OPTIMIZATION OF RECOMBINANT BACTERIAL FERMENTATIONS FOR PHARMACEUTICAL PRODUCTION

A Thesis

Submitted to the College of Graduate Studies and Research

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in the Department of Chemical Engineering

University of Saskatchewan

By

Hamid Reza Baheri

Spring 1998

PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirement for a postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this university may make it freely available for inspection. I further agree that permission for copying this thesis in any manner, in whole or in part, for scholarly purpose may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in this thesis.

Request for permission to copy or to make use of material in this theses in whole or in part should be addressed to:

Head of the Department of Chemical Engineering
University of Saskatchewan
Saskatoon, SK.,
S7N 5C9
Canada

ABSTRACT

In this research program, the optimization of a two stage continuous process and a fedbatch process for fermentation of recombinant *E. coli* has been investigated. The study was conducted in the following three phases: a) developing two computer programs for simulation and optimization of a continuous two stage and a fedbatch fermentation process, b) conducting batch culture fermentations to verify the performance of the biokinetic model for bacterial growth and cloned gene expression, and c) conducting fedbatch and two stage continuous fermentation experiments to closely examine the simulation and optimization results.

After studying several approaches, the optimization algorithm chosen for this simulation study was the Pattern-Search method developed by Hooke and Jeeves (1962). The Miao and Kompala (1992) biokinetic model was chosen to simulate the bacterial growth and cloned gene expression in recombinant cultures. The simulation results showed 30-40% higher productivities for the two stage continuous process over the fedbatch process when using the same medium in both processes.

The feasibility of using multiple CFSTR instead of two CFSTR for fermentation of recombinant bacteria was then studied for the first time. Extensive simulations of the optimum multiple CFSTR (up to four stages) with different combinations of growth and production stages showed that increasing the number of growth stages in the continuous fermentation did not improve the productivity at all. Increasing the number of production stages only slightly improved the productivity of the process (10-12%). This result was found to be generally applicable to many recombinant fermentation processes reported in the literature.

The information from the process optimization was then used to design both fedbatch and two stage continuous experiments and compare the performance of these two processes. Experiments were conducted in batch, fedbatch and the continuous two stage processes. Recombinant *E. coli* (strain *BL21DE3*) with an inducible gene (sensitive to *IPTG, isopropyl-β-D-thiogalactopyranoside*) was grown mainly in 500-1000 mL, temperature controlled bioreactors with the exception of batch experiments which were performed in shake flasks equipped with air spargers. A unique formulation of M9 minimal medium supplemented with a mixture of glucose and casamino acids (50-50%) was developed and used for fermentations

at substrate concentrations of 6 and 10 g/L. As the substrate concentration increased from 2 to 10 g/L, it was found that the enzyme activity increased by a factor of 10. This increase, which was in part due to the higher biomass concentrations and in part due to the higher specific growth rates in the new media, was successfully predicted by the model simulations after using new biokinetic parameters for the improved media.

The average deviation between the fedbatch experimental and simulation results was about 12% and about 15% for the two stage continuous process (over a period of 36 h continuous operation). The experiments also revealed that the maintenance of plasmid harboring cells over the long term operation was an important barrier in achieving the predicted high productivity in the two stage continuous process.

Finally, in addition to computer programs for optimizations of genetically modified microorganisms, a new computer program with a generic algorithm for optimization of multiple CFSTR fermentation with any kind of biokinetic model has been developed. Extensive testing of the program with published optimization data has confirmed the accuracy of the program (less than 3% error). Optimization of multiple CFSTR fermentations with the cybernetic

biokinetic model was also performed for the first time. Besides finding the optimum residence times for multiple CFSTR operation, the effect of inaccuracies in different cybernetic model parameters on the overall productivity of the process was investigated. The simulation results clearly illustrate that, a single CFSTR is more sensitive in its operation to *inaccuracies* in the biokinetic constants as compared to optimized CFSTR in series (2-8 times more sensitive).

ACKNOWLEDGEMENTS

I consider myself fortunate to come across many noble professors and friends during my PhD study in Canada. I would like to express my sincere thanks to all of the following:

My supervisors, Dr. G.A. Hill and Dr. W.J. Roesler, without their keen involvement, this work would have not been possible.

My other committee members: Drs. M. Foldvari, B. Milne, J. Postlewaite and S. Rohani for their support and critical feedback at every stage of the project; Dr. J. Piret, my external examiner and Dr. W.M. Ingledew, the chair of the defence.

The staff of the department of chemical engineering, especially Ms. K. Burlock, Ms. M. Grandrude, Ms. C. Goll, Mr. T. Wallentiny and Mr. D. Claude, and Chemical engineering graduate students, Sam, Ali, Gerald, Vickie, Greg, Jason, Steven, Mehdi, Hedi, Ashraf, Abdul, Rohit, Path, Is, Tracy, Bi and Gao.

MCHE of Iran and NSERC, for the financial support.

Finally I am short of words to express my gratitude to Dr. D.G. Macdonald, who was the chair of the advisory committee.

To my wonderful wife **Maryam**, who left her studies in Pharmacy to join me here, and whose support made this work possible,

To our daughter **Nadia**, whose presence has brought such good fortune.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

A	adenine	
a	constant in Equation (3.5)	g/L.h
A′, B′	constants in William's model	
B	constant in Equation (3.5)	g/L
BSA	bovine serum albumin	
C	cytosine	
CFSTR	continuous flow stirred tank reactor	
D	dilution rate (F/V)	h^{-1}
d_I	impeller diameter	m
d_T	tank diameter	m
d_W	impeller width	m
E	basal enzyme	
e	enzyme	
F	feed flow rate	L/h
f	function	
G	guanine	
G_p	plasmid concentration	mg/g dry cells

G'	growth	
I_e	extracellular inducer concentration	g/L
I_i	intracellular inducer concentration	g/L
IPTG	isopropyl-β-D-thiogalactopyranoside	
K	constant	g/L
K_s',K_s	Monod constants	g/L
k	rate constant	L/g.h
k_d	degradation rate constant	h^{-1}
k_μ	saturation constant	h^{-1}
Mg	main gene	
m	mass of biotic phase	g
N	impeller speed	rpm
n	adjustable parameter	
ONPG	o-nitriphenyl-β-D-galactosidase	
Og	Operator	
p	foreign protein concentration	g/gDW
P	Product concentration	g/L
PD	productivity	g/L.h
Pr	Promoter	
r	rate	g/L.h
R	Resource concentration	g/L

RPD	relative productivity to the optimum two stage fermentor	
S	substrate concentration	g/L
T	time	h
U	uracil	
U'	cybernetic variable	
v_R	cybernetic variable	g/L
T	thymine	
V	volume of the reactor	L
X	cell concentration	g/L
Y_{xs}	biomass yield	g dry cell/ g glucose
Y_{PX}	product yield	g product/ g biomass
Z	concentration of mature cells	g/L

Greek letters

α	protein synthesis rate constant	
β	non-growth associated enzyme production rate constant	h^{-1}
η	translation efficiency	
θ	segregation coefficient	

μ	specific cell growth rate	h^{-1}
α', α^{*}_{EML}, α'_{EML},		
β', γ', θ'	constants	
ρ	density of biotic phase	g/L
ξ	transcription efficiency	

Subscripts

0	initial value
1, 2, 3, 4	parameter index
A′	cell synthetic portion
B′	cell structured portion
EML	low maintenance enzyme
G	growth
i, j, k	counter
M	basal maintenance
ML	low maintenance
m	maximum
P	product (g/ L)
p	foreign protein (g/g cell)
R	resource
S	substrate

TM	maintenance
X	biomass
XS	biomass to substrate
S	substrate
t	total

Superscript

+	plasmid-bearing cell
-	plasmid-free cell
m, max	maximum
t	total

CHAPTER ONE - INTRODUCTION

Microorganisms have been used to produce foods such as vinegar, yogurt, and cheese for hundreds of years. However, until about a century ago, it was not realized that microorganisms were involved in the production of these bioproducts. The discovery only came when the French scientist Louis Pasteur discovered that microorganisms were the cause of beer and wine spoilage. Ethanol was one of the first chemicals which was produced commercially with the aid of biotechnology.

Several more significant advancements in biotechnology were made during the First and Second World Wars, especially with the introduction of penicillin, an antibacterial substance produced by the mold *Penicillium notatum.* But the most spectacular development in biotechnology was made by the introduction of recombinant DNA technology in the late 70's. This technology, which is now one of the most actively researched branches of biotechnology, is based on the fact that genetic information is carried in a molecule that has the same chemical structure in all organisms (deoxyribo nucleic acid or

DNA). This means that the DNA of humans is chemically similar to the DNA of a bacteria or fungi. The DNA molecule is a polymer made of four subunits often abbreviated as *A, C, G, T (*nitrogenous bases of adenine (A), cytosine (C), guanine (G), or thymine. DNA stores the information as a sequence of these four subunits, similar to a sequence of twenty six letters in the alphabet that can be made to carry information. The information in the DNA molecule is for the production of specific proteins and enzymes by the cell, which in turn control the metabolism and the structure of the cells and organism. Different microorganisms have different sequences of these subunits and sizes of the DNA molecule (amount of information they carry), but all DNA molecules follow the same rules and structure. This fact made it possible, for instance, to express the human genes coding for insulin or interferon in fast growing bacteria (using a modified DNA or "recombinant DNA"). It may sound simple, but it took a great deal of effort by scientist including biochemists, chemists, geneticists and biochemical engineers to make it possible and commercially feasible in the early 80s.

Today the production of some of the most important and complex pharmaceuticals such as insulin, hormones and antibodies are carried out using genetically modified microorganisms. But even

from the early stages of commercial production of recombinant proteins, engineers have been facing new challenges in the design and handling of processes for recombinant cultures. One of these challenges includes coping with the problem of instability in recombinant organisms. We know that commercial production of products on a large scale, especially in the pharmaceutical industry, depends largely on the stable maintenance of the organism. It is also evident that during the production process, the organism is forced to use a high percentage of its activity toward the production of something which is not useful for the cell. This will set them at a competitive disadvantage compared to non-producing cells. This can lead to a declining population of the producing cells and loss of productivity. Another problem is the plasmid loss from recombinant organisms as they undergo cell divisions. There has been a significant effort to overcome these problems and one of the most successful approaches has been the use of regulated genes (inducible genes) to separate growth and production. This allows the engineer to minimize expression pressure on the producing cells (recombinant cells) until the end of process when there is less opportunity (time and substrate) for non producing cells to predominate. The separation of growth and production phases has

successfully been implemented in both batch and fedbatch operations as well as in continuous process (Siegel et al., 1985; Seo and Bailey, 1986; Lee et al. 1988; Park et al., 1990; Miao and Kompala, 1992 & 1994).

In this investigation, two computer packages were developed for simulation and optimization of fermentation of recombinant bacteria in fedbatch and two stage continuous processes. The parameters considered for optimization in the fedbatch process were induction time, percentage of initial charge and in situ substrate concentration. For the two stage continuous process, the optimized parameters were feed distribution and volumes of reactors. Finally, several fermentation runs were carried out in both processes and in batch mode to directly examine the modeling equations and the optimization results.

In addition to the above study on the fermentation of recombinant bacteria, another study was conducted to develop a generic computer program for optimization of fermentation in multiple CFSTR. After testing the optimization program with the results from several earlier investigations on simple biokinetics, the computer program was used to study two different subjects:

- feasibility of using multiple CFSTR for fermentation of recombinant bacteria (using more than two CFSTR with feed distribution for each stage).
- effect of inaccuracies in the cybernetic model parameters on the performance of the continuous process.

CHAPTER TWO - LITERATURE SURVEY

2.1. Introduction to Genetic Engineering

Gene cloning, gene splicing, expression and recombinant DNA are some of the common words used in the new biotechnology era, the era of genetic engineering. Traditionally, biotechnology means the "life technology", or the application of our knowledge of living species for use by human beings such as in bread and cheese making (Primrose, 1987). However, the new biotechnology is centered on the internal mechanism of living things, the genetic material of the organism. Among several thousand different kinds of molecules found within the cell, DNA (deoxyribonucleic acid) is the master molecule of the internal mechanism in which all the information needed to create and direct the organism is encoded. Genetic engineering is a collection of techniques which enable us to manipulate this genetic material. Adding foreign genetic information to the DNA of an existing organism can be used both to produce

foreign proteins at high rates and yields (such as human antibodies and hormones), or to give new phenotypes to the organism (such as resistance to disease). But the starting point for this technology is to learn more about the DNA and a closely related molecule, RNA.

2.1.1. DNA and RNA

DNA (*deoxyribonucleic acid*) is perhaps the most important molecule of living things, because it stores the information to make and direct the machinery of the cells to grow and to divide. DNA and a closely related molecule, RNA *(ribonucleic acid*), are large polymers that are built of simple subunits. The subunits, called nucleotides, contain three major components (Fig. 2.1)

1- a cyclic sugar, deoxyribose for DNA and ribose for RNA;

2- a nitrogenous base, either adenine (A), cytosine (C), guanine (G), thymine (T), or uracil (U);

3- a phosphate group attached to the sugar unit.

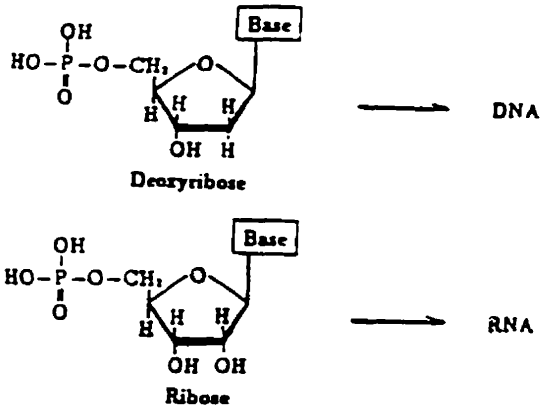


Figure 2.1. Structure of subunits (taken from Lee , 1992).

The names of DNA and RNA are derived from the sugar bases in each subunit (Figure 2.1). The DNA is made up of subunits having A, T, G and C nitrogenous bases while RNA uses subunits having A, C, G and U nitrogenous bases. The DNA structure of all living organisms are remarkably similar, the differences lie within the sequences of the subunits and the size of the DNA molecule in each organism (amount of information they carry). Another important feature of DNA is that it usually consist of two complementary strands coiled in the form of a double helix (Watson et al., 1992) (Fig. 2.2).

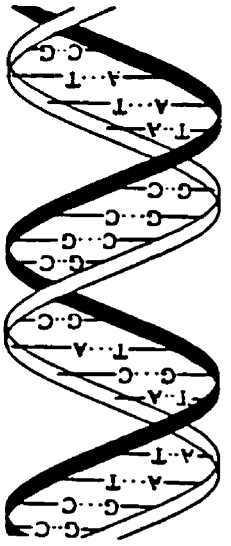


Figure 2.2 -Model of DNA double helix (taken from Lee, 1992).

The two chains are joined together by hydrogen bonds which occur only from A-T and G-C subunits. Adenine is always paired with thymine and guanine with cytosine. The RNA structure is very similar to DNA except uracil is used in place of thymine, and it is not usually a double helix but a single strand.

2.1.2. DNA - Protein Coding

From the early 20th century, it was known that proteins and enzymes were composed of amino acids linked to each other. It was later found that the sequence of bases in a strand of DNA specifies the order in which amino acids are assembled to form proteins. The sequencing code was fully deciphered by 1966. Since there are twenty amino acids and only four bases in DNA, each code in the DNA must consist of at least three bases to represent a unique

amino acid (two bases gives only 4^2 =16 combinations, three bases gives 4^3=64 triplets). Several triplets may designate the same amino acid, because the number of triplets (64) is much higher than number of amino acids (20). For example, UCU, UCC, UCA and AGU are all signal for serine (Watson et al., 1992). Figure 2.3 shows a simple translation of a DNA code to its equivalent amino acids.

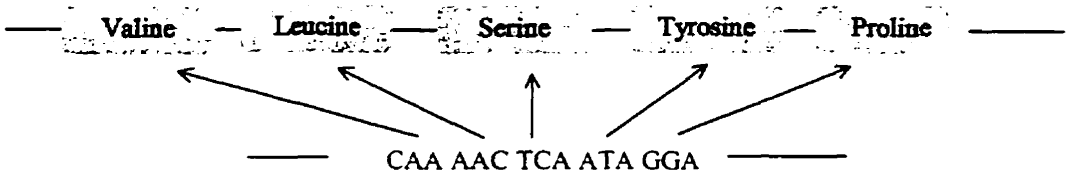


Fig 2.3- Simple translation of a DNA sequence to the amino acid sequence.

2.1.3. Cloning and Transformation

The production of recombinant DNA depends on utilizing enzymes that react with the DNA molecule in a very specific and reproducible way. The most important of these enzymes are the *restriction enzymes* which cut DNA molecules at specific sites. The cutting site is determined by a particular nucleotide sequence (*recognition sequence*).

There are many restriction enzymes and each enzyme cuts the DNA molecule at its own recognition sequence. For example, the restriction enzyme *Hinc*II cuts DNA symmetrically whenever it encounters a GTCGAC sequence to produce two fragments. These fragments are often called blunt fragments (Fig. 2.4a). Another restriction enzyme, *Eco*RI, obtained from *E. coli*, cuts DNA at GAATTC sequences in a staggered fashion to produce two sticky end fragments (Fig. 2.4b).

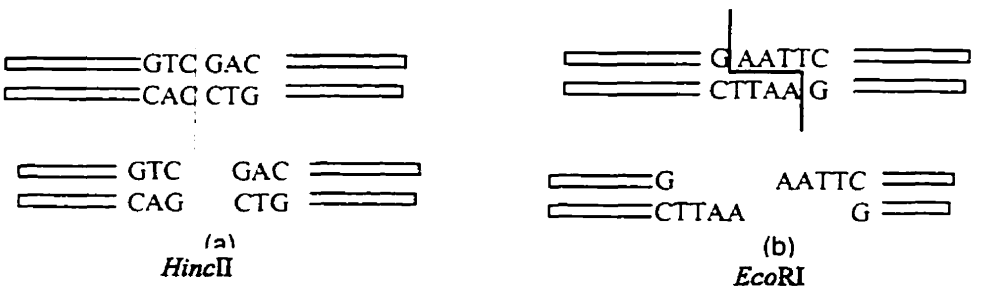


Figure 2.4. a & b- *Hinc*II and *Eco*RI restriction enzymes and their cleavage sequence.

The fragments are called sticky because they can be easily joined to each other or to other similar fragments with complementary base pairing. The enzyme which is used to catalyze the joining of the fragments is called *DNA ligase*.

The use of restriction enzymes (for cutting) and DNA ligase enzyme (for joining) allows scientists to break down DNA molecules

at specific sites (*splicing*) and then rejoin the pieces (plus desired foreign codes) into new DNA molecules called *recombinant molecules* (*cloning*). This process is most often performed outside the cell, in a test tube using small pieces of DNA known as plasmids which are found in a variety of bacteria. Plasmids contain genes that are beneficial to the bacterial host, such as genes for resistance to antibiotics, production of antibiotics, or degradation of complex compounds, but plasmids are not part of the main DNA molecule. A suitable plasmid for DNA manipulations should have the ability to replicate in high numbers inside the bacteria, contain a *selection marker* and be small enough to be inserted into the host cell easily (*transformation*). Figure 2.5 shows the simplified steps in producing a recombinant DNA bacterium.

Cloning and Transformation

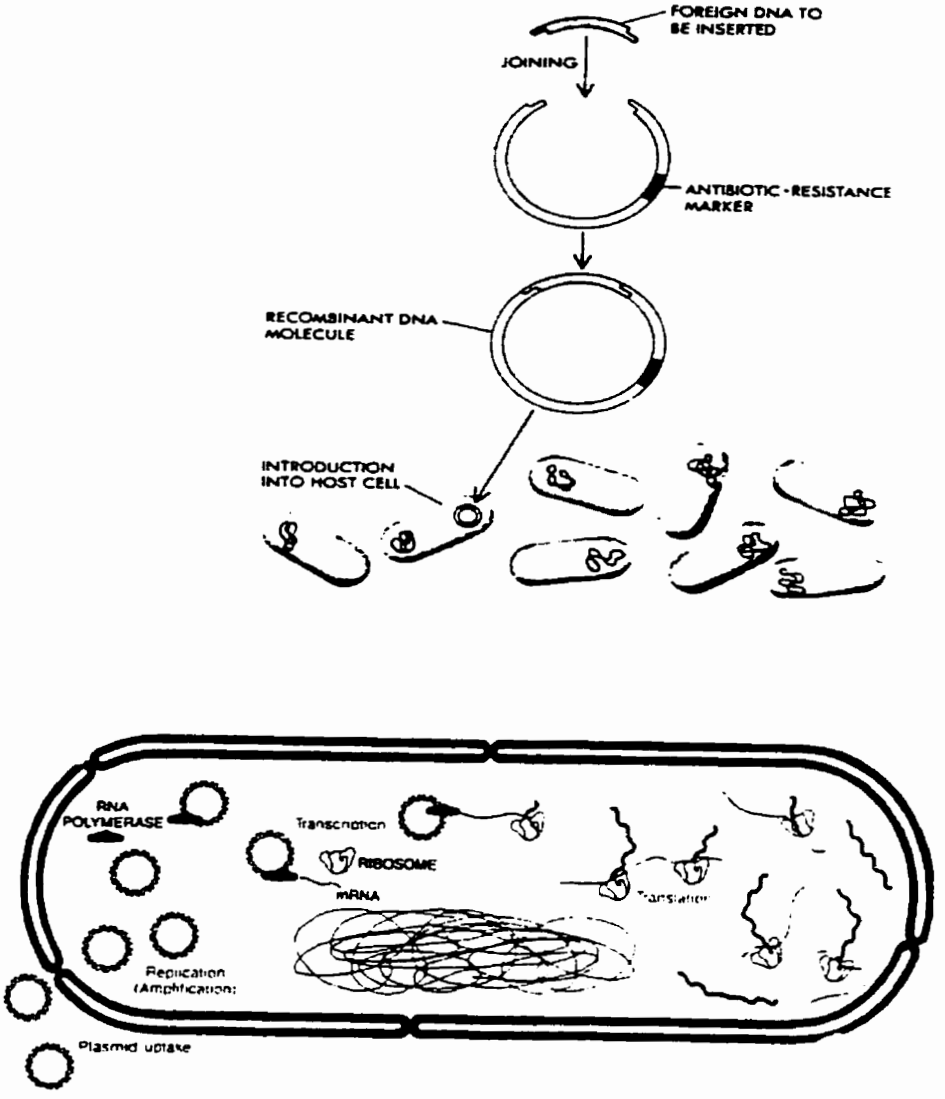


Figure 2.5. Basic steps of cloning and transformation in a recombinant bacteria (adapted from Watson et al., 1992).

2.1.4. Gene Expression

A gene is a part of the DNA molecule which codes for a specific protein. The process of expression of a gene to form a sequence of amino acids is achieved by two main steps called *transcription* and *translation*. Three forms of RNA are involved in these two processes. In the first step, a complex enzyme (RNA polymerase) transcribes the gene codes into a complementary RNA code (messenger RNA or mRNA) using DNA as the template. In the second step, the information on the mRNA is translated (decoded) to its equivalent proteins by using two other RNA molecule known as transfer RNA (tRNA) and ribosome RNA (rRNA). The ribosomal RNA acts as a template and binds to *ribosome binding sites* of mRNA, then tRNA molecules bring activated amino acids to the process of making proteins (Fig. 2.6). The process of expression of a gene is facilitated (controlled) by several coding signals at the beginning and end of a gene in the DNA molecule. Figure 2.7 shows a simplified scheme of information sequence in a DNA molecule which guides the expression process.

Figure 2.6. Two steps in DNA expression to protein

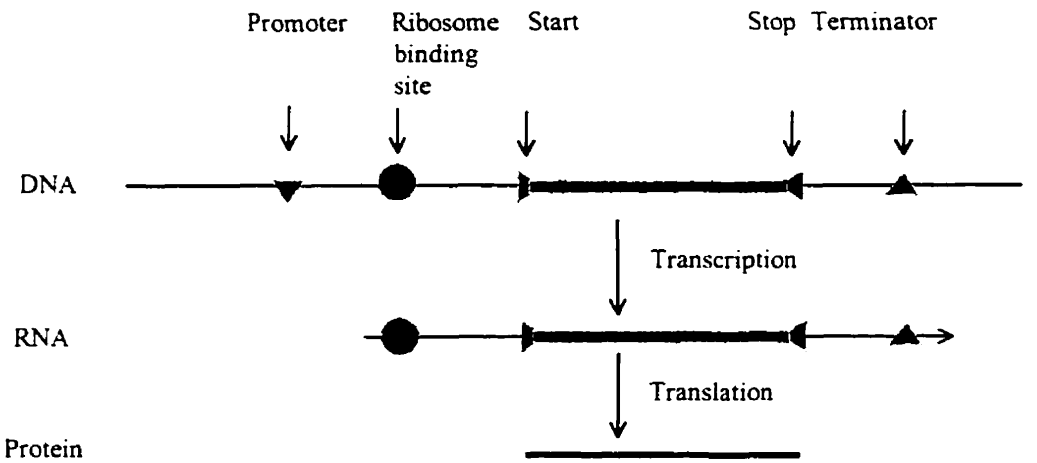


Fig 2.7. A typical DNA information sequence to guide the transcription and translation process (Adapted from Bailey and Ollis, 1986)

The *promoter* is a code that signals the start of the process of transcription. This code then is followed by the *ribosome binding sequence* which is necessary for binding mRNA to rRNA. The *start*

codon and *stop codon* signal the actual start and end of signals for the location of a specific protein and the *terminator* codon signals the end of transcription. As is evident from Figure 2.7, the final protein is translated only by codes between the start codon and the stop codon.

2.1.5. Protein Synthesis: Induction and Repression

The levels of different proteins and enzymes are not always constant in bacteria. A specific enzyme may be needed in one environment, but not required in another environment. It would be wasteful to make all the enzymes at the same rate, and in fact many bacterial genes are constructed to function at variable rates. Whenever the gene receives a signal from the outside, transcription starts, otherwise the transcription is blocked to reduce unnecessary production of unwanted proteins. The compound which causes the transcription to start is called the *inducer*, and the compound which blocks the transcription is called the *repressor*. In simple terms, the repressor is a small protein which binds to a specific location on the gene and blocks the transcription. The site the repressor binds to is called the *operator* and is often close to the promoter sequence. The binding of a repressor molecule to the operator sequence blocks the

transcription of the gene by RNA polymerase (Fig. 2.8a). On the other hand, the *inducer* is another molecule which can attach to the repressor and prevent it from binding to DNA, so that transcription can take place as shown in Fig. 2.8b.

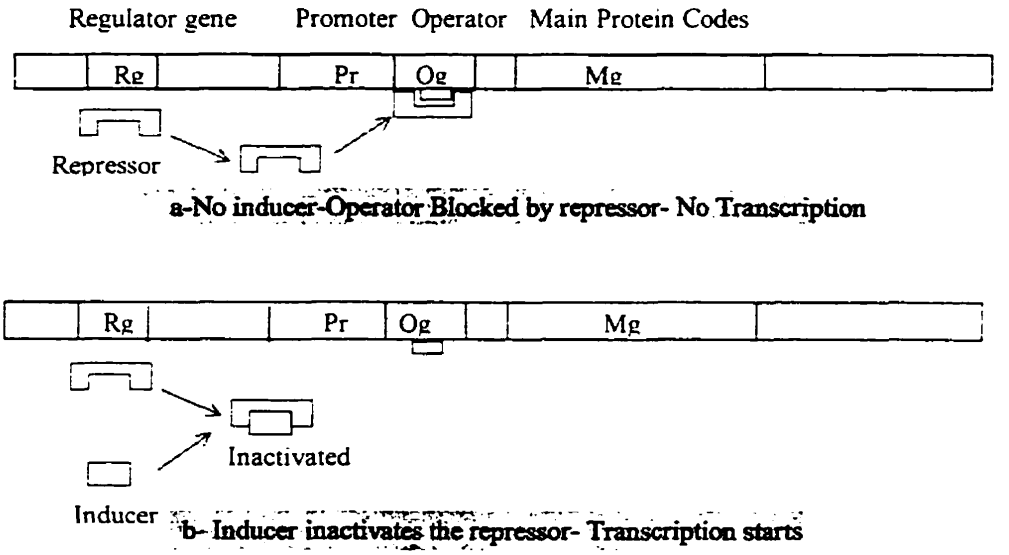


Figure 2.8 a & b. Induction and repression process (Adapted from Watson et al., 1992).

2.1.5.1. β-galactosidase (β-gal)

The enzyme β-galactosidase is often called *lactase*. It catalyzes the hydrolyses of lactose to the simpler sugars, glucose and galactose for bacteria, yeasts and fungus. Lactose is present in milk, where its concentration ranges from 0-7%. The β-gal enzyme is

present in infant intestines to hydrolyze lactose, however in many adults, the level of this enzyme is low which causes *milk intolerance*.

In *E. coli*, the gene for the β-gal enzyme is located adjacent to two additional genes involved in lactose metabolism. One gene codes for lactose permease, a protein that facilitates entrance of lactose to bacteria, while the other gene codes for thiogalactosidase transacetylase, an enzyme that may help the break down of lactose similar compounds which are not digested by the β-gal enzyme. The same mRNA that codes for β-gal, codes for permease and acetylase. This means that when lactose is added to *E. coli* cells (at low glucose concentration), the relative amount of these three enzymes rises at the same rate. This collection of adjacent genes that are transcribed into a single mRNA, together with the control region, is generally called the *operon* and in this case, the *lactose operon (lac operon)* . Genetic studies have shown that 1)- a specific repressor exists that binds to the beginning of the β-gal gene and prevents its transcription and 2)- allolactose acts as an inducer by binding to the repressor molecule, thus allowing expression of the *lac* operon.

2.1.6 Stability of Recombinant Microorganisms

The yield of recombinant microorganisms often decreases during fermentations to produce foreign proteins. This is caused to some extent by the loss of plasmids from organisms as they undergo cell division. The plasmid instability has been reported in systems employing bacteria as well as yeast (Kim & Ryu, 1984; Schwartz et al., 1988) .

In 1981, Imanaka and Aiba showed that the percentage of cells carrying foreign plasmids depends on the number of generations of growth. This means that as the number of generations increases, the fraction of plasmid-carrying cells decreases. This phenomenon becomes a serious problem in continuous operations. The instability has been characterized in two different modes (Lee, 1992): (a)- defective segregation of plasmids during cell division (uneven distribution of plasmids during cell divisions), and (b)- structural instability due to random mutation of the plasmid.

Structural instabilities are more difficult to prevent because the changes result in the loss of a specific gene or an inactivated gene. The plasmid still exists inside the cell. There is no simple way to detect this phenomenon. On the other hand, segregational instability

is easier to detect (often by plating) and several methods have been suggested to minimize it:

a) The application of selection pressure for plasmids containing a selection marker. Adding an antibiotic to the growth media prevents plasmid-free cells from growing. However, multiple additions of antibiotic to the media may be necessary due to decomposition. This is sometimes expensive, often inefficient at high cell concentrations, and may cause contamination of the final product (Watson et al. 1992; Miao & Kompala, 1992). Additionally, FDA (Food and Drug Agency) regulations does not allow the use of antibiotics in therapeutic productions due to concerns that it may be found in the product. However this method is still a common and popular method of choice in laboratory experiments mainly because of the simplicity. Table 2.1 lists some common antibiotics, the modes of action and resistance, and the working concentrations used in *E. coli* fermentation.

b) Using specially designed suicidal mechanisms for plasmid-free cells (Kuowei et al., 1994). This can be achieved by using a bacteria strain which is dependent on a plasmid product to survive; in a case of plasmid loss the bacteria will not be able to survive. Two limitations of this method are that 1) the expression of the survival

gene may interfere with the cellular functions, and 2) if the product of the survival gene accumulates inside the cell, it may be inherited by a plasmid-free daughter cell.

c) Using regulated genes. The segragational instability is very high only during the expression of foreign proteins. By using regulated plasmids, it is possible to separate the cell growth from the induction phase and therefore reduce the segregational instability.

Table 2.1 Common antibiotics and their mode of action and resistance in *E. coli* (from Watson et al., 1992).

Antibiotic	Mode of Action	Mode of Resistance	Concentration Used (mg/L)
Ampicillin	Inhibits cell wall synthetisis	β-lactamase inactivates ampicillin by hydrolyzing β-lactam ring	50-100
Gentamicin	Inhibits protein synthesis by binding to ribosome	Aminoglycoside acetyltransferase and aminoglycoside nucleotidyltransferase inactivate gentamicin by acetylation	15
Kanamycin	inhibits protein synthesis- causing misreading of mRNA	Aminoglycoside phosphotransferase and aminiglycoside acetyltransferase and aminoglycoside nucleotidyltransferase inactivate kanamycin by chemical modification	30
Tetracycline	inhibits protein synthesis by preventing translation	One polypeptide that prevents tetracycline from entering cell by decreasing permeability of cell wall	15

2.1.7. *Escherichia coli (E. coli)* Expression System

E. coli is a widely used microorganism in recombinant DNA technology to produce desired proteins. Perhaps the advanced knowledge about the genetics and the structure of *E. coli* has accounted for its preferential use in many recombinant DNA fermentations.

Besides *E. coli, Bacillus subtilis*, *yeast* and *mammalian cells* are other systems currently being used for expressing recombinant DNA products. The choice of the microorganism for the expression of a protein depends on many parameters, including the size, structure and type of protein being produced. *E. coli* has proved to be extremely effective for proteins in the range of 100-200 amino acids (Goeddel, 1990). On the other hand, yeast also grow fast and to high cell densities. They are able to perform *post-translational* modifications that bacteria are unable to perform. Most non-bacterial proteins need further modifications during the expression process to produce fully functional proteins. Therefore, for proteins which are not of microbial origin, yeast expression systems may often be considered to achieve the necessary post-translational modifications.

Mammalian cells are normally considered for the expression of proteins larger than about 500 amino acids (Goeddel, 1990).

2.1.7.1. *E. coli* BL21DE3 Expression System

T7 RNA polymerase (of bacteriophage T7) is very selective for specific promoters which are often not encountered in other DNA molecules, and it transcribes five times faster than *E. coli* RNA polymerase. Since efficient termination signals are also rare for T7 RNA polymerase, it is able to completely transcribe any DNA under the control of a T7 promoter. Additionally, bacteriophage T7 promoters are only recognized by T7 RNA polymerase. These properties make this system very attractive for expression in *E. coli* and even in other systems (Studier et al., 1990; Studier & Moffat, 1986). However, T7 RNA polymerase is not normally present in bacterial cells. One method of introducing the enzyme is by infection of the cell by the bacteriophage carrying this gene. The alternative method is to include the T7 polymerase gene in the cell. This is more convenient than the infection procedure. However, adding the gene to the plasmid will make it too large and unstable, therefore Studier et al. (1986) incorporated a copy of the gene directly into a host bacterial chromosome. An inducible promoter (*lac*UV5 promoter)

was also added upstream of the T7 RNA polymerase gene to control its expression. This combination made it possible to control the expression of the polymerase gene and therefore the expression of foreign genes on the plasmid by adding a chemical inducer (IPTG, isopropyl-β-D-thiogalactopyranoside). Major steps of expression in a T7 promoter system are: 1) binding of the inducer to the repressor 2) expression of T7 RNA polymerase from the chromosome and 3) expression of the target gene on the plasmid by T7 RNA polymerase.

2.2 Kinetics of Cell Growth and Protein Production

2.2.1 Simple Biokinetic Models for Bacterial Growth

Bacterial growth often follows an exponential pattern in a medium with proper nutrients. The change in the biomass with time can be described by the following equation (Lee, 1992):

$$\frac{dX}{dt} = \mu X \qquad (2.1)$$

Here X is the biomass and μ is the specific growth rate of the bacteria. If the concentration of essential medium components

changes, the specific growth rate of bacteria will change. One of the popular correlation between the specific growth rate and the essential nutrient in the medium was proposed by Monod (1942) which states that:

$$\mu = \frac{\mu_m S}{K_S + S} \tag{2.2}$$

Here S is the concentration of the limiting substrate, K_S is the Monod saturation constant, and μ_m is the maximum specific growth rate of the bacteria. The equation is an "engineering simplification" which predicts reasonably well a very complex process. Several modifications have been proposed to improve the model predictions. For the case of product inhibition, the following modification is suggested (Lee, 1992).

$$\mu = \frac{\mu_m S}{K_S + S}\left(1 - \frac{P}{P_m}\right)^n \tag{2.3}$$

Here P is the product concentration and P_m is the hypothetical value for maximum attainable product concentration. Two important other

parameters which affect the specific growth rate are the temperature and the pH of the culture.

2.2.2. More Sophisticated Models

The Monod model considers the cells as a simple component distributed throughout the culture (*distributed*), and also does not account for the changes in the composition of the cells (*unstructured*). Therefore, the model is satisfactory for the biomass growth phase, but is unable to account for the lag phase or changes in the cell size, etc. There are, however, more sophisticated models to recognize the cell components (*structured*), and/or the diversity of cell populations (*segregated*). A number of these models are briefly presented in this section, but more details of each model can be found in the cited references.

2.2.2.1. Blanch and Roger's Segregated Biokinetic Model

The following biokinetic model suggested by Blanch and Rogers (1971) is a good example of a segregated model. In this model, the cell is divided into a premature and mature state. Each state has different product formation rates. These modeling

equations can be used in the simulation of both primary and secondary product formation (Shimizu and Matsubara, 1987):

$$r_X = \mu X - \theta' X \tag{2.4}$$

$$r_S = \frac{\mu X}{Y_{XS}} - \alpha'_3 X \tag{2.5}$$

$$r_Z = \mu X - \beta' Z \tag{2.6}$$

$$r_P = \alpha'_1 \mu X - \alpha'_2 \beta' Z - \gamma' P \tag{2.7}$$

$$\mu = \frac{\mu_m S}{K_S + S}\left(1 - \frac{P}{P_m}\right) \tag{2.8}$$

in which S is the concentration of the limiting substrate, X is the total cell concentration, Z is the concentration of mature cells, P is the concentration of product, r_X, r_S, r_Z, and r_P are the rate equations for total biomass, substrate, mature cells, and product concentration respectively, θ' is the cell death coefficient, α_3 is the maintenance coefficient, and γ' is the product decay constant. α'_1,α'_2, α'_3 and β' are the constants for covering primary, secondary or mixed product formation biokinetics.

2.2.2.2. William's Two Compartment, Structured Model

The simplest structured model is the William's two compartment model, which considers the cell as two basic compartments, a synthetic portion (A′) and a structured section (B′), having different biokinetic equations for growth. The following equations represent a simplified form of this model (Lee 1992) :

$$r_{A'} = \frac{V}{V - \rho m} k_1 SX - \frac{V}{\rho m} k_2 X_{A'} X_{B'} \qquad (2.9)$$

$$r_{B'} = \frac{V}{\rho m} k_2 X_{A'} X_{B'} \qquad (2.10)$$

$$r_s = \frac{-1}{Y_n} \frac{V}{V - \rho m} k_1 SX \qquad (2.11)$$

$$X = X_{A'} + X_{B'} \qquad (2.12)$$

in which X is the total concentration of both compartments, $X_{A'}$ and $X_{B'}$ are the concentration of each compartment respectively, V is the total volume, ρ is the density of biotic phase, and m is the mass of biotic phase.

2.2.2.3. Ballo and Ramkrishna's Cybernetic, Structured Model

Cybernetic models are a class of structured models developed to account for the effects of cellular regulatory mechanism on the growth processes. Metabolic regulations which control the enzyme synthesis are represented by cybernetic variables. The following cybernetic model was suggested by Ballo and Ramkrishna in 1991.

$$r_{G'} = \mu_{G'}^{m} \frac{e}{e^{m}} \frac{S}{K_S + S} \tag{2.13}$$

$$r_{TM} = \mu_{M}^{m} \frac{e}{e^{m}} \frac{S}{K_M + S} + \mu_{ML}^{m} \frac{e_{ML}}{e_{ML}^{m}} \frac{S}{K_M + S} \tag{2.14}$$

$$r_E = \alpha_E^{\bullet} + \alpha'_E \frac{S}{K_E + S} \frac{R}{R + K_R} U'_{G'} \tag{2.15}$$

$$r_{EML} = \alpha_{EML}^{\bullet} + \alpha'_{EML} \frac{S}{K_{EML} + S} U'_{M} \tag{2.16}$$

$$r_R = \alpha'_R \frac{e}{e^{m}} \frac{S}{K_{RS} + S} v_R \tag{2.17}$$

in which $r_{G'}$, r_{TM}, r_E, r_{EML}, and r_R are the rate of growth, maintenance, basal enzyme, low maintenance enzyme, and resource respectively; X is the biomass, e and e_{ML} are the enzyme concentrations, R is the resource, α^{*}_{EML}, α^{*}_{EML} and α'_R are constants, and S is the substrate.

The cybernetic variables U'_G, U'_M, v_R can be calculated by (Ballo and Ramkrishna, 1991):

$$U'_G = \frac{r_{G'} / Y_{XS}}{r_{G'} / Y_{XS} + r_{TM} v_M} \tag{2.18}$$

$$U'_M = \frac{r_{TM} v_M}{r_{G'} / Y_{XS} + r_{TM} v_M} \tag{2.19}$$

$$v_R = \frac{r_{G'}}{\mu^m_{G'}} \tag{2.20}$$

2.2.2.4 Modeling of Recombinant Bacterial Growth

There are two types of cells in a recombinant culture, plasmid-carrying cells (X^+) and plasmid-free cells (X^-), which often exhibit significantly different growth rate behavior after induction of the culture. To model growth in a recombinant culture, Immanaka and Aiba (1981) suggested the following equations for the growth rate of plasmid-harboring and plasmid-free cells.

$$\frac{dX^{+}}{dt} = \mu^{+} X^{+} (1-\theta) \quad (2.21)$$

$$\frac{dX^{-}}{dt} = \mu^{-} X^{-} + \mu^{+} X^{+} \theta \quad (2.22)$$

In which θ is the probability of the plasmid-carrying cells converting to plasmid-free cells. This parameter accounts for the random generation of plasmid-free cells from plasmid-carrying cells. The population of plasmid-free cells then increases both from the growth of plasmid-free cells and from the growth of plasmid-harboring cells which have lost their plasmids during replication.

2.2.2 Product Formation Kinetics in Recombinant Bacterial Cultures

Microbial bioproducts can be broadly categorized as *primary*, *secondary* and *recombinant*. The simplest type of kinetic models are for primary products of cell metabolism, in which there is a direct connection between the product formation, substrate consumption and bacterial growth (*growth associated products*). This happens for example in the case of ethanol production from yeast. The rate of a primary product formation is proportional to biomass growth rate and can be written as:

$$r_P = Y_{PX}\ r_X \tag{2.23}$$

where Y_{PX} is a constant often called product *yield*. On the other hand, secondary products often do not appear during the growth phase of the microorganism. Penicillin, for example, appears late in batch fermentation. Therefore, secondary products are often called *non-growth associated products*. The rate of a secondary product formation is proportional to biomass concentration and can be written as:

$$r_P = \beta\ X \tag{2.24}$$

where β is the secondary product formation constant. Recombinant product formation is often modeled by an intermediate kinetic model between primary and secondary products. The simplest model for recombinant protein formation is based on the Luedeking and Piret (1959) kinetic equation which is stated as:

$$r_P = Y_{PX}\ r_X + \beta\ X \tag{2.25}$$

Equation 2.25 shows the product formation depends on both growth and biomass concentration.

2.3 Ideal Reactors

Ideal reactor models, which will be discussed shortly in more detail, are developed mainly on the assumption of an ideal flow pattern in the reactors. The flow pattern in the reactors is assumed to be completely *mixed* and homogeneous (*mixed reactors*) or uniformly segregated and *plug flow*. Numerous studies have shown that ideal reactor models predict fairly well the actual industrial reactors. In what follows, a summary of four of the most important models along with their mathematical performance equations are presented. The performance equations are derived for constant temperature and constant density (liquid density) reactions which are common in many biochemical processes. The starting point for all modeling equations (performance equations) is the material balance equation based on reactants or products.

$$\begin{pmatrix} Rate\ \ of \\ Accumulation \end{pmatrix}_t = \begin{pmatrix} Rate\ \ of \\ Input \end{pmatrix}_t - \begin{pmatrix} Rate\ \ of \\ Output \end{pmatrix}_t + \begin{pmatrix} Rate\ \ of \\ \Pr oduction \end{pmatrix}_t \qquad (2.26)$$

2.3.1 Batch Reactor

In a batch reactor (Figure 2.9) the reactants are initially charged into the reactor and left to react for a certain period of time. The reaction can be either a pure chemical reaction or a biochemical one. The content of the reactor is kept mixed in order to have a uniform composition throughout the reactor. The process is *unsteady state* (changes with time) and the reactor content is discharged at the end of the process. The performance equation of a batch reactor can be derived for the substrate from a material balance as follows (Levenspiel, 1972).

$$\frac{dS}{dt} = -r_S \tag{2.27}$$

in which t is the time, S is the substrate concentrations and r_S is the rate of substrate consumption. Solving Equation 2.27 will give the concentration of S at any desired time. If the temperature of the reactor changes, an energy balance will also be necessary for solving the performance equation of the batch reactor.

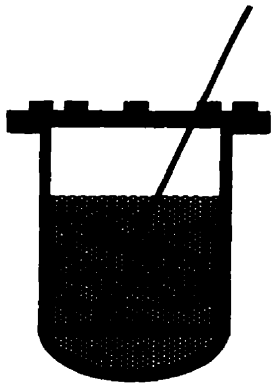


Figure 2.9. Schematic diagram of a batch reactor.

2.3.2. *Fedbatch* Reactors

In a *fedbatch (semi-batch)* reactor an initial amount of reactant or medium is charged in the reactor (*initial charge*) which is kept well mixed. Then more reactants or medium are added to the reactor as the reaction proceeds (Figure 2.10). The timing and the amount of additional reactant are determined mainly based on the concentration and the volume of the reactants in the reactor. The process is unsteady state and the contents of the reactor need to be discharged after a certain period of time. Fedbatch processes are often employed in fermentation to minimize substrate inhibition and overflow metabolism, or to avoid oxygen insufficiency and possible overheating (resulting from high biomass growth). Fedbatch operation allows more direct control over the biomass growth rate by adjusting the feed flow rate to avoid both substrate inhibition and

oxygen insufficiency. It is also possible to use different media for the initial charge and the feed in a fedbatch process, i.e. using a medium with low substrate concentration as the initial charge to facilitate bacterial growth and a very concentrated solution as the feed to lower the operating cost and to increase productivity.

Batch fermentations at very high substrate concentrations often result in a very poor bacterial biomass yield because of substrate inhibition, oxygen insufficiency, and the generation of toxic metabolic byproducts such as acetic acid (Luli & Strohl, 1990; Nailay et al., 1994; Naubauer, et al., 1995). For many microorganisms, the substrate inhibition or overflow metabolism occurs at low substrate concentrations of a few grams per liter. Nevertheless the feed concentration is often selected as high as possible to reduce processing costs. The fedbatch is a good choice to accommodate both these constraints.

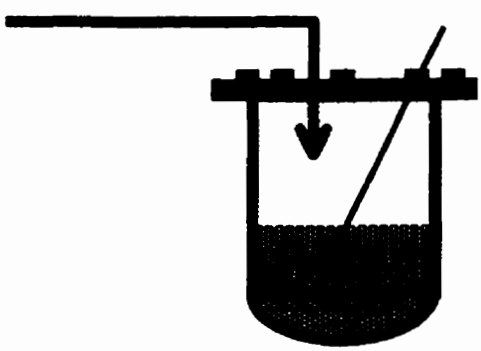


Figure 2.10- Schematic diagram of a fedbatch reactor

The performance equation of a fedbatch bioreactor can be derived from the material balance equations (Equation 2.26) for the biomass and substrate as follows:

$$\frac{dX}{dt} = r_X - \frac{F\,X}{V} \tag{2.28}$$

$$\frac{dS}{dt} = \left(\frac{F(S_F - S)}{V} \right) - r_S \tag{2.29}$$

Here F is the flow rate of fresh media, S_F is the concentration of substrate in the feed, and V is the volume of fluid in the reactor. The volume of fluid in the reactor changes with the addition of feed, and F need not be a constant.

2.3.3. Continuous Flow Stirred Tank Reactor (CFSTR)

The Continuous Flow Stirred Tank Reactor (CFSTR) is an ideal steady state reactor. The reactants are charged continuously into the stirred tank and the product is continuously discharged from the reactor (Figure 2.11). Thorough mixing keeps a uniform concentration throughout the reactor. The performance equation of a

single CFSTR can be obtained from the steady state version of the material balance equation (Equation 2.25), which yields for the substrate:

$$\frac{V}{F} = \frac{S_F - S}{r_S} \qquad (2.30)$$

Here S is the substrate concentration in the output product and S_F is the substrate concentration in the feed.

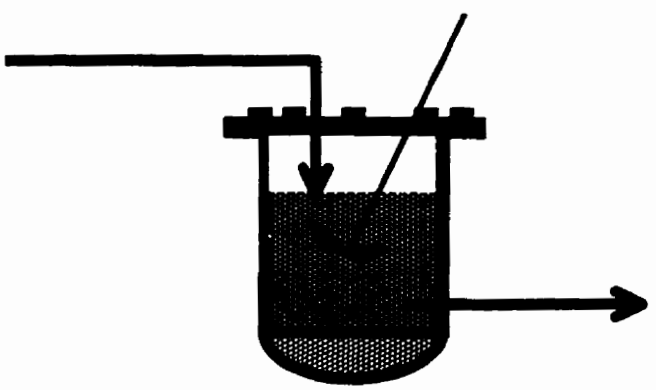


Figure 2.11- Schematic diagram of a single stage CFSTR

2.3.3.1 Two Stage CFSTRs in Fermentation of Recombinant Bacteria

The genetic instability of recombinant microorganisms is one of the most important problems in the continuous fermentation process. When cloned genes are highly expressed, the growth rate of plasmid-harboring cells are reduced and the non-productive cells easily become predominant in the total cell population. A number of research groups have found that separation of host cell growth and gene expression during fermentation is essential for effectively utilizing the limited biosynthetic potential of these microorganisms (Siegel and Ryu, 1985, Miao and Kompala, 1993; Park and Seo, 1989, Zhang et al.; 1993). This concept is the foundation of using a two stage CFSTR fermentation (separate growth and production stages) and was first proposed by Siegel and Ryu (1985). They used a recombinant bacteria with a *temperature sensitive-switching gene* (*pPLc23-trpA*) to carry out their experiments. The first stage was operated at low temperature to suppress product formation (*growth stage*), while the second stage was at higher temperature to induce the product of the cloned gene. Although most of the feed is required in the first stage to support active cell growth, a portion of feed is provided to the second stage to keep the substrate concentration

high enough for the optimum product formation. Figure 2.12 shows the schematic diagram of the two stage CFSTR first suggested by Siegel and Ryu (1985).

It is worth noting that some growth in the second stage is necessary to maximize the gene expression and the overall productivity (Park et al., 1990). In experimental studies of instability of the plasmid, Siegel and Ryu found that this two stage fermentation significantly increased the productivity and culture stability (Seigel and Ryu, 1985; Lee et al., 1988; Park et al., 1990). This idea has since been used to study many other recombinant bacteria with different regulated genes (Hortacsu and Ryu, 1990; Miao and Kompala, 1992; Seo and Bailey, 1986; Zhang et al., 1994).

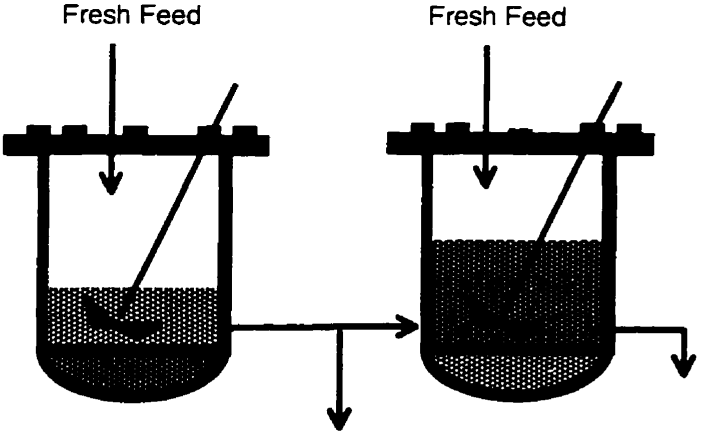


Figure 2.12. Schematic diagram of a two stage CFSTR (Siegel and Ryu, 1985).

2.3.4. Plug Flow Reactors

The other ideal steady state reactor is called *plug flow* reactor. The reactants are charged continuously into one end of a pipe and the product is continuously discharged from the other end (Figure 2.13). The flow pattern in the ideal reactor is *plug or piston* flow with no back mixing. The performance equation of a single plug flow reactor can be obtained from the material balance (Equation 2.26) which for the substrate yields:

$$\frac{V}{F} = \int_{S_F}^{S} \frac{-dS}{r_S} \tag{2.31}$$

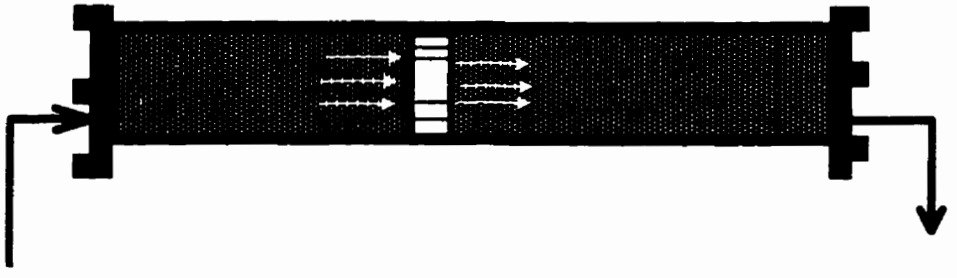


Figure 2.13. Schematic diagram of a plug flow reactor.

2.3.5. Reactors for Biochemical Processes *(Bioreactors)*

The choice of a reactor to carry out a reaction, whether a chemical reaction or a fermentation, depends on many factors such as cost of equipment, scale of production, stability and safety, and time span of the production. However the most important single factor is the reaction kinetics. The wrong choice of reactor type can increase the cost of reactor construction and operation hundreds of times. For example, the volume of a CFSTR reactor needed for up to 99% conversion of a first order reaction (or even zero and second order) is as much as 100 times more than a plug flow reactor (Levenspiel, 1972).

The nature of many biochemical reactions resembles *autocatalytic* reactions which strongly favors the use of CFSTRs and not plug flow reactors (Levenspiel, 1972). The use of plug flow reactors in fermentation processes is also very limited because of special operational problems such as handling two phase flow (for aerobic processes) and controlling both the pH and temperature along the reactor length.

2.4. Simulation and Optimization of Recombinant Bacteria Fermentation

A simple empirical equation for protein expression in recombinant cultures was first suggested by Ollis and Chang (1982). The equation was based on the Leudeking and Piret (1959) equation for product formation as a function of both bacterial growth rate and biomass concentration.

Lee and Bailey (1984a) suggested a mechanistic model for the bacterial growth and expression of a cloned gene in recombinant *E. coli*. The model, which was based on the molecular mechanism of expression in *E. coli*, employed a number of rate equations for the plasmid replication, transcription and translation to simulate the cloned gene expression. It was used to study the influence of growth rate on plasmid content and cloned gene expression in *E. coli*. Their results showed that the model correctly simulates the decrease in plasmid content with increase in growth rate. It also predicted the overall trends of the experimental data for the cloned gene expression in the bacteria.

Later Lee and Bailey (1984b) presented a genetically structured model for the *lac* promoter function in *E. coli*. The model was formulated based upon the known molecular interactions which describe regulation of expression of the *lac* operon in the *E. coli* chromosome. The cloned gene product formation was formulated based upon the transcription and translation efficiencies and was expressed as:

$$\frac{dp}{dt} = k_q \xi\,(mRNA) - K_e\, p - \mu\, p \qquad (2.32)$$

$$\frac{d(mRNA)}{dt} = k_p \eta G - K_d\,(mRNA) - \mu\,(mRNA) \qquad (2.33)$$

in which ξ and η were the transcription and translation efficiencies, and k_q, K_e, k_p and K_d are constants. Several other biokinetic models for the expression of cloned genes in recombinant bacteria with a regulated gene were developed based upon similar equations for product formation in recombinant cultures (Seo and Bailey, 1985; Miao and Ramkrishna, 1992).

Lee et al. (1985) presented a formulation for product formation in an unstable culture. The cell population was characterized by three different genotypes according to the absence and presence of plasmids (*segregational instability*) and of active cloned genes (*structural instability*). Empirical factors to account for inhibition of growth due to the presence of plasmids or product of cloned gene were assigned to the corresponding strains to simulate the process. They extended the kinetic model from batch to a single stage continuous process and theoretically studied the effects of certain process and host-vector parameters on the productivity of the system. Their simulation results indicated that, the apparent stability of the genetically engineered cell was determined not only by the intrinsic instability parameters associated with plasmid segregation and mutation but also by product synthesis activity of the cells and by the reactor operating conditions such as the dilution rate in a continuous process.

Seo and Bailey (1985) presented experimental results for continuous culture of a recombinant *E. coli*. The continuous culture fermentation had been conducted in a single stage bioreactor using *E. coli* HB101 (pPM246 plasmid). Based on the experimental data,

they suggested the existence of an optimum dilution rate for maximum plasmid and gene expression. No evidence of plasmid instability was observed during the fermentation.

The modeling of recombinant *E. coli* fermentations was first extended from batch and single stage continuous fermentation to two stage continuous fermentation process by Siegel and Ryu (1985). As was mentioned before, they suggested a two stage continuous process for fermentation of recombinant bacteria with regulated genes; a first stage for the bacterial growth and a second stage for the production of cloned gene. Part of the feed was distributed to the second stage to keep the concentration of the substrate high enough for optimum protein expression (Section 2.3.3.1).

In 1987 Lee et al. studied the dynamics of heterogeneous cell populations in the continuous culture and reported that recombinant cells could be maintained for significantly longer periods in the two stage system compared to a single stage system. Their mathematical modeling predicted reasonably well the general trends in the experimental data, although further refinement of the model was recommended.

Park et al. (1990) experimentally studied the effect of cell growth rate on the performance of a two stage continuous culture fermentation. As a part of process optimization, they focused on the influence of the specific growth rate in the second stage on overall productivity of the system. Their data for fermentation of a temperature sensitive recombinant *E. coli* showed that there is an optimum specific growth rate for maximizing the productivity of the process.

The use of a suicidal gene for maintaining plasmid stability in continuous culture was reported by Wu et al. (1994). In this system, plasmid bearing cells produce a repressor in excess which prevents transcription of a killer gene. Upon loss of plasmid, the repressor decays and the killer protein is produced which rapidly kills the cell. Their experimental results from single stage continuous fermentation showed a significant improvement in the maintenance of plasmid stability. However, it was found that the single stage system is extremely sensitive to contamination or appearance of plasmid-free cells mainly because of the low growth rate of induced cells.

A new mechanistic biokinetic model for growth and foreign protein expression in *E. coli* with a T7 promoter, was first suggested by Miao and Kompala (1992). The bacteria contained plasmids with a bacteriophage promoter (T7 promoter). In the model, the biomass was divided into two sub-populations, the plasmid containing and plasmid-free cells. The two sub-populations were assigned different growth rates and their competition was simulated. The model parameters were determined from batch cultures. Later, Miao and Kompala (1993) successfully used the model to simulate a two stage continuous fermentation of a recombinant *E. coli*. All the experiments were conducted at the low substrate concentration of 2 g/L.

There is also a large number of studies on the fedbatch fermentation of recombinant bacteria, however, most of these studies are dealing with fedbatch problems at high biomass concentrations (Bailey et al., 1987; Yee and Blanch, 1993; Jong et al., 1995; Panda et al. 1995).

Zabriskie et al. (1987) studied the effects of feeding strategies prior to induction of expression of a recombinant malaria antigen in *E. coli*. They maintained the specific growth rate in the fedbatch

process at 0.1 to 0.5 h^{-1} by adjusting the feeding profile. The experimental data showed that the feeding profile can limit the growth rate without reducing the expression levels. They suggested using the feeding profile to optimize fedbatch fermentations of recombinant bacteria to manage the oxygen demand and catabolic byproducts.

Park et al. (1989) theoretically studied the optimization of both fedbatch and continuous two stage fermentation of a recombinant *E. coli* with an environmentally regulated gene. They presented modeling equations and analytical solutions for calculation of the performance of the fedbatch and two stage continuous fermentation processes based on the Kim and Ryu (1984) biokinetics for bacterial growth and protein expression. Analytical solution of the performance equations was made possible by assuming negligible plasmid instability in both processes. In addition, they assumed a negligible volume change for the fedbatch process by assuming a highly concentrated feed stream. The simulation results indicated the existence of an optimum induction time for the fedbatch process and a best combination of the dilution rates for the two stage continuous process.

It can be concluded from the above review that most of the simulation or optimization studies of recombinant bacteria fermentation in fedbatch or two stage continuous process focused on finding the effect of individual parameters on the process productivity. A study on simultaneous optimization of several fermentation parameters should provide both a better understanding of the behavior of these processes and result in higher overall productivities.

2.5. Bioreactor Scale up

Correct translation of data from a small size bioreactor (model) to a large scale one (prototype) is essential for achieving the same production performance. Two different conditions are often mentioned to ensure similarity between the lab scale (model) and large scale (prototype) operations. These conditions are:

1. Geometric similarity, requiring the model and prototype to be the same shape while all linear dimensions of the prototype are related to the model by a constant scale factor.

2. Dynamic similarity, which requires the same ratio of velocities at the corresponding positions, as well as the same ratio of forces acting on the elements of fluid.

It is often easy to meet the geometric similarity requirement during the scale-up, but it is very difficult, if not impossible, to meet the dynamic similarity requirement. There are several forces acting on a fluid element inside the fermenter, and each varies differently with the bioreactor size and the agitation speed. For instance, the viscous force is proportional to the square of the impeller size while the gravity force is proportional to the cube of the impeller size (Lee, 1992). Therefore the dynamic similarity requirement is met in many cases by choosing one important fermentation parameter as the scale up-criterion. Most often, a *constant volumetric mass transfer* rate *or a constant shear stress* are employed as the criterion in scale up of bioreactors. In general, the volumetric mass transfer in agitated bioreactors is proportional to the power consumption per unit volume of the fluid. This scale up criterion can be expressed as a function of fluid density, size of the impeller, and mixing speed as follows (Lee, 1992):

$$\left(\frac{\rho N^3}{d_I}\right)_{model} = \left(\frac{\rho N^3}{d_I}\right)_{prototype} \qquad (2.34)$$

where ρ is the fluid density, N is the impeller speed and d_I is impeller diameter.

In the case where microorganisms are shear sensitive and the agitation generated shear is able to damage or kill the microorganism, then a constant shear rate is employed as the scale up criterion. Due to the complex nature of the fluid dynamics around the blade, it is very difficult to accurately determine the shear rate in agitated tanks. There are several empirical correlations for estimating the shear rate in agitated tanks. The following scale up criterion is based on the Boven (1986) empirical correlation for estimating shear rate in agitated tanks.

$$\left(N\left(\frac{d_I}{d_T}\right)^{0.3}\left(\frac{d_W}{d_I}\right)^{-1}\right)_{model} = \left(N\left(\frac{d_I}{d_T}\right)^{0.3}\left(\frac{d_W}{d_I}\right)^{-1}\right)_{prototype} \qquad (2.35)$$

where d_T and d_W are tank diameter and impeller width respectively.

CHAPTER THREE - SIMULATION AND OPTIMIZATION

This chapter briefly covers the methods used in the theoretical part of this research.

3.1. Modeling and Optimization of Recombinant *E. coli* (BL21DE3) Fermentation in Fedbatch and Two Stage CFSTR Process

3.1.1. Kinetic Model of Cell Growth and Protein Production

The kinetic equations of cell growth and foreign protein production are taken from the study of Miao and Kompala (1992) for *E. coli* BL21DE3. The biomass production is modeled by:

$$\frac{dX^{+}}{dt} = \mu^{+} X^{+} (1-\theta) \qquad (3.1)$$

$$\frac{dX^{-}}{dt} = \mu^{-}_{t} X^{-} + \mu^{+} X^{+} \theta \qquad (3.2)$$

where X^+ and X^- are the biomass concentration of plasmid bearing and plasmid free cells, and θ is the segregation coefficient. The specific growth rates are predicted by the Monod kinetic model which has been modified to account for inducer growth rate reduction effects:

$$\mu^+ = \frac{\mu_m^+ S}{K_S^+ + S}\left(1 - \frac{k_1 I_i^n}{k_i + I_i^n}\right) \tag{3.3}$$

$$\mu^- = \frac{\mu_m^- S}{K_S^- + S}\left(1 - \frac{k_1' I_i'^n}{k' + I_i'^n}\right) \tag{3.4}$$

in which μ^+ and μ^- designate the specific growth rates of plasmid harboring and plasmid free cells, I_i and I_i' denote the intracellular IPTG concentration, S is the extracellular glucose concentration, K^+_S and K^-_S are Monod saturation constants, k_i is the saturation constant, and k_1', k_1 and n are adjustable parameters. The intracellular, inducer concentration is calculated from:

$$\frac{dI_i}{dt} = a\left(\frac{I_e}{B + I_e}\right)\left(\frac{\mu}{\mu + k_\mu}\right) - \mu I_i \tag{3.5}$$

where I_e is the bulk concentration of inducer, a and B are pH dependent constants, μ is the specific growth, and k_μ is the saturation constant. The intracellular foreign protein concentration (p) is determined by:

$$\frac{dp}{dt} = \alpha G_p f_{Inducer} f_{Growth} f_{Inhibition} - k_d p - \mu p \qquad (3.6)$$

where α is the protein synthesis rate constant and G_p is the plasmid concentration; f_{Growth}, $f_{Inducer}$ and $f_{Inhibition}$ are functions to account for the effect of bacterial growth, inducer concentration and protein self inhibition on protein synthesis rate; and k_d is the protein decay rate constant. The inducer effect on the protein synthesis is expressed in a similar equation as the inducer effect on the growth rate:

$$f_{Inducer} = \left(1 + \frac{k_2 I_i^n}{k_i + I_i^n}\right) \qquad (3.7)$$

The formation of recombinant proteins often follows an intermediate path between primary products (growth associated) and

secondary product (non-growth associated). Therefore the growth dependent function can be expressed as:

$$f_{Growth} = \left(\mu^{*} + \beta\right) \tag{3.8}$$

where β accounts for the non-growth associated protein synthesis. The protein self-inhibition function ($f_{Inhibition}$) is expressed by:

$$f_{Inhibition} = \left(1 - \frac{p}{p_m}\right) \tag{3.9}$$

where p_m is the hypothetical maximum foreign protein concentration. Substituting these functions back into Equation 3.6 gives:

$$\frac{dp}{dt} = \alpha G_p \left(1 + \frac{k_2 I_i^n}{k_1 + I_i^n}\right)\left(\mu^{*} + \beta\right)\left(1 - \frac{p}{p_m}\right) - (\mu^{*} + k_d)p \tag{3.10}$$

The plasmid concentration is calculated from:

$$\frac{dG_p}{dt} = k_3 - (k_4 + \mu^{*})G_p \tag{3.11}$$

where k_3 and k_4 are replication and degradation constants. Table 3.1 lists the values of the constants and parameters used in the model.

Table 3.1. Model parameters and constants (Miao and Kompala, 1992)

Parameter	Value	Parameter	Value
μ_{max}^+	0.75 h^{-1}	A	1 g/L.h
μ_{max}^-	0.76 h^{-1}	B	0.12 g/L
K_S^+	0.3 g/L	k_μ	0.02 h^{-1}
K_S^-	0.3 g/L	α	0.08 -
k_1	0.78	β	0.01 h^{-1}
k'_1	0.15	k_2	19 -
k_i	0.012 $(g/L)^n$	p_m	0.4 g/g cell
n	2.5	k_d	0.01 h^{-1}
k_3	0.27 mg/gDW.h	k_4	0.067 h^{-1}
θ	0	Y_{xs}	0.4 g cell/g gluc.

Since it is difficult to measure accurately θ (which accounts for the random generation of plasmid free cells) the plasmid instability of cultures was simulated by growth rate differences between plasmid free and harboring cells and by assuming 1 to 2 % of plasmid free cells in the initial cultures (Miao and Kompala, 1992). Therefore, in reality the simulations only indirectly account for plasmid instability.

3.1.2. Modeling of the Fedbatch Process

Figure 3.1 shows a schematic diagram of a fedbatch process and the designations used for different components in the feed and bioreactor media.

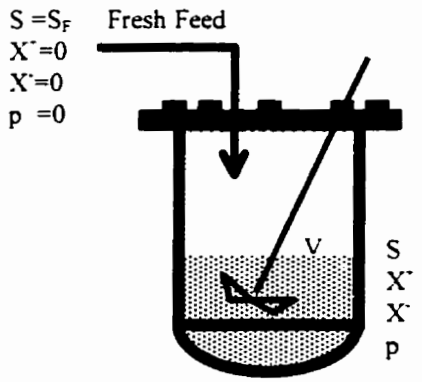


Figure 3.1. Schematic diagram of a fedbatch process.

The material balance equations for different species in the fedbatch process can be written as follows :

$$\frac{dX}{dt} = \left(r_X V - X F\right)/V \qquad (3.12)$$

$$\frac{dX}{dt} = \left(r_X V - X F\right)/V \qquad (3.13)$$

The change in substrate concentration at any time can be calculated from the following differential equation; however, in most applications

the substrate concentration is kept at a constant value (optimum value) by adjusting the flow rate of fresh media to the bioreactor.

$$\frac{dS}{dt} = (FS_{\cdot} - r_{\backslash}V - SF)/V \qquad (3.14)$$

in which r_S is the rate of consumption of substrate in the medium and is determined from the following equation:

$$r_{\backslash} = (r_{\cdot} + r_{\backslash})/Y_{\cdot} \qquad (3.15)$$

where r_X^+ and r_X^- are the growth rates of the plasmid harboring and plasmid free cells which are determined from Equations 3.1 and 3.2 respectively. The productivity of the fedbatch process can be calculated from the following equation:

$$PD = \left(\frac{Pp}{(V_{\cdot} + V_{\cdot})t} \right) \qquad (3.16)$$

in which Pp is the final enzyme concentration in the bioreactor, V_0 is the volume of the initial charge, V_t is the total volume of the process and t is the fedbatch fermentation duration.

Numerical solutions of the above equations were used to simulate the fedbatch fermentation process (Appendix C). These modeling equations can also be applied to the batch phase of the process.

3.1.3. Modeling of a Two Stage CFSTR or a Multiple CFSTR Process

Figure 3.2 shows the schematic diagram of a typical CFSTR (stage j) which is used to model two stage or multiple stage CFSTR fermentations of recombinant bacteria. The material balance for each species in stage (j) can be written as follows:

$$\frac{dX_j^+}{dt} = \left(F_{j-1}X_{j-1}^+ - F_jX_j^+\right)/V_j + r_{X^+,j} \tag{3.17}$$

$$\frac{dX_j^-}{dt} = \left(F_{j-1}X_{j-1}^- - F_jX_j^-\right)/V_j + r_{X^-,j} \tag{3.18}$$

$$\frac{dS_j}{dt} = \left(F_{j-1}S_{j-1} + F_{0,j}S_F - F_jS_j\right)/V_j + r_{S,j} \tag{3.19}$$

For the sake of simplicity, another variable is defined to designate the total concentration of the product in the medium

(P=p X^+) rather than the concentration of product per dry biomass (p). Then the material balance for the product can be written as follows:

$$\frac{dP_j}{dt} = \left(F_{j-1}P_{j-1} - F_jP_j\right)/V_j + r_{P,j} \qquad (3.20)$$

where P is the concentration of foreign protein in the medium ($P=X^+p$); r_p is the rate of production of the foreign protein and is determined from the following equation:

$$r_p = X^+ r_p + p r_{X^+} \qquad (3.21)$$

where p is the concentration of foreign protein per gram dry cells and Y_{XS} is the biomass yield per substrate consumption.

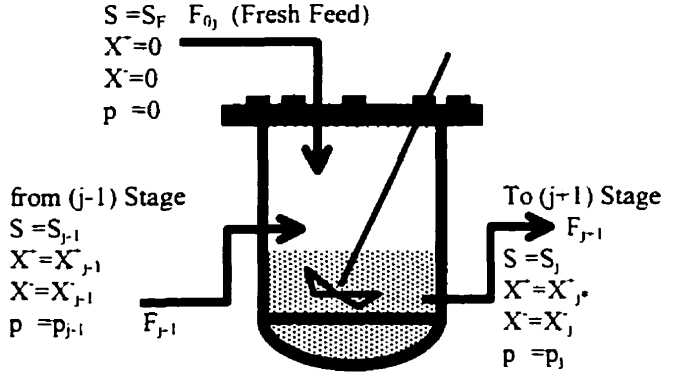


Figure 3.2- Schematic diagram of a typical CFSTR (stage j).

The productivity of the two stage CFSTR is calculated from the following equation:

$$PD = \left(\frac{F\, Pp_2}{V_1 + V_2} \right) \tag{3.22}$$

in which F is the total feed flow rate, Pp_2 is the enzyme concentration of enzyme in the outlet stream, and V_1 and V_2 are the volumes of each stages respectively.

Simultaneous numerical solution of the above system of differential equations for two or more bioreactors is used to simulate the two stage or multiple stage continuous processes (Appendix C).

3.1.4. Process Optimization

A *direct search optimization technique* known as *pattern search* was used to find the optimum fermentation parameters. This technique, which is the combination of local *exploration* and *extrapolation steps*, was first proposed by Hooke and Jeeves (1961). The basic steps in the exploration was to assume an initial value and a small increment for each of the optimization variables. The

increment does not have to be the same for all variables and is often a fraction of the distance between the lower and upper boundaries of the variable. Then the improvement in the objective function (cost, productivity, etc.) is examined for moving each of the variables by that increment in both directions. If the objective function improves for one of these perturbations, the value of the variable is set to the new explored value (old value plus or minus the increment). When repeated for all variables, the procedure is called an *exploratory* step. The values of increments are kept constant until no further improvement results from the exploratory step, then each increment is divided by half. The next step in the optimization is the *extrapolation*. A newer value for each of the variables is found by extrapolating beyond the initial values and the exploration values. The successive alteration between exploration and extrapolation is continued as long as it improves the objective function.

As a first step in this study, a computer program based on the pattern search technique was prepared to perform multi-variable optimization (Turbo Pascal). The program was then used in combination with different process simulations to find out the optimum operating parameters. The simplified flow chart of the

program is presented in Figure 3.3 (see Appendix C (C.5. & C.6.) for the list of optimization computer programs).

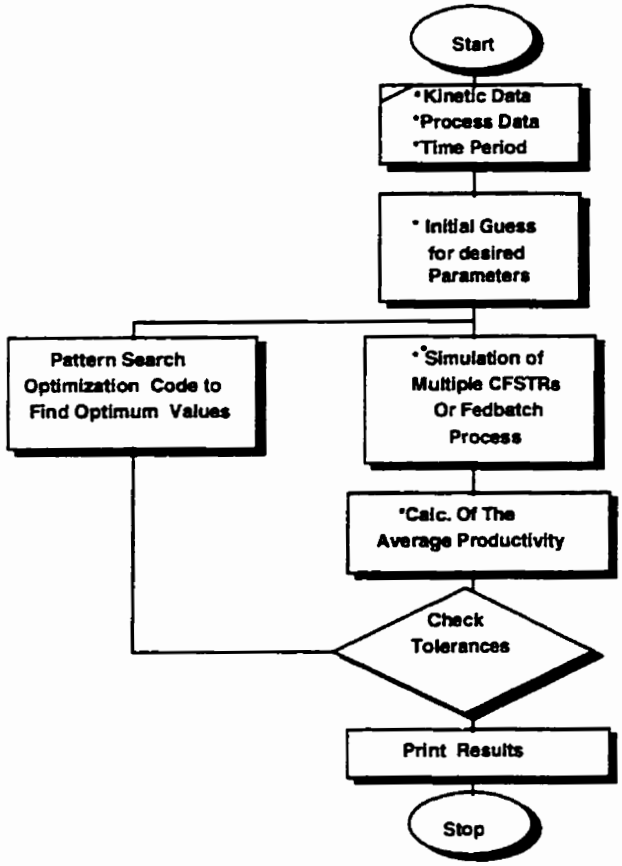


Figure 3.3. Flow diagram of the optimization process.

3.2. Modeling and Optimization of Primary and Secondary Product Formation in Multiple CFSTRs (a Generic Algorithm)

The computer program for multiple CFSTRs optimization uses numerical methods for both optimization and calculation of performance of each stage. The software accepts biokinetic equations, does necessary material balances for each species, numerically solves the material balance equations and optimizes the system under a given objective function.

After accepting the biokinetic data and equations, a system of nonlinear equations is constructed from the material balances for species in each stage. This system of nonlinear equations is then solved by the Newton method (Gottfried and Weisman, 1973) to calculate the performance of each stage (values of all parameters for each stage). A Pattern Search optimization code is used to optimize the system based on the stage performance data (Section 3.1.4). Figure 3.4 shows the flow chart of the generic algorithm for optimization of multiple CFSTRs in series.

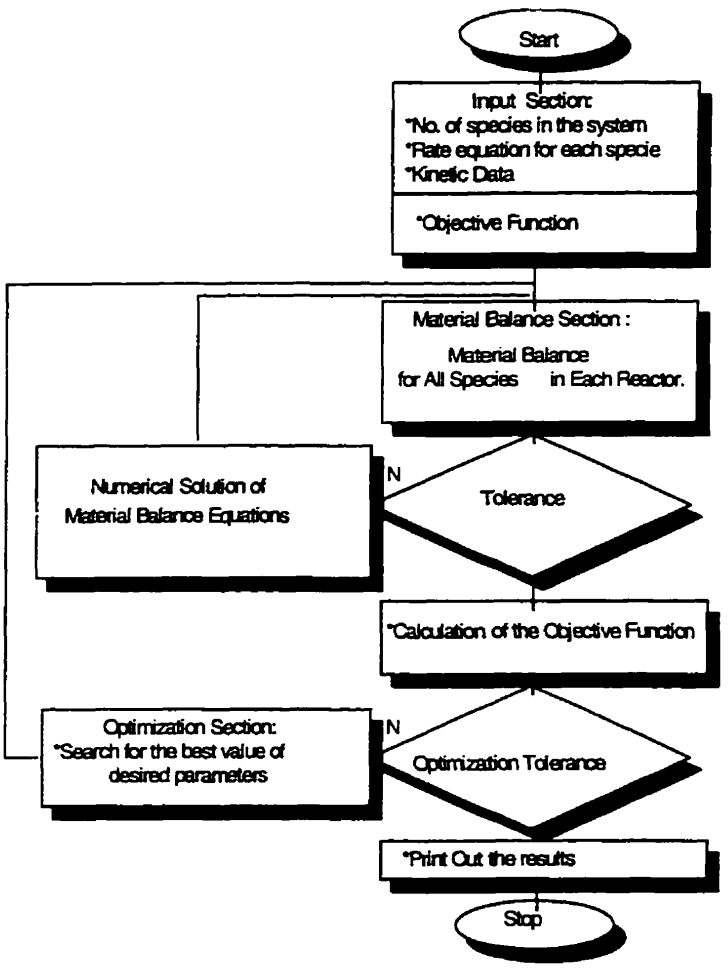


Figure 3.4. Flow chart of the generic algorithm for optimization of multiple CFSTRs fermentation.

CHAPTER FOUR- MATERIALS AND METHODS

This chapter covers the material and methods used in the experimental part of the research.

4.1. Microorganism and Medium

The microorganism used in this study was *E. coli* strain *BL21DE3* which was ordered from Inter Medico, Ontario, Canada. The stock culture of the bacteria was kept at -70 °C. The bacterium carries a plasmid with a gene for resistance to kanamycin, a target gene and a T7 promoter upstream of the target gene to control its transcription. The bacterial chromosome carries a gene for T7 RNA polymerase which is under the control of an *inducible promoter* (*lac*UV5 promoter) to control transcription of the polymerase gene. With this system the overall induction of the target gene is controlled by addition of a chemical inducer (IPTG, isopropyl-β-D-thiogalactopyranoside).

The bacteria were grown on basic M9 medium supplemented with either glucose (for 2 g/L) or a mixture of glucose and casamino

acids (50-50 % mixture, in 6 and 10 g/L tests). The composition (in g/L) of different salts in the M9 medium is as follows: Na_2HPO_4, 6; KH_2PO_4, 3; NaCl 0.5; NH_4Cl, 1; $CaCl_2$, 0.015; and $MgSO_4$, 0.25 (Appendix A1). The casamino acid was added because the M9 medium did not support high biomass growth with glucose as the only carbon source. A similar medium formulation was used by Whitney et al. (1989).

Other chemicals used in the fermentation experiments included, Antifoam "B" (BDH Inc., Toronto) to control foam formation in fedbatch and continuous operation, and NaOH and H_3PO_4 to maintain the pH.

4.2. Biomass (Total protein) and Enzyme (β-galactosidase) Measurement

Both protein and enzyme measurements were performed using sonicated cell samples. The cell sonications were performed according to the following procedure: 1) 1 mL of the ice-cold cell culture was centrifuged to separate cells, followed by re-suspension in 1 mL of Tris-HCl (0.25 M, pH 7.8) and 2) 250 µl of this solution was sonicated for 1 min using a Brunson® Sonifier 450 at 50 % load, using a micro tip and a control output of 2. A test on the sonication

time showed that there is no further change in the protein and enzyme content of the sample after about 45 seconds. The samples are kept on ice to minimize enzyme degradation.

The total protein content was assayed using a Bio-Rad dye assay (Bio-Rad Laboratories, Inc.) with bovine serum albumin (BSA) as the standard protein solution. A total of 5- 25 μl of the sonicated solution was mixed with 1 mL of the diluted dye reagent (4 times dilution) and kept at room temperature for about 5-7 minutes. The OD change (at 595 nm) of the solution was then compared against standard bovine serum albumin (BSA) OD change to calculate the protein content (BSA equivalent protein content). The biomass concentration can also be measured immediately after taking the samples by measuring the OD at 600 nm and comparing to a standard curve (Appendix A2).

The level of the enzyme activity was measured by an assay with ONPG (o-nitrophenyl-β-D-galactosidase), which is hydrolyzed by β-galactosidase to give a yellow-colored product (Maniatis, 1982). 2-5 μl of sonicated solution was mixed with 230 μl of standard ONPG solution at 37 °C. 500 μl of 1.0 M Na_2CO_3 was added to stop the reaction when a faint yellow color appeared (15 seconds to 5 minutes). This color change was then measured with a

spectrophotometer at 420 nm to determine the enzyme activity (Appendix A3).

Both the protein and enzyme measurements were often repeated twice, three times if there was too much discrepancy between the two test results. The average experimental error in both the biomass and enzyme concentration measurements was found to be around 10 % .

4.3 Determination of Plasmid-Free and Plasmid-Harboring Population

Replica plating provides a simple method for both screening and determination of the plasmid free population. In this procedure, several consecutive dilutions of the bacterial culture are prepared by diluting 100 µl of the solution with 900 µl of sterilized water (1/10 dilution). Then 100 µl of each dilution is spread over both agar plates and agar/kanamycin plates (30 µl kanamycin/mL agar solution). Following incubation at 37 °C for approximately 24 h, the number of colonies appearing on the agar and agar/kanamycin is counted and taken as the total population and plasmid-harboring population respectively. This method reproduces well if at least three

replicates are prepared and the number of colonies on the plates are around 30-300.

4.4. Batch, Fedbatch, and Two Stage Continuous Process Setup

4.4.1 Batch Setup

Batch fermentations were mainly performed to assess the ability of the modeling equations to predict the cell growth and foreign protein production. Most batch tests were carried out in 250 mL shake flasks equipped with air spargers in a water bath shaker (New Brunswick Scientific) at 37 ±0.5 °C and 200 rpm. An overnight culture of bacteria in M9 supplemented with 2 g/L glucose and 30 µg/mL kanamycin was used for inoculation of 50 or 100 mL of the fresh medium in each bioreactor (1-5 % inoculation). The culture was then left for about 20-30 minutes to allow adjustment of the bacteria to the fresh medium before the actual start of the experiment. The air flow rate was kept between 5-10 L/h.L and the pH was maintained at 7 ±0.5 by addition of 1.0 N NaOH or 1.0 N H_3PO_4.

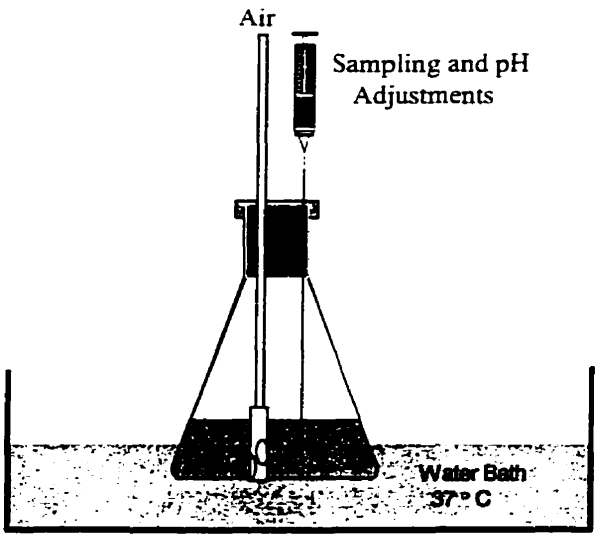


Figure 4.1. Schematic diagram of batch fermentation setup.

4.4.2. Fedbatch Setup

Fedbatch fermentations were carried out in a 2.0 L, temperature controlled fermentor with a magnetic stirrer (BioFlo® Fermentor, New Brunswick Scientific). The schematic diagram of the bioreactor setup is shown in Figure 4.2. The experiments were started by adding the initial charge to the bioreactor. An overnight culture of bacteria in M9 supplemented with 2 g/L glucose and 30 μg/mL kanamycin was then used for inoculation (3-6%). The feeding of the reactor was started when each culture reached its pre-selected biomass concentration (measured by OD at 600 nm). The feed flow rate was adjusted through two computer-interfaced peristaltic pumps (Cole-Parmer Instrument Co., Chicago, IL, 100 rpm max.).

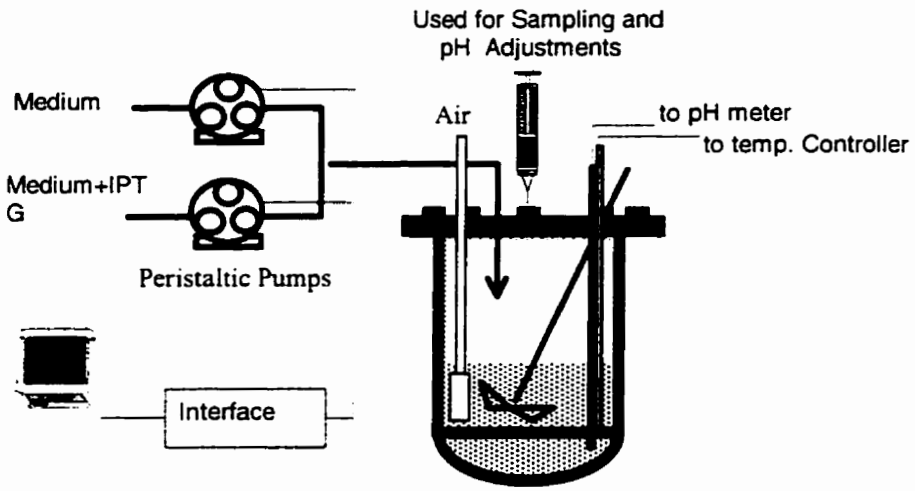


Figure 4.2. Schematic diagram of the fedbatch fermentation setup.

Two types of tubings were used for low and high flow rates: (1) PharMed® size 13 for flow rates of 0.1-3 mL/min and (2) PharMed® size 14 for flow rates of 1-15 mL/min. The computer adjusted the feed flow rates to the bioreactors according to data files produced by the process simulations (open loop control). IPTG was added to induce the synthesis of the foreign protein by switching the feed from fresh medium to fresh medium + IPTG. At the optimum induction time or any desired induction time, the computer switched the pumps from fresh medium to the medium containing IPTG. Additional IPTG was added manually at the induction time to increase its concentration in the bioreactor to the desired level. The air flow rate was kept

between 3-6 L/h.L and the pH was maintained at pH 7 ±0.5 by addition of 1.0 N NaOH or 1.0 N H_3PO_4. The fedbatch process ended when the culture volume reached 1 L (maximum operating volume).

4.4.3. Two Stage Continuous Setup

Two stage continuous experiments were carried out by using two temperature controlled fermentors in series (1 L and 2 L BioFlo® Fermentor, New Brunswick Scientific). Figure 4.3 shows the schematic bioreactor setup for the two stage continuous process, in which F_1 and F_2 refers to the feed flow rate to the first and second tank, V_1 and V_2 refer to volume of bioreactor one and two.

The inducer (IPTG) was added to the second bioreactor through the feed to that bioreactor. In addition, antifoam (Antifoam B, BDH), acid or base were manually added to the feed reservoir because of the difficulty in continuous injection of these chemicals to each bioreactor. Both bioreactors were operated initially in the batch mode. An overnight culture of bacteria in M9 supplemented with 2 g/L glucose and 30 μg/mL kanamycin was used for inoculation (5- 10%). The continuous operation started when the OD at 600 nm of culture reached about 30 to 40 % of the maximum value. Two computer-

interfaced peristaltic pumps (Cole-Parmer) were used to maintain feed flow rates, to transfer the culture from the first tank to the second tank and to transfer the product out of the second tank. Two types of tubing were used for low and high flow rates; PharMed® size 13 for flow rates of 0.1-3 mL/min and size 14 for flow rates of 1-15 mL/min. The process reached steady state after about 8-12 hours based on the biomass concentrations measured by OD at 600 nm.

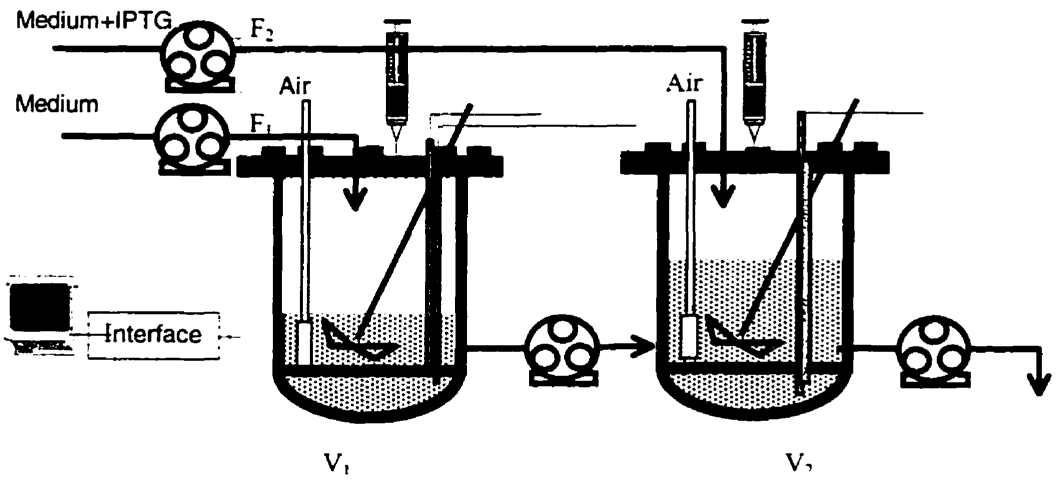


Figure 4.3. Schematic diagram of two stage CFSTR fermentation setup.

CHAPTER FIVE - RESULTS and DISCUSSION

The theoretical and experimental results from the study of batch, fedbatch and two stage CFSTR processes are presented and discussed in this chapter. Each section starts with a brief review of the modeling results and is then followed by the experimental results. Batch, fedbatch, and two stage process simulations and optimizations are based on the models presented in Chapter 3.

5.1. Optimization of Recombinant E. coli (BL21DE3) Fermentation in Fedbatch and Two Stage CFSTR Process

5.1.1. Batch Fermentation

A typical simulation of bacterial growth and product formation during batch fermentation is presented in Figure 5.1. As with many other recombinant cultures, the growth rate of plasmid-harboring bacteria reduces substantially with induction. This is evident from the comparison of the concentration of plasmid-harboring cells in the presence and absence of inducer in Figure 5.1.

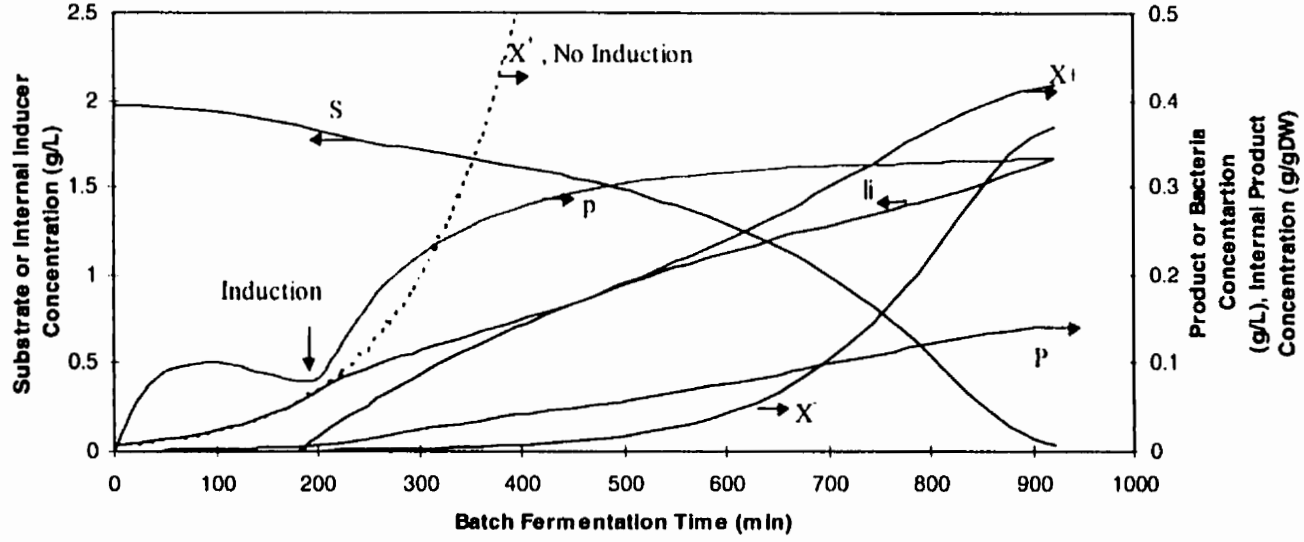


Figure 5.1. Model simulations for the substrate, biomass and product concentration in a batch fermentation of recombinant bacteria with induction. The initial substrate concentration is 2 g/L; inoculation 1%; induction at 180 min with IPTG (50 μg /ml). The small overshoot in the internal concentration of protein around 50 h is due to errors of numerical solution.

The reduction in the growth rate of recombinant bacteria which is attributed to the metabolic burden of the expression of the foreign protein depends on the strength of the expression and the concentration of the inducer (Miao and Kompala, 1992; Bailey, 1986). It should be noted that in the simulation shown, an enzyme concentration of 0.15 g/L is reached, but this value can be affected by adjustment of the induction time and/or the initial inoculum. Finding the best values of these variables to maximize enzyme production is the main topic of this thesis. The first two sets of experimental work provide data to assess how well the modeling equations predicted biomass growth and foreign protein production. The first set of experiments was designed to evaluate the modeling equations for predicting the induction effect on the biomass growth rate and the second set for predicting the enzyme production. These experiments will be discussed in more detail in the next two sections.

5.1.1.1 Inducer Concentration Effects on Biomass Growth

In these experiments, several batch cultures were induced with different IPTG concentrations of 0, 25, and 50 μg /mL and cell growth was monitored by a spectrophotometer (OD at 600 nm). The experiments were carried out in either a selective medium (M9 +

kanamycin) or non-selective medium (M9), and the results were then compared against the modeling predictions. The experimental and theoretical biomass data from these tests can be seen in Figures 5.2 and 5.3. As can be seen from both figures, the specific growth rate of bacteria significantly decreases with the increase in the inducer concentration. The model predictions show slightly more deviations from the experiments carried out in the non-selective medium, but still there is a good overall agreement between the modeling and experimental data. The bacterial expression of foreign protein finally saturates at a certain inducer concentration and no further drop in the growth rate can be observed after the saturation concentration. The saturation of expression occurs for this bacteria around an IPTG concentration of 50 μg /mL (Miao and Kompala, 1992).

5.1.1.2 Induction Time Effects on the Enzyme production

A more thorough evaluation of the modeling equations were performed by inducing several batch cultures at different times from 1 to 6 h and then comparing the final biomass and enzyme concentration in each culture against the modeling predictions. The experiments were planned with two IPTG concentration of 25 and 50

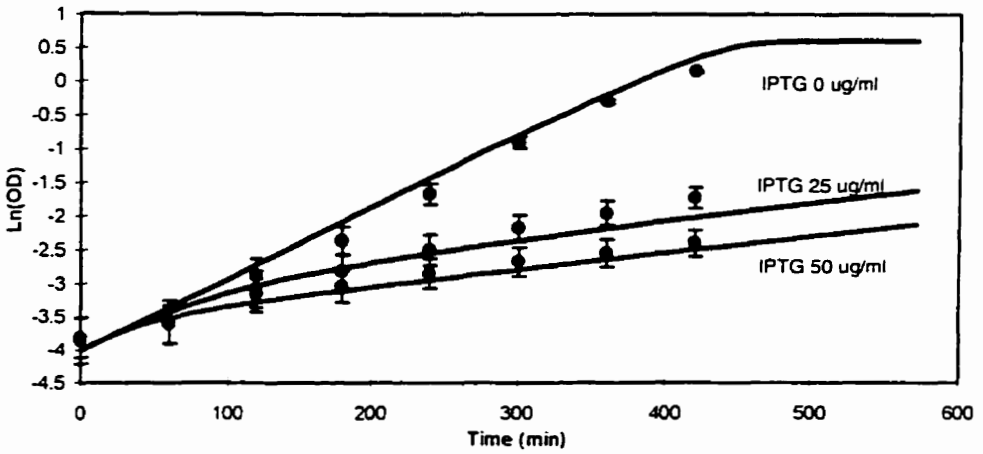


Figure 5.2. Effect of inducer concentration on the biomass growth in a selective medium (M9 2g/L in glucose + kanamycin). Three batch cultures induced at time zero with IPTG concentration of 0, 25, and 50 µg /mL and the biomass growth is monitored by measuring the OD at 600 nm. Solid lines represent model simulations.

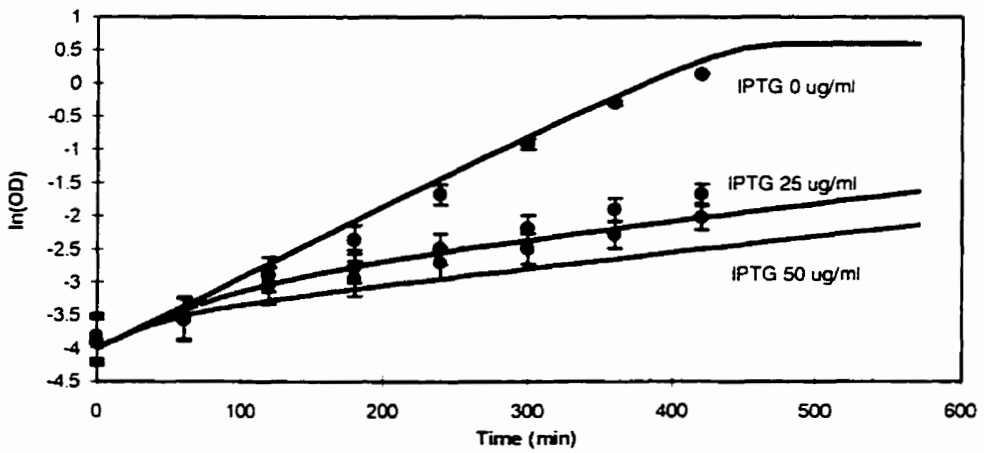


Figure 5.3. Effect of inducer concentration on the biomass growth in a non-selective medium (M9 2g/L in glucose). Three batch cultures induced at time zero with IPTG concentration of 0, 25, and 50 µg /mL and the biomass growth is monitored by measuring the OD at 600 nm. Solid lines represent model simulations.

μg /mL and the glucose concentrations of 2, 6, and 10 g/L over a 7 h fermentation period. However, preliminary tests revealed that the basic M9 medium supplemented with pure glucose (10 g/L) as the only carbon source resulted in a very poor biomass yield (≅0.15 g cell/g glucose). After several batch tests to diagnose and fix the problem, a mixture of casamino acid and glucose (50-50%) was selected as a carbon source and nutrient supplement for fermentation experiments at high concentrations. The maximum specific growth rate and the biomass yield of the bacteria with this new substrate were found to be 1.2 h^{-1} and 0.32 g cells/g substrate compared to 0.75 h^{-1} and 0.4 g/g substrate in the M9 medium supplemented with glucose alone (Appendix B). The experimental results for the final biomass concentrations in the batch cultures along with modeling predictions are presented in Figures 5.4 and 5.5. The data are for three different substrate concentrations of 2, 6, and 10 g/L with the IPTG concentrations of 25 and 50 μg/mL. Most of the experimental data demonstrate the same trends as predicted by the model, however, there are noticeable deviations for glucose

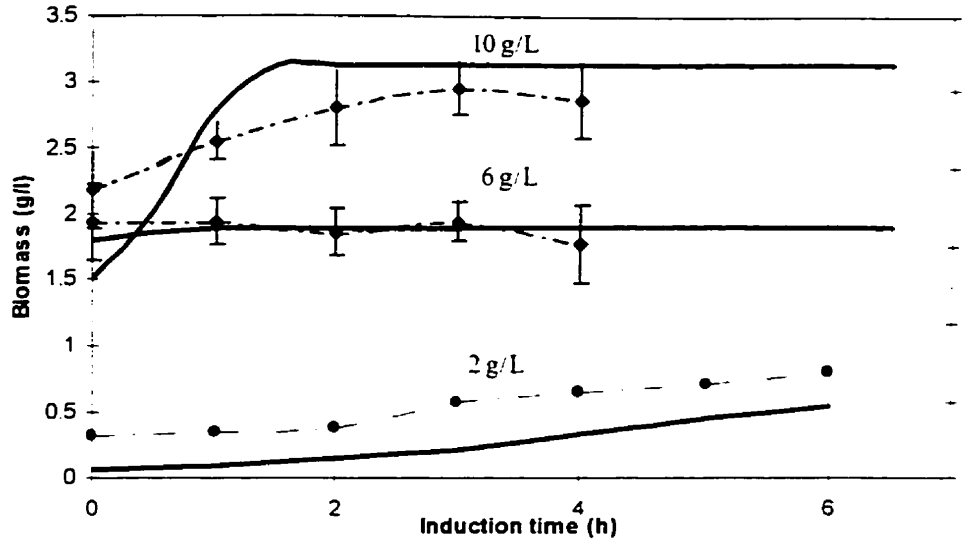


Figure 5.4. Final biomass concentrations after 7 h in batch cultures induced with IPTG (25 µg /mL) at times from 1 to 6 h. Each point represent a batch culture induced at that specific induction time and left for 7 h. Solid lines are model simulations and points represent the average of two experimental data.

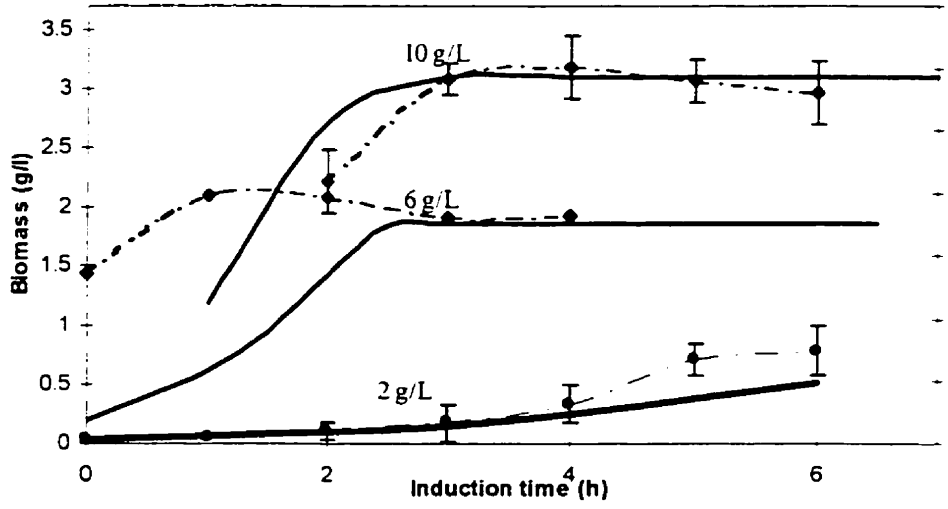


Figure 5.5. Final biomass concentrations after 7 h in batch cultures induced with IPTG (50 µg /mL) at times from 1 to 6 h. Each point represent a batch culture induced at that specific induction time and left for 7 h. Solid lines are model simulations and points represent the average of two experimental data.

concentrations of 2 g/L (IPTG 25 μg/mL) and 6 g/L (IPTG 50 μg/mL). These deviations could be due to errors in the determination of the initial inoculum. Considering the exponential nature of bacterial growth, even small deviations in the measurement of the initial inoculum can lead to significant errors in the prediction of biomass concentration after 7 hours.

The experimental results for the final enzyme concentrations in batch cultures along with the modeling predictions can be seen in Figures 5.6 and 5.7. The data show good agreement with the modeling predictions for the enzyme concentration. The enzyme concentration data also suggest that both early and late induction of the cultures results in very low enzyme activities. Early induction reduces enzyme production due to the low biomass concentration while late induction leaves no time for the expression of the enzyme.

It is also interesting to note that while the medium nutrient content increases from 2 to 10 g/L (5 times), the enzyme activity increases from about 0.1 to 1 g/L (10 times). Part of this increase is due to the increase in biomass concentration from about 0.8 to 3 g/L (≅4 times), but the remainder is due to the higher specific growth rate in the new medium, which increases the specific enzyme production.

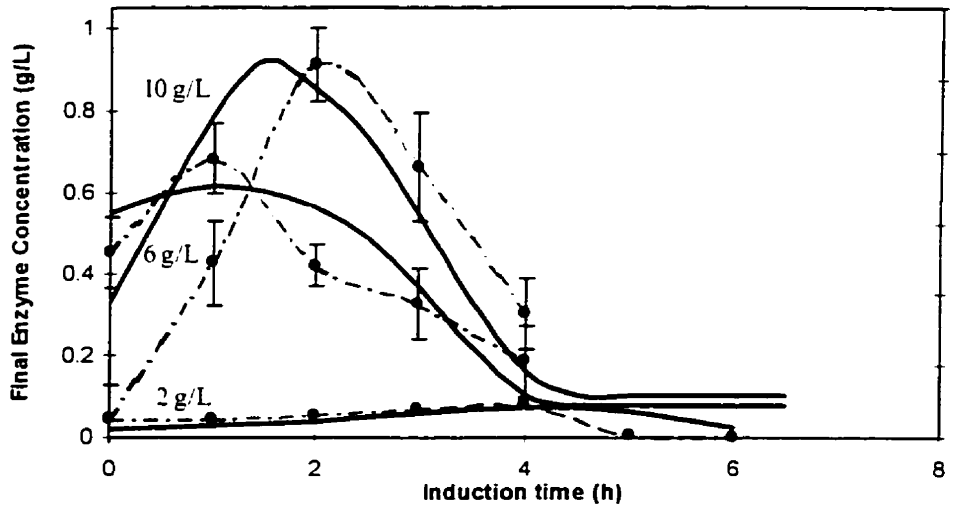


Figure 5.6. Final enzyme concentrations in batch cultures induced with IPTG (25 μg /mL). Each point represents a batch culture induced at that specific induction time. Total fermentation period was 7 h. Solid lines are model simulations and points represent the average of two experimental data.

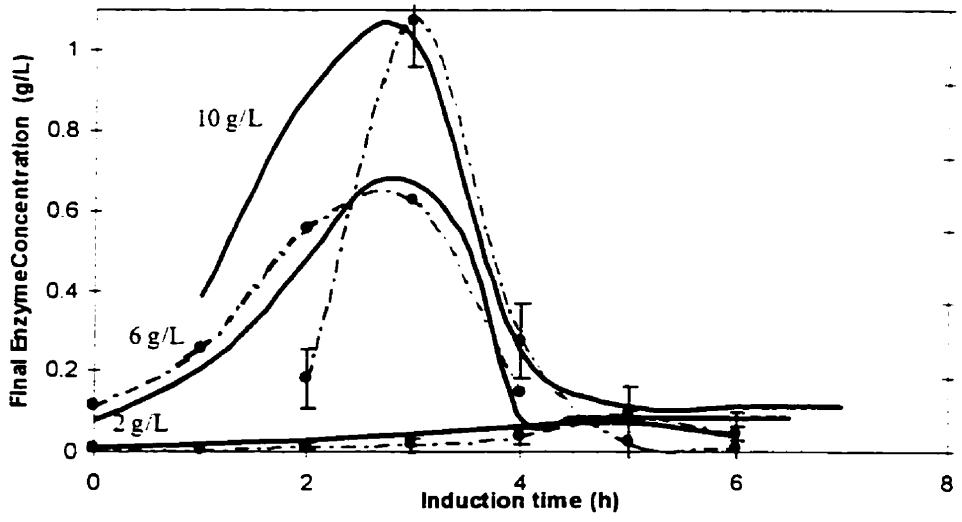


Figure 5.7. Final enzyme concentrations in batch cultures induced with IPTG (50 μg /mL). Each point represents a batch culture induced at that specific induction time. Total fermentation period was 7 h. Solid lines are model simulations and points represent the average of two experimental data

.

This increase in the enzyme activity was successfully predicted by the model simulations.

5.1.2. Fedbatch Fermentations

A typical simulation result for bacterial growth and product formation during a fedbatch fermentation can be seen in Figure 5.8. The data are for a fedbatch operation using M9 medium (2 g/ L in glucose) for both the feed and initial charge, with 20% initial charge and the induction time of 180 min (3h). Compared to the batch fermentation (Figure 5.1), the fedbatch process (Figure 5.8) finished much faster (in 600 min vs. 900 min), with a higher product concentration (0.22 in fedbatch vs. 0.14 g/L in batch) and higher plasmid stability (0.7 in fedbatch vs. 0.45 g/L plasmid-harboring cells in batch).

The productivity of a fedbatch process can be optimized by the proper selection of induction time, operating substrate concentration, and initial charge (initial batch volume). The effect of induction time and initial charge on the productivity of a fedbatch operation is presented in Figure 5.9. In fact, the proper selection of the initial charge and the induction time is crucial in obtaining high

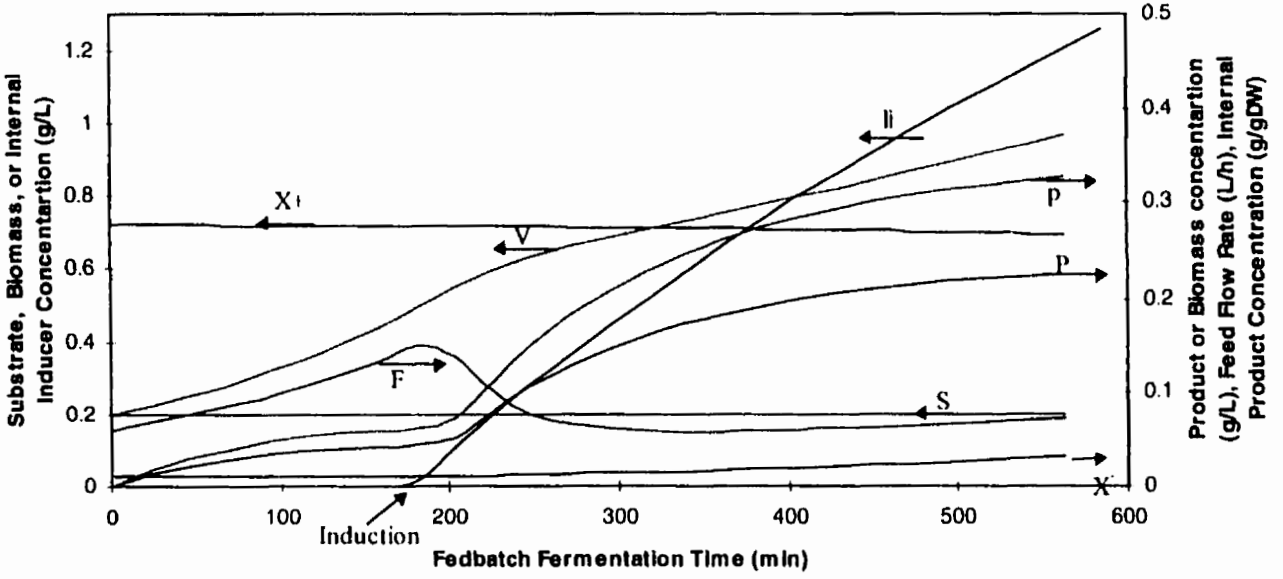


Figure 5.8. Model simulations for the substrate, biomass, and product concentrations and feed flow rate and volume changes in a fedbatch fermentation of recombinant bacteria. The data is for a fedbatch using M9 media (2 g/L) for both feed and initial charge; with 20% percent initial charge; induction time of 180 min with IPTG (50 μg /ml). The substrate concentration is maintained at 0.2 g/L during the process by adding fresh feed to the bioreactor.

productivities in a fedbatch process, as some combinations can result in almost no productivity (Figure 5.9). The substrate concentration is often kept below 2-3 g/L during fedbatch fermentation of *E. coli* to avoid overflow metabolism and excessive acetic acid formation (Luli & Strohl, 1990; Neilay et al., 1994; Naubauer et al., 1995).

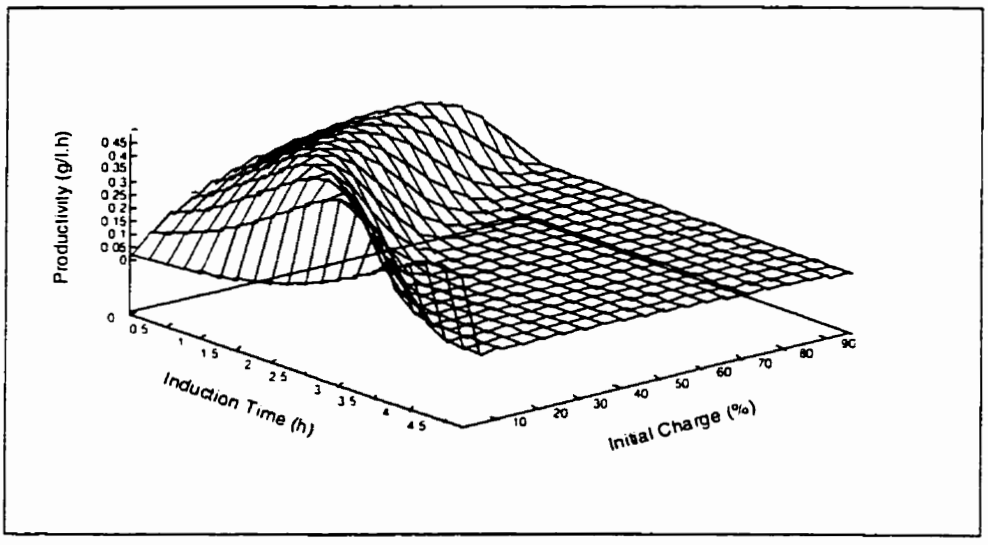


Figure 5.9. Model simulations for the effect of induction time and initial charge on the productivity of a fedbatch fermentation process. The data are for a fedbatch process using M9 medium with 50 g/L substrate as feed and 10 g/L as initial charge. The substrate concentration is maintained at 2 g/L by adding fresh feed and the induction is by IPTG (50 μg /mL).

In this investigation, and within the applied concentrations of substrate (up to 10 g/L), the optimum operating substrate concentration was found to be less than 2 g/L. The percentage of initial charge was often found to be much higher than 1-5%

inoculation which is commonly used in batch fermentations. Besides the kinetics of the cell growth, the percentage of initial charge mainly depends on the substrate concentration in both feed and initial charge.

The optimum operating parameters for a number of fedbatch operations, along with the theoretical and experimental productivities are presented in Table 5.1. The deviation between experimental and modeling data was often higher for those processes using different substrate concentrations for the feed and the initial charge such as the last two experiments recorded in Table 5.1. Part of the deviations can be due to the time lag necessary for the bacteria to adjust to the new medium at the beginning of the feeding process. However, as discussed later, the kinetics of bacterial growth in the fedbatch culture may not be the same as in the batch culture especially when using the mixed nutrient substrate. This will directly affect the accuracy of the simulation and optimization predictions. However, with the exception of the last two sets of data in Table 5.1, there is a good agreement between the modeling and experimental results.

Table 5.1. Optimum operating parameters for a number of fedbatch operations, along with the theoretical and experimental data for the final enzyme concentrations.

Substrate Concentration		IPTG	Induction	Initial	Operating Substrate	Final Enzyme Conc.		Error
Feed	Initial Charge	(μg/ml)	Time	Charge	Concentration	Theory	Experiment	(%)
(g/L)			(min)	(%)	(g/L)	(g/L)		
2	2	25	0	50	0.39	0.134	0.145	+8.1
2	2	50	0	65	0.44	0.119	0.127	+6.5
6	6	50	0	41	0.76	0.285	0.260	-8.6
10	10	25	0	23	0.55	0.538	0.466	-13.3
10	2	25	180	16	0.74	0.694	0.502	-28.0
10	2	50	234	15	0.80	0.698	0.537	-23.0

Several more fedbatch tests were carried out to ensure that the calculated values were in fact the optimum parameters. In these experiments, one parameter (induction time or percentage of the initial charge) was changed and the productivity was compared with the values predicted by the model. The experimental result of this part are presented in Figures 5.10 to 5.12 and show that the calculated values are in fact close to the experimental values. A typical biomass and enzyme concentration profile during a fedbatch experiment can be seen in Figure 5.13.

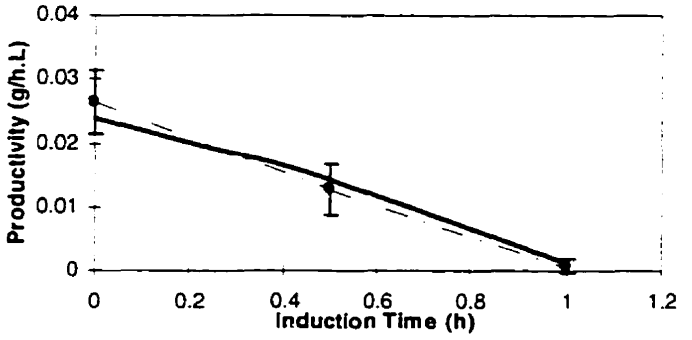


Figure 5.10. Effect of induction time on the productivity of fedbatch fermentation. The data are for a fedbatch fermentation with M9 medium (2 g/L) for both feed and initial charge, with 50% initial charge, operating substrate concentration of 0.4 g/L and IPTG of 25 μg /mL. The productivity is calculated based on the net enzyme production during the fedbatch phase.

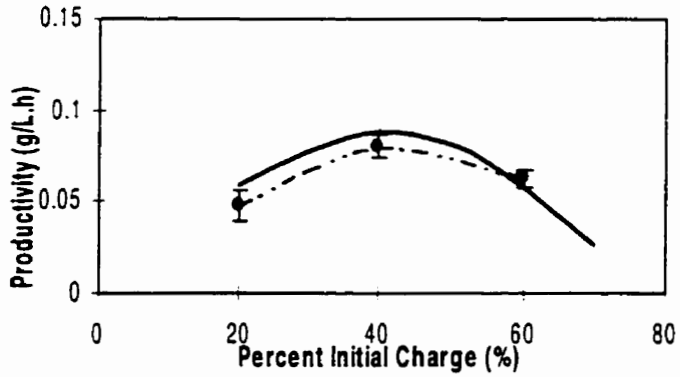


Figure 5.11. Effect of initial charge on the productivity of fedbatch fermentation. The data are for a fedbatch with 41% initial charge (optimum value), using M9 6 g/L in both feed and initial charge (IPTG 50 μg /mL). The operating substrate concentration is 0.78 g/L. The productivity is calculated based on the net enzyme production during the fedbatch phase.

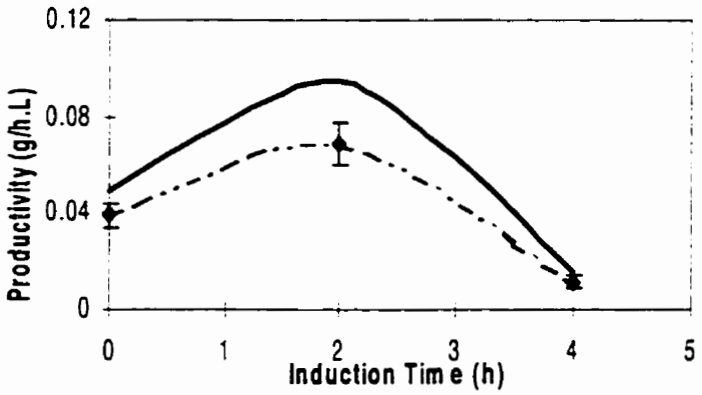


Figure 5.12. Effect of induction time on the productivity of fedbatch fermentation. The data are for a fedbatch with 16% initial charge (optimum value), using M9 10 g/L mix nutrient for feed, 2 g/L glucose for the initial charge (IPTG 25 μg /mL). The operating substrate concentration is 0.75 g/L. The productivity is calculated based on the net enzyme production during the fedbatch phase.

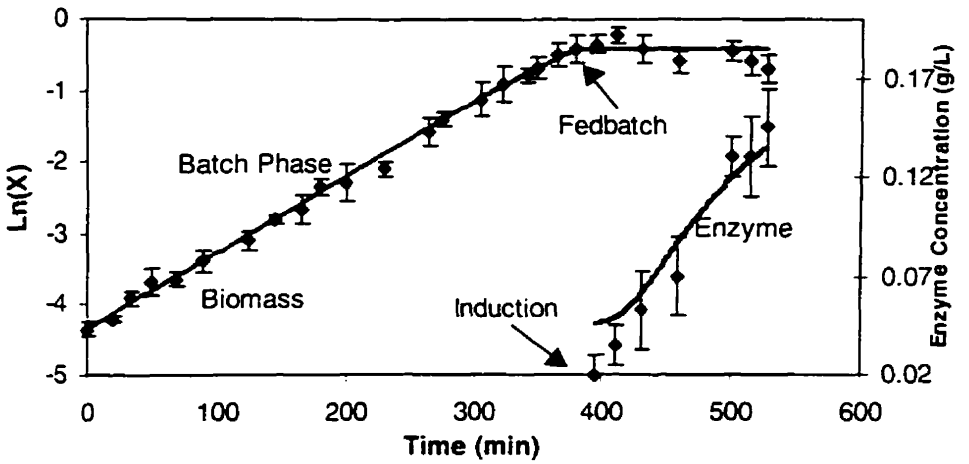


Figure 5.13- Biomass and enzyme concentration profile in a fedbatch fermentation (substrate concentartion of 2g/L in both initial charge and feed and IPTG concentration of 25 μg/mL). Solid lines represent simulation results.

5.1.3. Two Stage and Multiple CFSTR

Figures 5.14 and 5.15 show typical simulation results for the bacterial growth and product formation during *a single stage* and *a two stage* (growth and production stage) fermentation of recombinant bacteria. In a single stage continuous operation, the concentration of plasmid-harboring cells rapidly decreases within a few hours, and plasmid free cells overtake the cultures (Figure 5.14). The product disappears as the concentration of plasmid free cells decreases. However, in the two stage process, the first stage maintains a relatively stable supply of plasmid-harboring cells to the second stage. The whole process is much more stable over longer periods

of time and the product concentration is much higher than the single stage process (Figure 5.15).

It is still possible to increase further the product concentration or productivity of the two stage process by optimizing the volumes of the bioreactors and the feed flow rate to each bioreactor. Figure 5.16 shows the effect of the operating parameters of a two stage process (feed distribution and volume of reactors) on the productivity of the process. The selection of wrong operating parameters results in very poor productivities. For instance, the combination of a high feed flow rate and a small volume for the first stage results in a total cell washout from the stage with no sustainable product formation. Table 5.2 lists typical improvement in the productivity of the two stage process by adopting optimum parameters instead of equal feed and volume arrangement, which in these cases the improvements are about 35-42%.

Another study was carried out to evaluate the feasibility of using multiple CFSTR (using more than two CFSTR) in the fermentation of a recombinant microorganism. The idea of using multiple CFSTR instead of one CFSTR is well known in ordinary fermentation to produce primary products.

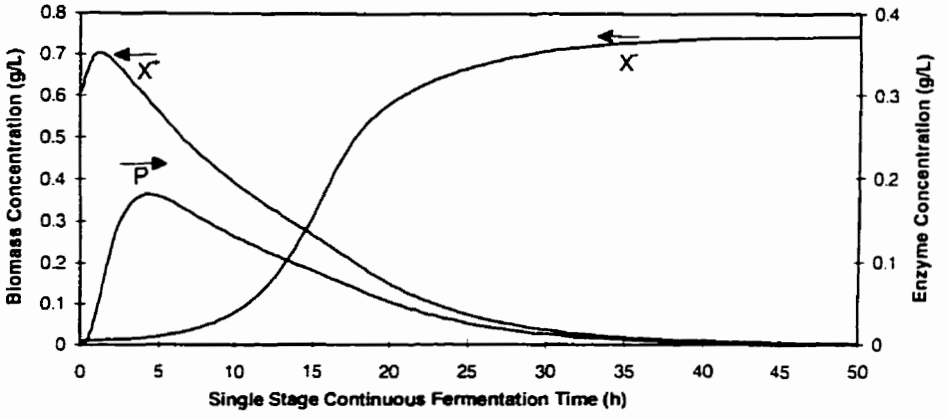


Figure 5.14. Model simulations for the biomass and product concentration during a single stage CFSTR fermentation of recombinant bacteria. Both growth and production occurs in a single stage. The substrate concentration in the feed is 2 g/L and the IPTG concentration is 50 μg /mL.

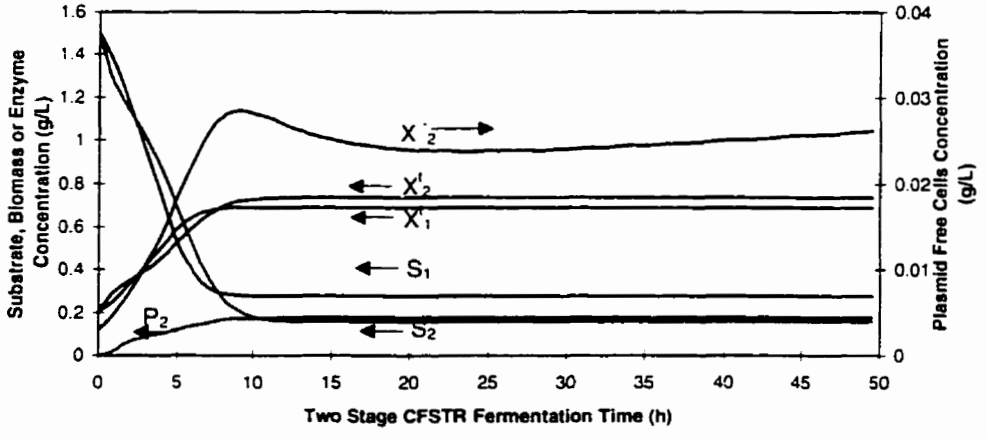


Figure 5.15. Model simulations for the substrate, biomass and product concentration during a two stage CFSTR fermentation of recombinant bacteria (one growth stage and one production stage). The substrate concentration in the feed is 2 g/L and the IPTG concentration is 50 μg/mL. The indexes of 1 and 2 designate the first and second stages respectively.

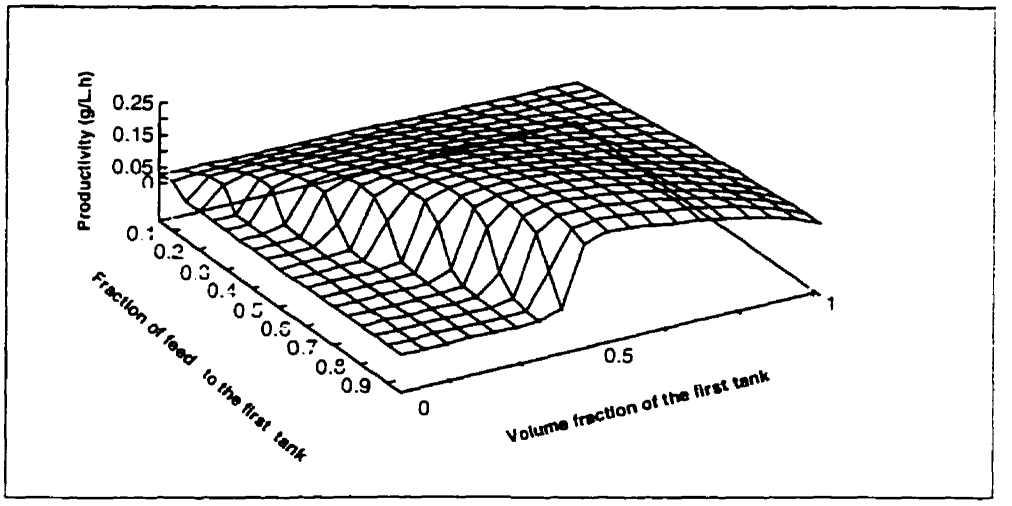


Figure 5.16. Model simulations for the effect of feed distribution and the volume of bioreactors on the productivity of two stage fermentation process. The data are for a feed substrate concentration of 10 g/L and an IPTG level of 25 µg /mL in the second stage

Table 5.2. Model simulations showing improvement in the productivity of two stage continuous process by choosing optimum parameters. The data are for a two stage CFSTR with a total volume of 1 L and total flow rate of 0.2 L/h. For each feed substrate concentrations of 2, 6, and 10 g/L, the process has been simulated with *a*)- equal volumes and feed for each bioreactor, and *b*) with optimum volumes and feed for each bioreactor. The percent improvement is determined based on the improvement in the productivity.

Substrate Conc. (g/L)	V_1 (L)	V_2 (L)	F_1 (L/h)	F_2 (L/h)	D_1 (h^{-1})	PD (g/L.h)	Improvement (%)
2	0.50	0.50	0.100	0.100	0.200	0.027	-
2*	0.42	0.58	0.184	0.016	0.435	0.036	35
6	0.50	0.50	0.100	0.100	0.200	0.071	-
6*	0.20	0.80	0.174	0.026	0.875	0.105	41
10	0.50	0.50	0.100	0.100	0.200	0.119	-
10*	0.18	0.82	0.162	0.038	0.895	0.169	42

* Optimum operation.

There has been substantial research on increasing the number of CFSTR in series (Baheri et al., 1995; Bishoff, 1966; Hill and Robinson, 1989; Shimizu and Mutsubara, 1987). However, no study has been reported the effect of increasing the number of CFSTR in the fermentation of recombinant bacteria. Therefore the objective of this part of study was to investigate the feasibility of using multiple CFSTR (more than two) in the fermentation of recombinant bacteria. Two different arrangements of bioreactors were considered by increasing the number of growth stages or production stages from one to three (Figure 5.17).

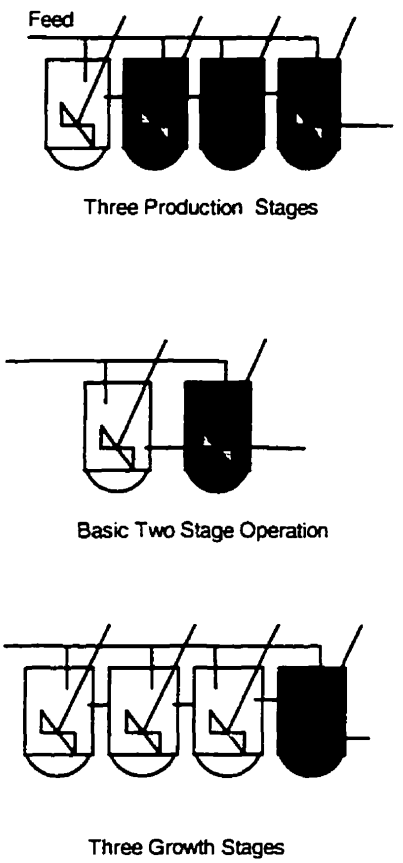


Figure 5.17. Different bioreactor arrangements considered in the feasibility study of using multiple CFSTR in the fermentation of recombinant bacteria. The shaded bioreactors represent production stages.

In the first step, these arrangements were simulated and optimized to find their best operating parameters. The optimum productivity of each process was taken as the basis for improvement in the performance of the system (at constant final substrate concentration). It is known that the productivity is not the only parameter considered in the design of the microbial processes, but it is a relatively good measure of both the total product formation rate and the total volume of the process which are the two important factors contributing to the cost of the process. A typical simulation result for five arrangements can be seen in Figure 5.18. Increasing the number of growth stages from 1 to 3 does not have any effect on the productivity of the system because the volume of the additional reactors approaches zero at their optimum productivities. On the other hand, increasing the number of production stages increases the productivity. However the increase is only about 12% which is not significant compared to the improvement in the performance of most primary product fermentations (2-10 times) as reported in the literature (Bishoff, 1966; Baheri et al., 1995; Hill and Robinson, 1989; Shimizu and Mutsubara, 1987).

In order to explain and extend the results, it is necessary to closely look at the optimum two stage fermentation (one growth and

one production stage). The optimum design parameters at three final conversions are listed in Table 5.3. It can be seen that the dilution rate of the first stage (F_1/V_1) is around 0.43 h^{-1} which appears to be independent of both the second stage and the final conversion. It can be shown that this dilution rate is very close to the optimum dilution rate for biomass production in a single CFSTR (Park et al., 1989, Baheri et al., 1996), which can be calculated from the following equation for Monod biokinetics (Bailey and Ollis, 1986):

$$D_{opt} = \mu_m \left[1 - \left(\frac{K_S}{K_S + S_0} \right)^{0.5} \right] \qquad (5.1)$$

Calculating the optimum dilution rate for a single CFSTR with a substrate concentration of 2 g/L in the feed, gives the value of 0.48h^{-1} (Equation 5.1), which is in fact close to the optimum dilution rates of the two stage fermentation (Table 5.3). In using the above equation, the plasmid free cells were considered to have very little effect on the dilution rate of the process (less than 5% in population).

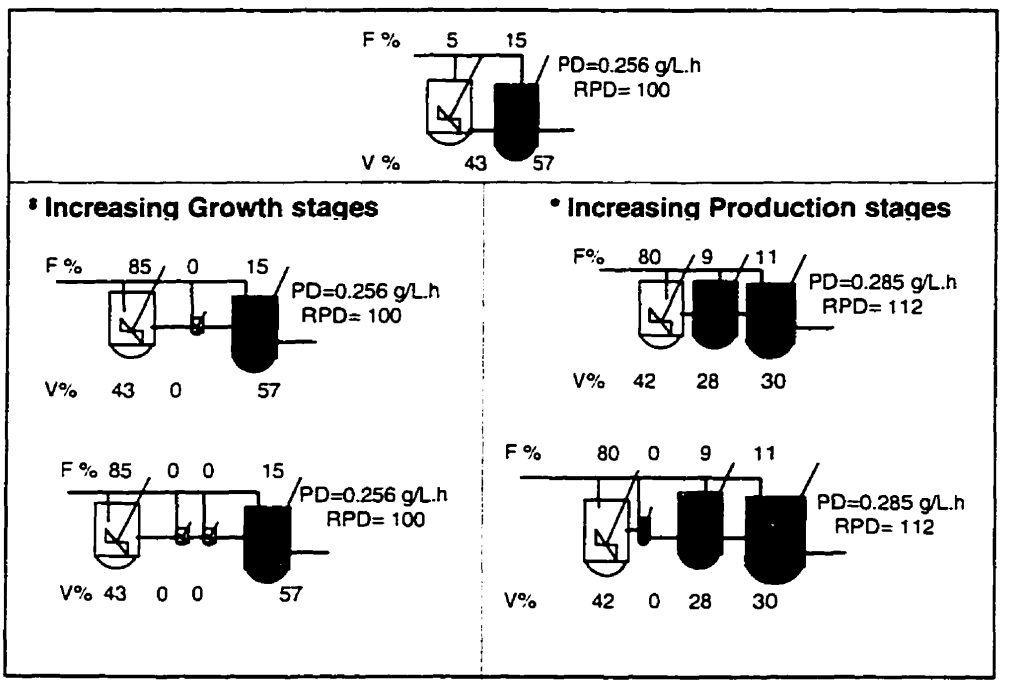


Figure 5.18. Effect of increasing the number of growth or production stages on the productivity of continuous fermentation process. Optimum parameters and productivities for five different bioreactor setups shows no improvement by increasing the number of growth stages, but some improvement with increasing the number of production stages. RPD is the relative productivity of each setup to the standard two stage process. The shaded bioreactors represent production stages.

Now it is evident that increasing the number of CFSTR in the growth stage does not improve the productivity of the process, because the first stage is already operating near the optimum dilution rate. On the other hand, increasing the number of production stages increases the productivity because of the approach to plug flow behavior for this case of non-autocatalytic reaction kinetics (Levenspiel, 1972).

Table 5.3. Model simulations for the optimum dilution rates in the first and second stage of recombinant *E. coli* (BL21DE3) fermentation. The substrate concentration in the feed is 2 g/L and the IPTG concentration is 25 μg /mL.

Final Substrate Conversion (%)	D_1 (h^{-1})	D_2 (h^{-1})	PD (g/L.h)
86	0.463	0.504	0.039
91	0.435	0.347	0.036
96	0.417	0.190	0.028
98	0.310	0.082	0.016

It is interesting to review the reported results on some other fermentations of recombinant bacteria with a regulated gene. Park et al. (1989) studied the two stage fermentation of a recombinant *E. coli* sensitive to IAA (3 β-indoleacrylic acid). The study was based on a defined performance index covering main process expenses. The results showed a constant dilution rate for the first tank when the productivity of the process was the main contributor to the cost of the

process (Table 5.4). This dilution rate was close to the optimum dilution rate calculated from Equation (8) (≅0.489 h^{-1}). In another study on *E. coli* with a temperature sensitive gene, the same agreement was observed between the optimization results and the above equation for the dilution rate of the first tank (Baheri et al., 1996). The reported results suggest that one can generally extend the results of this study to using multiple CFSTR in fermentation of *E. coli* (BI21DE3) to other recombinant DNA bacterial fermentations.

Table 5.4. Summary of results for dilution rates at high V_2/V_1 taken from the study of Park et al. (1989) on the fermentation of an environmentally regulated *E. coli*. (D_1=0.489 h^{-1} from Equation (8)).

V_2 / V_1	D_1 (h^{-1})	D_2 (h^{-1})	PD* (g/L.h)
5	0.403	0.343	0.0473
10	0.489	0.309	0.0470
50	0.489	0.267	0.0459

* Value of performance index which is close to productivity.

Finally, the experimental study of a two stage CFSTR process was started by conducting tests on plasmid instability in the single stage continuous process. As mentioned before, a stable supply of plasmid-harboring cells from the growth stage to the production stage is essential in maintaining a continuous fermentation process for the recombinant microorganism. The plasmid instability tests were

carried out using a single reactor running over a period of 4-6 days at a substrate concentration of 2 g/L. The percentage of plasmid-harboring cells was determined by the replica plating method on samples taken every 12-24 h. Figure 5.19 shows the maintenance of plasmid-harboring cells during a single stage continuous operation at three different dilution rates (no induction).

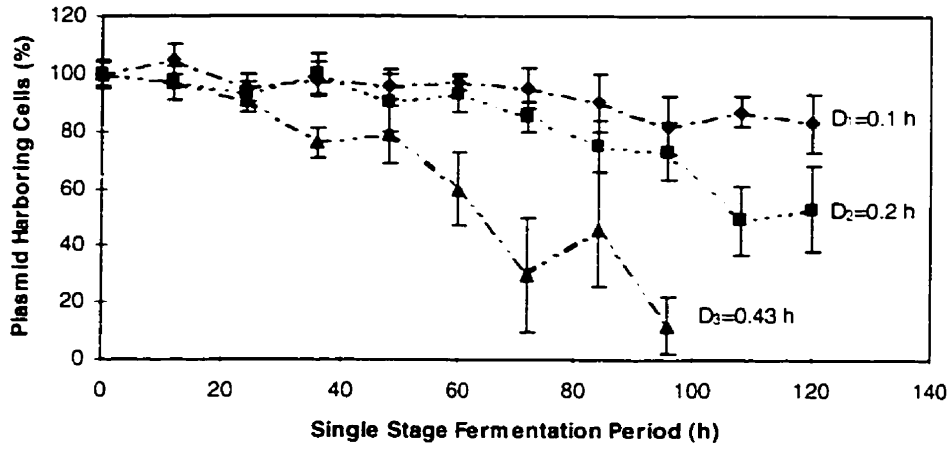


Figure 5.19. Maintenance of plasmid-harboring cells in a single stage CFSTR at three different dilution rates. The substrate concentration was 2 g/L and the working volume of reactor was 300 mL. The percentage of plasmid-harboring cells was determined by the replica plating method.

As is evident from the data, the instability increases with an increase in the dilution rate. This is mainly due to the fact that a higher dilution rate means a higher bacterial growth and generation rate, which directly increases the instability. Moreover, at higher growth rates, the number of random mutations (mistakes) at the molecular level increases which in turn increases the rate of plasmid loss. The

data in Figure 5.19 also suggests that in order to have a stable supply of plasmid-harboring cells from the bioreactor, the dilution rate must be chosen as low as possible. This means that a two stage process operating at optimum dilution rate for the first stage is not sustainable over the long term. Figure 5.20 shows such a rapid drop in the enzyme concentration of a two stage process operating at the optimum dilution rates.

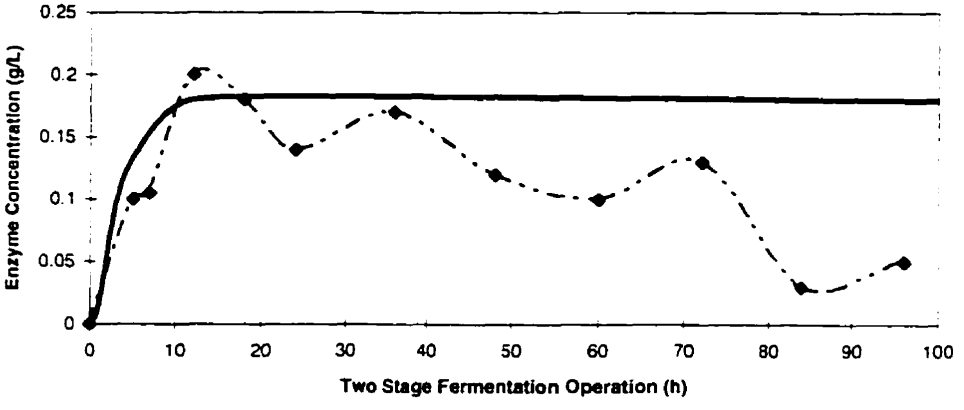


Figure 5.20. Performance deterioration of a two stage CFSTR operating at the optimum dilution rates. The data are for the two stage fermentation with S_0= 2 g/L, D_1=0.435 h^{-1} , D_2=0.346 h^{-1}; and IPTG= 25 µg /mL. High performance of the system is not sustainable over the long term because of the poor maintenance of plasmid-harboring cells in the first stage. Solid line represents model simulations.

The dilution rate in the second stage is not critical for process stability. However proper selection of this dilution rate is necessary

to achieve the optimum productivity. Figure 5.21 shows the effect of dilution rate of the second stage on the biomass concentration and overall productivity of a two stage continuous process.

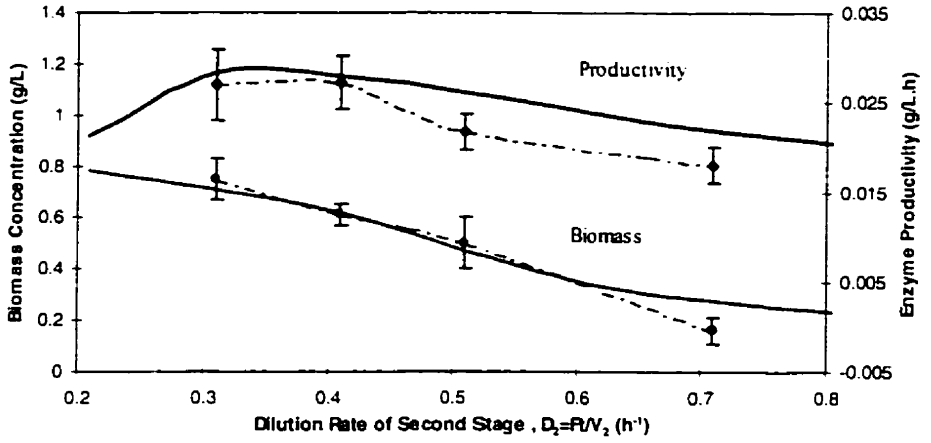


Figure 5.21. Effect of the second stage dilution rate on the productivity of a two stage process. The data are for S_0= 2 g/L, D1=0.2 h^{-1} and IPTG= 25 µg /mL. The dilution rate of the second stage was adjusted by changing the feed flow rate to the second stage. Solid lines represent model simulations

Initial experiments on the two stage continuous fermentation at higher substrate concentrations of 6 and 10 g/L resulted in surprisingly poor productivity, 60 to 70% lower than expected. Analysis of the steady state biomass concentrations from the first stage showed around 20-40 percent reduction in the growth rate of the bacteria. Using this lower growth rate, the model simulations showed the same sharp reduction in the productivity. In simple terms, the lower specific growth rate caused partial cell washout from the

first stage, which in turn reduced the biomass and enzyme productivity in the second stage. There are several factors that may contribute to the reduction of the growth rate of bacteria in the continuous process:

a) Generally the kinetics of cell growth and protein formation is often slightly different in the continuous process compared to the batch process because of different environments in the batch and continuous bioreactors. In the batch process, the concentration of ingredients in the environment (medium) changes with time, while this parameter is relatively constant in the continuous operation. This will affect the type of cell interactions with the medium and therefore the cell growth rate and protein formation. When using a feed with substrate concentration of 2 g/L, no significant reduction in the specific growth rate was observed in the continuous process. This may be explained by the use of pure glucose as the only substrate at this concentration. Glucose can be easily assimilated with no need for any enzymatic pretreatment. Other carbon sources need to be cleaved into simpler molecules to function as the food source. The breakdown of complex carbon sources is often partially achieved outside the cell by enzymes released into the medium.

Therefore, the deviation between the continuous and the batch process increases significantly when using complex carbon sources, because they are exposed to different levels of enzymatic activity.

b) To control foam formation, an antifoam was added to the medium which might have affected the cell membranes and their ability to uptake substrate and IPTG. In addition, the antifoam used in the experiment (Antifoam "B") was preserved by an antibacterial agent. The concentration of the preservatives was very small when the antifoam was mixed with the medium but still might pose some limitations on the bacterial growth (30-40 ppm antifoam was used for the continuous process at 2 g/L).

c) It was difficult to continuously inject the necessary chemicals to maintain proper pH, therefore some NaOH was added directly to the feed reservoir to compensate for the acetic acid formation during the fermentation process. This increased the pH of the feed from 7 to around 7.6-7.8. It also caused some precipitation, which made the medium less rich in mineral nutrient.

Despite lower biomass growth rates in the continuous process, it was still possible to get high productivities by using the proper

optimum parameters determined based on the new growth rates. Single stage continuous runs were used to find the new kinetic parameters of biomass growth. The new maximum specific growth rate and Monod constant (K_S) were determined from the experimental data to be 1.26 h^{-1} and 3.08 g/L respectively (Appendix B). The new bacterial kinetic parameters in the continuous process show a lower specific growth rate for the bacteria compared to the batch process (Figure 5.22). The new bacterial growth parameters were used to carry out two stage continuous experiments with mixed nutrient (6 and 10 g/L).

Figures 5.23 and 5.24 show the effect of the volume ratio of the first to second stage on the optimum performance of the two stage process at substrate concentration of 6 and 10 g/L. As can be seen from Figures 5.23 and 5.24, the deviation from the theoretical results are higher at the concentration of 10 g/l. More difficulty in controlling the pH and foam formation are two factors that may contribute to the problem.

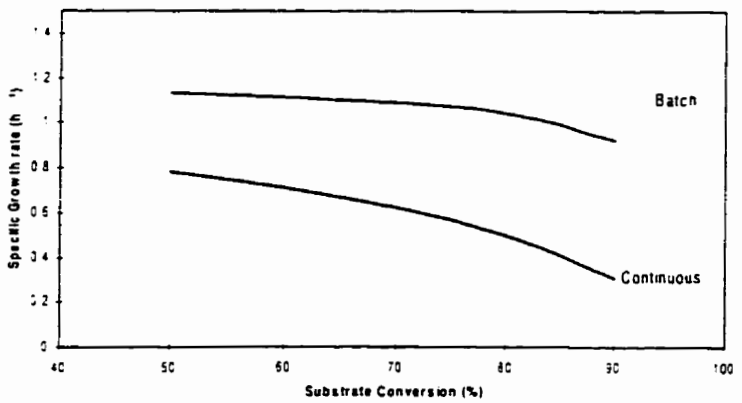


Figure 5.22. Specific growth rates at different substrate conversions in the batch culture and continuous culture. The batch data are based on μ_m=1.2 h^{-1}, K_S=0.3 g/L, and the continuous data are based on μ_m=1.26 h^{-1} , K_S=3.08 g/L. As can be seen the specific growth rates are lower for the continuous process, especially at high substrate conversions.

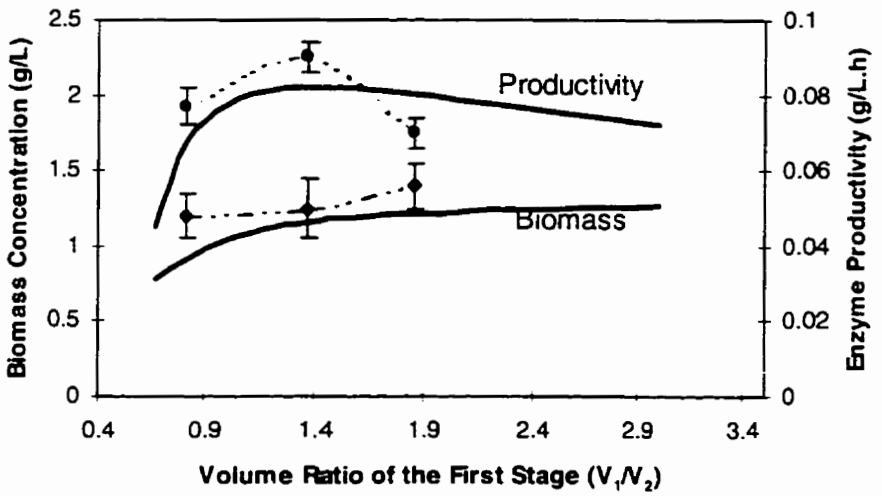


Figure 5.23. Effect of the volume ratio of the first to second stage on the optimum productivity of two stage CFSTR with feed substrate concentration of 6 g/L. The data are for a two stage process with F_t=0.15 L/h, V_t=0.4 L, S_0=6 g/L, and IPTG=25 µg/mL. The optimum parameters are F_1/F_t=72.5%, V_1/V_t= 57%. Solid lines represent model predictions.

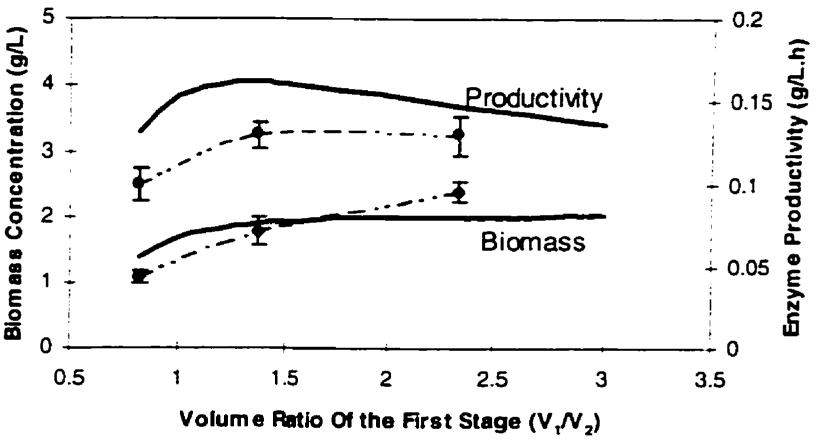


Figure 5.24. Effect of the volume ratio of the first to second stage on the optimum productivity of two stage CFSTR with feed substrate concentration of 10 g/L. The data are for a two stage process with F_t=0.2 L/h, V_t=0.4 L, S_0=10 g/L, and IPTG=25 µg/mL. The optimum parameters are F_1/F_t= 79.3%, V_1/V_t=58%. Solid lines represent model predictions.

5.2. Optimization of Multiple CFSTR Fermentation with Complex Biokinetics

Continuous culture fermentation is attractive from various points of view, such as high productivity, constant product quality and ease of process control (Shimizu and Matsubara 1987, Cacott 1981, Ricica 1970). There have been many studies concerning optimization of CFSTR in series with different mathematical methods and biokinetic equations (Bischoff 1966, Luyben and Temper 1982, Hill and Robinson 1989, Wall and Hill 1992, Wen and Chang 1967). Analysis of optimization problems by the analytical approach is quite lengthy and time consuming even for simple biokinetic equations. By simple biokinetic equations, we mean those which provide analytical equations for the performance of the stage (steady state solution or explicit stage transformation). This includes most of the unstructured primary product biokinetics, such as Monod (1942), Aiba and Shoda (1969), and Haldane (1930). Complex biokinetic equations can be classified as those for which it is difficult to find the performance (require significant mathematical manipulation or may be impossible), such as models for secondary products, recombinant DNA products or substrate co-metabolisms. In these cases, stage wise optimization

problems can be best approached by numerical optimization techniques such as nonlinear programming or dynamic programming etc. (Curel and Lapidus 1968, Gottfried and Weisman 1967, Roberts 1966). Application of these methods (especially nonlinear programming techniques) to simple biokinetic equations has been demonstrated by Wen and Chang 1967, Ong 1986, Blanch and Rogers 1972, and Park and Seo 1989. For more complex biokinetic equations however, the application of the above methods are not simple because obtaining analytical expressions for the performance of a stage is very difficult or in some cases impossible (even for the simplest structured William's model).

In this study, the generic optimization code for multiple CFSTR, presented in Section 3.2, was tested under different conditions of biokinetic equations and objective functions, and the results were compared with previously published data. Finally, the generic optimization model was applied to multiple CFSTR with *cybernetic* biokinetics and the effect of model parameters on the optimum total residence time was shown. In this section, a summary of results for a few cases are presented.

5.2.1. Optimization for Both the Best Residence Time and Number of Stages with Simple Biokinetics.

Ong (1986) studied the optimization of the Michaelis Menten kinetic model (Bailey and Ollis 1986, Lee 1992) in CFSTR by using Dynamic Programming. He reported the optimum number of CFSTR in series with different cost related objective functions. The problem can be stated as follows: finding the best number of bioreactors at their optimum residence time distribution, subject to the following conditions:

$$r_S = \frac{\mu^m S}{K_s + S} \qquad (5.2)$$

The kinetic data are:(Ong 1986): K_S= 100 g/L , μ_m=1 h^{-1}, S_0=100 g/L, F=1 L/h.

The objective function is : min $\sum V_i^n$ over all stages, n= 0.6, 0.65,.. 0.8, and the final substrate conversion of 99 %.

S is the concentration of the limiting substrate and V_i is the volume of each reactor. The objective function is a simple power function

commonly used to relate the volume of reactors to their capital costs, and the exponent n is often taken to be around 0.6. The optimum number of reactors can be obtained either by passing it to the optimization program as an integer number greater then zero or by taking a fixed but high number of bioreactors. In the latter case, the residence time of the additional reactors converges to zero. The latter approach was used in this case and the results were found to be exactly the same as Ong (1986), as shown in Table 5.5.

In addition, the program was tested with a number of other simple biokinetic equations (Monod 1942, Haldane 1930, and Aiba and Shoda 1969) used in the study of Hill and Robinson (1989), and in all cases exact agreement was found with their analytical approach.

Table 5.5. Optimization results for the optimum number of CFSTR in series compared with the results of Ong (1986).

Exponent (n) in ΣV_i^n	Minimum Value of Objective Function		Optimum No. of CFSTR in Series	
In both cases	Ong (1986)	This work	Ong (1986)	This work
0.6	107.13	107.126	4	4
0.65	140.82	140.821	5	5
0.70	182.21	182.201	6	6
0.75	232.29	232.285	7	7
0.80	291.97	291.965	9	9

5.2.2. Optimization for the Best Residence Time with Segregated Biokinetics

Shimizu and Matsubara (1987) studied the feasibility of using multiple CFSTR (up to three) by using the Blanch and Rogers (1972) segregated biokinetic model. The modeling equations have been presented in Chapter 2, (2.2.2.1). They studied maximum productivity and product concentration in multiple CFSTR with an analytical approach for *Ethanol* and *Gramicidin S* production. The objective was to find the maximum productivity versus product concentration in a series of one to three CFSTR. The biokinetic data used for the *Ethanol* and *Gramicidin S* production were as follows (Shimizu & Matsubara, 1987):

Ethanol parameters: K_S=1.6 g/L, μ_m=0.19 h^{-1}, Y_{xs} =0.06, Y_p=0.16, P_m=90 g/L, S_0=220 g/L.

Gramicidin S parameters: K_S=0.75, μ_m =1.19 h^{-1}, Y_{xs} =0.2, α'_2=1, β'=0.7, P_m= infinite, S_0=0.6 %.

They assumed some of the modeling parameters to be negligible and used the analytical approach for optimization, although this is generally not a correct assumption.

The above problems were solved using the computer model and the results can be seen in Figures 5.25 and 5.26. The results for ethanol production were found to be exactly the same as Shimizu and Matsubara (1987) as can be seen in Figure 5.25. In the case of Gramicidin S production an average deviation of (-3%) was found for both one stage and two stage, with almost no deviation for three stage operation. No explanation could be found for the deviations in one and two stage operations.

5.2.3. Optimization for the Best Residence Time with William's Structured model

William's two-compartment model is the simplest structured model which has been already presented in Chapter 2 (2.2.2.2). We used the following biokinetic data from Lee (1992) for the William's model to optimize a number of multiple CFSTR for maximum productivity:

k_1=1.25 L/g.h, k_2=0.0002 L/g.h, 1/ρ=0.0002 L/g, Y_{xs}=0.2

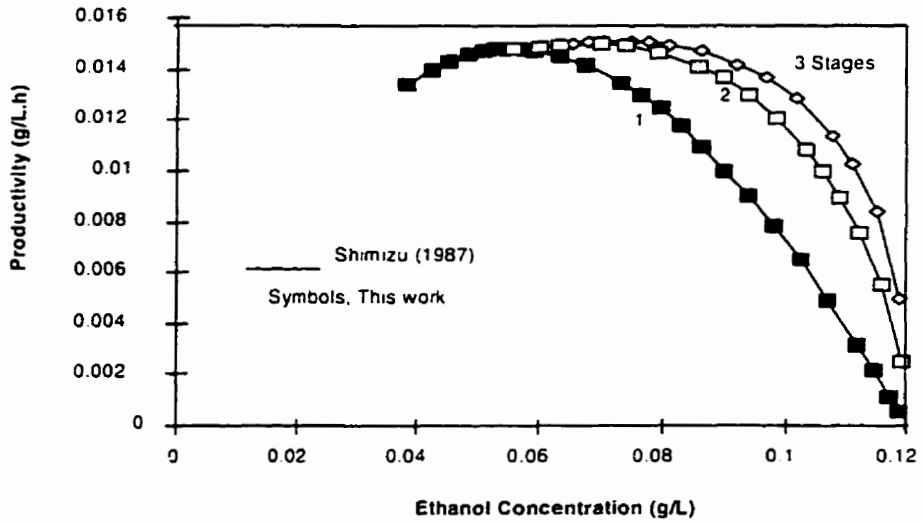


Figure 5.25. Analytical results and computer optimization results for optimum *Ethanol* productivity versus ethanol concentration (up to three optimized CFSTR). Symbols represent optimization results from the analytical approach.

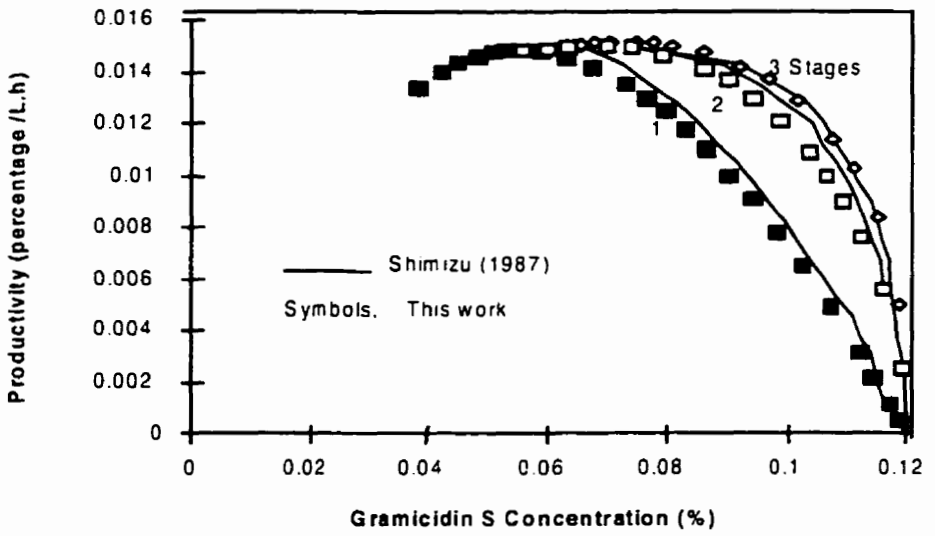


Figure 5.26. Analytical results and computer optimization results for optimum *Gramicidin S* productivity versus concentration (up to three optimized CFSTR). Symbols represent optimization results from the analytical approach.

Table 5.6 shows the optimization results for a number of multiple CFSTR (up to 4 CFSTR) in series. It can be seen that four optimum bioreactors increase the productivity by a factor of 10 compared to single stage bioreactor using William's biokinetics. The bioreactors also decrease in size as we proceed downstream, similar to the findings of Hill and Robinson (1989) for simple biokinetic models. Since no data was found for comparison of optimization with William's model, the results were tested for the single CFSTR at its maximum productivity. The optimization program shows 50% final substrate conversion for a single CFSTR operating at the optimum productivity which is the same as reported by Lee (1992) (Baheri et al., 1996).

Table 5.6. Optimum residence time distribution, final product concentration and overall productivity of multiple CFSTR with William's model.

No. of CFSTR	Residence time distribution in each stage (h) First	Second	Third	Fourth	Product	Productivity
Biomass production at 99% substrate conversion						
	h				g/L	g/L.h*10^3
One	80.00	-	-	-	0.198	0.25
Two	8.04	7.23	-	-	0.198	13.0
Three	3.83	2.99	2.88	-	0.198	20.3
Four	2.73	1.84	1.71	1.602	0.198	24.8

5.2.4. Optimization for the Best Residence Time with the Cybernetic Model.

The balance equations for different cellular components of the Ballo and Ramkrishna (1991) cybernetic model has been already presented in Chapter 2 (2.2.2.3). The optimization program was used to determine the best residence time in a multiple CFSTR fermentation with the cybernetic biokinetic model. Table 5.7 lists the cybernetic parameters used in the computer program (for growth of *Klebsiella pneumoniae* on glucose).

The optimization results for the residence time distribution of up to four CFSTR in series can be seen in Table 5.8. The objective function was maximum biomass productivity, and the substrate conversion was 99%. Table 5.8 shows that the overall productivity of the process increases by increasing the number of bioreactors. In addition, bioreactors become smaller as we proceed downstream.

The cybernetic equation has 18 kinetic parameters (Table 5.7). It was found that one hundred percent increase in 6 parameters (K_{EML}, K_M, α'_E, α'_{EML}, α^*_{EML}, β'_{EML}) had an effect of less than 5% on the total residence time in optimum bioreactors, change in another 5 parameters (α^*_E, α'_R, K_{RS}, β'_E, $\mu_M{}^m$) had a moderate effect on the

Table 5.7. Cybernetic model parameters for *Klebsiella pneumoniae* growth on glucose (Ballo and Ramkrishna 1991)

Parameter	Unit	Value
$\mu_G{}^m$	h^{-1}	1.223
$\mu_M{}^m$	h^{-1}	0.17
$\mu_{ML}{}^m$	h^{-1}	0.39
Y_{xs}	g/L	0.51
K_S	g/L	0.003
K_E	g/L	0.003
K_{RS}	g/L	0.003
K_R	g/L	0.003
K_{EML}	g/L	1×10^{-6}
K_M	g/L	1×10^{-6}
α'_E	h^{-1}	8×10^{-6}
α'_E	h^{-1}	0.001
α'^*_{EML}	h^{-1}	8×10^{-6}
α'_{EML}	h^{-1}	0.05
α'_R	h^{-1}	0.125
β'_E	h^{-1}	0.05
β'_{EML}	h^{-1}	6
β'_R	h^{-1}	2.5

total residence time (5% to 100%), and change in the rest of parameters (K_S, K_E, K_R, β'_R, $\mu_G{}^m$, $\mu_{MI}{}^m$) affected the results by 100% to 400%. Among the sensitive kinetic parameters, change in $\mu_G{}^m$ had the greatest effect on the optimum total residence time for a conversion of 99%. The reason behind these results is that the first group of parameters are related to cell maintenance, which is not important in the steady state growth; while others are directly related to the cell growth rate and therefore to the residence time of the process

Table 5.8. Optimum residence time distribution, final product concentration and overall productivity of multiple CFSTR with cybernetic model.

No. of CFSTR	Residence time distribution in each stage (h) First	Second	Third	Fourth	Product	Productivity
Biomass production at 99% substrate conversion, S_0=0.5 g/L						
	h^{-1}				g/L	g/L.h
One	1.46	-	-	-	0.239	0.163
Two	0.942	0.078	-	-	0.249	0.244
Three	0.931	0.052	0.024	-	0.250	0.248
Four	0.930	0.041	0.0201	0.013	0.250	0.249

The effect of change in (K_E, β'_E) on the total residence time of both a single and two stage CFSTR are presented in Table 5.9. It can be seen that a single CFSTR is much more sensitive in its operation to *inaccuracies* in the biokinetic constants as compared to optimized CFSTR in series.

Table 5.9. Effect of change in K_E, and β'_E on the optimum total residence time for cybernetic model (percentage change in the optimum total residence time)

Factor	Effect of Change in β'_E		Effect of Change in K_E	
	Two Stage	Single Stage	Two Stage	Single Stage
5 x	1.7 %	7.4 %	18 %	60 %
10 x	3.5	16	35	100
25 x	8.3	43	65	142
50 x	14	88	96	160

CHAPTER SIX

CONCLUSIONS AND RECOMMENDATIONS

6.1. CONCLUSIONS

The following conclusions can be made within the limits of this investigation.

- Two computer programs for optimization of recombinant bacterial fermentation in fedbatch and two stage continuous processes were developed and successfully used in the experimental study of both processes. The two stage continuous process was optimized in both the volume of each reactor and the distribution of feed to each reactor, and the fedbatch process in the percentage of initial charge, the induction time, and the *in situ* substrate concentration. The simulation and optimization results closely follow the experimental results for both processes (average deviation of 15%). This clearly illustrates the potential advantage of using mathematical modeling of growth and gene

expression along with the computer programs for optimization and design of recombinant fermentation processes.

- A successful medium was developed for high biomass fermentation of recombinant bacteria as the standard M9 medium did not support high biomass concentration with glucose as the only carbon source. A 50-50% mixture of glucose and casamino acids was used as the carbon source in the new medium. In addition, the kinetic parameters of Miao and Kompala for the bacterial growth and cloned gene expression were successfully adjusted for the new improved medium. It is interesting to note that while the medium nutrient content increases from 2 to 10 g/L (5 times), the enzyme activity increases from about 0.1 to 1 g/L (10 times). Part of this increase is due to the increase in biomass concentration from about 0.8 to 3 g/L ($\cong$4 times), but the remainder is due to the higher specific growth rate in the new medium, which increases the specific enzyme production. This increase in the enzyme activity was successfully predicted by the model simulations.

- Simulation results for the fedbatch fermentation of recombinant *E. coli* indicated the existence of an optimum induction time and initial charge. Subsequent experimental studies of the fedbatch fermentation process clearly confirmed the results of the optimization program.

- In a two stage continuous fermentation of recombinant *E. coli*, the optimum dilution rate of the first stage was found to be independent of the second stage. This optimum dilution rate is very close to the optimum dilution rate calculated for a single stage process operating at the highest biomass productivity. For Monod type bacterial growth rates, the optimum dilution rate for the first stage can be calculated analytically. In addition, there is an optimum feed distribution between the stages for maximizing the overall foreign protein production.

- For the first time, the feasibility of using multiple CFSTRs (more than two stages) for fermentation of recombinant bacteria was studied. It was found that, unlike the situation in primary products fermentation, it is not feasible to use more than two CFSTRs (one growth and one production stage) for the

continuous fermentation of recombinant bacteria. The simulation results showed that there can be a small improvement in the productivity of a system by increasing the number of production stages, but it is not significant (10-15%). There would be no improvement in the productivity of a two stage fermentation of recombinant bacteria by increasing the number of growth stages. It has been shown that, these conclusions can be reliably extended to many other fermentation processes involving recombinant bacteria with a regulated gene.

- The theoretical productivity of the two stage continuous process was found to be 30-40% higher than the fedbatch process when using the same feed in both processes (over substrate concentration of 2-10 g/L). However, there are several obstructions in achieving high productivity in a two stage continuous processes. The maintenance of plasmid harboring cells over the long term operation and often lower bacterial growth rates (especially for complex nutrients) were among the two important factors which eventually reduce the productivity of two stage continuous operation.

- The bacterial growth behavior in continuous culture was found to be different from the batch culture when using complex carbon sources. Therefore, the biokinetic parameters of the Miao and Kompala model for biomass growth and foreign gene expression were separately adjusted for both the continuous and batch processes.

- A new computer package was developed for optimization of multiple CFSTR fermentations with any biokinetic equations. The program accepts biokinetic equations, does the necessary material balances, and optimizes the system under a specific objective function. The program was tested with four different biokinetic models (structured, unstructured, distributed and segregated models) and the results compared to published data. In all cases, good agreement was found (less than 3% deviation). The computer program was also used to study the optimization of multiple CFSTR fermentation with the cybernetic biokinetic model for the first time. Besides finding the optimum residence times for multiple CFSTR operation, the effect of inaccuracies in different cybernetic model parameters on the overall productivity of the process was investigated. It was found that the

inaccuracies in the maintenance parameters of the cybernetic biokinetics has negligible effect on the performance of multiple CFSTRs fermentation. Additionally, a single CFSTR was found to be much more sensitive in its operation to *inaccuracies* in the biokinetic constants as compared to optimized CFSTRs in series (2-8 times more sensitive).

6.2. RECOMMENDATIONS

- The kinetic model of Miao and Kompala (1992) for the cell growth and protein formation in recombinant *E. coli* (BL21DE3) was reliable at higher substrate concentrations. However, the model predictions for maintenance of plasmid in the continuous process were not satisfactory. A more rigorous kinetic model for both bacterial growth and segregational instability should be developed and incorporated in Miao and Kompala's modeling of recombinant bacteria growth and protein formation. The new model should provide a more accurate prediction of bacterial growth rate in a changing environment such as batch and fed batch with a complex carbon source and a more accurate prediction of plasmid instability in the continuous processes.

- Excessive production of acetate is a problem which is often encountered in high cell density fermentation of *E. coli.* This will require pH adjustment and result in loss of productivity. A kinetic model for acetate formation during fermentation should be developed and incorporated with the growth equations. This should lead to more accurate predictions of the optimum operating point and better pH control, especially at high cell concentrations.

- The bulk concentration of IPTG is assumed to be constant in the Miao and Kompala biokinetic model. However, part of the IPTG is absorbed by the cells and another part may disintegrate in the fermentation medium. A model for prediction of the bulk concentration of IPTG should be developed and incorporated in the biokinetic model. This is especially necessary for high cell density fermentations, when a considerable portion of IPTG may be absorbed by the cells.

- Further experimental work should be undertaken to look at the expression of different foreign proteins in recombinant *E. coli at* much higher biomass concentrations. The experiments should

provide a better understanding of the problems with biomass growth and foreign protein expression at high biomass concentrations.

- A closed loop control system should be developed and used in the fedbatch process to control the feed flow rate. This would provide more accurate control of biomass and substrate concentrations in the bioreactor.

- Further experimental work should be undertaken to investigate the effect of intermittent injection of antibiotics for restoring plasmid stability in the continuous process. This could provide a new way to reduce problems with plasmid instability and bacterial contamination for the continuous process.

- The Search Pattern optimization technique was an efficient tool for process optimizations (bounded variables). However, for optimizing a large number of variables, the method was rather sensitive to the initial values of the parameters and the increments. Therefore, it would be beneficial to study the use of

other optimization techniques when optimizing a large number of parameters.

REFERENCES

Aiba S. and M. Shoda, "Reassessment of the Product Inhibition in Alcohol Fermentation ", J. Ferment. Technol., **47**, 790-794 (1969).

Baheri, H.R. and G.A. Hill and W.J Roesler, "Optimum Design for Product of *Trp-α* from Recombinant *E. coli* in a Two Stage CFSTR", Chem. Eng. J. **62**, 143-148 (1995).

Baheri, H.R., G.A. Hill and W.J Roesler, "A Generic Algorithm For Optimization Of CFSTRs In Series with any Biokinetic Equation" Can. J. Chem. Eng. **73,** 765-771 (1996).

Baheri, H.R, G.A.Hill and W.R. Roesler, "Feasibility Study of Using Multiple CFSTRs in Fermentation of Recombinant Bacteria with A Regulated Gene" Can. J. Chem. Eng., **75**, 631-635 (1997).

Bailey, J.E. and D.F. Ollis, Biochemical Engineering Fundamentals, 2nd ed.,McGraw Hill, New York, 1986.

Bailey, F.J., J. Blankenship, J.H. Condra and R.W. Ellis, "High Cell Density Fermentation Studies of Recombinant *E. coli* that Expresses Atrial Natriuretic Factor", J. Ind. Microbiology, **2**, 47-58 (1987).

Ballo, S. and, D. Ramkrishna, "Metabolic Regulation in Bacterial Continuous Cultures: I", Biotechnol. Bioeng., **38**, 1337-1352 (1991).

Bishoff, K.B, "Optimal Fermentation Reactor Design", Can. J. Chem. Eng. **44,** 281-284 (1966).

Blanch H.W. and P.L. Rogers, "Optimal Conditions for Gramicidin S production in Continuous Cultures", Biotechnol. Bioeng. **24**,151-171 (1972).

Boven, R.L., "Unraveling the Mysteries of Shear Sensitive Mixing Systems", Chem. Eng., **9**, 55-67 (1986).

Calcott P.H., Continuous Culture of Cells, **1**, 1, CRC Press Inc., Boca Raton, FL, 1981.

Childs, J., V.K Villanueba, D. Barrick, T.D. Schneider, D. Stormo, L. Gold, M.Leither and M. Garathers, "Ribosome Binding Site Sequences and Function", in "UCLA Symposia on Molecular and Cellular Biology", R. Calendar, Ed., Liss, New York, Vol. **30**, 341-350 (1985).

Carnahan, B., H.A. Luther and J.O.Wilkes, Applied Numerical Methods, John Wiley and Sons, Inc., New York, USA, 1969.

Goeddel,D., "Systems for Hetrologous Gene Expression", in Methods in Enzymology, **185**, 3-13 (1990).

Gottfried B.S., Introduction to Optimization Theory, Prentice Hall, Inc, Englewood Cliffs, New Jersey, USA, 1973.

Haldane J.B.S., Enzymes, Longman, Green and Co., London, 1930.

Hill, G.A. and C.W. Robinson, "Minimum Tank Volumes For CFSTBs In Series", Can. J. Chem. Eng. **67,** 818-824 (1989).

Hortacsu, A. and D.D.Y. Ryu, "Optimal Temperature Control Policy for a Two-Stage Recombinant Fermentation Process", Biotechnol. Prog. **6,** 403-407 (1990).

Hooke R. and T.A. Jeeves, "Direct Search Solution of Numerical and Statistical problems", J. Assoc. Comp. Mach. **2**, 8-22 (1961).

Immanaka T. and S. Aiba, "Stability of Recombinant Plasmid", Ann. N.Y. Acad. Sci., **369**, 1-14 (1981).

Kuowei W., J. Deokjin, and T.K. Wood, "Temperature and Growth Rate Effect on the hok/sok Killer Locus for Enhanced Plasmid Stability", Biotechnol. Prog. **10**, 621-629 (1994).

Lee, J.M., Biochemical Engineering, 2nd ed., Prentice-Hall Inc., 1992.

Lee, S.B., J.E. Bailey, "Analysis of Growth Rate Effects on Productivity of Recombinant *E. coli* Populations Using Molecular Mechanism Models", Biotechnol. Bioeng., 26, 66-75 (1984a).

Lee, S.B. and J.E. Bailey, "Genetically Structured Models for *lac* Promoter-Operator Function in the E. coli Chromosome and in the Multicopy Plasmids: *lac* Operator Function" Biotechnol. Bioeng., **26**, 1372-1385 (1984b).

Lee S.B., D.D.Y. Ryu, R. Seigel and S.H. Park, "Fermentation and Evaluation of Gene Expression Efficiency for Gene Product in

Two-Stage Continuous Culture System" Biotechnol. Bioeng. **31**, 805-820 (1988).

Lee, L. and Blanch H.W., "Recombinant Trypsin production in High Density Fedbatch Cultures", Biotechnol. Bioeng., **41**, 781-794 (1993).

Levenspiel, O., "Chemical Reaction Engineering", 2nd. ed., John Wiley & Sons, New York, 1972.

Luedeking, R., and E.L. Piret, "A Kinetic Study of Lactic Acid Fermentation", J. Biochem. Microbiol. Technol. Eng. **1**, 393-405 (1959).

Luli, G.W., and W.R. Strohl, " Comparison of Acetate Production and Inhibition of *E coli* in Batch and Fedbatch Fermentation", App. Environ. Microbiol., **56**, 1004-1015 (1990).

Luyben K. Ch. A.M. and J.Temper, "Optimal Design for CSTRs in Series Using Michaelis-Menten Kinetics", Biotechnol. Bioeng. **24**,1217-1220 (1982).

Maniatis, T., E.F. Fritsch, and J. Sambrook, Molecular Clonning, Cold Spring Harbor Laboratory, New York, 1982.

Miao F. and D.S. Kompala, "Overexpression Of Cloned Genes Using Recombinant *E. coli* Regulated by a T7 Promoter; Batch Culture and Kinetic Modeling", Biotechnol. Bioeng. **40,** 787-796 (1992).

Miao F. and D.S. Kompala, "Overexpression of Cloned Genes Using Recombinant *E. coli* Regulated by a T7 Promoter; Two Stage Continuous Culture", Biotechnol. Bioeng. **42,** 74-85 (1993).

Monod J., Recherches sur la Croissance des Cultures Bacteriennes", Herman and Cie, Paris, 1942.

Neilay N.D., T. Hotlinger, J.E. Bailey, "Overproduction of glycogen in *E. coli* Blocked by Acetate Pathway Improves Cell Growth", Biotechnol. Bioeng., **44**, 132-146 (1994).

Neubauer, P., L. Haggstrom, and S.O. Enfors, "Influence of Substrate Oscillations on Acetate Formation and Growth Yield in *E. coli* Fedbatch Cultivations" Biotechnol. Bioeng., **47**, 139-146 (1995).

Ong S.L., "Optimization of CFSTRs in series by Dynamic Programming", Biotechnol. Bioeng. **28**, 818-823 (1986).

Panda, A.K., A. Ghorpade, and G.P. Talwar, "High Cell Density Fermentation of Recombinant *Vibrio cholerae* for the Production of B Subunit of *E. coli* Enterotoxin", Biotechnol. Bioeng., **45**, 245-257 (1995).

Park T.H. and Seo J.H., "Optimization of Fermentation Process Using Recombinant *E. coli* with the Cloned *trp* Operon", Biotechnol. Bioeng. **34**,1167-1177 (1989).

Park, S., D.D.Y. Ryu and J.Y Kim, "Effect of Cell Growth Rate on the Performance of a Two-Stage Continuous Culture System in a Recombinant *E. coli* Fermentation", Biotechnol. Bioeng. **36,** 493-505 (1990).

Park, T.H., J.H. Seo and H.C. Lim, "Optimization of Fermentation Processes Using Recombinant *E. coli* with the Cloned *trp* Operon", Biotechnol. Bioeng. **34,** 1167-1177 (1989).

Primrose, S.B., Modern Biotechnology, Blackwell Scientific Publication, London, England, 1987.

Ricica J., "Multi-Stage Systems" in Methods in Microbiology, Norris J.R and D.W. Ribbons, Vol 2, Academic Press, New York, 329-347 (1970)

Remaut E., Stanssens P. and Fiers W., "Plasmid Vectors for High Efficiency Expression Controlled by pL Promoter of Coliphage Lambda", Gene, **15**, 81-95, (1981)

Seigel, R. and D.D.Y. Ryu, "Biokinetic Study of Instability of Recombinant Plasmid in *E. coli* Using Two-Stage Continuous Culture System", Biotechnol. Bioeng. **27**, 28-33 (1985).

Seo, J. and J.E. Bailey, "Continuous Cultivation of Recombinant *E. coli*: Existence of an Optimum Dilution Rate for Maximum Plasmid and Gene Product Concentration", Biotechnol. Bioeng. **28**, 1590-594 (1986).

Seressiotis, A. and J.E. Bailey, "Optimal Gene Expression and Amplification Strategies for Batch and Continuous Recombinant Cultures" Biotechnol. Bioeng., **24**, 392-400 (1987).

Shimizu K., and M. Matsubara, "Product Formation Patterns and the Performance Improvement for Multistage Continuous Stirred Tank Fermentors", Chem. Eng. Commun. **52**, 61-70, (1987).

Stryer L., Biochemistry, W.H. Freemen and Company , San Francisco, 1981.

Studier F.W., A.H. Rosenberge, J. Dumn, and J.W. Dubendruff, "Use of T7 RNA Polymerase to Direct Expression of Cloned Gene", Methods in Enzymology, **185**, 50-59 (1990).

Studier, F.W. and B.A. Moffatt, "Use of Bacteriophage T7 RNA Polymerize to Direct Selective High-Level Expression of Cloned Gene", J. Molec. Biol. **189,** 113-130 (1986).

Wall G. and G.A. Hill , "Optimum CFSTR Bioreactor Design", Can. J. Chem. Eng, **70,** 148-152 (1992).

Watson J.D., M. Gilman, M. Zoller, and J. Witkowski, Recombinant DNA, 2nd Ed., Freeman and Company, New York., 1992.

Wen C.Y. and T.M. Chang, "Optimization of Stage Wise Bioreactor System with Product Feedback", Biotechnol. Bioeng. **19**, 113-127 (1967).

Whitney, G., B.R. Glick and C.W. Robinson, "Induction of T4 DNA Ligase in a Recombinant Strain of *E. coli.*", Biotechnol. Bioeng. **33,** 991-998 (1989).

Wu, K., D. Jahng, and T.K. Wood, "Temperature and Growth Rate Effects on the hok/sok Killer Locus for Enhanced Plasmid Stability" Biotechnol. Prog., **10**, 621-634 (1994).

Yansura, D.G., and D.J. Henner, "Use of *E. coli trp* Pomoter for Direct Expression of Proteins", Methods in Microbiology, **185**, 55-67 (1990).

Yee, L., and H.W. Blanch, "Recombinant Trypsin Production in High Cell Density Fedbatch Cultures in *E. coli*", Biotechnol. Bioeng., 41, 781-792 (1993).

Zabriskie, D.W., D.A. Wareheim, and M.J. Polansky, "Effects of Fermentation Feeding Strategies prior to Induction of Expression of a Recombinant Malaria Antigen in *E. coli*", J. Ind. Microbiology, **2**, 87-100 (1987).

Zhang J., Kalegerakis N., Behie L. and Latrou K., "A Two Stage Bioreactor System for the Production of Recombinant Proteins Using Genetically Modified Baculovirus/Insect Cell System", Biotechnol. Bioeng., **42**, 357-365 (1993).

Zhang J., N. Kalogerakis, L.A. Behie and K. Latrou, "Optimum Infection Condition for Recombinant Protein Production in Insect Cell Suspension Culture", Biotechnol. Prog. **10,** 636-643 (1994).

APPENDIX A1

M9 MEDIUM PREPARATION

(adapted from Maniatis, 1992)

To 700 ml of sterilized de-ionized water (T< 50 °C) add:

a- 2 ml of 1 M $MgSO_4$
b- 0.1 ml of 1 M $CaCl_2$
c- 200 ml 5XM9 Salt
d- Appropriate amount of 20 % solution of carbon source

5XM9 salts is made by dissolving the following salts in de-ionized water to the final volume of 1 L.

Na_2HPO_4	33.91	g
KH_2PO_4	15	g
NaCl	2.5	g
NH_4Cl	5.0	g

Each of these solutions (a-d) should be prepared and sterilized separately, except for the glucose solution which should be sterilized by filtration (0.2 μm filter).

The final volume of the medium is adjusted to 1 L by adding more de-ionized sterile water. The final solution is clear and without any precipitation.

APPENDIX A2 - TOTAL PROTEIN MEASUREMENT

(Bio-Rad Protein Assay)

(Adapted from Bio-Rad Protein Assay manual)

The Bio-Rad Protein Assay is dye-binding assay in which a differential color change of a dye occurs in response to various concentrations of protein. The obsorbance maximum of Coomassie Brilliant Blue G-250 shifts from 465 nm to 595 nm when binding to protein occurs. The dye binds primarily basic and aromatic amino acid residues, especially arginine.

Micro-assay Procedure (modified)
Standard Total Protein Data

1-Prepare enough 5 times diluted dye solution (1 ml for each sample to be tested).

2-Prepare *five standard solutions* by adding BSA to 1 ml of diluted dye solutions to the final concentrations of 0,2,4, 6 and 8 µg/ml (linear range).

3- Wait for at least 5 minutes (not more than 20 minutes) at room temperature and then measure the OD at 595 nm. The data will be used later as the standard value for the conversion of the actual OD values to total protein content.

Total Protein Measurement

1- Add 3-10 µl of sonicated cell samples to 1 ml of 5 times diluted dye solutions.

2- Wait for at least 5 minutes and measure the OD at 595 nm.
3- Convert the OD data to protein concentrations based on BSA standard data.

APPENDIX A3- β-Galactosidase Activity Measurements
(adapted from Maniatis, 1992)

1. For each sample to be assayed mix (in a water batch at 37 °C):
   - a- 3 µl 100xMg solution*
   - b- 66 µl 1XONPG solution**
   - c- 228 µl 0.1 M sodium phosphate (pH 7.5)***

2.. Add 3 µl (2-10 µl) of the sonicated cell extract and incubate at 37 °C. The enzyme is sensitive to heat and therefore the sonicated samples should be kept on ice prior to the incubation.

3. Stop the reaction whenever a faint yellow color has developed by adding 500 µl of 1 M NaCO3 (20- 300 seconds).
4. Read the OD at 420 nm (linear 0.2-0.8).

5. The enzyme activity can be calculated using a standard curve (from a standard enzyme solution) or in terms of color change of the solution.

$$Activity = \frac{OD_{change\ at\ 420\ nm}}{(t_{incubation})\,1\,ml\,{}^{3\,\mu l}/_{1000\,\mu l}}$$

* 100XMg solution is 0.1 M $MgCl_2$, 4.5 M β-mercaptoethanol.

** 1XONPG is a 4 mg/ml solution of ONPG in 0.1 M sodium phosphate buffer.

*** 0.1 M sodium phosphate buffer is made by mixing 41 ml of 0.2 M Na_2HPO_4 and 9 ml of 0.2 M NaH_2PO_4 solution.

APPENDIX B

EXPERIMENTAL DATA FOR CALCULATION OF BACTERIAL GROWTH RATE PARAMETERS

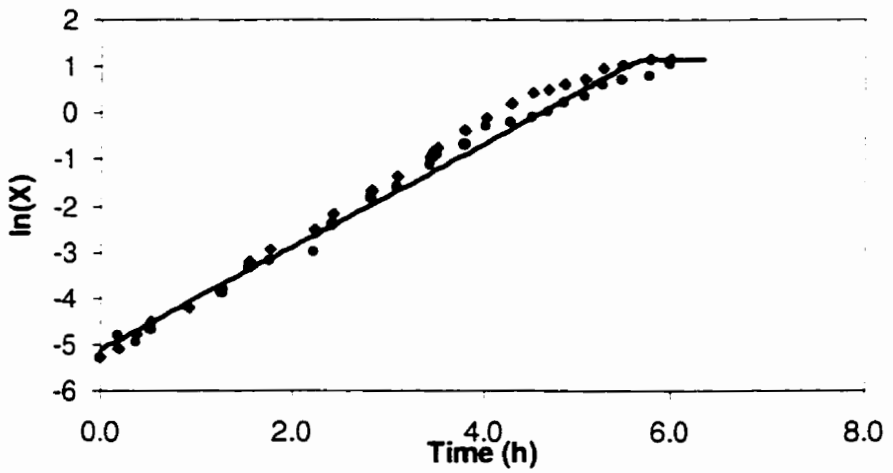


Figure B.1. Batch experimental data for calculation of growth rate parameters in M9 medium with 10 g/L mix nutrient. Solid line represents simulation results based on the K_s=0.3 g/L and μ_m=1.2 h^{-1} which are obtained from non-linear regression.

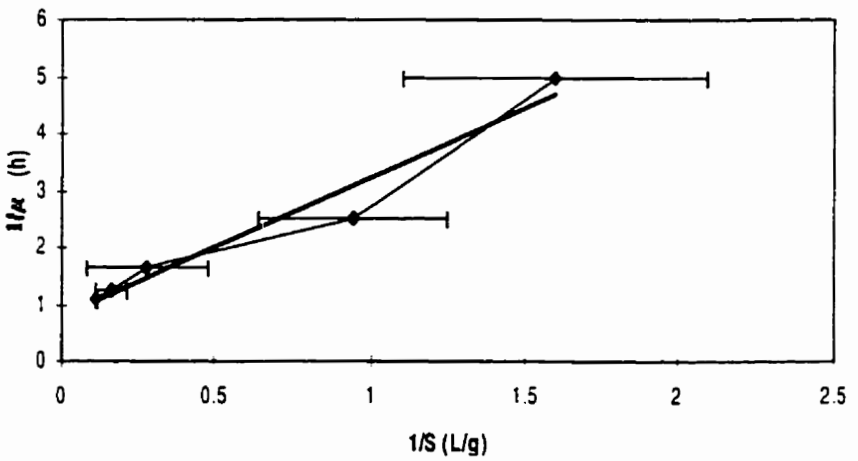


Figure B.2. Experimental data from single stage continuous operation for calculation of growth rate parameters in M9 medium with 10 g/L mix nutrient. Solid line represents simulation results based on the K_s=3.08 g/L and μ_m=1.26 h^{-1} which are obtained from linear regression.

APPENDIX C

COMPUTER PROGRAMS

C.1. Pattern Search optimization method

One of the *direct search techniques* which has been found to be particularly effective is that of Hooke and Jeeves (1961), known as *pattern search*. This algorithm can be explained as follows: an incremental value e_i is assigned to each independent parameter x_i. This incremental value controls the size of variable move and need not be the same for all variables. It is often represent a fraction of the distance between the lower and the upper bound of the variable. The value of e_i remains constant until it becomes necessary at some point of the calculation to reduce them simultaneously.

The optimization begins by selecting an initial *exploratory point* $X^{\cdot}_1=(x_1, x_2,...x_n)$ in the region of interest and evaluating the *objective function* (y) at this point. Knowing the objective function at this point, x_1 is increased to x_1+e_1 and the objective function is evaluated at this new point. If the objective function shows improvement as a result of this move (i.e. if $y(x_1+e_1,x_2...x_n) > y(x_1,x_2,..x_n)$ for a maximization problem) then the new $x_1=x_1+e_1$ and the new exploratory point is $X^{\cdot}_{11}$. If the move does not improve the objective function, then we try $x1=x_1-e_1$. In the case that objective function show no improvement at either movements, we set the new exploratory point $X^{\cdot}_{11}=X^{\cdot}_1$.

From $X^{\cdot}_{11}$, we proceed to change the $x_2 \pm e_2$ and thus find the next exploratory point $X^{\cdot}_{12}$ as above. The procedure is repeated until at some point we establish $X^{\cdot}_{1n}$. This terminal point in the exploratory step is called a *base point,* which is designated by simply by X_1. A *pattern move* is now made, providing that some improvement has been obtained in the *exploratory sequence.* This is achieved by extrapolating from the initial exploratory point $X^{\cdot}_1$ through the base point X_1 to find the new exploratory point $X^{\cdot}_2$.

$$X^{\cdot}_2=X^{\cdot}_1+2(X_1-X^{\cdot}_1) \qquad (C.1)$$

Another sequence of exploratory move is then carried out to find the next base point X_2. The next pattern move is made by extrapolation through the two previous base points.

$$X^{\cdot}_3=X_1+2(X_2-X_1) \qquad (C.2)$$

This type of extrapolation move was not possible to make in the first round, because there was only one base point.

The successive alteration of an exploratory and pattern move is continued as long as the strategy improves the objective function. All pattern moves (except the first one) are made according to the following expression:

$$X^{\cdot}_{i+1}=X_{i-1}+2(X_i-X_{i-1}) \qquad (C.3)$$

In the event that pattern move does not improve the objective function, the new exploratory point is set to the last base point ($X^{*}_{i+1}=X_i$). If the exploratory sequence is not successful, then all e_i are reduced by a constant factor and a new exploratory sequence is initiated. The optimization process terminates when the value of all e_i are reduced below a certain pre-assigned value (tolerance)

C.2. Objective Functions

The optimization of the fedbatch and two stage CFSTR were based on maximizing the processes productivity. The productivity of the two stage CFSTR can be calculated from the following equation:

$$PD=\left(\frac{F\,Pp}{V_1+V_2}\right) \qquad \text{(C.4)}$$

in which F is the total feed flow rate, Pp is the concentration of enzyme in the outlet stream, and V_1 and V_2 are the volumes of stages respectively.

The productivity of the fedbatch process can be calculated from a similar equation:

$$PD=\left(\frac{Pp}{(V_0+V_t)\,t}\right) \qquad \text{(C.5)}$$

in which Pp is the final enzyme concentration in the bioreactor, V_0 is the volume of the initial charge, V_t is the total volume of the process and t is the fedbatch fermentation duration. To avoid *singularity* of

the objective function at time zero, the final enzyme concentration can be corrected for the enzyme concentration in the batch process as follows:

$$PD = \left(\frac{Pp - Pp_{batch}}{(V_0 + V_f)\, t} \right) \qquad \text{(C.6)}$$

in which Pp_{batch} is the enzyme concentration at the beginning of the fedbatch process and is calculated directly from the batch growth phase in the computer program.

C.3. Numerical Method for Solving Differential Equations

Euler equation with *correction* was used to solve the systems of differential equations. Consider the following system of differential equation:

$$Y = (y_1, y_2, \ldots\ldots, y_n) \qquad \text{(C.7)}$$

$$Y' = (y_1', y_2', \ldots\ldots, y_n') \qquad \text{(C.8)}$$

$$\frac{dY}{dt} = Y' \qquad \text{(C.9)}$$

with the initial condition of: $Y = Y_0 \; at \; t = 0$.

The value of the Y at the next increments are calculated from the following equations (*predictor corrector equations*):

$$Y_{i+1} = Y_i + \Delta t \, Y_i' \qquad \text{(Predictor)} \qquad \text{(C.10)}$$

$$Y_{i+1} = Y_i + \Delta t\left(\frac{Y_i' + Y_{i+1}'}{2}\right) \qquad \text{(corrector)} \qquad \text{(C.11)}$$

in which Δt is the time increment.

C.4. Initial Data of the Computer Programs

The following is the list of the initial data used in both computer programs.

a) Variables increment for the *Pattern-Search* optimization method:

e_1=0.000911, e_2=0.00011,
e_3=0.0011 , e_4=0.000211 (for two stage CFSTR)

or

e_1=0.0011 , e_2=0.01,
e_3=0.01511, e_4=0.000211 (for fedbatch)

b) Optimization Tolerance:

Tolerance=0.05 %=0.0005

C.5. Two Stage CFSTR Optimization Program

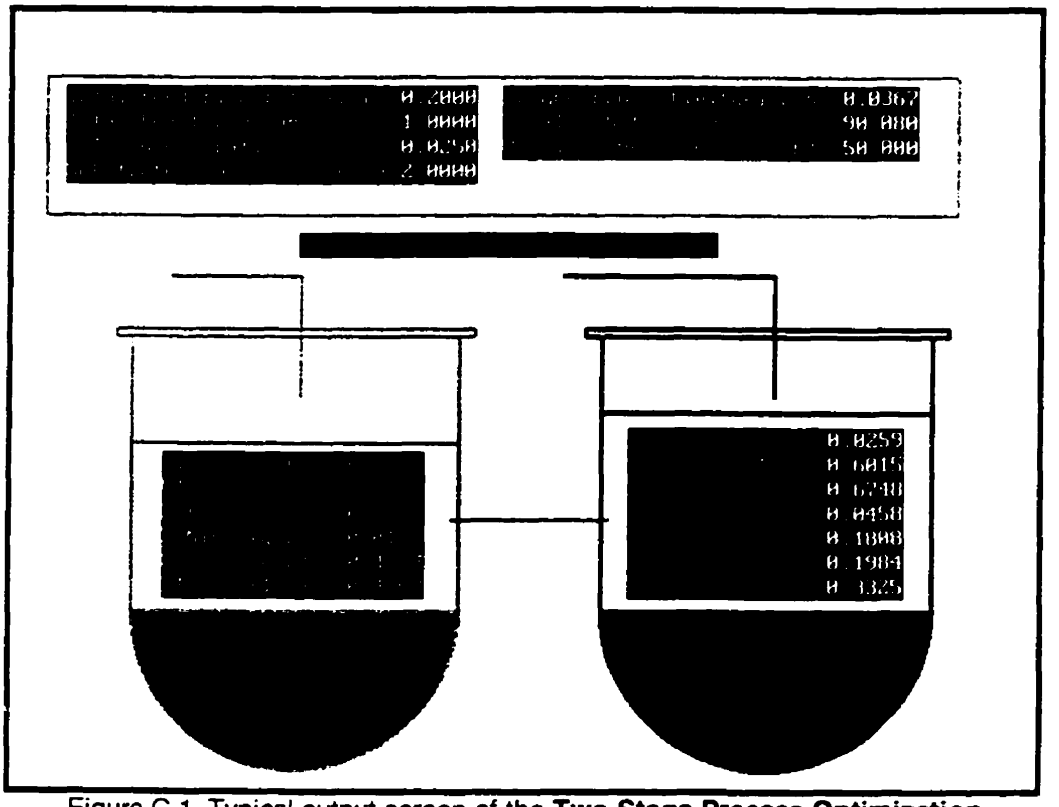


Figure C.1. Typical output screen of the **Two Stage Process Optimization Software**.

LIST OF THE COMPUTER PROGRAM FOR TWO STAGE CFSTR OPTIMIZATION

```
(*-------------------------------------------------------------------------------*)
  Program NLR;
  {$N+}
  Uses CRT,Dos,graph,grfcfstr;
  Type   Array50=Array [1..10] of real;
       Array50i=Array [1..25] of integer;
       real=Extended;
       Obj=Procedure (Var F:Real );
       Type Array5=Array [0..5] of real;
  Var   Y,Y0,Hix,Hiy:Array50;
    Xp,Xm,S,Pp,P,Ie,Ii,Ipi,DXp,DXm,DS,Dpp,Dli,Dlpi,
    F,F0,V,DXp0,DXm0,DS0,DPp0,Dli0,Dlpi0,Dp,Dp0,Cc:Array5;
    mum,mup,mupmax,mummax,mup0,Kps,Kms,k1,Gf,
    Kp1,Ki,n,k3,A,B,kmu,alfa,beta,k2,pmax,kd,k4,Yxs,t:real;
    RXp,RXm,RS,RPp,Rp,dt,Ptotal:real;
    i,Stg,NStg,Steps:Integer;
    Data:Text;
    sS:char;
(-------------------------------------------------------------------------------===>
Presskey)
Procedure PressKey;
begin
  TextBackground(10);
  GoToxy(30,8);TextColor(14+blink);
  write('!!! Press Any Key to Cont.!!!');
  ss:=readKey;TextColor(10);
  Textbackground(0);ClrScr;
End;
(-------------------------------------------------------------------------------====>
WriteR)
Procedure WriteR(PD:real);
Var i,j:integer;
Begin
Gotoxy(5+27,2); write(Ft:6:4);
Gotoxy(5+27,3); write(Vt:6:4);
Gotoxy(5+27,4); write(IPTG:6:4);
Gotoxy(5+27,5); write(S[0]:6:4);
```

```
Gotoxy(67 ,3); write((S[0]-S[2])/S[0]*100:6:3);
Gotoxy(67 ,2); write(-PD:6:4);
Gotoxy(67,4); write(hours:6:3);
Gotoxy(67,5); write('');
TEXTCOLOR(10);
Gotoxy(26,17);  write(F[1]:8:4);
Gotoxy(26,18);  write(V[1]:8:4);
Gotoxy(26 ,19); write(Xp[1]:8:4);
Gotoxy(26 ,20); write(Xm[1]:8:4);
Gotoxy(26 ,21); write(S[1]:8:4);
Gotoxy(26 ,22); write(F[1]/V[1]:8:4);
TEXTCOLOR(14);
Gotoxy(64,16); write((Ft-F[1]):8:4);
Gotoxy(64,17); write(V[2]:8:4);
Gotoxy(64,18); write(XP[2]:8:4);
Gotoxy(64,19); write(XM[2]:8:4);
Gotoxy(64,20); write(Pp[2]:8:4);
Gotoxy(64,21); write(S[2]:8:4);
Gotoxy(64,22); write(F[2]/V[2]:8:4);
TextBackground(7);
GoToxy(24,8);TextColor(14);
write('!!! Optimization in Progress !!!');
Textbackground(0);
End;
(-------------------------------------------------------------------------====>
Optimization code)
Procedure Optimize(ObjFunc:obj;
                   var C,Guess:Array50;UpL:Array50;
                   Nmax:integer;Tol:Real       );
  Var  i,j:integer;
       A,B,Cc,D,E,LOL,Sw:Array50;
       N,Nn: Array [1..10] of integer;
       SwDiv,SwRun,F,FF,SwPm,PMCount:Real;
       s:char;
(-------------------------------------------------------------------------====>
Range check)
Procedure RangeCheck(j:integer);
Begin
  IF (C[j]<0)    then C[j]:=0.0005;
End;
```

```
(-------------------------------------------------------------------------====>
Smoothing)
Procedure Smooth(var C:array50);
var k:integer;
   Sv,Sf:real;
Begin
  Sv:=0;Sf:=0;
  For k:=1 to (Nmax div 2 +1) do begin Sv:=Sv+C[2*k-
1];Sf:=Sf+C[2*k];end;
  For k:=1 to (Nmax div 2 +1) do begin C[2*k-1]:=C[2*k-
1]/Sv;C[2*K]:=C[2*k]/Sf;end;
End;
(-------------------------------------------------------------------------====> opt. var.
initialization)
Procedure InData;
var i:integer;
begin
    Smooth(Guess);
    For i:=1 to Nmax+2 do begin
      C[i]:=Guess[i];E[i]:=0.000911;
      A[i]:=C[i];B[i]:=C[i];
      Cc[i]:=C[i];N[i]:=1;
      Sw[i]:=0;Nn[i]:=0;
    end;
    E[1]:=0.000911;E[2]:=0.00011 ;E[3]:=0.0011;
    E[6]:=0.0003  ;E[5]:=0.000511;E[4]:=0.000211;
    PMCount:=0;
end;
(*-------------------------------------------------------------==>Explore-*)
Procedure Explore;
Label 10;
var j:integer;
 Begin
   For j:=1 to Nmax do begin
10:    C[j]:=Cc[j]+N[j]*E[j];
       RangeCheck(j);
       Smooth(C);
       ObjFunc(FF);

       IF FF<F Then begin F:=FF;Sw[j]:=0;Nn[j]:=Nn[j]+1;end
       Else
```

```
          IF (FF>=F) and (Sw[j]=0) then
                begin
                  N[j]:=-N[j];Sw[j]:=1;
                  Nn[j]:=0; Goto 10;
                end
          Else
                begin
                  C[j]:=Cc[j];
                end;
    End;
End;
(-------------------------------------------------------------------------====>
Explor ckeck)
Procedure ExplorCheck;
var j:integer;
Begin
 SwDiv:=1;  SwRun:=0;
 For j:=1 to Nmax do begin
    IF Sw[j]=0 then begin SwDiv:=0;SwPM:=0; end Else Sw[j]:=0;
    IF (Nn[j]>=2) and (E[j]<0.2) then begin E[j]:=2*E[j];Nn[j]:=1;end;
    IF E[j]> Tol Then SwRun:=1;
 End;
 WriteR(F);
 IF (SwRun=0) or (F=0) then
   begin
     TextBackground(7);
     GoToxy(24,8);TextColor(13);
     write('!!!  Optimization Completed  !!!');
     Textbackground(0);
     ReadIn;
     exit;
   end;
 IF (SwDiv=1) and (SwPM=0) then
        begin
          For j:= 1 to Nmax do E[j]:=E[j]/2;
          Explore;  ExplorCheck;
        end;

 IF (SwDiv=1) and (SwPM=1) then
         begin
          For j:= 1 to Nmax do begin C[j]:=B[j];A[j]:=B[j];Cc[j]:=B[j];end;
```

```
            SwPM:=0;  Explore ;  ExplorCheck;
          end;
End;
(-------------------------------------------------------------------------====>
Pattern Move)
Procedure PatternMove;
var j:integer;
Begin
  For j:=1 to Nmax do begin
     A[j]:=B[j];   B[j]:=C[j];
     C[j]:=A[j]+2*(B[j]-A[j]);
     RangeCheck(j);
  end;
 Smooth(C);  SwPM:=1;
 For j:=1 to Nmax do Cc[j]:=C[j];
End;
(-------------------------------------------------------------------------====> Main
Opt. procedure)
Begin
  InData    ;SwPM:=0;
  ObjFunc(F) ;
  Repeat
     Explore;
     ExplorCheck;
     IF (SwRun=0) or (F=0) then
              begin
                For j:=1 to Nmax do Guess[j]:=C[j];
                Exit;
              end;
     PatternMove;
  Until (F=0) ;
End;
(-------------------------------------------------------------------------====>
Simulation)
(*-----------------------------------------==> Power Func.          *)
Function Pxy(x,y:Real):Real;
Begin
  If x=0 then Pxy:=0 else Pxy:=Exp(Y*Ln(X))  end;
(-----------------------------------------------====> Kinetic Constants -
Initial Values)
Procedure  Initials;
```

```
Var i,j:integer;
Begin
   mupmax:=0.75    ;mummax:=0.76     ;Kps:=0.3    ;Kms:=0.3  ;
   k1:=0.78        ;Kp1:=0.15        ;Ki:=0.012   ;n:=2.5    ;
   k3:=0.27        ;A:=1             ;B:=0.12     ;kmu:=0.02 ;
   alfa:=0.08      ;beta:=0.01       ;k2:=19      ;pmax:=0.4 ;
   kd:=0.01        ;k4:=0.067        ;Yxs:=0.4    ;
   IF S[0]>2 then begin  mupmax:=1.2 ;mummax:=1.2 ;Yxs:=0.32
;end;
(*-------------------------------------------------==> Initials-------*)
   For i:=0 to 5 do
      begin
        Xp[i]  :=0   ;Xm[i]  :=0  ;Pp[i]  :=0;
        Ii[i]  :=0   ;Ipi[i] :=0  ;P[i]   :=0;
        DXp0[i]:=0   ;DXm0[i]:=0  ;
        DS0[i] :=0   ;Dp0[i] :=0  ;DPp0[i]:=0;
        DIi0[i]:=0   ;DIpi0[i]:=0;
      end;
(-------------------------------------------------------------------------====>
Starting points)
   For i:=1 to 5 do
      begin
        S[i] :=0.8;
        Xp[i]:=Yxs*(S[0]-S[i]);
        Xm[i]:=0.015*Xp[i];
      end;
(-----------------------------------------------------------------====> Stage 1
to 4 conditions)
   (*-ONE-*)       (*-TWO-*)      (*-Three-*)     (*-Four-*)   (*-Five*)
   F[0] :=0      ;
   F0[1]:=Ft*Y[1] ; F0[2]:=Ft*Y[3]   ;F0[3]:=Ft*Y[5]  ; F0[4]:=Ft*Y[7] ;
F0[5]:=Ft*Y[9];
   V[1] :=Vt*Y[2] ; V[2] :=Vt*Y[4]   ;V[3]:=Vt*Y[6]   ; V[4] :=Vt*Y[8] ; V[5]
:=Vt*Y[10];
   Ie[1]:=0      ;   Ie[2]:=IPTG     ;Ie[3]:=IPTG     ; Ie[4]:=IPTG    ;
Ie[5]:=IPTG;
F[1] :=F0[1] ; F[2] :=F[1]+F0[2];F[3]:=F[2]+F0[3];
F[4]:=F[3]+F0[4];F[5]:=F[4]+F0[5];
   Cc[1]:=1       ; Cc[2]:=1.       ;Cc[3]:=1       ; Cc[4]:=1       ;Cc[5]:=1
;
End;
```

```
(-------------------------------------------------------------------------====>
Derivatives)
Procedure Derivative(j:Integer);
Var  Iin,Ipin:real;
begin
   Iin   :=Pxy(Ii[j],n); Ipin:=Pxy(Ipi[j],n);
   mup   :=mupmax*S[j]/(Kps+S[j])*(1-k1*Iin/(ki+Iin));
   mum   :=mummax*S[j]/(Kms+S[j])*(1-kp1*Ipin/(ki+Ipin));
(*-----------------------------------------------===>Rate equations *)
   DIi[j] :=A*(Ie[j]/(B+Ie[j]))*(mup/(kmu+mup))-mup*Ii[j];
   DIpi[j]:=A*(Ie[j]/(B+Ie[j]))*(mum/(kmu+mum))-mum*Ipi[j];
   Gf    :=k3/(k4+mup);
   RXp   :=mup*Xp[j]*Cc[j];
   RXm   :=mum*Xm[j]*Cc[j];
   RS    :=-1/Yxs*(RXp+RXm);
   Dp[j]  :=alfa*(1+k2*Iin/(ki+Iin))*(mup+beta)*(1-p[j]/pmax)*Gf-
(mup+Kd)*p[j];
   RPp   :=Cc[j]*Xp[j]*Dp[j]+p[j]*RXp;
(*-----------------------------------------------===> Derivatives..*)
   DXp[j] :=(F[j-1]*Xp[j-1]-F[j]*Xp[j]+RXp*V[j])/V[j];
   DXm[j] :=(F[j-1]*Xm[j-1]-F[j]*Xm[j]+RXm*V[j])/V[j];
   DS[j]  :=(F[j-1]*S[j-1]+F0[j]*S[0]-F[j]*S[j]+RS*V[j])/V[j];
   DPp[j] :=(F[j-1]*Pp[j-1]-F[j]*Pp[j]+RPp*V[j])/V[j];
End;
(-------------------------------------------====> Main-Simulation)
Procedure  Simulate(var PD:real);FAR;
Var i,j:integer;
Fratio:real;
begin
   Initials;
   PD:=0; Ptotal:=0;t:=0;
   For i:=0 to Steps do begin
      t:=t+dt;
      For Stg:=1 to NStg do
         begin
          IF (V[Stg]/Vt >0.01) and (F[Stg]/V[Stg]<1.5) then
            begin
              Derivative(Stg);
              Xp[Stg] :=Xp[Stg] +(DXp[Stg]+DXp0[Stg])  /2*dt;
              Xm[Stg] :=Xm[Stg] +(DXm[Stg]+DXm0[Stg])  /2*dt;
              S[Stg]  :=S[Stg]  +(DS[Stg]+DS0[Stg])    /2*dt;
```

```
                Pp[Stg] :=Pp[Stg] +(DPp[Stg]+DPp0[Stg])  /2*dt;;
                IF Xp[Stg]> 0 then p[Stg]  :=Pp[Stg]/Xp[Stg];
                Ii[Stg] :=Ii[Stg] +(DIi[Stg]+DIi0[Stg])  /2*dt;
                Ipi[Stg]:=Ipi[Stg]+(DIpi[Stg]+DIpi0[Stg])/2*dt;
DXp0[Stg]:=DXp[Stg];DXm0[Stg]:=DXm[Stg];DS0[Stg]:=DS[Stg];
DPp0[Stg]:=DPp[Stg];DIi0[Stg]:=DIi[Stg];DIpi0[Stg]:=DIpi[Stg];
              end
            Else
              begin
                Fratio:=F[Stg-1]/F[Stg];
                Xp[Stg] :=Xp[Stg-1]*Fratio;  Xm[Stg] :=Xm[Stg-1]*Fratio ;
                S[Stg]  :=S[Stg-1]*Fratio;   Pp[Stg] :=Pp[Stg-1]*Fratio ;
                Ii[Stg] :=Ii[Stg-1]*Fratio;  Ipi[Stg]:=Ipi[Stg-1]*Fratio;
                P[Stg]  :=P[Stg-1] ;
              end;
          end;
          IF (Xp[stg]<0) then begin Xp[stg]:=1e-
5;Gotoxy(30,10);write('!-Xp!');end;
          IF (S[stg]<0)  then begin S[stg] :=1e-5;Gotoxy(30,10);write('!-
S !');end;
          IF (Ii[stg]<0) then begin Ii[stg]:=1e-5;Gotoxy(30,10);write('!-
Ii!');end;
          IF t>4 then Ptotal:=Ptotal+Pp[NStg]*dt;
      end;
      PD:=-Ptotal*F[NStg]/Vt/(Hours-4);

end;
(----------------------------------------====>MAIN-Main-main Code)
begin
dt  :=0.0125/4     ;t  :=0;  S[0]  :=Sfeed ;  NStg  :=2
;Steps:=Round(Hours/dt);
(----------------------------====> Initial values for opt. parametrs)
 Y0[1]:=0.3 ; Y0[3]:=0.3 ; Y0[5]:=0.4 ; Y0[7]:=0.6 ;Y0[9] :=0.3 ;
 Y0[2]:=0.3 ; Y0[4]:=0.3 ; Y0[6]:=0.4 ; Y0[8]:=0.6 ;Y0[10]:=0.3 ;
 Optimize(Simulate,Y,Y0,Hiy,(2*NStg-2),0.5e-3);
TextBackground(7);
     GoToxy(24,8);TextColor(13);
     write('!!!  Optimization Completed  !!!');
     Textbackground(0);
     Readln;Closegraph;end.
(----------------------------====> END-End-end of the Main Code)
```

```
*****************************UNIT GRFCFSTR*****************************
 Unit GRFcfstr;
Interface
uses
  Crt, Dos, Graph,Box;
const
  { The five fonts available }
  Fonts : array[0..4] of string[13] =
  ('DefaultFont', 'TriplexFont', 'SmallFont', 'SansSerifFont',
'GothicFont');
  { The five predefined line styles supported }
  LineStyles : array[0..4] of string[9] =
  ('SolidLn', 'DottedLn', 'CenterLn', 'DashedLn', 'UserBitLn');
  { The twelve predefined fill styles supported }
  FillStyles : array[0..11] of string[14] =
  ('EmptyFill', 'SolidFill', 'LineFill', 'LtSlashFill', 'SlashFill',
   'BkSlashFill', 'LtBkSlashFill', 'HatchFill', 'XHatchFill',
   'InterleaveFill', 'WideDotFill', 'CloseDotFill');
  { The two text directions available }
  TextDirect : array[0..1] of string[8] = ('HorizDir', 'VertDir');
  { The Horizontal text justifications available }
  HorizJust  : array[0..2] of string[10] = ('LeftText', 'CenterText',
'RightText');
  { The vertical text justifications available }
  VertJust   : array[0..2] of string[10] = ('BottomText', 'CenterText',
'TopText');
var
  GraphDriver : integer;  { The Graphics device driver }
  GraphMode   : integer;  { The Graphics mode value }
  MaxX, MaxY  : word;     { The maximum resolution of the screen }
  ErrorCode   : integer;  { Reports any graphics errors }
  MaxColor    : word;     { The maximum color value available }
  OldExitProc : Pointer;  { Saves exit procedure address }
  Xcenter,Ycenter:integer;
  Csize,I     : integer;
  S:char;
  Sfeed,Ft,Vt,Hours,IPTG:real;
{$F+}
 implementation
```

```
(-------------------------------------------------------------------------====>
Presskey)
Procedure PressKey;
begin
  TextBackground(10);
  GoToxy(15,22);TextColor(14+blink);
  write('!!! Data Input Completed, Press Any Key to Cont. !!!');
  s:=readKey;TextColor(10);
  Textbackground(15);ClrScr;
End;
(-------------------------------------------------------------------------====>
MyExitProc)
Procedure MyExitProc;
begin
 {ExitProc := OldExitProc}; { Restore exit procedure address }
 CloseGraph;            { Shut down the graphics system }
 Halt;
end; { MyExitProc }
{$F-}
Procedure frame;
Begin
  MakeBox(1,1,80,24);
end;
(-------------------------------------------------------------------------====>
Initialize)
Procedure Initialize;
{ Initialize graphics and report any errors that may occur }
var
 InGraphicsMode : boolean; { Flags initialization of graphics mode }
 PathToDriver   : string;  { Stores the DOS path to *.BGI & *.CHR }
begin
 { when using Crt and graphics, turn off Crt's memory-mapped writes
}
 DirectVideo := False;
 OldExitProc := ExitProc;            { save previous exit proc }
 ExitProc := @MyExitProc;              { insert our exit proc in chain }
 PathToDriver := 'c:\tp\bgi';
 Repeat
{$IFDEF Use8514}                     { check for Use8514 $DEFINE }
  GraphDriver := IBM8514;
  GraphMode := IBM8514Hi;
```

```
{$ELSE}
   GraphDriver := Detect;              { use autodetection }
{$ENDIF}
   InitGraph(GraphDriver, GraphMode, PathToDriver);
   ErrorCode := GraphResult;           { preserve error return }
   if ErrorCode <> grOK then           { error? }
   begin
     Writeln('Graphics error: ', GraphErrorMsg(ErrorCode));
     if ErrorCode = grFileNotFound then  { Can't find driver file }
     begin
       Writeln('Enter full path to BGI driver or type <Ctrl-Break> to
quit:');
       Readln(PathToDriver);
       Writeln;
     end
     else
       Halt(1);                      { Some other error: terminate }
   end;
  until ErrorCode = grOK;

  Randomize;              { init random number generator }
  MaxColor := GetMaxColor;  { Get the maximum allowable drawing
color }
  MaxX := GetMaxX;        { Get screen resolution values }
  MaxY := GetMaxY;
end; { Initialize }
(-------------------------------------------------------------------------====>
Def.colors)
procedure DefaultColors;
{ Select the maximum color in the Palette for the drawing color }
begin
  SetColor(MaxColor);
end; { DefaultColors }
(-------------------------------------------------------------------------====>
DrawBorders)
Procedure DrawBorder;
{ Draw a border around the current view port }
var
  ViewPort : ViewPortType;
begin
  DefaultColors;
```

```
  SetLineStyle(SolidLn, 0, ThickWidth);
  GetViewSettings(ViewPort);
  setcolor(10);
  with ViewPort do
    Rectangle(0, 0, x2-x1, y2-y1);
end; { DrawBorder }
(-----------------------------------------------------====> WaitToGo)
Procedure WaitToGo;
{ Wait for the user to abort the program or continue }
const  Esc = #27;
var  Ch : char;
begin
  Writeln('Esc aborts or press a key...');
  repeat until KeyPressed;
  Ch := ReadKey;
  if ch = #0 then ch := readkey;      { trap function keys }
  if Ch = Esc then
    Halt(0)                       { terminate program }
  else
    ClearDevice;                    { clear screen, go on with demo }
end; { WaitToGo }
(-----------------------------------------------------====> MAIN-Main)
Begin
Initialize;
DrawBorder;
Textcolor (10);
setcolor(10);
SetTextStyle(TriplexFont, HorizDir, 6);
OutTextXY(20,10, '   Two-Stage CFSTR');
SetTextStyle(TriplexFont, HorizDir, 4);
outTextXY(10,100, '  This program optimizes a Two-Stage');
OutTextXY(10,140, 'CFSTR process for the volume of bio-');
OutTextXY(10,180, '-reactors  and  distribution of feed');
OutTextXY(10,220, 'to each bioreactor. Kinetic data are');
OutTextXY(10,260, 'for E.coli BL21DE3 Sensitive to IPTG.');
OutTextXY(10,300, '');
OutTextXY(10,360, ' **H.R. Baheri, G.A. Hill and W.J. Roesler* ');
OutTextXY(10,400, '              1997      ');
Waittogo;CloseGraph;
textbackground(15);textcolor(15);
for i:=1 to 25 do write('                                                    ');
```

```
textcolor(10);
Frame;
TextColor (12);TextBackGround(15);
GoToXY(15,5); write('********** Input Two-Stage CFSTR Data
**********');
GoToXY(15,6); write('----------*----------*----------');
TextColor (10);TextBackGround(15);
GoToXY(15,7); write('Feed Substrate Concentration (2-40
g/L)=');Read(Sfeed);
TextColor (10);
GoToXY(15,8); write('Total Available Volume    ( 1 -  10 L
)=');Read(Vt);
GoToXY(15,9); write('Total Feed Flow rate      (0.2-  5
L/h)=');Read(Ft);
GoToXY(15,10);write('IPTG Concentration     (0.01-0.05
g/L)=');Read(IPTG);

TextBackground(10); TextColor(14);
GoToXY(15,12); write('Recombinant  cultures are dynamic in cell  ');
GoToXY(15,13); write('population,  even in a continuous process. ');
GoToXY(15,14); write('Therefore, the optimization result changes ');
GoToXY(15,15); write('slightly  with the  fermentation  duration.');
TextColor (10);Textbackground(15);
GoToXY(15,17);write('Fermentation Period        (40-100
h)=');Read(Hours);
PressKey;
Initialize;
SetLineStyle(SolidLn,0,ThickWidth);
setcolor (10);
DrawBorder;
Floodfill(20,20,15);
SetLineStyle(SolidLn,0,ThickWidth);
setcolor(10);
Rectangle(20,10,600,100);
Gotoxy(5,2); Writeln('Total Feed Flow Rate (L/h)=');
Gotoxy(5,3); Writeln('Total Reactor Volume (L)  =');
Gotoxy(5,4); Writeln('IPTG Concentration   (g/L)=');
Gotoxy(5,5); Writeln('Substrate Conc.      (g/L)=');
Gotoxy(40,3); Writeln('Final Subst. Conv. (%)    =');
Gotoxy(40,2); Writeln('Enzyme Productivity(g/L.h)=');
Gotoxy(40,4); Writeln('Total Fermen. Period  (h) =');
```

```
TEXTCOLOR(10);
Gotoxy(13,17);  Writeln('F1     (L/h)=');
Gotoxy(13,18);  Writeln('V1     (L)  =');
Gotoxy(5+8,19); Writeln('X Plus (g/) =');
Gotoxy(5+8,20); Writeln('X Minus(g/L)=');
Gotoxy(5+8,21); Writeln('S      (g/L)=');
Gotoxy(5+8,22); Writeln('D1     (1/h)=');
TEXTCOLOR(14);
Gotoxy(50,16); Writeln('F2     (L/h)= ');
Gotoxy(50,17); Writeln('V2     (L)  = ');
Gotoxy(50,18); Writeln('X Plus (g/L)= ');
Gotoxy(50,19); Writeln('X Minus(g/L)= ');
Gotoxy(50,20); Writeln('Enzyme (g/L)= ');
Gotoxy(50,21); Writeln('S      (g/L)= ');
Gotoxy(50,22); Writeln('D2     (1/h)= ');

Xcenter:=480;
Ycenter:=360;
Setcolor (yellow);
SetFillStyle(LtSlashFill,13 );
SetLineStyle(SolidLn,0,ThickWidth);
Rectangle(Xcenter-105,Ycenter-180,Xcenter+105,Ycenter);
rectangle(Xcenter-105,Ycenter-130,Xcenter+105,Ycenter);
rectangle((Xcenter-115) ,(Ycenter-180),(Xcenter+115),(Ycenter-
185));
PieSlice(Xcenter,Ycenter,180,360,105);
Line(485,140,350,140);
Line(485,140,485,220);
Xcenter:=180;
Ycenter:=360;
Csize:=1;
Setcolor (10);
SetFillStyle(LtSlashFill,13 );
SetLineStyle(SolidLn,0,ThickWidth);
Rectangle(Xcenter-105,Ycenter-180,Xcenter+105,Ycenter);
rectangle(Xcenter-105,Ycenter-110,Xcenter+105,Ycenter);
rectangle((Xcenter-115) ,(Ycenter-180),(Xcenter+115),(Ycenter-
185));
PieSlice(Xcenter,Ycenter,180,360,105);
Line(280,300,380,300);
Line(185,140,100,140);Line(185,140,185,220);end.
```

**********UNIT BOX **********

```
(-------------------------------------------------------------------====> Unit
Box)
Unit box;
Interface
Uses CRT;
type Grafrec=Record
       Ul,
      ur,ll,lr,hbar,vbar,
      lc,td,tu,tr,tl:string[4]
    end;
Var grafchars:grafrec;
Procedure MakeBox(x,y,w,h:integer);
implementation
Procedure Define(Var Grafchars:Grafrec);
Begin
  With Grafchars do
      Begin
        ul:=chr(201);
        ur:=chr(187);
        ll:=chr(200);
        lr:=chr(188);
        hbar:=chr(205);
        vbar:=chr(186);
        lc:=chr(206);
        td:=  chr(203);
        tu:=    chr(202);
        tr:=      chr(185);
        tl:=        chr(204);
      end;
 end;
(-------------------------------------------------------------------====>
MakeBox)
Procedure MakeBox(x,y,w,h:integer);
Var i,j:integer;
Begin
 If x<0 then x:=(80-w) div 2;
 With Grafchars do
     begin
```

```
        gotoxy(x,y);Write(ul);
        for i:=3 to w do write(hbar);
        write(ur);
        gotoxy(x,Y+h-1);write(ll);
        for i:=3 to w do write (hbar);
        write(lr);
        for i:=1 to h-2 do begin
                gotoxy(x,y+i);write(vbar);
                gotoxy(x+w-1,y+i);write(vbar);
                end;
    end;
end;

Begin
Define(Grafchars);
end.
```

C.6. Fedbatch Optimization Program

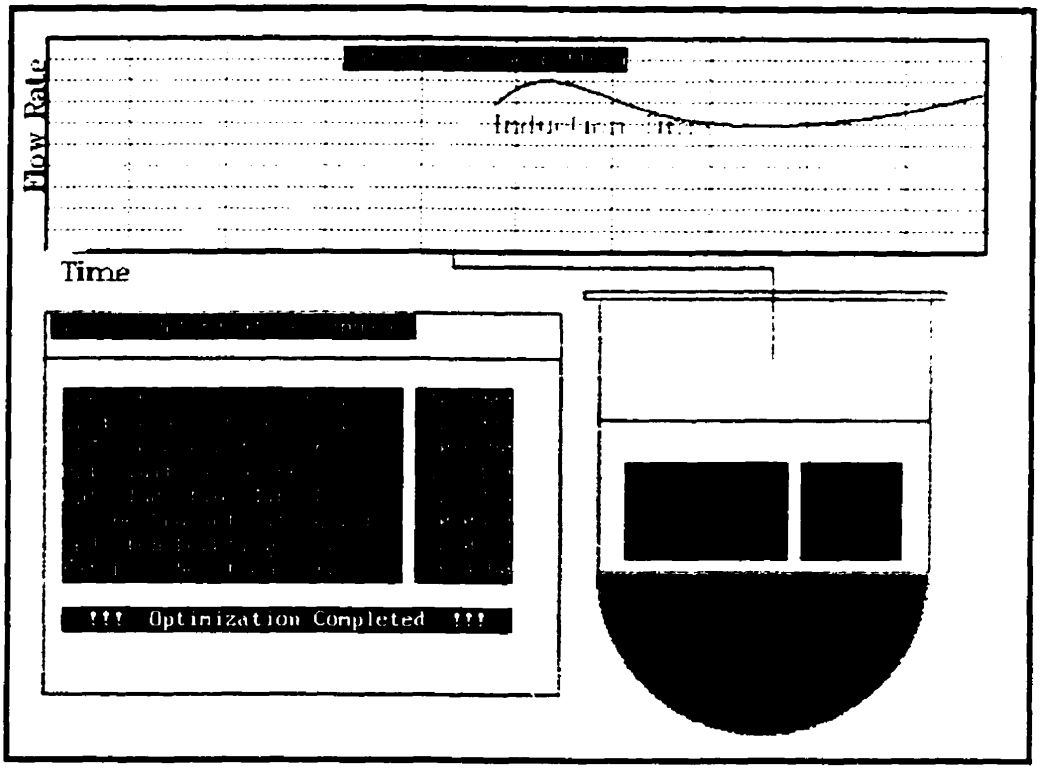


Figure C.2. Typical output screen of the **Fedbatch Optimization Software**.

LIST OF THE COMPUTER PROGRAM FOR FEDBATCH OPTIMIZATION

```
(*-------------------------------------------------------------------------*)
  Program NLR;
  Uses CRT,Dos,Box,Graph,Grffbch;
  Type   Array10=Array [1..10] of real;
         Array50i=Array [1..25] of integer;
         real=Double;
         Obj=Procedure (Var F:Real );
         Type Array5=Array [0..5] of real;
         Array4=Array [0..4] of real;
  Var    Y,Y0,Hix,Hiy:Array10;
        Xp,Xm,S,S0,Pp,P,Ie,Ii,Ipi,DXp,DXm,DS,Dpp,DIi,DIpi,Pps,Ps,
F,F0,V,DXp0,DXm0,DS0,DPp0,DIi0,DIpi0,Dp,Dp0,Cc,DGf0,tbatch,
        mum,mup,mupmax,mummax,mup0,Kps,Kms,k1,Gf,Dgf,
        Kp1,Ki,n,k3,A,B,kmu,alfa,beta,k2,pmax,kd,k4,Yxs,t:real;
        RXp,RXm,RS,RPp,Rp,Hours,dt,Ptotal,Ft,Vt,Xps,Gfs,Viptg:real;
        DV,Stank,
DV0,V0,Vmax,SwIPTG,SwBatch,PD,Fmax,Fmin:real;
        Mode,SwFile:Char;
        Steps,I,Stage,Nstage:Integer;
        Datafile:String;
        Data:Text;
        Flow:Array [0..5000] of real;
(-------------------------------------------------------------------------====> Screen
Frame)
Procedure frame;
Begin
  MakeBox(1,1,80,24);
end;
(-------------------------------------------------------------------------====> Output
Screen)
Procedure WRT;
Var  Xn,Yn,Xold,Yold,j:integer;
begin
  TextColor(10);
  Gotoxy(5+28,16); Writeln(Sfeed:8:4);
  Gotoxy(5+28,17); Writeln(Scharge:8:4);
  Gotoxy(5+28,18); Writeln(IPTG:8:4);
  Gotoxy(5+28,19); Writeln(Y[1]/Vmax*100:8:4);
```

```
   Gotoxy(5+28,20); Writeln(SwIPTG:8:4);
   {Gotoxy(5+28,21); Writeln(S:8:4);}
   Gotoxy(5+28,21); Writeln(Pp/(t-tbatch)*V/(V+Y[1]):8:4);
   Gotoxy(5+28,22); Writeln(t-tbatch:8:4);
   Gotoxy(5+28,23); Writeln(tBatch:8:4);
 Textcolor(12);
   Gotoxy(50+14,19); Writeln(Xp:8:4);
   Gotoxy(50+14,20); Writeln(Xm:8:4);
   Gotoxy(50+14,21); Writeln(Pp:8:4);
   Gotoxy(50+14,22); Writeln(S:8:4);
   GoToxy(5,25);Textbackground(13);TextColor(15);
   Write('  !! Optimization in Progress !!  ');
   setcolor(blue);
   IF SWfile='Y' then
     begin
       Xold:=20;Yold:=150;
       SetLineStyle(SolidLn,0,ThickWidth);
       For j:= 1 to round((t-tbatch)/dt) do
          begin
            Xn:=20+Round((j*dt/(t-tbatch)*600));
            Yn:=150-Round(((Flow[j]-Fmin)/(1.25*(Fmax-Fmin))*140));
 IF   j=Round(SwIPTG/dt)+5 then
                 begin
                   Setcolor(10);
                   SetTextStyle(TriplexFont, HorizDir,1);
                   OutTextXY(Xn, Yn, 'Induction Time');
                 end;
 Line(Xold,Yold,Xn,Yn);
            Xold:=Xn;Yold:=Yn;
          end;
 SetTextStyle(defaultfont, HorizDir,1);
        GoToxy(5,25);Textbackground(13);TextColor(15);
        Write('  !!!  Optimization Completed  !!!  ');
     end;
 end;
 (-------------------------------------------------------------------====> MAIN-
 Main)
 Procedure Optimize(ObjFunc:obj;
                 var C,Guess:Array10;UpL:Array10;
                 Nmax:integer;Tol:Real      );
 Var  i,j:integer;
```

```
      A,B,Cc,D,E,LOL,Sw:Array10;
      N,Nn: Array [1..10] of integer;
      SwDiv,SwRun,F,FF,SwPm,PMCount:Real;
      s:char;
 (-------------------------------------------------------------------====>
Rangecheck)
Procedure RangeCheck(j:integer);
Begin
  IF (C[j]<0)    then C[j]:=0.001;
  IF C[j]>Hiy[j]  then C[j]:=0.99*Hiy[j];
End;
(-------------------------------------------------------------------====>
Presskey)
Procedure PressKey;
begin
  GoToxy(5,25);Textbackground(13);TextColor(15+blink);
  Write('  !!! Optimization Completed !!!  ');
  s:=readKey;TextColor(10);
  Textbackground(0);ClrScr;
End;
(-------------------------------------------------------------------====>Opt.
Initialization)
Procedure InData;
var i:integer;
begin

    For i:=1 to Nmax do begin
       C[i]:=Guess[i];E[i]:=0.000911;
       A[i]:=C[i];B[i]:=C[i];
       Cc[i]:=C[i];N[i]:=1;
       Sw[i]:=0;Nn[i]:=0;
    end;
    E[1]:=0.011  ;E[2]:=0.01 ;E[3]:=0.01511;
    E[6]:=0.0003  ;E[5]:=0.000511;E[4]:=0.000211;
    PMCount:=0;
end;
(-------------------------------------------------------------------====>Explore)
Procedure Explore;
Label 10;
var j:integer;
  Begin
```

```
   For j:=1 to Nmax do begin
10:    C[j]:=Cc[j]+N[j]*E[j];
       RangeCheck(j);
       ObjFunc(FF);
       IF FF<F Then begin F:=FF;Sw[j]:=0;Nn[j]:=Nn[j]+1;end
       Else
           IF (FF>=F) and (Sw[j]=0) then
                 begin
                   N[j]:=-N[j];Sw[j]:=1;
                   Nn[j]:=0; Goto 10;
                 end
           Else
                 begin
                   C[j]:=Cc[j];
                 end;
   End;
End;
(-------------------------------------------------------------------====>
ExploreCheck)
Procedure ExplorCheck;
var j:integer;
Begin
 SwDiv:=1;  SwRun:=0;
 For j:=1 to Nmax do begin
    IF Sw[j]=0 then begin SwDiv:=0;SwPM:=0; end Else Sw[j]:=0;
    IF (Nn[j]>=2) then begin E[j]:=2*E[j];Nn[j]:=1;end;
    IF E[j]> Tol Then SwRun:=1;
 End;
 IF (SwRun=0) or (F=0) then begin ;exit;end;
  IF (SwDiv=1) and (SwPM=0) then
        begin
          For j:= 1 to Nmax do E[j]:=E[j]/2;
          Explore;  ExplorCheck;
        end;
 IF (SwDiv=1) and (SwPM=1) then
         begin
          For j:= 1 to Nmax do begin C[j]:=B[j];A[j]:=B[j];Cc[j]:=B[j];end;
          SwPM:=0;  Explore ;  ExplorCheck;
        end;
End;
```

```
(-------------------------------------------------------------------====> Pattern
Move)
Procedure PatternMove;
var j:integer;
Begin
  For j:=1 to Nmax do begin
     A[j]:=B[j];   B[j]:=C[j];
     C[j]:=A[j]+2*(B[j]-A[j]);
     RangeCheck(j);
  end;
  SwPM:=1;
  For j:=1 to Nmax do Cc[j]:=C[j];
End;
(-----------------------------------------====> Main-OPTIMIZATION Code)
Begin
  InData    ;SwPM:=0;
  ObjFunc(F) ;
  Repeat
     Explore;
     ExplorCheck;
     IF (SwRun=0) or (F=0) then
            begin
              For j:=1 to Nmax do Guess[j]:=C[j];
              Swfile:='Y';
              ObjFunc(FF);
              readln;
              Exit;
            End;
      PatternMove;
  Until (F=0) ;
End;
(-------------------------------------------------------------------
====>SIMULATION)
(*-----------------------------------------==> Power Func.*)
Function Pxy(x,y:Real):Real;
Begin If x=0 then Pxy:=0 else Pxy:=Exp(Y*Ln(X))  end;
(-----------------------------------------------------====>Kinetic data-
Initialization)
Procedure  Initials;
Var i,j:integer;
Begin
```

```
(*-------------------------------------------------==> Data - Cons.---*)
   mupmax:=0.75    ;mummax:=0.76     ;Kps:=0.3    ;Kms:=0.3  ;
   k1:=0.78        ;Kp1:=0.15        ;Ki:=0.012   ;n:=2.5    ;
   k3:=0.27        ;A:=1             ;B:=0.12     ;kmu:=0.02 ;
   alfa:=0.08      ;beta:=0.01       ;k2:=19      ;pmax:=0.4 ;
   kd:=0.01        ;k4:=0.067        ;Yxs:=0.4    ;
(*-------------------------------------------------==> Initials*)
   Pp:=0   ;p:=0 ; Ii:=0; Ipi:=0  ; Ie :=0  ;t :=0; Viptg:=0;
   Tbatch:=0;
   DXp0:=0 ; DXm0:=0  ;  DS0:=0  ; Dp0:=0  ;DPp0:=0;
   DIi0:=0 ; DIpi0:=0 ;  DV0:=0  ;DGf:=0   ;DGf0:=0;
   V0:=0.02   ; Vmax:=1 ;Gf:=4; Mode :='B' ;S0:=Scharge;
   Fmax:=0;Fmin:=100;
(*-------------------------------------------------==>Initial Xp ----*)
   V :=Vmax+V0;S:= S0*(Vmax/V);
   Xp:=V0*Yxs*2/V ;Xm:=0.015*Xp    ;
   IF S0>2 then begin mupmax:=1.2 ;mummax:=1.22; Yxs:=0.32 ;
end;
End;
(*-------------------------------------------------==>Derivative------*)
Procedure Derivative    ;
Var  Iin,Ipin:real;
begin
   Iin    :=Pxy(Ii,n); Ipin:=Pxy(Ipi,n);
   mup    :=mupmax*S/(Kps+S)*(1-k1*Iin/(ki+Iin));
   mum    :=mummax*S/(Kms+S)*(1-kp1*Ipin/(ki+Ipin));

(*-------------------------------------------------===>Rate equations *)
   DIi :=A*(Ie/(B+Ie))*(mup/(kmu+mup))-mup*Ii;
   DIpi:=A*(Ie/(B+Ie))*(mum/(kmu+mum))-mum*Ipi;
   DGf:=K3-(k4+mup)*Gf;
   Gf:=Gf+(DGf+DGf0)/2*dt;

   RXp :=mup*Xp;
   RXm :=mum*Xm;
   RS  :=-1/Yxs*(RXp+RXm);
   Dp  :=alfa*(1+k2*Iin/(ki+Iin))*(mup+beta)*(1-p/pmax)*Gf-
(mup+Kd)*p;
   RPp :=Xp*Dp+p*RXp;
   (*-------------------------------------------------===> Derivatives*)
   IF Mode='F' then DV  :=-RS*V/(S0-S) else DV:=0;
```

```
    DXp :=(RXp*V-Xp*DV)/V;
    DXm :=(RXm*V-Xm*DV)/V;
    DS  :=(RS*V-S*DV)/V;
    DPp :=(RPp*V-Pp*DV)/V;
End;
(*-------------------------------------------------===> Simulate*)
Procedure   Simulate(var PD:real);FAR;
Var i,j:integer;
    C,CI:Real;
begin
    Initials;
      For i:=0 to Steps do
       begin
        t:=t+dt;

        IF ((t-tbatch)>SwIPTG) and (Mode='F')  then Ie:=IPTG;
         IF ((Scharge=2) and (S0>2) and (Mode='F')) then
           begin
            IF (t-tbatch)<0.25 then
              begin  mupmax:=0.75;mummax:=0.76; end
            Else
              begin  mupmax:=1.2;mummax:=1.22;  end;
       end;
       Derivative;
       V  :=V  +(DV+DV0)   /2*dt;   DV0:=DV                ;
       Xp :=Xp +(DXp+DXp0)  /2*dt;   Xm :=Xm +(DXm+DXm0)
/2*dt;
       S  :=S  +(DS+DS0)   /2*dt;   IF Mode='F' then S:=Stank ;
       Pp :=Pp +(DPp+DPp0)  /2*dt;   Ii :=Ii +(DIi+DIi0)  /2*dt;
       Ipi:=Ipi+(DIpi+DIpi0)/2*dt;   P  :=P  +(DP+  DP0)  /2*dt;
    IF (SwFile='Y') and (Mode='F') then
      begin
       {WriteIn (data, (t-Tbatch)*60:8:4,Xp+Xm:5:2,V:8:4,'
',(DV+DV0)/2*        1000/60:8:4);}
        Flow[round((t-tbatch)/dt)]:=(Dv+DV0)/2;
        IF Flow[round((t-tbatch)/dt)]>Fmax then Fmax:=Flow[round((t-
tbatch)/dt)];
        IF Flow[round((t-tbatch)/dt)]<Fmin then Fmin:=Flow[round((t-
tbatch)/dt)];
       end;
```

```
    DXp0:=DXp;DXm0:=DXm;DS0:=DS; DPp0:=DPp;Dli0:=Dli;
       Dlpi0:=Dlpi;Dp0:=Dp;
    IF ((t-tbatch)>SwIPTG) and (Mode='F') and (Viptg=0)  then
Viptg:=V;;

       IF (Mode='B') and (S<=Y[3]) then
          begin
            Cl:=Xp/(Xp+Xm);
            V:=Y[1];SwIPTG:=Y[2] ; Stank:=Y[3];
            S:=Stank;
            Xp:=(Cl)*(S0-S)*Yxs;
            Xm:=(1-Cl)*(S0-S)*Yxs;
            Ps:=P; Pps:=Xp*P;
            S0:=Sfeed;
            Mode:='F';tbatch:=t;
            Xps:=Xp;Gfs:=Gf;
          End;
       IF (Mode='F') and (V>=Vmax) then
          begin
            Hiy[2]:=t-tbatch+1;
            PD:=-(Pp-Pps)/(t-tbatch)*V/(V+Y[1]);
            WRT; exit;
          end;
    end;
    PD:=-(Pp-Pps)/(t-tbatch)*V/(V+Y[1]);
    WRT;
end;
(*-------------------------------------------------===> Fedbatch Opt.-Main-
Code)
begin
   SwFile:='N';
   Steps:=29999  ; dt  :=0.0125/2 ;SwBatch:=100 ;SwIPTG:=100;
(*--------------------===> Initial Values for the Fedbatch Paramaetres)
   { V0                SwIPTG              Stank  }
    Y0[1] :=0.4315 ; Y0[2] :=0.6453 ; Y0[3] :=0.615;
    Hiy[1]:=1      ; Hiy[2]:=25     ; Hiy[3]:=Scharge ;
    Optimize(Simulate,Y,Y0,Hiy,3,1e-3);
ClearDevice;
End.
(*-------------------------------------------------===> Fedbatch Opt.-End of
Main-Code)
```

```
*****************************UNIT GRFFBCH*************************
Unit Grffbch;
INTERFACE
uses  Crt, Dos, Box,Graph;
const  { The five fonts available }
  Fonts : array[0..4] of string[13] =
  ('DefaultFont', 'TriplexFont', 'SmallFont', 'SansSerifFont',
'GothicFont');
  { The five predefined line styles supported }
  LineStyles : array[0..4] of string[9] =
  ('SolidLn', 'DottedLn', 'CenterLn', 'DashedLn', 'UserBitLn');
  { The twelve predefined fill styles supported }
  FillStyles : array[0..11] of string[14] =
  ('EmptyFill', 'SolidFill', 'LineFill', 'LtSlashFill', 'SlashFill',
   'BkSlashFill', 'LtBkSlashFill', 'HatchFill', 'XHatchFill',
   'InterleaveFill', 'WideDotFill', 'CloseDotFill');
  { The two text directions available }
  TextDirect : array[0..1] of string[8] = ('HorizDir', 'VertDir');
  { The Horizontal text justifications available }
  HorizJust  : array[0..2] of string[10] = ('LeftText', 'CenterText',
'RightText');
  { The vertical text justifications available }
  VertJust   : array[0..2] of string[10] = ('BottomText', 'CenterText',
'TopText');
var
  GraphDriver : integer;  { The Graphics device driver }
  GraphMode   : integer;  { The Graphics mode value }
  MaxX, MaxY  : word;     { The maximum resolution of the screen }
  ErrorCode   : integer;  { Reports any graphics errors }
  MaxColor    : word;     { The maximum color value available }
  OldExitProc : Pointer;  { Saves exit procedure address }
  Xcenter,Ycenter:integer;
  Csize,I     : integer;
  Sfeed,Scharge, IPTG:Real;
 Procedure MyexitProc;
 {$F+}
 Implementation
(*-------------------------------------------------------------Myexit-*)
Procedure MyExitProc;
begin
  Cleardevice;
```

```
  ExitProc := OldExitProc; { Restore exit procedure address }
  CloseGraph;              { Shut down the graphics system }
end; { MyExitProc }
{$F-}
(*-------------------------------------------------------Initialize-*)
Procedure frame;
Begin
  MakeBox(1,1,80,24);
end;

Procedure Initialize;
{ Initialize graphics and report any errors that may occur }
var
  InGraphicsMode : boolean; { Flags initialization of graphics mode }
  PathToDriver   : string;  { Stores the DOS path to *.BGI & *.CHR }
begin
  { when using Crt and graphics, turn off Crt's memory-mapped writes
}
  DirectVideo := False;
  OldExitProc := ExitProc;              { save previous exit proc }
  ExitProc := @MyExitProc;              { insert our exit proc in chain }
  PathToDriver := 'c:\tp\bgi';
  repeat
{$IFDEF Use8514}                        { check for Use8514 $DEFINE }
    GraphDriver := IBM8514;
    GraphMode := IBM8514Hi;
{$ELSE}
    GraphDriver := Detect;              { use autodetection }
{$ENDIF}
    InitGraph(GraphDriver, GraphMode, PathToDriver);
    ErrorCode := GraphResult;           { preserve error return }
    if ErrorCode <> grOK then           { error? }
    begin
      WriteIn('Graphics error: ', GraphErrorMsg(ErrorCode));
      if ErrorCode = grFileNotFound then  { Can't find driver file }
      begin
        WriteIn('Enter full path to BGI driver or type <Ctrl-Break> to
quit:');
        ReadIn(PathToDriver);
        WriteIn;
      end
```

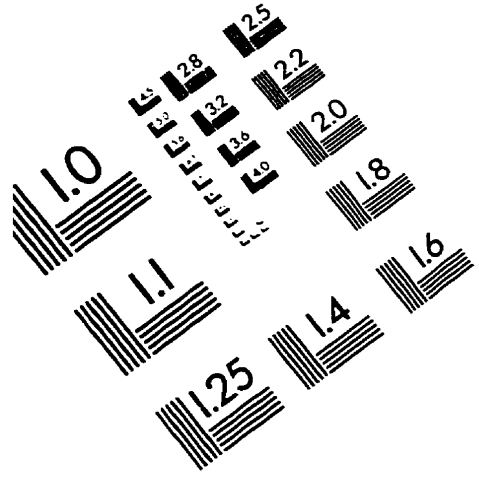
```
    else
      Halt(1);                    { Some other error: terminate }
    end;
  until ErrorCode = grOK;
  Randomize;              { init random number generator }
  MaxColor := GetMaxColor;  { Get the maximum allowable drawing
color }
  MaxX := GetMaxX;          { Get screen resolution values }
  MaxY := GetMaxY;
end; { Initialize }
(*-------------------------------------------------------Drawborder--*)
procedure DefaultColors;
{ Select the maximum color in the Palette for the drawing color }
begin
  SetColor(MaxColor);
end; { DefaultColors }

Procedure DrawBorder;
{ Draw a border around the current view port }
var
  ViewPort : ViewPortType;
begin
  DefaultColors;
  SetLineStyle(SolidLn, 0, ThickWidth);
  GetViewSettings(ViewPort);
  setcolor(10);
  with ViewPort do
    Rectangle(0, 0, x2-x1, y2-y1);
end; { DrawBorder }
(*-------------------------------------------------------WaittoGo---*)
Procedure PressKey;
var s:string;
begin
  TextBackground(15);
  GoToxy(15,20);TextColor(10+blink);TextBackground(15);
  write('!!! Data Input Completed, Press Any Key to Cont.!!!');
  s:=readKey;TextColor(10);Textbackground(15);
  ClrScr;
End;
Procedure WaitToGo;
{ Wait for the user to abort the program or continue }
```

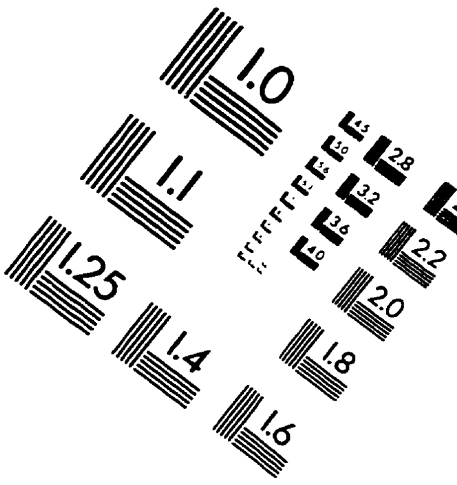
```
const
  Esc = #27;
var
  Ch : char;
begin
  Textcolor(15);
  Writeln('Esc aborts or press a key...');
  repeat until KeyPressed;
  Ch := ReadKey;
  if ch = #0 then ch := readkey;      { trap function keys }
  if Ch = Esc then
    Halt(0)                          { terminate program }
  else
    ClearDevice;                     { clear screen, go on with demo }
end; { WaitToGo }
 (*----------------------------Main  Main----------------------------*)
Begin
Initialize;
SetLineStyle(SolidLn,0,ThickWidth);
DrawBorder;
setcolor(10);
SetTextStyle(TriplexFont, HorizDir, 6);
OutTextXY(20,10, 'FedBatch Optimization   ');
SetTextStyle(TriplexFont, HorizDir, 4);
outTextXY(10,100, 'This program  optimizes  a fedbatch');
OutTextXY(10,140, 'process  for  the  proper induction');
OutTextXY(10,180, 'time, initial bioreactor volume and');
OutTextXY(10,220, 'the in situ substrate concentration.');
OutTextXY(10,260, '(E. coli BL21DE3 sensitive to  IPTG.)');
OutTextXY(10,300, '');
OutTextXY(10,360, ' *****H.R. Baheri, G.A. Hill, & W.J. Roesler** ');
OutTextXY(10,400, '              1997      ');
waittogo;
CloseGraph;
textbackground(15);textcolor(15);
for i:=1 to 25 do write('                                                    ');
Textcolor(14);
Frame;
TextColor (12);TextBackGround(15);
GoToXY(15,5); write('*************** Input Fedbatch Data
*******************');
```

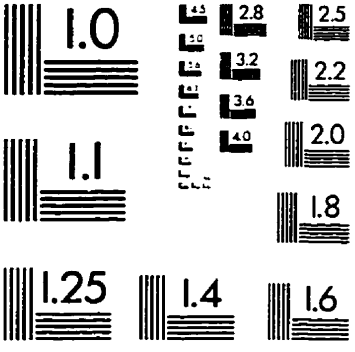
```
GoToXY(15,6); write('--------------*---------------------*-----------------');
TextColor (10);TextBackGround(15);
GoToXY(15,7); write('Feed Substrate Concentration (2-100
g/L)=');Read(Sfeed);
TextColor (10);
GoToXY(15,8); Write('Init. Charge Subs. Conc.    (2-100
g/L)=');Read(SCharge);
TextColor (10);
GoToXY(15,9); write('Inducer Concentration (0.01-0.05
g/L)=');Read(IPTG);
PressKey;
Initialize;
setcolor(10);
SetLineStyle(SolidLn,0,ThickWidth);
DrawBorder;
Floodfill(20,20,15);
SetLineStyle(SolidLn,0,ThickWidth);
setcolor(10);
Rectangle(20,190,350,440);
Rectangle(20,190,350,220);
TextColor(10);textbackground(15);
Gotoxy(4,13);Writeln('Process Optimization Summary ');
Gotoxy(5,16); Writeln('Feed Conc.      (g/L) =');
Gotoxy(5,17); Writeln('Init. Charge Conc. (g/L) =');
Gotoxy(5,18); Writeln('IPTG Concentration (mg/L) =');
Gotoxy(5,19); Writeln('Opt. Initial Volume( %)  =');
Gotoxy(5,20); Writeln('Opt. Induction Time(h)   =');
Gotoxy(5,21); Writeln('Enzyme Productivity(g/L.h)=');
Gotoxy(5,22); Writeln('FedBatch Duration  (h)  =');
Gotoxy(5,23); Writeln('Batch  Duration  (h)  =');
textcolor(12);
Gotoxy(50,19); Writeln('X Plus (g/L)=');
Gotoxy(50,20); Writeln('X Minus(g/L)=');
Gotoxy(50,21); Writeln('Enzyme (g/L)=');
Gotoxy(50,22); Writeln('S     (g/L)=');
setcolor(12);
Rectangle(20,10,620,150);
SetLineStyle(DottedLn,0,NormWidth);
For I:=1 to 10 do
  begin
   Line(10+I*62,150,10+I*62,10);
```

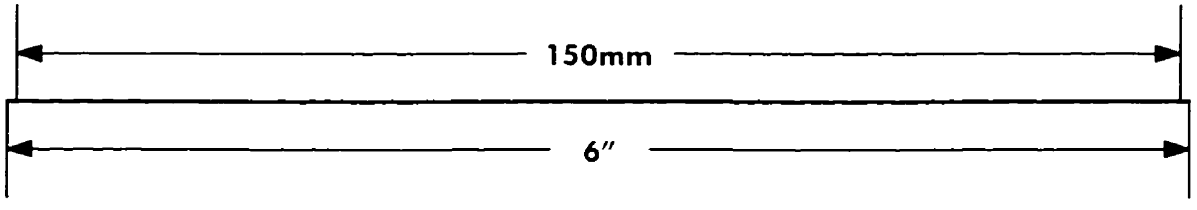
```
    Line(620,10+14*I,20,14*I+10);
  end;
SetTextStyle(TriplexFont,VertDir , 1);
OutTextXY(0,20, 'Flow Rate');
SetTextStyle(TriplexFont, HorizDir, 1);
outTextXY(30,151, 'Time');
Xcenter:=480;
Ycenter:=360;
Csize:=1;
textcolor(10);
Gotoxy(27,2);Writeln('Optimum Feeding Pattern');
Setcolor (10);
SetFillStyle(LtSlashFill,10 );
SetLineStyle(SolidLn,0,ThickWidth);
Rectangle(Xcenter-105,Ycenter-180,Xcenter+105,Ycenter);
rectangle(Xcenter-105,Ycenter-100,Xcenter+105,Ycenter);
rectangle((Xcenter-115) * Csize,(Ycenter-180)Div
Csize,(Xcenter+115)Div Csize,(Ycenter-185)Div Csize);
PieSlice(Xcenter,Ycenter,180,360,105);
Line(485,160,280,160);
Line(485,160,485,220);
Line(280,160,280,150);
setcolor(14);
readln;
end.
```

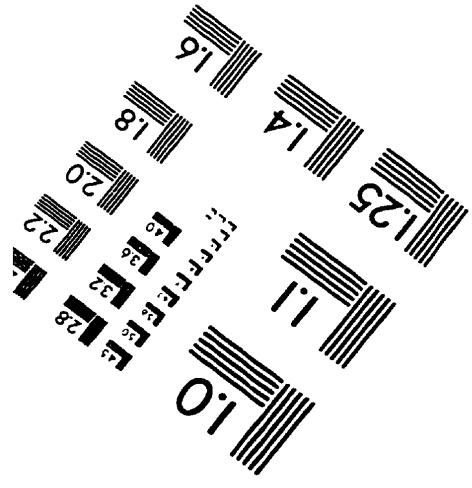
IMAGE EVALUATION
TEST TARGET (QA-3)











APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989