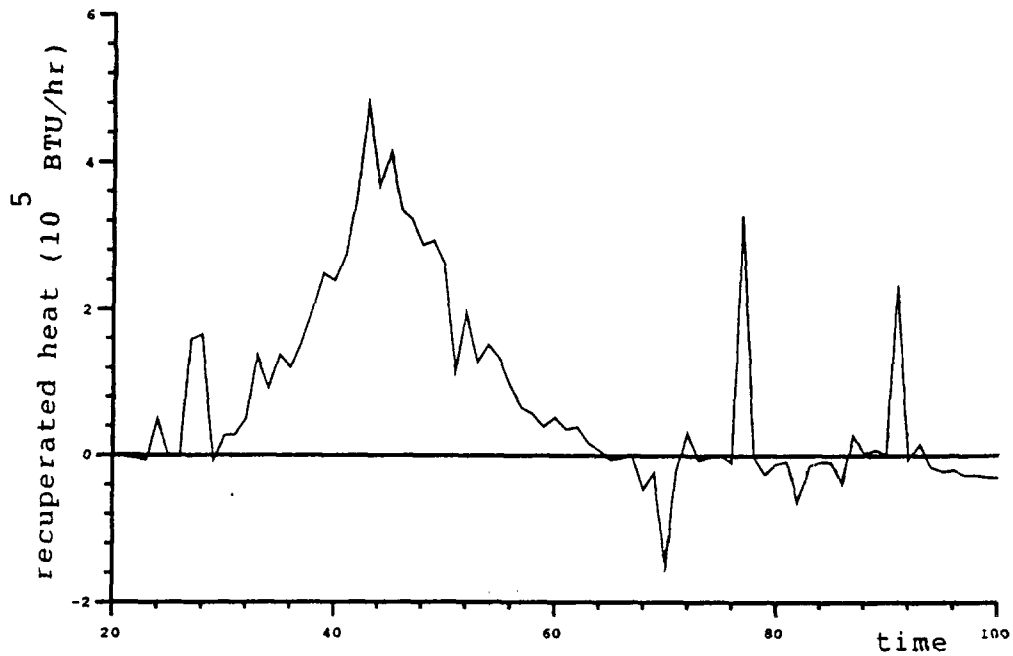

Fig.5.5.15. Difference between Estimates for Recuperated Calculated Heat for Variable (Full Line) and Constant (Dotted Line) as a Function of Time for a Ramp in HX-1 Kerosene Inlet Temperature.

An illustration of convergence of the on-line optimizer in spite of the model plant mismatch is given in figure 5.5.17. The top graph in figure 5.5.17 gives the true (from the rigorous model – full line) and the estimated (from the simplified model – dotted line) heat recuperation as a function of time. The bottom graph gives the implemented setpoint (full line) and the optimizer result (dotted line). No noise was added in this run. At time 45 the HX-1 kerosene inlet temperature increases from 290 F to 490 F. At time 70 the original situation is restored. All the other values are nominal. It is clear that after the first change in setpoint, some time is spent reestimating the changed model. The changing model parameter estimate creates the image that the calculated heat transfer actually decreases. However, the true heat recuperation improves.

5.5.4.1.2. Effects of Noise

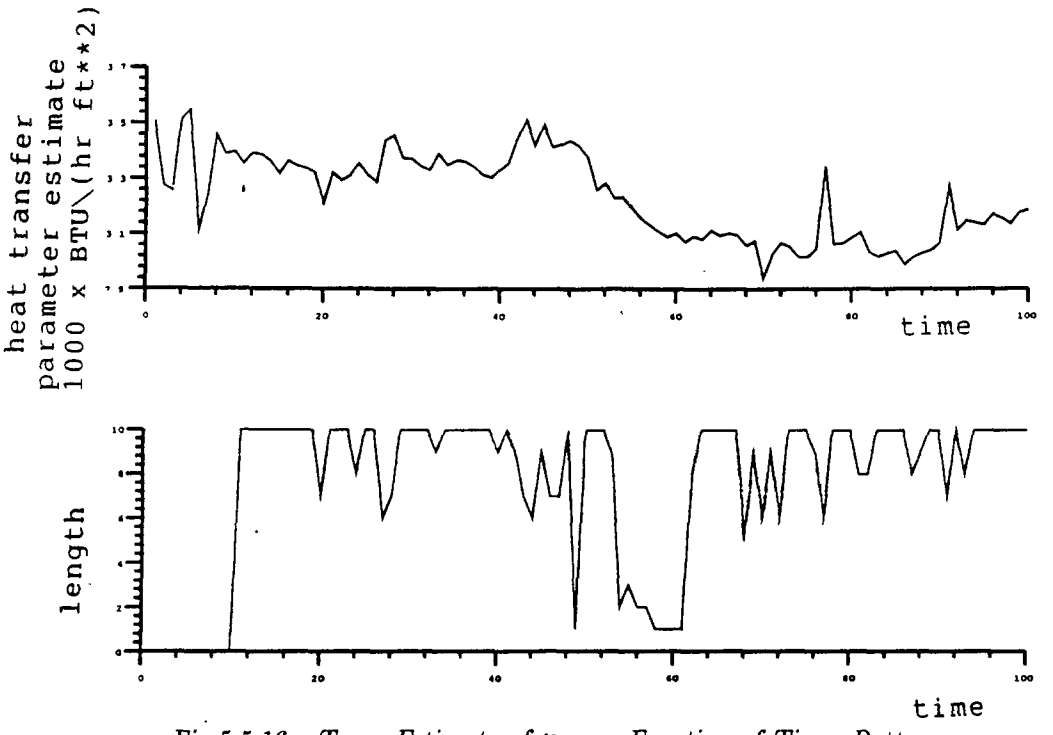

Fig.5.5.16. Top : Estimate of φ as a Function of Time, Bottom : Length of Variable Data Base as a Function of Time.

Results for different types and levels of noise are discussed and analyzed. The first results concern stationary white noise, and stationary non-white noise. Then a result using nonstationary noise is presented. In all runs the same input is used. A run is 120 time units long. From time 0 to time 20, the kerosene HX-1 inlet temperature equals 390 F. Then this temperature increases to 490F until time unit 45. From time unit 45 to time unit 70, the HX-1 kerosene inlet temperature drops to 290F. Then this temperature rises again to 490F until the time 95. Finally the temperature drops and stays at 290F until the end of the simulation. Variable data base length and significance check of setpoint changes are applied.

Stationary White Uncorrelated Noise

A stationary white noise is added to all measurements. The noise is zero mean, and gaussian. A detailed description is given in Appendix G (Run #

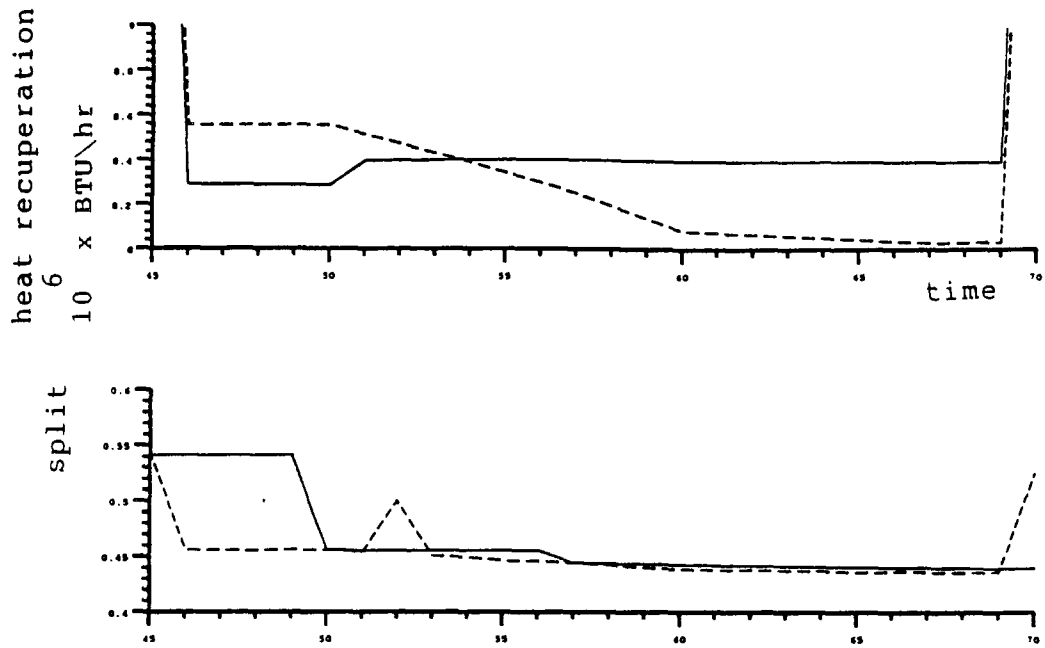

Fig.5.5.17. Optimizer Results (Dotted Line) and Implemented Set-point (Full Line) as a Function of Time

22). Figure 5.5.18 gives the change in performance (average recuperated heat) for different levels of noise compared to the noiseless case. The noise level is given in standard deviation as a fraction of the nominal measurement value. For instance, a value of 1% means that noise was added to each measurement, and that that this noise has a standard deviation of 1% of the nominal value of that measurement. It can be seen from the figure that a very important transition takes place between 1 and 2 percent noise. Figure 5.5.19 makes clear what causes this transition. Figure 5.5.19 presents the implemented allocations for increasing levels of noise. If one compares the setpoints for a noise level of 1% with that of 2%, then it can be noted that the check for significant change of setpoint does not always detect a significant change at the same times. At a noise level of 5 % hardly any significant changes in setpoint are detected. This does not mean that the test for significant changes obstructs optimizer

performance. The curve in figure 5.5.18 does not change significantly if the test is not applied, because only steady states are considered here. It means that not enough information is available in the data, to carry out on-line optimization. The results in these two figures indicate that for as much as 2 percent white gaussian zero mean noise, on-line optimization of the load allocation of these two heat exchangers becomes virtually impossible.

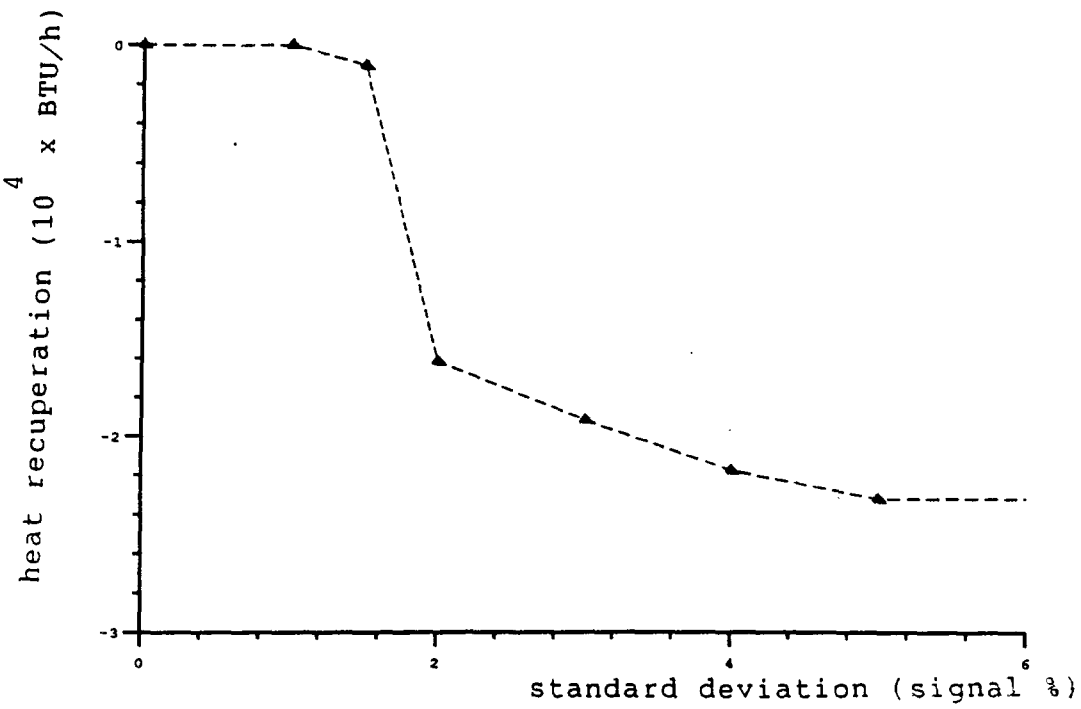

Fig.5.5.18. Change of Heat Transfer as a Function of Standard Deviation of the Noise. The Standard Deviation is Given in Percentage of the Nominal Value of the Measurement to which it is Added

Stationary Non-White Correlated Noise

Since flows are often measured as differential pressures, it makes sense to assume that not the flow, but the square of the flow signal will have white noise superimposed on it. In this case it is assumed that the square of the flow has zero mean random walk noise. Furthermore, the noise on measurements of mass flow and temperature has been correlated for the same flow. The following

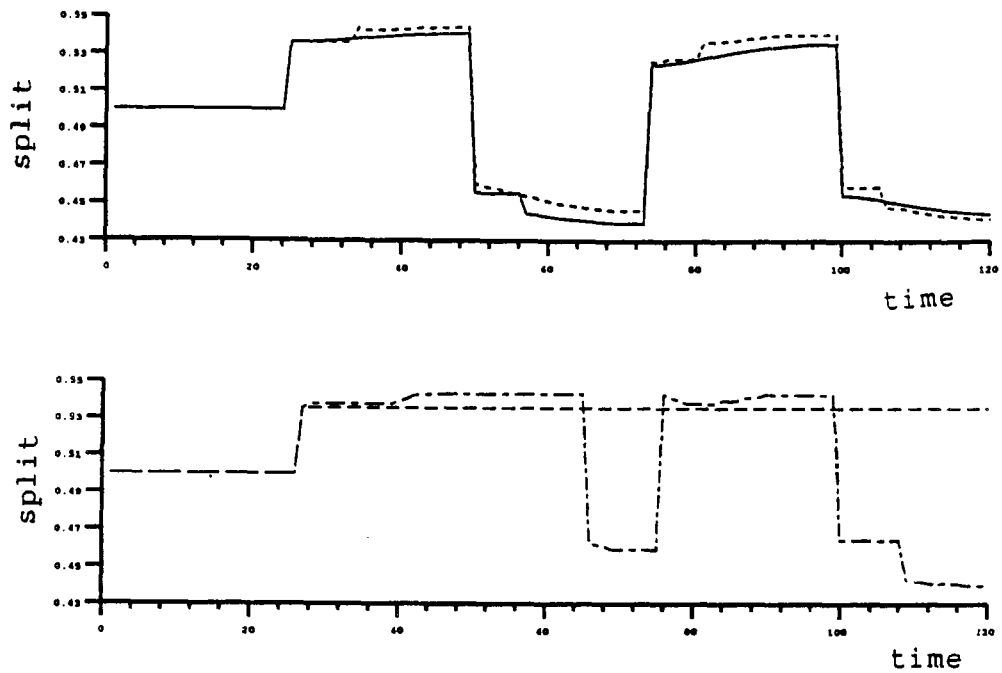

Fig.5.5.19. Setpoints for Different Levels of Noise. Top : Full Line = 1% Standard Deviation, Dotted Line = 1.5% Standard Deviation; Bottom : Short Long Interrupted Line = 2% Standard Deviation, Dotted Line = 5% Standard Deviation.

equations are used :

$$\hat{F}^2 = F^2 + RW; \tag{5.5.6}$$

$$\hat{T} = T + K \cdot \frac{1}{2} \cdot (RN - RW). \tag{5.5.7}$$

In these equations, F is a flow measurement, and T a temperature measurement. RW is a random walk variable (and is function of the previous random walk number added to that measurement) and RN is a random white gaussian variable. K is a scaling constant to match the variances. (More details in Appendix G (Run # 23).)

With this noise applied, results are obtained that are not significantly different from the results that are obtained using white gaussian uncorrelated noise for noise with the same variance. The system is quite sensitive to noise, and actually performs already relatively poor under white uncorrelated noise. It

is interesting to point out that for a high amplitude of noise, the system will move to the use of long databases. Because of that reaction, the effect of time correlated noise becomes less effective.

Non-Stationary Non-White Correlated Noise

The noise model that was described in the previous section is applied in this case, but with a shifting variance. From time 30 to time 75, a standard deviation of 5% is implemented. At all other times, the standard deviation is 0.5% . It is important that the technique recovers from the period of "loud" noise. (More details are available in Appendix G (Run # 24).) Figure 5.5.20 gives results. The top of the figure gives a graph with the setpoint (full line) and the optimizer result (dotted line) for the described change in noise. The figure below, gives the same without the change in noise level between time 30 and time 75. First of all, look at the dotted line (optimizer result) in the top graph. At time instant 30, the optimizer result does show some increased noise, but the amplitude is rather low, since the parameter estimation maintained a long data base, and hence filters out the noise substantially. The first change in kerosene temperature is taken fairly well, although from then on the model starts to deteriorate rapidly. This is due to the fact that the data base length becomes smaller, since the change in the parameter (because of the change in the setpoint) becomes apparent. The next step in temperature is handled well by the optimizer results. However, the accuracy of the calculated setpoint is quite low by then, and the significance check will not let the setpoint follow sudden moves any more. By the time the noise level drops again, the data base length is increasing again, since time instant 75 comes after a change in temperature. Furthermore, high noise levels tend to increase the data base length in order to obtain the minimal expected bias. This high noise level remains responsible for a relatively low expectation of accuracy for some time.

After the next temperature shift, the system recovers completely. This means that although the consequences of a sudden drop in data quality due to increased noise can be noticed for a certain period of time after the period of increased noise, the system does recover, and is kept at a safe (conservative) setpoint by the test for significant changes in setpoint.

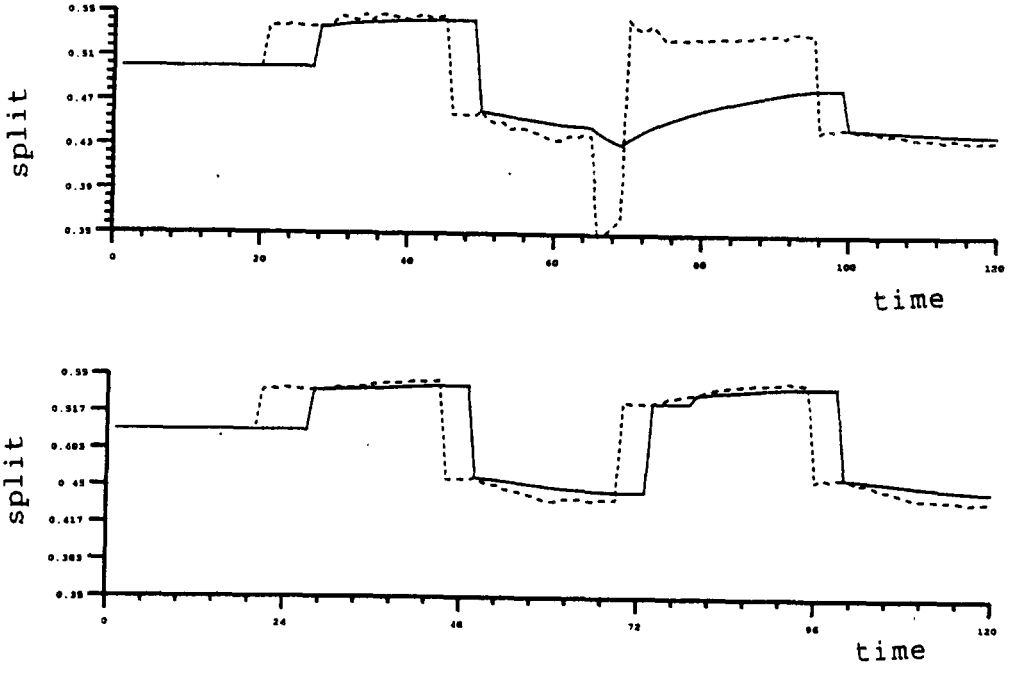

Fig.5.5.20. Optimizer Result (Dotted Lines)and Setpoint (Full Lines) for an Increased (5%) Standard Deviation Between Time Units 30 and 75 (Top Graph) and for constant 0.5% Standard Deviation Non-White Correlated Noise (Bottom Graph).

5.5.4.2. Boiler Case Study

The boiler case study provided similar results as the heat exchanger case study.

Stationary White Uncorrelated Noise

- Accuracy Check on Setpoints

White noise with zero mean was added to the measurements that are used

by the on-line optimizer and to the measurement (pressure) that is used by the control. In that way, process correlated noise was obtained. A stepwise increase in boiler network is implemented using the load sequence in table 5.4.4. The on-line optimizer is run every minute. See Appendix G (Run # 25) for more details. Figure 5.5.21 shows the change in performance (overall efficiency) as a function of the standard deviation. A level '1' refers to a standard deviation of 0.1 lb/sec on the flow measurements, and 1 F on the temperature measurements. The other levels are proportional to this. A trend towards loss in performance is visible.

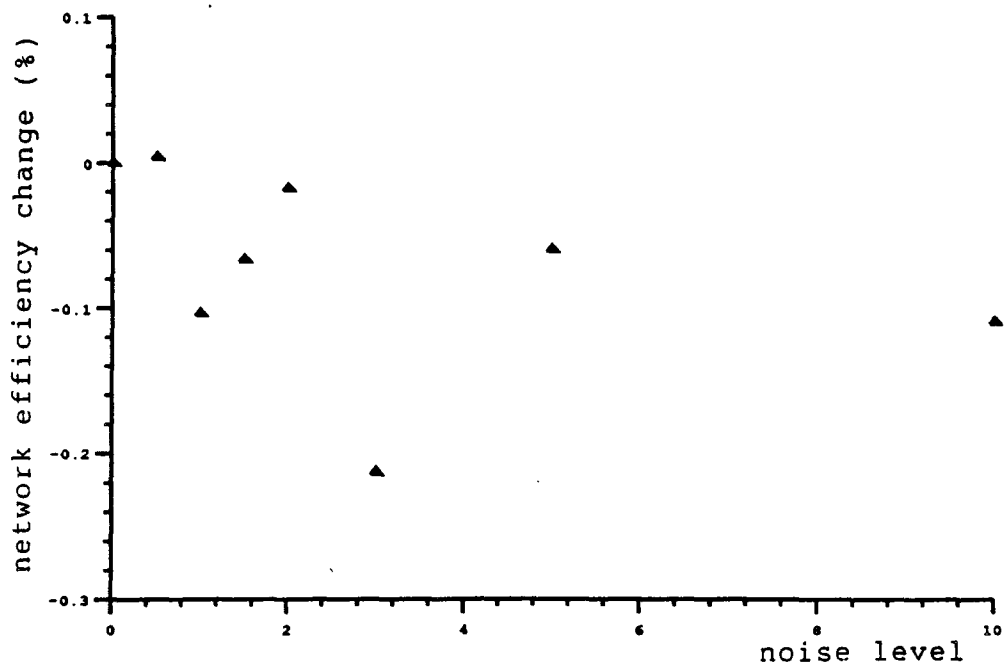

Fig.5.5.21. Change in Time Average Boiler Network Performance a a Function of Noise Standard Deviation

The noise that is added to the measurement data influences the model updating. The steady state model of the boilers that is used is merely a second order polynomial. These curves are relatively "flat", this means that the range over which the efficiency changes within the normal operating condition is relatively small. Therefore, even small amount of noise will make model updating

very difficult. Boiler efficiency is a calculated variable, and the noise in this calculated variable as a result of the noise in the contributing measurements is considerable. If standard deviation of the noise on temperature measurements is 1F (95% inaccuracy of ± 2F) and the standard deviation of the noise on flow measurements is 0.1 lb/sec (95% inaccuracy of ± 0.2 lb/sec), then the noise on the calculated efficiency has a standard deviation of 0.6 %. That results in a 95% confidence interval of ± 1.2%. This figure is a result from simulations with the rigorous dynamic boiler model as described in Chapter 4. Green and Al ai Shaikh [1980] list 1.17% as a 95% confidence limit on the efficiency calculation for a typical gas fired boiler, and 1.19% for an oil fired boiler. The efficiency of the boilers that are studied in this case study only varies about 2% over the range of operation. In other words, even for the low noise that is implemented, no more efficiency information as a function of boiler load, is available. The steady state data are of course filtered, but nevertheless the influence of noise is very significant. If the model updating is disabled, then the sudden large increases and decreases do not occur, since no new (faulty) steady state data are accumulated. Small levels of noise will be reflected in the model parameter estimates, and will in turn influence the setpoints significantly, since these are quite sensitive to changes in model parameters . That is clearly visible in the following figure (5.5.22). In that figure, the fuel loads for boiler 2 are given as a function of time, for different noise levels and for the load increments as given in Table 5.4.4. Soon after the start of the simulation the parameter updates cause changes in the setpoints that are suggested by the optimizer. The results are in some sense random. When the noise level becomes high, the test for significance prevents fast changes of the setpoints, and the fuel loads only change sluggishly. Therefore the performance of the system at very high noise levels can be better than for intermediary noise levels.

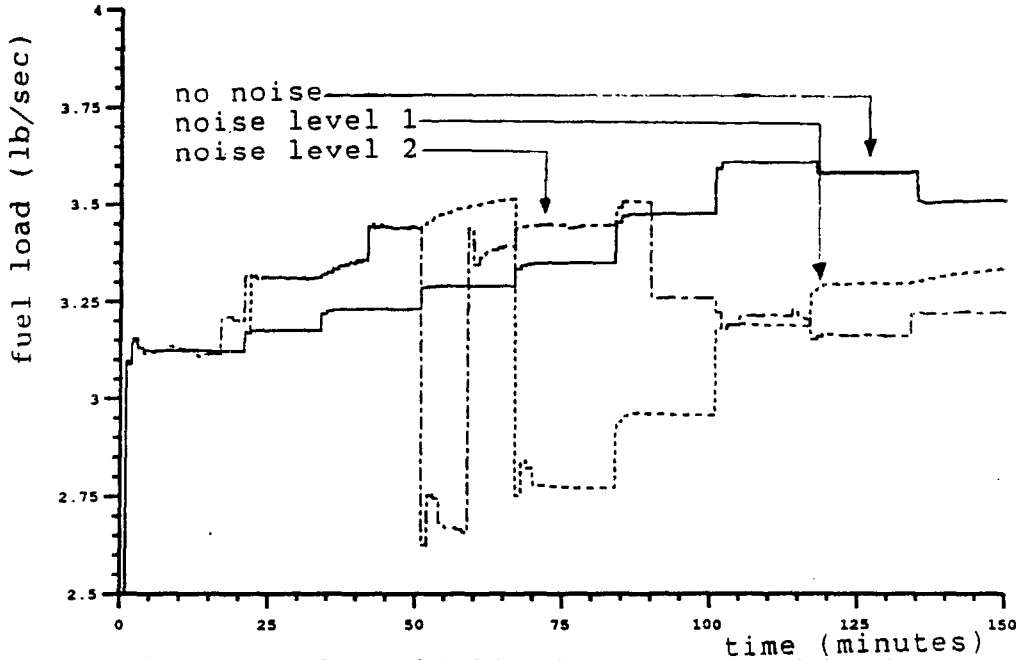

Fig.5.5.22. Implemented Fuel Setpoints as a Function of Time for Different Levels of Noise

First of all, these results show once more that the load allocation of boilers feeding a common header is not a good candidate for on-line optimization. The efficiency of a boiler is too sensitive to noise compared to the range over which it varies, to allow an effective data collection.

These results also show that a test for significant change in the setpoints may not always work. Indeed, in figure 5.5.22 the fuel setpoint for boiler 2 does change over large values. If these changes are due to inaccuracies in the model parameters, then the test should consider the jumps in setpoint as insignificant, and neglect them. Apparently, the confidence intervals of the setpoints is underestimated. This is due to a violation of the assumptions that were accepted in section 3.6 while the test for significant change in setpoints was described. The optimization results are not linear functions of the model parameters.

Stationary White Uncorrelated Noise

- Accuracy Check on Setpoints and Objective

Details on this example are given Appendix G (Run # 26). Since the

test for significant changes in setpoints does not seem to function entirely as expected, a check for sufficient accuracy of the optimal value function is introduced as well. The test compares the expected 95% confidence interval of the expected network efficiency with a threshold. If the network efficiency cannot be determined accurately (compared to a threshold) the results of the optimization is not implemented. In this example the results of the on-line optimization were not implemented if the expected efficiency could not be obtained with at least 0.1% accuracy. This is *not* a very low value, since the optimal network efficiency of the network does not vary over more than 1% over the complete range of operation. An equivalent of figure 5.5.21 is given in figure 5.5.23. Note that the maximal loss of performance is much lower than the results obtained without the test on the objective and given on figure 5.5.21. Figure 5.5.24 gives fuel loads for boiler 2 for different levels of noise. It is clear that even for small levels of noise the test inhibits implementation of the optimizer results. At lower noise levels this happens later, since the uncertainty increases with time after new steady state points are collected. The results do not change significantly if the threshold is increased to 0.2% . An important question is whether or not the system will recover from a period of high noise. That issue is dealt with at the end of this section.

Stationary Non-White Correlated Noise

In the section on results obtained with the heat exchanger network, random walk noise was introduced on the square of the flow measurements. Correlation was introduced between the flow and temperature measurements in the heat exchanger network example as well. The same was done in the boiler case study. Details are provided Appendix G (Run # 27). Results that were obtained in this way are not significantly different from the results that are obtained with white uncorrelated noise. The influence of even small levels of noise on this

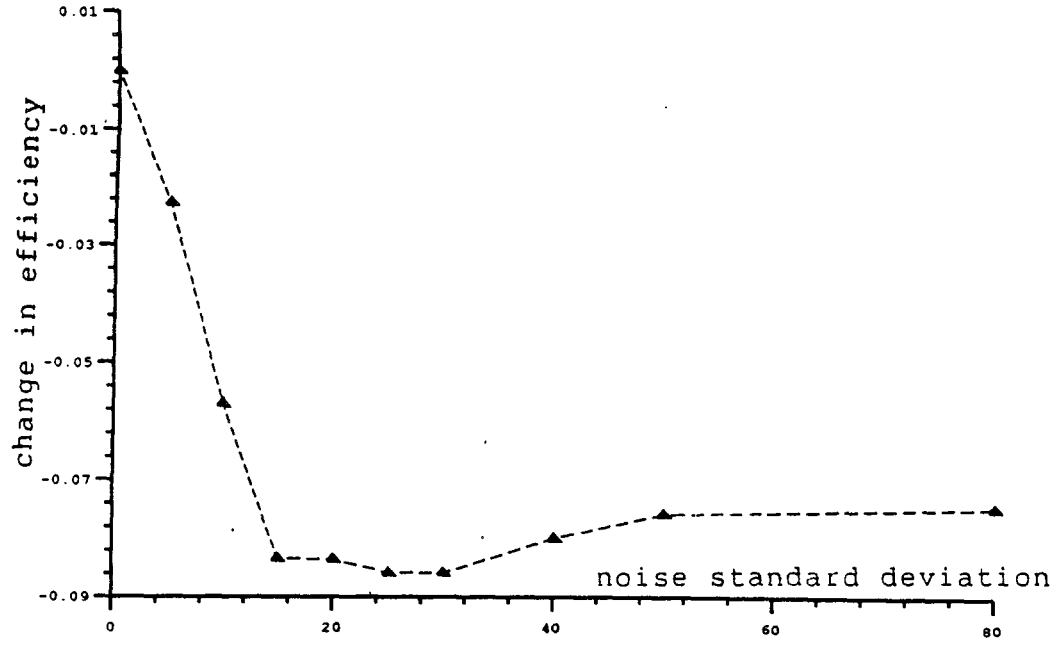

Fig.5.5.23. Change in Time Average Boiler Network Performance a a Function of Noise Standard Deviation

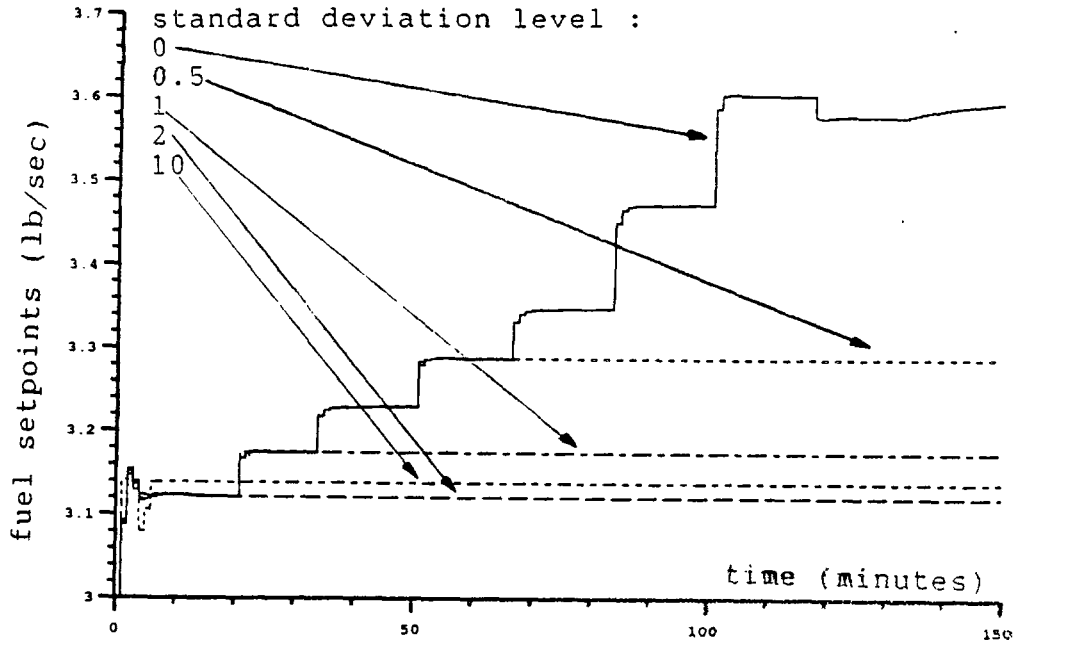

Fig.5.5.24. Implemented Fuel Setpoints as a Function of Time for Different Levels of Noise

boiler case, is also for white noise large. Deviations of the noise type from the ideal white noise does not deteriorate the performance further in a significant

way.

Non-Stationary Non-White Correlated Noise

It is important that the system will recover from periods of increased noise. From the results of an experiment described below, it seems that the boiler on-line optimizer does not recover completely in a short period of time. The boiler network is presented with four phases while the boiler loads increase as detailed in Table 5.4.4. During phase 1 the boiler noise is on level 0.5, during phase 2 the noise level is doubled. During phase 3 the noise level increases to the tenfold of the initial level. Finally, during phase 4 the noise level returns to the initial noise level. Figure 5.5.25 presents the fuel load for boiler 2 as a function of time (full time). The dotted line gives the fuel load for boiler 2 in case the noise stays the same throughout the run. Only a test for significant changes in setpoint is used here. The noise is correlated, non-white. Details are listed in Appendix G (Run # 28). It can be seen that the performance of the on-line optimizer is poor during the periods of high noise. During that period (actually during phase 2) a steady state is picked up that causes an offset between the full and the dotted curve for the length of the simulation. This offset will be present until a new steady state data point will replace the data point that was collected during phase 2. Nevertheless the general trend of the full and the dotted line in phase 4 is the same. Most differences are due to actions of the setpoint test.

5.5.4.3. Conclusions

Noise influences the performance of an on-line optimizer in different ways.

First of all, more noise makes steady state detection more difficult. Also model updating becomes exceedingly difficult as the noise level increases. For the case studies that are considered here, model updating becomes nearly im-

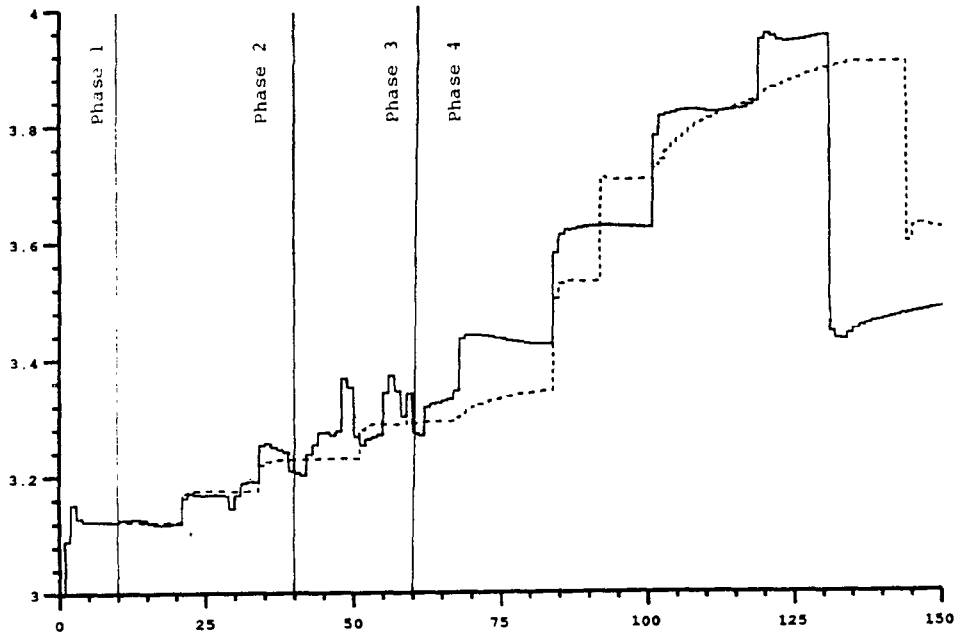

Fig.5.5.25. Boiler 2 Fuel Setpoints (in lb/sec) as a Function of Time (in Minutes) for a Period of Increased Noise (Full Line) and for Constant Noise Level (Dotted Line).

possible under noisy conditions. The boiler optimization performs well under high levels of noise, if the model updating is disabled, and the boiler characteristics are constant. The case studies that are considered here, in particular the boiler case study, have models that are too sensitive to data variance to be updated under noisy conditions. In the boiler case, the boiler efficiency is the model response. For low noise level the efficiency "signal" disappears under the noise.

As the noise level becomes higher, safety structures will be activated. Tests for significance of the optimization objective and the setpoints are designed to keep the process from performing worse than without on-line optimizer. It was shown that if the assumptions of these tests are violated considerably, also these tests can fail. The performance of the on-line optimizer plant then becomes virtually identical to the plant without on- line optimizer.

The actual influence of noise on the expected performance is very difficult

to predict, since it is not known exactly how the noise will influence the model updating routine.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1. CONCLUSIONS

A real time optimization structure is presented and described in Chapter 3. It can be applied to continuous chemical plants that are operated under steady state such that transient periods are short compared to periods of steady state operation. The on-line optimizer makes use of a steady state model, that in general is nonlinear. Model parameters are updated to adjust the model for changes in operating condition as well as changes in the characteristics of the equipment. Optimization results are implemented after being checked for accuracy.

The following new methods were proposed :

- ◇ a steady state check that works under noisy conditions ;
- ◇ applications of sensitivity analysis in on-line optimization :
  - off-line applications focussing on feasibility ;
  - on-line applications, combining statistical analysis results with sensitivity results, focussing on accuracy and confidence in new on-line optimizer results ;
- ◇ distributed gross error identification.

The use of sensitivity results was illustrated extensively.

Some important points with respect to on-line optimization were pointed out and/or illustrated :

- ◇ the possible interaction between data reconciliation and model estimation ;
- ◇ practical problems with data reconciliation and gross error identifation ;
- ◇ usefulness of statistical analysis ;
- ◇ the influence of noise ;
- ◇ the determination of the optimizer execution frequency.

These points are illustrated with case study examples. The use of the SQP optimization algorithm was suggested.

Furthermore, the following topics are reviewed :

- ◇ current on-line optimization applications that are reported in the literature ;
- ◇ observability, redundancy, data reconciliation and gross error reconciliation ;
- ◇ model parameter estimation (criterion and its minimization, detection of outliers) ;
- ◇ distributed optimization efforts.

6.1.1. Sensitivity Analysis

Important results were obtained in the application of sensitivity analysis to real-time optimization problems. Sensitivity analysis provides an analytical approximation of the solution of the optimization problem in a neighborhood around the current operating conditions. Theory on nonlinear programming sensitivity results are extensive and powerful, nevertheless implementation of these results is not very widespread. Applications of sensitivity analysis are proposed in this work, and illustrated in case study results. Industrial application of real time optimization on a wide scale is not likely without sensitivity analysis. Sensitivity results can be obtained without large CPU effort. At the same time, they contain significant information concerning the status of the

optimization solution.

6.1.1.1. Off-Line Use of Sensitivity Analysis

Two methods were suggested for the discrimination between interesting and uninteresting candidate systems for real time optimization at an early stage in a project. A detailed description is given in Chapter 3. Illustrations are provided in Chapter 5. Both methods rely on results of a sensitivity analysis.

6.1.1.1.1. Marginal Optimization Gain

This method obtains an upper limit for the gain that can be expected with an on-line optimizer, compared to a plant that is optimized off-line. In an illustration on the boiler case study, it was shown that the load allocation of boilers feeding a common header is not expected to be a profitable application of on-line optimization.

6.1.1.1.2. Minimal Accuracy Study

A comparison is made between the expected range over which the objective function varies, and the accuracy with which it can be calculated. This is not unlike a comparison of noise to signal. If the objective does not vary over a range that is large compared to the accuracy with which it can be obtained, on-line optimization should not be applied. It was shown that the distillation example passed this test satisfactorily.

6.1.1.2. On-Line Use of Sensitivity Analysis

Since a sensitivity analysis is computationally inexpensive, sensitivity analysis can be carried out on-line. Two applications are proposed and illustrated. In both techniques the results of a sensitivity analysis are combined with the results of a statistical analysis of measurements and/or model parameters.

6.1.1.2.1. Sufficient Accuracy of Setpoints and Objective

By combining sensitivity and statistical results, confidence intervals can be calculated for setpoints and objectives. These confidence intervals can be compared to preset values. If confidence intervals become wider than the preset value, implementation of the optimization result is interrupted. This technique adds safety to the on-line optimizer. It is illustrated in the boiler case study.

6.1.1.2.2. Significant Change in Setpoint

Setpoint changes should only be implemented if they are statistically significant. The plant will indeed filter out some of the insignificant setpoint changes, but nevertheless the implementation of this test will quiet the plant behavior considerably. A very efficient method is proposed for a multivariable check for significant differences between arrays of correlated setpoints. Again the boiler case study serves as illustration. The test is also implemented in the heat exchanger case study.

6.1.2. Data Reconciliation

An on-line optimizer is dependent upon data of good quality. Poor data will result in large modeling errors and will result in optimization results that do not have sufficient accuracy to be implemented. Data reconciliation and gross error identification offer the possibility to compare data with mass and energy balances.

It was shown that data reconciliation can be useful, and improve the results of the estimation and the optimization steps. It was also illustrated that data reconciliation is not without risk, and that reconciled data can actually be poorer than the original data under certain conditions. Gross error identification is a very useful tool to detect systematic errors in the data set, and can expected to become a standard process monitoring module. Its effectiveness is illustrated

in an example.

Problems with data reconciliation and gross error detection are pointed out as well. Many algorithms do not provide the possibility of introducing inequality constraints, which is necessary to guarantee realistic optimization results. No algorithms guarantee good performance if more than one gross error is present. The most important problem concerns the high demand in CPU time posed by data reconciliation and gross error detection routines. A remedy is offered by splitting the plant in sections and by carrying out reconciliation and gross error detection in every subsection. This reduces the CPU effort considerably, but it is illustrated that at the same time the performance suffers from this remedy.

Finally it is noted in Chapter 3 that data reconciliation and model updating have to be used together with some caution. If the data reconciliation introduces linear relationships between model responses, the estimation criterion can become singular.

6.1.3. Modeling

The on-line optimizer, as it is conceived in this work, uses a nonlinear steady state model. In the selection of this model, a trade off has to be made between model accuracy and simplicity. Model updating problems can reduce the optimizer performance considerable. Illustrations are provided.

In an on-line optimizer, the steady state model captures the available knowledge on the plant behavior and characteristics. The numerical optimization can be large and involved, but the state of the art optimization algorithms (e.g. SQP) that are available, can solve the optimization problems that are encountered in most applications of on-line optimization. However, it is important to determine *what* is being optimized. The steady state model determines that. Therefore good modeling and model updating is of essential importance

in on-line optimization. In an illustration it is shown that most of the optimization profit can be lost by some convergence problem in the distillation model fitting. In the boiler case it is shown that the faulty use of unsteady data as steady state data can lead to important modeling errors.

6.1.4. Dynamic Interaction between Plant and On-Line Optimizer

Plants and their control systems can be considered as continuous systems, while the on-line optimizer is a discrete time system. The interaction between both is important, and is studied in this work. The stability of an on-line optimized system is only guaranteed under strict conditions, and these conditions are given in Chapter 2. Most practical on-line optimization applications violate these conditions, but stability problems are not reported in the literature. The dynamic interaction between plant and on-line optimizer dictates the execution frequency at which an on-line optimizer can be executed. This is discussed in Chapter 5.

6.1.5. Influence of Noise on the Optimizer Performance

Measurement and process noise deteriorates the performance of an on-line optimizer. The influence of noise is difficult to predict. Any level of noise reduces the quality of the steady state model. High levels of noise will deactivate the on-line optimizer, since significance tests will find the objective and setpoints too inaccurate and/or will not find any more significant setpoint changes.

6.2. FUTURE RECOMMENDATIONS

The quality of the model is certain in on-line optimization. Therefore, future work has to focus on improving model quality (structure and estimation methods) and data acquisition. Cheaper data reconciliation and gross error

detection algorithms will improve the model as well. Finally, the operation of an on-line optimized plant would profit of the availability a *finite difference* (approximative) sensitivity and stability analysis for nonlinear programs.

APPENDIX A

Short-Cut Boiler Efficiency Model

In this appendix an alternative short-cut model for the steady state relationship between boiler load and boiler efficiency is proposed.

It is industrial standard to make use of a second order polynomial to model boiler efficiency as a function of boiler load. This second order polynomial is of course an empirical or black box model. It makes more physical sense to model $\frac{\text{load}}{\text{efficiency}}$ with a second order polynomial. An approximate modeling of the most important efficiency determining factors and their dependence upon the boiler load makes this clear.

Let the useful output of a steam boiler (the steam leaving the superheater) be equal to the boiler load l. Consider two different type of losses. First, consider the loss through the stack. The amount of flue gas leaving the stack is approximately proportional to the boiler load. Say that the amount of flue gas equals αl. The temperature of the flue gas is approximately proportional to the boiler load l. As the load increases, the mean temperature difference over the heat exchanging surfaces has to increase. Therefore the flue gas temperature in the stack is higher, and the enthalpy loss through the stack can be written as the following function of l :

$$\alpha l(\beta + \gamma l).$$

Second, consider the loss due to radiation. This loss is less dependent on the boiler load. For the sake of simplicity, let the loss because of radiation be equal

to the constant ω. Then :

$$\begin{aligned} \text{efficiency} &= \frac{\text{useful energy output}}{\text{useful} + \text{useless energy output}} \\ &\simeq \frac{l}{l + (\omega + \alpha l(\beta + \gamma l).} \end{aligned} \tag{A.1}$$

This equation can be rewritten as (η = efficiency) :

$$\eta = \frac{l}{a + bl + cl^2}. \tag{A.2}$$

In which a corresponds to ω, b to $1 + \alpha\beta$ and c to $\alpha\gamma$. Figure A.1 compares fits of the empirical model and the new short-cut model to a set of efficiency data for the boiler referred to as # 2 in table 4.3.2. From statistical test, it is clear that the short-cut model fits better than the empirical model. From the figure it is clear that this result is especially true for low boiler loads.

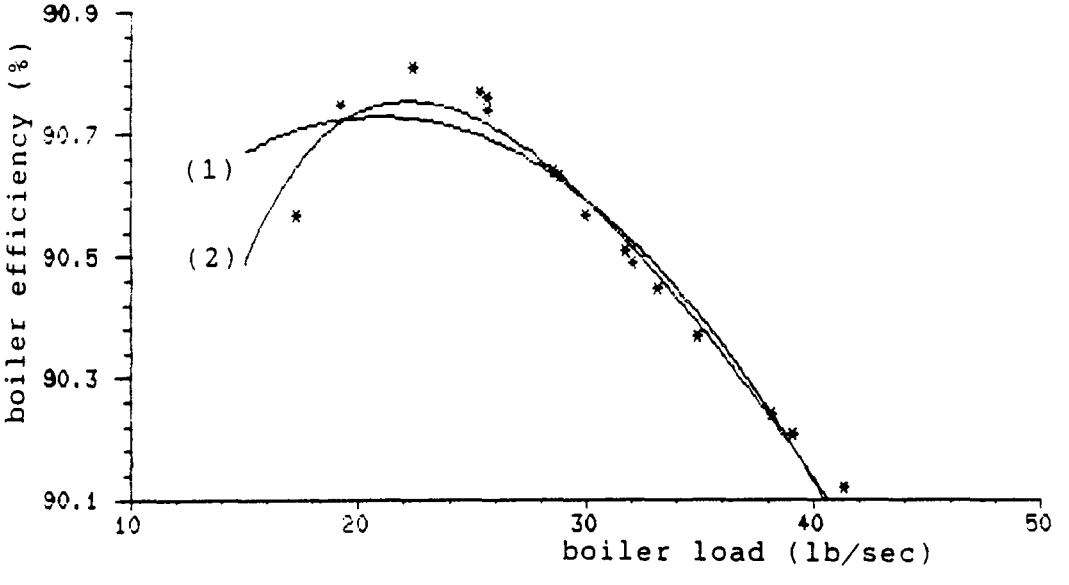

Fig.A.1. Fitting Results of the Empirical Efficiency to Boiler Load Model (1.) to the Short-Cut Method (2.) for Boiler #2 Data

Although the short-cut method is not more complicated than the empirical method, it makes more physical sense and fits better. Nevertheless, the

empirical model (2^{nd} order polynomial) will be used in the case study, because it allows us better to compare the results that are obtained with the system described here with the ones presented in the literature.

APPENDIX B

Parameters in Eduljee's Model are Correlated

One can expect a strong correlation between the parameters. The following lines make this clear.

Consider the equation for the minimum reflux :

$$R_m = \frac{1}{\alpha - 1}\left(\frac{x_D}{x_F} - \frac{1 - x_D}{1 - x_F}\right). \tag{4.2.10}$$

This equation can be rewritten as :

$$R_m = a_i\left(\frac{1}{\alpha - 1}\right) \tag{B.1}$$

with :

$$a_i = \frac{x_{D,i}}{x_{F,i}} - \frac{1 - x_{D,i}}{1 - x_{F,i}}; \tag{5.3.4}$$

in which i is an index that indicates the data record for which a_i is calculated. Similarly we can rewrite Fenske's equation as :

$$N_m = \frac{b_i}{\ln \alpha} \tag{B.2}$$

with :

$$b_i = \ln\left(\frac{x_{D,i}}{1 - x_{D,i}} \cdot \frac{1 - x_{B,i}}{x_{B,i}}\right). \tag{5.3.5}$$

Equations B.1 and B.2 can be substituted in the equation of Eduljee :

$$\frac{N - N_m}{N + 1} = \frac{3}{4}\left(1 - \left(\frac{R - R_m}{R + 1}\right)^{0.5668}\right) \tag{4.2.11}$$

which after rewriting becomes :

$$N\left(\left(\frac{3}{4}\left(1 - \left(1 - \left[\frac{1 + R_m}{1 + R}\right]\right)^{0.5668}\right)\right) - 1\right) =$$

$$\underbrace{\frac{-3}{4}\Big(1-(1-[\frac{1+R_m}{1+R}])^{0.5668}\Big)}_{\text{term 1}}-N_m. \qquad (B.3)$$

Using the values of the nominal case for the propane propylene splitter as given by Martin *et al.* [1981], one finds that the factor indicated "term 1" is equal to 0.4 while N_m equals 60.25. It is therefore fairly accurate to simplify the previous expression to :

$$N\Big((\frac{3}{4}(1-(1-[\frac{1+R_m}{1+R}])^{0.5668}))-1\Big)\simeq -N_m. \qquad (B.4)$$

Substituting B.1 and B.2 in B.3 results in :

$$\frac{b_i}{N}\simeq \ln\alpha\Big(1-\frac{3}{4}(1-(1-[\frac{1+\frac{a_i}{\alpha-1}}{1+R_i}])^{0.5668})\Big). \qquad (B.5)$$

This intermediary result gives an approximate relation between all α's and N's that fit one single operating condition. In the nominal case $a_i = 1.105$, $b_i = 6.016$ and $R = 17.2$. This equation becomes :

$$\frac{6.016}{N}\simeq \ln\alpha\Big(1-\frac{3}{4}(1-(1-[\frac{1+\frac{1.105}{\alpha-1}}{18.2}])^{0.5668})\Big). \qquad (5.3.6)$$

Compare two factors in the right hand side of equation B.5.

$$\begin{cases} A=\ln\alpha \\ B=\Big(1-(1-\frac{1+\frac{a_i}{\alpha-1}}{1+R_i})^{0.5668}\Big). \end{cases} \qquad (B.6)$$

For the nominal case ($\alpha = 1.105$), $A = 0.0998$ and $B = 0.9502$. However, more important are their derivatives with respect to α. These are follows for the nominal case :

$$\begin{cases} \frac{dA}{d\alpha}=9.92 \\ \frac{dB}{d\alpha}=0.02. \end{cases} \qquad (B.7)$$

In other words, the B factor is almost constant compared to A as a function of α. Define c_i for every data set :

$$c_i=\Big(1-(1-\frac{1+\frac{a_i}{\alpha-1}}{1+R_i})^{0.5668}\Big)\simeq \text{constant with respect} \rightarrow \text{ changes in } \alpha \text{ ;} \qquad (5.3.3)$$

and hence :

$$\frac{1}{N} \simeq \frac{1. - \frac{3}{4}c_i}{b_i} \ln \alpha. \tag{5.3.2}$$

For databases that contain consistent data (data from the same distillation column, a column that meets the requirements to be modeled by the Eduljee and Fenske equations) these c_i's will be relatively constant. b_i can be expected to vary only over a narrow range. In all plant data it can be expected that $x_{D,i}$ is close to 0.95, and also $x_{B,i}$ is not expected to vary over a wide range. This means that significant correlation can be expected between the parameter estimates for N and for α.

APPENDIX C

Estimating in a Transformed Space

In the previous appendix, it was shown that Eduljee's equation could be written as :

$$N\Big((\frac{3}{4}(1-(1-[\frac{1+R_m}{1+R}])^{0.5668}))-1\Big) =$$

$$\underbrace{\frac{-3}{4}\Big(1-(1-[\frac{1+R_m}{1+R}])^{0.5668}\Big)}_{\text{term 1}} - N_m \qquad (B.3)$$

It was also shown that "term 1" was neglectable compared to N_m, and using :

$$a_i = \frac{x_{D,i}}{x_{F,i}} - \frac{1-x_{D,i}}{1-x_{F,i}} \qquad (5.3.4)$$

and,

$$b_i = \ln\big(\frac{x_{D,i}}{1-x_{D,i}} \cdot \frac{1-x_{B,i}}{x_{B,i}}\big) \qquad (5.3.5)$$

the Eduljee equation appears as

$$\frac{b_i}{N} - \ln\alpha\Big(1-\frac{3}{4}(1-(1-[\frac{1+\frac{a_i}{\alpha-1}}{1+R_i}])^{0.5668})\Big) \simeq 0 \qquad (B.5)$$

It appeared that the second factor of the second term at nominal conditions only accounts for approximately two percent of the variation in α (see Appendix B), and therefore equation B.5 shows that the Eduljee equation is almost linear in $\ln\alpha$, N^{-1}. If the curves on figure 5.3.1 is plotted with respect to these transformed variables, almost a straight line is obtained. This is shown in figure 5.3.2.

The fourth estimation technique proposed here, operates in the $(\ln\alpha,\ N^{-1})$ plane. Using the above definitions of a_i for every data point, another factor c_i is defined as before :

$$c_i = \left(1 - \left(1 - \frac{1 + \frac{a_i}{\alpha - 1}}{1 + R_i}\right)^{0.5668}\right). \tag{5.3.3}$$

Using this definition, the Eduljee equation can be rewritten, without neglecting the term that was shown to be small in expression B.5.

$$\frac{1}{N} - \frac{1 - \frac{3}{4}c_i}{b_i + c_i \ln\alpha} \cdot \ln\alpha \simeq 0 \tag{C.1}$$

The equation will only be exactly equal to zero for data sets i for which the model fits exactly. This is in general not the case. It is clear that Fenske's and Eduljee's equations are almost linear in $\ln\alpha$ and $\frac{1}{N}$.

APPENDIX D

The MIMT Data Reconciliation Algorithm Applied to the Distillation Case Study

In this appendix, the MIMT data reconciliation algorithm is presented. This method is presented by Serth *et al.* [1986], and is slightly adjusted for the case under consideration. The equations used by the reconciler are the mass balances over the column :

$$F = D + B; \tag{4.2.13}$$

$$Fx_F = Dx_D + Bx_B. \tag{4.2.14}$$

All variables are partitioned in two classes. All flows are collected in class I, the compositions in class II.

STEP 1

When a dataset is processed by the data reconciliation module, then an option is given to enter a status for every measurement with the raw data. This information can be provided by the operator, or automatically e.g. by the transmitter. A measurement can be a priori unreliable due to maintenance of the transmitter, saturation etc. If one measurement is a priori unreliable, then proceed with step 4, if two measurements are unreliable, then proceed with step 5. If more than two measurements have the a priori unreliable status, then proceed with step 6. If no measurements are a priori unreliable, the proceed with step 2.

STEP 2

The optimal correction is calculated. Therefore the sum of squares of the corrections divided by the square root of the variance is minimized subject to the reconciliation equations. If the variances are unknown, the the absolute measurement value (if not zero) will often replace the square root of the variance as an alternative scaling. For measurements that are linearly related the technique for the estimation of the correction variances from process data presented by Almasy and Mah [1984] can be considered. The least square problem is solved with a Marquardt-Levenspiel (CMLIB) routine. Note that the second reconciliation equation is bilinear in the reconciled variables. An analytical solution to the least squares problem is not available.

STEP 3

Compute for every correction e_j, $z_j = e_j(v_{jj})^{-\frac{1}{2}}$ with :

$$V = QA^T(AQA^T)^{-1}AQ. \qquad (D.1)$$

The derivation of equation $D.1$ is similar to the derivation of equation 3.3.3 in section 3.3.3.2.1., with :

$$A = \begin{pmatrix} 1 & -1 & -1 & 0 & 0 & 0 \\ x_{F,nom} & -x_{D,nom} & -x_{B,nom} & F_{nom} & -D_{nom} & -B_{nom} \end{pmatrix}, \qquad (D.2)$$

and with Q the covariance matrix of the measurement corrections. A is is the reconciliation equation matrix linearized around the nominal conditions. Q is assumed to be diagonal in this case. Find the measurement for which $|z_j|$ is largest. If $|z_j| \leq 2.63$ then the reconciled data only contain random errors. The reconciliation is then completed. If $|z_j| > 2.63$ than at least this measurement j is unreliable. It contains a systematic or gross error. The correction is too large to be due to noise.

Serth *et al.* [1986] list as limit value of $|z_j|$: $z_{1-\beta/2}$ or the $(1-\frac{\beta}{2})$ value of the normal distribution. Furthermore :

$$\beta = 1 - (1-\psi)^{\frac{1}{n}}, \qquad (3.3.5)$$

with n the number of measurements involved in the reconciliation (6), and ψ the confidence limit. In this case $\psi = 0.05$ was used. It is more conservative to use $\psi = 0.01$, and that is often applied as well. For $n = 6$ and $\psi = 0.05$: $\beta = 8.5 \cdot 10^{-3}$, and $z_{1-\beta/2} = 2.63$.

STEP 4

If at least one measurement is unreliable, then this measurement can be reconstructed using the other measurements and the reconciliation equations. Therefore the remaining variables of the same class and the corresponding reconciliation equation have to be used. The other class of measurements can be reconciled for random errors with the complimentary reconciliation equation.

Note that depending upon whether the suspected measurement is a flow or a composition, the order of the reconstruction and the reconciliation has to be reversed. If the unreliable measurement is a flow, then this flow has to be reconstructed from the other flows. After the reconstruction is completed, the purities can be reconciled. If the unreliable measurement is a composition, then first the "correct" streams have to be obtained through reconciliation of the flows with the global mass balance. Afterwards the unreliable purity can be reconstructed using the other compositions and the reconciled streams.

A check analogous to the one in step 3 is performed to determine whether the corrections to the remaining variables that can be reconciled are or are not too large to be random. If the reconciliation of the reduced set of variables also results in at least one unreliable measurement, then continue to step 5. Else, if no other measurement is unreliable, then the process is ended. Part of the result is then obtained through reconciliation, and the other part through reconstruction.

STEP 5

Two variables are considered unreliable. If both measurements belong to the same type (both flows or both compositions) then it is impossible to reconstruct these measurements because not enough reliable information is available. If two flows are unreliable, then the compositions can still be used as raw data. However, they cannot be reconciled. Another alternative is that the dataset is removed.

If both variables belong to a different type, then they can still be reconstructed using the remaining data, and the reconciliation equations. However, the reliability of these results remain an open question.

STEP 6

If more than two measurement are unreliable, then the dataset is useless.

APPENDIX E

Generalized Likelihood Ratio Method for Gross Error Identification

This appendix summarizes the algorithm for gross error detection that was proposed by Narasimhan and Mah [1987]. This method is able to detect multiple gross errors that do not interfere with each other. The method is applicable to a set of linear reconciliation equations. Let a vector of raw measurements be represented by z, a $n \times 1$ matrix. Furthermore, let x be a $n \times 1$ matrix of the true values of the measurements, a let v, also a $n \times 1$ matrix, be a vector of measurement errors. Then :

$$z = x + v, \tag{E.1}$$

and :

$$Ax = 0. \tag{E.2}$$

with A the $m \times n$ matrix of linear reconciliation equation coefficients. An important assumption is that the measurement errors v are normally distributed with zero mean and with a known error covariance matrix Q. Section 3.3.5.1 discusses the problems related to obtaining the error covariance matrix.

Two types of systematic or gross errors can be modeled in this algorithm. These two types are measurement biases and node leaks. A measurement bias of magnitude b in measurement i will add a term to equation $E.1$:

$$z = x + v + be_i. \tag{E.3}$$

In this equation, e_i is the unit vector along the ith coordinate axis. A leak of

magnitude b in node j can be modeled by adding a term to equation $E.2$:

$$Ax - bm_j = 0. \tag{E.4}$$

In this last equation, m_j is a vector with zero for equations that are not affected by the leak, and proportional terms for equations that are offset by the leaking node. A practical example makes this clear in section 5.2.2.

Initially, the matrices A and Q are assumed to be known, and the vectors e_i and m_j have to be provided for every leak or bias that one wants to check. The algorithm will attempt iteratively to determine the magnitudes of the gross errors that can be detected in the data set that is presented to it.

Define matrices $E_k^\star$, $M_k^\star$ and $G_k^\star$ as follows :

$$E_k^\star = [e_1^\star, e_2^\star, \ldots, e_k^\star]; \tag{E.5}$$

with :

$$e_i^\star = \begin{cases} 0 & \text{if less than } i \text{ gross errors} \\ & \text{are detected yet} \\ & \text{or the } i\text{th gross error} \\ & \text{is not a bias;} \\ e_j & \text{if the } i\text{th bias detected} \\ & \text{was in measurement } j; \end{cases} \tag{E.6}$$

$$M_k^\star = [m_1^\star, m_2^\star, \ldots, m_k^\star]; \tag{E.7}$$

with :

$$m_i^\star = \begin{cases} 0 & \text{if less than } i \text{ gross errors} \\ & \text{are detected yet} \\ & \text{or the } i\text{th gross error} \\ & \text{is not a bias;} \\ m_j & \text{if the } i\text{th leak detected} \\ & \text{was in measurement } j; \end{cases} \tag{E.8}$$

and

$$G_k^\star = AE_k^\star + M_k^\star. \tag{E.9}$$

Finally, let

$$B^\star = \left[b_1^\star, b_2^\star, \ldots, b_k^\star\right], \tag{E.10}$$

be the vector of magnitudes of the systematic errors.

STEP 1

Initially, $E_k^\star$, $M_k^\star$ and $G_k^\star$ are set equal to zero. The number of biases and the number of leaks that is detected is set equal to zero. Set $k = 0$.

STEP 2

Let

$$z_k = z - E_k^\star b_k^\star, \tag{E.11}$$

be the compensated measurements. That means that z_k contains measurements that are corrected for the measurement errors that are detected already. Similarly, let

$$\begin{aligned} r_k =& Az_k - M_k^\star B_k^\star \\ =& r - G_k^\star B_k^\star; \end{aligned} \tag{E.12}$$

with

$$r = Az. \tag{E.13}$$

The vector r_k represents the compensated residuals, that is the residuals after correction of the already detected gross errors. Of course, during the first pass of this iterative procedure $z_k = z$ and $r_k = r$.

STEP 3

In this step, a maximum likelihood estimate of the magnitudes of the remaining candidate gross errors is calculated. With "remaining candidate gross error" is meant a candidate gross error that was not confirmed as an actual gross error yet. The number of remaining gross error candidates equals the

total number of gross errors that is modeled minus the number of gross errors that is already detected (k).

If no gross errors are left, in other words all gross errors are detected in previous passes, or no gross errors are present if this is the first pass, then the expected value of r_k is equal to 0 :

$$E(r_k) = 0. \tag{E.14}$$

If a gross error is due to a bias of magnitude b in a measurement i, then :

$$E(r_k) = bAe_i. \tag{E.15}$$

If on the other hand, a gross error is due to a node leak, then :

$$E(r_k) = bm_j. \tag{E.16}$$

Equations $E.15$ and $E.16$ can be combined in equation $E.18$ by introducing f_i in the next equation :

$$f_i = \begin{cases} Ae_i & \text{for a bias in measurement } i \\ m_j & \text{for a leak in node } j. \end{cases} \tag{E.17}$$

$$E(r_k) = bf_i \tag{E.18}$$

In order to obtain a maximum likelihood estimate of b for every remaining candidate, it is necessary to have the covariance matrix for the *compensated* residuals. If Q is known, then the residual covariance matrix is given by :

$$V = AQA^T + Uz^TA^T + AzU^T + UU^T. \tag{E.19}$$

In this equations U is defined as :

$$U = r_k - r. \tag{E.20}$$

Then the maximum likelihood estimate of b, the magnitude of a remaining candidate gross error, is given as :

$$\hat{b} = \frac{d_i}{c_i}; \qquad (E.21)$$

with :

$$c_i = f_i^T V^{-1} f_i; \qquad (E.21)$$

$$d_i = f_i^T V^{-1} r. \qquad (E.22)$$

STEP 4

In this step, the most likely gross error is identified among the remaining gross errors. Then, it is checked whether or not the most likely $E.18$ hypothesis is more or less likely than the $E.14$ hypothesis. In other words, it is checked whether or not the most likely gross error is a gross error at all. To test whether or not either the hypothesis expressed by $E.14$ or one of the hypotheses expressed by $E.18$ holds, a generalized likelihood ratio method is applied. For more details on the generalized likelihood ratio, see e.g. Hogg and Tanis [1983]. In this case, it is sufficient to calculate

$$T_i = \frac{d_i^2}{c_i}, \qquad (E.23)$$

for every remaining candidate gross error. Let T be the largest T_i, and let $i_{\max}$ be the corresponding gross error. Then $i_{\max}$th gross error is the most likely gross error among the remaining candidate gross errors.

It can be shown (Narasimhan and Mah [1987]) that T_i has a χ^2 distribution with one degree of freedom if $E.14$ holds. The distribution of T is unknown, since the various T_i's are not independent. However, Narasimhan and Mah [1987] determine that T can be compared with a critical value that is equal to $\chi^2_{1,1-\beta}$, the upper $1-\beta$ quantile of the χ^2 distribution with one degree of

freedom. For a level of significance α (for the results in Chapter 5, $\alpha = 99\%$), $\beta = 1-(1-\alpha)^{\frac{1}{p}}$ with p the number of remaining candidate gross errors for this pass through the loop.

STEP 5

If $E.14$ holds, then all gross errors are detected, and the iterative procedure is complete. If this is so during the first pass, then no gross errors are detected in the data set. If on the other hand, one of the hypothesis represented by $E.18$ holds, then the procedure goes back to **STEP 2**, after including the newly found gross error to the list of already detected gross errors. Increase k by one.

It is worth noting that the accuracy of the algorithm is increased significantly if all reconciliation equations are scaled to the same order of magnitude.

APPENDIX F

Optimal Data Base Length

In this appendix, equation 5.5.4 is derived.

Let s be a parameter, and let $\hat{s}$ be the estimate of that parameter. In general, $\hat{s}$ will not be equal to s, instead there will be a difference between the true and the estimated value. Assume that $\hat{s}$ is distributed normally with mean μ_s and variance σ_s^2. Note that μ_s is not necessary equal to s. The objective of this derivation is to minimize the expected value of the absolute value of $\hat{s} - s$, i.e. $E(|\hat{s} - s|)$. Let $\delta s = \hat{s} - s$. Then $E(\delta s) = \mu_s - s$. Let $\mu_s - s = \Delta s$.

The frequency distribution of $\hat{s}$ is given by :

$$f(\hat{s}) = \frac{1}{\sqrt{2\pi}\sigma_s} \exp\Big(- \frac{(\hat{s} - \mu_s)^2}{2\sigma_s^2}\Big). \tag{F.1}$$

This can be expressed in terms of δs :

$$f(\delta s) = \frac{1}{\sqrt{2\pi}\sigma_s} \exp\Big(- \frac{(\delta s + s - \mu_s)^2}{2\sigma_s^2}\Big); \tag{F.2}$$

or

$$f(\delta s) = \frac{1}{\sqrt{2\pi}\sigma_s} \exp\Big(- \frac{(\delta s - \Delta s)^2}{2\sigma_s^2}\Big). \tag{F.3}$$

Then :

$$\begin{aligned} f(|\delta s|) &= f(\delta s) + f(-\delta s) \\ &= \frac{1}{\sqrt{2\pi}\sigma_s}\Big[\exp\Big(- \frac{(\delta s - \Delta s)^2}{2\sigma_s^2}\Big) + \exp\Big(- \frac{(\delta s + \Delta s)^2}{2\sigma_s^2}\Big)\Big]. \end{aligned} \tag{F.4}$$

The expected value of this expression can be obtained through integration :

$$E(|\delta s|) = \int_0^{+\infty} |\delta s| f(|\delta s|) d|\delta s|. \tag{F.5}$$

It can be shown that :

$$\int_0^{+\infty} x \exp\Big(-\frac{(x-\mu)^2}{2\sigma^2}\Big)dx = \sigma^2 \exp\big(\frac{-\mu^2}{2\sigma^2}\big) + \mu\sigma\sqrt{2\pi}\big(1 - P(-\frac{\mu}{\sigma})\big). \tag{F.6}$$

with :

$$P(-\frac{\mu}{\sigma}) = \frac{1}{\sqrt{2\pi}} \int_{-\frac{\mu}{\sigma}}^{+\infty} \exp\big(-\frac{x^2}{2}\big)dx. \tag{F.7}$$

If that result is applied to equation $F.5$, then it follows that :

$$E(|\delta s|) = \frac{\sigma_s}{\sqrt{2\pi}}\left(\exp\Big(-\frac{\Delta s^2}{2\sigma_s^2}\Big) + \exp\Big(\frac{\Delta s^2}{2\sigma^2}\Big)\right) + \Delta s\big(P(\frac{\Delta s}{\sigma_s}) - P(-\frac{\Delta s}{\sigma_s})\big). \tag{F.8}$$

This expression has to be minimized. For n points in the data base (or a data base of length n, $\Delta s = \frac{n-1}{2}\Delta\varphi$, and $\sigma_s = \frac{\sigma_\varphi}{\sqrt{n}}$. This is so because $\Delta\varphi$ is the change in φ per time unit, and σ_φ is the variance for the estimate of the parameter φ if only one data point is used. The values for $\Delta\varphi$ and σ_φ are constant. If these definitions are inserted in equation $F.8$, equation $F.9$ is obtained.

$$E(|\delta s|) = \sqrt{\frac{2}{\pi n}}\sigma_\varphi \cosh\Big(\frac{(n-1)^2\Delta\varphi^2\sqrt{n}}{8\sigma_\varphi^2}\Big) + \frac{n-1}{2}\Delta\varphi\left(P\Big(\frac{(n-1)\Delta\varphi\sqrt{n}}{2\sigma_\varphi}\Big) - P\Big(-\frac{(n-1)\Delta\varphi\sqrt{n}}{2\sigma_\varphi}\Big)\right). \tag{F.9}$$

Expression $F.9$ has to be minimized with respect to the *integer* n. This is equivalent to minimizing

$$E\Big(\frac{|\delta s|}{\Delta\varphi}\Big) = \sqrt{\frac{2}{\pi n}}\frac{\sigma_\varphi}{\Delta\varphi}\cosh\Big(\frac{(n-1)^2\Delta\varphi^2\sqrt{n}}{8\sigma_\varphi^2}\Big) + \frac{n-1}{2}\left(P\Big(\frac{(n-1)\Delta\varphi\sqrt{n}}{2\sigma_\varphi}\Big) - P\Big(-\frac{(n-1)\Delta\varphi\sqrt{n}}{2\sigma_\varphi}\Big)\right). \tag{F.10}$$

After introducing

$$D = \frac{\Delta\varphi}{\sigma_\varphi} : \tag{5.5.5}$$

the right hand side of expression $F.10$ becomes 5.5.4 :

$$\sqrt{\frac{2}{\pi n}} \cdot \frac{1}{D} \cdot \cosh\Big(\frac{(n-1)^2 n^{\frac{1}{2}} D^2}{8}\Big) + \frac{n-1}{2} \cdot \Big(P\big(\frac{(n-1)\sqrt{n}}{2} \cdot D\big) - P\big(-\frac{(n-1)\sqrt{n}}{2} \cdot D\big)\Big). \tag{5.5.4}$$

Figure F.1 represents expression 5.5.4 for different values of n, for $D = 0.1$.

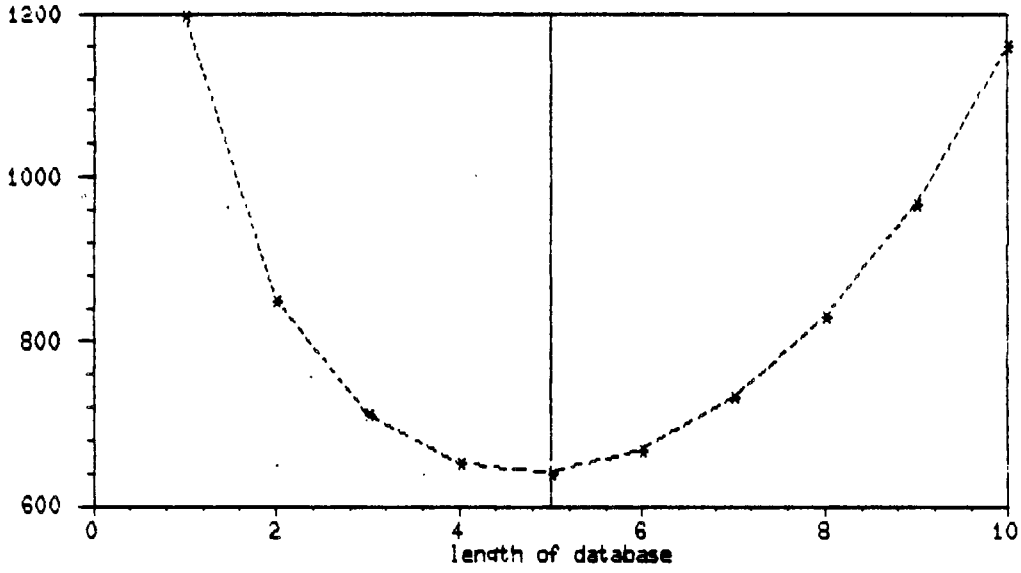

Fig.F.1. Data Base Length Criterion as a Function of Data Base Length for a Normalized Linear Drift of 0.1.

APPENDIX G

Simulation Run Specifics

Remarks

- In many simulations, some form of noise was added to simulation inputs and/or outputs. Therefore, numbers were randomly generated. A routine called URN14 is used to generate (single precision) random numbers with uniform distribution between 0 and 1. The used routine is a FORTRAN translation of part of the IRAND package of random generators developed by Dr. R.L. Zarling in 1971 at Mariette College, Mariette, Ohio (Dudewitz and Ralley [1981]). The translation was provided by Tina L. Vital. Normally distributed numbers (standardized : zero mean and unit variance) are obtained from the uniformly distributed numbers by applying a Box-Muller transformation (Hogg and Tanis, p.261, [1983]).
- For certain simulation examples, more information is available on computertape. This information is organized in various directories. Filenames (with directory specification) are given below.

RUN # 1

- Case Study: Propane/Propylene Splitter.
- Subject or Objective: Effect of Data Reconciliation on Model Quality.
- "Plant" Model: Eduljee and Fenske (Nominal Parameter Values see Table 4.2.1).
- Optimizer Model: Eduljee and Fenske.
- Noise: Gaussian, Standard Deviation is 5% of nominal value (Table 4.2.2.)

for F, D, B and x_B, 5% of one minus the nominal value for x_D and x_F.

- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: The simulation input variables feed mass flow, feed composition, top composition and reflux ratio are perturbed normally around the nominal values (Table 4.2.2) with standard deviation of respectively 150000, 0.03, 0.03, 1.12. One hundred data points are used, and are given in file [RUN]RUN123_C3C3DAT.DAT.
- ◇ Steady State Detection: N/A(steady state study)
- ◇ Data Reconciliation Algorithm: MIMT
- ◇ Gross Error Detection Algorithm: MIMT, gross errors rejected.
- ◇ Model Parameter Updating: $\ln \alpha$, $1/N$; fixed data base length (using Marquardt Levenspiel).
- ◇ Outlier Detection: 99% Confidence Test on Residuals
- ◇ Steady State Data Base Length: 30
- ◇ Check Significantly Different New Data: N/A
- ◇ Optimization Algorithm: N/A
- ◇ First Order Sensitivity Objective: N/A
- ◇ Second Order Sensitivity Objective: N/A
- ◇ First Order Sensitivity Setpoints: N/A
- ◇ Check for Suff. Accuracy Objective: N/A
- ◇ Check for Sign. Difference Setpoints: N/A
- ◇ Implementation of Setpoints: N/A

RUN # 2

- ◇ Case Study: Propane/Propylene Splitter.
- ◇ Subject or Objective: Effect of Data Reconciliation on Model Quality – Negative Effect.
- ◇ "Plant" Model: Eduljee and Fenske (Nominal Parameter Values see Table

4.2.1).

- ◇ Optimizer Model: Eduljee and Fenske.
- ◇ Noise: Gaussian, Standard Deviation is 5% of nominal value (Table 4.2.2.) for F, D, B and x_B, 5 % of one minus the nominal value for x_D and x_F.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: The simulation input variables feed mass flow, feed composition, top composition and reflux ratio are perturbed normally around the nominal values (Table 4.2.2) with standard deviation of respectively 150000, 0.03, 0.03, 1.12. One hundred data points are used. Now in half of the data points a 20% error is introduced in the feed mass flow data. The data file has not been stored.
- ◇ Steady State Detection: N/A(steady state study)
- ◇ Data Reconciliation Algorithm: MIMT
- ◇ Gross Error Detection Algorithm: MIMT, gross errors rejected.
- ◇ Model Parameter Updating: $\ln \alpha$, $1/N$; fixed data base length (using Marquardt Levenspiel).
- ◇ Outlier Detection: 99% Confidence Test on Residuals
- ◇ Steady State Data Base Length: 30
- ◇ Check Significantly Different New Data: N/A
- ◇ Optimization Algorithm: N/A
- ◇ First Order Sensitivity Objective: N/A
- ◇ Second Order Sensitivity Objective: N/A
- ◇ First Order Sensitivity Setpoints: N/A
- ◇ Check for Suff. Accuracy Objective: N/A
- ◇ Check for Sign. Difference Setpoints: N/A
- ◇ Implementation of Setpoints: N/A

RUN # 3

- ◇ Case Study: Propane/Propylene Splitter.
- ◇ Subject or Objective: Effect of Data Reconciliation on Model Quality – Positive Effect for Failing Sensor.
- ◇ "Plant" Model: Eduljee and Fenske (Nominal Parameter Values see Table 4.2.1).
- ◇ Optimizer Model: Eduljee and Fenske.
- ◇ Noise: Gaussian, Standard Deviation is 5% of nominal value (Table 4.2.2.) for F, D and B, 5 % of one minus the nominal value for x_D and x_F. In one quarter of the data a very high value is introduced for the bottom composition (more than 0.7). For the other cases a 12% standard deviation from the nominal value is used for the bottom composition.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: The simulation input variables feed mass flow, feed composition, top composition and reflux ratio are perturbed normally around the nominal values (Table 4.2.2) with standard deviation of respectively 150000, 0.03, 0.03, 1.12. One hundred data points are used.
- ◇ Steady State Detection: N/A(steady state study)
- ◇ Data Reconciliation Algorithm: MIMT
- ◇ Gross Error Detection Algorithm: MIMT, gross errors rejected.
- ◇ Model Parameter Updating: $\ln \alpha$, $1/N$; fixed data base length (using Marquardt Levenspiel).
- ◇ Outlier Detection: 99% Confidence Test on Residuals
- ◇ Steady State Data Base Length: 30
- ◇ Check Significantly Different New Data: N/A
- ◇ Optimization Algorithm: N/A
- ◇ First Order Sensitivity Objective: N/A
- ◇ Second Order Sensitivity Objective: N/A

- ◇ First Order Sensitivity Setpoints: *N/A*
- ◇ Check for Suff. Accuracy Objective: *N/A*
- ◇ Check for Sign. Difference Setpoints: *N/A*
- ◇ Implementation of Setpoints: *N/A*

RUN # 4

- ◇ Case Study: Heat Exchanger
- ◇ Subject or Objective: Effect of Partitioning on Gross Error Identification Performance.
- ◇ "Plant" Model: as specified in 4.4.4.2.
- ◇ Optimizer Model: /na
- ◇ Noise: in all cases specified per example
- ◇ Bias or Other Systematic Errors: in all cases specified per example
- ◇ Data Base: one specific datapoint (noise is added)
- ◇ Steady State Detection: *N/A*(steady state study)
- ◇ Data Reconciliation Algorithm: none
- ◇ Gross Error Detection Algorithm: Narasimhan and Mah [1987].
- ◇ Model Parameter Updating: *N/A*
- ◇ Outlier Detection: *N/A*
- ◇ Steady State Data Base Length: *N/A*
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: *N/A*
- ◇ First Order Sensitivity Objective: *N/A*
- ◇ Second Order Sensitivity Objective: *N/A*
- ◇ First Order Sensitivity Setpoints: *N/A*
- ◇ Check for Suff. Accuracy Objective: *N/A*
- ◇ Check for Sign. Difference Setpoints: *N/A*
- ◇ Implementation of Setpoints: *N/A*

RUN # 5

- ◇ Case Study: Propane/Propylene Splitter
- ◇ Subject or Objective: Influence on Model Updating Problems on Optimizer Performance.
- ◇ "Plant" Model: Eduljee and Fenske with Nominal Parameters as in Table 4.2.1.
- ◇ Optimizer Model: Eduljee and Fenske
- ◇ Noise: see Run # 1
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: 16 Datapoints Chosen Randomly From the Data Base used in Run # 1.
- ◇ Steady State Detection: *N/A*(steady state study)
- ◇ Data Reconciliation Algorithm: MIMT
- ◇ Gross Error Detection Algorithm: MIMT, gross errors rejected.
- ◇ Model Parameter Updating: Two model updating methods are compared with each other here. The first method estimates in the α, N space. The second method estimates in the $\ln\alpha$, $1/N$ space. Both use Marquardt Levenspiel.
- ◇ Outlier Detection: 99% Confidence Test on Residuals
- ◇ Steady State Data Base Length: fixed, 16.
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: Rosenbrock Hillclimb
- ◇ First Order Sensitivity Objective: No
- ◇ Second Order Sensitivity Objective: No
- ◇ First Order Sensitivity Setpoints: No
- ◇ Check for Suff. Accuracy Objective: *N/A*
- ◇ Check for Sign. Difference Setpoints: *N/A*

◇ Implementation of Setpoints: *N/A*

RUN # 6

◇ Case Study: Propane Propylene Splitter

◇ Subject or Objective: Influence on Model Updating Problems on Optimizer Performance.

◇ "Plant" Model: Underwood.

◇ Optimizer Model: Fenske and Eduljee.

◇ Noise: None.

◇ Bias or Other Systematic Errors: None.

◇ Data Spread: Input for the simulation is obtained from a grid as specified in Table 5.3.1. Ninety points were used.

◇ Steady State Detection: *N/A*(steady state study)

◇ Data Reconciliation Algorithm: None

◇ Gross Error Detection Algorithm: None

◇ Model Parameter Updating: Least squares (using Marquardt Levenspiel).

◇ Outlier Detection: 99% Confidence Test on Residuals

◇ Steady State Data Base Length: All 90 points were used.

◇ Check Significantly Different New Data: *N/A*

◇ Optimization Algorithm: Rosenbrock Hillclimb

◇ First Order Sensitivity Objective: No

◇ Second Order Sensitivity Objective: No

◇ First Order Sensitivity Setpoints: No

◇ Check for Suff. Accuracy Objective: *N/A*

◇ Check for Sign. Difference Setpoints: *N/A*

◇ Implementation of Setpoints: *N/A*

RUN # 7

◇ Case Study: Propane Propylene Splitter

- ◇ Subject or Objective: "Inverse Response".
- ◇ "Plant" Model: Eduljee and Fenske (the first thirty data sets with nominal model parameters (see Table 4.2.2), the latter thirty data sets with $\alpha = 1.110$ and $N = 99$.)
- ◇ Optimizer Model: Eduljee and Fenske
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Spread: Input Data File is stored in [RUN].TRCK5_DAT.DAT.
- ◇ Steady State Detection: N/A(steady state study)
- ◇ Data Reconciliation Algorithm: MIMT
- ◇ Gross Error Detection Algorithm: MIMT, gross errors rejected.
- ◇ Model Parameter Updating: $\ln \alpha$, $1/N$; fixed data base length (using Marquardt Levenspiel).
- ◇ Steady State Data Base Length: fixed, 30.
- ◇ Outlier Detection: 99% Confidence Test on Residuals
- ◇ Check Significantly Different New Data: No
- ◇ Optimization Algorithm: Rosenbrock Hillclimb
- ◇ First Order Sensitivity Objective: No
- ◇ Second Order Sensitivity Objective: No
- ◇ First Order Sensitivity Setpoints: No
- ◇ Check for Suff. Accuracy Objective: N/A
- ◇ Check for Sign. Difference Setpoints: N/A
- ◇ Implementation of Setpoints: N/A

RUN # 8

- ◇ Case Study: Propane Propylene Splitter
- ◇ Subject or Objective: Residual Distribution
- ◇ "Plant" Model: Eduljee and Fenske (with nominal model parameters as

shown in Table 4.2.1).

- ◇ Optimizer Model: Eduljee and Fenske
- ◇ Noise: See Run # 1
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: See Run # 1
- ◇ Steady State Detection: N/A(steady state study)
- ◇ Data Reconciliation Algorithm: MIMT
- ◇ Gross Error Detection Algorithm: MIMT
- ◇ Model Parameter Updating: $\ln \alpha$, $1/N$ using Marquardt Levenspiel
- ◇ Outlier Detection: 99% Confidence Test on Residuals
- ◇ Steady State Data Base Length: 100 (All Points Used At Once).
- ◇ Check Significantly Different New Data: N/A
- ◇ Optimization Algorithm: N/A
- ◇ First Order Sensitivity Objective: N/A
- ◇ Second Order Sensitivity Objective: N/A
- ◇ First Order Sensitivity Setpoints: N/A
- ◇ Check for Suff. Accuracy Objective: N/A
- ◇ Check for Sign. Difference Setpoints: N/A
- ◇ Implementation of Setpoints: N/A

RUN # 9

- ◇ Case Study: Boiler Case Study
- ◇ Subject or Objective: Correlation Between Parameter Estimates
- ◇ "Plant" Model: Rigorous Simulation (Bertrand [1986])
- ◇ Optimizer Model: Second Order Polynamial.
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: These data are extracted from the steam simulation after ob-

taining steady state with a total network load of 72.8 lb/s and with individual load allocations of 25.5, 30.3 and 17.0 lb/s for boilers 1, 2 and 3 respectively. The optimization database for boiler 1 contains the following load to efficiency couples (load in lb/s) : (19.174, 90.87), (22.355, 90.86), (25.556, 90.76), (28.753, 90.63), (31.955, 90.49). For boiler 2 : (19.170, 91.37), (22.356, 91.42), (25.553, 91.42), (28.750, 91.43), (31.945, 91.23). For boiler 3 : (19.174, 90.22), (22.370, 90.15), (25.566, 90.02), (28.758, 89.88), (31.953,89.72). These efficiency data are initially put in the steady state data base, and are obtained off-line. These data are therefore quite reliable.

- ◇ Steady State Detection: Band Test
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: One Step Least SQ
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: fixed, 5
- ◇ Check Significantly Different New Data: Yes
- ◇ Optimization Algorithm: SQP
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: Yes
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^*_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 10

- ◇ Case Study: Boiler Case Study

- ◇ Subject or Objective: Correlation Between Parameter Estimates
- ◇ "Plant" Model: Rigorous Simulation (Bertrand [1986])
- ◇ Optimizer Model: Second Order Polynamial.
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: The initial data are given in the specifications of run # 9. The initial data are not given in figure 5.3.16. Instead, the true efficiency is plotted. The data for the second fit for boiler 2 are as follows : (32.478, 91.21), (22.356, 91.42), (25.553, 91.42), (28.750, 91.43), (31.945, 91.23). Only the first point has been replaced. For the third fit, the second data set changes : (32.478, 91.21), (32.680, 91.20), (25.553, 91.42), (28.750, 91.43), (31.945, 91.23). It is true that in the example with boiler two, the poorer fit can only be due to a poor representation of low load data sets in the steady state data base. An error resulting from early steady state detection can be found in the boiler 1 steady state data base, were a combination of load equal to 19 lb/sec with an efficiency of 90.85 was recorded. "Fit 2" is recorded after 60 minutes simulated time, "Fit 3" is recorded after 100 minutes simulated time.
- ◇ Steady State Detection: Band Test
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: One Step Least SQ
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: fixed, 5
- ◇ Check Significantly Different New Data: Yes
- ◇ Optimization Algorithm: SQP
- ◇ First Order Sensitivity Objective: Yes

- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: Yes
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^{\star}_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 11

- ◇ Case Study: Propane Propylene Splitter
- ◇ Subject or Objective: Sensitivity Results
- ◇ "Plant" Model: *N/A*
- ◇ Optimizer Model: Eduljee and Fenske
- ◇ Noise: *N/A*
- ◇ Bias or Other Systematic Errors: *N/A*
- ◇ Data Base: Nominal Conditions (see Tables 4.2.1–2).
- ◇ Steady State Detection: *N/A*(steady state study)
- ◇ Data Reconciliation Algorithm: *N/A*
- ◇ Gross Error Detection Algorithm: *N/A*
- ◇ Model Parameter Updating: *N/A*
- ◇ Outlier Detection: *N/A*
- ◇ Steady State Data Base Length: *N/A*
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: Rosenbrock Hillclimb
- ◇ First Order Sensitivity Objective: Yes. Sensitivity results are obtained using formula's listed by Fiacco (see section 3.6). Necessary derivatives were obtained by perturbation.
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes

- ◇ Check for Suff. Accuracy Objective: No
- ◇ Check for Sign. Difference Setpoints: No
- ◇ Implementation of Setpoints: *N/A*

RUN # 12

- ◇ Case Study: Propane Propylene Splitter
- ◇ Subject or Objective: Monte Carlo Simulation
- ◇ "Plant" Model: Eduljee and Fenske (with nominal parameters as shown in Table 4.2.1).
- ◇ Optimizer Model: *N/A*
- ◇ Noise: see Run # 1
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: see Run # 1. Statistical analysis was based on estimation results obtained from using all 100 data points.
- ◇ Steady State Detection: *N/A*(steady state study)
- ◇ Data Reconciliation Algorithm: *N/A*
- ◇ Gross Error Detection Algorithm: *N/A*
- ◇ Model Parameter Updating: Least Squares in $\ln \alpha$ and $1/N$ using Marquardt Levenspiel.
- ◇ Outlier Detection: Yes
- ◇ Steady State Data Base Length: All 100 datapoints are used.
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: Rosenbrock Hillclimb
- ◇ First Order Sensitivity Objective: *N/A*
- ◇ Second Order Sensitivity Objective: *N/A*
- ◇ First Order Sensitivity Setpoints: *N/A*
- ◇ Check for Suff. Accuracy Objective: *N/A*
- ◇ Check for Sign. Difference Setpoints: *N/A*

◇ Implementation of Setpoints: *N/A*

RUN # 13

◇ Case Study: Boiler Network

◇ Subject or Objective: Sensitivity Results

◇ "Plant" Model: *N/A*

◇ Optimizer Model: Second Order Polynomial

◇ Noise: *N/A*

◇ Bias or Other Systematic Errors: None

◇ Data Base: Optimal individual boiler loads with respect to the boiler model coefficients listed on Table 5.4.3. These individual loads are given in figure 5.4.4.

◇ Steady State Detection: *N/A*(steady state study)

◇ Data Reconciliation Algorithm: *N/A*

◇ Gross Error Detection Algorithm: *N/A*

◇ Model Parameter Updating: *N/A*

◇ Outlier Detection: *N/A*

◇ Steady State Data Base Length: *N/A*

◇ Check Significantly Different New Data: *N/A*

◇ Optimization Algorithm: SQP

◇ First Order Sensitivity Objective: Yes

◇ Second Order Sensitivity Objective: Yes

◇ First Order Sensitivity Setpoints: Yes

◇ Check for Suff. Accuracy Objective: No

◇ Check for Sign. Difference Setpoints: No

◇ Implementation of Setpoints: *N/A*

RUN # 14

◇ Case Study: Boiler Network

◇ Subject or Objective: Variance of the Optimal Value Function.

◇ "Plant" Model: Rigorous Dynamic Boiler Model (Bertrand, [1986]).

◇ Optimizer Model: Second Order Polynomial

◇ Noise: None

◇ Bias or Other Systematic Errors: None

◇ Data Base: Boiler Network Loads as Given in Table 5.4.4.

◇ Steady State Detection: Band Test

◇ Data Reconciliation Algorithm: None

◇ Gross Error Detection Algorithm: None

◇ Model Parameter Updating: One step least squares.

◇ Outlier Detection: None

◇ Steady State Data Base Length: fixed, 5.

◇ Check Significantly Different New Data: Yes

◇ Optimization Algorithm: SQP (execution frequency: every 5 minutes)

◇ First Order Sensitivity Objective: Yes

◇ Second Order Sensitivity Objective: Yes

◇ First Order Sensitivity Setpoints: Yes

◇ Check for Suff. Accuracy Objective: Yes

◇ Check for Sign. Difference Setpoints: Yes

◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^*_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 15

◇ Case Study: Boiler Network

◇ Subject or Objective: Variance of the Optimal Value Function.

◇ "Plant" Model: Rigorous Dynamic Boiler Model (Bertrand, [1986]).

◇ Optimizer Model: Second Order Polynomial

◇ Noise: None

- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Boiler Network Loads as Given in Table 5.4.4.
- ◇ Steady State Detection: Band Test
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: One step least squares.
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: fixed, 5.
- ◇ Check Significantly Different New Data: Yes
- ◇ Optimization Algorithm: SQP (execution frequency: every minute)
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: Yes
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^{\star}_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 16

- ◇ Case Study: Boiler Network
- ◇ Subject or Objective: Marginal Optimization Gain
- ◇ "Plant" Model: *N/A*
- ◇ Optimizer Model: Second Order Polynomial
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Different network loads under optimal individual boiler loads with respect to the model parameters given in Table 5.4.3.
- ◇ Steady State Detection: *N/A*(steady state study)

- ◇ Data Reconciliation Algorithm: *N/A*
- ◇ Gross Error Detection Algorithm: *N/A*
- ◇ Model Parameter Updating: *N/A*
- ◇ Outlier Detection: *N/A*
- ◇ Steady State Data Base Length: *N/A*
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: SQP
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: No
- ◇ First Order Sensitivity Setpoints: No
- ◇ Check for Suff. Accuracy Objective: *N/A*
- ◇ Check for Sign. Difference Setpoints: *N/A*
- ◇ Implementation of Setpoints: *N/A*

RUN # 17

- ◇ Case Study: Boiler Network as reported on by Cho [1978]
- ◇ Subject or Objective: Marginal Optimization Gain
- ◇ "Plant" Model: *N/A*
- ◇ Optimizer Model: Second Order Polynomial
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Different network loads under optimal individual boiler loads with respect to the model parameters given in Table 5.4.5. It is important to note that minimal and maximum boiler load constraints were introduced. The minimal load for boilers 1 and 4 is 10000 lb/h and for boilers 2 and 3 is 15000 lb/h. The maximal load for boilers 1 and 4 is 120000 lb/h and for boilers 2 and 3 is 80000 lb/h. These constraints become active/inactive for certain network loads, which causes the jumps in the curve on figure

5.4.19.

- ◇ Steady State Detection: *N/A*(steady state study)
- ◇ Data Reconciliation Algorithm: *N/A*
- ◇ Gross Error Detection Algorithm: *N/A*
- ◇ Model Parameter Updating: *N/A*
- ◇ Outlier Detection: *N/A*
- ◇ Steady State Data Base Length: *N/A*
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: SQP
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: No
- ◇ First Order Sensitivity Setpoints: No
- ◇ Check for Suff. Accuracy Objective: *N/A*
- ◇ Check for Sign. Difference Setpoints: *N/A*
- ◇ Implementation of Setpoints: *N/A*

RUN # 18

- ◇ Case Study: CSTR with consecutive reactions
- ◇ Subject or Objective: Dynamic interaction between a steady state on-line optimizer and a continuous plant.
- ◇ "Plant" Model: Laplace domain model, without transient effects (only sustained oscillations), specified by equation 5.5.2. The control action is assumed to be immediate.
- ◇ Optimizer Model: Optimizer given by equation 5.5.3.
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: *N/A*– frequency analysis
- ◇ Steady State Detection: *N/A*(steady state study)

- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: None (model is not updated)
- ◇ Outlier Detection: *N/A*
- ◇ Steady State Data Base Length: *N/A*
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: Analytical Solution
- ◇ First Order Sensitivity Objective: No
- ◇ Second Order Sensitivity Objective: No
- ◇ First Order Sensitivity Setpoints: No
- ◇ Check for Suff. Accuracy Objective: No
- ◇ Check for Sign. Difference Setpoints: No
- ◇ Implementation of Setpoints: Immediate Implementation of Optimizer Results.

RUN # 19

- ◇ Case Study: Boiler Network
- ◇ Subject or Objective:Dynamic interaction between a steady state on-line optimizer and a continuous plant.
- ◇ "Plant" Model: Rigorous Dynamic Boiler Model (Bertrand, [1986])
- ◇ Optimizer Model: Second Order Polynomial
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Increasing boiler network loads as specified in Table 5.4.4., but at different frequencies.
- ◇ Steady State Detection: Band Test
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None

- ◇ Model Parameter Updating: One step least squares
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: fixed, 5
- ◇ Check Significantly Different New Data: Yes
- ◇ Optimization Algorithm: SQP, different frequencies are applied.
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: Yes
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^*_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 20

- ◇ Case Study: Heat Exchanger Network
- ◇ Subject or Objective: Model Plant Mismatch
- ◇ "Plant" Model: Model as specified in 4.4.4.1.
- ◇ Optimizer Model: Simple model as in 4.4.4.2.
- ◇ Noise: None
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: For the results in figure 5.5.12 the crude mass flowrate (149000 lb/h), the crude inlet temperature (100 F) and the kerosene inlet temperature (390 F) remained constant. The kerosene mass flowrate was increased from 33800 to 53800 lb/h in two thousand steps. With these input values, one heat exchanger was simulated using the rigorous model. To the output values a zero mean, gaussian noise was added with a standard deviation of 1% the nominal values. The noise was generated using the Box Muller transformed URN14 generated random numbers. The simple model

was fitted to these perturbed values. For the results in figure 5.5.13 the crude mass flowrate (149000 lb/h), the crude inlet temperature (100 F) and the kerosene mass flow (43800 F) remained constant. The kerosene inlet temperature was increased from 290 to 490 F in two thousand steps. With these input values, the same heat exchanger was simulated. To the output values a zero mean, gaussian noise was added with a standard deviation of 1% the nominal values. Again, the noise was generated using the Box Muller transformed URN14 generated random numbers. Also here the simple model was fitted to these data. The data files were not stored.

- ◇ Steady State Detection: *N/A*(steady state study)
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: Least squares using Marquardt Levenspiel
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: All 2000 points
- ◇ Check Significantly Different New Data: *N/A*
- ◇ Optimization Algorithm: *N/A*
- ◇ First Order Sensitivity Objective: *N/A*
- ◇ Second Order Sensitivity Objective: *N/A*
- ◇ First Order Sensitivity Setpoints: *N/A*
- ◇ Check for Suff. Accuracy Objective: *N/A*
- ◇ Check for Sign. Difference Setpoints: *N/A*
- ◇ Implementation of Setpoints: *N/A*

RUN # 21

- ◇ Case Study: Heat Exchanger Network
- ◇ Subject or Objective: Demonstration of variable database length
- ◇ "Plant" Model: Rigorous Model (section 4.4.4.1)

- ◇ Optimizer Model: Simplified Model (section 4.4.4.2)
- ◇ Noise: Zero mean, Gaussian. Standard deviation is 2% of nominal values.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Description in text.
- ◇ Steady State Detection: N/A(steady state study)
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: Least Squares using Marquardt Levenspiel.
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: variable, maximal length 10
- ◇ Check Significantly Different New Data: None
- ◇ Optimization Algorithm: Search (one dimensional)
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: No
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: Only significant changes are implemented with settling time (4 time units).

RUN # 22

- ◇ Case Study: Heat Exhanger Network
- ◇ Subject or Objective: Influence of Noise on Optimizer Performance.
- ◇ "Plant" Model: Rigorous Model (section 4.4.4.1)
- ◇ Optimizer Model: Simple Model (section 4.4.4.2)
- ◇ Noise: Different levels of white, zero mean gaussian noise.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Described in the beginning of section 5.5.4.1.2.

- ◇ Steady State Detection: N/A(steady state example)
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: Least Squares using Marquardt Levenspiel
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: Variable, maximal length is 10.
- ◇ Check Significantly Different New Data: No
- ◇ Optimization Algorithm: Search (one dimensional)
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: No
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: Only significant changes are implemented with settling time (4 time units).

RUN # 23

- ◇ Case Study: Heat Exhanger Network
- ◇ Subject or Objective: Influence of Noise on Optimizer Performance.
- ◇ "Plant" Model: Rigorous Model (section 4.4.4.1)
- ◇ Optimizer Model: Simple Model (section 4.4.4.2)
- ◇ Noise: Different levels of noise. The noise is colored as expressed by equations 5.5.6–7. The noise is stationary.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Described in the beginning of section 5.5.4.1.2.
- ◇ Steady State Detection: N/A(steady state example)
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None

- ◇ Model Parameter Updating: Least Squares using Marquardt Levenspiel
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: Variable, maximal length is 10.
- ◇ Check Significantly Different New Data: No
- ◇ Optimization Algorithm: Search (one dimensional)
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: No
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: Only significant changes are implemented with settling time (4 time units).

RUN # 24

- ◇ Case Study: Heat Exhanger Network
- ◇ Subject or Objective: Influence of Noise on Optimizer Performance.
- ◇ "Plant" Model: Rigorous Model (section 4.4.4.1)
- ◇ Optimizer Model: Simple Model (section 4.4.4.2)
- ◇ Noise: Different levels of noise. The noise is colored as expressed by equations 5.5.6–7. The noise is not stationary. The noise level suddenly increases for a short period of time.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: Described in the beginning of section 5.5.4.1.2.
- ◇ Steady State Detection: N/A(steady state example)
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: Least Squares using Marquardt Levenspiel
- ◇ Outlier Detection: None

- ◇ Steady State Data Base Length: Variable, maximal length is 10.
- ◇ Check Significantly Different New Data: No
- ◇ Optimization Algorithm: Search (one dimensional)
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: No
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: Only significant changes are implemented with settling time (4 time units).

RUN # 25

- ◇ Case Study: Boiler Networks
- ◇ Subject or Objective: Influence of Noise on Optimizer Performance
- ◇ "Plant" Model: Rigorous Dynamic Boiler Model (Bertrand, [1986])
- ◇ Optimizer Model: Second Order Polynomial
- ◇ Noise: Different levels of zero mean, gaussian stationary noise.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: As specified in Table 5.4.4.
- ◇ Steady State Detection: Wilcoxon Based Test
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: One step least squares
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: fixed, 5
- ◇ Check Significantly Different New Data: Yes
- ◇ Optimization Algorithm: SQP
- ◇ First Order Sensitivity Objective: Yes

- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: No
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^*_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 26

- ◇ Case Study: Boiler Networks
- ◇ Subject or Objective: Influence of Noise on Optimizer Performance
- ◇ "Plant" Model: Rigorous Dynamic Boiler Model (Bertrand, [1986])
- ◇ Optimizer Model: Second Order Polynomial
- ◇ Noise: Different levels of zero mean, gaussian stationary noise.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: As specified in Table 5.4.4.
- ◇ Steady State Detection: Wilcoxon Based Test
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: One step least squares
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: fixed, 5
- ◇ Check Significantly Different New Data: Yes
- ◇ Optimization Algorithm: SQP
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: Yes
- ◇ Check for Sign. Difference Setpoints: Yes

◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^{\star}_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 27

◇ Case Study: Boiler Networks

◇ Subject or Objective: Influence of Noise on Optimizer Performance

◇ "Plant" Model: Rigorous Dynamic Boiler Model (Bertrand, [1986])

◇ Optimizer Model: Second Order Polynomial

◇ Noise: Different levels of noise. The noise is colored as expressed by equations 5.5.6–7. The noise is stationary.

◇ Bias or Other Systematic Errors: None

◇ Data Base: As specified in Table 5.4.4.

◇ Steady State Detection: Wilcoxon Based Test

◇ Data Reconciliation Algorithm: None

◇ Gross Error Detection Algorithm: None

◇ Model Parameter Updating: One step least squares

◇ Outlier Detection: None

◇ Steady State Data Base Length: fixed, 5

◇ Check Significantly Different New Data: Yes

◇ Optimization Algorithm: SQP

◇ First Order Sensitivity Objective: Yes

◇ Second Order Sensitivity Objective: Yes

◇ First Order Sensitivity Setpoints: Yes

◇ Check for Suff. Accuracy Objective: Yes

◇ Check for Sign. Difference Setpoints: Yes

◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^{\star}_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

RUN # 28

- ◇ Case Study: Boiler Networks
- ◇ Subject or Objective: Influence of Noise on Optimizer Performance
- ◇ "Plant" Model: Rigorous Dynamic Boiler Model (Bertrand, [1986])
- ◇ Optimizer Model: Second Order Polynomial
- ◇ Noise: Different levels of noise. The noise is colored as expressed by equations 5.5.6–7. The noise is not stationary. The noise level suddenly increases for a short period of time.
- ◇ Bias or Other Systematic Errors: None
- ◇ Data Base: As specified in Table 5.4.4.
- ◇ Steady State Detection: Wilcoxon Based Test
- ◇ Data Reconciliation Algorithm: None
- ◇ Gross Error Detection Algorithm: None
- ◇ Model Parameter Updating: One step least squares
- ◇ Outlier Detection: None
- ◇ Steady State Data Base Length: fixed, 5
- ◇ Check Significantly Different New Data: Yes
- ◇ Optimization Algorithm: SQP
- ◇ First Order Sensitivity Objective: Yes
- ◇ Second Order Sensitivity Objective: Yes
- ◇ First Order Sensitivity Setpoints: Yes
- ◇ Check for Suff. Accuracy Objective: Yes
- ◇ Check for Sign. Difference Setpoints: Yes
- ◇ Implementation of Setpoints: $l_{n+1} = \frac{n}{n+1} l_n + \frac{1}{n+1} l^*_{n+1}$, if no significant change in setpoints. Significant changes are applied as such.

REFERENCES

Almasy G.A., Mah R.S.H.; "Estimation of Measurement Variances from Process Data"; I&EC Chem. Prcc. Des. Dev., **23**, p779-784, 1984

Åström K.J., Wittenmark B.; "Computer Controlled Systems"; Prentice-Hall Information and System Sciences Series, Englewood Cliffs, NJ, 1984

Atkins D.; "Managerial Decentralization and Decomposition in Mathematical Programming"; Oper. Res. Quart. **25**, 4, 1973

Bamberger W., Isermann R.;"Adaptive On-line Steady State Optimization of Slow Dynamic Processes", Automatica, **17**, 199, 1978

Barnett V., Lewis T.; *Outliers in Statistical Data* 2nd Edition ; Wiley Series in Probability and Mathematical Statistics, John Wiley and Sons, Chicester, 1984

Bartman R.V.;"Dual Composition Control in a C_3/C_4 Splitter"; Chemical Engineering Progress, Sept. 1981

Beck J.V., Arnold K.J.; *Parameter Estimation in Engineering and Science*; Wiley Series in Probability and Mathematical Statistics, John Wiley & Sons, New York, 1977

Bergh P.A.; "Planning for Millwide Control : The Building Block Approach"; PIMA Magazine, January 1984

Bernard J.W.; "Plan Control at the Right Level"; Control Engineering, Sept. 1966

Bernard J.W., Howard G.M.; "Organizing Multilevel Process Control Systems"; Automation, March 1970

Bertrand C.R.; *A Study on Dynamic Simulation and Control of Steam Systems*; M.S. Thesis, University of Maryland, Nuclear & Chemical Engineering Department, Graduate School, University of Maryland at College Park (UMCP), 1986

Bhattacharya A., Joseph B.; "On-Line Optimization of Chemical Processes"; Proceedings of the ACC Conference, 1982

Biegler L.T., Cuthrell J.E.; "Improved Infeasible Path Optimization for Sequential Modular Simulators - II: The Optimization Algorithm"; Computers & Chemical Engineering, **9**,3, 1985

Biggs M.C.; "Constrained Minimization Using Recursive Equality Quadratic Programming"; in: Lootsma F.A. (editor); *Numerical Methods for Nonlinear Optimization*, Academic Press, 1972

Biggs M.C.; "Constrained Minimization Using Recursive Quadratic Programming : Some Alternative Subproblem Formulations" ; in : Dixon L.C.W. and Szegö G.P. (editors); *Towards Global Optimization 2*; North-Holland Publishing Company, 1978

Biggs (also Bartholomew-Biggs) M.C.; "A Numerical Comparison between two Approaches to the Nonlinear Programming Problem"; in: Dixon L.C. W. and Szegö G.P. (editors); *Towards Global optimization 2*; North-Holland Publishing Company, 1978

Bird R.B, Stewart W.E., Lightfoot E.N.; *Transport Phenomena*; John Wiley & Sons, New York, 1960

Blevins T., Roberts D., Block L., Andreasen C.; "A Standardized Software Package for Energy Monitoring and Energy System Optimization in the Pulp and Paper Industry", Proceedings of the ISA Conference, Houston TX, published in Advances in Instrumentation, **35**, 1, October 1980

Box G.E.P., Draper N.R.; "The Bayesian Estimation of Common Parameters from Several Responses"; Biometrika, **52**, 1965

Box G.E.P, Hunter W.G., Mc Gregor J.F., Erjavec J.; "Some Problems Associated with the Analysis of Multiresponse Data"; Technometrics, **15**, 1, 1973

Box G.E.P. and Mc Gregor J.F.; "The Analysis of Closed Loop Dynamic Stachastic Systems"; Technometrics, **16**, 3, August 1974

Brdyś M., Roberts P.D.; "Convergence and Optimality of Modified Two-Step Algorithm for Integrated System Optimization and Parameter Estimation"; Int. J. Systems Sci., **18**, 7, p. 1305-1322, 1987

Chang T.S., Seborg D.E.; "Process Control in the Presence of Constraints"; Proceedings of the ACC Conference, 1982

Chen C.Y., Joseph B.; "On-Line Optimization using a Two-Phase Approach : An Application Study"; Ind. Eng. Chem. Res., **26**, pp. 1924-1930, 1987

Cho C.H.; "Optimum Boiler Load Allocation"; Instrumentation Technology, Oct 1978

Cho C.H., Blevins T.L.; "Applying Energy Management in Pulp and Paper Mills"; TAPPI (The Journal of the Technical Association of the Pulp and Paper Industry), **63**, 6, June 1980

Christian S.D., Lane E.H, Garland F.; "Linear Least Squares Analysis", Journal of Chemical Education, **51**, 7, July 1974

Clarck P.A., Westerberg A.W.; "Optimization for Design Problems Having more than one Objective", Computers & Chemical Engineering, **7**, 4, 1983

Coleman T.F.; *Large Sparse Numerical Optimization*; Lecture Notes in Computer Science # 165, Springer-Verlag, Berlin, 1984

Contino A.V.; "Improve Plant Performance via Statistical Process Control"; Chemical Engineering, July 1987

Coombes G., Randall K., Wiseman J.V.; "Pulp Mill Setpoint Control System"; Paper Presented at the ACC Conference 1983

Cooper L., Steinberg D.; *Introduction to Methods of Optimization*; W.B. Saunders Company, Philadelphia, 1970

Craw I.A., Foster D., Rossiter A.P.; "Non-linear Thermodynamically Based Models are Critical for Global Optimization of Large Scale Cogeneration Systems"; presented at the ACC Conference, Atlanta, Georgia, 1988 (not printed in the Proceedings, available through ICI-TENSA Services, 17300 Saturn Lane, Suite 113, Houston, TX 77058)

Cronin P., Woerde H., Vasbinder R.; "Large Scale Plant Optimization Problems"; Proceedings of the 1988 ACC Conference, Atlanta, 1988

Crowe C.M., Garcia Campos Y.A., Hrymak A.; "Reconciliation of Process Flow Rates by Matrix Projection"; AIChE Journal, —bf 29, p.881, 1983

Cutler C.R., Perry R.T.; "Real Time Optimization with Multivariable Control is Required to Maximize Profits"; Computers and Chemical Engineering, 7, 5, pp. 663-667, 1983

Darby M.L., LeJune C.A.; "Optimization Control for Olefins Plant Refrigeration Area"; presented at the AIChE Spring National Meeting, New Orleans, March 1988

Demming W.E.; *Statistical Adjustment of Data*, Wiley and Sons, New York, 1946

D'Hulster F.M., Van Cauwenberghe A.R., Van Damme P.S., Froment G.F.; "A Digital Computer Configuration for Optimized Control of a Thermal Cracking Reactor"; Proceedings of an IFAC Symposium on Digital Computer Applications in Düsseldorf, F.R.G., 1980

Dixon L.C.W., Szegö G.P. (editors); *Numerical Optimization of Dynamic Systems*; North-Holland Publishing Company, Amsterdam, 1980

Dudewitz E.J., Ralley T.G.; *The Handbook of Random Number Generations and Testing with TESTRAND Computer Code*, American Series in amthematical and Management Science, 1981.

Duran M.A., Grossmann I.E.; "Simultaneous Optimization and Heat Integration of Chemical Processes", AIChE Journal, **32**, 1, January 1986

Edgar T.F.,Himmelblau D.M.; *Optimization of Chemical Processes*; ChE 356 Class Notes, University of Texas, Austin, expected 1988

Fenske M.R.; "Fractionation of Straight-Run Pennsylvania Gasoline", I&EC, **24**, pp482, 1932

Fiacco A.V.; "Nonlinear Programming Sensitivity Analysis Results Using Strong Second Order Assumptions"; in : Dixon L.C.W. and Szegö G.P. (editors); *Numerical Optimization of Dynamic Systems*; North-Holland Publishing Company, Amsterdam, 1980a

Fiacco A.V.; "Continuity of the Optimal Value Function under the Mangasarian Fromovitz Constraint Qualification"; Technical Paper T-432, Institute for Management Science and Engineering, George Washington University, Washington D.C., 1980b

Fiacco A.V.; "Optimal Value Continuity and Differential Stability Bounds under the Mangaserian Fromovitz Constraint Qualification"; Technical Paper T-435, Institute for Management Science and Engineering, George Washington University, Washington D.C., 1980c

Fiacco A.V., Hutzler W.P.; "Optimal Value Differential Stability Results for General Inequality Constrained Differentiable Mathematical Programs", in Fiacco A.V. (editor); *Mathematical Programming with Data Perturbations I*; Lecture Notes in Pure and Applied Mathematics, Volume 73, Marcel Dekker Inc., New York, 1982

Fiacco A.V.; *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*; Mathematics in Science and Engineering, Volume 165, Academic Press Inc., New York, NY, 1983

Findeisen W., Brdyś M., Malinowski K., Tatjewski P., Wózniak A.; "On-Line Hierarchical Control for Steady State Systems", IEEE Trans. Autom. Control **23**, pp.189-209, 1978

Funk G.L., Kania C.C., Terhune R.D.; "Why Risk Partial Optimization", Hydrocarbon Processes, pp101-104, Nov. 1984

Garcia C.E., Prett D.M.; "Advances in Industrial Model-Predictive Control"; in Morari M., Mc Avoy T.J. (editors); Proceedings of the Third International Conference on Chemical Process Control, Asilomar, CA, published by Elsevier, Amsterdam, 1986

Garcia-Borras T.; *(Manual for Improving) Boiler and Furnace Performance*; Gulf Publishing Co., Houston, 1983

Gauvin J., Tolle J.W.; "Differential Stability in Nonlinear Programming"; SIAM Journal Control Optimization, **15**(2), 1977

Gauvin J., Dubeau F.; "Some Examples and Counterexamples for the Stability Analysis of Nonlinear Programming Problems", in Fiacco A.V. (editor); *Mathematical Programming*, Study 21, The Mathematical Programming Society, North-Holland, Amsterdam 1984

Gentler J., Almasy G.A.; "Balance Calculations Through Dynamic System Modeling"; Automatica, **9**, 1973

Gijsbrechts E.; "Hierarchical Approaches to Organizational Problems", Large Scale System **8**, p.87, 1985

Gilliland E.R.; "Multicomponent Rectification", I&EC, **32**, pp 1220-1230, 1940

Golub G.H., Van Loan C.F.; *Matrix Computations*; Johns Hopkins Series in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, 1983 (fifth printing, 1987)

Gomes V.G.; "Controlling Fired Heaters"; Chemical Engineering, January 1985

Green D., Al ai-Shaikh; "Evaluating and Improving Efficiency in a Boiler Complex through Computer Control", JACC, 1980

Grossmann I.E., Halemane K.P.; "Decomposition Strategy for Designing Flexible Chemical Plants"; AIChE Journal, **28**, 4, 1982

Grossmann I.E., Halemane K.P., Swaney R.E.; "Optimization Strategies for Flexible Chemical Processes"; Computers & Chemical Engineering, **7**, 4, 1983

Himmelblau D.M., "New Strategies for Rectification of Uncertain Data for Process Design", publication expected 1988

Hlavácek V.; "Analysis of a Complex Plant Steady State and Transient Behavior - I. Plant Data Estimation and Adjustment", Computers & Chemical Engineering **1**, 1, p.75, 1977

Hogg R.V., Tanis E.A.; *Probability and Statistical Inference* Second Edition; Macmillan Publishing Co., New York, 1983

Holm S., Klein D.; "Three Methods for Postoptimal Analysis in Integer Linear Programming",in Fiacco A.V. (editor);*Mathematical Programming*, Study 21, The Mathematical Programming Society, North-Holland, Amsterdam 1984

Hwang C.L., Masud A.S.M., Paidy S.R., Yoon K.; *Multiple Objective Decision Making - Methods and Applications : A State of the Art Survey*; Springer-Verlag, Berlin 1979

Hwang C.L., Paidy S.R., Yoon K., Masud A.S.M.; "Mathematical Programming with Multiple Objectives : A Tutorial"; Computers and Oper. Res., **7**, 1980

Iordache C., Mah R.S.H., Tamhane A.C.; "Performance Studies of the Measurement Test for Detection of Gross Errors in Process Data"; AIChE Journal, **31**, 7, pp. 1187-1201, 1985

Isermann R.; "Advanced Methods of Process Computer Control for Industrial Processes"; Computers in Industry **2**, pp. 59-72, 1981

Jackson P.J., Agnew J.B.; "A Model-Based Scheme for the On-line Optimization of a Liquid Extraction Process"; Computers and Chemical Engineering, **4**, pp. 241-250, 1980

Jang S.S., Joseph B., Mukai H.; "On-line Optimization of Constrained Multivariable Processes"; AIChE Journal, **33**, 1, pp. 26-35, 1987

Jennergren L.P.; "On the Concept of Coordinability in Hierarchical Systems Theory"; Int. J. Systems Sci. **5**, pp. 493-497, 1974

Joliffe I.T.;"A Note on the Use of Principal Components in Regression"; Appl. Statist., **31**, pp.300–303, 1982

Kaya A., Moss W.H., Sommer A.C.; "Application of Optimization Methods to Save Energy in Plant Chillers"; Proceedings of the ACC Conference 1983

Kern D.Q.; *Process Heat Transfer*; Mc Graw Hill, New York , 1950

King C.J.; *Seperation Processes* , second edition; Chemical Engineering Series, Mc Graw Hill, New York, 1979

Klatte D.; "A Sufficient Condition for Lower Semicontinuity of Solution Sets of Systems of Convex Inequalities"; in Fiacco A.V. (editor); *Mathematical Programming Study 21*, A Publication of the Mathematical Society, North-Holland, Amsterdam, 1984

Ko D.Y.; "Optimizing Control of Powerhouse Boiler Load"; ISA, Paper 87-1002, 1987

Kretsovalis A., Mah R.S.H.; "Observability and Redundancy Classification in Multicomponent Process Networks"; AIChE Journal, **33**, 1, 1987

Kuehn D.R., Davidson H., "Mathematics of Control"; Chemical Engineering Progress, **57**, 44, June 1961

Lasdon L.S., Baker T.E.; "The Integration of Planning, Scheduling and Process Control"; in Morari M., Mc Avoy T.J. (editors); Proceedings of the Third International Conference on Chemical Process Control, Asilomar, CA, published by Elsevier, Amsterdam, 1986

Latour P.R.; "Requirements for Succesful Closed Loop Optimization of Petroleum Refining Processes"; Proceedings of an IFAC Symposium on Digital Computer Applications in Düsseldorf, FRG, 1980

Lee K.S., Lee W.K.; "On-Line Optimizing Control of a Fixed Bed Reactor"; Proceedings of the ACC Conference, 1983

Levenberg K.; "A Method for the Solution of Certain Nonlinear Problems in Least Squares"; Quart. Appl. Math., **2**, p.164, 1944

Lipták B.G.; "Improving Boiler Efficiency"; Chemical Engineering, May 1987

Luenberger D.V.; *Linear and Nonlinear Programming* (Second Edition) ; Addison-Wesley Publishing Company, Reading, Massachusetts, 1984

Luyben W.L.; "Steady State Energy Conservation Aspects of Distillation Column Control System Design"; I&EC Fundamentals, **14**, pp321-325, 1975

Mah R.S.H.; "Design and Analysis of Process Performance Monitoring Systems"; Proc. Eng. Found. Conf. on Chem. Process Control 2, Sea Island, Georgia, 1981

Mah R.S.H., Stanley G.M., Downing D.M.; "Reconciliation and Rectification of Process Flow and Inventory Data"; I&EC Proc. Des. Dev., **11**, p.175, 1976

Mah R.S.H., Tamhane A.C.; "Detection of Gross Errors in Process Data"; AIChE Journal, **28**, p.828, 1982

Marro G., Rossi R., Tonielli A.; "Applying a Two-Level Optimization Technique in Material Balance Problems"; in Titli A., Singh M.G.; *Large Scale Systems. Theory and Applications"*; IFAC Symposium 1980, published by Pergamon Press 1981

Martin G.D., Latour P.R., Richard L.A.; "Closed Loop Optimization of Distillation Energy", CEP, **77**, pp33-37, Sept. 1981

Martin D.E., Sharpe J.H., Latour P.R.; "Implementing On-line Optimization of a Crude Unit's Reflux Systems"; presented at the National AIChE Spring Meeting in New Orleans, March 1988

Marquardt D.W.; "An Algorithm for Least Squares Estimation of Nonlinear Parameters"; J. Soc. Ind. Appl. Math., **11**, p.431, 1963

Matsko T.N., Moss W.H., Scheib T.J.; "Optimal Boiler Load Allocation in Distributed Control", Proceedings of the ACC Conference 1982

Meyer E.F.; "Comments on Curve Fitting Methods", Anal. Chem., **54**, p.1878, 1982

Mc Avoy T.J.; *Interaction Analysis*; ISA Monograph Series 6, Research Triangle Park, NC, 1983

Mc Avoy T.J.; "Fitting Steady State Models to Distillation Tower Operating Data for On-Line Optimization and Control", to be published, 1988

Mc Donald R.J., Howat C.S.; "Data Reconciliation and Parameter Estimation in Plant Performance Analysis"; AIChE Journal, **34**, p.1, 1988

Mc Keown J.J.; "An Approach to Sensitivity Analysis"; in : Dixon L.C.W. and Szegö G.P. (editors); *Numerical Optimization of Dynamic Systems*; North-Holland Publishing Company, Amsterdam, 1980

Murray W.; "An Algorithm for Constrained Minimization"; in Fletcher R. (editor); *Optimization*; Academic Press, London 1969

Narasimhan S., Mah R.S.H.; "Generalized Likelihood ratio Method for Gross Error Identification"; AIChE Journal, **33**, 9, p.1514, 1987

Näsi M., Alikoski M., White D.C.; "Advanced Control of Acetylene Hydrogenation Reactors"; Hydrocarbon Processing, June 1985

Nath R.,Libby D.J., Duhon H.J.; "Joint Optimization of Process Units and Utility Systems",CEP, May 1986

Nye W.T.; *DELIGHT : An Interactive System for Optimization-Based Engineering Design*; Ph.D. Thesis, University of California at Berkeley, 1983

Nye W.T., Tits A.L.; "An Enhanced Methodology for Interactive Optimal Design"; Presented at the 1983 IEEE International Symposium on Circuits and Systems, Newport Beach, California, May 1983; see also (more recent) Nye W.T., Tits A.L.; "An Application Oriented, Optimization Based Methodology for Interactive Design of Engineering Systems"; International Journal of Control, **43**, 6, pp. 1693-1721, 1986

Park S.W., Himmelblau D.M.;"Diagnosis of Faults in Process Instrumentation"; ACC Conference (MP5-3:30), 1982

Payne F.W. (editor); *Efficient Boiler Operation Sourcebook*; The Fairmont Press Inc., Atlanta, Georgia, 1985

Poje J.B., Smart A.M.; "On-line Energy Optimization in a Chemical Complex"; CEP, May 1986

Powell M.J.D.; "A Method for Minimizing a Sum of Squares of Nonlinear Functions without Calculating Derivatives"; The Computer Journal, **7**, p.303, 1965

Powell M.J.D.; "A Fast Algorithm for Nonlinear Constrained Optimization Calculations"; presented at the Dundee Conference on Numerical Analysis, Dundee, Scotland, 1977

Rabin E.; "Dynamic Modeling of Steam Nodes", Summer Computer Simulation Conference Proceedings, pp.868-870, 1982

Radian Corporation, Energy Conservation ; *A Route to Improved Distillation Profitability*, Technology Transfer Manual (DOE 1CS14431-T2), Department of Energy, May 1980

Ray W.H.; *Advanced Process Control*; Chemical Engineering Series, Mc Graw Hill Book Company, New York, 1981

Reklaitis G.V., Ravindran A., Ragsdell K.M.; *Engineering Optimization, Methods and Optimization*; John Wiley & Sons, New York, 1983

Renfro J.G.; *Computational Studies in the Optimization of Systems Described by Differential / Algebraic Equations*; Ph.D. Dissertation, University of Houston (Department of Chemical Engineering), August 1986

Ripps D.L.; "Adjustment of Experimental Data"; in *Chemical Engineering Progress Symposium Series nr. 55*, Process Control and Applied Mathematics, Volume 61, AIChE, 1965

Roberts P.D., Roussias L.; "Overcoming Model Uncertainties in Optimising Supervisory Control"; Proceedings of an IFAC Symposium on Digital Computer Applications to Process Control in Düsseldorf FRG, 1980

Robinson, S.M.; "Perturbed Kuhn Tucker Points and Rates of Convergence for a Class of Nonlinear Programming Algorithms"; Math. Programming **7**(1), 1974

Romagnoli J.A., Stephanopoulos G.; "On the Rectification of Measurement Errors for Complex Chemical Plants"; Chemical Engineering Science, **35**, 1980

Serth R.W., Heenan W.A.; "Gross Error Detection and Data Reconciliation in Steam Metering Systems", AIChE Journal, **32**, p.773, 1986

Shetty C.M.; "Sensitivity Analysis in Linear Programming"; J. Ind. Eng., **10**, pp.397-386, 1959

Shinskey F.G.; *Energy Conservation through Control*; Academic Press, New York, NY, 1978

Shinskey F.G.; *Process Control Systems* second edition; Mc Graw Hill, New York, 1979

Sofer S.S., Mulholland R.J., Bare W.H.; "From Simple Regulatory Control to Computer Optimization"; Hydrocarbon Processing, January 1988

Sourander M.L., Kolari M., Cugini J.C., Poje J.B., White D.C.; "Control and Optimization of Olefin-Cracking Heaters"; Hydrocarbon Processing, June 1984

Sourander M.L., Gors S.; "Controlling and Optimizing Integrated Fractionation Processes in a Petroleum Refiney"; from *Dynamics and Control of Chemical Reactors and Distillation Columns*, Preprints of an IFAC Symposium in Bournemouth, UK, December 1986

Stanley G.M.; "On-Line Data Reconciliation for Process Control", Paper presented at the AIChE Annual Meeting, Nov. 14-18, Los Angeles, CA, 1982

Stanley G.M., Mah R.S.H.; "Estimation of Flows and Temperatures in Process Networks"; AIChE Journal, **23**, 1977

Stanley G.M., Mah R.S.H.; "Observability and Redundancy in Process Data Estimation"; Chemical Engineering Science, **36**, 1981a

Stanley G.M., Mah R.S.H.; "Observability and Redundancy Classification in Process Networks"; Chemical Engineering Science, **36**, 1981b

Takama N., Kuriyama T., Shiroko K., Umeda T.; "Optimal Water Allocation in a Petroleum Refinery"; Computers and Chemical Engineering, **4**, pp. 251-258, 1980

Tatjewski P.; "On-Line Control of Systems Using the Augmented Interaction Balance Method with Feedback"; Large Scale Systems **8**, pp.1-18, 1985

Tulpule B.R., Knapp C.H.; "Parametric Failure Detection & Estimation in Linear Systems"; ACC Conference (TP7-4:30) 1982

Tsang E.H., Milo M.D.; "Large Scale Ammonia Process Optimization in Real Time"; Proceedings of the ACC Conference 1988, Atlanta, 1988

Václavek V., Rajniak P., Ilavsky J.; "Sensitivity Analysis in Balance Computations of Complex Chemical Processes", Chemical Engineering Communications ,**3**, 1979

Václavek, V., Loucka M.; "Selection of Measurement Necessary to Achieve Multicomponent Mass Balances in Chemical Plants"; Chemical Engineering Science, **31**, 1969

Van Winkle M.; *Distillation*; Chemical Engineering Series, Mc Graw-Hill, New York, 1967

Weber R., Brosilow C.; "The Use of Secondary Measurements to Improve Control", AIChE Journal **18**, 3, 1972

Wen J.; "Mill-Wide Production Coordination Possible With Distributed Control"; Pulp & Paper, July 1983

Wentworth W.E.; "Rigorous Least Squares Adjustment", Journal of Chemical Education, **42**, 2, February 1965

Wilcox J.C. jr.; "Improving Boiler Efficiency"; Chemical Engineering, October 1978

Wilcoxon F., "Individual Comparisons by Ranking Methods", Biometrics Bulletin, **1**, pp.80–83, 1945

Zlobec S., Gardner R., Ben-Israel A.; "Regions of Stability for Arbitrarily Perturbed Convex Programs", in Fiacco A.V. (editor); *Mathematical Programming with Data Perturbations I*; Lecture Notes in Pure and Applied Mathematics, Volume 73, Marcel Dekker Inc., New York, 1982

Zolezzi T.; "On Stability Analysis in Mathematical Programming", in Fiacco A.V. (editor); *Mathematical Programming Study 21*, A Publication of the Mathematical Society, North-Holland, Amsterdam, 1984

CURRICULUM VITAE

Name:	Jan Koninckx
Permanent address:	Ryenlanddreef 32 B-2060 Merksem (Antwerpen) Belgium.
Degree and date to be conferred:	Ph.D., 1988.
Date of birth:	December 18, 1961.
Place of birth:	Merksem, Antwerpen, Belgium.
Secondary education:	St. Jan Berchmans College Antwerpen, 1980.

College institutions attended:

	Dates	Degree	Date of Degree
Rijksuniversiteit, Gent	1980-1985	Burgerlijk Ingenieur	1985
University of Maryland	1985-1988	Ph.D. Ch.E.	1988

Major: Chemical Engineering

Professional Publications:

J. Koninckx, T.J. McAvoy, T.E. Marlin; "On-line Optimization Using Steady State Models", AIChE National Spring Meeting 1988, New Orleans.

M.K.H. Fan, L.S. Wang, J. Koninckx, A.L. Tits; "A CAD Tandem for Optimization Based Design Interacting With User Supplied Simulators", Proceedings of the ACC Conference, Atlanta, 1988

Professional positions held:

August 1980 Process Engineer
Esso Belgium N.V.
B-2000 Antwerp, Belgium

1985-1988 Research Assistant
University of Maryland
College Park, MD 20740