

PENINSULA TECHNIKON



Department of Electrical Engineering
Faculty of Engineering

Modelling and optimal control of fed-batch fermentation
process for the production of yeast

by

Ncedo S. Mkondweni

Submitted in fulfillment of the requirement for
Masters degree of technology (Mtech): Electrical engineering

Under the supervision of

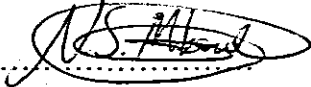
Prof. Raynitchka Tzoneva

DECEMBER 2002

DECLARATION

I, Ncedo S. Mkondweni, hereby declare that this dissertation represents my own work and the opinions contained are my own and not necessarily those of the Technikon. All references used have been accurately reported.

Signed:.....

A handwritten signature in black ink, appearing to read 'N.S. Mkondweni', enclosed within a hand-drawn oval border.

Name: Ncedo S. Mkondweni

SUMMARY

Fermentation is the process that results in the formation of alcohol or organic acids on the basis of growth of bacteria, moulds or fungi on different nutritional media (Ahmed et al., 1982). Fermentation process have three modes of operation i.e. batch, fed-batch and continuous mode of operation. The process that interests a lot of control engineers is the fed-batch fermentation process (Johnson, 1989). The Fed-batch process for the production of yeast is considered in the study.

The considered yeast in the study is the *Saccharomyces cerevisiae*. It grows in both aerobic and anaerobic environmental conditions with maximum product in the aerobic conditions, also at high concentration of glucose (Njodzi, 2001). Complexity of fed-batch fermentation process, non-linearity, time varying characteristics, application of conventional analogue controllers provides poor control due to problems in tuning individual loops and the process characteristics. The problem for control of the fed-batch process for the production of yeast is further complicated by the lack of on-line sensors, lack of adequate models as a result of poorly understood dynamics. The lack of on-line sensors results in the impossibility of tuning the analogue controllers in real time. The process for propagation of yeast in aerobic conditions is considered in the dissertation. The experiments are conducted at the University of Cape Town (UCT), Department of Chemical Engineering with a bioreactor and bio-controller are combined in a Biostat ® C lab scale plant (B. Braun Biotech International, 1996).

The bio-controller has built in PID controller loops for control variables, with the ability to adjust the controller parameters i.e. P, D and I through the serial interface (Seidler, 1996). Even though the used lab scale bio-controller has the ability to monitor certain variables, the automation of the industrial bioreactors is still developing slowly (Dochan and Bastin, 1990) with major problems experienced in

modelling and measuring important control variables on-line. This existing situation is due to the characteristics of the fermentation processes as an object of control with highly non-linear, non-stationary, slow dynamics and complex relationships between variables. The existing control strategies on industry are based only of local PID control of some easy for measuring variables. No computer systems for monitoring and optimisation of the process (Morari and Stephanopoulos, 1980).

The dissertation is overcoming the mentioned above drawbacks by developing methods, algorithms and programmes for building of a two layer system for optimal control of the Biostat ® C pilot plant with the following subsystems:

- Data acquisition,
- Modelling and simulation,
- Model parameter estimation,
- Process optimisation,
- PID controller parameter tuning,
- Real time control implementation

The system is based on LabVIEW™ and serial communication protocol. The interface between the Bio-start and the host computer is through a standard communication serial port. The development in the dissertation are described as follows:

Chapter 1 describes the necessity of the research discussed in the dissertation and highlights comparison between the different approaches for modelling and control of fed-batch processes for the production of yeast, (Johansson, 1993). The aim and the objectives of the dissertation are stated and explained.

Chapter 2 describes the process as an object of control looking precisely at the influence of the physiochemical variables on the biological variables. The relationship is identified through the enzymes. The results from previous experiments are

discussed to illustrate the constraints associated with the control of the process under study

Chapter 3 describes the different types of models as applicable to the dissertation. The comparison between different types is highlighted. The derivation of the developed yeast model using mass balance equations and rate laws is discussed and presented in the chapter. The problem for simulation of the model is solved using Matlab and LabVIEW™ programs.

Chapter 4 describes the formulation of the problem for estimation of the fed-batch model coefficients and the method, algorithm and programme developed to solve this problem.

In chapter 5 the optimal control problem is formulated and solved using optimal control theory, the approach of the functional of Lagrange is used. The optimisation layer problems are determined and based on the solutions of the previous upper layer i.e. the model parameters from the adaptation layer. The optimal operation of the process or yield of the yeast is based on some criteria for the production of biomass, and some constraints over minimal and maximal values of the variables. Decomposition method to solve the optimal control problem is developed on the bases of an augmented functional of Lagrange and decomposition in time domain. Algorithm of the method and program in Matlab are developed. Tuning of PID parameters for the controllers in Biostat ® C is described based on the optimal trajectories for the physiochemical variables obtained from the optimal control problem solution.

Chapter 6 presents algorithms and programmes for monitoring and real time control of fed-batch fermentation process for the production of yeast, using Personal Computer, B Braun Biostat ® C Lab scale fermentation unit and LabVIEW™ as

driver software. Hardware and software parts of the control systems are described and discussed. LabVIEW™ code is described.

Chapter 7 presents the users manual. The main functionality of the developed application and programmes is described and discussed.

The source code of the developed programmes is presented in chapter 8.

Chapter 9 presents the conclusion highlighting the developments in the dissertation as well as the future work on the topic and the possible application of the developed work in industry on the bigger scale fermentors.

The positive characteristics of the developed methods algorithms and programmes are:

- The developed model incorporates the physiochemical variables in the biological mass balance equations. In this way: the influence of the enzymes over the biological variables is utilised and possibilities for process optimisation is created.
- The process can be optimised in both physiochemical and biological variables.
- The physiochemical variables can be used as control inputs to reach the process optimisation.
- Data acquisition system gives good possibilities for analysis and simulation of the process.
- The optimal control of the process is achieved without using expensive on-line sensors for measurement of the biological variables. This is why the developed system is applicable to the existing hardware and software control and measurement systems in industry.

→ The control system automates the operation of the lab scale fermentation unit. It is safe, stable and operational.

“The dissertation is a dedication to my late mother”

Nolundi “Dawa” Mkondweni.

ACKNOWLEDGEMENTS

The research would not have been a success without the help and the support of the following people, company and institution through out the duration of the study.

Professor R. Tzoneva, my supervisor who made me believe that we can do it, through the encouragement and support that she gave me during the worst time of the study. National Research Foundation for funding the project. Marine Data Systems, the company I work for, for availing me the time to make possible developments of the dissertation. Njodzi Zizhou, for allowing us access to the experiments and giving all the help that he could. Finally and far most family, friends, students and staff of the Department of Electrical engineering at Peninsula Technikon for making the environment conducive for the completion of my dissertation.

TABLE OF CONTENTS

SUMMARY	i
ACKNOWLEDGEMENTS	vi
TABLE OF CONTENTS	vii
LIST OF FIGURES.....	xii
LIST OF TABLES	xiv
NOMENCLATURE.....	xv
Chapter 1	1
The problem for real time optimal control of the fed-batch fermentation process	1
1. Introduction.....	1
1.1 Awareness of the problem.....	1
1.2 Necessity of a research work in the field of optimisation and control of fermentation process in the production of yeast	3
1.3 Comparative analysis of the approaches for modelling and control of fed-batch fermentation process	5
1.4 The project aim and statement of the problem.....	12
1.4.1 Design sub-problems.....	12
1.4.2 Real time implementation sub-problems	12
1.5 The significance and objectives of the research project.....	12
1.6 Proposed strategy for control	14
1.6.1 Layers of the control system	14
1.6.2 Adaptation layer	15
1.6.2.1 Process modeling	15
1.6.2.2 Parameter estimation.....	15
1.6.3 Optimization layer.....	15
1.6.4 Direct control layer	16
1.7 Software	18
1.8 Conclusion.....	18
Chapter 2	19
Process description.....	19
2. Introduction.....	19
2.1 Fermentation process	19

2.2.	Fed-batch fermentation process	20
2.2	Fed-batch fermentation for production of yeast.....	22
2.3.1	Aerobic fed-batch culture.....	22
2.3.2	Anaerobic fed-batch culture.....	22
2.3.3	Variables characterizing growth kinetics of yeast.	22
2.3.3.1	The influence of temperature	23
2.3.3.2	The influence of pH	24
2.3.3.3	The influence dissolved oxygen (DO ₂).....	25
2.3.4	Results in yeast production caused by the influence of different variables.	26
2.3.5	Incorporation of the physiochemical variables in the mathematical model.....	27
2.3	Conclusion.....	28
	Chapter 3	29
	Layer for adaptation	29
	Process modelling	29
3.	Introduction	29
3.1	Process modelling	29
3.2	Model types.....	29
3.3	Model describing the behaviour of physiochemical variables.....	31
3.4	Model describing mass balance equations of the fed batch fermentation process.....	32
3.4.1	Expressions for growth and production rates.....	33
3.4.1.1	Growth rate laws	33
3.4.1.2	Stoichiometry	35
3.4.2	Biomass mass balance equation.....	37
3.4.3	Substrate mass balance equation.....	38
3.4.4	Product mass balance equation	39
3.4.5	Volume mass balance equation.....	40
3.4.6	Summary of derived mass balance equations	40
3.5	Incorporation of physiochemical variables into the biological model.....	41
3.6	Discrete representation of the model.....	42
3.7	Simulation of the model.....	43
3.7.1	LabVIEW TM 6.0Ii program for simulation.....	46
3.7.2	MATLAB and SIMULINK program for simulation	49

3.8	Conclusion.....	50
	Chapter 4.....	51
	Layer for adaptation.....	51
	Parameter estimation.....	51
4.	Introduction.....	51
4.1	Experimental data.....	51
4.2	Mathematical formulation of the problem for estimation.....	54
4.3	Method of solution.....	55
4.4	Algorithm of the method.....	58
4.5	Simulation and implementation of the algorithm.....	63
4.5.1	LabVIEW™6.01i program.....	63
4.5.2	MATLAB 6.1.....	65
4.6	Conclusion.....	67
	Chapter 5.....	68
	Layer for optimization.....	68
	Optimal control of the fermentation process.....	68
5.1	Problem for optimal control of the production of yeast.....	68
5.1.1	Formulation of the problem for optimal control.....	69
5.1.2	Method for problem solution.....	70
5.1.2.1	Necessary conditions for optimality.....	71
5.1.2.2	Coordinating vector.....	78
5.1.2.3	First level sub-problems.....	79
5.1.2.4	The coordinating sub-problem.....	88
5.1.3	Algorithm for calculation.....	89
5.2	Proportional Integral and Derivative (PID) controller.....	94
5.2.1	Background.....	94
5.2.1.1	PID.....	94
5.2.2	The Auto tuning Algorithm.....	95
5.2.2.1	Description of the input and outputs of the vi.....	97
5.3	Conclusion.....	98
	Chapter 6.....	99
	Monitoring and real time control implementation.....	99
6.	Introduction.....	99

6.1	HARDWARE ENVIRONMENT	101
6.1.1	Bio-controller Biostat ® C from B.Braun International Gmbh	101
6.1.2	IFB RS-422 box	103
6.1.3	Personal Computer (PC)	103
6.1.4	Description of serial interface	103
6.1.4.1	The RS-232 Interface Standard for Serial Data Communications	104
6.1.4.2	The RS-422 Interface Standard for Serial Data Communications	105
6.2	SOFTWARE ENVIRONMENT	106
6.2.1	Tools And Methodologies.....	106
6.2.2	Main LaBVIEW™ Libraries For The Design	108
6.2.3	Description Of Serial Interface Between Host And Control Units Of The B.Braun Biotech International Gmbh	109
6.2.3.1	Substation Initialization	110
6.2.3.2	Break-Generation Via Host System	110
6.2.3.3	Remote Mode.....	110
6.2.3.4	Reserved Control Character	111
6.2.3.5	Telegram Format.....	111
6.2.3.6	Response Telegrams Of Substations.....	112
6.2.3.7	End Of Data Transmission.....	112
6.2.3.8	Check sum.....	112
6.2.3.9	Requests And Commands.....	113
6.2.4	Developed software.....	113
6.2.4.1	Description of the DAQ program.....	113
6.2.4.2	Description of the Auto tuning program	119
6.3	Conclusion.....	120
	Chapter 7	121
	Operator's manual	121
7.	Introduction.....	121
7.1	Functional Overview.....	121
7.2	Main program.....	122
7.2.1	Start acquisition.....	123
7.2.2	Control loops settings.....	124
7.2.3	Initialize Bio-controller [Substation]	125

7.2.4	Off-line variables	126
7.2.5	Reports	127
7.2.6	Help	129
7.3	Parameter Estimation	130
7.3.1	Yeast Model	131
7.4	Optimal control	133
7.5	PID Auto tuning	134
7.5.3	Auto tuner	134
Chapter 8	136
Software description	136
8. Introduction	136
8.1 LabVIEW code	136
8.2 Matlab / Simulink	140
Chapter 9	141
Conclusions	141
9.1 Problems solved in the dissertation	141
9.1.1 Data acquisition and real time implementation	142
9.1.2 Modeling and simulation	142
9.1.3 Model parameter estimation	143
9.1.4 Process optimization	144
9.1.5 PID controller parameter tuning	144
9.2 Benefits of the two-layer control strategy	144
9.3 Future developments of methods and applicability	145
9.4 Application of developed strategy, methods, algorithm and programs	145
9.5 Publications in connection with the dissertation	145
References:	147
Appendix A	156

LIST OF FIGURES

Figure 1.1: Three layers of the control structure.....	14
Figure 1.2: Control structure.....	17
Figure 2.1 Diagram for stirred fed-batch fermentor.....	21
Figure 3.1 Model 1 inputs and their respective outputs.....	31
Figure 3.2 Control and Simulation sub pallets.....	46
Figure 3.3 GSim Unit Delay.vi.....	46
Figure 3.4 Hierarchy of the developed program for simulation of the yeast model ...	47
Figure 3.5 Flow of the model calculations.....	47
Figure 3.6 Yeast model front panel.....	48
Figure 4.1: Common schemes for estimation of the model parameters.....	54
Figure 4.2: Scheme for estimation of the model parameters	57
Figure 4.3 Block scheme of the algorithm for parameter estimation.....	62
Figure 4.4 Hierarchy of the developed algorithm for parameter estimation in LabVIEW™	63
Figure 4.5 Parameter estimation front panel.....	64
Figure 4.6 Cluster with the parameter est. variables.....	65
Figure 4.7 Matlab program flow for parameter estimation.....	66
Figure 5.1: Two level calculating structures.....	79
Figure 5.3: PID with auto-tuning	95
Figure 5.4: Process under PID control with Set point relay.....	96
Figure 6.1: Overview of the hardware components	101
Figure 6.2: B. Braun Biostat ® C bio-controller.....	102
Figure 6.3: LabVIEW™ front panel and processing code diagram.....	106
Figure 6.4: Suggested software architecture.....	107
Figure 6.5: Main Function menu.....	108
Figure 6.6: VI Menu and Sub menus	109
Figure 6.7: Schematic of DAQ program.....	114
Figure 6.8: Serial Port Read/Write LabVIEW functions	115
Figure 6.9: Checksum calculation.....	115
Figure 6.10: Strip_protocol.vi.....	116
Figure 6.11: Insert into database	117

Figure 6.12: Tables of the Process Variables in the database.....	117
Figure 6.13: Reports.....	118
Figure 6.14: Off-line measured variables.....	118
Figure 6.15: Auto tuner.....	119
Figure 7.1: Main program window.....	122
Figure 7.2: Acquisition indicators& controls.....	123
Figure 7.3: Control loops settings window.....	124
Figure 7.4: Main menu.....	125
Figure 7.5: Off-line variables.....	126
Figure 7.6: Reports window.....	128
Figure 7.7: Help window.....	129
Figure 7.8: Parameter estimator.....	130
Figure 7.9: Yeast model.....	131
Figure 7.10: Yeast model equation pop up window.....	132
Figure 7.11: Optimization pop up window.....	133
Figure 7.12: Auto tuning window.....	134
Figure 8.1 Hierarchy of the developed application.....	136

LIST OF TABLES

Table 2.1 Variables measured or controlled in bioreactors for yeast production.	23
Table 3.1 Input variables to the yeast model.....	45
Table 3.2 Output variables to the yeast model.....	45
Table 3.3 Simulation of the yeast model in MATLAB / SIMULINK	49
Table 4.1 Summary of data received from experiments conducted at UCT	53
Table 8.1 List of subvi's used in the dissertation.....	139

NOMENCLATURE

Abbreviations and definitions

Algorithm	→	Normally used as basis for writing a computer program. This is a set of fine rules with finite number of steps for solving a problem.
ANSI	→	American National Standards Institute. The principle standards development body in the USA.
ASCII	→	American Standard Code for Interchange. A universal standard for encoding alphanumeric characters into 7 or 8 binary bits. Drawn by ANSI to ensure compatibility between computer systems.
DCE	→	Data Communication Equipment. Devices that provide the functions required to establish, maintain and terminate a data transmission connection. Normally it refers to a modem.
DTE	→	Data Terminal Equipment. Devices acting as data source, data sink, or both.
EIA	→	Electronic Industries Association. A standard organisation in the USA specialising in the electrical and functional characteristics of the interface equipment.
EISA	→	Enhanced Industry Standard Architecture.
OTR	→	Oxygen transfer rate
OUR	→	Oxygen utilisation rate
ODBC	→	Open Database Connectivity
SQL	→	Sequential Query Language
TD	→	Transmit Data
RD	→	Receive Data
rpm	→	Revolutions per minute
vvm	→	Volume air flow per liquid volume in a reactor per minute
DO ₂	→	dissolved oxygen

Symbols

C	→	Dissolved oxygen concentration (mg/l)
C_s	→	Saturation dissolved oxygen concentration (mg/l)
D	→	Dilution rate,
F	→	Volumetric feed flow rate (l/hr)
F_i	→	Input volumetric flow rate (l/hr)
F_o	→	Output volumetric flow rate (l/hr)
K_s	→	Monod constant or simply saturation constant,
K_{La}	→	Mass transfer coefficient (hr^{-1})
K_d	→	Specific death rate (hr^{-1})
m_s	→	Maintenance co-efficient,
p_o	→	Product concentration (g/l)
s	→	Substrate concentration (g/l)
s_i	→	Initial substrate concentration (g/l)
s_o	→	Reactor substrate concentration (g/l)
μ	→	Specific growth rate
μ_{\max}	→	Maximum specific growth rate
V	→	Culture volume in the reactor vessel (litres)
V_o	→	Initial culture volume in the reactor vessel (litres)
x	→	Biomass concentration (g/l)
x_o	→	Initial biomass concentration (g/l)
$Y_{x/s}$	→	Production co-efficient biomass from substrate,
$Y_{p/x}$	→	Production co-efficient product from substrate,
L	→	Functional for Lagrange,
λ	→	Conjugate variables of the functional of lagrange,
v	→	Index for x, s, p, v variables,
δ	→	Symbol for interconnection in time domain,
$\left. \begin{array}{l} \alpha_{\lambda x} = alx \\ \alpha_{\lambda s} = als \\ \alpha_{\lambda p} = alp \\ \alpha_{\lambda v} = alV \end{array} \right\}$	→	Step of the gradient procedures for improving the conjugate variables.

$$\left. \begin{aligned} \alpha_{\delta x} &= adx \\ \alpha_{\delta s} &= ads \\ \alpha_{\delta p} &= adp \\ \alpha_{\delta V} &= adV \end{aligned} \right\} \rightarrow \text{For the interconnection of time steps}$$

$$\left. \begin{aligned} \alpha_x &= ax \\ \alpha_s &= as \\ \alpha_p &= ap \\ \alpha_V &= aV \end{aligned} \right\} \rightarrow \text{For state steps}$$

$$\left. \begin{aligned} \alpha_F &= aF \\ \alpha_T &= aT \\ \alpha_{pH} &= apH \\ \alpha_{DO_2} &= aDO_2 \end{aligned} \right\} \rightarrow \text{For control steps}$$

$$\left. \begin{aligned} e_\gamma &= epsl \\ \varepsilon_x &= epsx \\ \varepsilon_s &= epss \\ \varepsilon_p &= epsp \\ \varepsilon_V &= epsV \\ \varepsilon_F &= epsF \\ \varepsilon_T &= epsT \\ \varepsilon_{pH} &= epspH \\ \varepsilon_{DO_2} &= epsDO_2 \end{aligned} \right\} \rightarrow \text{Error values}$$

$$\left. \begin{aligned} \lambda_x &= lx \\ \lambda_s &= ls \\ \lambda_p &= lp \\ \lambda_V &= lV \end{aligned} \right\} \rightarrow \text{Conjugate variables}$$

$$\left. \begin{aligned} \delta_x &= dx \\ \delta_s &= ds \\ \delta_p &= dp \\ \delta_V &= dV \end{aligned} \right\} \rightarrow \text{Interconnections in time domain}$$

$$\left. \begin{aligned} e_{\lambda_x} &= elx \\ e_{\lambda_s} &= els \\ e_{\lambda_p} &= elp \\ e_{\lambda_V} &= elV \end{aligned} \right\} \rightarrow \text{Gradient for the conjugate variables}$$

$$\left. \begin{aligned} \frac{dm_{\max}}{dT} &\equiv \frac{\partial \mu_{\max}}{\partial T} \\ \frac{dK_s}{dT} &\equiv \frac{\partial K_s}{\partial T} \\ \frac{dY_{xs}}{dT} &\equiv \frac{\partial Y_{xs}}{\partial T} \\ \frac{dY_{ps}}{dT} &\equiv \frac{\partial Y_{ps}}{\partial T} \end{aligned} \right\} \rightarrow \text{Derivatives}$$

$$\left. \begin{aligned} m_{\max} &\equiv \mu_{\max} \\ K_s &\equiv K_s \\ Y_{xs} &\equiv Y_x/s \\ Y_{px} &\equiv Y_p/x \end{aligned} \right\} \rightarrow \text{Kinetic coefficients}$$

$$ff = \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n$$

$$\left. \begin{aligned} fx &\equiv \varphi_x \\ fs &\equiv \varphi_s \\ fp &\equiv \varphi_p \\ fV &\equiv \varphi_V \\ f1 &\equiv \varphi_1 \\ f2 &\equiv \varphi_2 \end{aligned} \right\} \rightarrow \text{Functions } \varphi$$

$$\left. \begin{aligned} eex &= \lambda_x - \mu_x e_{\lambda_x} \\ ees &= \lambda_s - \mu_s e_{\lambda_s} \\ eep &= \lambda_p - \mu_p e_{\lambda_p} \\ eeV &= \lambda_V - \mu_V e_{\lambda_V} \end{aligned} \right\} \rightarrow \text{From formulae}$$

$$ex = e_x$$

$$es = e_s$$

$$ep = e_p$$

$$eV = e_V$$

Chapter 1

The problem for real time optimal control of the fed-batch fermentation process

1. Introduction

This chapter describes the necessity of the research discussed in the dissertation and highlights comparison between the different approaches for modelling and control of fed-batch processes for the production of yeast. The aim and the objectives of the dissertation are stated and explained.

1.1 Awareness of the problem

The research is based on the process that has been investigated for the past decades. That is a fermentation process. Fermentation is the process that result in the formulation of alcohol or organic acids on the basis of growth of bacteria moulds or fungi on different nutritional media (Ahmed et al., 1982). Fermentation process is classified according to the mode that has been chosen for process operation that is batch, fed-batch or continuous. Fed-batch fermentation process for a production of yeast is considered in this study. The latter is for the production of yeast cells. Final product of fermentation includes foods, drinks, medicines, beer, etc.

It must be clear that the pursuit to maximise the productivity whilst minimising the production costs is not the only reason for the control engineer's interest in fed-batch fermentation process. (Johnson, 1985). These processes are not only important in industry but from the control point of view they are technically challenging. The non-stationary process dynamics, slow response, non-linearity, sudden unexplained changes and irregularities make it more interesting for researchers.

Even though the process have been investigated in the past it is not clear as to which kind of models are suitable for it, also the dependencies between input and output variables are not clearly distinguished. The variables that are significant are not clearly stated, also the way these variables influence the process is not clear enough nowadays. Another problem is the lack of on-line sensors for measuring the various variables in the reactor. (Flaus et al., 1989).

Changes in process physiochemical variables can make a lot of improvements in the final yield. Hence the proposed strategy is looking at addressing this possibility by monitoring and control of these variables at their optimal operating values, meaning improved final yield. Varying the feed rate makes a lot of improvements on the biomass production and it also reduces the Crab-tree effects. The proposed project is looking at overcoming the difficulties mentioned above by developing adaptive multi-layer strategy (Fisher et al., 1984), (Lefkowitz, 1966) for control in which the problem for modelling, parameter estimation, process optimisation and closed loop control will be designed and implemented in integrity in order to achieve maximum production of yeast at the end of fermentation process.

The research is a part of the joint project between Peninsula Technikon's Department of Electrical Engineering and the University of Cape Town's (UCT) Department of Chemical Engineering. The research and control developments are conducted at Peninsula Technikon. The experiments and control implementation part were conducted at UCT.

The developed project is based on the latest control technology (SA Instrumentation & Control Journal, 1999) using PC with LabVIEW™ software technology. The last challenge is to integrate the whole system i.e. DAQ and real time control system with the design methods for modelling, parameter estimation and optimal control determination, in order to build a fully automatic, adaptive and robust control system.

1.2 Necessity of a research work in the field of optimisation and control of fermentation process in the production of yeast

Fermentation is the intrinsic capability of the microorganism to perform complex chemical transformations upon compounds by means of the metabolic activity and the action of enzymes. These processes are performed in tanks, called bioreactors, or fermentors and are used in the production of foods, drinks, medicines, chemical products, etc. The fermentation processes have been investigated for the past decades (Pomerleau et al., 1995). Although the fermentation has its own quality control systems, sudden unexplained changes occur from time to time in characteristics of the process, which includes: fermentation time, attenuation, contamination, pH, Temperature, Dissolved Oxygen, volumetric flow rate profiles with time (Michael and Fikret, 1992). The fermentation processes are described by two groups of variables; biological: concentrations of substrate, biomass and product; and physiochemical: temperature, pH, DO_2 , agitation (Michael and Fikret, 1992).

The problem for control of fed-batch fermentation process is very important these days because of population increasing and industrial developments. The fermentation process has proved to have positive effects in many fields of domestic life and industry, but the fermentation process for a production of yeast still need deeper understanding and improvement furthermore this process has not been studied fully yet as an object of control. It is not clear:

- Which kind of models are convenient,
- Which are dependencies between input and outputs,
- Which process variables are the most significant and in which way they affect the quality of the process or its products,
- What the optimal combination of settings for these significant variables is, according to the goals posed on the process quality and the product quantity.
- There is a lack of methods for measuring of important variables, for modelling and parameter estimation and for optimisation and control calculation.

The study is overcoming the constraints highlighted above in the following way:

→ For the considered case the biological variables: biomass concentration $[x]$, substrate concentration $[s]$, and product concentration $[p]$ are described on the basis of the systematic approach using mass balance equations. On the same token equations, kinetic parameters μ_{\max} , K_s , $Y_{x/s}$ and $Y_{p/s}$ are obtained on the basis of the empirical approach with experiments and parameters estimation technique.

Two types of models of the process are considered. Model describing the behaviour only of physiochemical variables, by first order transfer function. Unstructured model describing the behaviour of the biological variables identifying the dependencies between physiochemical and biological variables.

→ The considered input variables are:

→ Inputs used as control signals

- Base/acid flow rate
- On/off heating energy
- Flow rate of oxygen
- Flow rate of substrate
- T
- pH
- DO_2 concentration.

→ Outputs and state space variables

- Concentration of biomass
- Concentration of substrate
- Concentration of product
- Volume of the reactor.

→ The influence of each physiochemical variable is describing behaviour of the biological variables by representation of the kinetic parameters as quadratic function of these variables.

→ The optimal settings of the process variables are obtained by solution of the problem for optimal control, together with the problem for local control in a scheme of repetitive optimisation based on real time model parameter estimation.

- Measurement only of the physiochemical variables is used for the adaptive, optimal control of the process.
- New methods for estimation and optimal control are developed and connected in a common control system.

1.3 Comparative analysis of the approaches for modelling and control of fed-batch fermentation process

The requirements for implementation of the model and the control of the fed-batch process are evident on the basis of the existing situation in industry. They can be used as criteria for comparison of the existing methods for modelling and control, as follow (Luyben, 1990):

- To use simple model, convenient for calculations in real time,
- To use simple methods for modelling and control, which do not require complex calculation and more expensive hardware.
- Complex sensors for measurements of biological variables not to be used,
- Control to be based on the available hardware in industry to minimize costs.
- The hardware and software system to be simple and user friendly as possible to allow easy dialog with the operator.

Modelling

For a long time, the modelling of the dependence of the specific growth rate in microbial growth systems with respect to the process components and the physiochemical conditions has been on active research area (Hwang et al., 1995). The modelling is based on using unstructured models. They give the most fundamental observations concerning microbial metabolic processes and can be considered a good approximation when the cell composition is time dependent or when the composition is time dependent or when the substrate concentration is high compared to the saturation concentration (Birol et al., 1998, Moser, 1985).

The model used for control of industry fermentor are often based on simple, unstructured models since the process computer will adjust the model parameters based on the response of the system to disturbances.

The mathematical description of the considered in the dissertation model is concerned with the basic metabolic processes of *Saccharomyces cerevisiae*. The possibility for glucose or ethanol inhibition on both yeast cell growth and ethanol production is considered. For ethanol fermentation, the utilisation of the carbon source cannot be split into two processes, namely production of biomass and formation of product, since ethanol is the product of energy metabolism associated directly with biomass growth. According to, the most important metabolic processes accompanying ethanol fermentation can be arranged into some fundamental types of reaction patterns:

- Catabolic pathways (breakdown of substrates into energy and small molecules).
- Anabolic pathways (Synthesis of precursors for biomass).
- Polymerisation of the precursors to biomass.

Maintenance metabolism keeping the cellular machinery operative.

Optimisation

In industry the aim of the process control for production of yeast is to maximize yield but very often the profiles of the feeding substrate for maximum yield may be contradictory with that for maximum productivity. In this case in (Belgardt and Juan, 1991) and (Gou et al., 1995) a general objective function is used for global process optimization. It is named profit function and it is defined as the gross profit of a process divided by the production period. For a given yeast plant, the gross profit may be evaluated by material and energy balances using some simplified metabolic model of the process. This approach requires parameter estimation, but maximizing the profit function is not based on formulation and solution of a problem for optimal control.

Determination of the feeding trajectory is done in different ways in existing literature. It can be done based on an open-loop optimization. If an existing mathematical model is at disposal (Meszaros and Boles, 1992). The feeding profile can be determined also according to different stages of the fermentation process (Patnaick, 1995), (Tartanousky et al., 1995) and (Johnson, 1993). These two approaches consider that the system can be exactly translated in a set of mass balance equations. However, due to a non-identified physiological problem of the cells the specific growth rate can be either higher or lower than the one that was previously established. The open-loop optimisation does not result in optimal operation. Adaptive and closed loop control is

needed to overcome the process of non-linearity and non-stationary behaviour. The literature reveals that only in a few cases the optimisation is combined with underlying feedback regulators.

Adaptive control

Usually adaptive control is realised as continuous updating the coefficients of a linear controller, (Williams et al., 1984) based on an on-line model parameter estimation and controller coefficient calculation. In this case the non-linear plant is considered as a sequence of pseudo-linear descriptions. The drawback of the approach is that the set points are kept constant and the non-linear dynamics are not taken into account (Isidori, 1995)

Most of the papers consider only one control parameter the feeding rate of the substrate. In the dissertation the control input is extended with the values of T, pH, DO₂. This makes control more flexible and corresponding to the characteristics of the process.

Since the fermentation processes are inherently non-linear the control design must be able to deal efficiently with process variations. There are in principle three different approaches

- Robust control design,
- Adaptive control,
- Non-linear control (Anderson et al., 1991).

The difficulty of the robust control method is that it requires a description of the model uncertainties, which reflects only the expected variations in process behaviour, and either too conservative or too optimistic control design will result.

The adaptive approach changes the model parameters in real time and design the controller parameters in real time. The non-linear approach uses the non-linear model of the process to design a non-linear controller. It however is very sensitive to parameter variations. The best results can be obtained from approach of combination of these three techniques. When non-linear model and non-linear linearising controllers are used there are two main approaches to cope with model uncertainties.

- Finding proper parameterisations of the model and its estimation in real time (Bastin and Dochan, 1990).

- The controller has to be tuned in such a way that it compensate for the errors in the feed forward path of the controller by sensitivity analysis (Claes and Van Impe, 1998a,b, Claes and Van Impe, 1999)

Feedback control of yeast production processes

Industrial fed-batch production is traditionally carried out in an open loop using precalculated substrate feeding profiles. During the last decade the improved production is obtained with feeding profiles that are calculated on-line in a feedback loop.

There are basically 3 reasons for the use of advanced control strategies in the fed-batch yeast production:

1. The conflict between yield and productivity (Njodzi, 2001).
2. The level of ethanol concentration can enhance the production of inhibitory substances.
3. The reproducibility of cultivations is an important factor for a good and or uniform quality of the yeast and is not easy to be obtained without efficient feedback control.

Numerous feedback control strategies have been described in the literature to solve these difficulties by using the substrate feed rate as the control action in order to: (Chen et al., 1992)

- To set the respiratory quotient RQ close to 1. RQ is a ratio between the carbon dioxide evolution rate (CER) and the oxygen uptake rate (OUR).
- To maintain the glucose concentration at a constant low level.
- To track an exponential profile for the amount of biomass.
- To set the overall specific growth rate μ_{tot} , or OUR or CER at a constant value.
- To keep the ethanol concentration at a constant level, or to track a given ethanol profile.

The most frequently used strategy is the first one.

The controller techniques used to realise these strategies are:

1. PID
 - a. Early attempts for feedback control were made with a variety of PID controllers using glucose feed rate as a control action. The performance of a

classical PID controller around a pre-calculated profile F^0 can be improved by calculating this profile on-line in order to reduce the disturbance on the control action (Keulers et al., 1993)

2. Linear adaptive techniques

- a. The next step is development of linear adaptive control in order to cope with the non-linearity, the non-stationary behaviour and the modelling uncertainties. The controllers are based on linearization around a set point and the non-linearity is interpreted by the time varying parameters of the model, which are adapted on-line.

In these applications the parameters of the linear model are estimated on-line, in general with a recursive identification algorithm. The estimated parameters are then used to design a linear controller at each sampling time. The design of the controller is based either on a quadratic cost function or on the pole placement principle (Williams et al., 1986)

3. Adaptive non-linear control

Since fed-batch systems are non-linear, it is more interesting to exploit in the design the non-linear structure and the available physical knowledge about the system. At the same time the kinetic parameters are highly uncertain. The adaptive non-linear control technical is very convenient (Pomerlean and Viel, 1992).

The results are applied to industrial fermentor without changing the tuning parameters. The non-linear linearising controller is designed. It is used with state estimator in order to realise the control action.

Predictive control

A predictive feedback control can also be set up to form a product or grow cells (Kleman et al., 1991). The control scheme is divided in two parts:

A feed forward component that predicts (according to some statistical method and the previous collected data) the "need" of a certain substrate measurable on-line, and a feedback controller, which corrects for minor errors in the predicted "need". The main advantage to this system is that the investigator does not need to know the metabolic constants for a given microorganism prior to growth of that organism in the system and it can be applied to any substrate that can be measured on-line, being particularly valuable in the minimisation of by-products.

Real time control

In the fermentation industry computer facilities have been used only off-line for acquisition of data and some mathematical analysis. Only few companies have installed computers for process control and optimisation for fermentation plants of manufacturing scales (Williams et al., 1984). By using fed-batch fermentation, industry takes advantage of the fact that the concentration of the limiting substrate may be maintained at a very low level, thus:

- Avoiding repressive effects of high substrate concentration.
- Controlling the organism's growth rate and consequently controlling the oxygen demand of the fermentation.

Saccharomyces cerevisiae is industrially produced using the fed-batch techniques so as to maintain the glucose at very low concentration, maximizing the biomass yield and minimizing the production of ethanol (MC Neil and Harvey, 1990), (Neway, 1990), (Standbury, et al., 1993)

Control of fermentation processes operated in fed-batch mode is discussed in several studies in the literature (Jorgensen and Jensen, 1989).

Advanced monitoring and control functions are performed by using the graphical programming environment called LabVIEW (Claus and Van Impe, 1998a) in a system for adaptive control of fed-batch yeast fermentation (Claus and Van Impe, 1999).

New directions in the control of fermentation processes

The research trend towards the exploitation of new methods capable of manipulating and utilising uncertain, qualitative and informal knowledge appeared recently. They are (Shimizu and Ye, 1995):

- Control based on expert identification of the physiology state of the cells. (Konstantino and Yoshida, 1989), (Shimizu et al., 1994). The problem is to identify adequately the physiological state on-line.
- *Fuzzy control*
It is applicable for biotechnological processes difficulty to be described by mathematical models. Different applications are described in (Nakamura et al., 1985) (Rosimeire et al., 2002), (Park et al., 1993). Combination of

optimal feed trajectory with fuzzy logic control is good achievement of real time process control (Jin et al., 1994).

→ Neuro and neuro-fuzzy control

The artificial neural networks (ANN) have ability of knowledge acquisition focusing on the application to the identification and control of chemical and biotechnological processes. ANN can approximate large classes of non-linear functions and are capable of adjusting dynamically to environment changes. The applications are done in (Karim and Rivera, 1992), (Shi and Shimizu, 1993).

→ Genetic algorithms

They have the ability of global optimisation and are used for on-line process optimisation. (Roubos et al., 2000).

Common characteristics of the process influencing its control

Yeast requires aerobic conditions to grow rapidly. Oxygen starvation leads to ethanol production with a reduction in cellular growth. An excess of nutrient will also produce ethanol even in the presence of sufficient oxygen.

The yeast production requires the control of different parameters at different stages of the fermentation. Two phases can be distinguished:

- A phase in which the substrate needs to be controlled so as to avoid by-products formation.
- A second stage in which, due to the high cell density, oxygen transfer is limiting and so this parameter is the one to be controlled above a critical value under which the cellular metabolism changes. (Bajpai and Lueke, 1990)
- $DO_2 \text{ min}$ is introduced to overcome the oxygen starvation.
- F_{max} is introduced to overcome the excess of nutrients.

The choice of each parameter to control the process is system dependent and the decision should be based on convenience and experimental data (Turner et al., 1994)

1.4 The project aim and statement of the problem

The main aim of the study is the developments of control strategy, methods, algorithms and programs for modelling and optimal control of fed-batch fermentation process for maximum production of yeast.

The work on the problem is decomposed into four researchable areas, i.e. (a) fermentation process study, (b) mathematical modelling (c) optimal control theory and (d) software and hardware implementation of the real time control system.

Two types of sub-problems are solved in each of these areas: Design sub-problems and real time implementation sub-problems.

1.4.1 Design sub-problems

The following are the main design sub-problems that have been investigated.

- ⇒ Strategy for adaptive and optimal control,
- ⇒ Methods of modelling,
- ⇒ Methods for parameter estimation,
- ⇒ Methods of modelling,
- ⇒ Methods for optimisation of the process dynamics behaviour,
- ⇒ Adjustment of the controller parameters.

1.4.2 Real time implementation sub-problems

The following are the main implementation sub-problems that have been investigated.

- ⇒ Design of the data acquisition system for the process variables,
- ⇒ Real time implementation of the developed DAQ,
- ⇒ Integration of the design problem with the real time control,
- ⇒ Real time implementation on the designated control actions.

1.5 The significance and objectives of the research project

The significance of the research is in its contribution to the following areas:

- ⇒ To the developments of mathematical models for fed-batch fermentation processes.
- ⇒ To the development of methods of control for non-linear, non-stationary systems.

- ⇒ To the real time implementation of the developed control system, using existing technology for realisation and applications in industry.
- ⇒ Implementation of the developed system in industry will enable the process to function at its optimal operating point, resulting in maximum production of yield at minimal cost and time.

Other than the contribution to the control theory and its application in industry is the use of the most latest and modern control technology in real time. The research is a joint between two tertiary institutions, which implies that expertise from both institutions is shared at high level enabling the vast progress in biological technology. The work on the project will give future possibilities to apply its results in different fields of industry, which will increase the production and improve the quality of the control strategies.

The project objectives are:

- ⇒ To identify important variables that influences the fermentation process.
- ⇒ To identify the relationship between all process variables, which will assist in the identification of the model.
- ⇒ To develop mathematical model analytically connecting the physiochemical and biological variables.
- ⇒ To develop strategy for optimal control of the fed-batch fermentation process for the production yeast.
- ⇒ To develop method, algorithm and programmes for parameter estimation of the developed model.
- ⇒ To develop a method, algorithm and program for the design of optimal control of the process on the basis of the developed model.
- ⇒ To develop PC based data acquisition and real time control of the process variables using Biostat lab scale fermentation unit.
- ⇒ To develop a program that will automatically co-ordinate all programs for the design and real time operation of the process.

1.6 Proposed strategy for control

The proposed control strategy is based on the characteristics of the process as an object of control and on the aims of control implementation. The strategy is based on combination of the optimal control (Roberts and Lin, 1991) with adaptive control approaches.

1.6.1 Layers of the control system

The control system is based on adaptive control strategy where the fermentation process is optimised according to a certain criterion and optimal control is realised by the PC (Morari, and Stephanopoulos, 1980). In the considered case the controller learns about the process by acquiring data from the process and keeps on updating a control model on-line and second, to implement the designed models and control. The three-layer control structure is used to solve the problem for modelling and optimal control first on design stage and second on control implementation stage.

There are three layers that make the proposed control strategy feasible Fig 1.1. (Sing and Titli, 1978)

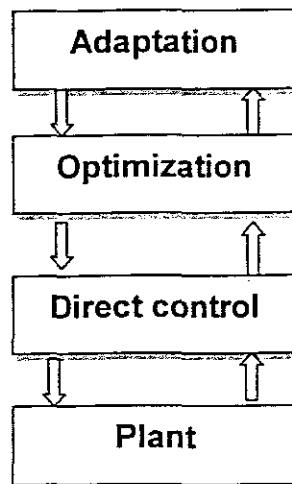


Figure 1.1: Three layers of the control structure

The problem solved on each layer is as follows:

→ Adaptation = model identification (model parameter estimation).

- Optimisation = optimisation based on the model parameters from the adaptation layer above, and design of the direct controller parameters.
- Direct control = realisation and implementation of the closed loop linear and non-linear control according to the parameter estimation from the adaptation layer and controller parameters and the reference trajectories from the optimisation layer.

1.6.2 Adaptation layer

Adaptive control is the name given to a control system in which the controller learns about the process by acquiring data from a certain process and keeps on updating a control model.

The latter is achieved by identifying the accurate mathematical model that represents the reaction and reactor environment. Using the control model we can progress beyond environmental control of bioreactors into the area of direct biological control.

1.6.2.1 Process modeling

Models are mathematical relationship between process variables. Traditionally the theoretical relationships provide the structure of the model. The latter is not applicable for biological process, because of the complexity of cellular processes and a larger number of environmental factors that affect cell culture.

Note: Models can be different, but they have one thing in common that they predict the output from a set of input.

Two types of unstructured models of the process are derived in the dissertation.

- ⇒ Model describing the behaviour of physiochemical variables.
- ⇒ Model describing the behaviour of the biological variables identifying the dependencies between physiochemical and biological variables.

1.6.2.2 Parameter estimation

The parameters of the model are determined on the bases of data from the process variables, control and state variables. Measurements and calculations are carried out on-line in real time. The obtained parameters are sent to the optimisation layer.

1.6.3 Optimization layer

The optimal control problem is solved using optimal control theory.

The optimisation layer is determined and based on the previous layer i.e. adaptation layer. The optimal operation or yield of the product is based on some criteria for the production of biomass, product, and some constraints over minimal and maximal values of the physiochemical variables. The obtained optimal trajectory of pH, temperature and DO₂ are used to determine the set points of the controllers from the direct control layer.

1.6.4 Direct control layer

This is the bottom layer. The optimal trajectory of the physiochemical variables from the optimisation layer above is passed through to this layer. On their basis and on the basis of the real measurements of the variables and the model of the physiochemical variables the optimal tunings of the controller parameters is calculated. Then the optimal trajectories and optimal settings are sent to the local controllers for the optimal control realisation.

The overall scheme for on-line implementation of the adaptive control strategy is presented in Fig1.2.

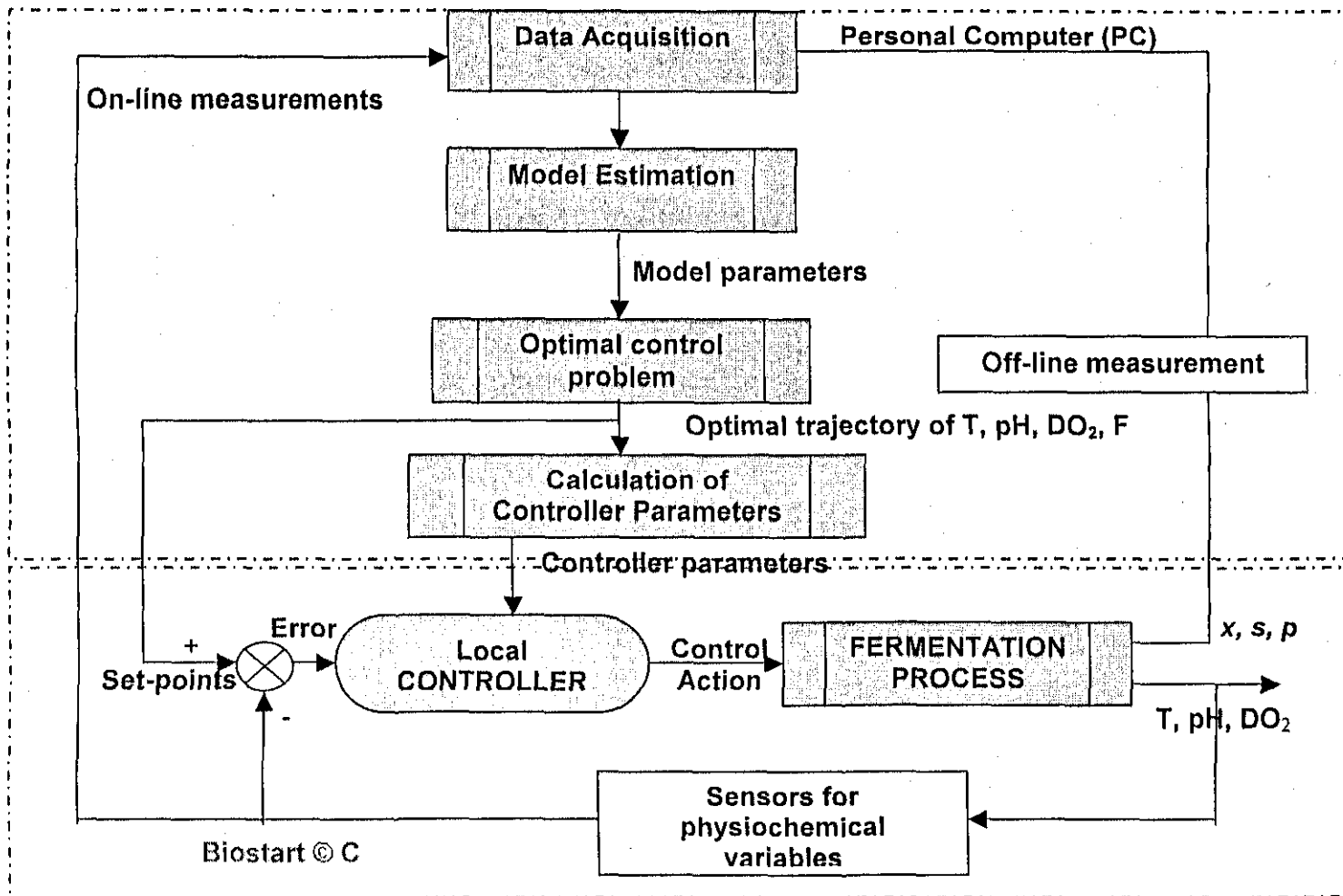


Figure 1.2: Control structure

The work of the scheme is based on the development of data acquisition system, algorithms and programs organising implementation of the programs for layer's problems and connection between layers.

1.7 Software

LabVIEW™ is used as driver software. LabVIEW™ is a graphical programming language called "G" and icons are dragged and grouped to define functionality of the program. LabVIEW™ programs are called virtual instruments (VI's).

The whole evolution of graphical programming language has made a tremendous increase in the developments of control technology in general (National Instruments, 1998). One of the advantages is that it is very easy to learn programming in "G" and also, it gives a good visual idea of how the final product will look like by so doing giving the operator an opportunity to give some of his views on the operators display. The former and latter would not have achieved with the text based programming language like C, C++ etc. The main drawback of graphical programming is the fact that processing time of programs is a bit slow. The latter does not make any difference in the developed application, because the fermentation processes are very slow. The execution time in most cases is affected by the way the source code is structured.

1.8 Conclusion

This chapter describes the necessity of the research and highlights comparison between the different approaches for modelling and control of fed-batch processes for the production of yeast. The aim and the objectives of the dissertation are stated and explained. The selected development and simulation software used is discussed.

Chapter 2

Process description

2. Introduction

This chapter describes the process as an object of control looking precisely at the influence of the physiochemical variables on the biological variables. The relationship is identified through the enzymes and the latter is discussed in the chapter. The results from previous experiments are discussed to illustrate the constraints associated with the control of the process under study.

2.1 Fermentation process

The fermentation process takes place in tanks called bioreactors or fermentors. Fermentation processes are complex, dynamic autocatalytic processes in which transformation takes place caused by biological catalysts called enzymes.

Enzymes are nothing but proteins of high molecular weight that act as catalysts. They are substrate specific, versatile, and very effective biological catalysts resulting in much higher reaction rates as compared to chemically catalysed reactions under ambient temperature. Enzymes are classified according to the reaction they catalyse. Enzymes decrease the activation energy $\{E\}$ of the reaction catalysed by binding the substrate and forming an enzyme-substrate $\{ES\}$ complex. Molecular aspects of enzymes-substrate interaction are not yet fully understood, but yet the enzymes require optimal conditions for pH, temperature, ionic strength, etc for their maximum activity. (Michel et al., 1992).

The technology of the fermentation is as follows:

The nutrients are added and the cells are inoculated into the fermentation vessel, environmental conditions such as pH, temperature, DO_2 , etc are kept convenient. The cells starts multiplying forming mass cells called biomass, and form other substances called secondary metabolites, or products. There are three different modes of the fermentation processes.

⇒ **Batch fermentation:**

The input (feed-nutrients) to the process is added during the initial stage of the process and the product (process output) is not removed until the process is finished. The main disadvantage of the batch process is that it is characterised by high down time between batches, comprising the charge and discharge of the fermentor vessel, the cleaning, sterilisation and the re-start of the whole process.

⇒ **Fed-batch fermentation:**

The input to the process is continuously added from the initial stage through the end of the process, but the process output is not removed until the process is finished.

⇒ **Continuous fermentation:**

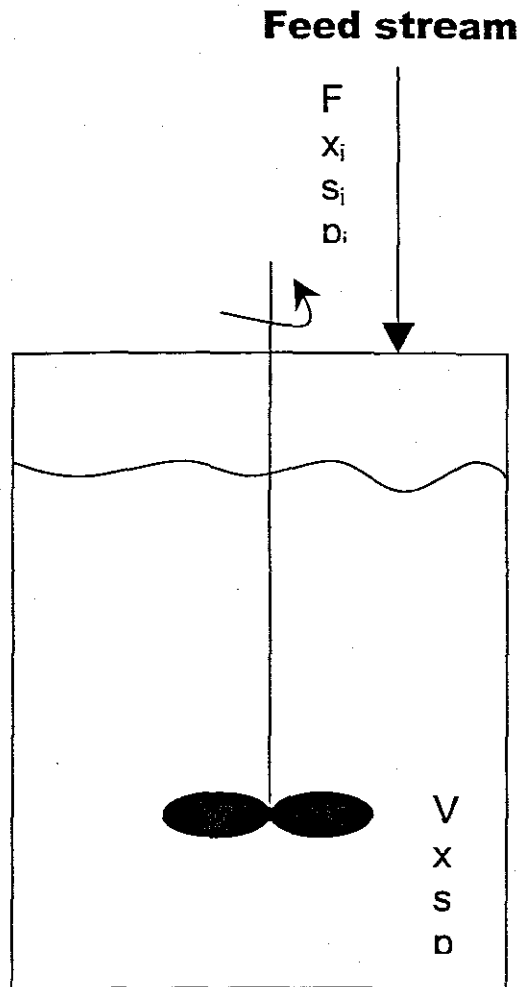
The input to the process is continuously added and the process output is continuously removed making it more economic compared to the other two type mentioned above.

2.2. Fed-batch fermentation process

The three processes mentioned above have their advantages and disadvantages (University of Maryland Baltimore County), but the one that interests most of the control engineers is the *Fed-batch fermentation process*. Yeast producers originally devised the fed-batch technique in the early 1900s to regulate the growth in batch culture of *Saccharomyces cerevisiae*.

The fed-batch processes start with the cells being grown under the batch movement for some time, usually until close to the end of the exponential growth phase. At this point solution of substrate (nutrients) is feed into the reactor, without the removal of culture fluids. This feed should be balanced enough to keep the growth of the microorganisms at a desired specific growth rate and reducing simultaneously the production of by-products. During the growth phase nutrients are added (Flow rate - F) when needed and acid and base is added to control the level of pH. If the cells are grown in an aerobic condition, using appropriate meters the oxygen flow rate can be measured. Also the rate into which the oxygen is supplied can be controlled.

Temperature is also one of the variables that need to be controlled precisely. There are many variables that influence the operation of the process. The ones listed above are just the few that can be measured on-line. The diagram of the fed-batch process is given on Fig 2.1.



- F \Rightarrow Feed rate for nutrient substrate in $[m^3 / s]$,
 x_i \Rightarrow Biomass concentration input flow in $[mg / l]$,
 s_i \Rightarrow Substrate concentration input at initial moment $s_i(0)=s_i=\text{constant}$ in $[mg / l]$,
 p_i \Rightarrow product concentration input flow, dt initial moment $p(0)=0$ in $[mg / l]$,
 V \Rightarrow Culture volume in the reactor vessel, at initial moment $V(0)=V_0$ in $[l]$,
 x \Rightarrow Biomass concentration, at initial moment $x(0)=x_0$ in $[mg / l]$,
 s \Rightarrow Substrate concentration, at initial moment $s(0)=s_0$ in $[mg / l]$,
 p \Rightarrow Product concentration, in the fermentor $[mg / l]$

Figure 2.1 Diagram for stirred fed-batch fermentor

2.2 Fed-batch fermentation for production of yeast

Yeast is widely used in brewing, baking, wine industry, research etc. The optimal operation of the processes discussed above depends on the efficient inoculum's development stage. The dissertation is considering the growth of yeast cells using the fed-batch fermentation process. The study is looking at the optimal operation of the fermentation of yeast used to inoculate fermentation of beer. The aim is to maximise the production of yeast. Yeast grows in two different cultures, presence of oxygen (Aerobic culture) and absence of oxygen (Anaerobic culture).

2.3.1 Aerobic fed-batch culture

The fed-batch operation requires the identification of the control variables and their respective limits. The oxygen availability have a great impact on the biomass, even though it is inoculated in excess it may inhibit the growth rate.

2.3.2 Anaerobic fed-batch culture

Saccharomyces cerevisiae is able to grow in the absence of oxygen i.e. anaerobic fed-batch culture. The maximum specific growth is lower in anaerobic growth (Njodzi, 2001). The latter resulted in the anaerobic fed batch culture not to be considered in this case. Anaerobic propagation is not ideal for inoculums development and is used and is used for fermentation processes.

2.3.3 Variables characterizing growth kinetics of yeast.

The fermentation process is characterised by three key groups of variables Table 3.1. Most of the physiochemical variables can be measured very easy in real time with the available sensors. In contrast biological variables cannot be measured on-line with the sensors that are available. The investigation and experiments illustrates that physiological variables affect biological variables in a certain way, and by identifying the optimal conditions of the physiological variables, will conclude the optimal conditions for biological variables through the enzymes link (Dochan, 1999).

Physical	Chemical	Biological
⇒ Temperature	PH	Biomass concentration
⇒ Pressure	Dissolved O ₂ (DO ₂)	Enzyme concentration
⇒ Reactor weight	Dissolved CO ₂	Biomass composition
⇒ Liquid level	Redox potential	(such as DNA, RNA,
⇒ Foam level	Exit gas composition	Protein, ATP/ADP/AMP,
⇒ Agitator speed	Conductivity	NAD/NADH levels)
⇒ Power consumption	Broth composition	Viability
⇒ Gas flow rate	(substrate, product, ion	Morphology
⇒ Medium flow rate	Concentration, etc.)	
⇒ Culture viscosity		
⇒ Gas hold-up		

Table 2.1 Variables measured or controlled in bioreactors for yeast production.

The table above illustrates clearly that there is a lot of factors that affect the growth of the yeast cells. The study does not consider all the variables mentioned in table 3.1 as the latter may lead into the process model being complex and impossible to implement it in real time with the available software and hardware. For the purpose of the dissertation and from the control point of view the following notation and variables are used in the dissertation.

- *Physiochemical variables used for influencing the process: Temperature (T), pH, Dissolved Oxygen (DO₂) and Flow rate (F).*
- *Biological variables that is biomass (x), substrate (s), and product (p) concentrations.*

The substrate and product are considered in connection to the biomass and because of that they are considered as biological variables.

2.3.3.1 The influence of temperature

Growth of yeast cells is affected by fluctuations in temperature. The latter implies that an optimal fermentation temperature has to be identified. Optimal temperatures are classified according to the groups of the microorganisms to which they are applied as

- ($T_{opt} < 20^{\circ}C$), Psychrophiles,

→ (T_{opt} = between 20°C to 50°C), Mesophiles, and

→ (T_{opt} => 50°C), Thermophiles.

Looking at temperature against growth rate, it is concluded that temperature can influence biological process by kinetic effects on reaction rate and catalytic effects on the activity of enzymes.

As the temperature increases towards the optimal temperature T_{opt} , μ the growth rate approximately doubles for every 10°C increase in temperature.

Above the optimal temperature range, the growth rate decreases and the thermal death may occur due to high temperatures.

Net growth rate can be expressed above the optimal temperature as follows:

$$\frac{dx(t)}{dt} = (\mu - k_d)x \quad (2.1)$$

Where μ is the growth rate and k_d is the death rate.

At high temperatures, the thermal death rate exceeds the growth rate, which result in a decrease in the concentration of viable cells.

Both μ and k_d vary with temperature according to the Arrhenius equations

$$\mu = Ae^{-E_a/RT}, k_d = A'e^{-E_d/RT} \quad (2.2)$$

Where E_a and E_d are activation energy for growth and thermal death. Activation energy for growth rate is typically 10 to 20 [kcal/mol] and for thermal death, 60 to 80 [kcal/mol] implying that the thermal death is more sensitive to temperature changes compared with microbial growth.

Looking at temperature against product formed, the optimal temperature for growth and product formation may be different. Growth rate and product formation are not the only ones affected by the temperature but the yield co-efficient $Y_{x/s}$ is also affected. As with the product formation as the temperature increases above the optimal temperature, the maintenance requirements of the cell increases. That is, the maintenance coefficient increases with increasing temperature with activation energy of 15 to 20 [kcal/mol], resulting in a decrease in the yield coefficient.

2.3.3.2 The influence of pH

Industry in general has a tendency of keeping some of controllable variables constant. pH being one of those variables.

Hydrogen ion concentration (pH) affects the activity of the enzymes and therefore the microbial growth rate. Same with the temperature the optimal pH for growth rate may be different from that for product formation. Different organisms have different pH optimal values and they vary about the optimum by +-1 to 2 pH units.

Yeast optimal pH varies between 3 and 6. Note, when the pH differs from the optimal value, the maintenance energy requirements increases.

Nitrogen source affects the increase or decrease in pH level. The latter is due to the fact that hydrogen ions are either released or removed from the medium, resulting in pH variations. Also, pH can change because of the production of organic acids, the utilization of acids or the production of bases. Thus the control and monitoring of pH is very important.

2.3.3.3 The influence dissolved oxygen (DO₂)

Microorganisms need oxygen to develop properly (Njodzi, 2001). *Saccharomyces C.* can grow in the presence or absence of oxygen that is, either in aerobic or anaerobic conditions. The latter implies that dissolved oxygen (DO₂) concentration has to be monitored and most of all controlled in the aerobic condition. This is because the concentration of DO₂ affects the final biomass yield, growth rate and other by products and may be the limiting substrate, since oxygen is a sparingly soluble gas in water. Nutrients flow rate (F) and oxygen transfer rate (OTR) as well as other control physiochemical variables needs to be controlled and maintained at a specific optimal value such that optimal results are obtained as discussed in the paragraph above.

The dissolved oxygen is represented by the following mass balance equation in the bioreactor:

$$\frac{dC}{dt} = OTR - OUR - DO_2 \quad (2.3)$$

Where:

C is the DO₂ concentration

DO₂ is the DO concentration in the reactor.

OTR is the oxygen transfer rate.

OUR is the oxygen uptake rate.

OUR depends on the biomass growth and is given by

$$OUR = k_3 \mu X \quad (2.4)$$

with k_3 being yield coefficient. A term $k_r x$ is sometimes included to account for the endogenous respiration.

$$OUR = k_3 \mu x + k_r x \quad (2.5)$$

By using a line of reasoning based on Henry's law to model the liquid gas transfer dynamics the OTR becomes:

$$OTR = k_L a (C_s - C) \quad (2.6)$$

Where $k_L a$ = mass transfer coefficient and C_s being oxygen saturation coefficient.

There are some constraints associated with mass balance equation. It is often useless because the coefficient C_s and $k_L a$ may be unknown and highly time varying. The literature reveals that oxygen saturation concentration C_s depends on variables such as oxygen partial pressure in surrounding atmosphere, temperature, salinity and concentration of surface in the liquid and other factors such as the type and the geometry of the reactor or the airflow rate determines the value of $k_L a$.

Industry tries to overcome the constraints above by measuring on-line input and output gaseous flow rates. Hence, if the liquid-gas transfer dynamics are negligible as often assumed, the OTR can simply be expressed from the gaseous oxygen balance:

$$OTR = Q_1 - Q_2 \quad (2.7)$$

Where $Q_1 - Q_2$ are respectively the input and output oxygen flow rates (per volume unit). The latter has its own constraints in this study, because the sensors to measure $Q_1 - Q_2$ are not available.

The study has overcome the constraints highlighted above by measuring on-line dissolved oxygen concentration (DO_2). The measured DO_2 is one of the parameters of the developed model of the process.

2.3.4 Results in yeast production caused by the influence of different variables.

The observation from the different experiments showed the following:

- It has been observed that at high concentration substrate (malt) a by-product ethanol is produced (inhibition factor) while in low concentration of malt, the yeast growth is restricted (limiting factor) (Njodzi, 2001).
- The result from the production of ethanol during aerobic growth at high sugar concentration is called Crab-tree effects or overflow metabolism. The latter reduces the biomass production.

- Also oxygen is proportional to the yeast propagation process (biomass yield), on the other hand very high dissolved oxygen (DO_2) concentration may induce stress on yeast that may lead to lower biomass productivity.
- Quality of fermentation is also affected by the quality of the yeast used for inoculation.

2.3.5 Incorporation of the physiochemical variables in the mathematical model

The variables characterising the fermentation process for production of yeast can be grouped according to their role in the process of controlling the biomass and production as described in chapter 1 in the following way:

- Inputs used as control signals:
 - Base/acid flow rate, On/off heating energy, Flow rate of oxygen, Flow rate of substrate, T, pH, DO_2 concentration.
- Outputs and state space variables:
 - Concentration of biomass, Concentration of substrate, Concentration of product, Volume of the reactor.

The technology of the process and the influence between the variables determine the dynamic behaviour of the fermentation.

The model developed in the dissertation is based on the:

- a) Possibility to measure the physiochemical variables,
- b) Introducing the influence of the physiochemical variables by mathematical expression into the non structured model of the process, and
- c) Possibility to control the biological variables through changes of the physiochemical variables.

This model is presented in chapter 3.

2.3 Conclusion

Fed-batch fermentation for the production of yeast like object of control looking precisely looking at the influence of the physiochemical variables on the biological variables is presented and described in the chapter. The relationship is identified through the enzymes. The results from previous experiments are discussed to illustrate the constraints associated with the control of the process under study.

Chapter 3

Layer for adaptation

Process modelling

3. Introduction

This chapter describes the different types of models as applicable to the dissertation; the comparison between different types is highlighted. The derivation of the developed yeast model using mass balance equations and rate laws is discussed and presented. The problem for model simulation is solved using Matlab and LabVIEW™ programs and the programs are described at the end of the chapter. LabVIEW program is used for the realization of the problem in real time.

3.1 Process modelling

To fulfil the adaptive control an accurate mathematical model representing the reaction and reactor environment has to be identified. The model is used as a tool for process analysis, for the design of control systems and for realization of the adaptive control. The question that rises is how to obtain a good process model. In this regard there are three fundamental issues:

- Model formulation,
- Parameter estimation and
- Model validation.

Traditionally the theoretical relationships provide the structure of the model. The latter is not applicable for a biological process, because of the complexity of cellular processes and a larger number of environmental factors that affect cell culture. As a result of these biological models are developed on the basis of observations rather than theoretical laws of science (Doran et al., 1995).

3.2 Model types

A model fundamental result is to obtain functional relationship between various process variables that explains the behaviour of the observed process. The underlying belief is that a true but unknown fundamental relationship dictated by natural laws

exists between process variables. The modelling exercise is directed towards discovering this relationship. Traditionally, models are based on a combination of 'theoretical' relationships, which provide the structure of the model, and experimental observations, which provide the numerical values of the model coefficients. The above is very difficult when it comes to the biological process; this is due to the complexity of cellular process and also the number of environmental factors, which affect the cell structure.

There are different types of models, there is:

- Theoretical models
 - These models are identified using the systematic application of natural laws in order to determine the functional relationship between the process variables in the observed process.
- Empirical (Experimental) models
 - In contrast the empirical models find a way to approximate the functional relationship by some usually simple mathematical functions, using the data acquired during experiments, hence they are called *empirical models*.
- Semi-theoretical
 - These models are developed using both systematic and empirical approach. The developed model is semi theoretical in the sense that both systematic approach and empirical approach are applied in solving the problem under study.

From the three types of models described above it is evident that, the application into which the model has to be derived determines the model type that needs to be used. The availability of the process measurements also determines which model will be appropriate for the specific application. It is common to adopt the theoretical model when the process under study is well understood. On the other hand if the process is too complicated {implying not well understood} or the model equations are extremely complex, the empirical approach is the most appropriate one.

The main objectives of this section is to identify the model by finding the functional relationship highlighted above using the natural laws that exists between process variables in the observed process data.

For the considered case the biological variables x , s , and p are described on the basis of the systematic approach using mass balance equations. On the same token equation parameters μ_{\max} , K_s , $Y_{x/s}$ and $Y_{p/s}$ are obtained on the basis of the empirical approach with experiments and parameters estimation technique.

Two types of models of the process are considered.

- Model describing the behaviour of physiochemical variables.
- Unstructured model describing the behaviour of the biological variables identifying the dependencies between physiochemical and biological variables.

3.3 Model describing the behaviour of physiochemical variables.

The first model is described by the first order transfer function with time delays.

$$G(s) = \frac{k^{ph} e^{-\tau s}}{T_{ph}s + 1} \quad (3.1)$$

where K is the gain, τ is the time delay, T is the time constant.

The inputs to these functions are the flow rate of acid or base, heat source, on/off, flow rate of dissolved oxygen (DO_2), rate of electrical motor with their respective outputs: pH, temperature, dissolved oxygen concentration, number of turns of the stirrer in revolutions per minute (rpm). Fig 3.1.

Equation (3.1) has the same structure for all the physiochemical variables listed above, only the parameters K , τ , T are different.

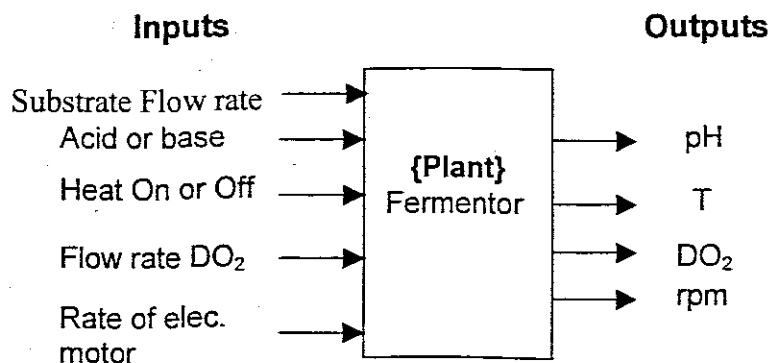


Figure 3.1 Model 1 inputs and their respective outputs

The model described above is used for optimal tuning the controllers on each physiochemical variable on the bio-controller. Parameter estimation for this model is not done in the dissertation separately as it is included as a part of the used program for auto tuning of PID controller described in chapter 5.

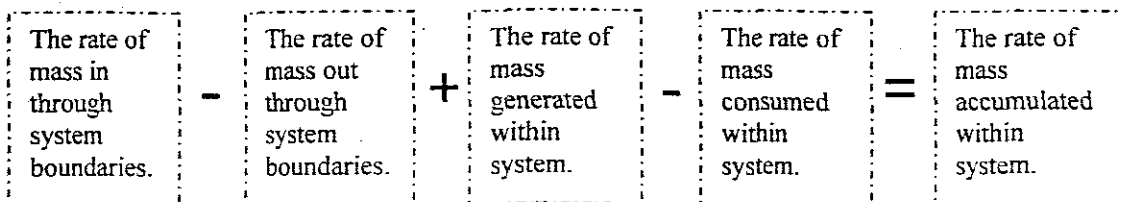
3.4 Model describing mass balance equations of the fed batch fermentation process

The second model is described on the basis of mass balance equations. These equations describe the dynamics of biological variables, concentrations of substrate (s), biomass (x) and product (p) and of the reactor volume (V). The equations are fundamentally derived from the law of conservation of mass that says, "Mass is conserved in ordinary chemical and physical processes". The growth rate dependencies are based on the Monod (1949) kinetics where sugar uptake is directly dependent on sugar and enzymes concentration.

The schematic diagram of the fed-batch fermentation is given on Fig 2.1.

The process starts with relatively dilute solution of substrate and more nutrients are added as the process progresses. The rate into which the nutrients are added is controlled to avoid high concentration of substrate as this may inhibit or switch on undesirable metabolic pathways (Jens et al., 2002). In cultures where oxygen is used to grow the yeast cells (aerobic conditions), more oxygen is pumped to the process increasing the growth rate. From the other side high growth rate of biomass can inhibit the biomass yield by bigger amounts of produced products e.g. ethanol.

The mass balance equations are derived from the basic laws of conservation of mass represented by the following general equation:



$$\frac{dM(t)}{dt} = \hat{M}_i(t) - \hat{M}_o(t) + R_G(t) - R_C(t) \quad (3.2)$$

Where:

M is the mass accumulated within the system

\hat{M}_i is the mass flow rate of substance "A" entering the system.

\hat{M}_o is the mass flow rate of "A" leaving the system.

R_G is the mass rate of generation of species "A" by chemical reaction.

R_C is the mass rate of consumption "A" by reaction.

The dimensions of \hat{M}_i , \hat{M}_o , R_G and R_C are mass per unit time with units such as g s^{-1} , kg h^{-1} , lb min^{-1} etc. All the variables can vary with time.

Using mass balance equations the theoretical model representing the concentration of biomass yield, product and substrate is described. The equations are obtained sequentially for the biological variables i.e. biomass, substrate and product.

3.4.1 Expressions for growth and production rates

3.4.1.1 Growth rate laws

The letter t used for time dependence is missing in this part, but all variables are time dependent. The common expression for the cell growth in the batch reactor is given by the equation

$$\text{Cells} + \text{substrate} = \text{more cells} + \text{product} \quad (3.3)$$

The growth of cells in this equation can be expressed mathematically by the cell growth rate with which the new cells are formed.

$$r_g = \mu x \quad (3.4)$$

Where r_g is the cell growth, $\text{g/dm}^3 \cdot \text{s}$, x is the cell concentration g/dm^3 ; μ is the specific growth rate.

The most commonly used expression for the exponential growth of microorganisms is Monod's equation. The specific growth rate can be expressed as:

$$\mu = \mu_{\max} \frac{s}{K_s + s} \quad (3.5)$$

Where μ_{\max} is the maximum specific growth rate [s^{-1}], and K_s is called Monod constant or simply the substrate saturation constant. The importance of K_s is that,

when substrate concentration is numerically equal to K_s , growth rate is exactly half of the maximum growth rate. For a number of bacteria and fungi the constant K_s is small and $r_g = \mu_{\max}$.

In common case:

$$r_g = \frac{\mu_{\max} xS}{k_s + S} \quad (3.6)$$

In many systems the product inhibits the rate of growth.

The equation, which is used to represent the product inhibition on the growth rate, is: (Dochan et al., 1990)

$$r_g = k_{obs} \frac{\mu_{\max} xS}{k_s + S} = \left(1 - \frac{P}{P^*}\right)^n \frac{\mu_{\max} xS}{k_s + S} \quad (3.7)$$

$$\text{Where } k_{obs} = \left(1 - \frac{P}{P^*}\right)^n \quad (3.8)$$

P^* is the product concentration at which all metabolism cease, [mg/l], n is the empirical constant for *Saccharomyces cerevicerae*.

Other used equations for the description of cell growth are:

→ Tessier equation

$$r_g = \mu_{\max} \left(1 - \exp\left(-\frac{S}{k}\right)\right) \quad (3.9)$$

→ Moser equation

$$r_g = \frac{\mu_{\max} x}{(1 + k_s \exp(-\eta))} \quad (3.10)$$

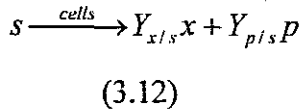
Where η and K_s are empirical constant determined by a best fit of the data. These equations have been found to better-fit experimental data at the beginning or at the end of fermentation.

The cell death rate is given by

$$r_d = k_d x \quad (3.11)$$

3.4.1.2 Stoichiometry

The stoichiometry for cell growth is very complex and varies with micro organism, nutrient system and environment conditions such as pH, Temperature, redox potential etc. The considerations are for cell growth, limited by only one nutrient in the medium. The equation (3.3) can be written in the following way



where the yield coefficient for the growth of biomass is:

$$Y_{x/s} = \frac{\text{mass_of_new_cells_formed}}{\text{mass_of_substrate_consumed_to_produce_new_cells}} \quad (3.13)$$

The stoichiometric yield coefficient that relates the amount of product formed per mass of substrate consumed is:

$$Y_{p/s} = \frac{\text{mass_of_product_formed}}{\text{mass_of_substrate_consumed_to_form_product}} \quad (3.14)$$

In addition to consuming substrate to produce new cells, part of the substrate must be just to maintain a cell's daily activities. The corresponding maintenance utilization term is:

$$m = \frac{\text{mass_of_substrate_consumed_for_maintenance}}{\text{mass_of_cells_} * \text{time}} \quad (3.15)$$

The rate of substrate consumption for maintenance whether or not cells are growing is:

$$r_{sm} = mx \quad (3.16)$$

Product formation can take place during different phases of cell growth. When product is only produced during the growth phase then the rate of product formation is:

$$r_p = Y_{p/x}r_g \quad (3.17)$$

However when the product is produced during the stationary phase, then the product formation is related to substrate consumption by:

$$r_p = Y_{p/s}(-r_s) \quad (3.18)$$

The equation which relate the rate of nutrient consumption, $-r_s$, to the rates of cell growth, product generation and cell maintenance is:

$$\boxed{\text{Net Rate of substrate consumption}} = \boxed{\text{Rate consumed by cells}} + \boxed{\text{Rate consumed to form product}} + \boxed{\text{Rate consumed for maintenance}} \quad (3.19)$$

$$-r_s = Y_{s/x} r_g + Y_{s/p} r_p + mx \quad (3.20)$$

If the product is produced during the growth phase it may not be possible to separate out the amount of substrate consumed for growth from that consumed to produce the product. Under these circumstances all the substrate consumed is lumped into the stoichiometric coefficient, $Y_{s/x}$ and the rate of substrate disappearance is:

$$-r_s = Y_{s/x} r_g + ms \quad \rightarrow \text{product formation in the growth phase} \quad (3.21)$$

The corresponding rate of product formation is:

$$r_p = r_g Y_{p/x} \quad (3.22)$$

These equations cannot be used during the stationary phase. When the product is produced only during the stationary phase the nutrient consumed for growth has become virtually exhausted and a different nutrient is used for cell maintenance and to produce the desired product.

The net rate of substrate consumption during the stationary phase is:

$$-r_s = mx + \frac{Y_{s/p} K_p s_n x}{K_{sn} + s_n} \quad (3.23)$$

where:

s_n is the concentration of the secondary nutrient [g/m^3]

K_p is the specific rate constant with product, [s^{-1}],

x is the cell concentration [g/m^3],

K_{sn} is the constant, [gm/m^3],

$$r_p = Y_{p/sn} (-r_{sn}) \quad (3.24)$$

The third situation is when the product is formed during both logarithmic and stationary phase. This is the considered case of *Seccharomyces cerevicerae*.

The equations for the substrate consumption is

$$-r_s = Y_{s/x} r_g + Y_{s/p} r_p + mx \quad (3.25)$$

Substituting equation (3.23) into equation (3.20) is obtained:

$$-r_s = Y_{s/x} r_g + Y_{s/p} Y_{p/x} r_g + mx \quad (3.26)$$

After grouping in equation (3.26) it is obtained:

$$-r_s = [Y_{s/x} + Y_{s/p} Y_{p/x}] r_g + mx = Y_{s/x} r_g + mx = \frac{1}{Y_{x/s}} r_g + mx \quad (3.26)$$

This common presentation in equation (3.26) shows that the rate of substrate consumption for both cell growth and product formation can be presented by one common stoichiometric coefficient. The expressions for rate laws and stoichiometry are used in mass balance equations for full description of the process behavior.

3.4.2 Biomass mass balance equation

In fed-batch operation $\hat{M}_{ox} = 0$; \hat{M}_{ix} is equal to the feed flow rate F multiplied by the cell concentration x_i in the feed.

$$\hat{M}_{ix} = Fx_i = \text{mass of cells entering the reactor.}$$

$M_x = xV$, where x is the cell concentration and V is the liquid volume of the reactor vessel,

$$R_{Gx} = \mu xV, \text{ where: } \mu \text{ is the specific growth rate,}$$

$$R_{Cx} = k_d xV, \text{ where } k_d \text{ is the specific death rate constant.}$$

Substituting in equation (3.2) above the following is obtained:

$$\frac{d(xV)}{dt} = Fx_i - 0 + \mu xV - k_d xV$$

Note V is the function of time so it cannot be cancelled out, rather a product rule for derivative can be used.

$$x \frac{dV}{dt} + V \frac{dx}{dt} = Fx_i + (\mu - k_d)xV, \text{ Substituting flow rate } F \text{ with:} \quad (3.27)$$

$$F = \frac{dV}{dt} \quad (3.28)$$

$$xF + V \frac{dx}{dt} = Fx_i + (\mu - k_d)xV \quad (3.29)$$

$$\frac{dx}{dt} = \frac{F}{V} x_i + x(\mu - k_d - \frac{F}{V}) \quad (3.30)$$

Substituting dilution rate

$$D = \frac{F}{V} \quad (3.31)$$

$$\frac{dx(t)}{dt} = D(t)x_i(t) + x(t)(\mu(t) - k_d - D(t)), x(0) = x_0 \quad (3.32)$$

In fed-batch the volume increases with time, therefore if F is constant, D decreases as the reaction progresses.

Generally the feed material is sterile so that $x_i = 0$, also if cell death is negligible compared with growth so that $k_d \ll \mu$; the equation above becomes.

$$\frac{dx(t)}{dt} = x(t)(\mu(t) - D(t)), x(0) = x_0 \quad (3.33)$$

3.4.3 Substrate mass balance equation

When substrate concentration is limited in the fed-batch reactor the mass balance relations for it are:

$$\hat{M}_{os} = R_{Gs} = 0$$

\hat{M}_{is} = is the mass flow rate of substrate entering the system = $F S_i$.

M_s = is the mass of substrate in the reactor = sV .

$$R_{cs} = r_s V = \left(\frac{1}{Y_{x/s}^1} r_g + m_s \right) V, \quad (3.34)$$

The equation describing the substrate behaviour is:

$$\frac{d(sV)}{dt} = F S_i - \left(\frac{r_g}{Y_{x/s}^1} + Y_{s/p} r_p + m_s x \right) V \quad (3.35)$$

$$\frac{d(sV)}{dt} = F S_i - \left(\frac{r_g}{Y_{x/s}^1} + \frac{r_p}{Y_{s/p}} + m_s x \right) V \quad (3.36)$$

$$\frac{d(sV)}{dt} = F S_i - \left(\frac{\mu}{Y_{x/s}} + \frac{\mu}{Y_{s/p} Y_{p/s}} + m_s \right) x V \quad (3.37)$$

$$\frac{d(sV)}{dt} = F S_i - \left(\frac{\mu}{Y_{x/s}} + m_s \right) x V \quad (3.38)$$

where:

$$Y_{x/s}^1 = \frac{Y_{x/s}}{2}$$

$Y_{x/s}$ is the true biomass yield from the substrate.

- r_p is the rate of product formation.
 $Y_{p/s}$ is the true product yield from the substrate.
 m_s is the maintenance co-efficient.
 r_s is the volumetric rate of substrate uptake

Expanding equation (3.10) and substituting $F = \frac{dV}{dt}$ it is obtained:

$$\frac{ds}{dt} = D(s_i - s) - (Y_{s/x}\mu + Y_{s/p}\mu + m_s)x, s(0) = s_0 \quad (3.39)$$

$$\frac{ds}{dt} = D(s_i - s) - \left(\frac{\mu}{Y_{x/s}} + m_s\right)x \quad (3.40)$$

3.4.4 Product mass balance equation

When product concentration is considered in the fed-batch reactor the mass balance relations for it are:

\hat{M}_{ip} = is the mass flow rate of product entering the system = Fp_i .

M_p = is the mass of the product in the reactor = pV .

\hat{M}_{op} = is the mass flow rate of product output = 0.

$$R_{Gp} = r_g V$$

$$R_{Cp} = 0$$

$$\frac{dM}{dt} = \frac{d(pV)}{dt} = p_i F + r_g V \quad (3.41)$$

$$V \frac{dp}{dt} = p_i F + r_g V - pF = r_g V + F(p_i - p) \quad (3.42)$$

$$\frac{dp}{dt} = r_g + \frac{F}{V}(p_i - p) = Y_{p/x}\mu x + \frac{F}{V}(p_i - p) \quad (3.43)$$

Normally $p_i = 0$ for sterile input flow F and then

$$\frac{dp}{dt} = Y_{p/x}\mu x + \frac{F}{V}p \quad (3.44)$$

3.4.5 Volume mass balance equation

In fed-batch process M_v is the volume V , where

$$M_v = V$$

$$M_{iv} = F$$

$$\hat{M}_{ov} = 0$$

$$R_{GV} = 0$$

$$R_{CV} = 0, \text{ then}$$

$$\frac{dM}{dt} = \frac{dV(t)}{dt} = F(t), V(0) = V_0, V(t_f) = V_{\max}$$

3.4.6 Summary of derived mass balance equations

The set of mass balance equations in function of time determines the process behaviour. Then the model of the process is:

$$\dot{x}(t) = x(t)(\mu(t) - k_d - D(t)), x(0) = x_0 \quad (3.45)$$

$$\dot{s}(t) = D(t)(s_i(t) - s(t)) - x(t)\left(\frac{2\mu(t)}{Y_{x/s}} + m_s\right), s(0) = s_0 \quad (3.46)$$

$$\dot{p}(t) = Y_{p/x}\mu(t)x(t) + Dp(k) \quad (3.47)$$

$$\dot{V}(t) = F(t), V(0) = V_0 \quad (3.48)$$

$$D(t) = \frac{F(t)}{V(t)} \quad (3.49)$$

$$\mu(t) = \frac{\mu_m(t)s(t)}{K_s + s(t)} \quad (3.50)$$

For inhibition of the cell growth by the ethanol production, the Monod equation can be written as:

$$\mu(t) = \left(1 - \frac{p(t)}{p^*}\right) \frac{\mu_m(t)s(t)}{K_s + s(t)} \quad (3.51)$$

Once the values of μ_{\max} , K_s , $Y_{x/s}$ and $Y_{p/s}$ have been determined, the model is used to estimate cell biomass, substrate and product concentration as a function of time. The main problem with the fermentation process is that model parameters are difficult to be calculated because they change with time.

3.5 Incorporation of physiochemical variables into the biological model

The influence of physiochemical variables i.e. T, pH, DO₂ is very important for the process behaviour. These variables influence the cell enzymes and through them influence the biomass growth or product formation. The complex relationship between physiochemical and biological variables and sudden unexplained changes in the process is reflected in the values of the equation kinetic parameters μ_{\max} , K_s , $Y_{x/s}$ and $Y_{p/x}$. They can be represented as functions of the physiochemical variables in the following way:

$$\mu_{\max} = a_1 + a_2T + a_3T^2 + a_4pH + a_5pH^2 + a_6DO_2 + a_7D0^2 \quad (3.52)$$

$$k_s = b_1 + b_2T + b_3T^2 + b_4pH + b_5pH^2 + b_6DO_2 + b_7D0^2 \quad (3.53)$$

$$Y_{x/s} = c_1 + c_2T + c_3T^2 + c_4pH + c_5pH^2 + c_6DO_2 + c_7D0^2 \quad (3.54)$$

$$Y_{p/x} = d_1 + d_2T + d_3T^2 + d_4pH + d_5pH^2 + d_6DO_2 + d_7D0^2 \quad (3.55)$$

The mass balance equations (3.45)-(3.48) together with equations (3.52)-(3.55) represent the model that is used in the dissertation for process optimisation and control. Input variables for this model are the physiochemical variables T, pH, DO₂, output variables are the biological variables x, s, p, where p is the product of ethanol giving inhibitory effect on the cell growth.

The coefficients values of $a_1 \rightarrow a_7$, $b_1 \rightarrow b_7$, $c_1 \rightarrow c_7$, and $d_1 \rightarrow d_7$ are not known and also they cannot be determined theoretically. As a result of the latter, these coefficients are determined on the basis of experimental data and application of the least mean square [LSM] approach for the parameter estimation.

In equations (3.52)-(3.55) the value of the coefficients k_d and m are determined previously. Usually these values are very small approximately $k_d = 0.01\text{hr}^{-1}$, $m = 0.03$ [g of substrate/g.cells.hr].

Model (3.52)-(3.55) is a nonlinear one according to both biological variables and coefficients.

3.6 Discrete representation of the model

In order to do calculations easily in the PC, model equations are discretized using difference equations for representation of the derivatives. The model is then described by difference equations.

Representation of the derivative:

$$\dot{x}(t) = \frac{x(k+1) - x(k)}{\Delta t}, \quad (3.56)$$

where Δt is a sampling period. The number of steps in the period of fermentation is $K = t_f/\Delta t$, where t_f is the end time of the fermentation.

Discrete model is obtained after substitution equation (3.56) into the continuous model equations (3.45)-(3.48).

Discrete model:

$$\frac{x(k+1) - x(k)}{\Delta t} = \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - k_d x(k) - \frac{F(k)x(k)}{V(k)}, x(0) = x_0 \quad (3.57)$$

$$\frac{s(k+1) - s(k)}{\Delta t} = -\frac{\mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - mx(k) + \frac{F(k)}{V(k)} [s_i - s(k)], s(0) = s_0 \quad (3.58)$$

$$\frac{p(k+1) - p(k)}{\Delta t} = \mu_{\max} Y_{p/x} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \Delta t \frac{F(k)p(k)}{V(k)}, p(0) = 0 \quad (3.59)$$

$$\frac{V(k+1) - V(k)}{\Delta t} = F(k), V(0) = V_0, V(k) = V_{\max} \quad (3.60)$$

The above equations can be written in the following way:

$$x(k+1) = x(k) + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \Delta t k_d x(k) - \Delta t \frac{F(k)x(k)}{V(k)}, x(0) = x_0$$

(3.61)

$$s(k+1) = s(k) - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \Delta t m x(k) + \Delta t \frac{F(k)}{V(k)} [s_i - s(k)], s(0) = s_0$$

(3.62)

$$p(k+1) = p(k) + \Delta t \mu_{\max} Y_{p/x} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \Delta t \frac{F(k)p(k)}{V(k)}, p(0) = 0$$

(3.63)

$$V(k+1) = V(k) + \Delta t F(k), V(0) = V_0, V(k) = V_{\max}$$

(3.64)

$$D(k) = \frac{F(k)}{V(k)}$$

(3.65)

where the kinetic parameters $\mu_{\max}, K_s, Y_{x/s}, Y_{p/x}$ are described by the equations (3.52)-(3.55). The parameters $a_1 \rightarrow a_7, b_1 \rightarrow b_7, c_1 \rightarrow c_7$, and $d_1 \rightarrow d_7$ are unknown. They are determined on the basis of experimental data and estimation method in the next chapter.

3.7 Simulation of the model

Simulations of the model is realized using the packages:

- LabVIEW™6.01i
- MATLAB 6.1 / Simulink

These packages are used by scientist, engineers, researchers, industry etc for calculation, simulation and real time control. LabVIEW™ is the graphical programming language called “G”. Matlab is an acronym for MATrix. Simulink is an extension of Matlab that allows development of models using block diagrams notation.

To allow easy of testing the model is developed in both LabVIEW™ and MATLAB. The input parameters and constants are the same to both models. Table 3.1 and Table 3.2 have the input and output variables for both simulation packages and they are used in the next paragraphs.

Variable	Value	Description
TT	50	→ Number of samples
t	1:1:TT	→ Time [h]
F	0	→ Volumetric substrate flow rate [mg/l].
si	0	→ Input substrate [mg/l]
pH	6.7	→ pH
T	30	→ Temperature in degrees Celsius
O	1	→ Dissolved Oxygen (DO ₂) is %.
a1	0.08	→ Model parameter.
a2	0.03	→ Model parameter.
a3	0.01	→ Model parameter.
a4	0.02	→ Model parameter.
a5	0.002	→ Model parameter.
a6	0.001	→ Model parameter.
a7	0.002	→ Model parameter.
b1	0.2	→ Model parameter.
b2	0.03	→ Model parameter.
b3	0.01	→ Model parameter.
b4	0.01	→ Model parameter.
b5	0.001	→ Model parameter.
b6	0.002	→ Model parameter.
b7	0.001	→ Model parameter.
c1	2	→ Model parameter.
c2	0.3	→ Model parameter.
c3	0.021	→ Model parameter.
c4	0.051	→ Model parameter.
c5	0.00211	→ Model parameter.

c6	0.0211	→ Model parameter.
c7	0.002	→ Model parameter.
d1	0.004	→ Model parameter.
d2	0.006	→ Model parameter.
d3	0.001	→ Model parameter.
d4	0.005	→ Model parameter.
d5	0.0005	→ Model parameter.
d6	0.001	→ Model parameter.
d7	0.0002	→ Model parameter.
k _d	0.09	→ Death co-efficient k _d .
pl	50	→ Inhibition factor p*.
m	0.0011	→ Maintenance co-efficient.
n	0.52	→ power coefficient

Table 3.1 Input variables to the yeast model

Variable	Value	Description
X		Biomass concentration [mg/l]
S		Substrate concentration [mg/l]
P		Product concentration [mg/l]
u _{max}		Maximum growth rate
K _s		Substrate saturation / Monod's constant
Y _{x/s}		Stoichiometry coefficient for biomass growth
Y _{x/p}		Stoichiometry coefficient for product growth

Table 3.2 Output variables to the yeast model

3.7.1 LabVIEW™ 6.01i program for simulation

The control and simulation software for G is used to realize the yeast model. The Control and Simulation Software for G (GSIM) provides routines for modelling, analyzing and simulating dynamic systems. This software handles both linear and nonlinear, continuous and discrete time systems. GSIM supports the combinations of these two time models and handles real-world control problems, simulation tasks, and a combination of both. This support is useful while developing a control system that is partly simulated and partly realized (National Instruments, 1998). The developed model resembles the latter hence the software is selected to realize the control system. The control and simulation software has 12 sub pallets shown in Fig 3.2. The developed model is implemented in the discrete form and the GSim Unit Delay.vi from the main pallets is used. Fig 3.3.

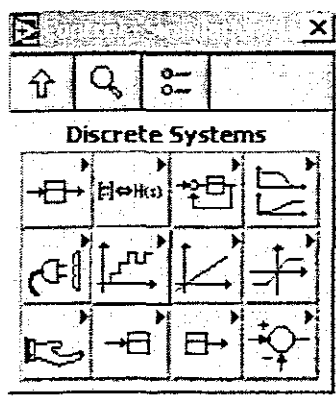


Figure 3.2 Control and Simulation sub pallets



Delays the input exactly one time step, i.e. $output[n]=input[n-1]$, where $output[0]=initial\ condition$.

Figure 3.3 GSim Unit Delay.vi

The simulation program hierarchy of the developed yeast model using equations (3.57)-(3.60) is given in Fig 3.4. The algorithm for realization of the simulation in real time system is given on Fig. 3.5.

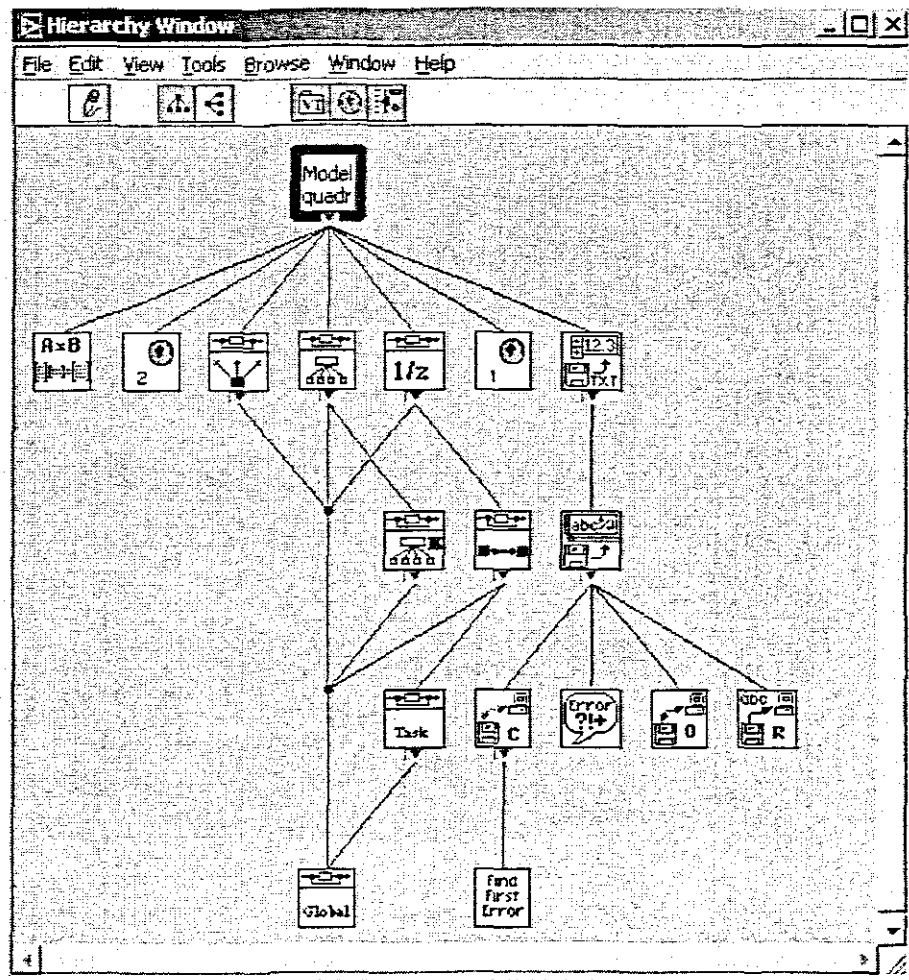


Figure 3.4 Hierarchy of the developed program for simulation of the yeast model

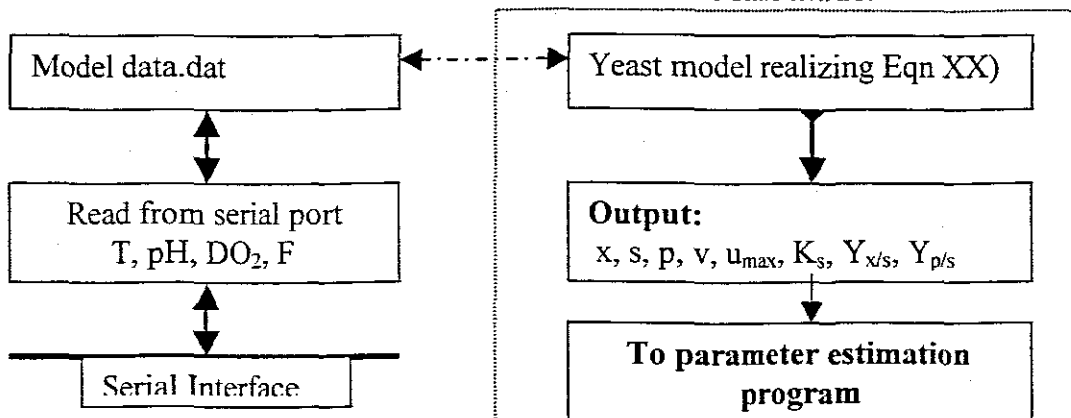


Figure 3.5 Flow of the model calculations

The data from the on-line measurements through the serial port is read using the serial function see Chapter 6. The data is stored into a file [table.dat] at a certain interval specified by the operator. This model reads that data from the file and recalculate the new values for u_{max} , K_s , $Y_{x/s}$, and $Y_{p/s}$. The new values of the parameters form the latter are passed into the biological model for the calculation of x , s , p , and v . The output of the model is y_model with dimensions $y^{[3 \times K]}$ where K is the number of samples for the recalculation. The output is then sent to the parameter estimation program for the recalculation of the model co-efficient $a_1 \rightarrow a_7$, $b_1 \rightarrow b_7$, $c_1 \rightarrow c_7$, and $d_1 \rightarrow d_7$ using least square method function. The front panel of the model is shown in Fig 3.6.

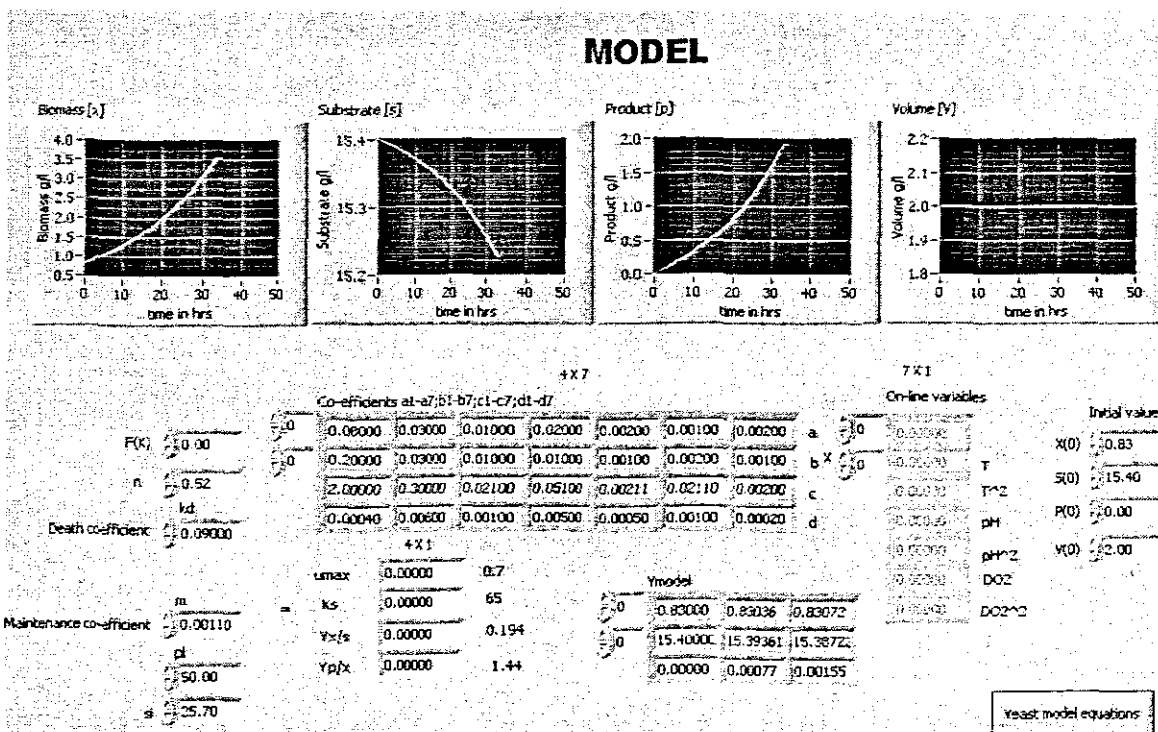


Figure 3.6 Yeast model front panel

3.7.2 MATLAB and SIMULINK program for simulation

Table 3.3 gives the flow of simulation of the yeast model in MATLAB and Simulink. MATLAB files are called 'm' files. The following procedure is followed on running the model simulation in MATLAB workspace. From the workspace in MATLAB type the following command:

→ **YM_var.m**

Where YM_var.m is the file that declares all the variables that the yeast model needs to be able to do the calculation in Simulink. The acronym YM_var stands for Yeast Model variables.m. Note before running the m file make sure that the current directory in MATLAB is pointing in the right path i.e.

"E:\Source code\Matlab Simulations\Model".

From SIMULINK open the file "yeast_model", and double click on the scope indicators to view the outputs [Table 3.2] of the model on the graph.

Model in MATLAB / SIMULINK	
Workspace	Simulink
→ YM_var.m → Input variables Table 3.1 → Output variables Table 3.2	→ yeat_model → Input variables YM_var.m → Output variables Table 3.2

Table 3.3 Simulation of the yeast model in MATLAB / SIMULINK

3.8 Conclusion

Many aspects of fermentation are poorly understood, the latter makes it difficult to derive a model that will cover all the areas of interest. Evidence that all important fermentation variables have not yet been identified is the significant batch-to-batch variation that occurs in the fermentation industry.

The key fermentation variables cannot be measured on-line. Delays are often experienced during the off-line measurements of the concentrations of substrate, biomass and the product.

The developed model is semi-theoretical in the sense that mass balance equations are theoretical mathematical model, but coefficient cannot be calculated mathematically. These coefficients are introduced mathematically to represent the influence of the physiochemical variables to the biological variables. The latter is achieved by representing the kinetic parameters of biological model through quadratic functions of physiological variables.

Parameters estimation is used to determine the coefficients of these functions.

The utilization of Matlab/Simulink and LabVIEW packages made the development much easier when it comes to testing of the prototype. The results received on both packages were the same, proving the operation of the algorithm and also the consistence was tested.

The results from simulation are used later for model parameter estimation and process optimization.

Chapter 4

Layer for adaptation

Parameter estimation

4. Introduction

The chapter describes the formulation of the problem for estimation of the fed-batch model coefficients and the method, algorithm and programme developed to solve this problem. Parameter estimation's fundamental use is to validate the calculated model parameters on the basis of experimental data. Experimental data is in principle used to compare how well does the experimental data agrees to the predictions from the model. If the model results from the latter compares favourably, the model is said to be appropriate. If the model results does not compare favourably, the model has to be reviewed or other estimation technique is implemented. The comparison is determined on the basis of error between experimental (measured) data and the model results. The approach into which the latter has been fulfilled is described and discussed below.

4.1 Experimental data

The experiments were conducted at the University of Cape Town (UCT) chemical engineering laboratory. The results used in the dissertation are from different batch and fed-batch experiments that were run and the average for each fed-batch is used Table 4.1.

Propagation of yeast during fed-batch fermentation process

Biological variables					Average values during fed-batch process			Variables			
No. of samples	Time in hrs	Biomass average	Substrate average	Product average	Biomass	Concentration in g/l	Deviation	Name	Value	Limits	Initial value
0	0	0.825333333	15.4	2.1	Biomass	14.1	0.6	T	16 deg.	-0.5	16 deg.
1	2.5	0.862666667	15.26666667	2.275	Substrate	9.45	0.78	pH	4.8 - 5.0	-0.3	4.8
2	5	0.9	15.13333333	2.45	Product	29 - 22	Declines	DO2	37%	34.5 - 39%	
3	7.5	0.916666667	15	2.625				Volume	2 Litres	0 - 4 L	2 Litres
4	10	0.933333333	14.86666667	2.8				F			
5	12.5	1.133333333	14.68333333	3.5				Ks	65 g/l	63 - 70 g/l	65 g/l
6	15	1.333333333	14.5	4.2				umax	0.7	0.027 - 0.7	0.065
7	17.5	1.55	14.26666667	4.9				Yx/s	0.194		
8	20	1.766666667	14.03333333	5.6				Yp/s	0.28		
9	22.5	2.65	13.6	7.9875				m		0.5% s util.	
10	25	3.533333333	13.16666667	10.375				kd	0		
11	27.5	4.516666667	12.7	12.7625				Delta t	2.5 hrs		
12	30	5.5	12.23333333	15.15				si			
13	32.5	6.916666667	11.81666667	18.2125				Yp/x	1.44		
14	35	8.333333333	11.4	21.275							
15	37.5	9.75	10.93333333	24.3375							
16	40	11.16666667	10.46666667	27.4							
17	42.5	12.55	9.85	30.05							
18	45	13.93333333	9.233333333	29.03							
19	47.5	14.01666667	9.341666667	28.515							
20	50	14.1	9.45	28							
21	52.5	14.2	9.435	27.5							
22	55	14.3	9.42	27							
23	57.5	14.35	9.44	26.5							
24	60	14.4	9.46	26							
25	62.5	14.2	9.455	25.5							

26	65	14	9.45	25
27	67.5	14	9.425	24.5
28	70	14	9.4	24
29	72.5	14.3	9.375	23.5
30	75	14.1	9.35	23
31	77.5	14.05	9.4	22.5
32	80	14	9.45	22

Table 4.1 Summary of data received from experiments conducted at UCT

4.2 Mathematical formulation of the problem for estimation

The problem is formulated on the basis of experimental data, model equations and criterion for minimization of the error between data and the model output as follows:

- Experimental data, which represent the trajectories of the substrate, biomass, product and reactor volume in function of temperature, pH, dissolved oxygen, and flow rate, Table 4.1.
- The model of the process, equations (3.57)-(3.60)
- Criterion is minimisation of the quadratic error between the model output and the experimental data.

The common scheme for the problem formulation is in Fig 4.1.

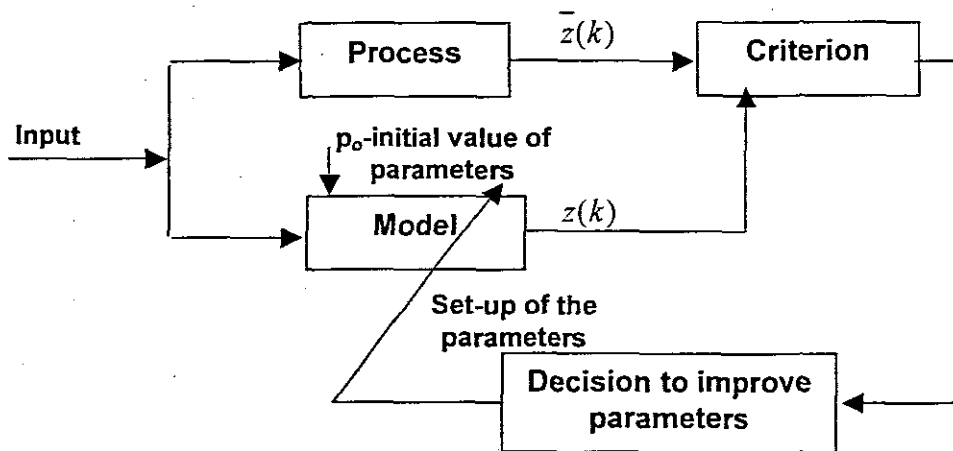


Figure 4.1: Common schemes for estimation of the model parameters

where

\bar{z} is the process output vector

z is the model output vector.

The approach represented on the Fig. 4.1:

- To send the same input signal to the process and model.
- To calculate the model output and measured process output.
- To compare them and on the basis of the error between them to take decision for changing model parameters.

Then the problem for parameter estimation can be formulated mathematically in the following way:

Find the values for the parameters $a_i, i = \overline{1,7}, b_i, i = \overline{1,7}, c_i, i = \overline{1,7}, d_i, i = \overline{1,7}$, in such a way that the error between the measured values of the outputs of the process $\overline{x}(k), \overline{s}(k), \overline{p}(k), \overline{V}(k), k = \overline{0, K}$, and the output of the model equation (3.57)-(3.60) $x(k), s(k), p(k), V(k), k = \overline{0, K}$, expressed by the criterion:

$$J = \sum_{k=0}^{K-1} \|e(k)\|^2 = \sum_{k=0}^{K-1} \|\overline{z}(k) - z(k)\|^2 \rightarrow \min \quad (4.1)$$

is minimized under the model equations (3.57)-(3.60), where the number of points of

measurements in the time interval $\overline{0, t_f}$ is K , and $z(t) = \begin{bmatrix} V(k) \\ x(k) \\ s(k) \\ p(k) \end{bmatrix}$ is the state and output

vector for the model and $\overline{z}(t) = \begin{bmatrix} \overline{V}(k) \\ \overline{x}(k) \\ \overline{s}(k) \\ \overline{p}(k) \end{bmatrix}$ is a vector of the measured data.

The criterion expresses the sum of squares of the error between the measured and the calculated from the model trajectories. It gives the best strategy for fitting the trajectories of the model to the measured trajectories. This criterion has many advantages as it gives a unique model trajectory for a given set of data.

4.3 Method of solution

The criterion is an implicit function of the parameters, as it is a function of the states, and they are function of the parameters. When looking for minimum of the criterion according to the parameters, it means according to the theory for optimality that the criterion's first derivative according to every of the parameters is necessary to be equal to zero. This derivative cannot be determined analytically in the considered case. Then the optimal solution can be found using numerical calculations by gradient

methods. The calculations are based on the gradients as the first derivatives of the criterion towards parameters but estimated and calculated in numerical way.

The vector of parameters is formed as:

$$\rho = [a_1, a_2, \dots, a_7, b_1, b_2, \dots, c_1, c_2, \dots, c_7, d_1, d_2, \dots, d_7]^T \text{ It has 28 elements.}$$

The derivatives of the criterion function towards the components of the vector of parameters are:

$$\frac{\partial J}{\partial \rho_i} = 0, i = 1, 2, \dots, r, r = 28 \quad (4.2)$$

As the criterion is not explicit function of the vector of parameters, it is necessary to use the dependencies of the output of the model from state space equations in connection with the algorithms of some of the different numeric gradient methods for sequentially computation of the improved value of the vector of parameters. The method of fastest descent is used in the study.

This method is characterised with the simplest iterative gradient procedure for improvement of the initial estimation of the vector of parameters $\rho \in R^r$. The direction of the fastest descent is opposite to the direction of the gradient and at the initial point coincides with the direction in which the criterion reduces in the fastest way for infinitely small changes of the vector of parameters. The vector of parameters gives the direction of the fastest descent according to the formula:

$$\rho^{j+1} = \rho^j + d\rho \quad (4.3)$$

Where $d\rho = [d\rho_1, d\rho_2, \dots, d\rho_r]^T$ is the direction of the descent and:

$$d\rho_i = \frac{-\alpha \frac{\partial J}{\partial \rho_i}}{\left[\sum_{i=1}^r \left(\frac{\partial J}{\partial \rho_i} \right)^2 \right]^{1/2}} \quad (4.4)$$

In equations (4.3) and (4.4) ρ^{j+1} is the improved value of the parameters, ρ^j is the previous value, $\frac{\partial J}{\partial \rho_i}$ is the gradient of J towards i -th component of the vector

$\rho, \alpha > 0$ is a step of the gradient procedure, j is the index of iterations.

It is not possibly to calculate directly analytically the components of the vector of gradients $\frac{\partial J}{\partial \rho_i}$, as the model is dynamical and the criterion is not explicit function of

the state and parameters. It is because of the latter that the numerical values of the components of the vector of gradient are determined as follows:

$$\frac{\partial J}{\partial \rho_i} = \frac{J(\rho_1, \rho_2, \dots, \rho_i + \Delta, \rho_{i+1}, \dots, \rho_r) - J(\rho_1, \rho_2, \dots, \rho_i, \rho_{i+1}, \dots, \rho_r)}{\Delta} \quad (4.5)$$

$i = 1, 2, \dots, r$

where Δ is a small deviation of the i -th component of the vector of parameters, and $\Delta \rho_i = \rho_i + \Delta$ is the deviation of the parameter used for derivative $\frac{\partial J}{\partial \rho_i}$

calculation.

The considered method is used in the scheme of estimation to calculate the improved estimated value of the parameters in such a way that the least squares criterion is minimized.

The scheme is:

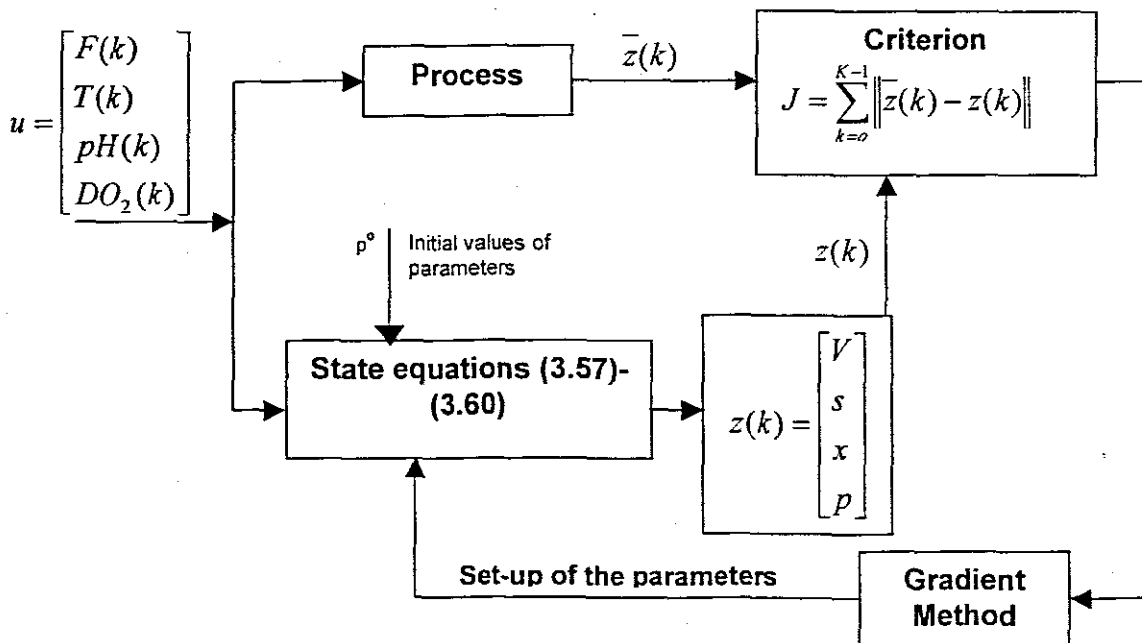


Figure 4.2: Scheme for estimation of the model parameters

4.4 Algorithm of the method

The algorithm is an iterative one. The calculations continue until the improved value of the parameters fulfils some conditions for the end of calculations (for optimality or until the given maximum number of iteration for example M is done. The calculations are done according to the steps of the following **algorithm for calculation**:

Algorithm:

Step One

At the beginning of the iterations the initial values of algorithm and problem estimation are set:

- the value of parameter deviation Δ
- maximum number of iterations M
- initial value of the state vector x_0, s_0, V_0, P_0
- step of the gradient procedure $\alpha = 0.1$
- the number of parameters $\rho \rightarrow r = 28$
- the number of steps in the optimisation period K .
- the error of the calculation of the estimated values ε
- the initial estimation of the vector of parameters

$$\rho^0 = [\dots\dots]$$

input trajectories for control variables $T(k), pH(k), DO_2(k), F(k), k = \overline{0, K-1}$

Step Two

The initial values of the parameters ρ^0 are passed into the model equations (3.57)-(3.60) and these equations are solved for the given initial conditions. The trajectories $x(k), s(k), p(k), V(k)$ are obtained for $k = \overline{0, K}$.

Step Three

The obtained values are passed into the criterion J and its value is calculated from equation (4.1).

Step Four

Set $j = 1$, calculate the deviated value of the parameters for i -th component of the vector of parameters.

$$\Delta \rho_i = \rho_i^j + \Delta, i = \overline{1, r} \quad (4.6)$$

For every $i = 1, 2, \dots, r$ and separately for every ρ_i^j

- a) Solve model equation (3.57)-(3.60), where all components of the vector ρ are the same, only ρ_i^j is changed to $\Delta\rho_i^j$.
- b) Calculate the deviated value of the criterion for i -th component $J(\rho_1, \rho_2, \Delta\rho_i^j, \rho_{i+1}, \dots, \rho_r)$ according to equation (4.1)
- c) Gradient $\frac{\partial J}{\partial \rho_i^j}$ are calculated using equation (4.5)

Step Five

Calculate the gradient for the fastest descent method $d\rho_i$.

- a) Calculate the weighted sum:

$$\Sigma = \left[\sum_{i=1}^r \left(\frac{\partial J}{\partial \rho_i^j} \right)^2 \right]^{1/2} \quad (4.7)$$

- b) Calculate the directions of the gradient

$$d\rho_i^j = -\alpha \frac{\partial J}{\partial \rho_i^j} / \Sigma, i = 1, 2, \dots, r \quad (4.8)$$

Where, $\frac{\partial J}{\partial \rho_i^j}$ has been calculated from step 4 © above.

Step Six

Calculate the improved estimate of the vector of parameter,

$$\rho_i^{j+1} = \rho_i^j + d\rho_i^j, i = 1, 2, \dots, r \quad (4.9)$$

Step Seven

Calculate the error for termination of the calculation

$$e_{\rho_i} = \left| \rho_i^{j+1} - \rho_i^j \right|, i = 1, 2, \dots, r \quad (4.10)$$

or

$$e_{\rho_i} = \left| d\rho_i^j \right|$$

Step eight

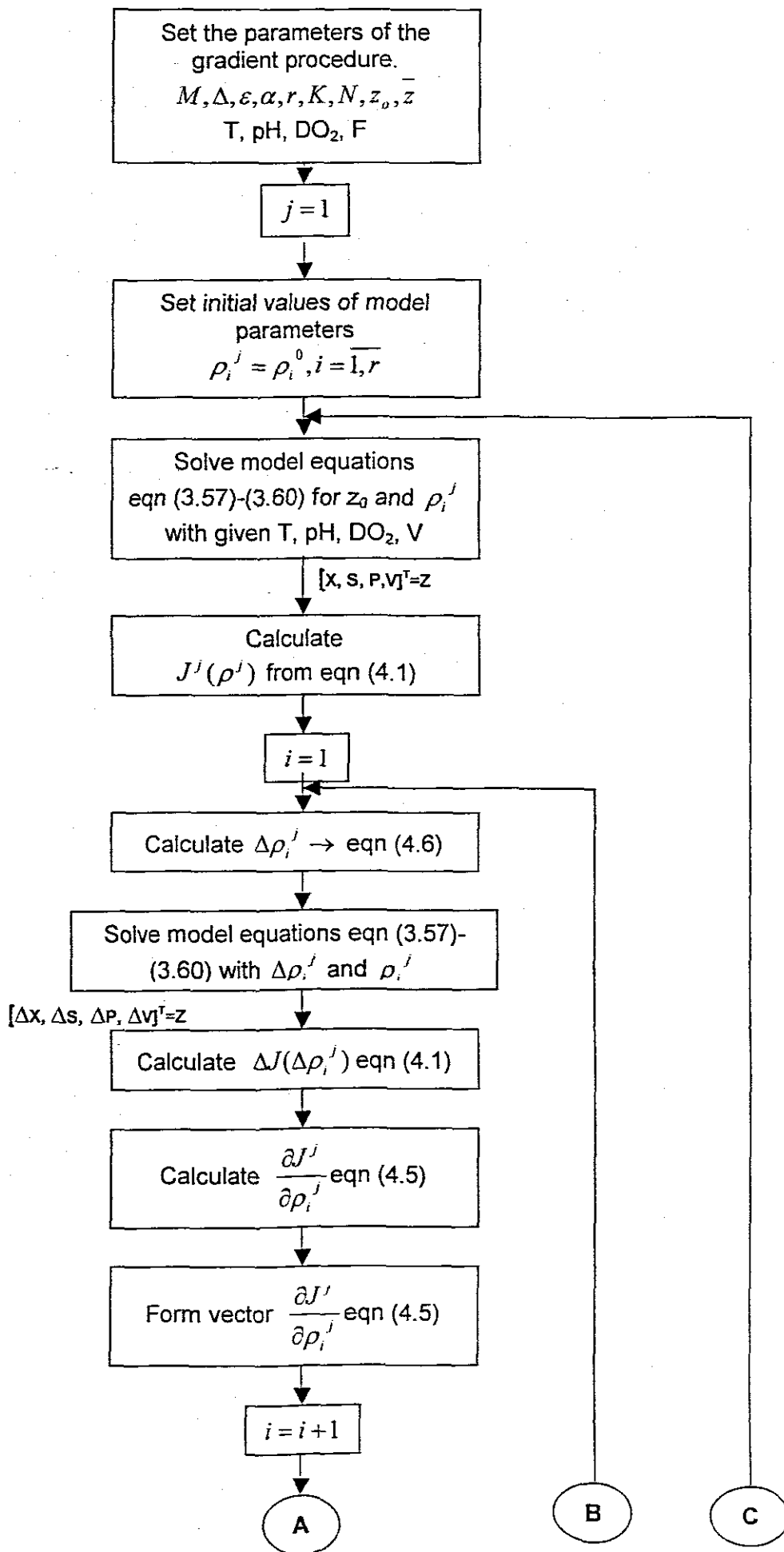
Check the criterion for termination of the calculations

$$\rightarrow \text{If } \left| d\rho_i^j \right| \leq \varepsilon, \varepsilon > 0, \varepsilon = 0.01$$

for every $i = 1, 2, \dots, r$, the optimal solutions is obtained. The calculations are terminated.

- If $|d\rho_i^j| > \varepsilon$, the calculations continue until the maximum number of iterations $j=M$ is reached. The calculations are terminated.
- If $|d\rho_i^j| > \varepsilon$, the calculations are repeated from step 2, until $|d\rho_i^j| < \varepsilon$ or $j=M$.
- If $j > M$ the calculations are terminated.

The block diagram of the described algorithm is given on Fig. 4.3.



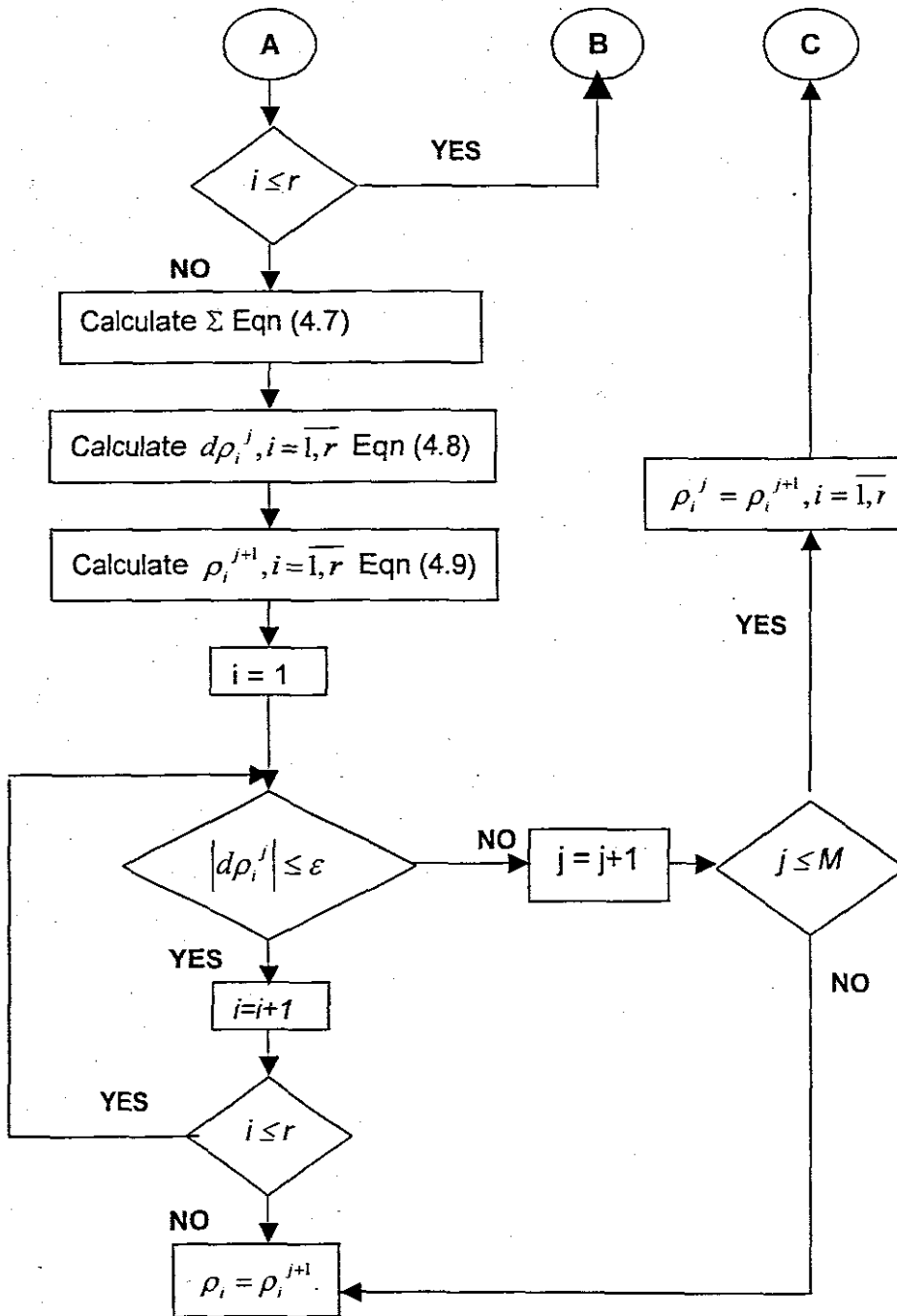


Figure 4.3 Block scheme of the algorithm for parameter estimation

4.5 Simulation and implementation of the algorithm

Simulations of the algorithm for parameter estimation are conducted in MATLAB. The implementation of the program for simulation and real time parameter estimation is done in LabVIEW™.

4.5.1 LabVIEW™ 6.01i program

The hierarchy of the developed algorithm realising the above algorithm is given in Fig 4.5.

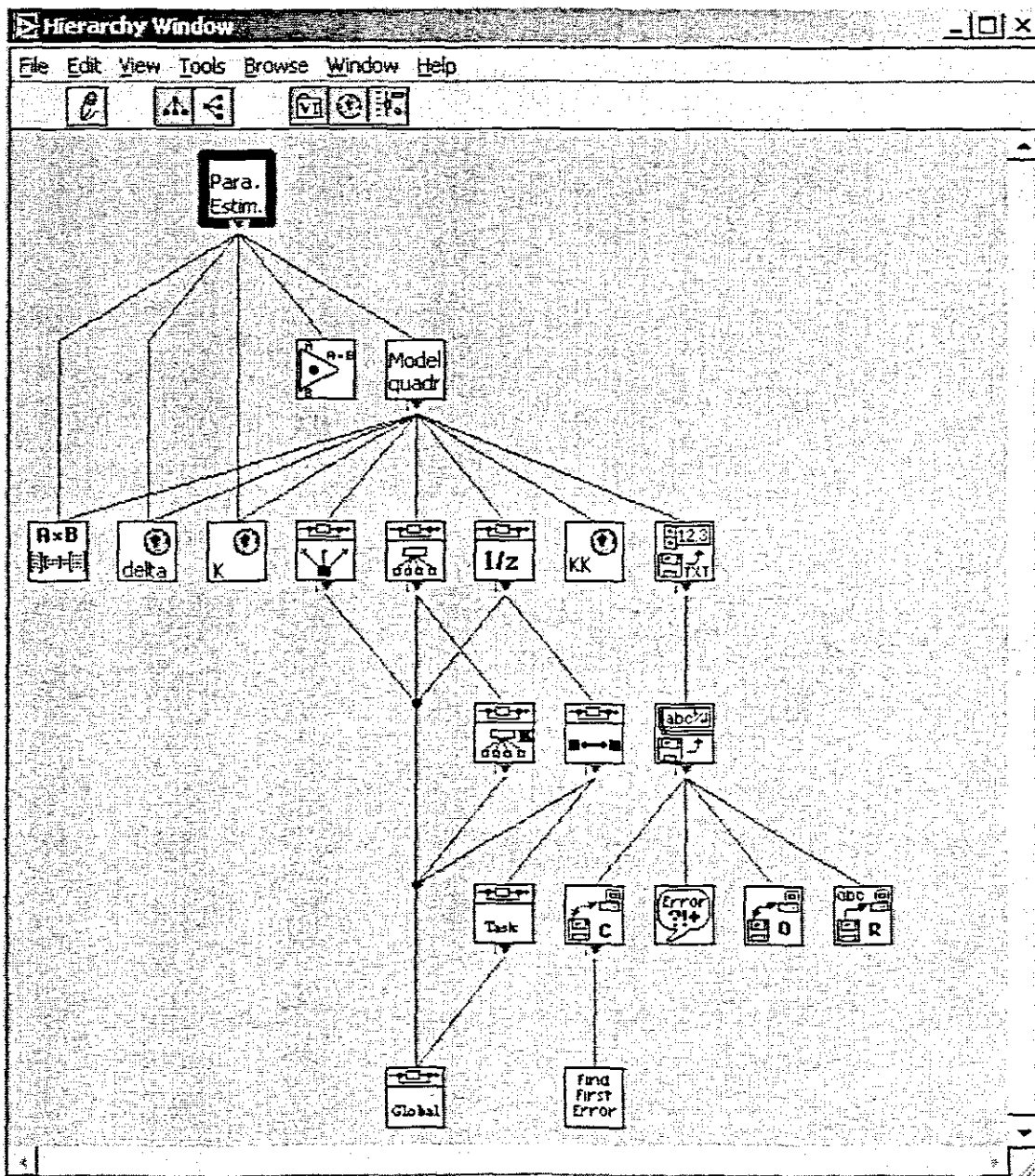


Figure 4.4 Hierarchy of the developed algorithm for parameter estimation in LabVIEW™

The parameter estimator recalculates the model parameters using the developed algorithm. The front panel of the developed parameter estimation window is given in Fig 4.5. The description of the interface is discussed in details in chapter 7.

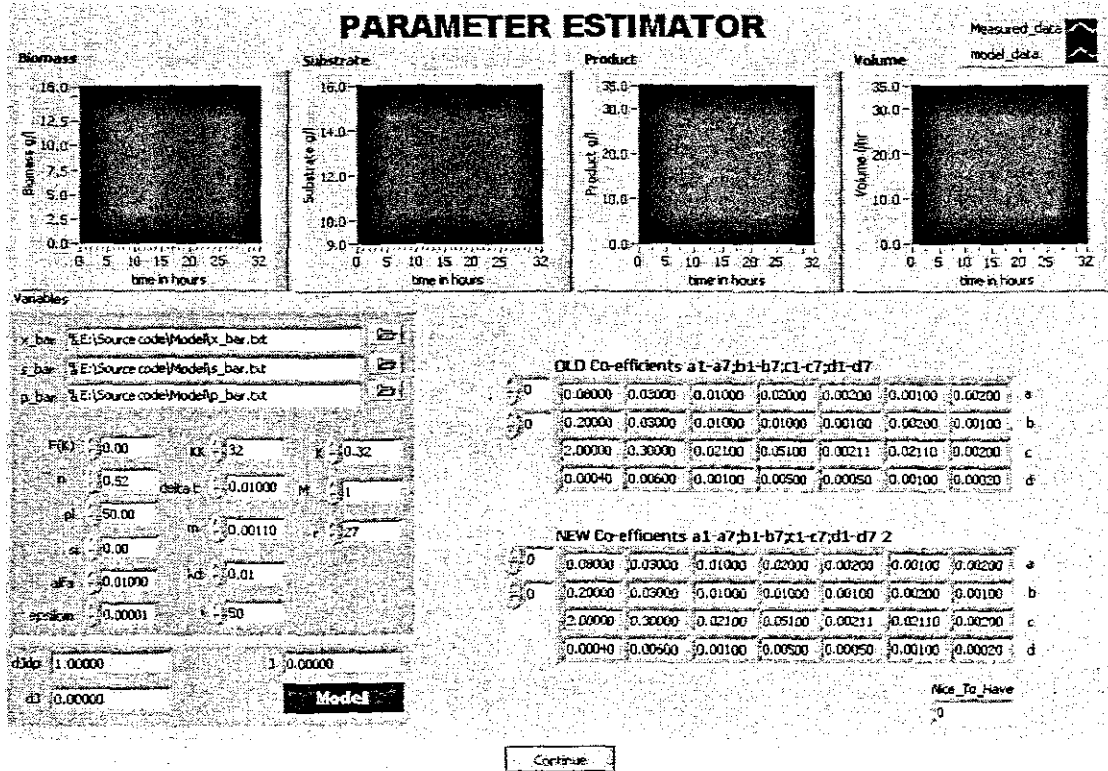


Figure 4.5 Parameter estimation front panel.

4.5.2 MATLAB 6.1

The inputs to the algorithm are arranged into a control input cluster Fig 4.6.

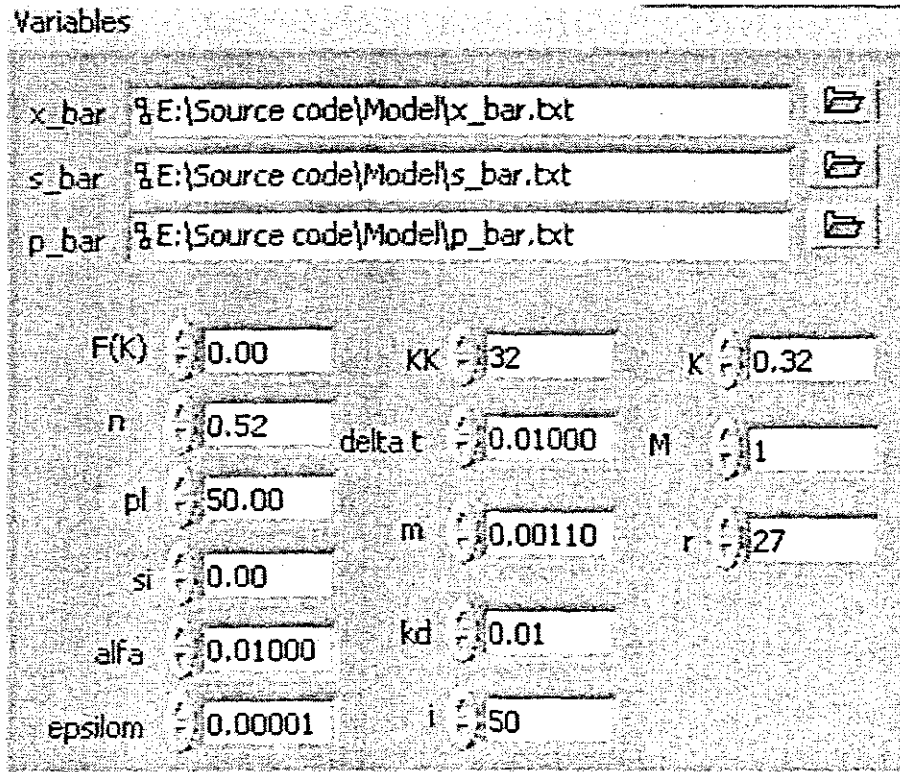


Figure 4.6 Cluster with the parameter est. variables

The following .m files are developed to realise the parameters estimation algorithm in MATLAB.

- **Model_variables.m**: This file declares all the parameter variables needed to run the algorithm.
- **YM_var_realdata.m**: This file is implementing the yeast model equations (3.45)-(3.48). The output of this file sent to the parameter estimator for recalculation of the parameters for the yeast model.
- **Parameter_Estimator.m**: This file realises the algorithm and passes the output to the model equation for the calculation of x , s , p , v , u_{\max} , K_s , $Y_{x/s}$, and $Y_{p/s}$.

The structure of the Matlab program is given on Fig. 4.7.

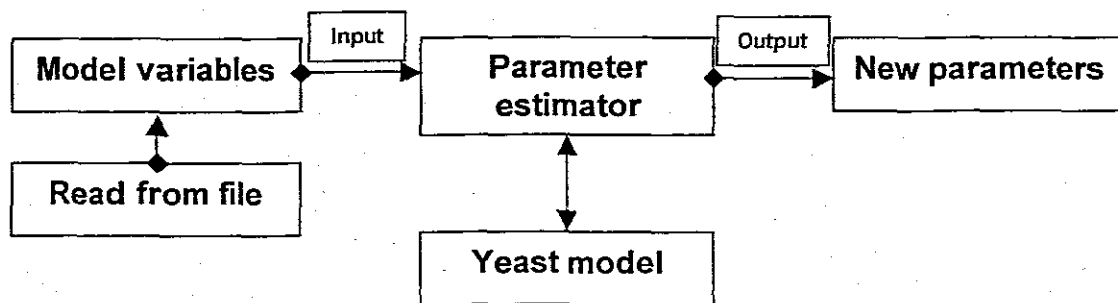


Figure 4.7 Matlab program flow for parameter estimation

The following procedure is followed on running the model simulation in MATLAB workspace. From the workspace in MATLAB type the following command:

→ **Parameter_Estimator**

Note before running the m file make sure that the current directory in MATLAB is pointing in the right path i.e.

“E:\Source code\Matlab Simulations\Parameter Estimator”.

This m file reads data from model_variables.m file Fig 4.7. It then computes the parameters for yeast_model.m file. On the iteration it updates the parameters for the model and sends the new ones for recalculation. The calculations continue until the model data is close to the measured data.

4.6 Conclusion

The problem for parameter estimation is solved by gradient method. The solution is simple but it uses a very big number of parameters (28).

The recursive least squares techniques for parameter estimation is selected, as it is suitable for operation in a noisy practical environment, also it has minimum storage requirements and does not need computational time for matrix inversion.

The use of Matlab and LabVIEW simulation and realisation software packages made it easy to implement the algorithm in real time. The comparison is made on the results received from the two packages. The comparison proves that the implementation of the algorithm is operational by receiving same results on both simulation tools.

Chapter 5

Layer for optimization

Optimal control of the fermentation process

Introduction

The optimal control problem is solved using optimal control theory, the approach of the functional of Lagrange. The optimisation layer problems are determined and based on the solutions of the previous upper layer i.e. the model parameters from the adaptation layer. The optimal operation of the process or yield of the product is based on some criteria for the production of biomass, product, and some constraints over minimal and maximal values of the variables. In the study it is based on maximizing production of yeast. Two types of problems are solved on the layer of optimisation:

- Problem for optimal control of the process.
- Problem for tuning of the PID controller parameters for direct control implementation.

The solution of the first problem determines the solution of the second one. The optimal trajectories of the control variables F , T , pH , DO_2 are used as reference trajectories for the PID controllers. Formulation of the problem, description of the developed methods, algorithms and programs is given in the chapter. The problem is solved using Matlab and LabVIEW programs and the programs are described at the end of the chapter.

5.1 Problem for optimal control of the production of yeast

The aim in producing yeast, needed, as inoculant in beer fermentation is maximum quantity of biomass, which means maximum concentration of biomass at the end of the fermentation process. The problem is how to influence the process in order to achieve this aim. The possibilities for influence are the input substrate flow rate, temperature, pH and DO_2 concentration, which in this study are considered as control variables. Their optimal trajectories can be

obtained through the formulation and solution of the problem for optimal control of the process.

5.1.1 Formulation of the problem for optimal control

Find the trajectories of the feed flow rate, temperature, pH and dissolved oxygen concentration: $F(k), T(k), pH(k), DO_2(k), k = 0, K-1$ in such a way that the concentration of biomass at the end of the process

$$J = x(K) \rightarrow \max \quad (5.1)$$

is maximized under the model equations:

$$x(k+1) = x(k) + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \Delta t k_d x(k) - \Delta t \frac{F(k)x(k)}{V(k)} = f_x(k),$$

$$x(0) = x_0 \quad (5.2)$$

$$s(k+1) = s(k) - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \Delta t m x(k) + \Delta t \frac{F(k)}{V(k)} [s_i - s(k)] = f_s(k),$$

$$s(0) = s_0 \quad (5.3)$$

$$p(k+1) = p(k) + \Delta t \mu_{\max} Y_{p/x} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \Delta t \frac{F(k)p(k)}{V(k)} = f_p(k), p(0) = 0 \quad (5.4)$$

$$V(k+1) = V(k) + \Delta t F(k) = f_v(k), \quad V(0) = V_0, \quad V(K) = V_{\max} \quad (5.5)$$

$$\mu_{\max} = a_1 + a_2 T(k) + a_3 T^2(k) + a_4 pH(k) + a_5 pH^2(k) + a_6 DO_2(k) + a_7 DO_2^2(k) \quad (5.6)$$

$$K_s = b_1 + b_2 T(k) + b_3 T^2(k) + b_4 pH(k) + b_5 pH^2(k) + b_6 DO_2(k) + b_7 DO_2^2(k) \quad (5.7)$$

$$Y_{x/s} = c_1 + c_2 T(k) + c_3 T^2(k) + c_4 pH(k) + c_5 pH^2(k) + c_6 DO_2(k) + c_7 DO_2^2(k) \quad (5.8)$$

$$Y_{p/x} = d_1 + d_2 T(k) + d_3 T^2(k) + d_4 pH(k) + d_5 pH^2(k) + d_6 DO_2(k) + d_7 DO_2^2(k) \quad (5.9)$$

and constraints over minimum and maximum values of the variables

$$T_{\min} \leq T(k) \leq T_{\max} \quad (5.10)$$

$$pH_{\min} \leq pH(k) \leq pH_{\max} \quad (5.11)$$

$$DO_{2\min} \leq DO_2(k) \leq DO_{2\max} \quad (5.12)$$

$$V_{\min} \leq V(k) \leq V_{\max} \quad (5.13)$$

$$F_{\min} \leq F(k) \leq F_{\max} \quad (5.14)$$

$$p_{\min} \leq p(k) \leq p_{\max} \quad (5.15)$$

$$s_{\min} \leq s(k) \leq s_{\max}, \quad k = \overline{0, K-1} \quad (5.16)$$

where $v_{\min}, v_{\max}, v = T, pH, DO_2, V, F, p, s, x$ and equations (5.10)-(5.16) are respectively min and max constraint of the corresponding variables and K is the number of steps in the optimization interval.

5.1.2 Method for problem solution

The problem for optimal control is characterized with non-quadratic criterion, non-linear model and constraints over the variables trajectories. Some of the variables such as $F(k)$ and $V(k)$ enter the model equations in linear way and at the same time they do not appear in the criterion function. This means that the optimal control problem according to these variables will be singular one. To overcome this difficulty the problem is solved on the basis of an augmented functional of Lagrange, which introduces quadratic terms of the model equations into the ordinary functional of Lagrange. The augmented functional can be written in the following way:

$$L = x(K) + \sum_{k=0}^{K-1} \left\{ \begin{aligned} &\lambda_x(k) f_x(k) + \lambda_s(k) f_s(k) + \lambda_p(k) f_p(k) + \lambda_v(k) f_v(k) + \frac{1}{2} \mu_x [x(k+1) \\ &- f_x(k)]^2 + \frac{1}{2} \mu_s [s(k+1) - f_s(k)]^2 + \frac{1}{2} \mu_p [p(k+1) - f_p(k)]^2 + \\ &+ \frac{1}{2} \mu_v [V(k+1) - f_v(k)]^2 \end{aligned} \right\} \quad (5.17)$$

where $\lambda_v, v = x, s, p, V$ are the conjugate variables, $\mu_v, v = x, s, p, V$ are penalty coefficients. The augmented functional has to be maximized according to state and control variables and minimized according to the conjugate variables on the basis of the theory of duality, under the process constraints represented by equations (5.10)-(5.16).

5.1.2.1 Necessary conditions for optimality

The optimal solution has to satisfy the necessary conditions for optimality according to all variables. The necessary conditions for optimality are:

→ For the state variables:

○ Biomass

$$\frac{\partial L}{\partial x(k)} = 0, \quad \frac{\partial L}{\partial s(k)} = 0, \quad \frac{\partial L}{\partial p(k)} = 0, \quad \frac{\partial L}{\partial V(k)} = 0, \quad (5.18)$$

→ For the control variables

$$\frac{\partial L}{\partial T(k)} = 0, \quad \frac{\partial L}{\partial pH(k)} = 0, \quad \frac{\partial L}{\partial DO_2(k)} = 0, \quad \frac{\partial L}{\partial F(k)} = 0, \quad (5.19)$$

→ For the conjugate variables

$$\frac{\partial L}{\partial \lambda_x(k)} = 0, \quad \frac{\partial L}{\partial \lambda_s(k)} = 0, \quad \frac{\partial L}{\partial \lambda_p(k)} = 0, \quad \frac{\partial L}{\partial \lambda_v(k)} = 0. \quad (5.20)$$

The analytical expressions for the derivatives is represented in the following way:

→ For the state variables

$$\frac{\partial L}{\partial x(k)} = \lambda_x(k) \left[1 + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{s(k)}{K_s + s(k)} \right] - \Delta t k_d - \frac{\Delta t F(k)}{V(k)} \right] +$$

$$\mu_x(k) [x(k+1) - f_x(k)] \left[1 + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{s(k)}{K_s + s(k)} \right] - \Delta t k_d - \frac{\Delta t F(k)}{V(k)} \right] +$$

$$\lambda_s(k) \left[\frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{s(k)}{K_s + s(k)} \right] - \Delta t m \right] -$$

$$\begin{aligned}
& -\mu_s(k)[s(k+1) - f_s(k)] \left[\frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{s(k)}{K_s + s(k)} \right] - \Delta t m \right] + \\
& + \lambda_p(k) \left[\Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{s(k)}{K_s + s(k)} \right] \right] - \\
& - \mu_p(k)[p(k+1) - f_p(k)] \left[\Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{s(k)}{K_s + s(k)} \right] \right] = 0 = e_x(k)
\end{aligned}$$

$$\frac{\partial L}{\partial x(K)} = 1 + \lambda_x(K) + \mu_x(K) \quad (5.21)$$

○ Substrate

$$\begin{aligned}
\frac{\partial L}{\partial s(k)} &= \lambda_x(k) \left[\Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)(K_s + s(k)) - x(k)s(k)}{[K_s + s(k)]^2} \right] \right] - \\
& - \mu_x(k)[x(k+1) - f_x(k)] \left[\Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{[K_s + s(k)]^2} \right] \right] + \\
& + \lambda_s(k) \left[1 - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{[K_s + s(k)]^2} \right] - \frac{\Delta t F(k)}{V(k)} \right] - \\
& - \mu_s(k)[s(k+1) - f_s(k)] \left[1 - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{[K_s + s(k)]^2} \right] - \frac{\Delta t F(k)}{V(k)} \right] + \\
& + \lambda_p(k) \left[\Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{[K_s + s(k)]^2} \right] \right] - \\
& - \mu_p(k)[p(k+1) - f_p(k)] \left[\Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k)}{[K_s + s(k)]^2} \right] \right] = 0 = e_s(k), k = \overline{0, K-1}
\end{aligned}$$

$$\frac{\partial L}{\partial s(K)} = \lambda_s(K) + \mu_s(K) \quad (5.22)$$

○ Product

$$\begin{aligned}
\frac{\partial L}{\partial p(k)} &= \lambda_p(k) \left[1 + \Delta t Y_{p|x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^{n-1} \left[-\frac{1}{p^*} \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \frac{\Delta t F(k)}{V(k)} \right] - \right. \\
&- \mu_p(k) [p(k+1) - f_p(k)] \left. \left[1 + \Delta t Y_{p|x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^{n-1} \left[-\frac{1}{p^*} \left[\frac{x(k)s(k)}{K_s + s(k)} \right] - \frac{\Delta t F(k)}{V(k)} \right] \right] \right. \\
&= 0 = e_p(k), k = \overline{0, K-1}
\end{aligned}$$

$$\frac{\partial L}{\partial p(K)} = \lambda_p(K) + \mu_p(K) \quad (5.23)$$

○ Volume

$$\begin{aligned}
\frac{\partial L}{\partial V(k)} &= \lambda_x(k) \left[-\Delta t \frac{-F(k)x(k)}{V^2(k)} \right] - \mu_x(k) [x(k+1) - f_x(k)] \left[\Delta t \frac{F(k)x(k)}{V^2(k)} \right] + \\
&+ \lambda_s(k) \left[\Delta t \frac{-F(k)}{V^2(k)} [s_i - s(k)] \right] - \mu_s(k) [s(k+1) - f_s(k)] \left[\Delta t \frac{-F(k)}{V^2(k)} [s_i - s(k)] \right] + \\
&+ \lambda_p(k) \left[-\Delta t \frac{-F(k)x(k)}{V^2(k)} \right] - \mu_p(k) [p(k+1) - f_p(k)] \left[\Delta t \frac{F(k)p(k)}{V^2(k)} \right] + \\
&+ \lambda_v(k) [1 - \mu_v(k) [V(k+1) - f_v(k)]. [1]] = 0 = e_v(k), k = \overline{0, K-1}
\end{aligned}$$

$$\frac{\partial L}{\partial V(K)} = \lambda_v(K) + \mu_v(K) \quad (5.24)$$

→ For the conjugate variables

$$\frac{\partial L}{\partial \lambda_x(k)} = x(k+1) - f_x(k) = 0 = e_{\lambda_x}(k) \quad (5.25)$$

$$\frac{\partial L}{\partial \lambda_s(k)} = s(k+1) - f_s(k) = 0 = e_{\lambda_s}(k) \quad (5.26)$$

$$\frac{\partial L}{\partial \lambda_p(k)} = p(k+1) - f_p(k) = 0 = e_{\lambda_p}(k) \quad (5.27)$$

$$\frac{\partial L}{\partial \lambda_v(k)} = V(k+1) - f_v(k) = 0 = e_{\lambda_v}(k) \quad (5.28)$$

→ For the control variables

○ Flow rate

$$\begin{aligned}
\frac{\partial L}{\partial F(k)} &= \lambda_x(k) \left[\frac{-x(k)\Delta t}{V(k)} \right] - \mu_x(k)[x(k+1) + f_x(k)] \left[\frac{\Delta t x(k)}{V(k)} \right] + \\
&+ \lambda_s(k) \left[\frac{\Delta t[s_i - s(k)]}{V(k)} \right] - \mu_s(k)[s(k+1) - f_s(k)] \left[\frac{\Delta t[s_i - s(k)]}{V(k)} \right] + \\
&+ \lambda_p(k) \left[\frac{-\Delta t p(k)}{V(k)} \right] - \mu_p(k)[p(k+1) - f_p(k)] \left[\frac{-\Delta t p(k)}{V(k)} \right] + \\
&+ \lambda_v(k)\Delta t - \mu_v(k)[V(k)+1] - f_v(k)\Delta t = 0 = e_F(k), \quad k = \overline{0, K-1}
\end{aligned}$$

(5.29)

○ Temperature

$$\begin{aligned}
\frac{\partial L}{\partial T(k)} = & \lambda_x(k) \left[\frac{\Delta t \partial \mu_{\max}}{\partial T} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] - \\
& - \mu_x(k) [x(k+1) - \\
& - f_x(k)] \left[\frac{\Delta t \partial \mu_{\max}}{\partial T} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] + \\
& + \lambda_s(k) \left[-\Delta t \frac{\frac{\partial \mu_{\max}}{\partial T} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial T}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} - \right. \\
& \left. - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] - \\
& - \mu_s(k) [s(k+1) - f_s(k)] \left[-\Delta t \frac{\frac{\partial \mu_{\max}}{\partial T} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial T}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} - \right. \\
& \left. - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] + \\
& + \lambda_p(k) \left[\Delta t \left[\frac{\partial Y_{p/x}}{\partial T} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial T} \right] \frac{x(k)s(k)}{K_s + s(k)} + \right. \\
& \left. + \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] - \\
& - \mu_p(k) [p(k+1) - f_p(k)] \left[\Delta t \left[\frac{\partial Y_{p/x}}{\partial T} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial T} \right] \frac{x(k)s(k)}{K_s + s(k)} + \right. \\
& \left. + \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] = 0 = e_T(k)
\end{aligned}$$

(5.30)

○ pH

$$\begin{aligned}
\frac{\partial L}{\partial p H(k)} = & \lambda_x(k) \left[\Delta t \frac{\partial \mu_{\max}}{\partial p H} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right] \\
& - \mu_x(k) [x(k+1) - f_x(k)] \left[\Delta t \frac{\partial \mu_{\max}}{\partial p H} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right] \\
& + \lambda_s(k) \left[- \Delta t \frac{\frac{\partial \mu_{\max}}{\partial p H} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial p H}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} \right. \\
& \left. - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right. \\
& \left. - \mu_s(k) [s(k+1) - f_s(k)] + \left[- \Delta t \frac{\frac{\partial \mu_{\max}}{\partial p H} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial p H}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} \right. \right. \\
& \left. \left. - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right] + \right. \\
& \left. + \lambda_p(k) \left[\Delta t \left[\frac{\partial Y_{p/x}}{\partial p H} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial p H} \right] \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \right. \right. \\
& \left. \left. + \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right] - \right. \\
& \left. - \mu_p(k) \left[\Delta t \left[\frac{\partial Y_{p/x}}{\partial p H} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial p H} \right] \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} \right. \right. \\
& \left. \left. + \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right] = 0 = e_{ph}(k)
\end{aligned}$$

(5.31)

○ Dissolved oxygen

$$\begin{aligned}
 \frac{\partial L}{\partial DO_2(k)} &= \lambda_x(k) \left[\Delta t \frac{\partial \mu_{\max}}{\partial DO_2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] \right] \\
 &- \mu_x(k)[x(k+1) - f_x(k)] \\
 &\left[\Delta t \frac{\partial \mu_{\max}}{\partial DO_2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] \right] \\
 &+ \lambda_s(k) \left[-\Delta t \frac{\frac{\partial \mu_{\max}}{\partial DO_2} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial DO_2}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} \right. \\
 &\left. - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] \right. \\
 &\left. \mu_x(k)[s(k+1) - f_s(k)] + \left[-\Delta t \frac{\frac{\partial \mu_{\max}}{\partial DO_2} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial DO_2}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} \right. \right. \\
 &\left. \left. - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] \right] + \right. \\
 &+ \lambda_p(k) \left[\Delta t \left[\frac{\partial Y_{p/x}}{\partial DO_2} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial DO_2} \right] \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \right. \\
 &\left. + \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] \right] \\
 &\mu_p(k)[p(k+1) - f_p(k)] \left[\Delta t \left[\frac{\partial Y_{p/x}}{\partial DO_2} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial DO_2} \right] \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} \right. \\
 &\left. + \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] \right] = 0 = e_{DO_2}(k), \quad k = \overline{0, K-1},
 \end{aligned}$$

(5.32)

where the partial derivatives of the kinetic parameters are:

→ According to the temperature

$$\left. \begin{aligned} \frac{\partial \mu_{\max}}{\partial T} &= a_2 + 2a_3 T(k), & \frac{\partial K_s}{\partial T} &= b_2 + 2b_3 T(k), \\ \frac{\partial Y_{x/s}}{\partial T} &= c_2 + 2c_3 T(k), & \frac{\partial Y_{p/x}}{\partial T} &= d_2 + 2d_3 T(k), \end{aligned} \right\} \quad (5.33)$$

→ According to the DO_2 .

$$\left. \begin{aligned} \frac{\partial \mu_{\max}}{\partial DO_2} &= a_6 + 2a_7 DO_2, & \frac{\partial K_s}{\partial DO_2} &= b_6 + 2b_7 DO_2(k), \\ \frac{\partial Y_{x/s}}{\partial DO_2} &= c_6 + 2c_7 DO_2(k), & \frac{\partial Y_{p/x}}{\partial DO_2} &= d_6 + 2d_7 DO_2(k), \end{aligned} \right\} \quad (5.34)$$

→ According to the pH.

$$\left. \begin{aligned} \frac{\partial \mu_{\max}}{\partial pH} &= a_4 + 2a_5 pH(k), & \frac{\partial K_s}{\partial pH} &= b_4 + 2b_5 pH(k), \\ \frac{\partial Y_{x/s}}{\partial pH} &= c_4 + 2c_5 pH(k), & \frac{\partial Y_{p/x}}{\partial pH} &= d_4 + 2d_5 pH(k), \end{aligned} \right\} \quad (5.35)$$

$$k = \overline{0, K-1}$$

5.1.2.2 Coordinating vector

The optimal trajectories of the variables can be calculated as a solution of the obtained set of non-linear, algebraic equations (5.21)-(5.35). It can be seen that the number of equations and number of variables is very big. The solution could be found easier if the problem is decomposed. The decomposition in time domain could reduce complexity of the system of equations (Tamura, 1975) and (Singandlitty, 1982). The solutions in time domain will consists of $K+1$ separate solutions, every one at separate time moment k . The decomposition in time domain can be obtained on the basis of the coordinating procedure in two level computing structure using the conjugate variables λ as coordinating ones (Tamura, 1975), but this method is applicable for normal functional of Lagrange. The used augmented functional could not be fully decomposed because of the quadratic terms and variables $x(k+1)$, $s(k+1)$, $p(k+1)$, $v(k+1)$ in them. They play a role of interconnections in time domain (Lin, 1992). This type

of functional can be fully decomposed if the considered variables are selected also as a part of the coordinating vector.

If it is denoted

$$\delta_v(k) = v(k+1), v = x, s, p, V, k = \overline{0, K-1} \quad (5.36)$$

then the coordinating vector will be

$$\lambda_v(k) = \lambda_v^j, \quad (5.37)$$

$$\delta_v(k) = \delta_v^j, v = x, s, p, v, k = \overline{0, K-1}, \quad (5.38)$$

where j is an index of iterations in the coordinating procedure. If equations (5.37)-(5.38) are substituted in the augmented functional of Lagrange, the functional and the necessary conditions for optimality are fully decomposed in time domain. The common problem is decomposed into one coordinating sub-problem and $K+1$ sub-problems on the first level of the calculating two level structure, Fig 5.1:

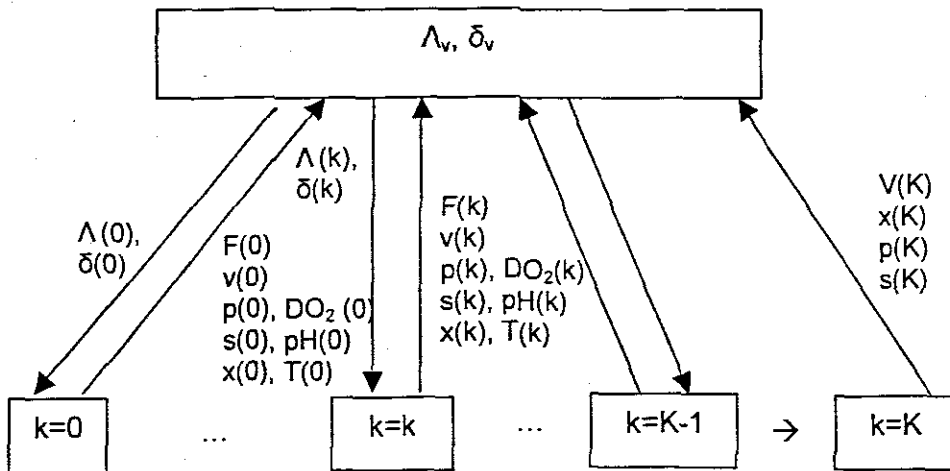


Figure 5.1: Two level calculating structures

5.1.2.3 First level sub-problems

The solutions of the first level sub-problems represent solution of the set of equations (5.21)-(5.24), (5.29) – (5.35) in which the coordinating variables are substituted. These equations are solved for every separate moment of time in a parallel way. It can be seen from the structure of the equations that analytical solution is impossible. The gradient procedure can be used in the following way

$$v^{j,i+1}(k) = v^{j,i}(k) + \alpha_v e_v^{j,i}(k), v = x, s, p, v, F, T, pH, DO_2 \quad (5.39)$$

where $\alpha_v > 0$ are the steps of the gradient procedures, $e_v(k)$ are the gradients according to the equations (5.21)-(5.35) and i is the iteration index of the procedures. The equations (5.21)-(5.35) can be simplified in the following way:

→ For the state variables

$$\begin{aligned} \frac{\partial L}{\partial x(k)} = e_x^{j,i}(k) &= \left[1 + \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{s(k)}{K_s + s(k)} - \Delta t k_d - \Delta t \frac{F(k)}{V(k)} \right] + \\ &+ \left[\frac{\Delta t \mu_{\max}}{Y_{p/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{s(k)}{K_s + s(k)} - \Delta t m \right] [\lambda_x^j(k) - \mu_x e_{\lambda_x}^j(k)] + \\ &+ \left[\Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{s(k)}{K_s + s(k)} \right] [\lambda_p^j(k) - \mu_p e_{\lambda_p}^j(k)] \end{aligned} \quad (5.40)$$

$$\begin{aligned} \frac{\partial L}{\partial s(k)} = e_s^{j,i}(k) &= \left[\Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{[K_s + s(k)]^2} \right] [\lambda_x(k) - \mu_x e_{\lambda_x}(k)] + \\ &+ \left[1 - \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{s(k)x(k)}{[K_s + s(k)]^2} - \Delta t \frac{F(k)}{V(k)} \right] [\lambda_s^j(k) - \mu_s e_{\lambda_s}^j(k)] + \\ &+ \left[\Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} \right] [\lambda_p^j(k) - \mu_p e_{\lambda_p}^j(k)] \end{aligned} \quad (5.41)$$

$$\begin{aligned} \frac{\partial L}{\partial p(k)} = e_p^{j,i}(k) &= \\ &= \left[1 + \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^{n-1} \left[\frac{-1}{p^*} \frac{x(k)s(k)}{K_s + s(k)} - \Delta t \frac{F(k)}{V(k)} \right] \right] [\lambda_p^j(k) - \mu_p e_{\lambda_p}^j(k)] \end{aligned} \quad (5.42)$$

$$\begin{aligned}
\frac{\partial L}{\partial V(k)} = e_V^{j,j}(k) &= \left[\Delta t \frac{F(k)x(k)}{V^2(k)} \right] [\lambda_x(k) - \mu_x e_{\lambda_x}(k)] + \\
&+ \left[-\frac{\Delta t F(k)}{V^2(k)} [s_i - s(k)] \right] [\lambda_s^j(k) - \mu_s e_{\lambda_s^j}(k)] + \\
&+ \left[\Delta t \frac{F(k)p(k)}{V^2(k)} \right] [\lambda_p^j(k) - \mu_p e_{\lambda_p^j}(k)] + \\
&+ [\lambda_v^j(k) - \mu_v e_{\lambda_v^j}(k)]
\end{aligned}
\tag{5.43}$$

→ For the control variables

$$\begin{aligned}
\frac{\partial L}{\partial F(k)} = e_F^{j,j}(k) &= \left[\Delta t \frac{x(k)}{V(k)} \right] [\lambda_x(k) - \mu_x e_{\lambda_x}(k)] + \\
&+ \left[-\frac{\Delta t}{V(k)} [s_i - s(k)] \right] [\lambda_s^j(k) - \mu_s e_{\lambda_s^j}(k)] + \\
&+ \left[-\Delta t \frac{p(k)}{V(k)} \right] [\lambda_p^j(k) - \mu_p e_{\lambda_p^j}(k)] + \\
&+ \Delta t [\lambda_v^j(k) - \mu_v e_{\lambda_v^j}(k)] = e_F(k)
\end{aligned}
\tag{5.44}$$

$$\begin{aligned}
\frac{\partial L}{\partial p H(k)} &= e_{pH}^{jj}(k) = \\
&= \left[\Delta \frac{\partial \mu_{\max}}{\partial p H} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} - \Delta \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right]^* \\
&+ [\lambda_x(k) - \mu_x(k) e_{\lambda_x}(k)] + \\
&+ \left[-\Delta \frac{\frac{\partial \mu_{\max}}{\partial p H} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial p H}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \right. \\
&+ \left. \frac{\Delta \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right] [\lambda_s(k) - \mu_s(k) e_{\lambda_s}(k)] + \\
&+ \left[\Delta \left[\frac{\partial Y_{p/x}}{\partial p H} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial p H} \right] \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} - \right. \\
&- \left. \Delta Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k) \frac{\partial K_s}{\partial p H}}{[K_s + s(k)]^2} \right] \right] [\lambda_p(k) - \mu_p(k) e_{\lambda_p}(k)]
\end{aligned}$$

(5.45)

$$\begin{aligned}
\frac{\partial L}{\partial T(k)} &= e_T(k) = \\
&= \left[\frac{\Delta t \partial \mu_{\max}}{\partial T} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} - \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{-x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right]^* \\
&* [\lambda_x(k) - \mu_x(k) e_{\lambda_x}(k)] + \\
&+ \left[-\Delta t \frac{\frac{\partial \mu_{\max}}{\partial T} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial T}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \right. \\
&+ \left. \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] [\lambda_s(k) - \mu_s(k) e_{\lambda_s}(k)] + \\
&+ \left[\Delta t \frac{\frac{\partial \mu_{\max}}{\partial T} Y_{x/s} + \mu_{\max} \frac{\partial Y_{x/s}}{\partial T}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} - \right. \\
&- \left. \Delta t \mu_{\max} Y_{p/x} \left[1 - \frac{p(k)}{p^*} \right]^n \left[\frac{x(k)s(k) \frac{\partial K_s}{\partial T}}{[K_s + s(k)]^2} \right] \right] [\lambda_p(k) - \mu_p(k) e_{\lambda_p}(k)]
\end{aligned}$$

(5.46)

$$\begin{aligned}
\frac{\partial L}{\partial DO_2(k)} &= e_{DO_2}^{ij}(k) = \\
&= \left[\Delta t \frac{\partial \mu_{\max}}{\partial DO_2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} - \Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] * \\
&* [\lambda_x(k) - \mu_x(k) e_{\lambda_x}(k)] + \\
&+ \left[-\Delta t \frac{\frac{\partial \mu_{\max}}{\partial DO_2} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial DO_2}}{Y_{x/s}^2} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \right. \\
&+ \frac{\Delta t \mu_{\max}}{Y_{x/s}} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \left. [\lambda_s(k) - \mu_s(k) e_{\lambda_s}(k)] + \right. \\
&+ \left[\Delta t \left[\frac{\partial Y_{p/x}}{\partial DO_2} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial DO_2} \right] \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} + \right. \\
&+ \left. \Delta t Y_{p/x} \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k) \frac{\partial K_s}{\partial DO_2}}{[K_s + s(k)]^2} \right] [\lambda_p(k) - \mu_p(k) e_{\lambda_p}(k)]
\end{aligned} \tag{5.47}$$

These equations can be simplified further for the purpose of calculations. Some expressions in them are repeated, that is why the common notations are used to represent them:

$$\Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{s(k)}{K_s + s(k)} = \varphi_x(k) \tag{5.48}$$

$$\Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{[K_s + s(k)]^2} = \varphi_s(k) \tag{5.49}$$

$$\Delta t \frac{F(k)}{V^2(k)} = \varphi_v(k) \tag{5.50}$$

$$\frac{\Delta t \mu_{\max} Y_{p/x}}{p^*} \left[1 - \frac{p(k)}{p^*} \right]^{n-1} \frac{x(k)s(k)}{K_s + s(k)} = \varphi_p(k) \tag{5.51}$$

$$\Delta t \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{K_s + s(k)} = \varphi_1(k) \tag{5.52}$$

$$\Delta t \mu_{\max} \left[1 - \frac{p(k)}{p^*} \right]^n \frac{x(k)s(k)}{[K_s + s(k)]^2} = \varphi_2(k), k = \overline{0, K-1} \quad (5.53)$$

Using the notations (5.48)-(5.53), the equations (5.40)-(5.47) can be written in the following way:

→ For state variables

$$\begin{aligned} \frac{\partial L}{\partial x(k)} = e_x^{jj}(k) &= \left[1 + \varphi_x^{jj}(k) - \Delta t k_d + \Delta t \frac{F(k)}{V(k)} \right] \left[\lambda_x^j(k) - \mu_x(k) e_{\lambda_x^j}(k) \right] + \\ &+ \left[\frac{e_x^{jj}(k)}{Y_{x/s}} - \Delta t m \right] \left[\lambda_s^j(k) - \mu_s(k) e_{\lambda_s^j}(k) \right] + \\ &+ \Delta t \varphi_x^{jj}(k) Y_{p/x} \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p^j}(k) \right] \end{aligned} \quad (5.54)$$

$$\begin{aligned} \frac{\partial L}{\partial s(k)} = e_s^{jj}(k) &= \varphi_s^{jj}(k) \left[\lambda_x^j(k) - \mu_x(k) e_{\lambda_x^j}(k) \right] + \\ &+ \left[1 - \frac{\varphi_s^{jj}(k)}{Y_{x/s}} - \Delta t \frac{F^{jj}(k)}{V^{jj}(k)} \right] \left[\lambda_s^j(k) - \mu_s(k) e_{\lambda_s^j}(k) \right] + \\ &+ Y_{p/x} \varphi_s(k) \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p^j}(k) \right] \end{aligned} \quad (5.55)$$

$$\frac{\partial L}{\partial p(k)} = e_p^{jj}(k) = \left[1 + \varphi_p(k) - \Delta t \frac{F(k)}{V(k)} \right] \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p^j}(k) \right] \quad (5.56)$$

$$\begin{aligned} \frac{\partial L}{\partial V(k)} = e_v^{jj}(k) &= \varphi_v(k) x(k) \left[\lambda_x^j(k) - \mu_x(k) e_{\lambda_x^j}(k) \right] - \\ &- \varphi_v(k) [s_i - s(k)] \left[\lambda_s^j(k) - \mu_s(k) e_{\lambda_s^j}(k) \right] + \\ &+ \varphi_v(k) p(k) \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p^j}(k) \right] + \left[\lambda_v^j(k) - \mu_v(k) e_{\lambda_v^j}(k) \right] \end{aligned} \quad (5.57)$$

→ For the control variables

$$\begin{aligned} \frac{\partial L}{\partial F(k)} = e_F^{jj}(k) &= \\ &= \frac{\Delta t x(k)}{V(k)} \left[\lambda_x^j(k) - \mu_x(k) e_{\lambda_x^j}(k) \right] + \frac{\Delta t [s_i - s(k)]}{V(k)} \left[\lambda_s^j(k) - \mu_s(k) e_{\lambda_s^j}(k) \right] - \\ &- \frac{\Delta t p(k)}{V(k)} \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p^j}(k) \right] + \Delta t \left[\lambda_v^j(k) - \mu_v(k) e_{\lambda_v^j}(k) \right] \end{aligned} \quad (5.58)$$

$$\begin{aligned}
\frac{\partial L}{\partial T(k)} &= e_T^{j,j}(k) = \\
&= \left[\varphi_1(k) \frac{\partial \mu_{\max}}{\partial T} - \varphi_2(k) \frac{\partial K_s}{\partial T} \right] \left[\lambda_x^j(k) - \mu_x(k) e_{\lambda_x}^j(k) \right] + \\
&+ \left[-\varphi_1(k) \right] \left[\frac{\frac{\partial \mu_{\max}}{\partial T} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial T}}{Y_{x/s}^2} \right] + \\
&+ \frac{\partial_2(k)}{Y_{x/s}} \frac{\partial K_s}{\partial T} \left[\lambda_s^j(k) - \mu_s(k) e_{\lambda_s}^j(k) \right] + \left[\varphi_1(k) \left[\frac{\partial Y_{p/x}}{\partial T} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial T} \right] - \right. \\
&\left. - Y_{p/x} \varphi_2(k) \frac{\partial K_s}{\partial T} \right] \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p}^j(k) \right]
\end{aligned}
\tag{5.59}$$

$$\begin{aligned}
\frac{\partial L}{\partial pH} &= e_{pH}^{j,j}(k) = \\
&= \left[\varphi_1(k) \frac{\partial \mu_{\max}}{\partial pH} - \varphi_2(k) \frac{\partial K_s}{\partial pH} \right] \left[\lambda_x^j(k) - \mu_x(k) e_{\lambda_x}^j(k) \right] + \\
&+ \left[-\varphi_1(k) \right] \left[\frac{\frac{\partial \mu_{\max}}{\partial pH} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial pH}}{Y_{x/s}^2} \right] + \\
&+ \frac{\partial_2(k)}{Y_{x/s}} \frac{\partial K_s}{\partial pH} \left[\lambda_s^j(k) - \mu_s(k) e_{\lambda_s}^j(k) \right] + \left[\varphi_1(k) \left[\frac{\partial Y_{p/x}}{\partial pH} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial pH} \right] - \right. \\
&\left. - Y_{p/x} \varphi_2(k) \frac{\partial K_s}{\partial pH} \right] \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p}^j(k) \right]
\end{aligned}
\tag{5.60}$$

$$\begin{aligned}
\frac{\partial L}{\partial DO_2} &= e_{DO_2}^{j,j}(k) = \\
&= \left[\varphi_1(k) \frac{\partial \mu_{\max}}{\partial DO_2} - \varphi_2(k) \frac{\partial K_s}{\partial DO_2} \right] \left[\lambda_x^j(k) - \mu_x(k) e_{\lambda_x^j}(k) \right] \\
&+ \left[-\varphi_1(k) \left[\frac{\frac{\partial \mu_{\max}}{\partial DO_2} Y_{x/s} - \mu_{\max} \frac{\partial Y_{x/s}}{\partial DO_2}}{Y_{x/s}^2} \right] \right] + \\
&+ \frac{\partial_2(k)}{Y_{x/s}} \frac{\partial K_s}{\partial T} \left[\lambda_s^j(k) - \mu_s(k) e_{\lambda_s^j}(k) \right] + \left[\varphi_1(k) \left[\frac{\partial Y_{p/x}}{\partial DO_2} \mu_{\max} + Y_{p/x} \frac{\partial \mu_{\max}}{\partial DO_2} \right] - \right. \\
&\left. - Y_{p/x} \varphi_2(k) \frac{\partial K_s}{\partial DO_2} \right] \left[\lambda_p^j(k) - \mu_p(k) e_{\lambda_p^j}(k) \right]
\end{aligned} \tag{5.61}$$

The calculation of the gradients is done with the old values of the state and control variables $v^{j,j}$, $v = x, s, p, V, F, T, pH, DO_2$. The calculation with the gradient procedure will continue until the necessary conditions for optimality towards state and control variables are fulfilled. This means that the values of the gradients are approximately equal to zero:

$$\|e_v(k)\| \leq \varepsilon_v, \varepsilon_v > 0, v = x, s, p, V, F, T, pH, DO_2, k = \overline{0, K-1} \tag{5.62}$$

The new calculated values have to be checked for belonging to the area of constraints (5.10)-(5.16). Simple way to do and correct the obtained new values is to project them over the constraints domain according to the relation:

$$v^{j,j}(k) = \left\{ \begin{array}{l} v_{\min}, \text{ if } v^{j,j}(k) < v_{\min} \\ v^{j,j}(k), \text{ if } v_{\min} \leq v^{j,j}(k) \leq v_{\max}, k = \overline{0, K-1} \\ v_{\max} \text{ if } v^{j,j}(k) > v_{\max} \end{array} \right\} \tag{5.63}$$

where $v = x, s, p, V, F, T, pH, DO_2$

The obtained values of the trajectories of the state and control variables are function of the coordinating variables. They are optimal for the given values of the coordinating variables. If the coordinating trajectories are optimal ones, then the obtained state and control variables also will be the optimal according to the theory of the duality in optimal control and Lagrange methods.

5.1.2.4 The coordinating sub-problem

The coordinating variables are obtained from the necessary conditions for optimality of the augmented Lagrange functional according to them. For the conjugate variables these are equations are (5.25)-(5.28). The necessary conditions for optimality according to the new introduced interconnections in time domain are:

$$\frac{\partial L}{\partial \delta_x(k)} = \mu_x [\delta_x(k) - f_x(k)] = e_{\delta_x}(k) = 0 = \mu_x e_{\lambda_x}(k), \quad (5.64)$$

$$\frac{\partial L}{\partial \delta_s(k)} = \mu_s [\delta_s(k) - f_s(k)] = e_{\delta_s}(k) = 0 = \mu_s e_{\lambda_s}(k), \quad (5.65)$$

$$\frac{\partial L}{\partial \delta_p(k)} = \mu_p [\delta_p(k) - f_p(k)] = e_{\delta_p}(k) = 0 = \mu_p e_{\lambda_p}(k), \quad (5.66)$$

$$\frac{\partial L}{\partial \delta_v(k)} = \mu_v [\delta_v(k) - f_v(k)] = e_{\delta_v}(k) = 0 = \mu_v e_{\lambda_v}(k), \quad (5.67)$$

$$\text{where } \delta_v(k) - f_v(k) = e_{\lambda_v}(k), v = x, s, p, V, \quad (5.68)$$

according to equations (5.25)-(5.28).

The necessary conditions for optimality (5.25)-(5.28) and (5.64)-(5.68) can be combined in a gradient procedure. Equations (5.64)-(5.68) give the values of the gradients for the conjugate variables multiplied by the penalty co-efficient.

$$\mu_v, v = x, s, p, V.$$

It appears that:

$$\frac{\partial L}{\partial \delta_v(k)} = \mu_v \frac{\partial L}{\partial \lambda_v(k)} = \mu_v e_{\lambda_v}(k), \quad k = \overline{0, K-1} \quad (5.69)$$

Gradient procedure can be built for both type of coordinating variables.

→ **For conjugate variables:**

$$\lambda_v^{j+1}(k) = \lambda_v^j(k) - \alpha_{\lambda_v} e^j_{\lambda_v}(k), \quad (5.70)$$

$$e^j_{\lambda_v}(k) = \delta_v^j(k) - f^j_v(k), v = x, s, p, V, k = \overline{0, K-1}, \quad (5.71)$$

$$\delta_v^{j+1}(k) = \delta_v^j(k) + \alpha_{\delta_v} e_{\delta_v}^j(k) = \delta_v^j(k) + \alpha_{\delta_v} \mu_v e_v^j(k), k = \overline{0, K-1} \quad (5.72)$$

$$v = x, s, p, V$$

where α_{λ_v} and α_{δ_v} are the steps of the procedures for calculation of the conjugate variables and the interconnection in time, respectively.

As the calculations are based on gradient iterative procedure, they will stop when the gradient is going to zero, or practically achieving some conditions for optimality, given by

$$\|e_{\lambda\nu}(k)\| \leq \varepsilon_{\lambda\nu}, \varepsilon_{\lambda\nu} > 0, k = \overline{0, K-1}, \quad (5.73)$$

where $\varepsilon_{\lambda\nu}$ is a very small positive number.

For every step of the gradient procedure for the coordinating process the full iteration cycle of the sub problems of the first level is done. The calculations of the coordinating variables are done with the values of state and control variables of the first level sub problems and the calculations of the first level sub problems are done with the values of the calculated on the previous iteration coordinating variables. The iterative process of coordination and the optimal solution of the whole problem for optimal control is obtained when the optimal solution of the coordinating sub problem is reached, this means conditions (5.73) are fulfilled.

5.1.3 Algorithm for calculation

The calculations in the two-level structure can be grouped and represented in the following algorithm:

1. On the coordinating level the initial values of the coordinating and gradient procedures, initial trajectories of the state, control and coordinating variables are set as follows:
 - a) Maximal number of step of the coordinating process – M1.
Maximal number of steps for state and control variables gradient procedure – M2.
 - b) Steps of the gradient procedures
 - i. For coordinating conjugate variables $-\alpha_{\lambda\nu}, \nu = x, s, p, V$.
 - ii. For coordinating interconnection in time $-\alpha_{\delta\nu}, \nu = x, s, p, V$.
 - iii. For state variables and control variables
 $\alpha_{\nu}, \nu = x, s, p, V, F, T, pH, DO_2$

- c) Values for error and termination of the calculations for coordinating procedures. $\varepsilon_{\lambda\nu}, \varepsilon_{\delta\nu}, \nu = x, s, p, V$ and for gradient procedures on first level.

$$\varepsilon_{\nu}, \nu = x, s, p, V$$

- d) Initial values trajectories of the conjugate variables.

$$\lambda^l_{\nu}(k), \nu = x, s, p, V, k = \overline{0, K-1}$$

- e) Initial values of the state variables, $x(0), s(0), p(0), V(0)$.

- f) Number of steps in the optimization horizon – K .

- g) Initial trajectories of the control variables

$$F^{1,l}(k), T^{1,l}(k), pH^{1,l}(k), DO_2^{1,l}(k), k = \overline{0, K-1},$$

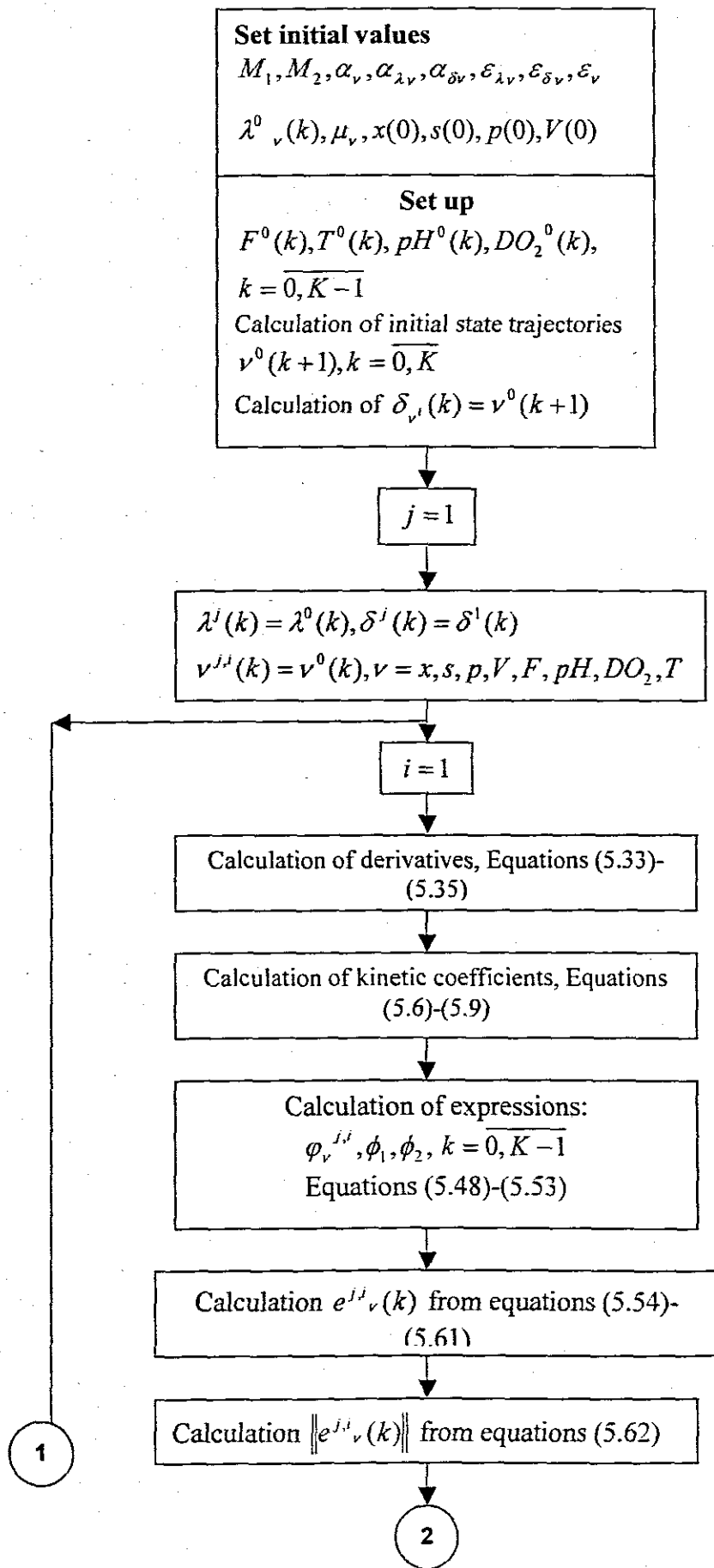
- h) The initial trajectories of the control variables are substituted in the model equations (5.2)-(5.9) and the state trajectories are calculated.

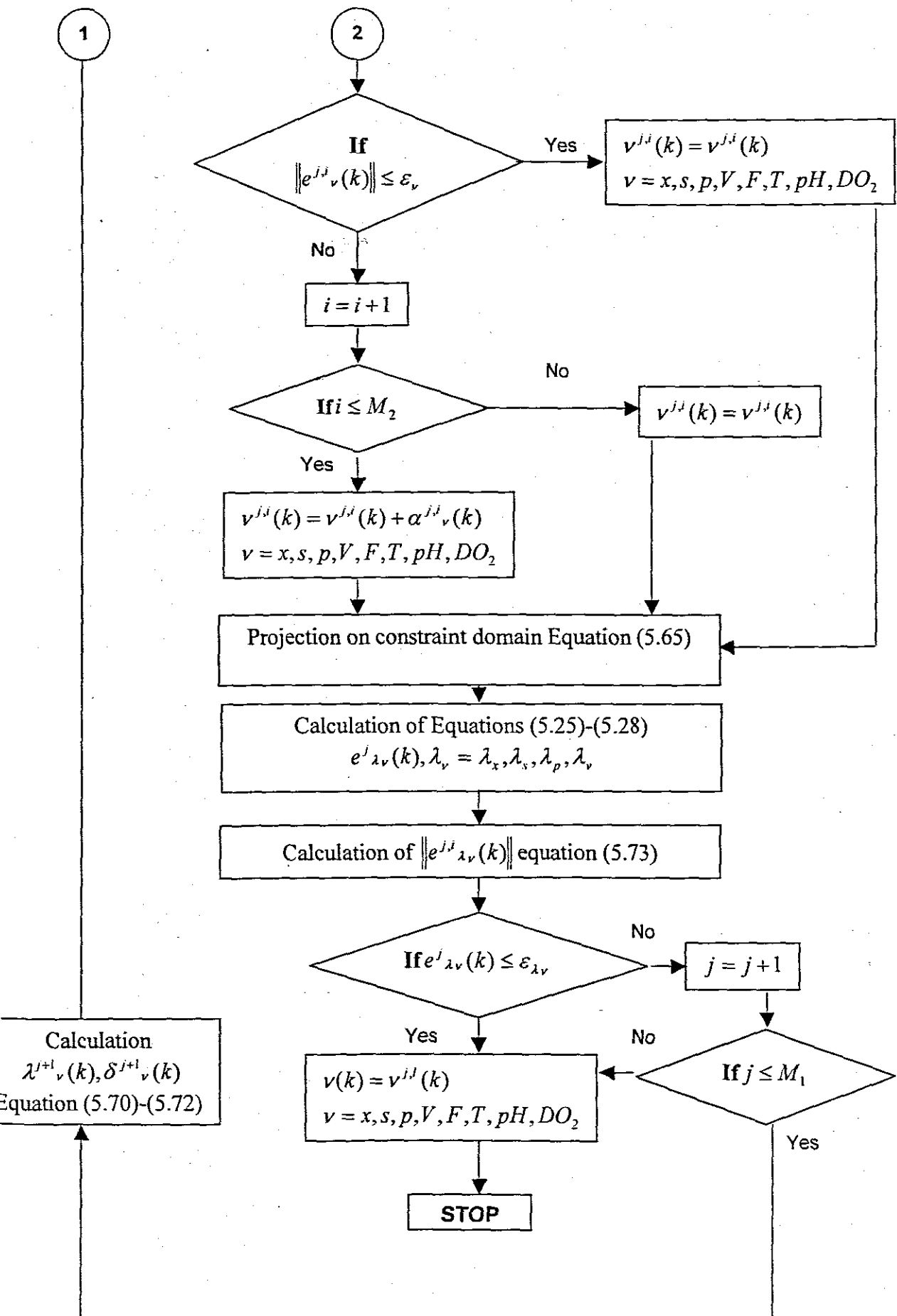
- i) The initial trajectories of the interconnections in time domain are calculated from $\delta_{\nu}(k) = \nu(k+1), k = \overline{0, K-1}$

- j) Initial trajectories of the coordinating variables and initial state and control trajectories are sent to the first level.

- 2) On the first level the following calculations are done:
 - a. Derivatives from equation (5.33)-(5.35) are calculated.
 - b. The kinetic coefficients $\mu_{\max}, K_s, Y_{x/s}, Y_{p/x}$ from equations (5.6)-(5.9) are calculated.
 - c. The expressions $\varphi_x(k), \varphi_s(k), \varphi_p(k), \varphi_V(k), \varphi_1(k), \varphi_2(k), k = \overline{0, K-1}$ are calculated from equations (5.48)-(5.53).
 - d. The gradients $e_{\nu}^{(j)}(k), k = \overline{0, K-1}, \nu = x, s, p, V, F, T, pH, DO_2$ are calculated from equations (5.54)-(5.61).
 - e. The errors for the end of the gradient procedures are calculated according to equation (5.62). If the error is less than ε_e , the calculations of the first level stop and the current values of the state and control variables are the optimal ones. If the error conditions are not satisfied the new values of the state and control values are calculated.
 - f. The new values of state and control variables are calculated from equation (5.39).
 - g. The obtained state and control variables are projected over the constraint domain according to equation (5.63). The obtained trajectories are sent to the second level.
- 3) On the second level the following calculations are done:
 - a. The gradients $e_{\lambda\nu}(k), k = \overline{0, K-1}, \nu = x, s, p, V$ are calculated from equations (5.25)-(5.28).
 - b. The conditions for the end of the calculations are checked according to Equation (5.73). If the conditions are fulfilled the optimal solution of the coordinating and of the whole problem are obtained. If the conditions are not satisfied the new values of the coordinating variables are calculated.
 - c. New values of the conjugate variables are calculated according to equation (5.70)-(5.71).
 - d. The new values of the interconnections are calculated from equation (5.72). These trajectories are sent to the first level sub problems where the calculations from p.2a till 3b are repeated and so on.

The schematic diagram of the algorithm is given on Fig 5.2.





5.2 Proportional Integral and Derivative (PID) controller

5.2.1 Background

A Proportional, Integral and Derivative (PID) is the feedback controller that is designed to generate an “output” that causes some corrective effort to be applied to a “process variable” towards a desired value known as the “setpoint”. Fig. 5.4 shows a typical feedback control loop, with blocks representing the dynamics elements of the system and arrows representing the signal flow. The objective of most controllers is to determine the error between the setpoint and the actual process variable measured. A typical example of a system where the controller would be necessary is the home thermostat. The controller will make sure that a desired temperature is maintained through the monitoring of process variable measured and compare it with the desired temperature (setpoint) to get the error, and the controller sends the output to the air conditioner if the error is too high (Vance, 1996).

5.2.1.1 PID

The PID basically performs the same function as the thermostat, with a more elaborate algorithm for determining its output. The algorithm is given by (National Instruments, 1998):

$$u(t) = K_c \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \quad (5.74)$$

where K_c is controller gain, T_i is the integral time in minutes (also called reset time), and T_d is the derivative time in minutes (also called rate). The proportional action is, and the derivative action is:

$$u_p(t) = K_c e(t) \quad (5.75)$$

the integral action is,

$$u_i(t) = \frac{K_c}{T_i} \int_0^t e(t) dt, \quad (5.76)$$

and the derivative action is

$$u_D(t) = K_c T_d \frac{de(t)}{dt} \quad (5.77)$$

The PID implements the above algorithms. Basically that is all with the PID, but the trick part is the tuning of the PID controller, setting P, I, and D tuning constants so that the weighted sum of the proportional, integral, and derivative terms produces a controller output that steadily drives the process variable in the direction required to eliminate the error. The tuning algorithm used in the dissertation is part of LabVIEW toolbox Fig 5.3.

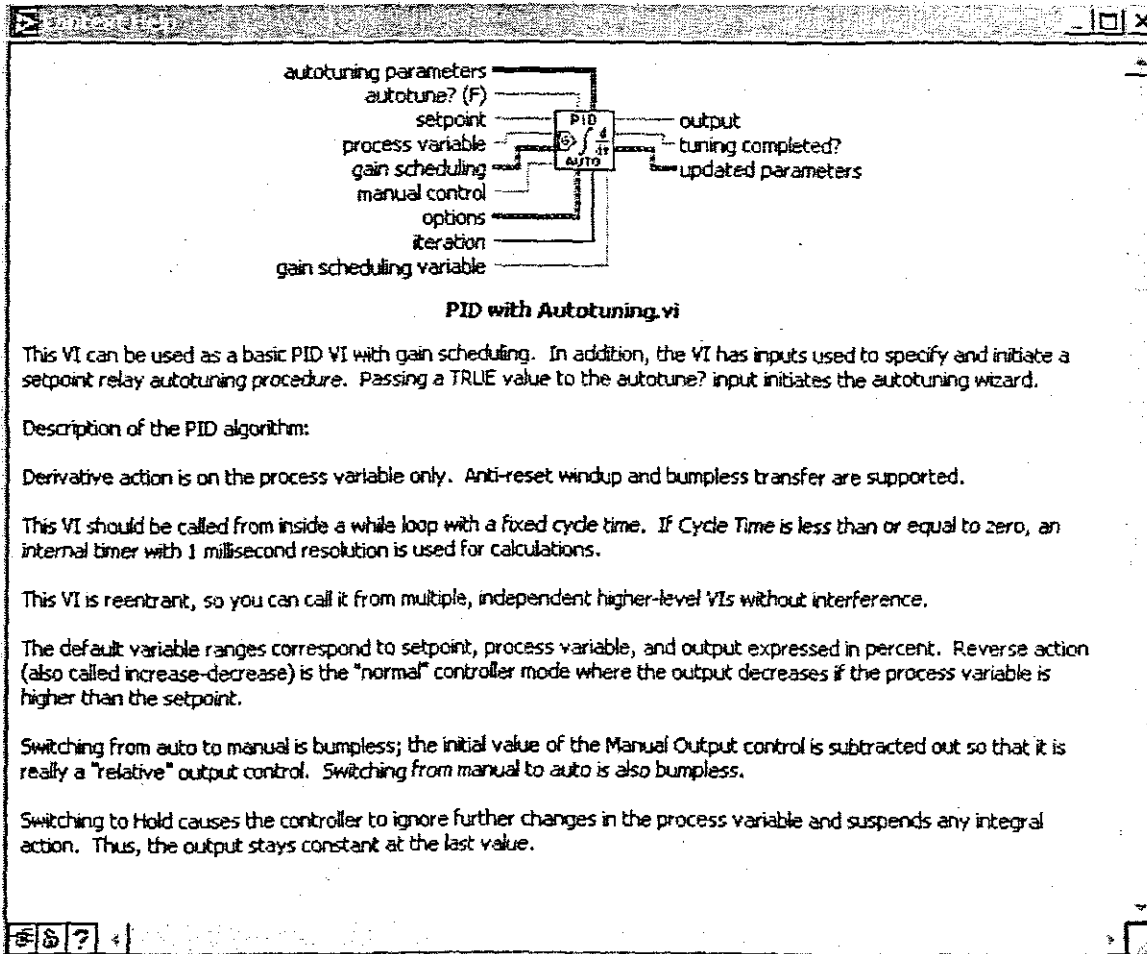


Figure 5.3: PID with auto-tuning

5.2.2 The Auto tuning Algorithm

PID controllers built in the bio-controller have to be tuned according to the requirements of the process in real time. The automatically tuning of the controllers can improve their performance.

Figure 5.4 illustrates the way auto tuning is implemented. The operator specifies the set points (SP); the process value (PV) is received through the serial port. The auto

tuning procedure is implemented directly with the PID auto tuning VI from LabVIEW™ PID toolbox (National Instruments, 1998).

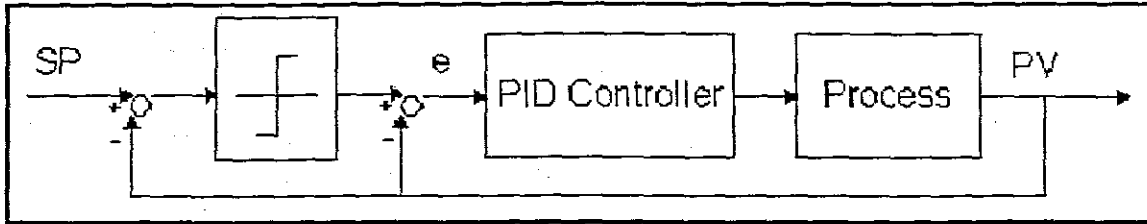


Figure 5.4: Process under PID control with Set point relay

Tuning a PID controller is conceptually simple—observe the behaviour of the controlled process and fine tune the controller's proportional (P), integral (I), and derivative (D) parameters until the closed-loop system performs as desired. However, PID tuning is often more of an art than a science. The best choice of tuning parameters depends upon a variety of factors including the dynamic behaviour of the controlled process, the controller's objectives, and the operator's understanding of the tuning procedures.

Self-tuning PID controllers simplify matters by executing the necessary tuning procedures automatically. Most observe the process' reaction to a disturbance and set their tuning parameters accordingly. However, no two go about accomplishing those tasks in the same way.

"Heuristic" self-tuners, for example, attempt to duplicate the decision-making process of an experienced operator. They adjust their tuning parameters according to a series of expert tuning rules such as "IF the controller overreacts to an abrupt disturbance THEN lower the derivative parameter."

Model-based approach

A more common approach to automatic parameter selection, however, involves a model.

Mathematical "model" of the process is an equation that relates the present value of the process output to a history of previous outputs and previous inputs applied by the controller. If the model is accurate, the controller can predict the future effect of its present efforts and tune it' self accordingly.

For example, a process that reacts sluggishly to a step input can be modelled with an equation that gives the current output as a weighted sum of the most recent output and the most recent input. A self-tuner can choose the weights in that sum to fit the model to the observed process behaviour. With the model in hand, the self-tuner can go on to determine how much proportional, integral, and derivative action the process can tolerate. In the case of a sluggish process, the model will show that the controller is free to apply aggressive control efforts. The self-tuner sets the parameters P, D, and I to their relatively high values.

Variations on the theme exactly how high or low the tuning parameters should be set depends on the performance objectives specified by the operator. If, for example, the settling time is to be limited to some maximum value, analyzing the time constant and the dead time of the process model can determine the required tuning parameters. On the other hand, if excessive overshoot is the operator's principal concern, the controller can be configured to select tuning parameters that will limit the rate of change of the process variable. Self-tuning controllers also differ in their data collection techniques. Some apply a series of artificial disturbances to the process in order to observe how it behaves. Others make do with data collected during normal loop operations. The latter approach limits the waste and inconvenience caused by intentionally disturbing the process, but generally produces much less useful information about the process' behaviour. Which of these many variations is appropriate for a given application of self-tuning control is up to the operator. A single universally applicable technique has yet to be developed.

The toolbox uses Ziegler and Nichols' heuristic methods for determining the parameters of a PID controller. When auto-tuning vi's, select one of three types of loop performance: fast (1/4 damping ratio), normal (some overshoot), and slow (little overshoot). Refer to the following tuning formula tables for each type of loop performance.

5.2.2.1 Description of the input and outputs of the vi.

The developed auto tuner is characterized by the following input and output variables:

Inputs:

- Measured process variable from the serial port: Temperature, pH, dissolved oxygen, and agitation.

Outputs:

- PID setting: K_c , which is the proportional gain, T_i , which is the integral time in minutes. A value of 0 disables integral action, and T_d , which is the derivative time in minutes. A value of 0 disables derivative action.

5.3 Conclusion

The developed two-layer control structure allows adaptive control through the process of repetitive model parameter estimation and process optimization. The decomposition method used to solve the problem for optimization (optimal control) allows the overall problem to be decomposed and simplified. The complexity of calculation is reduced by using the optimal trajectories of the physiochemical variables, as changing the set point of the local PID controllers allows the time varying nature of the process to be counted out.

Chapter 6

Monitoring and real time control implementation

This chapter presents algorithms and programmes for monitoring and control in real time of fed-batch fermentation process for the production of yeast, using Personal Computer, B Braun Biostat ® C Lab scale fermentation unit and LabVIEW™ as driver software. Hardware and software parts of the control systems are described and discussed. LabVIEW™ code is presented. The control system automates the operation of the lab scale fermentation unit. It is safe, stable and operational

6. Introduction

For the past decades research has been conducted on microbiology and microelectronics with both fields of technology succeeding tremendously (Montague et al., 1988), with *microelectronic* being more advanced. The next area of interest for the researcher was to integrate the two fields that is, to use microelectronics technology (Control and Instrumentation) to speed up the processes in microbiology.

The technology realized in this study is based on off the shelve hardware components. The latter is due to the latest development in control technology globally. It has been observed that a lot of hardware already exist (Ahmed el all, 1984), but what is missing is the brain (software programs) to customize the hardware for the specific purpose. This resulted in bigger companies developing software that is more general to accommodate a lot of customers. The draw backs of this approach are that the programs become so complicated that the operators ignores the package and use their standard procedures resulting in a loss from the buyers side and the computer not being used to its fullest.

This study is addressing the constraints mentioned above by developing a software control system for the Lab scale unit. The software is custom specific, but with minor changes it can be adapted for other industrial application. The developed programmes are in two application modules. The first module is for *monitoring and control* of the process using simple commands from the user interface window. The second module

is for the optimisation of the process on the basis of the developed mathematical model. The two modules are discussed and explained in the next paragraphs.

A control system has been developed using graphical programming language 'G', called LabVIEW™ (National Instruments, 1998). The programmes and the control system are implemented on a standard PC, running Windows 98 with LabVIEW™ as driver software. The connection between PC and the fermentation unit is through a standard IFB RS-422 box. The IFB RS-422 box converts RS-422 signal from the bio-controller into RS-232 signal to the computer and the visa versa.

6.1 HARDWARE ENVIRONMENT

The hardware of the developed control system consists of Bio-controller [Biostat® C from B.Braun International GmbH], IFB RS-422 and Personal Computer.

The hardware integration of the control system is illustrated in Fig 6.1.

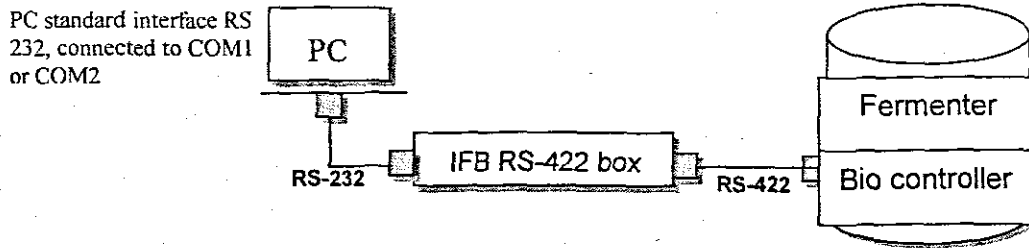


Figure 6.1: Overview of the hardware components

6.1.1 Bio-controller Biostat® C from B.Braun International GmbH

The whole infrastructure is based on the local control unit called bio-controller in the considered case Biostat® C from B. Braun Biotech International GmbH. Industrially bio-controllers are used for the purpose of acquiring, and controlling important variables from different types of fermentation processes.

The Biostat® C is a compact laboratory scale fermentor with in-situ sterilisable culture vessels up to 30l working volume [<http://www.bbraunbiotech.com/>] Fig 6.2.

The Biostat® C is delivered complete and ready to use, with probes, four pumps, vessel with drive, digital measurement and control as well as connections to external systems. Bioprocess capabilities of the Biostat® C include batch, fed-batch and continuous processes alike. And, for cell culture, a full line of accessories like spin filters, impellers, and gasmix units are also available. The modular design consists of the supply unit, the culture vessel and a digital control unit. The open frame design of the supply unit provides ready access and ease of maintenance for all process piping and actuators.

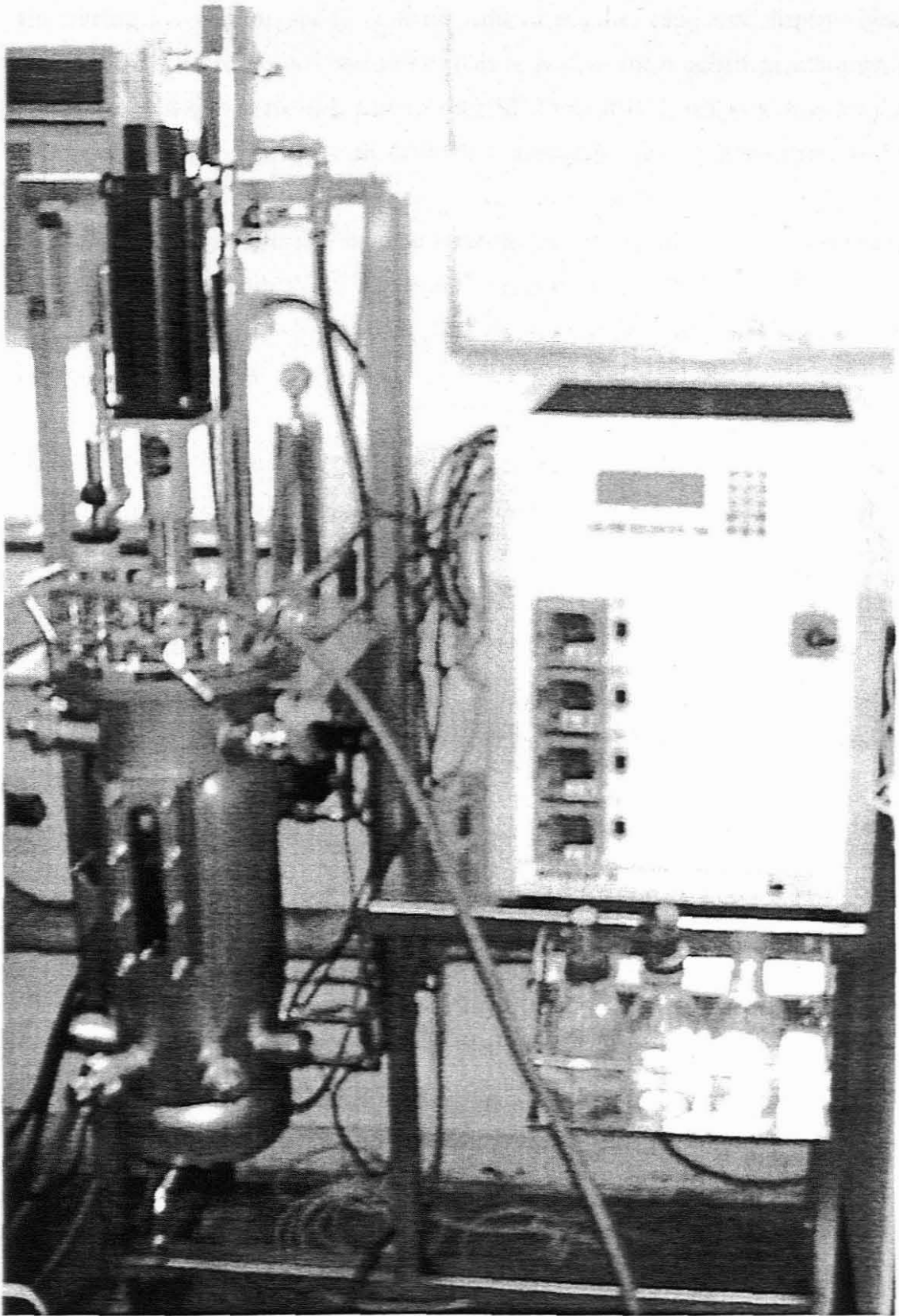


Figure 6.2: B. Braun Biostat ® C bio-controller

The intelligence and the ability of these units to acquire, compress, display using Liquid Cristal Display (LCD), and still be able to send all the acquired data through a commonly used communication port called RS232 and RS422, makes it easy for the control engineers to integrate their developed control system and intelligence to the existing hardware.

The fermentation variables will become inputs to the optimisation program that has to been developed in LabVIEW™, see section on control structure and realisation above.

6.1.2 IFB RS-422 box

The IFB RS-422 box's main function is to convert RS-422 signal from the bio-controller into RS-232 signal to the computer's serial interface and the visa versa. The serial interface of the computer (COM1 as default) is connected with the IFB RS-422 box with the serial cable supplied. The side labelled with IFB RS-422 must be connected to the IFB RS-422 box and the other side must be connected to the computer's serial interface (Braun, 1996).

6.1.3 Personal Computer (PC)

The computation time, user interface graphics and the complexity of the developed algorithms require the computer with the following minimum specifications.

- Pentium® Processor
- 128MRAM
- 10GHD
- 8MB VGA Graphic card
- Preferable a 17" Monitor for better view.

The application is developed under the Windows environment and using LabVIEW™ as drive software. Windows 98 is used as an operating system. The reason for using Windows environment is because the institution i.e. UCT had it as an available operation system and also students are used to windows applications.

6.1.4 Description of serial interface

Serial Interface is the most common transmission method in which information bits are sent sequential on a single data channel using shared boundary defined by common physical interconnection characteristics, signal characteristics and

measurement of interchanged signals. The definition clearly defines that the physical interconnection characteristics have to be common implying that a certain set of basic physical standards had to be developed to make communication system possible.

One of the most enduring physical standard, which has been developed, is the EIA-RS-232C standard (developed and issued in the USA in 1969). The EIA-RS-232C was developed to define the electrical and mechanical details of the interface between Data Terminal Equipment (DTE) and Data Communication Equipment (DCE), which employ serial binary data interchange. The generic terms of RS-422 and RS232 are used to represent the EIA/TIA-422 and EIA/TIA-232 Standards.

The RS-232C has its own drawbacks associated with it including its susceptibility to noise over a long distance transmission line. Consequently, several other EIA interface standards have been developed which overcome some of these limitations. Those that have become most commonly used for instrumentation and control systems are RS-423, RS-422, and RS-485.

→ **The RS-423 Interface standard**

Unbalanced system similar to RS-232C, with increased range and data transfer rates, with up to 10 line receivers per line driver.

→ **The RS-422 Interface standard**

Balanced differential system, with same range as RS-423 but increased data rates, with up to 10 line receivers per line driver.

→ **The RS-485 Interface standard**

Balanced differential system, with same range as RS-423 but increased data range with up to 32 line receivers possible per line.

The RS-232C and RS-422 are discussed in the next paragraphs as they are applicable in the dissertation and are the one of the most commonly used standards in industry.

6.1.4.1 The RS-232C Interface Standard for Serial Data Communications

The RS-232C belongs to the category of unbalanced data transmission type. Each signal that transmits in an RS-232C unbalanced data transmission system appears on the interface connector as a voltage with reference to a signal ground. For example, when data is transmitted (TD) from the host PC (DTE) it appears on pin 2 with respect to pin 5 (signal ground) on a DB-9 connector. This voltage is negative; the line is idle and alternate between that negative level and a positive level when data is sent with a

magnitude of +5 to +15 volts. The following are the main features of the equipment that uses the RS-232 serial interface:

- Communication is Point-to-Point,
- Suitable for serial, binary, digital, data communication (data is sent bit by bit in sequence),
- Most RS2-232 communication data are in the ASCII code, although that is not part of the standard,
- Communication is asynchronous (fixed timing between data bits, but variable time between character frames),
- Communication is full-duplex (both directions simultaneously) with a single wire for each direction and a common wire,
- Voltage signals are:
 - Logic 1: -3Volts to -25Volts (Mark)
 - Logic 2: +3Volts to +25Volts (Space)
- The communications signals voltages are “unbalanced”, making them more susceptible to noise,
- Reliable communication up to about 15m,
- Data rates of up to about 20 kbps.

In spite of its popularity and extensive use, it should be recalled that the RS-232 interface standard was originally developed for interfacing Data Terminals (e.g. printers) to Modems. The RS-232 serial interface is used in the dissertation between the PC and the IFB RS232/RS422 converter Fig 6.1.

6.1.4.2 The RS-422 Interface Standard for Serial Data Communications

Just after the issue of the RS-232 in 1969, the RS-422 standard was developed and introduced in the early 70's defining a 'balanced' (or differential) data communications interface using two separate wires for each signal. This permits very high data rates and minimizes problems with varying ground potentials because the ground is not used as a voltage reference (in contrast to RS-232).

In a balanced differential system the voltage produced by the driver appears across a pair of signal lines that transmits as only one signal. A balanced line driver produces a voltage from 2 to 6 volts across its A and B output terminals and will have a signal ground connection. The main advantage of the balanced line driver is that it is not

susceptible to noise and it can cover a long distance when it comes to cables lengths with the following features:

- o Distance of up to 1200m,
- o Data rates of up to 10Mbps,
- o Only one driver is permitted on a line,
- o Up to 10 line receivers can be driven by one line driver).

The RS-422 serial interface is used in the dissertation between the IFB RS232/RS422 converter and the Bio-controller Fig 6.1. The latter is from the advantages that the RS-422 has over the RS-232 hence the conversion is necessary.

6.2 SOFTWARE ENVIRONMENT

6.2.1 Tools And Methodologies

The software is developed in LabVIEW™ version 6.0.1 on Windows98/NT machine. Since the LabVIEW™ language is relatively new, a brief summary of its development environment, coding concept and built-in libraries is prepared. LabVIEW™ is a compiled, general-purpose language with all the normal control concepts and data types. LabVIEW™ differs from the textual programming language in many ways.

It is using graphical programming language called “G” and icons are dragged and groped to define functionality of the program. LabVIEW™ software routines are called “Virtual instruments” or “VI”, Fig. 6.3.

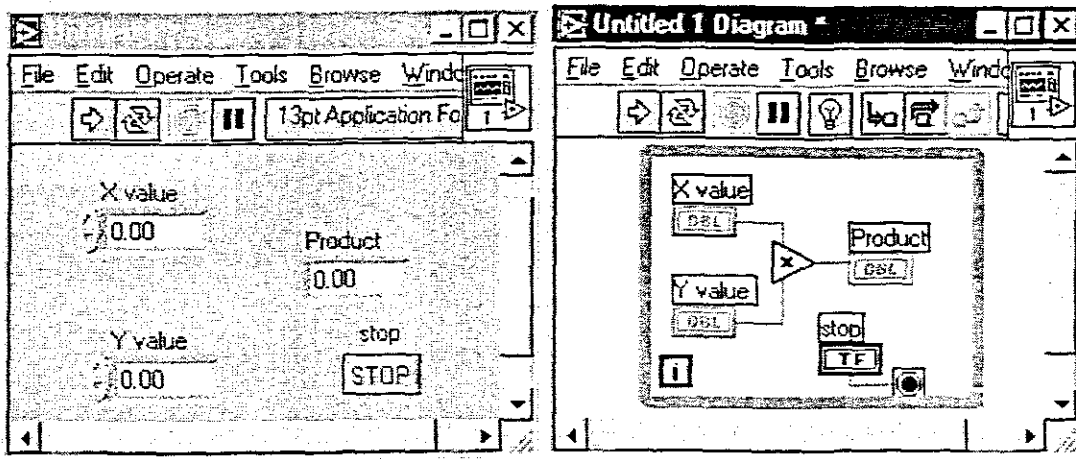


Figure 6.3: LabVIEW™ front panel and processing code diagram

Each VI is composed of three parts: the front panel (GUI), code diagram, and the sub VI connector.

The VI shown in Fig. 6.3 uses the front panel (GUI) on the left to get X and Y values from the operator and computes, $\text{Product} = X * Y$ in a loop in the diagram on the right until the loop is stopped using the STOP button and that will exit the while loop. A VI may call any other VI as a “subVI”. A collection of VIs may be stored in libraries (*.llb files).

The whole evolution of graphical programming language has made a tremendous increase in the developments of control technology in general. One of the advantages is that it is very easy to learn programming in “G” and also, it gives a good visual idea of how the final product will look like giving the operator an opportunity to give some of his views on the operators display. The former and latter would not have achieved with the text based programming language like C, C++ etc. The main drawback of graphical programming is the fact that processing time of programs is a bit slow and, because the operator can give his opinion before the final program is finished makes the program very complicated. The latter does not make any difference in our application, because the fermentation processes are relatively slow. The execution time in most cases is affected by the way the source code is structured. As the result suggested software architecture is given (Steve Orth, 1999)] in the Fig. 6.4.

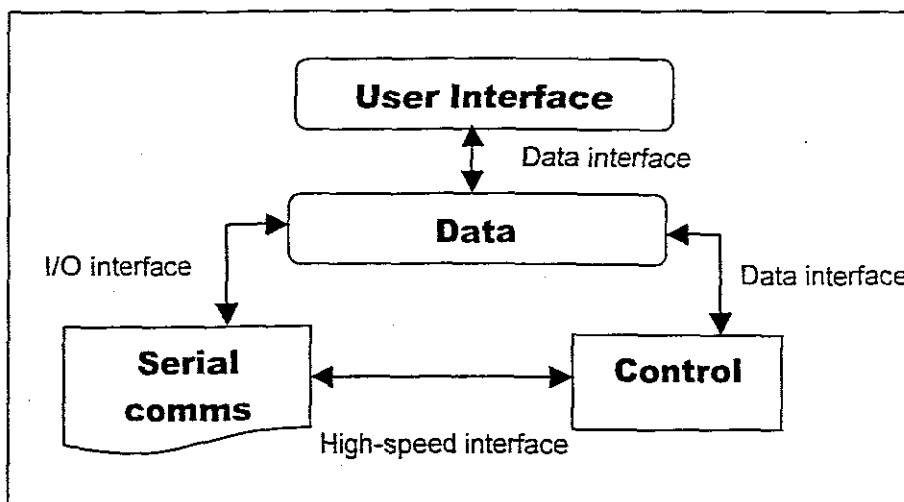


Figure 6.4: Suggested software architecture

6.2.2 Main LabVIEW™ Libraries For The Design

In developing the serial communication software between the Biostat® C and the PC heavy use was made of two built - in libraries:

1. **Serial Compatibility Vis** (Sub palette to serial compatibility) allows reading, writes, closing, initialisation, number of bytes from the serial interface.
2. **File I/O Vis** has several submenus for logging data, writing data to a file, etc. Allows data storage for acquired data in the application program.

These libraries are available as submenus to the main palette, Fig. 6.5.

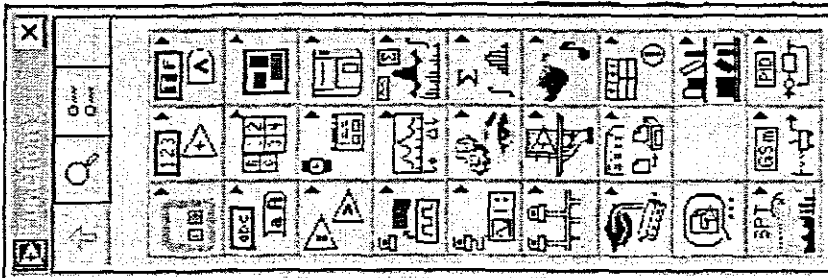


Figure 6.5: Main Function menu

LabVIEW Function Palette

It includes 25 sub palettes of built-in nodes and Vis to perform various tasks. The 21st button at the bottom allows entering of any subVI via a file dialog. The available palettes are: Control, Numerics, Booleans, Strings, Arrays, Clusters, Comparison, Time & Dialog, File, etc.

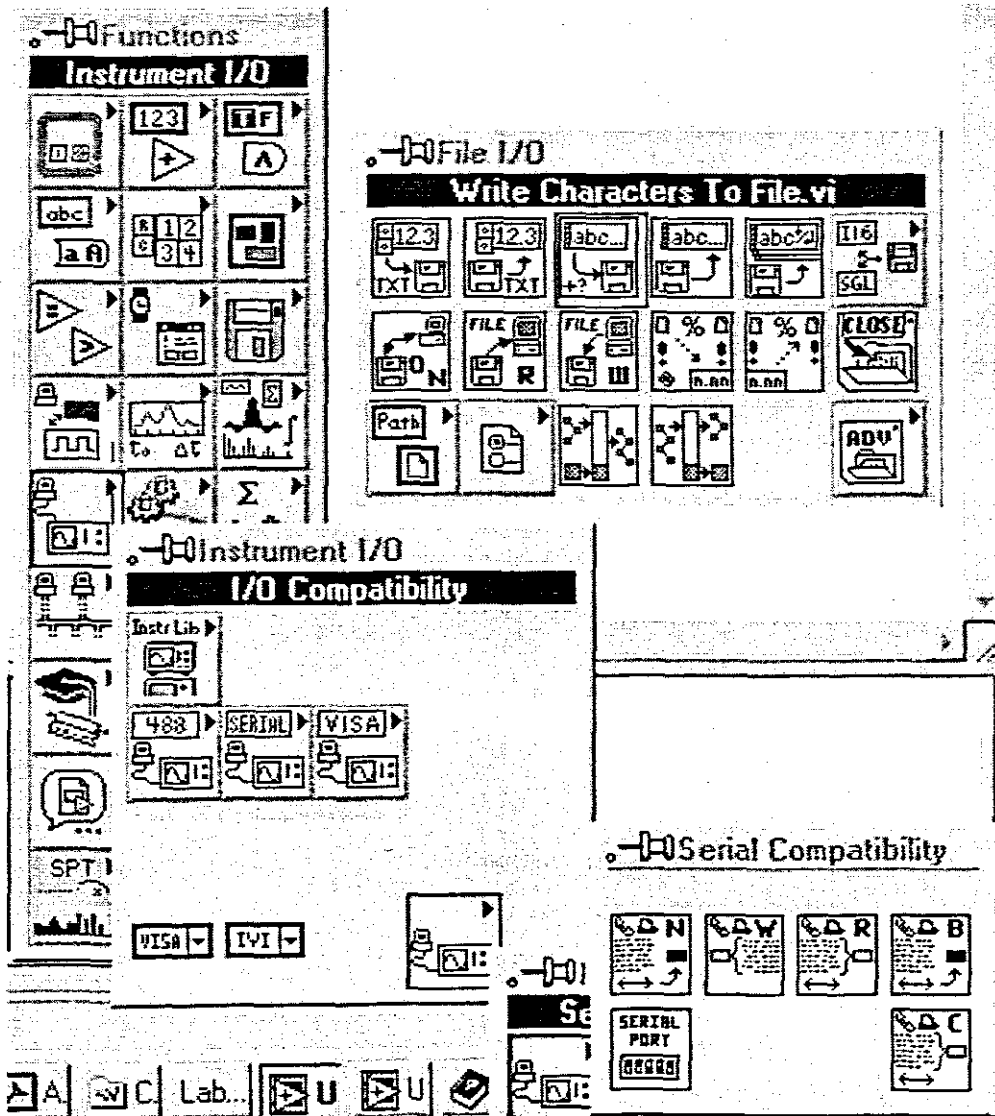


Figure 6.6: VI Menu and Sub menus

These were used extensively in the design. The serial communication sub palette handled all external connections to the Bio-controller. The File I/O took care of all data storage.

6.2.3 Description Of Serial Interface Between Host And Control Units Of The B.Braun Biotech International GmbH

The communication between host computer and the substation is based on master slave principle i.e. the data transmission can only be started by the Host Computer provided the REMOTE (REM) key on the bio-controller is ON. A communication between the Substations themselves (multi drop connection) is not possible. The host computer can control the data transmission to a Substation via commands.

Commands are available for:

- DR requests for process data from the subsystem

DS send process data to the subsystem

SU subsystem initialisation

An address field in the command telegram selects a specific Substation. For each Substation the address can be manually set.

6.2.3.1 Substation Initialization

Prior to the first data request the host computer must initialize the corresponding Substation (command DR). The initialisation command sends parameters to the Substation and thus adapts the data transmission to the requirements.

Without initialisation the Substation sends an error telegram after receiving a data request command (DR).

6.2.3.2 Break-Generation Via Host System

Before sending a command to the Substation a break condition must be generated on the interface in order to switch the connected Substation to the "Receive" mode. The latter is achieved with the break function from the serial compatibility submenu. The duration of the break (depends on the baud rate) should be approximately the same as the transmission time for two characters. After resetting the break a 2ms delay time is recommended before sending characters. In each case the "Break" condition must be generated prior to each command to the Substation by the Host system.

6.2.3.3 Remote Mode

The Substation has an ability to switch between two modes of operation i.e. "Local" and "Remote mode". The latter is achieved by using the "remote" key from the substation. But only the "Remote" mode allows a control via the Host Computer, i.e. only in the "Remote" mode the Substation accepts the set parameters of the Host System. Except for the initialization command and the setting of the real time and the process time, all commands with which parameters in the Substation can be changed will be responded by an error telegram if the Substation is not in the "Remote" mode.

If the host computer sets the process parameters in the "Remote" mode all other functions, direct access included (Shut down, Acknowledge etc.) can be realized at any time. A change from "Remote" to "Local" is only active (delay possible) when a running data transmission/processing of one block is finished.

6.2.3.4 Reserved Control Character

The following characters are required for data transmission control.

- * - Start character Host telegram
- # - Start character Substation telegram
- : - Delimiter for data fields
- @ - End mark for each telegram
- / - Delimiter for date and time and for initialisation formats
- , - Delimiter between integer fields of date and time

The connection between PC and Biostat ® C has to use the following built in unit protocol.

6.2.3.5 Telegram Format

All telegrams must have the following format. Depending on the data group, however, the message field can change in its structure. The status field always indicates the actual status of the telegram sender.

SC	ADR	:	COMMAND	:	SEQ	:	STAT	:	MESSAGE-FIELD	:	CS	EM
----	-----	---	---------	---	-----	---	------	---	---------------	---	----	----

- SC Start character
- ADR Unit address
- SEQ Sequence
- STAT Status of sender
- CS Check Sum
- EM End Mark

MSG – FIELD (Message Field) have the following structure:

DES	:	NO	:	VAL	:	NO	:	VAL	:	:	:	...	:	DES	:	NO	:	VAL	:	...
-----	---	----	---	-----	---	----	---	-----	---	---	---	-----	---	-----	---	----	---	-----	---	-----

- DES = Designation (PV, CS.)
- NO = Channel Number
- VAL = Value

Example 1: Data request (DR)

The following commands initialize the Biostat and request the process values (PV), controller setpoint (CS) as well as controller outputs (CO).

*1:SU:0:0:PV:5:2:1030@

*1:SU:0:0:PV/CS/CO/CM:5:2:1611@

Acknowledgement telegram for response from the Biostat:

#1:DR:0:16:PV:1:124.6:2:567.3:::15:250.0:16:240.5:XXXX@

Example 2: Sending commands

*1:DS:0:0:CS:3:625.0:CM:3:1:PA:3:0:0:656.6:1:0:0:0:2858@

- Controller Setpoint No.3 on pH 7.5

- Controller Mode No.3 on Auto

- Alarm parameter PV No.3 Limits Low=0/High=8pH,Mode=Enable,2xReserve

Acknowledgement telegram:

#1:DS:0:17:619@

6.2.3.6 Response Telegrams Of Substations

The contacted Substation responds within the Timeout limit (500ms) with the required response telegram (specification see below) or with an error telegram if all the following conditions are given.

1. Command telegram begins with * and ends with @
2. Sent check sum = calculated check sum
3. Received address = set address

If one condition is not given none of the connected Substations will respond.

6.2.3.7 End Of Data Transmission

The data transmission is only finished if all data blocks (cf. 2.14, block transmission) of one Substation telegram are correctly sent to the Host. A new response telegram is only sent to the Substation if the last data block has been sent to the Host and if the Host Computer has then sent a new data request command with the sequence 0.

6.2.3.8 Check sum

The check sum is formed by addition of all ASCII values from the start sign (* or #) up to and including the delimiter (:) directly in front of the check sum. The calculated

value cannot exceed the value of 32385 and is variable dependent on the message length.

6.2.3.9 Requests And Commands

The REMOTE (REM) key on the control unit must be switched ON. The latter allows the host computer to take control over the bio-controller.

The sequence into which the host communicates with the control unit (Substation) is shown below.

1. Initialisation

The substation needs to be initialised. A command to fulfil the above needs to be issued.

2. Process Data Request.

This command allows the transmission of the data (as determined in the initialisation) from the Substation to the host. All data must have the same time basis.

3. Transmission Of Process Data.

This command allows the data transmission from host to Substation (if the Substation is in the Remote mode). All process data with the specified transmission direction.

Sending commands has no special sequence as seen in Example 2 above.

6.2.4 Developed software

The software of the control system has three main software modules corresponding to the three main tasks for adaptive control implementation.

- Data Acquisition (DAQ) for monitoring.
- Optimisation for repetitive calculation of model parameters, optimal trajectories and local controller parameters.
- Control implementation for set up of the Bio-controller parameters to fulfil the designed control system.

6.2.4.1 Description of the DAQ program

The DAQ module provides the operator with the ability to monitor and supervisory control of the bio-controller. The description of the operation of the user interface and the functionality of the developed module is described in chapter 7 (Operator's

Manual). After the developments of the DAQ module a need for operator's manual was evident, hence it is given as a separate chapter.

The module is utilised by the model on updating its parameters. The model is at the bottom layer of the control system architecture, reason being it communicates with the serial communication port directly. The schematic of the DAQ program is given on Fig 6.7.

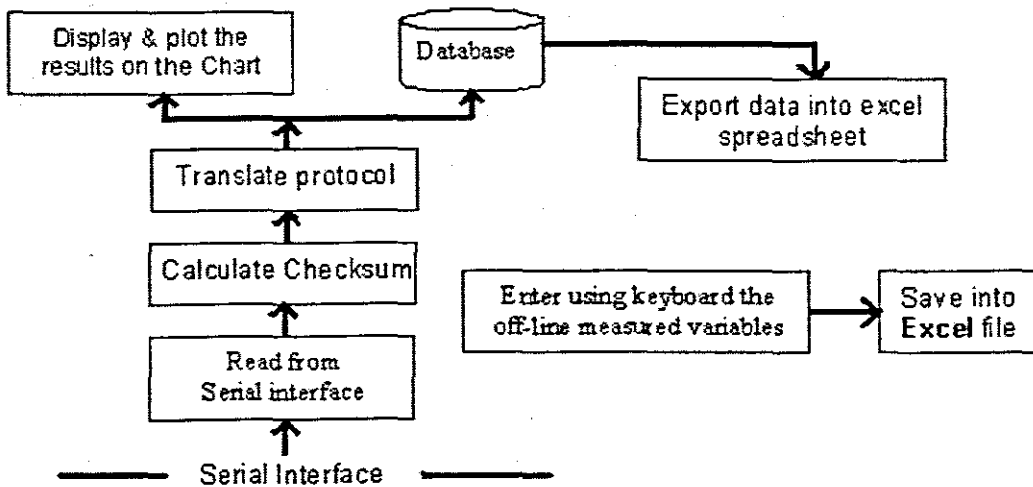
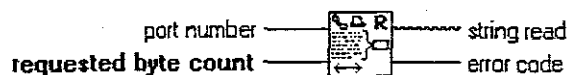


Figure 6.7: Schematic of DAQ program

Serial port read/write

Serial Port Read and Serial Port Write labVIEW serial functions are used to read and write data to and from the serial interface (6.2.2). The input to these functions is the port number and requested byte count and the output is an ASCII string of data. The requested byte count is the number of bytes received or sent to the serial port specified in the port number input.



Serial Port Read.vi

Reads the number of characters specified by **requested byte count** from the serial port indicated in **port number**.



Serial Port Write.vi
Writes the data in **string to write** to the serial port indicated in **port number**.

Figure 6.8: Serial Port Read/Write LabVIEW functions

Checksum calculator

The ASCII string from the serial functions is passed to the checksum calculator. Checksum calculator calculates the checksum of the string coming and going to the bio-controller and returns an error on the data format if it exists. The checksum is mainly used by the bio-controller to validate the data that the bio-controller is receiving from the host computer. The latter implies that if the operator changes a set point and sends that to the bio-controller, the developed checksum calculator will get the ASCII string of data, calculate the checksum and send that to the bio-controller. If the checksum is incorrect an error will be received from the bio-controller. A program to calculate the checksum is given in Fig.6.9.

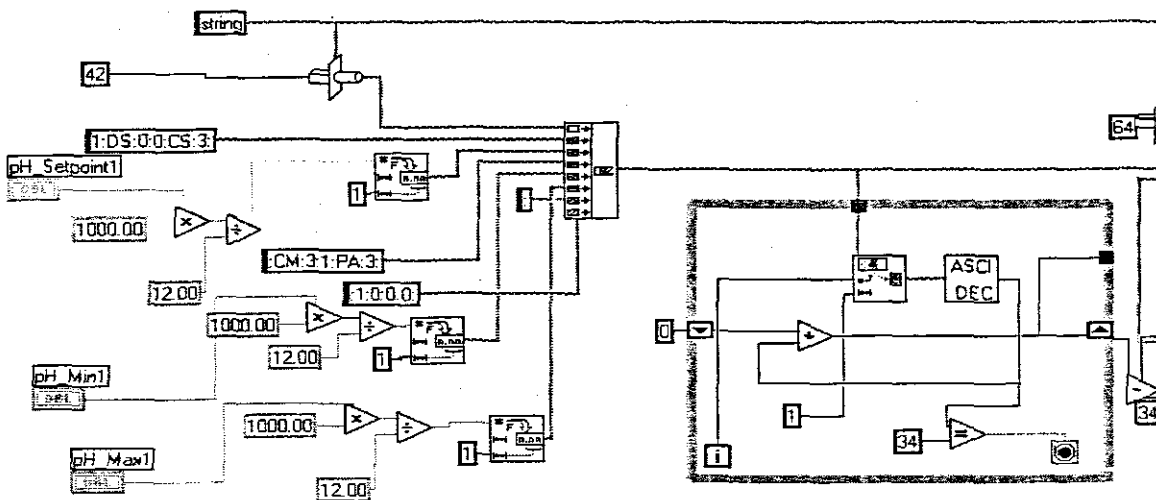
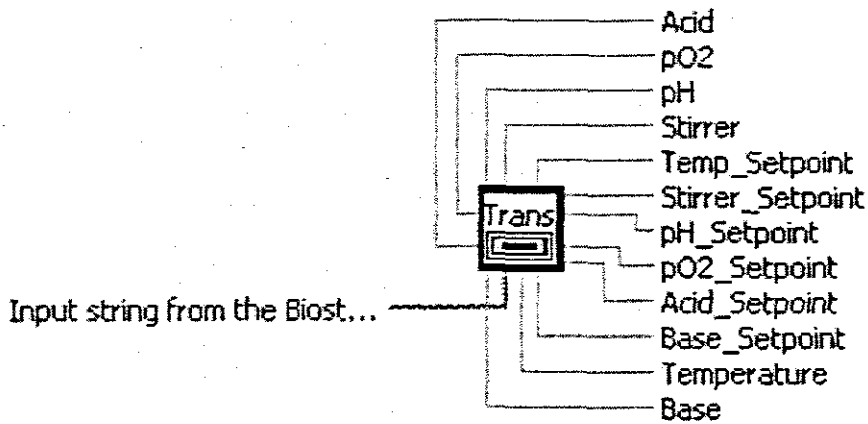


Figure 6.9: Checksum calculation

Translate protocol

A sub vi called Strip_protocol.vi is developed to translate the protocol from the serial port of the Biostart. The sub receives an ASCII string of data from the serial functions described above and translates it into meaning full process variables values that the operator can understand.



Strip_protocol.vi

The input data from the serial port is in ASCII format. The data is not represented as true temperature, pH, etc values. A conversion to proper meaningful values for the process variables is accomplished by using this VI.

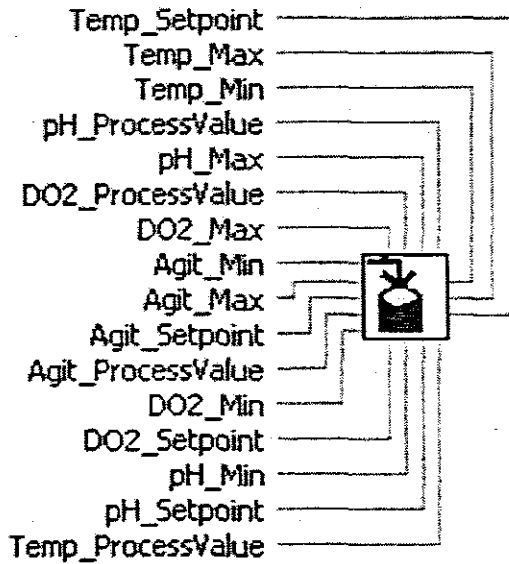
Input --> ASCII, delimited by commas

Output --> Temperature in degrees, pH in pH, Agitation in rpm, etc

Figure 6.10: Strip_protocol.vi

Database

The output of the strip_protocol.vi is sent to the MS access database, process model for updating and to the display interface. The developed sub vi for inserting data into the database uses the Open Database Connectivity (ODBC) and SQL command to access data to and from the MS Access database. The description on how to set up the ODBC is described in the user's manual, chapter 7.



insert_into_db.vi

This vi is used to store the PV values into the local database using ODBC connection. The database used is MSAccess from Microsoft. The latter is a standard database(db) that comes with MS Office for windows system.

Figure 6.11: Insert into database

The tables used to store data in the database are shown in Fig. 6.12.

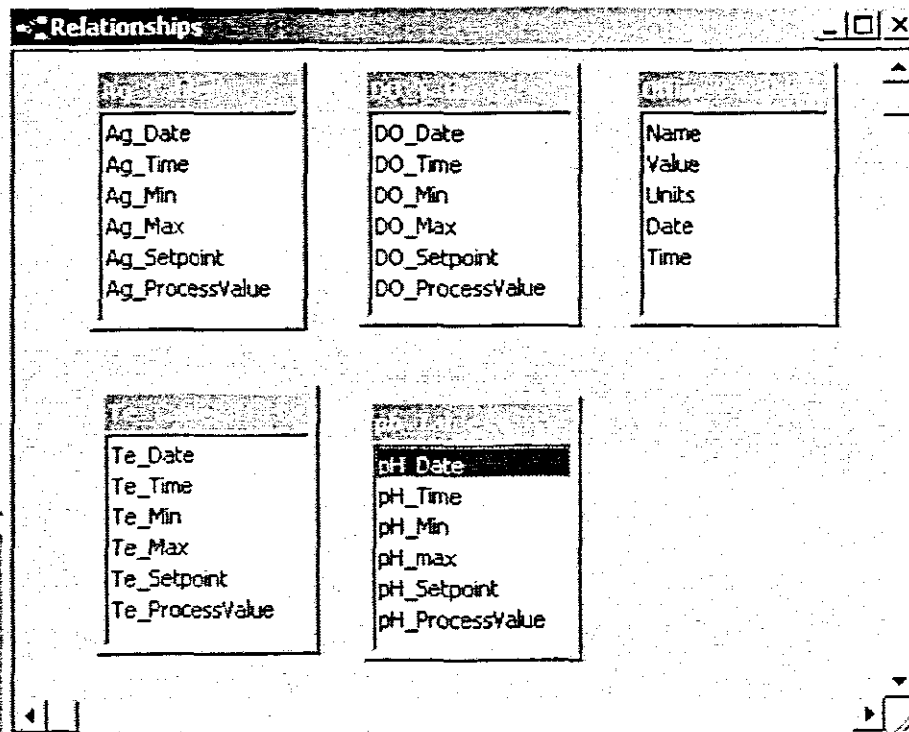


Figure 6.12: Tables of the Process Variables in the database

Export data into MS Excel spreadsheet

The data stored in the local database can be accessed from the database using the developed sub vi. The sub vi allows the user to go back and access old data using date and time index, a utility to sort the data according the operators interest is incorporated in the program.



Reports.vi

This vi receives data from the local database and exports it into excel spreadsheet using date and time index. The option sort the data is also available. The operator uses a REPORTS button from the main display window, on activating the button this vi is invoked and a pop-up window comes to interact with the user.

Figure 6.13: Reports

Input data from the keyboard for the biological variables

A function has been incorporated in the functionality of the program to allow the user to enter the off-line measured variables into the database. The following vi fulfils this functionality. The user interaction of the vi is discussed in chapter 7.



Off_Line_Variables [Off_Line_Variables.vi]

This vi allows the operator to store off-line measured variables in the local database. The data is enter using the keyboard.

Figure 6.14: Off-line measured variables

6.2.4.2 Description of the Auto tuning program

The developed auto tuner is displayed and described in Fig 6.15, based on the LabVIEW PID toolbox vi's.



Autotuner.vi

This VI is meant to be used with the process control package (PID controller) and an MIO-series or Lab-PC data acquisition board. Connect the analog output to the analog input through the resistor-capacitor network shown on the front panel (scroll down to view it). The VI attempts to adjust the analog output so that the input (process variable or PV) equals the setpoint (SP). SP and PV are displayed on a strip chart. You can experiment with different controller tuning methods and try for the fastest settling time with the least overshoot.

The input span is -10 to +10 V, or 20V = 100%. The output span is 0 to +10 V, or 10V = 100%. PV and SP are in Volts. Your MIO board should be set up for differential input and -10 to +10 V range; the output should be unipolar, 10V (these are all factory defaults). If you elect to use other settings, run the configuration VIs as necessary.

The recommended network has a DC gain of 0.33, an effective deadtime of about 5 Sec., and an effective time constant of about 30 Sec.

Figure 6.15: Auto tuner

6.3 Conclusion

The developed software system for data acquisition and real time control implementation is based on LabVIEW. LabVIEW is a widely used technology for building industrial control systems. The software connects in a simple way all parts of the adaptive control approach in one whole. The system is user friendly with help facility to allow easy of operation for the operator. The communication between the computer and the bio-controller is quick and simple. It allows exchange of information for all process variables and change of the bio controller parameters using the computer.

Chapter 7

Operator's manual

7. Introduction

Monitoring and Supervisory control of fed-batch fermentation processes (MSCFP) operator's manual. The MSCFP provides the capabilities to acquire, log data, monitor, estimate the model, optimize, adjust PID parameters, adjust set points to bio controller and be able to query previous data from the database (PVdb.mdb). The document is intended for users, with basic computer knowledge. The latter refers to being able to start and stop Microsoft windows applications. Students, lecturers and researchers in different fields of engineering, i.e. Bio-chemical, Electronic etc, intend to use this document as tool to assist them on operating the MSCFP application during experiments.

7.1 Functional Overview

The MSCFP application is the main program for the full control of fermentation processes. It has the following functionalities

- 1) Ability to initialize the bio-controller at any time whilst the program is running. [**Initialize Button**].
- 2) Ability to acquire and LOG data during experimental period on the local database and to query that data from the database using date and time index and export the data into the Microsoft excel spreadsheet for further analysis and plotting. [**Reports Button**].
- 3) Ability to acquire and log the off-line measured variables in Microsoft Excel spreadsheet using keyboard. [**OFF-line Variables Button**].
- 4) Ability to estimate model parameters. [**Parameter estimator Button**].
- 5) Ability to calculate the optimal trajectories of physiochemical variables from the given moment in time till the end of the process. [**Optimization Button**].
- 6) Ability to recalculate the PID controller parameters using built in LabVIEW™ PID auto tuning toolkit. [**Auto tuner Button**].

- Ability to send control loops settings i.e. Set point, Minimum and maximum values of variables etc. to the bio-controller using user friendly pop up window. **[Control Loops Settings Button]**.

7.2 Main program

The main window Fig 7.1 is what the operator sees when the program starts up. The function of the windows is to display graphically the data in real time as it is acquired from the serial port. The alarms are also displayed as they occur. The main window consists of menu, sub windows for the acquired data, alarm limits, buttons for optimizations, PID auto tuning, model parameter estimation and help.

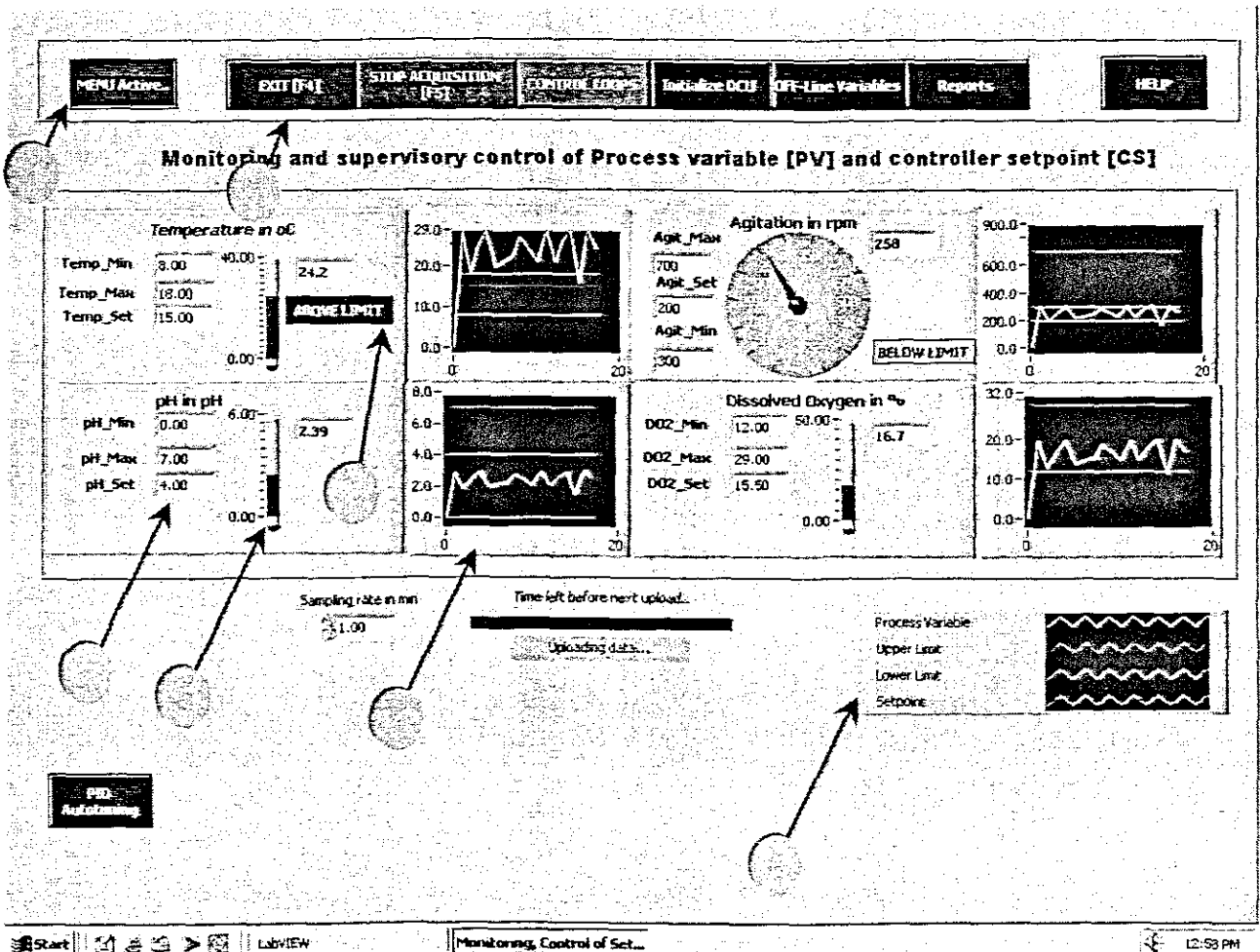


Figure 7.1: Main program window

The main sections for the interaction with the operator are listed below:

- Menu** with the following buttons for; *Exit, Start/Stop Acquisition, Control loops, Initialize DCU, Off-line measured variables and reports.*

The menu pops up on clicking the menu button.

- 2) Indicates minimum, maximum and set point values for the specific process variable in the variable sub window.
- 3) Indicator for process variable in the variable sub window.
- 4) Chart plotting together process variable, upper limit, lower limit as well as set point.
- 5) Alarm indicator
Red indicates **ABOVE LIMIT** and Blue indicates **BELOW LIMIT**.
- 6) Legend indicating the colours used for the process variable, its set point and limits.
- 7) Exit; press this button to exit the main program.

The sections that are not labelled will be discussed separately below as they have pop up windows.

7.2.1 Start acquisition

Press this button to start or stop the data acquisition. After pressing the button it is toggles to STOP ACQUISITION. The figure below displays the functionality linked to the button being pressed. The acquired data gets uploaded from the serial port at the rate specified below. Also an indicator is put up to indicate when the serial port is actually occupied that is when the **Uploading data...** indicator pops up, Fig. 7.2.

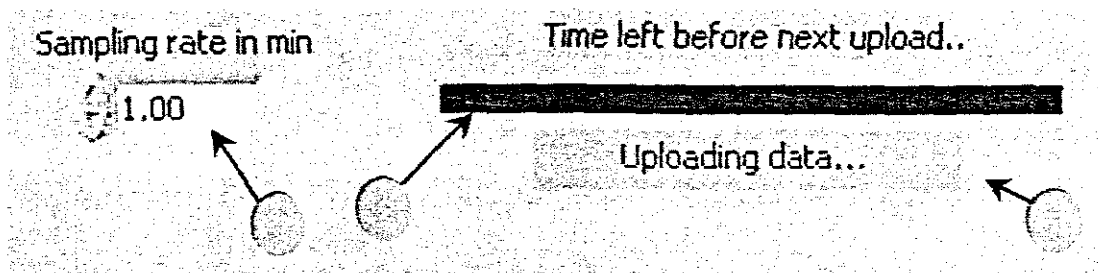


Figure 7.2: Acquisition indicators& controls

□ Acquisition indicators & controls

1) Sampling rate (min)

The operator uses this control to set the sampling rate. This is actually the acquisition rate that is the rate into which the data is being acquired from the serial port logged into the local database and displayed on the chart.

2) Time left before next upload

This is the progress bar displaying the time in the frameworks of one sampling interval left before the next data upload.

3) Indicator; indicating that the data is being uploaded from the serial port.

7.2.2 Control loops settings

The control loops settings windows Fig 7.3 pops up on clicking the control loops button from the main menu. The main function of this window is to send control loops setting to the bio-controller. The bio-controller has an option to execute the sent control loops immediately or to execute them later. This possibility is determined by the mode option discussed in the following sections below.

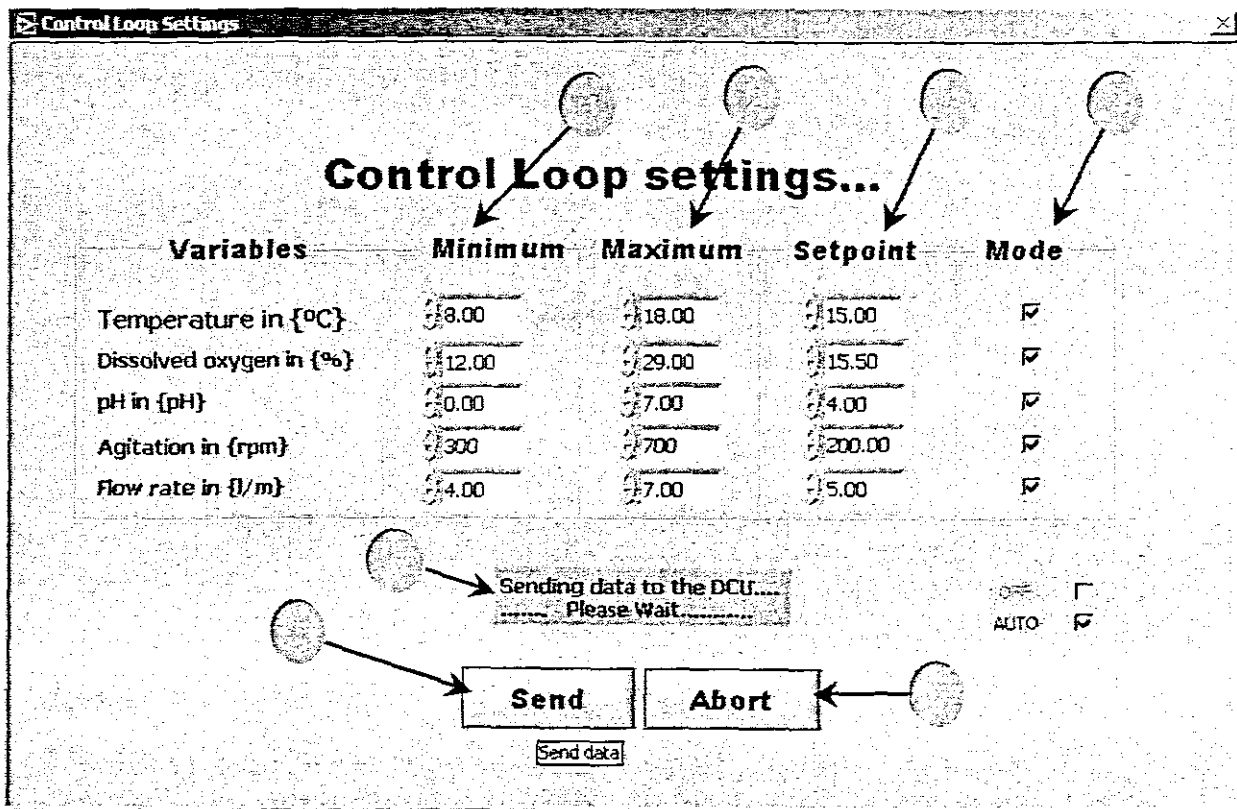


Figure 7.3: Control loops settings window

The main sections for the interaction with the operator are listed below:

□ Control loops settings

1) Minimum

Enter the minimum value for the specific process variable.

2) Maximum

Enter the maximum value for the specific process variable.

3) Set point

Enter the set point value for the specific process variable.

4) Mode

Specify the mode the process should start with. If the check box *is ticked* on pressing the **SEND** button the process will immediately try to reach the specified set point. Else if the check box *is NOT checked*, the process will not try to reach to specified set point rather it will run from the old set point values.

5) Send

After inserting the values on the fields, pressing of the **SEND** button will send data to the bio-controller.

6) Progress indicator

On pressing the **SEND** button the indicator pops up displaying the sending progress to the bio-controller.

7) Abort

Press this button to discard the changes made to the data fields and the process will run from the last send control settings.

7.2.3 Initialize Bio-controller [Substation]

The initialize DCU button is used to initialize the bio-controller at anytime while the process is running, no matter what the status of the **REM** (On the bio-controller) key and the operating mode (Recipe process, sterilization) are.

Prior to the first data request the host computer must initialize the corresponding Substation (command **DR**) (W. Seidler, 1996). The initialization command sends parameters to the Substation and thus adapts the data transmission to the requirements. Without initialization the Substation sends an error telegram after receiving a Data Request (**DR**) command.

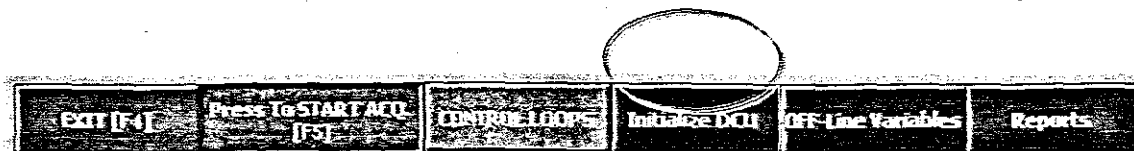


Figure 7.4: Main menu

7.2.4 Off-line variables

Not all the fermentation variables can be measured on-line. The latter results in the variables being measured off line in laboratory. The results are introduced in the PC by the keyboard and are logged in the spreadsheet file for further analysis Fig 7.5. From the menu clicking the off-line variables button invokes the window that allows the operator to enter the variable details. The entered data is then sent to the spreadsheet file using the send button.

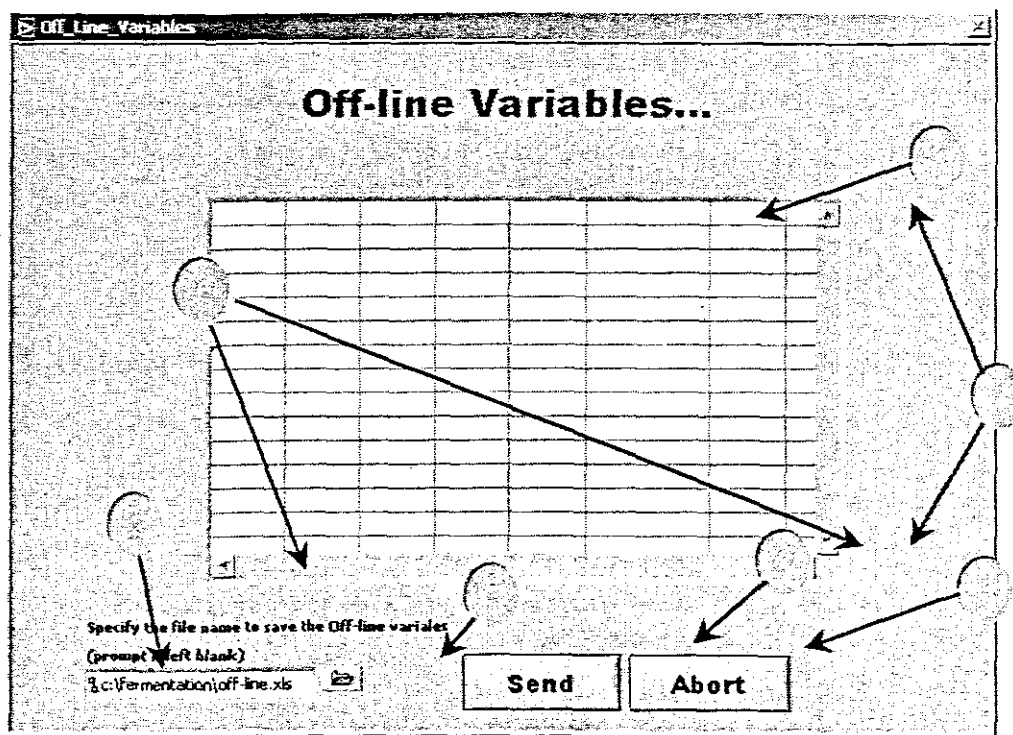


Figure 7.5: Off-line variables

The main sections for the interaction with the operator are listed below:

□ Off-line variables

- 1) Table to enter the variable data. The table is user specific.
- 2) Indicator of the path where to save the off-line variables
<c:\fermantation\off-line.xls> is the default path and file name.
- 3) This button opens or finds a file into which the off-line variables will be saved in.
- 4) Send

After inserting the values on the table, you can press SEND to send data to the bio-controller.

5) Abort

Press this button to discard the changes made to the data fields.

6) Scroll bars

Scroll left and right horizontally as indicated by the arrows.

7) Scroll bars

Scroll up and down as indicated by the arrows.

The name of the variables, time, date, columns of the table have to be specified by the operator depending on the order that suits the operator.

7.2.5 Reports

The reports window pops up on clicking **reports** from the main menu. The main function of the window is to interact with the operator on querying data from the database. During experiments the data from the process is logged continuously into the database at the acquisition rate specified by the operator. This data can then be exported latter from the local database into Microsoft excel spreadsheet for further analysis using this window, Fig 7.6.

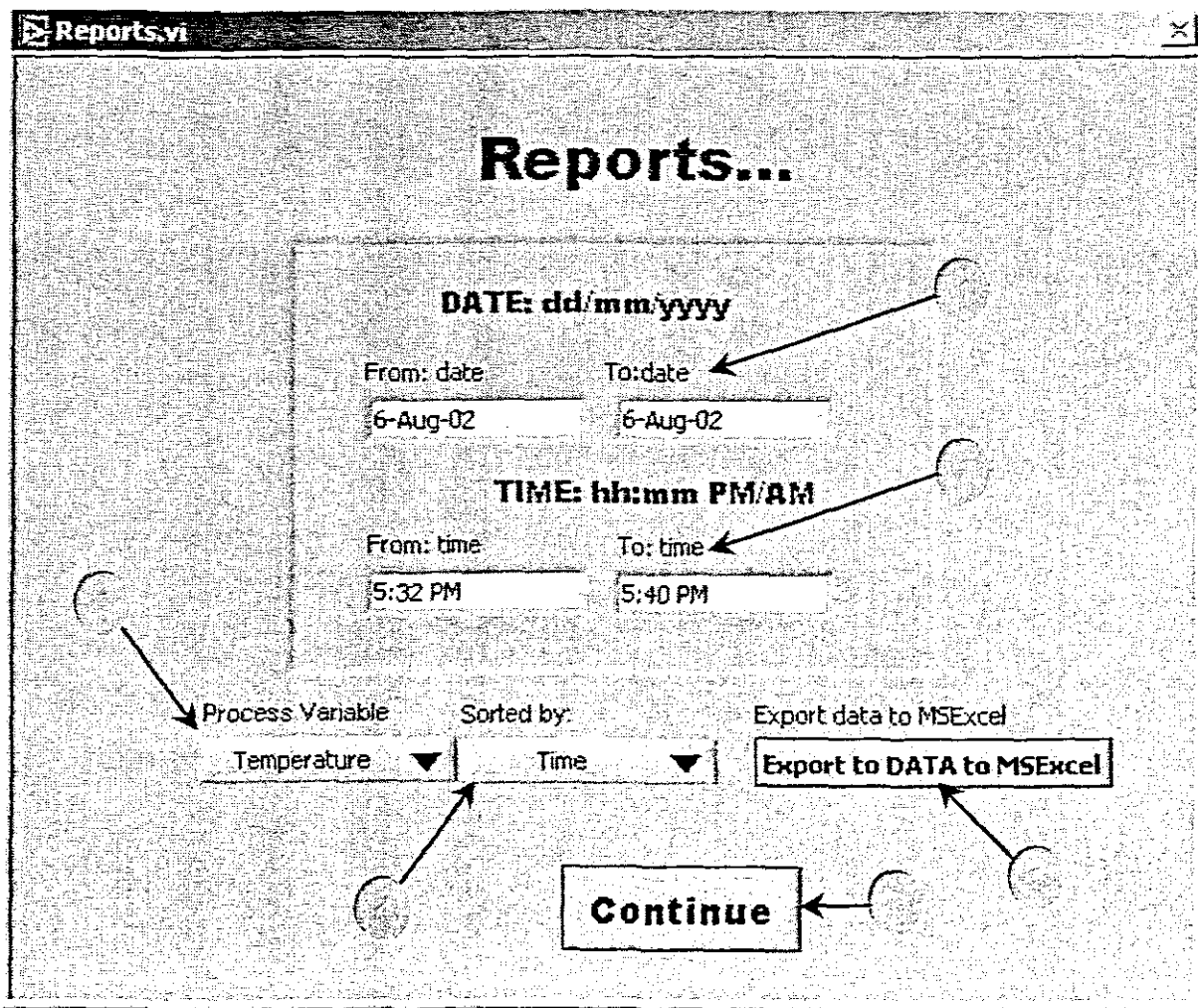


Figure 7.6: Reports window

The main section for the interaction with the operator are listed below:

□ Reports

- 1) Date format dd/mm/yyyy
Enter the *From* and *To* dates as per format highlighted above.
- 2) Time format hh:mm PM/AM
Enter the *From* and *To* times as per format highlighted above.
- 3) Process Variable
Select the process variable you want to query.
- 4) Sorted by
Select the way the output data has to be sorted by in MSEcel. The options are time and process variable.
- 5) Export data to MSEcel

On click the button Export Data into Excel a window pops up for the operator to specify the filename “XXXXX.xls”.

Where “XXXX” is the filename of the file generated from the export.

- 6) Click *continue* to close the reports window.

7.2.6 Help

The help window pops up on clicking the *help* button on the far right of the main program, Fig 7.7. The other help utility is to hold Ctrl down whilst holding down press H. After that move the course to the button the help is need for.

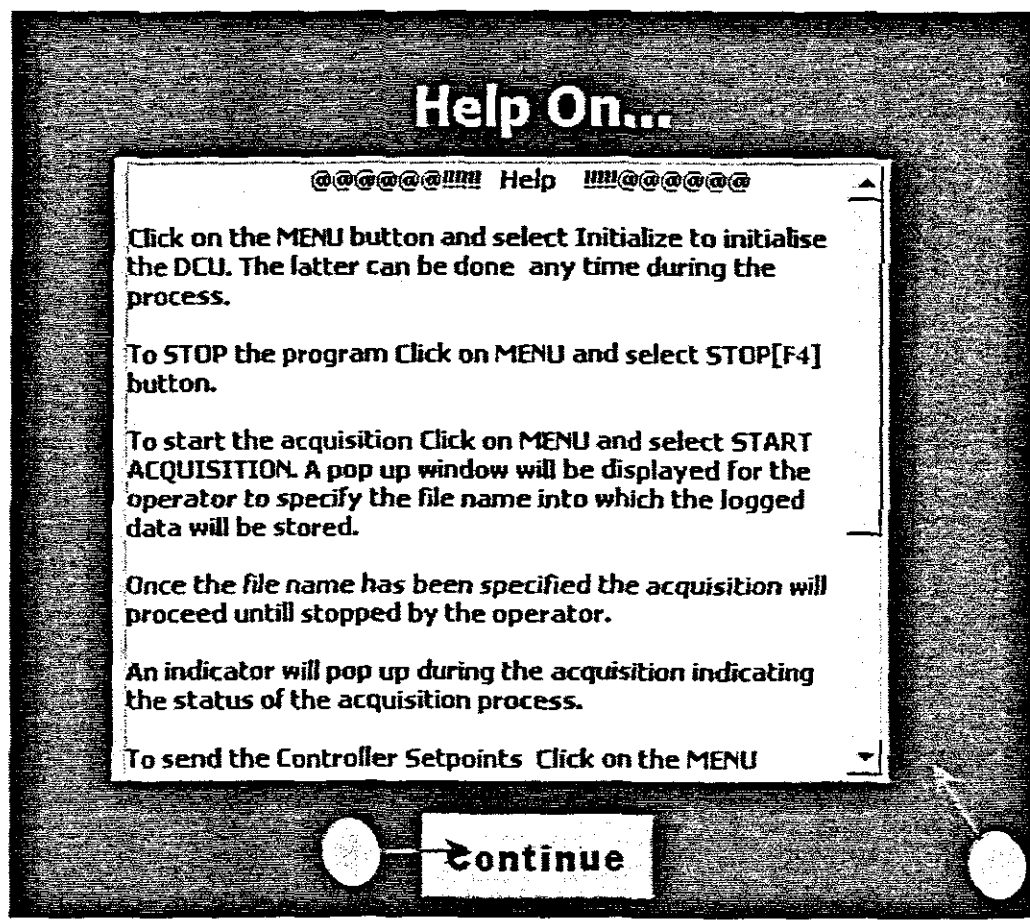


Figure 7.7: Help window

The main section for the interaction with the operator are listed below:

□ Help

- 1) Scroll down for more information.
- 2) Click *Continue* to close the help window.

7.3 Parameter Estimation

From the main user interface panel a button for invoking the parameters estimator is found [Parameter Estimation button]. On clicking this button the window in Fig 7.8 pops up with the following functionality.

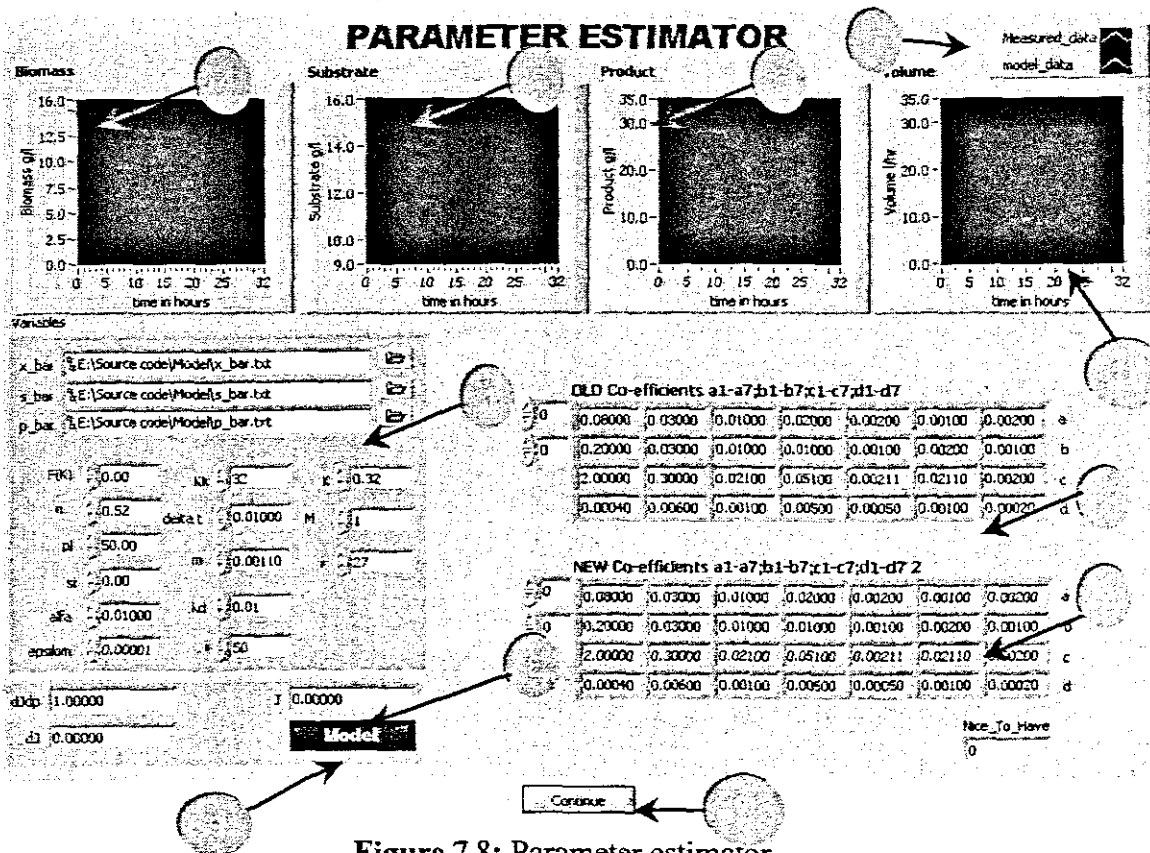


Figure 7.8: Parameter estimator

The main sections for the interaction with the operator are listed below:

- **Parameter estimator**

- 1) Cluster with input variables

The cluster is used to declare all the process variables that are necessary for the parameter estimator.

- 2) Old co-efficient of the yeast model

These are the initial co-efficients that the model uses to do the first calculations for the next part of the process time.

- 3) New co-efficients of the yeast model.

These are the new co-efficients that are recalculated using the parameter estimation algorithm.

- 4) Display for the concentration of the biomass yield, x.
- 5) Display for the concentration of the substrate, s.
- 6) Display for the concentration of the product, p.
- 7) Legend for the measured data and the output data from the model.
- 8) Indicators for the criterion values, J, dj, djdp.
- 9) Invoke the window for the yeast model user interface.
- 10) Click *Continue* to exit the parameter estimator window.
- 11) Display for the volume of the fermentor.

7.3.1 Yeast Model

The model's front panel is invoked from the parameters estimation sub window Fig 7.8. On clicking the model button the sub window is Fig 7.9 is pops up.

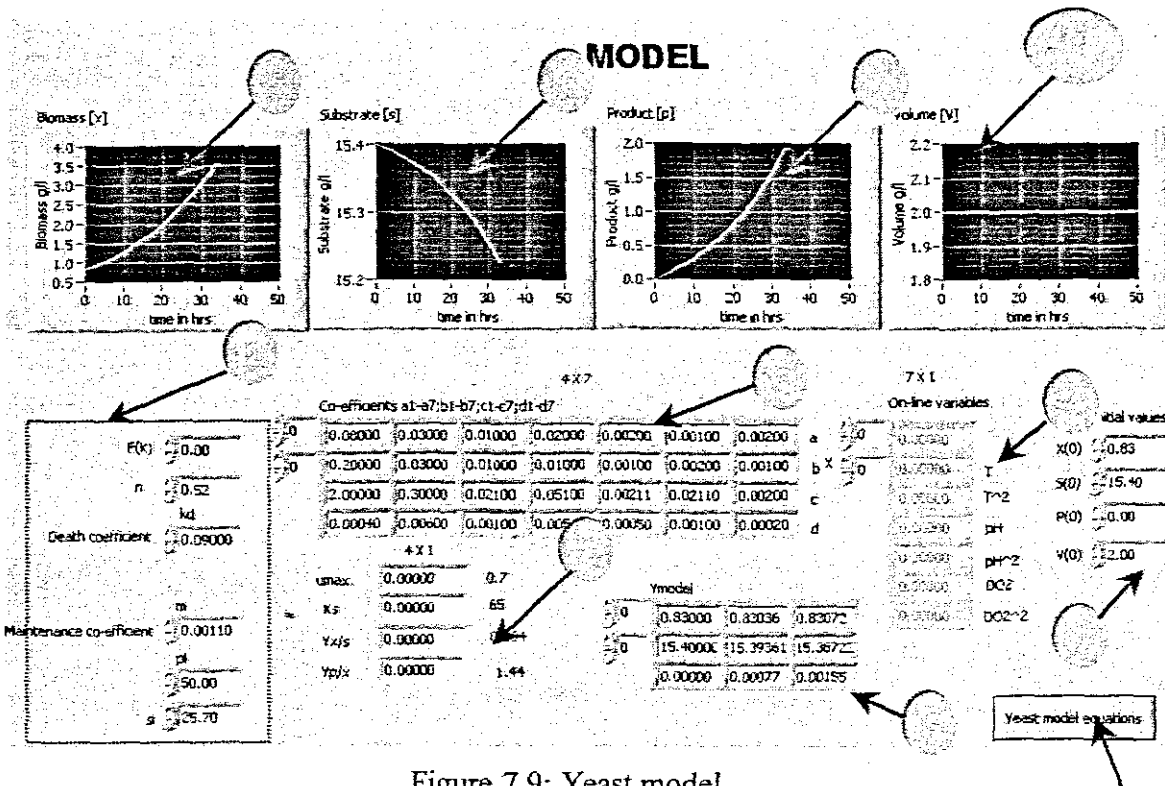


Figure 7.9: Yeast model

The main sections for the interaction with the operator are listed below:

□ Yeast model panel

- 1) Initial values for $x(0)$, $s(0)$, $p(0)$ and $V(0)$

2) Input variables to the model

The variables for the yeast model are declared using these controls.

3) Co-efficient of the yeast model.

a1 → a7, b1 → b7, c1 → c7, and d1 → d7. This matrix has the following dimensions $\text{par} \in \mathbb{R}^{4 \times 7}$.

4) On-line variables T, pH, F and DO₂.

5) Display for the concentration of the biomass, x.

6) Display for the concentration of the substrate, s.

7) Display for the concentration of the product, p.

8) Output kinetic parameters of the model μ_{\max} , K_s , $Y_{x/s}$ and $Y_{p/x}$.

9) The output of the model Y_{model} has dimensions $Y_{\text{bar}}^{[3 \times 1]}$. The elements of the matrix are x, s, p and V values from the model simulation.

10) Display of the volume of the bioreactor, V.

11) On clicking this button the yeast model equation are displayed on the pop up window Fig 7.10.

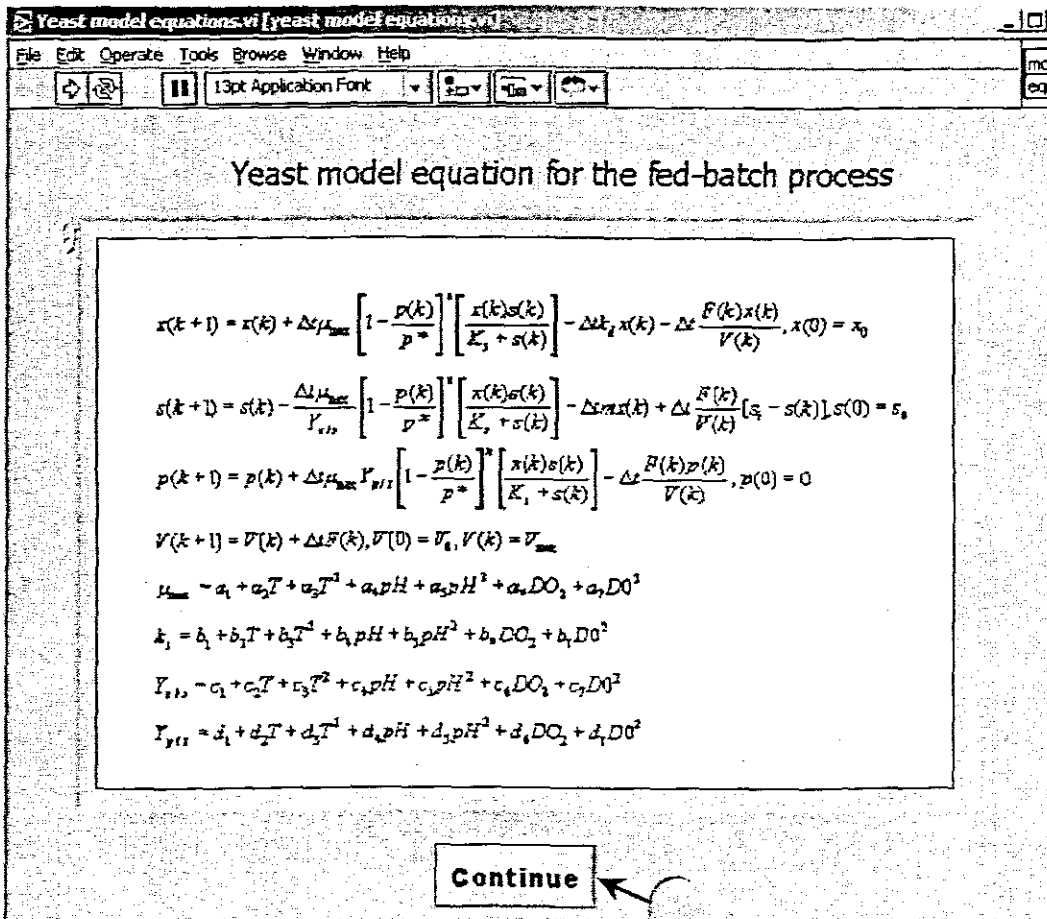


Figure 7.10: Yeast model equation pop up window

The main sections for the interaction with the operator are listed below:

□ Yeast model equation for the fed-batch fermentation process

1) [Click here to continue](#)

7.4 Optimal control

The developed application for optimal control calculated the optimal set points on the bases of the developed model and parameter estimation. The set points are determined from the optimal trajectories of pH, DO₂, T, and F they are then sent to the PID auto tuner. The developed optimization window is shown in Fig 7.11. This window is still under experimental tests.

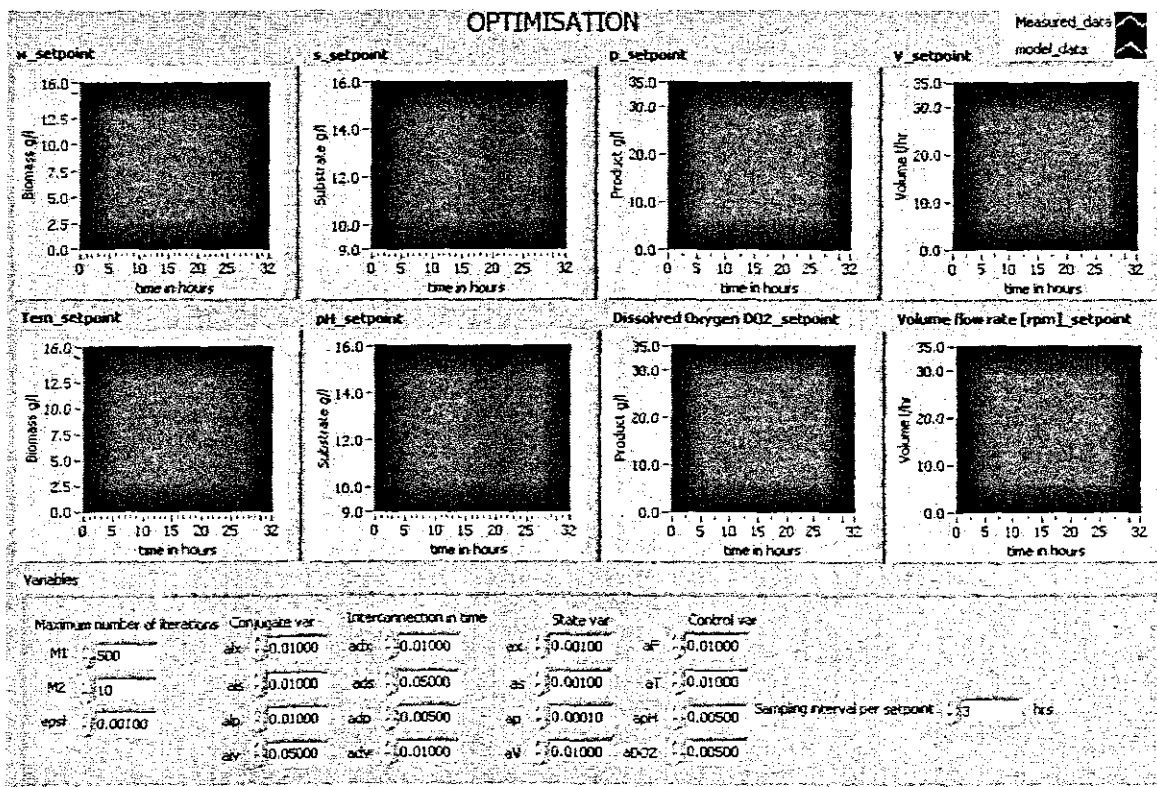


Figure 7.11: Optimization pop up window

7.5 PID Auto tuning

The developed application has an option that allows the operator to recalculate the controller parameters for a PID-Controller. The latter is available through the PID button as seen from the main program display Fig 7.1. The recalculated controller parameters can be compared with the previous used parameters. If the operator is satisfied with the recalculated parameters, they can then be entered manually into the bio-controller using the keypad user interface.

This operation cannot be done automatically by the PC as the used bio-controller model Biostat® C **DOES NOT** support sending the controller parameters from the host computer.

7.5.3 Auto tuner

The auto tuner is developed using LabVIEW™ PID auto tuner to recalculate the bio-controller's PID controller parameters. A pop up window appears after clicking on the button allowing the operator to *select, view, specify cycle time* and *tune* the selected process variables using the data received from the serial port in real time. See Fig 7.12.

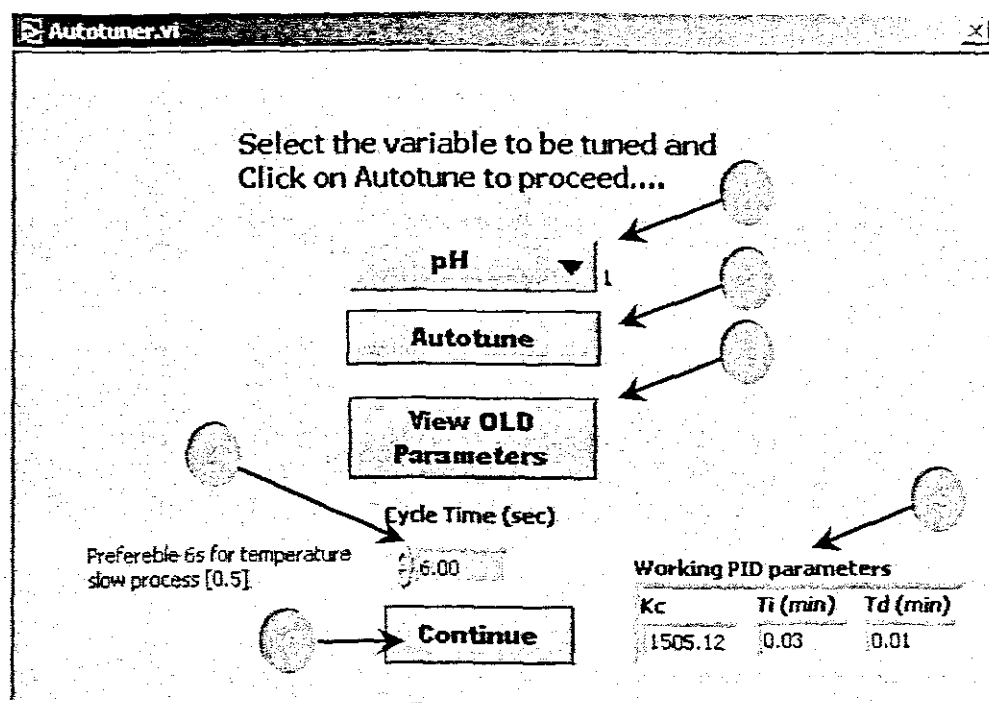


Figure 7.12: Auto tuning window

The main sections for the interaction with the operator are listed below:

□ **Auto tuner**

1) Process variable selector

Select the process variable that has to be tuned using this selector.

2) Auto tune

Click here to start the tuning and follow the steps as described in the windows that pop up as the next button is being pressed in respective steps.

3) Click *View OLD Parameters* to view old PID parameters.

4) Cycle time

The time for acquisition of data from the serial port before the start of calculation of the new PID parameters.

5) Indicator for old parameters

This indicator pops up when the *VIEW Old parameters* button have been pressed. The indicator stays on for 5 seconds and after that it disappears.

6) Click *Continue* to exit the auto tuner window

Chapter 8

Software description

8. Introduction

This chapter lists the file names of the developed source code. The source is code is developed on two different programming platforms that being, LabVIEW and Matlab/Simulink.

8.1 LabVIEW code

Fig. 8.1 gives the hierarchy of the developed subvi's and Table 8.1 list the file names of the developed vi. Note that some of the vi's come with LabVIEW.

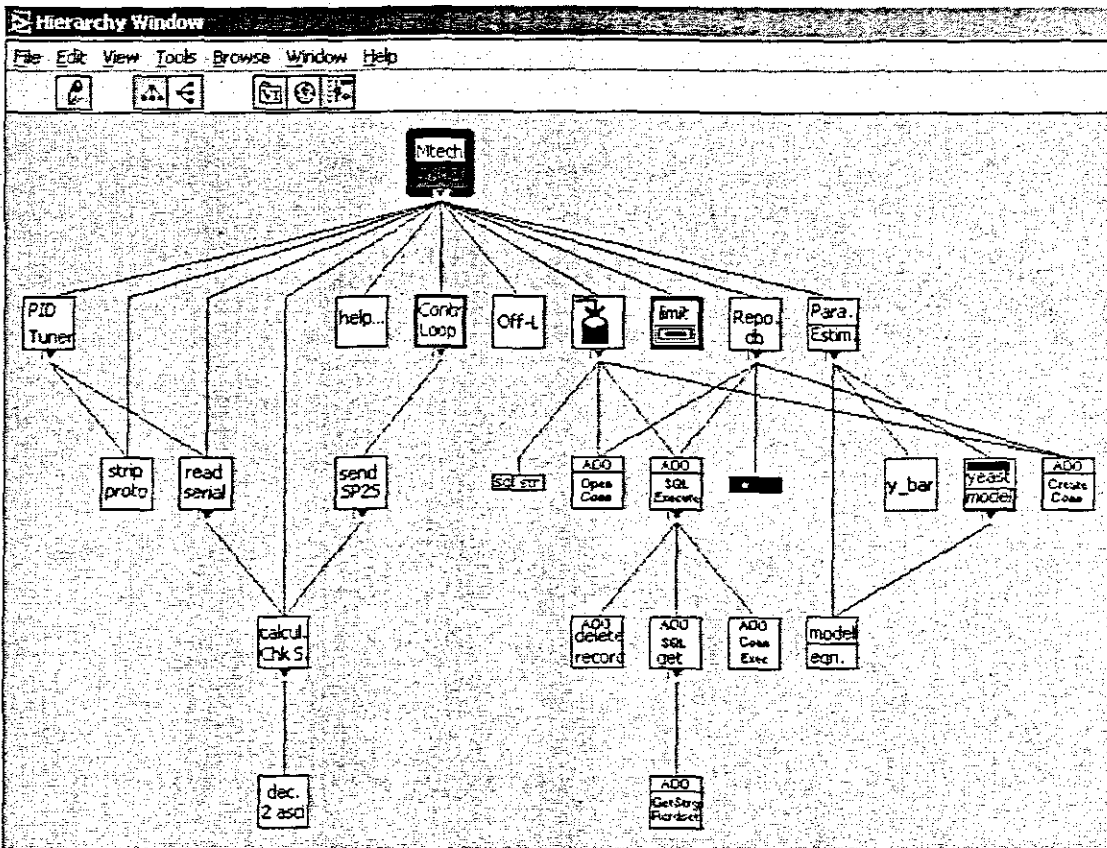


Figure 8.1 Hierarchy of the developed application

<p>Monitoring and Supervisory Control Program.vi</p> <p>Write Characters To File.vi</p> <p>port number.vi</p> <p>Serial Port Write.vi</p>	
--------------------------------------------------------------------------------------------------------------------------------------------------	--

CalculateChecksum.vi	
Serial Port Init.vi	
Statusofvariables.vi	
Control Loops Settings.vi	
Temp_Max.vi	
Temp_Min.vi	
pH_Max.vi	
pH_Min.vi	
Agitation_Max.vi	
Agitation_Min.vi	
DO2_Max.vi	
DO2_Min.vi	
insert_into_db.vi	
Reports.vi	
Agitation_Set.vi	
DO2_Set.vi	
pH_Set.vi	
Temp_Set.vi	
Strip_protocol.vi	
help.vi	
Acquire data from serial port.vi	
Autotuner.vi	
Off_Line_Variables.vi	
p_DO2_PV.vi	
pH_PV.vi	
Temp_PV.vi	
Agitation_PV.vi	
Parameter estimator_real data.vi	
General Error Handler.vi	
Close File+.vi	
Write File+ (string).vi	
Open/Create/Replace File.vi	
Open Serial Driver.vi	
serpConfig.vi	
dec2ascil.vi	

Serial Port Buffer Size.vi	
Serial Port Reset.vi	
Flowrate_Max.vi	
Flowrate_Min.vi	
Flowrate_Set.vi	
PVSetpoints.vi	
ADO Connection Open.vi	
ADO Connection Create.vi	
SQL String conform.vi	
SQL Execute.vi	
SQL String conform_NS.vi	
Serial Port Break.vi	
Bytes At Serial Port.vi	
Serial Port Read.vi	
Close Serial Driver.vi	
PID with autotuning.vi	
biological_optimal.vi	
Dot Product.vi	
A x B.vi	
K.vi	
delta.vi	
y_bar.vi	
yeast model equations.vi	
Find First Error.vi	
serpOpen.vi	
ADO Recordset Destroy.vi	
ADO Connection Execute.vi	
SQL Fetch Data (GetString).vi	
Relay Auto Tune.VI	
Simple Error Handler.vi	
Get Instrument State.vi	
Autotuning wizard.vi	
PID (gain schedule).vi	
Preload Instrument.vi	
Run Instrument.vi	

<p> GSim Unit Delay.vi GSim Initialize.vi GSim Manager.vi Read From Spreadsheet File.vi KK.vi ADO Recordset GetString.vi Mean.vi Relay Out.vi Get PID Parameters.vi Calculate PID.vi Open Panel.vi Close Panel.vi MeaNoise.vi State.gbl Get PID gains.vi Get dt.vi PV filter.vi Derivative action.vi Get error (sp-pv).vi Proportional action.vi Integral action.vi Auto/manual.vi Get PID Parameters (new).vi viRef buffer.vi GSim Synchronizer.vi GSim Global.vi GSim Manager Kernel.vi Read Lines From File.vi GSim Task.vi Read File+ (string).vi Open File+.vi </p>	
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

Table 8.1 List of subvi's used in the dissertation

8.2 Matlab / Simulink

The Matlab files are partitioned as follows:

Code realizing the model equations:

- quadratic_model / Simulink model file
- YM_var / Matlab file

Code realizing the parameter estimation algorithm:

- Model_variables / Matlab file
- Parameter_estimator / Matlab file
- quadratic_model / Matlab file

Code realizing the optimization algorithm:

- Optimize.m

See appendix A for the complete source code.

Chapter 9

Conclusions

Fed batch fermentation, due to an inherent flexibility of the methods provides a valuable tool in biotechnology, not only to study microorganisms, but also to commercially produce valuable components. However, many requisites and variables should be taken into consideration in its implementation. Moreover, the control of all the process is mandatory, which otherwise would render the mode of operation complexity useless. In addition to the complexity of the fermentation, the control mode and parameter to be controlled should also be analysed.

The control of a fed-batch fermentation can implicate many difficulties: low accuracy of on-line measurements of substrate concentrations, limited validity of the feed schedule under a variety of conditions and prediction of variations due to strain modification or change in the quality of the nutrient medium. These aspects point to the need of a fed-batch fermentation control strategy, which is model independent, identifies the optimal state on-line, incorporate a negative feedback control in the nutrients feeding system and contemplates saturation kinetic model, variable yield model, variations in feed substrate concentration and product inhibited fermentation. [Agrawal et al, 1994]

An adaptive control strategy for optimal control of the yeast production is proposed and developed in the dissertation.

9.1 Problems solved in the dissertation

The problems in the dissertation to develop methods, algorithms and programmes for building of a two-layer system for real time optimal control of the Biostat® C pilot plant with the following subsystems:

- Data acquisition and real time implementation,
- Modelling and simulation,
- Model parameter estimation,
- Process optimisation,
- PID controller parameter tuning,

LabVIEW™ is used as driver software and serial communication protocol is used between the host computer and the Biostat® C.

9.1.1 Data acquisition and real time implementation

The developed real time control system consists of two main parts:

- Hardware part.
PC with standard RS-232 serial interface connected to the Biostat® C bio-controller through the RS-232 / RS422 Converter. The integration of the whole system is achieved.
- Software part consisting of the following sub-programes, programs for:
 - Data acquisition through serial communication and dialog between the computer and bio-controller. Acquiring of data for the physiochemical variables, control loops parameters, set points and initial condition is achieved through commands from the bio-controller protocol and special functions of the driving software.
 - Programs for creating and saving data to the database using Microsoft access.
 - Programs for connections of the programs solving the problems for simulation, parameter estimation, optimal control and PID parameter setting according to the two-layer control strategy.

The developed software system for data acquisition and real time control implementation is based on LabVIEW. LabVIEW is a widely used technology for building industrial control systems. The software connects in a simple way all parts of the adaptive control approach in one whole. The system is user friendly with help facility to allow easy of operation for the operator. The communication between the computer and the bio-controller is quick and simple. It allows exchange of information for all process variables and change of the bio controller parameters using the computer.

9.1.2 Modeling and simulation

- The fed-batch fermentation process for production of yeast is described as an object of control with input, state space, output and disturbance variables.

- Model of the fed-batch process based on incorporation of the physiochemical variables into the mass balance equation is developed.
- The programs in Matlab / Simulink and LabVIEW are developed to simulate the model. Simulation can be done with real data from the process, or with selected input data (Deshpande and Chen, 1993).

The developed model is semi-theoretical in the sense that mass balance equations are theoretical mathematical model, but coefficient cannot be calculated mathematically. These coefficients are introduced analytically to present the kinetic parameters and in this way to represent the influence of the physiochemical variables over the biological variables. The latter is achieved by representing the kinetic parameters of biological model through quadratic functions of physiological variables. Parameters estimation is used to determine the coefficients of these functions. The utilization of Matlab/Simulink and LabVIEW packages made the development much easier when it comes to testing of the prototype. The results received on both packages were the same, proving the operation of the algorithm and also the consistence was tested. The results from simulation are used later for model parameter estimation and process optimisation.

9.1.3 Model parameter estimation

- The problem for parameter estimation is formulated.
- Method to solve the problem for parameter estimation is developed on the basis numerical procedures.
- Algorithm and programs in Matlab and LabVIEW are developed and implemented as a part of the common control strategy.

The problem for parameter estimation is solved by gradient method. The solution is simple but it uses a very big number of parameters (28). The recursive least squares techniques for parameter estimation is selected, as it is suitable for operation in a noisy practical environment, also it has minimum storage requirements and does not need computational time for matrix inversion.

The use of Matlab and LabVIEW simulation and realisation software packages made it easy to implement the algorithm in real time. The comparison is made on the results received from the two packages. The comparison proves that the implementation of the algorithm is operational by receiving same results on both simulation tools.

9.1.4 Process optimization

- The problem for maximum production of the biomass is formulated.
- Method and algorithm to solve the optimal control problem is developed on the basis of augmented functional of Lagrange and decomposition in time domain.
- Program in Matlab is developed to solve the problem for optimal control is developed.

The developed two-layer control structure allows adaptive control through the process of repetitive model parameter estimation and process optimisation. The decomposition method used to solve the problem for optimisation (optimal control) allows the overall problem to be decomposed and simplified (Morari *et al*, 1980). The complexity of calculation is reduced by using the optimal trajectories of the physiochemical variables, as changing the set point of the local PID controllers allows the time varying nature of the process to be counted out.

9.1.5 PID controller parameter tuning

- Program for representing the optimal trajectories of the physiochemical variables as a sequence of constant for some period of time set points is developed.
- The auto tuning PID program from LabVIEW is incorporated in the two-layer control strategy.

The fact that PID controllers are popular and well used controllers industrially does not mean that they control the plant at their optimal operation (Dorf, 1998). The main draw back in PID controllers is the fact that the set points are set once during the duration of the process, and are not changed during the variation in the process, even though the major parameter variables that affect the process do change. The latter result in the former. The methods, algorithms and programs developed in the dissertation solve the problem highlighted above using the adaptive control technique.

9.2 Benefits of the two-layer control strategy

Benefits of process optimization and adaptive control come from improved plant and business performance. In general terms, the revenues come from improved yields of valuable products, reduced energy consumption, and higher processing rates either

through increased capacity of existing equipment or longer stream factors. Optimization may also influence reduction in a number of other operating costs including maintenance, equipment wear and staff utilization.

The engineering benefits come from improved process troubleshooting and assistance in making quicker and more accurate decisions. This ultimately relates to improved process operations. There is also an interaction with process design of new units. The knowledge that comes from computer optimization can have a bearing on equipment sizing and upon assumed capacity factors.

9.3 Future developments of methods and applicability

The future developments can be decomposed in the following way:

- Testing of the two-layer control system on the working plant.
- Testing of the optimisation program for different experiments.
- Testing of the developed application on the bigger scale fermentor.
- Connection of estimation and optimal control problem solution in order to realise different operator strategies
- Application of the developed methods and programmes for different fed-batch processes.

9.4 Application of developed strategy, methods, algorithm and programs

- Waster water purification plants.
- Beer industry.
- Food industry.
- Mining industry for extraction of metal ions.
- Sugar and pharmaceutical industry.
- Control purposes in education.
- As an application plant for different control strategies (Dube, 2002).

9.5 Publications in connection with the dissertation

- Mkondweni, N.S., R.Tzoneva. Program for monitoring and control of a fed-batch fermentation process for the production of yeast. *Proc. of the International Conference AFRICON'02, 2002.*

- Mkondweni, N.S., R.Tzoneva. Program for control of ion exchange wastewater purification pilot plant using LabVIEW™. Proc. of the 3rd International Conference on Control Theory and Application (ICCTA'01), Pretoria South Africa, December 12-14, CDROM, ISBN: 981-04-4794-9, 2001, p114-128.

- Mkondweni, N.S., Susan T.L. Harrison and R. Tzoneva. DAQ system for monitoring and data logging for fed-batch fermentation process for propagation of yeast. Cape Biotech 2002 Programme, PP23

- Mkondweni N.S. and Tzoneva R, Development of a mathematical model of the fed-batch fermentation process for the production of yeast. Journal of engineering. Design and Technology, 2003. (Submitted)

- Mkondweni N.S. and Tzoneva R, Integrating LabVIEW capabilities for monitoring and supervisory control with a B-Braun Biotech unit functions for fermentation processing and direct control. IJEE journal, 2003. (Accepted).

References:

AGRAWAL P., KOSHY G., RAMSEIER M., 1989: An algorithm for operating a fed-batch fermentor at optimum specific growth rate. Biotechnology and Bioengineering, vol. 33, pp.115-125

AHMED S. FAWZY, OLIVER R. HINTON, 1984: Optimal and hierarchical control of fermentation processes using a microcomputer: A survey, IEEE.

ANDERSON M., BRABRAND H., JORGENSEN S., 1991: Model based control of continuous yeast fermentation. Proc. Of the ACC, 01, Boston, USA, pp.1329-1334

ANDRES-TOO B. DE, GIRON-SIERRA J.M., LOPEZ-OROZCO J.A., FERNANDEZ-CONDE C, PEINADO J.M., GARCIA-OCHOA F., 1998: A kinetic model for beer production under industrial operational conditions.

B&B ELECTRONICS, 1997: RS-422 and RS-485 Application Note. www.bb-elec.com

B. BRAUN BIOTECH INTERNATIONAL, 1996: Installation handbook 1.1 MFCS / win.

BAJPAI R., LUECKE R., 1990: Controller design for biochemical reactor, Chemical Engineering Problems on Biotechnology, SHULER H.L. Eds, American Institute of Chemical Engineering, NY, vol. 1, pp.301-332

BASTIN G., 1991: Non-linear and Adaptive Control in Biotechnology: A tutorial, European Control Conference (ECC-91)

BASTIN G., DOCHAN D., 1990: On-line estimation and adaptive control of bioreactors. Elsevier science Publishing Co. Oxford, UK

BELGARDT K., JUAN J., 1991: Optimisation of yeast production – A case study biotechnology, vol. 4, pp.383-406

BIROL G., DORUKER P., KIRDAR B., ONSON, Z., VLGEN K., 1998: Mathematical description of ethanol fermentation by immobilised *Saccharomyces cerevisiae*, Process Biochemistry, vol. 33, No. 7, pp.763-771

BridgeVIEW™ and LabVIEW™ NATIONAL INSTRUMENTS, 1998: PID Control Toolkit for G Reference Manual

CHEN L., BASTIN G., BREUSEGEM V., 1992: A case study of adaptive non-linear regulation of fed-batch biological reactors. Technical report, 92.37. Catholike universitat, Louvam, Belgium

CHERUY A., DURAND A., 1979: Optimisation of erythromycin biosynthesis by controlling pH and Temperature: Theoretical Aspects and Practical Application.

CLAES J., VAN IMPE J., 1998 (a): On-line monitoring and optimal adaptive control of the fed-batch baker's yeast fermentation. Preprints of the 7th International Conference on Computer Application in Biotechnology – CAB VII IT. Yoshida, S. SHIOYA, Eds, pp.405-410, Elsevier science Publishing Co., Osana, Japan

CLAES J., VAN IMPE J., 1998b: Sensitivity analysis of a linearising controller for the fed-batch baker's yeast fermentation. Technical report Biotech. Department of food and microbial technology, Katholiene universiteit Leuven, Belgium

CLAES J., VAN IMPE J., 1999: Sensitivity analysis of a linearising controller for the fed-batch baker's yeast fermentation. Proc. Of the 14th world congress of IFAC, vol. 0, pp.439-444.

DESHPANDE, P.B. & CHEN, L.C. 1993: Real Time Simulation and Advanced Process Control, Present Status and Future Trends. Proceedings of the 19th Annual Advanced Control Conference. Indiana: W.Lafayette.

DOCHAN D., BASTIN G., 1985: Stable adaptive algorithms for estimation and control of fermentation processes, IFAC Modelling and Control of Biotechnological Processes

DOCHAN D., BASTIN G., 1990: Adaptive Control Of Fed-batch Bioreactor, Chemical Engineering Comm., vol. 87, pp.67-85, Gordon and Breach Science Publishers S.A.

DORF, R.C. 1998: Modern Control Systems, 8th Edition, (BISHOP, R.H.). Addison Wesley FINDEISEN, W. 1980: Control and Coordination in Hierarchical Systems. Chichester, West Sussex: Wiley.

DUBE, N.M., 2002: Modelling and optimal control of counter current ion exchange process.

ECHEGARAY O.F., CARVALHO J.M., FERNANDES A.N.R., SATO S., AQUARONE E., VITOLO M., 2000: Fed-batch culture of *Saccharomyces cerevisiae* in sugar-cane blackstrap molasses: invertase activity of intact cells in ethanol fermentation

FISHER, W.; DOHERTY, M. & DOUGLAS J. 1984: An Evolution and Hierarchical Procedure for Optimization of Preliminary Process Design. Presented at an Annual Meeting of AIChE, Paper 104c. San Francisco.

HASKINS, D. 1983: Restraints on Entire Plant Optimization. Chemical Engineering Progress, vol. 79, No. 6.

FLAUS J.M., PONS M.N., CHERUY A., and ENGASSER J.M., 1989: Adaptive algorithm for estimation and control of fed-batch bioprocess.

FREDE LEI, MORTEN ROBOLL, and STEN BAY JORGENSEN, 2001: A bio-chemically structured model for *Saccharomyces cerevisiae*.

GARCIA A., ACEBES L.F., and C. de PRADA, 2002: Modelling and simulation of batch processes: A case study. Proc of the 15th Triennial world congress of the international federation of automatic control, Barcelona, IFAC.

GARY A. MONTAGUE, A. JUAN MORRIS, AND JOHN R. BUSH, 1988: Considerations in control scheme development for fermentation process control, IEEE.

GOU S., JUAN J., BELLGARDT., 1995: Profit optimisation for baker's yeast continuous fermentation, 6th International Conference on Computer Application in Biotechnology, Gamisch Partenkirchen, Germany

GULNUR BIROL, PEMRA DORUKER, BETUL KIRDAR, Z. ILSSEN ONSAN* AND KUTLU ULGEN, 1998: Mathematical description of ethanol fermentation by immobilised *Saccharomyces cerevisiae*.

GYEONGBEOM YI, YOUNG BO HWANG, HO NAM CHANG and KWANG SOON LEE, 1989: Computer Control of Cell Mass Concentration in Continuous Culture, IFAC, pp. 243-249,

HIGGIN, J. 1984. Process Control No Longer Separate from Simulation Design. Chemical Engineering News, vol. 62, No. 14.

HORIUCHI J., KAMASAWA, M., MIYANAWA H., KISHIMORO M., MOMOSE H., 1993: J. fermentation bioengineering, vol. 76, pp.207-211.

<http://www.aten.com.tw/> RS232/RS485 Converter, Users Manual

<http://www.bbraunbiotech.com/>

HWANG Y., DOCHAN D., BASTIN G., 1995: Performance of inhibition functions with total inhibition concentration. Preparation of the 6th International Conference on Computer Application in Biotechnology, Gamisch Partenkirchen, Germany, pp.259-262.

IGNOVA M., PAUL G.C., KENT C.A., THOMAS C.R., MANTAGUE G.A., GLASSEY J., and WARD A.C., 2002: Hybrid modelling for on-line penicillin fermentation optimisation. Proc of the 15th Triennial world congress of the international federation of automatic control, Barcelona, IFAC, 21-26 July 2002

ILSE SMETS, KRISTEL BERNAERTS, JUN SUN, KATHLEEN MARCHAL JOS VANDERLEYDEN and JAN VAN IMPE, 2002: Sensitivity function based model reduction. Proc of the 15th Triennial world congress of the international federation of automatic control, Barcelona, IFAC, 21-26 July 2002.

INSTRUMENT DATA COMMUNICATIONS, 1993: Data Communications for instrumentation and control

ISIDORI, A. 1995: Nonlinear Control Systems. London: Springer.

JACKSON A.T., Process Engineering in Biotechnology.

JENS E. HAAG, ALAIN VANDE WOUWER, and MARCEL REMY, 2002: S systematic model reduction procedure for complex bioprocesses described by metabolic pathway networks. Proc of the 15th Triennial world congress of the international federation of automatic control, Barcelona, IFAC, 21-26 July 2002.

JIN S. K. Ye, SIMIZU K., CHEN J., 1994: Tech. Biotech, vol. 61, pp.273-281.

JOHANSSON, R. 1993: System Modelling and Identification. New York: Prentice Hall

JOHNSON A, 1985: The control of fermentation processes, IFAC Modelling and Control of Biotechnological Processes.

JOHNSON A., 1993: The theory of LQ optimisation of fermentation. Proc. Of the XI1th congress, vol. 9, pp.115-119

JOHNSON, C.D. 1993: Process Control Instrumentation Technology, 4th Edition. New Jersey: Prentice Hall Inc.

JONES, L.D. 1991: Electronic Instruments and Measurements, 2nd Edition. New Jersey: Prentice Hall Inc.

JORGENSEN S., JENSEN N., 1989: Dynamics and control of chemical reactor-selectivity surveyed. Proc. Of DYC'D' 89, Maasticht, Holland

JUNKINS, J.L. 1991: An Introduction of Optimal Estimation of Dynamic Systems. Alphen Aan Den Rijn: SIJHOFF and NOORDHOFF INTERNATIONAL.

KARIM M., RIVERA S., 1992: Adv. Biochemical. Engineering / Biotech. vol. 76, pp.1-10.

KEULERS M., ARIAANS L., GIUSEPPIN M. AND SOETERBOEK R., Specific growth rate control in fed-batch baker's yeast fermentation,

KEULERS M., ARIAANS L., GUISEPPIN M., SOETERBOEK R., 1993: Specfic growth rate control in fed-batch baker's yeast fermentation. Proc of the X11 IFAC world congress, vol. 7, pp. 441-444

KLEMAN G., CHALMERS J., LULI G., STROHL W., 1991: A predictive and feedback control algorithm maintains a constant glucose concentration in fed-batch fermentations. Applied and Environment Microbiology, vol. 57, pp.910-917

KONSTANTINON K., YOSHIDA T., 1989: Biotech Bioengineering, vol. 33, pp.1145-1150.

KOZIETULSKI, M. 1981: Two-Layer Implementation of Repetitive Dynamic Optimization (Static-Optimally Aided Control). Archivum Automatiki I Telemeckaniki, vol. 26, No. 4, pp. 511 - 520.

LEFKOWITZ, 1966: Multilevel Approach to Control System Design. Transcript of ACME, Journal of BE, vol. 6.

LILIANA KRZYSZEK & STANISLAW LEDAKOWICZ, 1997: Yield and Maintenance Coefficients in *S. Cerevisiae* Cultures.

LIPTAK, B.G. 1999: Instrument Engineer's Handbook. Process Control, 3rd Edition. CRC Press LLC.

LUYBEN, W.L. 1990: Process Modelling, Simulation and Control for Chemical Engineers, 2nd Edition. McGraw Hill Publishing Co.

MC NEIL B., HARVEY L., 1990: Fermentation, a practical approach, IRL Press, Tokyo

MESAROVIC, M.D. 1970: Theory of Hierarchical, Multilevel Systems. New York: Academic Press.

MESZAROS A., BALES V., 1992: A contribution to optimal control of fed-batch biochemical processes. Bioprocess Engineering, vol. 7, pp.363-367

MICHAEL L. SHULER, FIKRET KARGI, 1992: Bioprocess Engineering Basic Concepts

MONTGOMERY P.A., WILLIAMS D., SWANICK B.H., Control of a fermentation process by an on-line adaptive technique

MORARI, M. & STEPHANOPOULOS, G. 1980: Studies in the Synthesis of Control Structures for Chemical Process. Part II: Structural Aspect and the Synthesis of Alternative Feasible Control Schemes. AIChE, (American Institute for Chemical Engineers) vol. 26, no. 2, pp. 232 – 247.

MORARI, M. & STEPHANOPOULOS, G. 1980: Studies in the Synthesis of Control Structures for Chemical Process. Part III: Optimal Selection of Secondary Measurements within the Framework of State Estimation in the Presence of Persistent Unknown Disturbances. AIChE, (American Institute for Chemical Engineers) vol. 26, no. 2, pp. 247 – 260.

MORARI, M.; ARKUN, J. & STEPHANOPOULOS, G. 1980: Studies in the Synthesis of Control Structures for Chemical Process. Part I: Formulation of the Problem. Process Decomposition and the Classification of the Control Tasks. Analysis of the Optimizing Control Structures. AIChE, (American Institute for Chemical Engineers) vol. 26, no. 2, pp. 220 – 232.

MOSER A., 1985: Kinetics of batch fermentation in biotechnology, Ed . H. REHM and REED G. VCH verlagsgesellschaft mgh, Weinheim, pp.273-283

MUBECCEL ERGUN, S. FERDA MUTLU, 1999: Application of a statistical technique to the production of ethanol from sugar beet molasses by *Saccharomyces cerevisiae*.

NAKAMURA T., KURATANI T., MORITA Y., 1985: Proc of the IFAC Modelling and control of biotechnology. Processes, Noordwijkerhout, pp.231-235

NATIONAL INSTRUMENTS, 1998: BridgeVIEW™ and LabVIEW™ Control and Simulation Software for G Reference Manual

NEWAY J., 1989: Fermentation process development of industrial organism, Narcd Dekker, NY

NJODZI Z., 2001: Studies on the fed-batch propagation of brewer's yeast in high gravity wort.

NORTON, J.P. 1986: An Introduction to Identification. London: Academic Press, Harcourt Brace Javanovic Publishers.

OGATA K., 1990: MODERN CONTROL ENGINEERING SECOND EDITION, Prentice-Hall, Inc., Englewood Cliffs, New Jersey

PARK Y., SHI Z., SHIBA S., CHANRAL C., RIJIMA S., KOBAYASHI T., 1993: Application, Microbial. Biotechnology. vol. 38, pp.649-655

PARTNAIK P., 1995: A heuristic approach to fed-batch optimisation of streptokinase fermentation. Bioprocess Engineering, vol. 13, pp.109-112

PASCAL SIEGWART, JOHANNE COTE, KEITH MALE, JOHN H.T. LUONG, MICHEL PERRIER, and AMINE KAMEN, 1999: Adaptive Control at Low Glucose Concentration of HEK-293 Cell Serum-Free Cultures

PAULINE M. DORAN, 1995: Bioprocess Engineering Principles Chapters 4,6 & 11.
POMERLEAN Y., VIEL G., 1992: Industrial application of adaptive non-linear control for baker's yeast production. Proc of IFAC symp. on modelling and control of biotech, processes, keystones, USA, pp.315-318

POMERLEAU Y., PERRIER M., BOURQUE D., 1995: Dynamics and control of the fed-batch production of poly - β - Hydroxybutyrate by Methyl bacterium extorquens

PONS M.N, BORDET J. AND ENGASSER J.M., 1982: A micro-mini computer hierarchical control system for a laboratory scale fermentor.

POPCHEV, T. & TSONEVA, R. 1990: Synthesis of Two-Layer Control of Interconnected Systems with Time Delays in the Presence of Slowly Varying Disturbance. Proceedings for the, Tallinn: XI World Congress IFAC, vol. 2, pp. 25 - 31.

POULISSE H. N. J. and C. VAN HELDEN, 1985: Adaptive LQ Control of fermentation processes, IFAC Modelling and Control of Biotechnological Processes

REULERS M., ARIAANS L., GIUSEPPIN M., SOETERBOEN R., 1993: Specific growth rate control in fed-batch baker's yeast fermentation. Proc. Of the X11th congress of IFAC, vol. 4, pp.441-444

ROBERTS, P.D. & LIN, J. 1991: Potential for Hierarchical Optimizing Control in the Process Industry. Proceedings of the 1st European Control Conference, Grenoble, France, vol. 1, pp. 213 – 217.

ROEL A., 1983: Energetics and kinetics in biotechnology, Elsevier, Biomedical Press, Amsterdam

ROSIMEIRE APARECIDA JERONIMO, CLAUDIO GARCIA, and OSCAR A. Z. SOTOMAYOR, 2002: Fuzzy identification of bioprocesses applying TSK-TYPE models with consequent parameter estimation through orthogonal estimator. Proc of the 15th Triennial world congress of the international federation of automatic control, Barcelona, IFAC.

ROUBOS J. A., DE GOOIJER C. D., VAN STRATEN G., VAN BOXTEL A.J.N., 1997: Comparison of optimisation methods for fed-batch cultures of hybridoma cells, Bioprocess Engineering.

ROUBOS J., KRABBen P. SETNES N., BABUSHA K., HEIJNEN J., VERBRUGGEN H., 2000: Hybrid model development for fed-batch bioprocesses; combining physical equations with the metabolic network and black box kinetics,

SA INSTRUMENTATION & CONTROL JOURNAL, 1999.

SALZMANN C.H., D. GILLET, AND P. HUGUENIN, 1999: Introduction to real-time control using LabVIEWTM with an application to distance learning.

SANTOS VICTOR J.A., 1991: Adaptive inferential predictive control for fermentation bioreactors

SCHUGERL K., 1985: Modelling of biotechnological processes, IFAC Modelling and Control of Biotechnological Processes.

SEIDLER W., 1996: Description of serial interface between host and control units of the B. Braun Biotech International GMBH

SEIKANENG D.L., 2000: Modelling and Optimal control of a propionibacteria fermentation process for maximum production of Vitamin B12.

SHI Z., SHIMIZU K., 1992: J. Ferm. Bioeng. Vol. 74, pp.39-75

SHIMIZU K. K. Ye, 1995: Development of intelligent control systems for bioreactors. Preparation of the 6th International Conference on Computer Applications in Biotechnology, Garmish patennirchen, Germany, pp.89-94

SHIMIZU KAZUYU And KAIMING YE, 1995: Development of intelligent control systems for bioreactors, *Computer Applications in Biotechnology* May 14 – 17.

SHIMIZU N., MIURA K., SHIOYA S., SUGA K., 1994: Proc of PSE'94, Kyongju, Korea

SHIOYA S., SHIMIZU H., OGATA M. And TAKAMATSU T., 1985: Simulation and experimental studies of the profile control of the specific growth rate in a fed-batch culture, *IFAC Modelling and Control of Biotechnological Processes*.

SING. M. A. TITLI., 1978: *Systems: Modelling, Optimization and Control*. New York, Pergamon Press.

SKOGESTAD, S.; HALVORSEN, I; LARSSON, T & GOVATSMARK, M. 1999: Plant-Wide Control: The Search for the self-optimizing Control Structure. *Proceedings of the 14th IFAC World Congress, China*, pp. 325 – 330.

SMITH J. E, 1985: *Biotechnology principles*.

STANDBURY P., WHITAKER A., HALL S., 1993: *Principles of fermentation technology*, Pergamon

STEVE ORTH, 1999: *Effective software architecture for control* National Instruments.

TAMURA, H. 1975. Decentralized Optimization for Distributed Lag Models of Discrete Systems. *Automatica*, vol. 11, pp. 593 – 602.

TARTAKOUSKY B., VLITZUR S., SHEINTUCH M., 1995: Optimal control of fed-batch fermentation with autoinoluction of metabolite production. *Biotechnology Prog.* vol. 11, pp.80-87.

TEO, K.L.; GOH, C.J. & WONG, K.H. 1991: *A Unified Computational Approach to Optimal Control Problems*. London: Longman.

TURNER C., GREGORY M., THNNHILL N., 1994: Closed loop control of fed-batch cultures of recombinant *Escherichia coli* using on-line HPLC. *Biotechnology and Bioengineering*, vol. 44, pp.819-829.

UNBAHANEN, H & RAO, G. 1997: Continuous Time Approach to System Identification: A Tutorial in Preprints of 11th IFAC Symposium on System Identification, vol. 3, pp. 1023 – 1049.

UNIVERSITY OF MARYLAND BALTIMORE COUNTY, Review on Fed-batch fermentations: Mathematical Modelling, Parameters and Control.

VAN IMPE J. F. and BASTIN G., 1995 (b): Optimal adaptive control of fed-batch fermentation processes. *Cont. Eng. Pract.* vol. 3, No. 7, pp.939-954

VAN IMPE J. F. and BASTIN G., 1995a: Optimal feed rate profiles for bioprocess optimisation and model identification.

VANCE J. VANDOREN, 1996: Examining the fundamentals of PID control, Control engineering

WILLIAMS D., C. Eng, JOUSEFPOUR M., SWANICK B., 1984: On-line adaptive control of a fermentation process. IEE Proc. vol. 131, Pt. D, No.4

WILLIAMS D., YOUSEFPOUR P., WELLINGTON E., 1986: On-line adaptive control of a fed-batch fermentation of *Saccharomyces cerevisiae*. Biotechnology and Bioengineering, vol. 18, pp.631-645

WITBOOIN, 2001, Data Acquisition system for monitoring a fermentation process

YU, W. & LIU, D. 1999: Time Variant Parameter estimation of a Nonlinear System using a Quasi-Newton Method. Proceedings of 14th world Congress of IFAC, China, vol. H, pp. 55 – 60

Appendix A

Monitoring, Control of Setpoint and Data Logger
E:\Source code\Main Source\DataAcquisitionReal.vi
Last modified on 11 Feb 2003 at 10:39 PM
Printed on 18 Feb 2003 at 11:29 AM

MENU Active..



HELP

Monitoring and supervisory control of Process variable [PV] and controller setpoint [CS]

<p>Temperature in oC</p> <p>Temp_Min: 0.00</p> <p>Temp_Max: 0.00</p> <p>Temp_Set: 0.00</p> <p>0.00 40.00</p> <p>0.0</p>	<p>29.0</p> <p>20.0</p> <p>10.0</p> <p>0.0</p> <p>0 43</p>	<p>Agitation in rpm</p> <p>Agit_Max: 0</p> <p>Agit_Set: 0</p> <p>Agit_Min: 0</p> <p>0</p> <p>0 250 350 450 500 550 650</p>	<p>900.0</p> <p>600.0</p> <p>400.0</p> <p>200.0</p> <p>0.0</p> <p>0 43</p>
<p>pH in pH</p> <p>pH_Min: 0.00</p> <p>pH_Max: 0.00</p> <p>pH_Set: 0.00</p> <p>0.00 6.00</p> <p>0.00</p>	<p>8.0</p> <p>6.0</p> <p>4.0</p> <p>2.0</p> <p>0.0</p> <p>0 42</p>	<p>Dissolved Oxygen in %</p> <p>DO2_Min: 0.00</p> <p>DO2_Max: 0.00</p> <p>DO2_Set: 0.00</p> <p>0.00 50.00</p> <p>0.0</p>	<p>32.0</p> <p>20.0</p> <p>10.0</p> <p>0.0</p> <p>0 43</p>

COM Port selector: COM 1

Sampling rate in min: 2.00

No of samples for model calculation: 50

Number of samples stored for model: 0.00

Time left before next upload..

Progress bar

Process PV to start reading

Process Variable

Upper Limit

Lower Limit

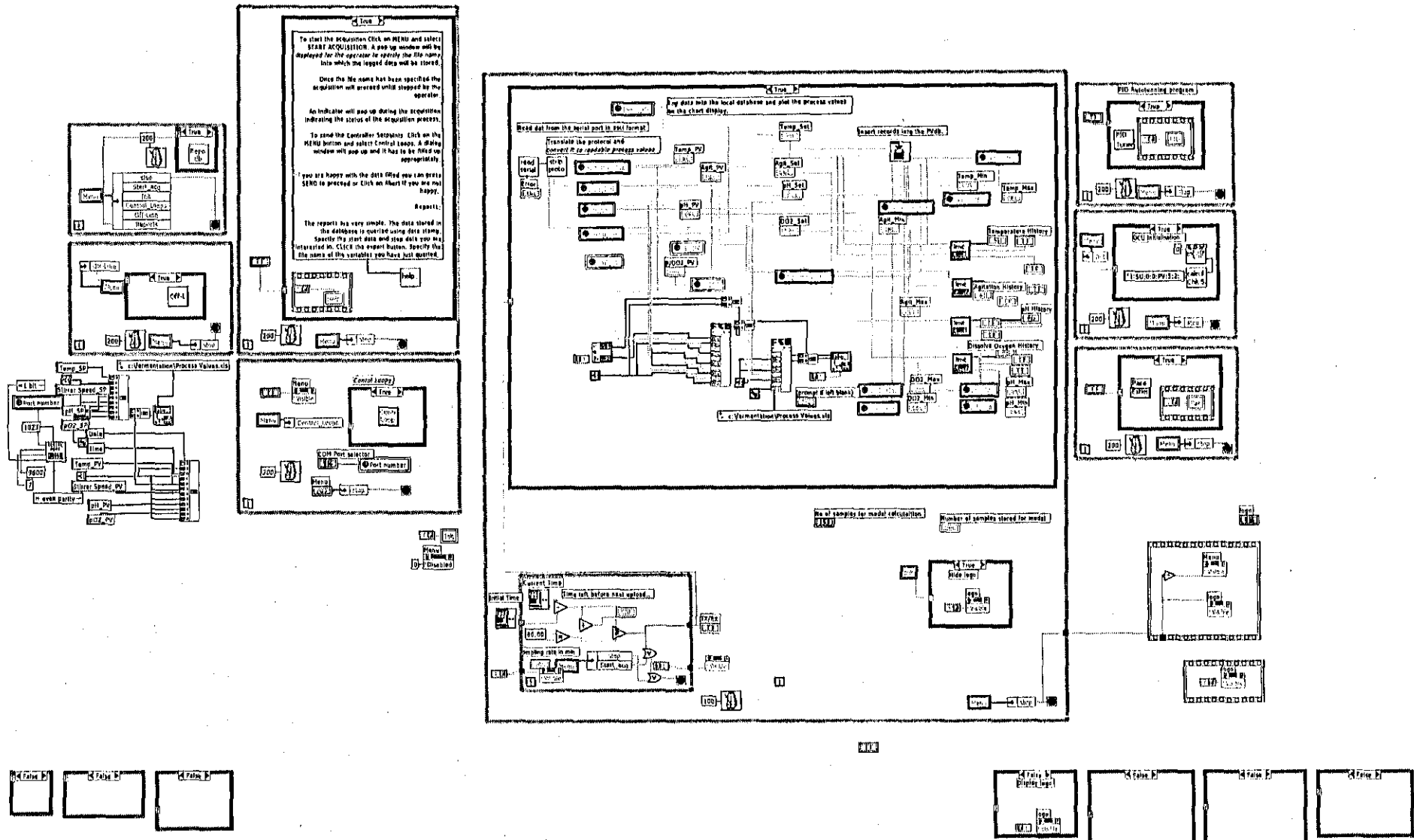
Setpoint

Parameter Estimation

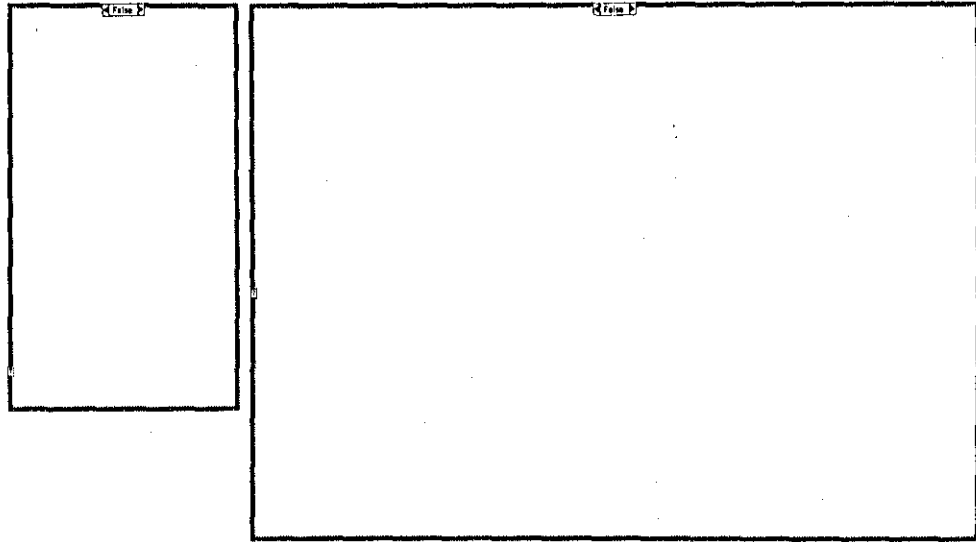
Optimisation

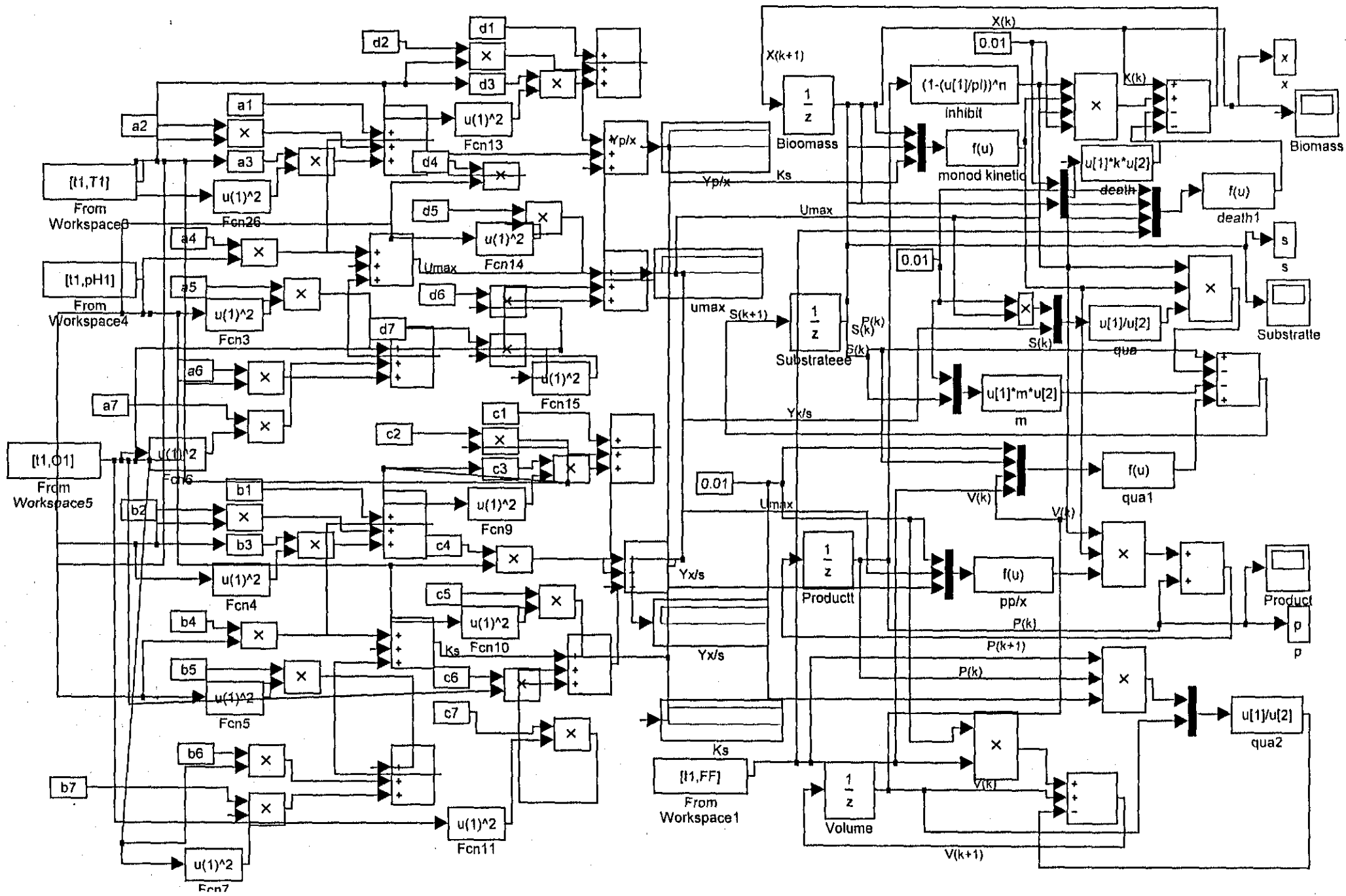
PID Autotuning

Monitoring, Control of Setpoint and Data Logger
E:\Source code\Main Source\DataAcquisitionReal.vi
Last modified on 11 Feb 2003 at 10:39 PM
Printed on 18 Feb 2003 at 11:31 AM



Monitoring, Control of Setpoint and Data Logger
E:\Source code\Main Source\DataAcquisitionReal.vi
Last modified on 11 Feb 2003 at 10:39 PM
Printed on 18 Feb 2003 at 11:31 AM





```
*****  
*****  
% Parameter estimator  
*****  
*****  
global par  
Model_variables;  
% forming the vector of parameters  
par=[a1 a2 a3 a4 a5 a6 a7 b1 b2 b3 b4 b5 b6 b7...  
      c1 c2 c3 c4 c5 c6 c7 d1 d2 d3 d4 d5 d6 d7];  
%introducing the vectors of the measured data for the biomass, substrate and  
d product  
xbar=0.12*exp(0.11*[0:50])-0.1*[0:50];  
sbar=42*exp(-0.1*[0:50]);  
pbar=0.001*exp(0.15*[0:50])-0.01*[0:50];  
ybar=[xbar;sbar;pbar];  
% solution of the model equations with the initial values of the parameters  
j = 1; %initial value of the number of iteration  
while j<=M  
    j  
    tspan = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 ...  
            20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 ...  
            38 39 40 41 42 43 44 45 46 47 48 49 50];  
    [t,y]=ode15s('quadratic_model',tspan,y0);  
    yt=y'  
    % calculation of the value of the criterion  
    eps=ybar-yt  
    J=0;  
for i = 1:K+1  
    J =J+sum(eps(:,i).*eps(:,i));  
end;  
  
% calculation of the deviation value of the vector of the parameter  
for i=1:r  
    par(i)=par(i)+delta;  
    %calculation of the values of y with par  
    [t,y]=ode15s('quadratic_model',tspan,y0);  
    yt=y';  
    % calculation of the deviated value of the criterion  
    eps=ybar-yt;  
    dJ=0;  
    for k=1:K+1  
        dJ=dJ+sum(eps(:,k).*eps(:,k));  
    end;  
    %calculation of the gradient of the criterion  
    dJdp = (dJ-J)/delta;  
    % forming the vector of the criterion gradients for all values of the pa  
rameters par  
    dJdpv(i)=dJdp;
```

```
% calculation of the old value of parameters
par(i)=par(i)-delta;
end;
% calculation of the gradient for the fastest descent method
%calculation of the weighted sum
dJdpv;
s1=dJdpv*dJdpv';
s=sqrt(s1);
% calculation of the direction of the gradient
dp=-alfa*dJdpv/s;
% calculation of the improved estimate of the parameters
par=par+dp;
% check for achieving of the optimal solution
%calculation of the norm
dnorm=norm(dp);
% check termination
if dnorm<=epsilon
    [t,y]=ode15s('quadratic_model',tspan,y0);
break
else j=j+1, alfa=alfa/2.
end
end
par
figure(1)
plot(xbar)
hold on
plot(sbar)
hold on
plot(pbar)
Xlabel('Time')
Ylabel('xbar(t), sbar, pbar')
title('Measured x, s, and p')
hold off

figure(2)
plot(t,y(:,1))
hold on
plot(t,y(:,2))
hold on
plot(t,y(:,3))
Xlabel('Time')
Ylabel('x(t),s(t) & p(t)')
title('Model x, s and p')

figure(3)
plot(sbar, 'b')
hold on
plot(y(:,2), 'r+:')
Xlabel('Time')
```

```
Ylabel('s(t)')  
title('Substrate')  
hold off
```

```
figure(4)  
plot(pbar, 'b-')  
hold on  
plot(y(:, 3), 'r+:')  
Xlabel('Time')  
Ylabel('p(t)')  
title('Product')  
hold off
```

```
figure(5)  
plot(xbar, 'b-')  
hold on  
plot(y(:, 1), 'r+:')  
Xlabel('Time')  
Ylabel('x(t)_bar, x(t)')  
title('Measured biomass and Model biomass')
```

```
%Create function
function ydot=bioreactorfc(t,y)
global par
Model_variables;
ydot(1) =(par(1)+par(2)*pH+par(3)*(pH*pH)+par(4)*T+par(5)*(T*T)+par(6)*DO2+
par(7)*(DO2*DO2))*((1-(y(3)/p1))^n)*y(1)*y(2)/(par(8)+par(9)*pH+par(10)*(pH
*pH)+par(11)*T+par(12)*(T*T)+par(13)*DO2+par(14)*(DO2*DO2)+y(2))-(kd*y(1));
ydot(2) =(par(1)+par(2)*pH+par(3)*(pH*pH)+par(4)*T+par(5)*(T*T)+par(6)*DO2+
par(7)*(DO2*DO2))*((1-(y(3)/p1))^n)*y(1)*y(2)/((par(8)+par(9)*pH+par(10)*(p
H*pH)+par(11)*T+par(12)*(T*T)+par(13)*DO2+par(14)*(DO2*DO2))+y(2))/-(par(15
)+par(16)*pH+par(17)*(pH*pH)+par(18)*T+par(19)*(T*T)+par(20)*DO2+par(21)*(D
O2*DO2))-m*y(1);
ydot(3) =(par(1)+par(2)*pH+par(3)*(pH*pH)+par(4)*T+par(5)*(T*T)+par(6)*DO2+
par(7)*(DO2*DO2))*((1-(y(3)/p1))^n)*y(1)*y(2)/((par(8)+par(9)*pH+par(10)*(p
H*pH)+par(11)*T+par(12)*(T*T)+par(13)*DO2+par(14)*(DO2*DO2))+y(2))* (par(22)
+par(23)*pH+par(24)*(pH*pH)+par(25)*T+par(26)*(T*T)+par(27)*DO2+par(28)*(DO
2*DO2));
ydot=[ydot(1) ydot(2) ydot(3)]';
```


%%%%%%%%%% Part of the optimisation program %%%%%%%%%%

% Global parameter's declaration%

M1 = 500;

M2 = 10; % Max number of iteration second and first level

% Steps of the gradient procedures

alx = 0.01;

als = 0.01;

alp = 0.001;

alv = 0.005; % for conjugate variables

adx = 0.01;

ads = 0.05;

adp = 0.005;

adv = 0.01; % for interconnections in time

ax = 0.001;

as = 0.001;

ap = 0.0001;

av = 0.01; % for state variables

aF = 0.01;

aT = 0.01;

apH = 0.005;

aDO2 = 0.005; % for control variables

% Values of the errors for termination of the calculations

eps1 = 0.001;

epsx = 0.01;

epss = 0.01;

epsp = 0.01;

epsv = 0.01; % for control variables

epsF = 0.01;

epsT = 0.01;

epspH = 0.01;

epsDO2 = 0.01; % for control variables

% Initial trajectories of the conjugate variables

K = 50;

for k = 0:K-1

lx(k)=0.01

ls(k)=0.005

lp(k)=0.002

lv(k)=0.003

end

% Initial values of state variables

x0 = ;
s0 = ;
p0 = ;
v0 = ;
y0 = [x0 s0 p0 v0]';

% Initial trajectories of control variables

for k = 0:K-1
 F(k) = 2;
 T(k) = 16;
 pH(k) = 4;
 DO2(k) = 37;
end

% Model co-efficients

a1 = 0.0004;
a2 = 0.006;
a3 = 0.001;
a4 = 0.005;
a5 = 0.0005;
a6 = 0.001;
a7 = 0.0002;

b1 = 5.2;
b2 = 1.03;
b3 = 0.1;
b4 = 0.7;
b5 = 0.01;
b6 = 0.02;
b7 = 0.01;

c1 = 0.00001;
c2 = 0.00001;
c3 = 0.00001;
c4 = 0.00001;
c5 = 0.00010;
c6 = 0.00010;
c7 = 0.00012;

d1 = 0.00040;
d2 = 0.006;
d3 = 0.001;
d4 = 0.005;
d5 = 0.0005;
d6 = 0.001;
d7 = 0.0002;

```
par = [a1 a2 a3 a4 a5 a6 a7 b1 b2 b3 b4 b5 b6 b7 c1 c2 c3 c4 c5 c6 c6 d1 d2✓  
d3 d4 d5 d6 d7];
```

```
k = 0.09;  
p1 = 50;  
m = 0.0011;  
n = 0.52;  
si = 0;
```

```
% Constraints of state and control variables
```

```
for k = 0: K-1
```

```
    Tmin(k) =  
    pHmin(k) =  
    DO2min(k) =  
    Fmin(k) =  
    Tmax(k) =  
    pHmax(k) =  
    DO2max(k) =  
    Fmax(k) =
```

```
end % for control variables
```

```
for k = 0:K-1
```

```
    Vmin(k) =  
    Vmax(k) =  
    pmin(k) =  
    pmax(k) =  
    smin(k) =  
    smax(k) =  
    xmin(k) =  
    xmax(k) =
```

```
end % for state variables
```

```
% Calculation of the initial trajectories of state variables
```

```
Come back
```

```
[t, y]=ode15s('quadratic_model', tspan, y0);  
x=y(:,1);  
s=y(:,2);  
p=y(:,3);  
v=y(:,4);
```

```
% Calculation of the initial trajectories of the interconnections in time d✓  
omain
```

```
for k = 0:K-1
```

```
    dx(k)=x(k+1);  
    ds(k)=s(k+1);  
    dp(k)=s(k+1);  
    sv(k)=v(k+1);
```

```
fs(k)=ff(k)*Ks(k)*x(k)/((Ks(k)+s(k))^n);  
fp(k)=dt*mmax(k)*Ypx(k)*((1-p(k)/pl)^(n-1))*x(k)*s(k)/(pl*(Ks(k)+s(k)));  
fv(k)=dt*f(k)/(v(k)^2)
```

```
f1(k)=ff(k)*x(k)*s(k)/(Ks(k)+s(k));  
f2(k)=f1(k)/(Ks(k)+s(k));
```

```
% Calculation of the gradients for state and control variables
```

```
eex(k)=lx(k)-mx*elx(k);  
ees(k)=ls(k)-ms*els(k);  
eep(k)=lp(k)-mp*elp(k);  
eev(k)=lv(k)-mv*elv(k);
```

```
ex(k)=(1+fx(k)-dt*kd-dt*F(k)/V(k))*eex(k)+(fx(k)/Yxs-dt*m)*ees(k)+dt*fx(k)*  
Ypx*eep(k);  
es(k)=fs(k)*eex(k)+(1-fs(k)/Yxs-dt*F(k)/V(k))*ees(k)+Ypx*fs(k)*eep(k);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%   Coefficients for Simulink simulation   %  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
TT = 32;  
t = 1:1:TT;  
F = 0;  
si = 0;  
pH = 4.8;  
T = 16;  
O = 37;  
pH = pH*ones(1,TT);  
T = T*ones(1,TT);  
O = O*ones(1,TT);  
F = F*ones(1,TT);  
a1 = 0.0004;  
a2 = 0.006;  
a3 = 0.001;  
a4 = 0.005;  
a5 = 0.0005;  
a6 = 0.001;  
a7 = 0.0002;  
  
b1 = 5.2;  
b2 = 1.03;  
b3 = 0.1;  
b4 = 0.7;  
b5 = 0.01;  
b6 = 0.02;  
b7 = 0.01;  
  
c1 = 0.00001;  
c2 = 0.00001;  
c3 = 0.00001;  
c4 = 0.00001;  
c5 = 0.00010;  
c6 = 0.00010;  
c7 = 0.00012;  
  
d1 = 0.00040;  
d2 = 0.006;  
d3 = 0.001;  
d4 = 0.005;  
d5 = 0.0005;  
d6 = 0.001;  
d7 = 0.0002;  
t1=t';  
pH1=pH';  
T1=T';  
O1=O';
```

```
FF=F';  
k = 0.09;  
pl = 50;  
m = 0.0011;  
n = 0.52;  
sim('quadratic_model_real')  
plot(x)  
hold on  
plot(s)  
hold on  
plot(p)  
hold off
```