A MULTI-SCALE APPROACH TO FED-BATCH BIOREACTOR CONTROL

by

Abhishek Soni

B.S. in ChE., University of Mumbai (U.D.C.T.), May 2000

Submitted to the Graduate Faculty of

the School of Engineering in partial fulfillment

of the requirement for the degree of

Master of Science in Chemical Engineering

University of Pittsburgh

2002

UNIVERSITY OF PITTSBURGH

SCHOOL OF ENGINEERING

This thesis was presented

by

Abhishek Soni

It was defended on

November 26, 2002

and approved by

Thesis Advisor: Robert S. Parker, Assistant Professor, Chemical and Petroleum Engineering Department

ABSTRACT

A MULTI-SCALE APPROACH TO FED-BATCH BIOREACTOR CONTROL

Abhishek Soni , M.S.

University of Pittsburgh

The rising energy costs, increased global competition in terms of both price and quality, and the need to make products in an environmentally benign manner have paved the way for the biological route towards manufacturing. Many of the products obtained by the biological route either cannot be produced, or are very difficult to obtain, by conventional manufacturing methods. Most of these products fall in the low volume/high value bracket, and it is estimated that the production of therapeutic proteins alone generated sales exceeding $25 billion in 2001 [1]*. By increasing our understanding of these systems it may be possible to avoid some of the empiricism associated with the operation of (fed-)batch bioreactors. Considerable benefit, in terms of reduced product variability and optimal resource utilization could be achieved, and this work is a step in that direction.

Biological reactors typically are governed by highly nonlinear behavior occuring on both a macroscopic reactor scale and a microscopic cellular scale. Reactions taking place at these scales also occur at different rates so that the bioreactor system is multi-scale both spatially and temporally. Since achievable controller performance in a model-based control scheme is dependent on the quality of the process model [2], a controller based on a model that captures events occuring at both the

*Bracketed references placed superior to the line of text refer to the bibliography.

reactor and cellular scales should provide superior performance when compared to a controller that employs a uniscale model. In the model considered for this work [3], μ, the specific growth rate is used as a coupling parameter integrating the behavior of both scales. On the cellular level, flux distributions are used to describe cellular growth and product formation whereas a lumped-parameter reactor model provides the macroscopic process representation.

The control scheme for the fed-batch bioreactor is implemented in two stages, and the substrate feed rate serves as the manipulated variable. Initially, a constrained optimal control problem is solved off-line, in order to determine the manipulated variable profile that maximizes the end of batch product concentration for the product of interest, while maintaining a pre-specified, fixed final volume. The next step involves tracking of the optimal control trajectory, in closed-loop operation. The Shrinking Horizon Model Predictive Control (SHMPC)[4] framework is used to minimize the projected deviations of the controlled variable from the specified trajectories. At every time step, the original nonlinear model is linearized and the optimization problem is formulated as a quadratic program [5], that includes constraints on the manipulated input and the final volume. Finally, the performance of the controller is evaluated, and strategies for disturbance compensation are presented. The results of this approach are presented for ethanol production in a baker's yeast fermentation case study [3].

DESCRIPTORS

Fed-Batch Bioreactors

Multi-scale Models

Model Predictive Control

Optimal Control

ACKNOWLEDGMENTS

First and foremost, I would like to thank my parents without whose love and support, I probably would not have reached this far. Special thanks to all of my teachers at various levels of my education, from whom I have gained more than just academic knowledge. They have positively influenced, and shaped my ideas and made me a better person.

I cannot thank my thesis advisor Prof. Parker, enough; his patience, enthusiasm, an almost contagious positive attitude, and his critical comments are largely responsible for a (mainly) enjoyable completion of this thesis. I would like to thank Prof. McCarthy and Prof. Ataai for serving on my thesis committee and I am grateful to Prof. Ataai for the valuable discussions during the course of my thesis.

Over the past two years, 1272 BEH has provided me with a great group of lab-mates who were a lot of fun to work with; John, Phil, Watson, Kunal, Aditya, Ade, Hongming, Jeff, Lei Hong, and James. The steady influx of undergrads also contributed to the great atmosphere in this lab. I am at a loss of words to describe John's role these past two years. From learning Linux, to debugging code, computer issues, squash, music, movies, cooking, talking, arguing, shopping, he has been an integral part of both dumb and exciting things. Special thanks to my amazing apartment-mate Yannick whose baking skills have yet to meet a match. Many a dull and un-interesting evening has been enlivened by his sense of humor, great cooking and entertaining small talk. Steelers games, Seinfeld, and must-see TV Thursday would not be half as fun without him. I would also like to thank a bunch of friends in Pittsburgh who have made these past two years comfortable and convenient; Saurabh, Abhishek, Mahesh, Amit Itagi, Vineet, Deepak, Laxmikant, Parag, Ananth, Brian, and Thalis. I am also thankful to my UDCT friends, Manish, Nandini, Sumati, and Nandan who have always been around when things looked bleak, and have been very supportive through the crests and troughs these past two years.

I am thankful to the University of Pittsburgh, Central Research and Development Fund for funding this work.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

The chemical industry has undergone significant changes in the past two decades. Faced with rising energy costs and increased global competition in terms of price and quality, the trend is shifting towards manufacturing in multi-product batch plants from the conventional continuous manufacturing plants [6]. There is also an increased awareness about environmental issues, and in many cases they dictate the process operation. The need to make products in an environmentally benign manner has paved the way for the biological route towards manufacturing. Biological reactors (or bioreactors) can be operated at ambient pressures and temperatures and in most cases the operation is free of organic solvents.

Research efforts in this direction have been further supported by the vastly increased understanding of the genome of bacteria such as *Escherichia coli* and fungi such as *Saccharomyces cerevisiae* (commonly known as baker's yeast) among many others. The genetic makeup of these organisms can be manipulated, and through a series of such manipulations, organisms can be made to overproduce valuable products. The applications of these genetic manipulations were first envisioned in the early 1990's by Bailey [7] and Stephanopoulos [8] who recognized that these were problems at the interface of biology and chemical engineering; they were the first to apply engineering principles to develop the nascent field of metabolic engineering. In fact, many of the products obtained by the biological route, such as monoclonal antibodies, proteins, and other therapeutic drugs, either cannot be produced, or are very difficult to obtain, by conventional manufacturing methods. Most of these products fall in the low volume/high value bracket, and there is a tremendous economic potential in this sector of the market. It is estimated that the production of therapeutic proteins alone generated sales exceeding $25 billion in 2001 [1].

However, the trend in these processes is to operate in a so called (fed-)batch mode to maintain sterile conditions and for operational flexibility. This operation follows a set pattern akin to a recipe and is repeated from one batch to the next. These recipes are usually obtained by a trial and error procedure with significant experimental effort involved in their determination [9]. By increasing our understanding of these systems it may be possible to avoid some of the empiricism associated with the operation of (fed-)batch bioreactors. Considerable benefit, in terms of reduced product variability and optimal resource utilization could be achieved, by integrating the existing biological knowledge within a dynamic mathematical framework.

1.1 Bioreactor Characteristics

Some of the key issues in bioreactor modeling, control and operation must be understood before a control scheme can be constructed. The bioreactor system can be decomposed into different scales, temporally and spatially. Features pertaining to each of these scales are first examined and the section concludes by examining the modes in which bioreactors can be operated.

1.1.1 Macroscopic Reactor Scale

Utilizing the biological route to manufacturing involves an organism that can express the desired product, where the product is a part of the metabolic cycle of the cell. The operation begins with addition of an inoculum, *i.e.* a small amount of actual living cells, to the liquid medium containing the essential nutrients required for growth of the organism. The carbon source which serves as the primary nutrient for the cell is called the substrate. In addition to the nutrients, a catalyst may be present in the medium and often it is an immobilized enzyme in solid form [10]. The growth process for unicellular organisms is accompanied by an increase in the number of cells which is commonly known as the biomass. Associated with this growth are the uptake of nutrients and generation of products, which if excreted may alter the pH and other conditions of the surrounding medium. Many of the cellular reactions are mildly exothermic, so that the operation is not isothermal. Gas is also sparged into the reactor to supply oxygen and remove carbon dioxide which is a product of the metabolic cycle of the cell. Owing to the bulk movement of cells, the gas sparging, and properties of the extra-cellular products, the rheological behavior of the medium may vary during the course of the reactor operation. Since the cells are grown in, and utilize substrate and nutrients from, the surrounding medium, any change in the medium properties can alter the reactions taking place within the cell; this results in the bioreactor operation being a highly interacting system [11].

1.1.2 Microscopic Cellular Scale

Each individual cell is also a reactor in its own sense, where a complicated series of reactions are taking place. These reactions occur simultaneously but are regulated by internal cellular controls. These controls empower the cell to modify the rates of reactions within its reaction network and production capabilities based on the environment and the nutritional availability. Furthermore, a growing cell population also exhibits cell to cell heterogeneity. Individual cells can be in various stages of growth ranging from daughter cells to fully grown cells, *i.e.* after cell division has taken place. In certain cell types, a budding cell prior to cell division, may also be part of the growth cycle. The metabolic activity of the cells in each of these phases is different. An additional concern is that long term cultivation of a cell population can lead to spontaneous mutations that may cause genetic alterations in the strain [11] and the loss of product generation capability.

Thus bioreactor operation is relatively complex, occurring at a macro (reactor) scale and a micro (cellular) scale with each of the scales having their own complex set of reactions, influenced by each other and also the properties of the surrounding medium. A schematic is shown in Figure 1.1.

1.1.3 Modes of Bioreactor Operation

Based on the environment, type of cell, and cellular product, the mode of reactor operation is selected. The reactor can be operated in a continuous, batch, or fed-batch mode. In the continuous mode of operation substrate is continuously added into the system and products removed simultaneously from the system. In a batch mode, all of the substrate is added at the beginning of the batch but no product is withdrawn until the end of the batch. Fed-batch operation is one where the substrate is added at, or over, particular intervals of time during the course of the batch, and, as in batch reactors, the product is withdrawn only at the end of the batch. Although continuous

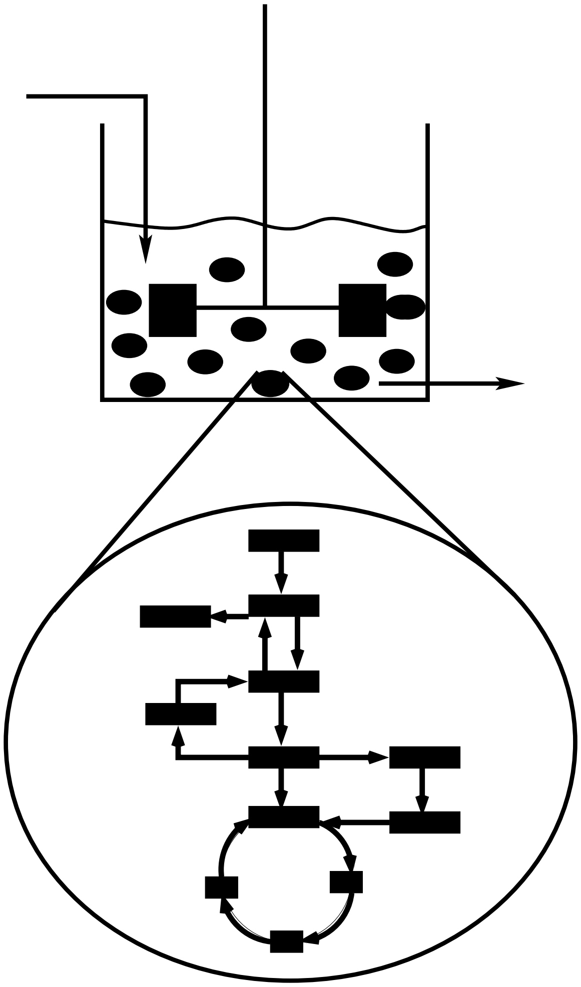


Figure 1.1: Multi-scale schematic for a continuous bioreactor system. Top: macroscopic reactor scale with cells growing in the medium; Bottom: microscopic cellular scale with a single cell reaction network schematic

processes offer advantages such as higher productivity and ease of operation compared to batch processes, they have certain disadvantages associated with them such as equipment failures, infection by other microorganisms, and spontaneous mutations in the strain [12]. On the other hand, batch and fed-batch modes are preferred as they have the advantage of avoiding excessive substrate feed which can inhibit microorganism growth. Since product is also withdrawn at the end of the batch, sterilized conditions can be maintained during process operation. The (fed-)batch modes of operation suffer from certain disadvantages as they are personnel intensive due to the need to sterilize the equipment after every batch. Furthermore, there are difficulties in operation and control such as maintaining the specific growth conditions for the microorganism in the face of variations in medium properties, the batch-to-batch system variability [13], and the inherent nonlinear dynamic nature of fed-batch bioreactors. In spite of these limitations, the important benefit of operating in a sterile atmosphere offsets some of the shortcomings. Also, many bio-pharmaceutical companies are multi-product; the unit operations and equipment in which manufacturing is carried out are common for a number of products [9]. In this setting the fed-batch mode offers operational flexibility and is preferred over the continuous mode of operation. Owing to its practical importance, this thesis focuses on fed-batch mode of bioreactor operation.

1.2 Modeling of Bioreactors

Models are the key components of many advanced control algorithms. Process operation, scheduling, optimization, and control all include a model in explicit or implicit form [14]. This section begins with a brief introduction to various types of mathematical models, then presents a motivation for modeling bioreactors. The focus then shifts to biological models used to describe fed-batch bioreactor operation, and it concludes with a brief literature survey of previous work in this field.

1.2.1 Types of Models

A model is a mathematical representation of a physical process, and it attempts to capture the relationship between the inputs and outputs for the system under study. In general, models do not fall into distinct categories but there exists a continuum for models, varying from fundamentally derived models based on process physics, to purely empirical models developed from input-output data.

Fundamental Models: Fundamental models, (also called first-principles models) are based on the underlying physics occuring in the system and are generally the preferred model structure for developing process representations for model based control systems [15]. These models are developed by applying mass and energy balances over the components or states and may also include a description of the fluid flow and transport processes that occur in the system. A number of fundamental models have been developed to model processes ranging from a single cell [16],[17],[18] to a complex two phase pulp digester [19]. Fundamental models offer several potential benefits. Since they include details about the physics of the system, they can better represent the nonlinear behavior and system dynamics; this allows the model to be used beyond the operating range in which the model was constructed. Another benefit of utilizing the first-principles approach is that the states are generally physical variables such as temperature or concentrations of the system components that can be measured. However these models are time consuming to develop and they often have a high state dimension. The involved mathematical description may result in a model with a large number of equations with many parameters that need to be estimated. Many of these parameters can be obtained experimentally or from previously reported data available in literature. In some cases, regression analysis is used or adjustable parameters are employed to adequately represent the system behavior. The considerable effort required in order to develop a first-principles model often leads to the utilization of a simpler model structure for control applications.

Empirical Models: In many cases the actual system may be complex and the underlying phenomena are not well enough understood to develop a first-principles model. Empirical models are useful in this scenario and actual plant data are used to capture the relationship between the process inputs and process outputs using a mathematical framework such as Artificial Neural Networks (ANN) [20],[21], polynomial models such as the Volterra-Laguerre models [22],[23],[24] *etc.* The parameters describing the model are then identified by a regression analysis of the process data [25]. As an example of the empirical modeling framework, Parker *et al.* utilize a second order Volterra series model framework to capture the dynamic behavior exhibited by continuous cultures of *Klebsiella pneumoniae* [22]. Although the time requirement to obtain these models is often significantly reduced, these models generally can be used with confidence only for the operating range in which they are constructed. Since these models do not account for the underlying physics, practical insight into the problem may be lost as physical variables of the system such as temperature and concentration are generally not the states of the empirical model [26].

Mixed Models: Mixed models are developed, as the name suggests, by combining the fundamental and empirical models, thus utilizing the benefits of both. As an example, mixed models have been used to model polymerization reactors in which the known mass balances for the reactants are modeled using fundamental models, whereas, the unknown rates of the reactions taking place are modeled using empirical models [27],[28]. In some cases mixed models have also been used to determine time-varying parameters that model the nonlinear process behavior [29]. Another approach as suggested by Henson [30], is to use a fundamental model to capture the key process dynamics and then utilize an empirical model to capture the modeling error between the model and the actual plant.

1.2.2 Motivation for Modeling of Bioreactors

Most industrial processes are operated in non-stationary conditions in a batch or fed-batch mode of operation [12]. This requires knowledge of the substrate profile that should be followed in order to achieve the (fed-)batch target. There are many candidate profiles, and testing these experimentally is an economically expensive exercise as each batch run is followed by product removal and reactor sterilization, which is time and personnel intensive and adds to the process downtime. In such a scenario, a mathematical model describing the system could prove useful. Tools from control theory can be used in conjunction with a mathematical model to obtain profiles that the system variables must follow to satisfy a pre-defined objective. This reduces the experimental effort as well as the time required to optimize the system, translating into economic savings.

As is seen in Sections 1.1.1 and 1.1.2, bioreactor operation involves both a macroscopic reactor scale and a microscopic cellular scale, and an accurate description of the system must include information at both of these scales. The reactor level description may involve macroscopic balance equations for substrate, biomass, product, and medium volume. The cellular events are mathematically represented by a biological model. These macroscopic and microscopic scales are linked together by the rates of substrate consumption, biomass growth and product formation which occur at the cellular level but are affected by the macro-scale concentrations. The complete process model is thus a combination of a reactor and a biological model. A discussion on the various categories of biological models and assumptions involved in their description follows.

1.2.3 Classification of Biological Models

In the literature, biological models have been classified based on the level of detail in the description of the cell and its intra-cellular processes. Considering biomass, Tsuchiya *et al.* [31] describe the model as segregated if all cells are treated differently *i.e.* they may be in different stages of growth. If the cells are all treated alike then the model is a non-segregated model[32]. Further if the intra-cellular processes are explicitly accounted for the model is a structured model, whereas if the model relies only on lumped cellular input-output behavior it is an unstructured model. A compact representation is presented in Table 1.1.

Table 1.1: Classification of biological growth models

MODELS	**Unstructured**	*Structured*
Unsegregated	cells indistinguishable **lumped metabolic and physiological processes**	cells indistinguishable *distinct physiological and metabolic processes*
Segregated	cells may be in different states **lumped metabolic and physiological processes**	cells may be in different states *distinct physiological and metabolic processes*

1.2.4 Bioreactor Modeling Literature Review

An overview of these model categories and a brief literature review on the specific modeling framework is now presented.

Unstructured Models: Unstructured models are the simplest of all modeling philosophies used to describe the biological model. They consider the cell mass as a single chemical species and do not consider any intracellular reactions occuring within the cell. Unstructured models typically describe the growth phenomena based on a single limiting substrate and consider only substrate uptake, biomass growth, and product formation in the modeling framework. Thus the biological component of the system depends directly on the macroscopic reactor variables. These models give an adequate representation of the biological growth phenomena in relatively simple cases, when the cell response time to environmental changes is either negligibly small or much longer than the batch time [12].

The most commonly used unstructured model in the literature is the Monod model [33]. and it is one of the earliest attempts at modeling biological systems. In this model the growth kinetics are expressed in terms of the specific growth rate, μ. This equation is of the form,

$$\mu = \mu_{max} \frac{S}{K_s + S} \tag{1-1}$$

where μ_{max} is the maximum achievable specific growth rate for $S \gg K_s$ and K_s is the value of the limiting substrate for which $\mu = \frac{\mu_{max}}{2}$. These parameters are obtained experimentally and they do not have a direct physical interpretation [11]. The Monod model is used to express growth based on a single limiting substrate S and can qualitatively describe growth phenomena for relatively simple systems.

A drawback of this model is that it does not capture the initial lag phase of growth which is observed in most batch cultures and also fails to capture the sequential consumption of glucose and ethanol as multiple substrates, as observed in cultures of *Saccharomyces cerevisiae* . There are variants to the Monod model that have also been used, such as the models by Messier, Contois *etc.* reported in [11]. These models differ in their substrate dependence and some include terms to account for saturation due to high substrate concentration and inhibition due to product or a competing inhibitor. However these models do not differ significantly from the Monod model in the fact that they are empirical and represent all of the cellular processes with just a single equation for the specific growth rate. As an example, Yang *et al.*[34] show that for yeast cultures with phenol as the limiting substrate, five different equations with inhibition terms represented the data equally well. Monod type models are a helpful tool for a preliminary analysis of the system, but they fail in more complex cases owing to the simplified description of the cell's biochemical machinery.

Structured Models: In contrast to the unstructured models, structured models include a greater level of detail about the biochemical phenomena occuring within the cell. The biological detail is no longer lumped into a single biomass variable. In structured models, the biomass is structured into several components or functional groups that are interconnected with each other and with the macroscopic reactor environment by material balances [12]. The functional groups that represent the biomass can themselves vary based on the level of detail included in their description. These functional groups could be reaction networks including details about known intracellular reactions, as is employed in the single cell models [16],[17],[18], or they could have a lumped representation of the actual detailed networks as in the case of the two-compartment model of Williams [35].

Single cell models have been successful in describing cellular events in a mathematical format. Generally, these models have been developed with a specific purpose. Jeong and Ataai [16] have captured the lag phase of growth for *Bacillus subtilis* and have included a detailed representation of various metabolic pathways in the cell, most notably, purine metabolism. This is a good model structure for understanding cellular regulation and also gives valuable insight into the mechanism by which sporulation occurs. Similarly, the model by Domach *et al.*[17] can predict changes in cell size, cell shape and the time at which chromosome synthesis initiates as a function of the limiting external glucose concentration for *Escherichia coli.* In an extension of this work, Perretti and Bailey [36] formulated a model that can describe the kinetics and control of transcription and translation. Steinmeyer and Shuler [18] also utilized a single cell model to capture the dynamic growth behavior of yeast cultures in batch mode. This model structures the biomass into twelve distinct pools and has over a hundred parameters. Determination of these parameters is not a trivial task and although many of these parameters can be obtained experimentally, there also exists an element of empiricism in the choice of the parameter value.

Segregated Models: Segregated models take into account the fact that a culture can have cells in different stages of growth, and they account for these stages using population balances. These models can be broadly grouped into two main types, age structured or mass structured. If the intracellular chemical structure is described in terms of a mass conservation then the population balance is mass-structured, whereas if age is used to differentiate cells in a population, then the model is referred to as an age structured model. Mantzaris *et al.*[37] utilize a mass structured segregated model in a continuous bioreactor. Cell cycle is assumed to occur in two stages and the product formation is considered only in the second stage of the cell cycle. Substrate concentration is used as the manipulated input to control the productivity of a desired product. A mass-structured cell population balance model has also been used by Zamamiri *et al.*[38] to capture the experimentally observed oscillatory dynamics exhibited by continuous cultures of yeast, using an unstructured biological model. The key variable that characterizes the cell cycle is the critical cell mass m_c. Cells with a mass greater than m_c are mother cells whereas those with a mass lower are known as daughter cells. The origin of the cell cycle known as “Start” is the point when the cell attains the critical mass m_c. Bud formation takes place once the cell has grown for a particular time and the budding cells continue to grow until the cell division stage is reached. At this division stage, the bud separates forming a new daughter cell of smaller mass than the mother cell which is returned to the “Start” point and has its age set to zero. Daughter cells become mother cells once they reach m_c and continue their growth until the budding stage is reached at which point they become mother cells.

With the use of population balances for model formulation, segregated models can no longer be described by ordinary differential equations. The constitutive equations are now described by partial differential equations in age-structured, and integral-partial differential equations in mass structured segregated models. In most cases analytical solutions of these equations are difficult to obtain and solution has to be accomplished numerically. This makes the computations associated with these models significantly more challenging. Additionally, identifying and mathematically expressing the intrinsic physiological state functions such as the growth rates, the stage-to-stage transition rates, that are parameters in these models is in itself a difficult exercise.

Material Balance Models: Wang and Cooney [39] have reported the control of bioreactors using a model based on material balances. They treat the biomass as a compound with a fixed elemental composition consisting of Carbon, Hydrogen, Oxygen, and Nitrogen. Based on a stoichiometric balance of these elements and the uptake and evolution of Oxygen and Carbon Dioxide gases, the state of the fermenter is obtained. This state information is somehow related to the physiological state of the growing cells and this relationship is usually determined empirically. Material balance models, (also known as stoichiometric models) were very popular in the 1970’s and 1980’s when computational resources were expensive and sensor technology was not well developed. However, these models do not account for the biomass as an actual living species and are thus only a chemical approximation to the biology. Although these models have been successfully used, they suffer from most of the disadvantages of empirical and unstructured models.

Cybernetic Models: Another framework that has been used in literature to describe biological growth is the cybernetic modeling framework first proposed by Ramkrishna and co-workers [40],[41]. The main principle of the cybernetic modeling approach can be summarized as[42] : "*The metabolic processes within cells are regulated in an optimal manner by the cell (developed through evolution), and an accurate description of the overall system can be formed if the net outcome of the process is captured, without knowing the full reaction mechanism* ". Thus, in cybernetic modeling, no mechanistic framework for the utilization of substrates is assumed, but it is believed that whatever the underlying framework may be, it leads to optimal regulation of the cells. Cellular controls are established by adjusting the level of enzyme activity for a particular substrate uptake system based on some criterion which is typically to maximize the cellular growth. It empowers microorganisms to allocate cellular machinery and resources for the uptake of those substrates that best fit the cellular requirements. However there are certain drawbacks to cybernetic models; in most bio-pharmaceutical applications the intent is the overproduction of a particular cellular product and this is not what the undisturbed cell would produce. Furthermore, the hypothesis that the cells are trying to maximize their own well-being cannot be directly tested whereas metabolic reaction networks (which cybernetic models do not recognize) can be tested by a number of techniques [43]. Finally, by ignoring cellular reaction networks, and relying on the net outcome of the process to capture the cellular behavior, cybernetic models possess some of the drawbacks of unstructured and empirical models. However, cybernetic models have had considerable success in modeling of yeast cultures and this framework has been shown to capture the growth on mixed substrates [44] and the oscillatory dynamics exhibited by continuous cultures of yeast [45].

Neural Network Models: Artificial Neural Network (ANN) models have also been used for describing the growth of yeast cultures [27],[28], and the general approach to using ANN's is to combine them with material balance models. Azevedo *et al.*[28] utilize a neural network structure to estimate the unknown rate equations for fed-batch yeast cultures. The inputs to the neural network are the feed glucose, ethanol, and the biomass and the calculation of the unknown rates is carried out at every time step utilizing available information from the mass balance equations. A similar approach is used by Tholudur and Ramirez [27] who utilize the neural network to capture the functional form of various metabolic parameters such as the specific growth rate, rate of product formation, and yield factors in conjunction with a mass balance model for the macroscopic variables namely the biomass, substrate and medium volume. Neural networks have been used because the time required for model-building is much less as compared to fundamental models, but there is a lot of empiricism associated with the training of the neural network. In addition, neural networks are generally not reliable beyond the region in which they are trained and extrapolation may lead to unpredictable results.

1.2.5 Control Relevant Modeling

Before moving on to the section on the control of fed-batch fermentors, model properties and usability, from a control perspective are discussed. For process control applications it may not be necessary for a model to describe the complete set of dynamic process behavior. This leads to the philosophy of control relevant modeling, whereby only those aspects affecting the ability to control the process are included in the model [26]. As an example, Skogestad and Morari consider a linear two time constant model for a distillation column where the two time constants that capture the dynamic behavior are motivated by the internal and external flows in the column. The main focus of their work was to develop a model, which included the factors that were most important for feedback control, rather than an accurate physical model for feed-back control [46]. Balasubramhanya and Doyle III [47] use traveling wave phenomena to develop a low order nonlinear model that can capture the nonlinear dynamic behavior of a high purity distillation column, and utilize the low order model to develop model-based control strategies. Thus, the main idea behind control relevant modeling is the recognition of the inherent trade-off between modeling the process completely, and modeling only the relevant process dynamics with a simple model, thereby leading to a less complex controller design.

1.3 Bioreactor Control

1.3.1 Motivation for Control

One of the prominent benefits of having a bioreactor model is the ability to use it for control. For fed-batch systems, a practically relevant goal is to follow a pre-determined trajectory for the controlled variable which maximizes (or minimizes) a particular performance objective. Examples could be maximizing cell production or target protein concentration at the end of the batch. Proper control could ensure high yield of pure product at reduced manufacturing costs. However, industrial implementation of advanced control strategies for fermentation reactors has not kept pace with those in conventional chemical and petroleum industries. In addition, the fed-batch mode, one of the most commonly used modes of operation, is also one of the most challenging for control[48]. The system behavior is essentially nonlinear and is further complicated by the fact that fed-batch systems are dynamic in nature, *i.e.* they do not operate about any single steady state. A classic example is the trajectory tracking for nonlinear, fed-batch systems where classical transfer-function based control methods break down as a single transfer function cannot capture the system behavior over the entire trajectory. From this discussion it is clear that there exist limitations for conventional automatic control techniques and the probability of their successful implementation is low [6]. To alleviate some of these problems modern control theory offers methods of adaptive, nonlinear, and multi-variable control which are often directly based on mathematical models.

1.3.2 Review of Controller Design Literature

As discussed before, control strategies for fed-batch systems may focus on maximizing the cell or product concentration or any other *cost function* at the end of the batch. Historically, control approaches for the fed-batch control problem have been classified into one of two categories, physiological model based and dynamic optimization [49]. Physiological control is a methodology, that is not strictly based on a mathematical model *per se* but is concerned with maintaining the fed-batch bioreactor operation about a particular set point. Possible choices for the set point that are considered include substrate concentration control, specific growth rate control, ethanol concentration control, and respiratory quotient control, *i.e.* controlling the ratio of the carbon dioxide evolution rate to the oxygen uptake rate. Emphasis is laid more on what state variable should be chosen as a set point rather than how the system would be regulated about a set point value. Since this is a problem that involves regulating the system about a set-point, the resulting controller is termed a *regulatory controller.* In the other approach termed as dynamic optimization, the goal is to minimize an appropriate cost function subject to the dynamic behavior constraint represented by the process model. This involves designing a controller to track pre-specified trajectories and is accomplished by a servomechanism or a *servo controller.*

The problem of fed-batch bioreactor control has been tackled by a variety of authors using different approaches that can be broadly classified into open-loop or closed-loop control strategies. Open-loop operation involves off-line calculation of the substrate feeding strategy that attains the (fed-)batch targets whereas closed-loop control involves on-line tracking of reference trajectories in the presence of disturbances by controller-induced changes in the input signals.

Open-loop Control: Open-loop control involves calculating the feed rate that provide an optimal state or controlled variable trajectory off-line using a process model and then implementing the resulting feed profile in an open-loop manner *i.e.* with no measurement feedback during process operation. The goal of the control strategy is to drive the controlled or state variable towards a desired final objective, and the solution involves determination of the time-varying substrate feed profile that minimizes the cost function subject to the dynamics of the process model. Traditionally, optimal control problems have been solved using Pontryagin's minimum principle [50], discussed in Chapter 3. However, for fermentations with substrate feed as the controlled variable the problem is singular in nature as the Hamiltonian of the system is linear with respect to the feed rate. The solution to the problem is then either bang-bang, *i.e.* is in the form of an on-off controller with a dead-band, or it follows a singular profile so that the optimal profile can be discontinuous [51]. A variety of different optimization techniques have been used to solve the singular control problem.

Banga and co-workers [52] use a control vector parametrization technique and transform the original Optimal Control Problem (OCP) into a Nonlinear Programming (NLP) problem. They present an algorithm for the solution of the NLP problem, in which deterministic elements of the solution such as Hessians and gradients for the objective function are calculated using second order sensitivities. The authors claim that their method solves the dynamic optimization problem even for high levels of control discretization.

Considerable work has been done by Luus and co-workers in the development of optimization techniques for optimal control problems [53],[54],[55]. As an alternative to Iterative Dynamic Programming (IDP), Luus and Hennessy [53] suggest a direct search optimization scheme where the optimal control is obtained by optimizing simultaneously over all the time stages into which the batch time is divided. The technique used in [53] is thus different from IDP in which the solution is obtained stage by stage.

Pushpavanam *et al.*[56] consider optimization of a fed-batch process for alcohol fermentation constrained by a fixed final volume using a Sequential Quadratic Programming (SQP) approach. They divide the entire batch into a series of equally spaced intervals and feed is assumed to be introduced in discrete pulses at the beginning of each interval. The effect of number of stages on the optimum performance for both free and fixed time cases is reported and finally comparisons with the minimum principle are presented.

Hong[57] has considered the problem of maximizing the ethanol concentration for product and substrate inhibited fed-batch fermentations. Kelley's transformations are used to reduce the number of system equations by two and an analytical expression for the conjunction point between the singular and non-singular arcs for the singular control problem is also presented. For the same system, Chen and Hwang [58] report on-off control to obtain the optimal substrate policy. The Differential Algebraic Equation (DAE) system that represents the process model is first simplified using Kelley's transformations and the input is parametrized over the batch period. SQP is used to solve the resulting NLP problem.

Closed-loop Control: Much of the pioneering work on bioreactor control has been done by Lim and co-workers [59],[60], and they are amongst the first to analyze the problem of feedback optimization for the singular control problem of fed-batch fermentors. They consider a general system composed of four differential equations that represent mass balances for the state variables *i.e.* substrate, biomass, product, and fermentor volume, and they formulate the problem based on singular control theory. A nonlinear relationship is obtained between the feed rate and the system state variables in a feedback form, and conditions such as fermentation kinetics, objective function, and initial conditions under which the feedback results hold are reported. Finally, the optimum feed profile in feedback form for ethanol fermentation is presented [59].

Palanki *et al.* [61] have also considered the problem of determining the optimal profile in feedback form. The authors analyze the optimal control problem from a geometric perspective and introduce the concept of degree of singularity that allows a better characterization of the necessary conditions for optimality. Optimal feedback laws are then derived for the singular region of operation and results are presented for time-invariant systems and extended to include time-varying systems as well. Finally, as an example, they consider the applicability of their analysis to yeast fermentation.

Hodge and Karim have studied the problem of maximizing ethanol production from fed-batch cultures of *Zymomonas mobilis.* The authors utilize a Model Predictive Control (MPC) framework for their analysis and utilize the square of the one step ahead error prediction as the objective function for control [62].

1.4 Thesis Overview

The structure for the remainder of this work is as follows. Chapter 2, concentrates on the process model. Motivation for the model selection is presented, followed by a description of the model structure. Open-loop behavior of the model concludes the chapter. Chapter 3 focuses on the controller development. An overview of optimal control theory is presented and the optimal reference trajectory for the process model is obtained. The remainder of the chapter is concerned with the development of the model predictive control formulation to track the optimal trajectory in a closed-loop implementation. The thesis concludes with Chapter 4 which presents the summary and recommendations for future work.

2.0 MODEL DEVELOPMENT

For the design of a model-based control scheme, the actual process model, plays a very important role. Process operation, optimization, and scheduling all require a suitable process model [14]. The fed-batch bioreactor operation considered in this work is dynamic in nature. As a result, the substrate profile that is fed into the reactor plays a very important role in the batch progression as well as the final product concentration that is obtained at the end of the batch. Since each reactor run is followed by a personnel intensive, cleaning and sterilization operation, determination of the best possible profile may be economically expensive. A process model in such a scenario, could prove very useful. Using tools from control theory, optimum substrate profiles could be determined in much less time compared to experimental determination, thereby translating into economic savings. Besides, a suitable model helps in understanding the system as well as the relative importance of parameters that affect the process behavior. In this chapter, reasons motivating the model selection are first presented, followed by a detailed description of the model structure. Finally, open-loop behavior of the process model is presented.

An adequate representation of the bioreactor operation must include description of events occuring at both the reactor and cellular scales. In this regard, one of the key variables that describes microorganism growth in a bioreactor model, is the specific growth rate, μ. The manner in which the specific growth rate is represented is different in each of the models. The specific growth rate typically ranges from a relatively simple and empirical representation in the case of unstructured models to a fairly detailed and complex function in the case of structured models. In a multi-scale model, μ serves as the coupling parameter between events occurring at the macroscopic and microscopic scales and a general schematic of a multi-scale model coupling is elucidated in Figure 2.1. At the macroscopic level, the system is described by mass balances for the biomass, substrate, reactor volume, and the product whereas the biological component is a combination of the metabolic pathway and the kinetic information. The specific growth rate, and product formation rate are dependent on the rate of substrate uptake from the bulk medium surrounding the cell, and on the

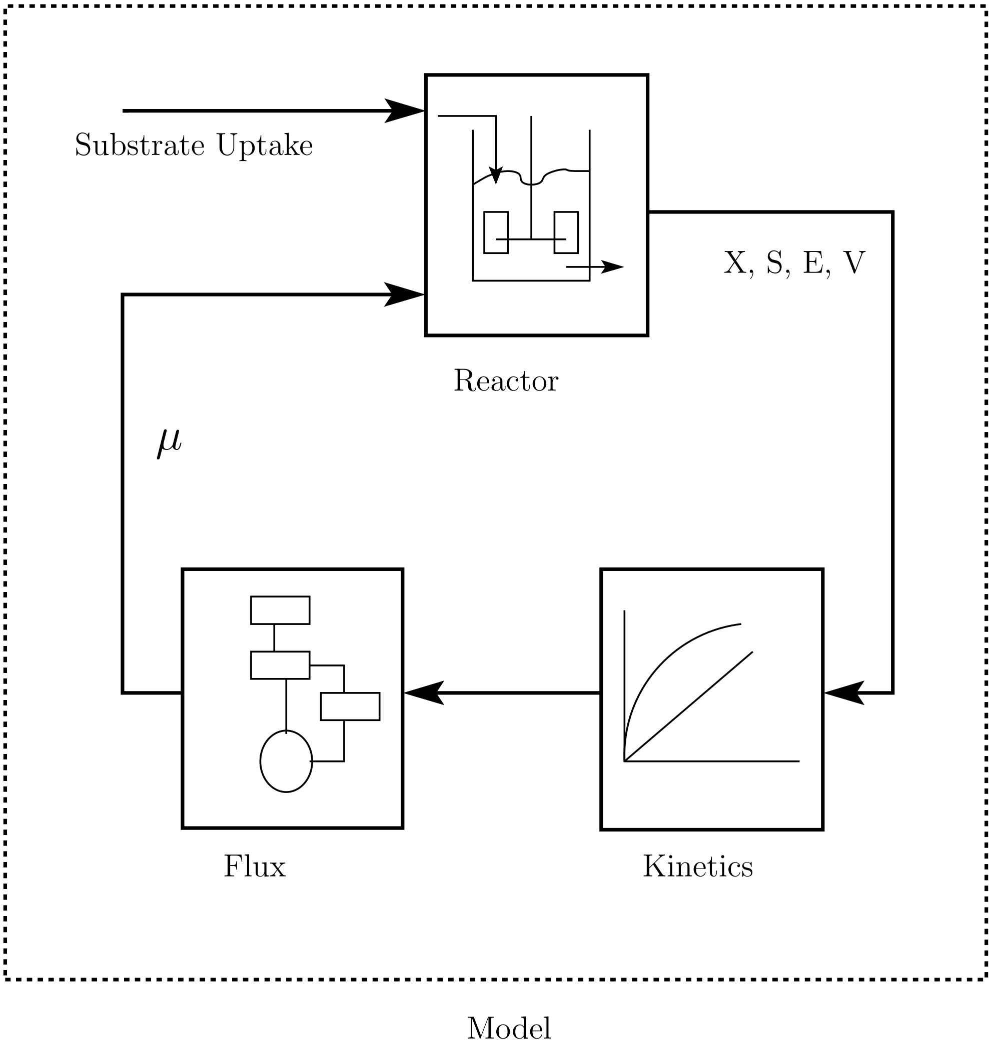


Figure 2.1: Combination of microscopic and macroscopic models

rates of reaction in the cellular metabolic pathway. Thus, at any given instant, the specific growth rate is dynamically calculated, and serves as an accurate descriptor of events occurring at the cellular level. This information can therefore be integrated with mass balances for the substrate, biomass, system volume, and the product to obtain the complete process model.

The uniqueness of this work is the application of a distinct model structure, that recognizes the inherent multi-scale nature of the bioreactor operation. The multi-scale model is more complex than the commonly used unstructured reactor models so that it captures more biological detail, but is simple enough as compared to the single cell models, so that it is computationally tractable for controller design.

2.1 Bioreactor Model Description

The process under consideration in this work was a fed-batch bioreactor growing *Saccharomyces cerevisiae* on glucose. The model was described by Enfors and co-workers [3] and is explained here for completeness. The notation used here is the same as that used in the original manuscript.

The process model includes a description of the metabolism in *Saccharomyces cerevisiae* in an aerobic fed-batch culture and this is used for the simulation of concentrations of the state variables, biomass (X), glucose (S), ethanol (E), medium volume (V), and dissolved oxygen tension (DOT). The rate equations for the metabolic fluxes are expressed as specific rates q $(g/g \cdot h)$. An algorithm for the central glucose flux distribution is shown in Figure 2.1. The first step in the flux pathway is the substrate uptake, q_S, that is assumed to follow Monod kinetics and includes an experimentally observed lag-phase term.

$$qS = qS_{max}\frac{S}{S+K_s}(1-e^{-t/t_L}) \tag{2-1}$$

In this equation, qS_{max}, is the maximum specific glucose uptake rate, t_L is the lag phase, and K_s is the saturation constant for glucose uptake. Initially, the sugar is assumed to be channeled into the oxidative metabolic pathway, qS_{ox}, which is divided into fluxes of sugar used for anabolism,

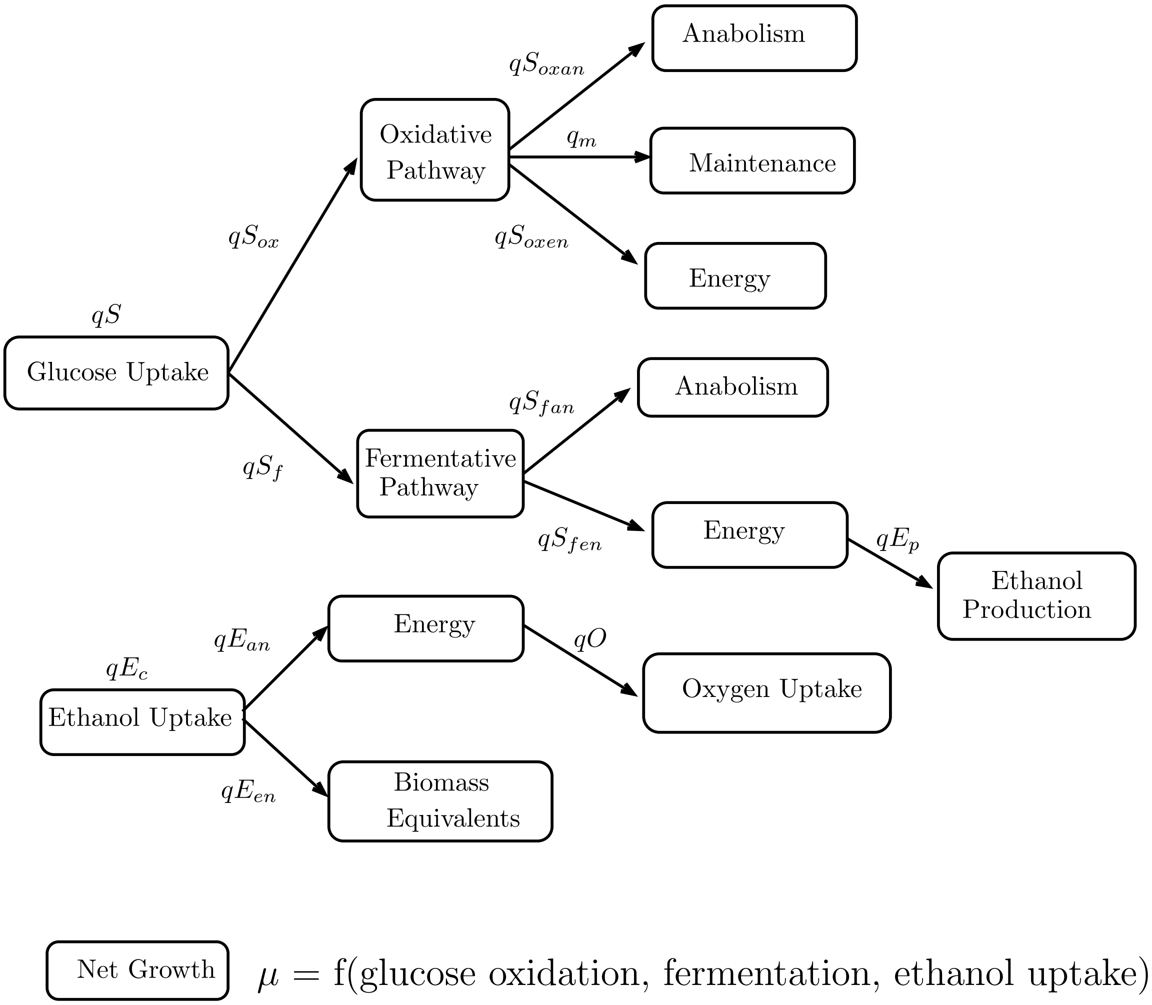


Figure 2.2: Schematic for the distribution of the central glucose flux

qS_{oxan}, and for energy metabolism, qS_{oxen}.

$$qS_{ox} = qS_{oxan} + qS_{oxen} \tag{2-2}$$

By setting the specific rate of carbon flux to anabolism equal to the specific rate of carbon accumulation, an expression for flux of sugar used for anabolism can be obtained.

$$qS_{oxan} = (qS_{ox} - q_m)Y_{XS_{ox}}\frac{C_X}{C_S} \tag{2-3}$$

The maintenance flux, q_m, is assumed to be constant. In this equation C_X and C_S are the carbon content in biomass and in glucose respectively (both in $molC/g$), and $Y_{XS_{ox}}$ is the oxidative biomass yield on glucose (g/g).

The flux of glucose used for oxidative energy production, qS_{oxen} is obtained from Equation 2-2. Using the stoichiometry of respiration, and the flux of sugar used for aerobic energy metabolism, qS_{oxen}, the rate of oxygen needed for sugar oxidation, qO_S is obtained,

$$qO_S = qS_{oxen}Y_{OS} \tag{2-4}$$

where Y_{OS}, is the coefficient of respiration on glucose (g/g).

However, qO_S is limited by the maximum respiration capacity, qO_{max} and qO_S causes a proportional reduction in the fluxes of qS_{oxan} and qS_{oxen}, when $qO_S > qO_{max}$. The expression for qO_{max} is given as,

$$qO_{max} = q\hat{O}\ \frac{1}{1 + E/K_i}\frac{DOT}{DOT + K_o} \tag{2-5}$$

Here, K_i is the inhibition constant for ethanol, and K_o is the saturation constant for the dissolved oxygen tension. Thus, the maximum respiratory capacity is not a constant but at any given time it is a function of the dissolved oxygen tension and the ethanol concentration at that time.

The flux of fermentative metabolism, q_{S_f}, is obtained from the difference between the flux for substrate uptake and the oxidative flux,

$$qS_f = qS - qS_{ox} \tag{2-6}$$

As with aerobic metabolism, qS_f is divided into fluxes for anabolism, qS_{fan} and for energy metabolism qS_{fen}, so that,

$$qS_f = qS_{fan} + qS_{fen} \tag{2-7}$$

The flux for ethanol production, qE_p is then obtained from the stoichiometry of the reactions from glucose to ethanol,

$$qE_p = qS_{fen}Y_{ES} \tag{2-8}$$

Here, Y_{ES} is the stoichiometric ethanol yield on glucose (g/g). The equations described thus far capture biomass growth and product formation on glucose which is the primary substrate.

The consumption of ethanol as a substrate by the cells occurs only during aerobic cultivations when the glucose flux is low such that, $qO_S < qO_{max}$. The rate of consumption of ethanol, qE_c, is in turn utilized for anabolism, qE_{an} and for energy utilization, qE_{en}.

$$qE_c = qE_{an} + qE_{en} \tag{2-9}$$

The maximal rate of ethanol consumption for energy is given as,

$$qE_{en} = \frac{qO_{max} - qO_S}{Y_{OE}} \frac{E}{E + K_e} \tag{2-10}$$

where, Y_{OE} is the coefficient of respiration on ethanol (g/g), K_e is the saturation constant for ethanol uptake (g/L). A similar equation for anabolism is expressed in terms of biomass equivalents as,

$$qE_{an} = qE_cY_{XE}\frac{C_X}{C_E} \tag{2-11}$$

and in this equation Y_{XE} is the biomass yield on ethanol (g/g), while C_E is the carbon content in ethanol $(molC/g)$. Equations 2-9 through 2-11 describe the mechanism by which ethanol can be used as a substrate for biomass growth.

The rate of oxygen uptake, qO, is given as,

$$qO = qO_S + qE_{en}Y_{OE} \tag{2-12}$$

Finally, the specific growth rate as a function of the metabolic rates is given as,

$$\mu = (qS_{ox} - q_m)Y_{XS_{ox}} + qS_fY_{XS_f} + qE_cY_{XE} \tag{2-13}$$

where, Y_{XS_f} is the fermentative biomass yield on glucose(g/g).

Using these rates, mass balance equations for the state variables are given as,

$$\frac{dX}{dt} = \mu X - \frac{F}{V}X \tag{2-14}$$

$$\frac{dS}{dt} = \frac{F}{V}(S_i - S) - XqS \tag{2-15}$$

$$\frac{dV}{dt} = F \tag{2-16}$$

$$\frac{dDOT}{dt} = KLa(100 - DOT) - 14000XqO \tag{2-17}$$

$$\frac{dE}{dt} = X(qE_p - qE_c) - \frac{F}{V}E \tag{2-18}$$

The equations for biomass, substrate, and ethanol consist of two terms. A positive, growth or accumulation term whereas the second term containing the the factor F/V, is due to the dilution of these components as feed is added during the reactor operation. The concentration of substrate in the feed is S_i, and it is assumed that the feed is sterile, *i.e.* it does not contain any biomass. KLa is the liquid side mass transfer coefficient in the balance equation for dissolved oxygen. The parameters used in the model are described in Table 2.1

Table 2.1: Parameters used for the growth model

Parameter	**Value**	**Units**
Biomass composition	$CH_{1.82}O_{0.58}N_{0.16}$	-
C_E	0.0435	$molC\ g^{-1}$
C_S	0.0333	$molC\ g^{-1}$
C_X	0.0384	$molC\ g^{-1}$
K_e	0.1	$g\ L^{-1}$
K_i	10.0	$g\ L^{-1}$
K_s	0.12	$g\ L^{-1}$
q_m	0.01	$g\ g^{-1}\ h^{-1}$
qO_{max}	0.3	$g\ g^{-1}\ h^{-1}$
qS_{max}	2.4	$g\ g^{-1}\ h^{-1}$
t_L	0.75	h
V_m	24.04	$L\ mol^{-1}$
Y_{ES}	0.510	$g\ g^{-1}$
Y_{OE}	2.087	$g\ g^{-1}$
Y_{OS}	1.067	$g\ g^{-1}$
Y_{XE}	0.72	$g\ g^{-1}$
Y_{XS_f}	0.10	$g\ g^{-1}$
$Y_{XS_{ox}}$	0.50	$g\ g^{-1}$

A MATLAB (© 2002, The MathWorks, Natick, MA) MEX-file of the bioreactor model is located in Appendix A . In order to obtain the concentration profiles, the differential equations are integrated forward in time for the duration of the batch which is 20 hrs. The initial conditions are the same for all model simulations in this chapter: 0.2 g/L biomass, 100 g/L of glucose, 0 g/L ethanol, 1 L of reactor volume and 100 % dissolved oxygen tension. The feed into the reactor is sterile and has a substrate concentration $S_i = 100\ g/L$. The fed-batch process has a single controlled output, the ethanol concentration (g/L) at the end of the batch, which was manipulated using the feed flow rate F (L). The model can capture diauxic growth observed in *Saccharomyces cerevisiae*, i.e. sequential growth on glucose and ethanol as substrates. This is shown in Figure 2.3, a model simulation of batch growth. Once glucose is consumed completely, biomass growth continues by consuming ethanol as the substrate. Model simulation results for a step change in the input are as shown in Figure 2.4, and this figure demonstrates the integrating nature of the system volume. The specific growth rate for biomass growth on ethanol is also much less than that on glucose. This is observed from the variation of the specific growth rate, in relation to the substrate, biomass, and ethanol concentration profiles in Figure 2.5 and it represents both batch and fed-batch growth on glucose. The final volume of the reactor at the end of both the batch and the fed-batch operation is 20 L. The initial conditions used for both the simulations are also identical and are the same as those previously used to generate Figure 2.3.

The series of plots on the left, show the concentration profiles for the biomass, glucose, specific growth rate, and ethanol for a pure batch growth with initial concentration of substrate as 100 g/L. It is observed that, during the initial part of the fermentation, the substrate, glucose, is utilized for both biomass growth, and ethanol production. The specific growth rate, during glucose consumption is fairly high. However, as the batch proceeds the glucose consumption increases, thus reducing the specific growth. This continues until all of the initial glucose is completely consumed, and from this point on-wards biomass growth continues on ethanol as a substrate. As a result, the concentration of ethanol decreases and the specific growth with ethanol as a substrate is much less than that on glucose.

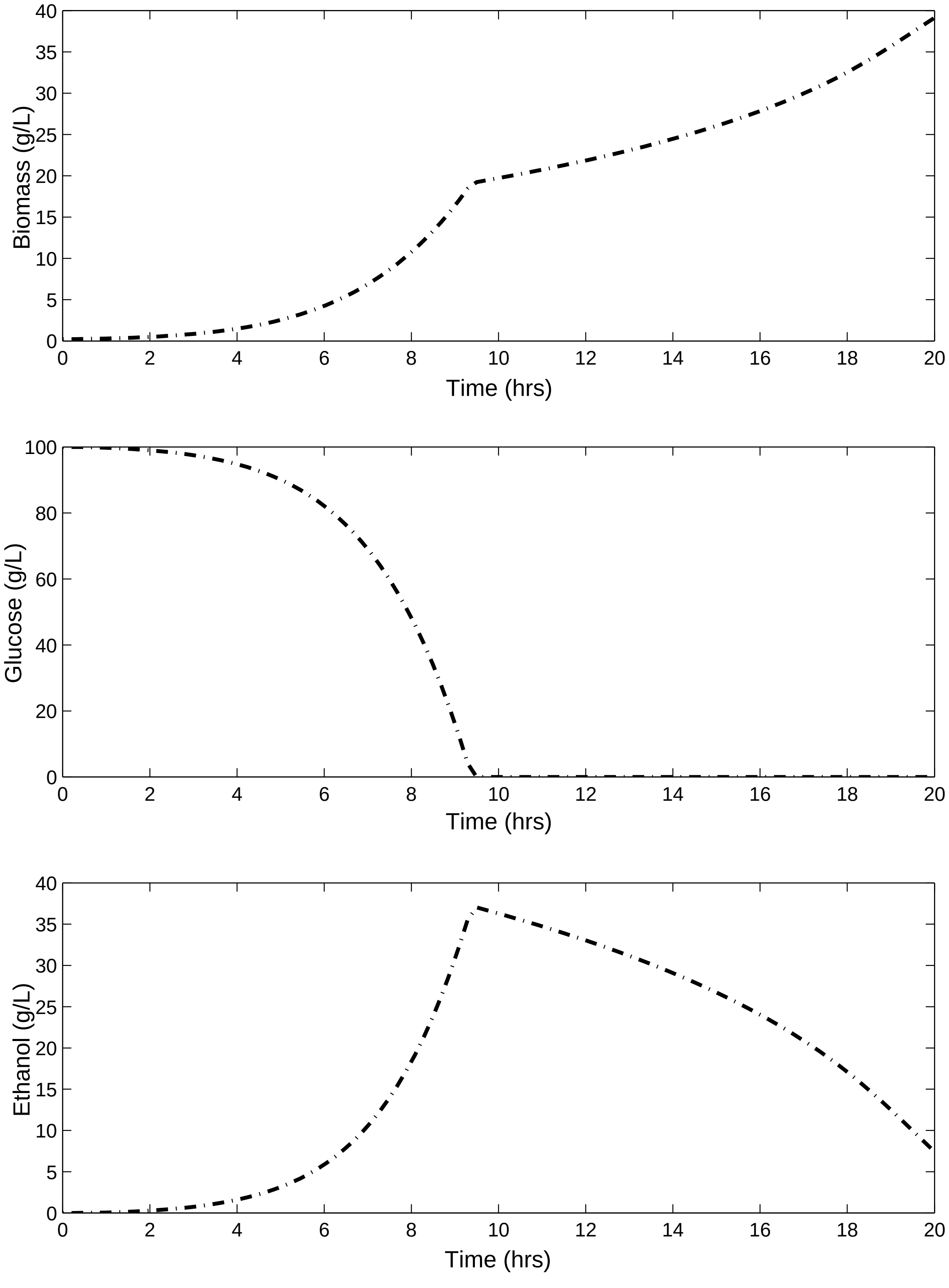


Figure 2.3: Plot showing sequential growth of biomass on glucose and ethanol under batch growth conditions. $(X_0 = 0.2\ g/L,\ S_0 = 100\ g/L,\ E_0 = 0\ g/L)$

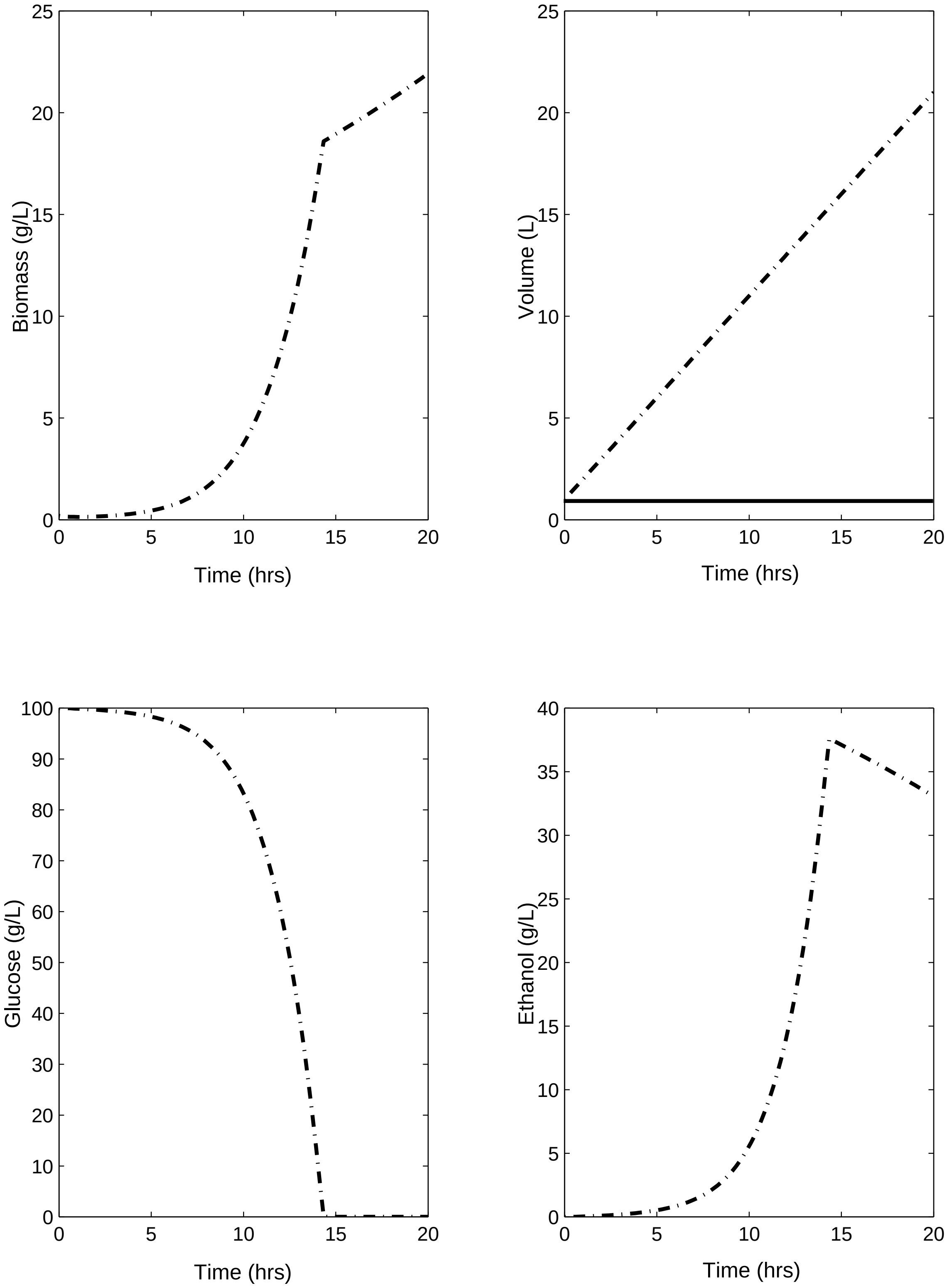


Figure 2.4: Plot of the state variable profiles($- \cdot -$) for a unit step change in input(—) applied at the start of the batch. ($X_0 = 0.2\ g/L,\ S_0 = 100\ g/L,\ E_0 = 0\ g/L,\ V_0 = 1\ L, S_i = 100\ g/L$)

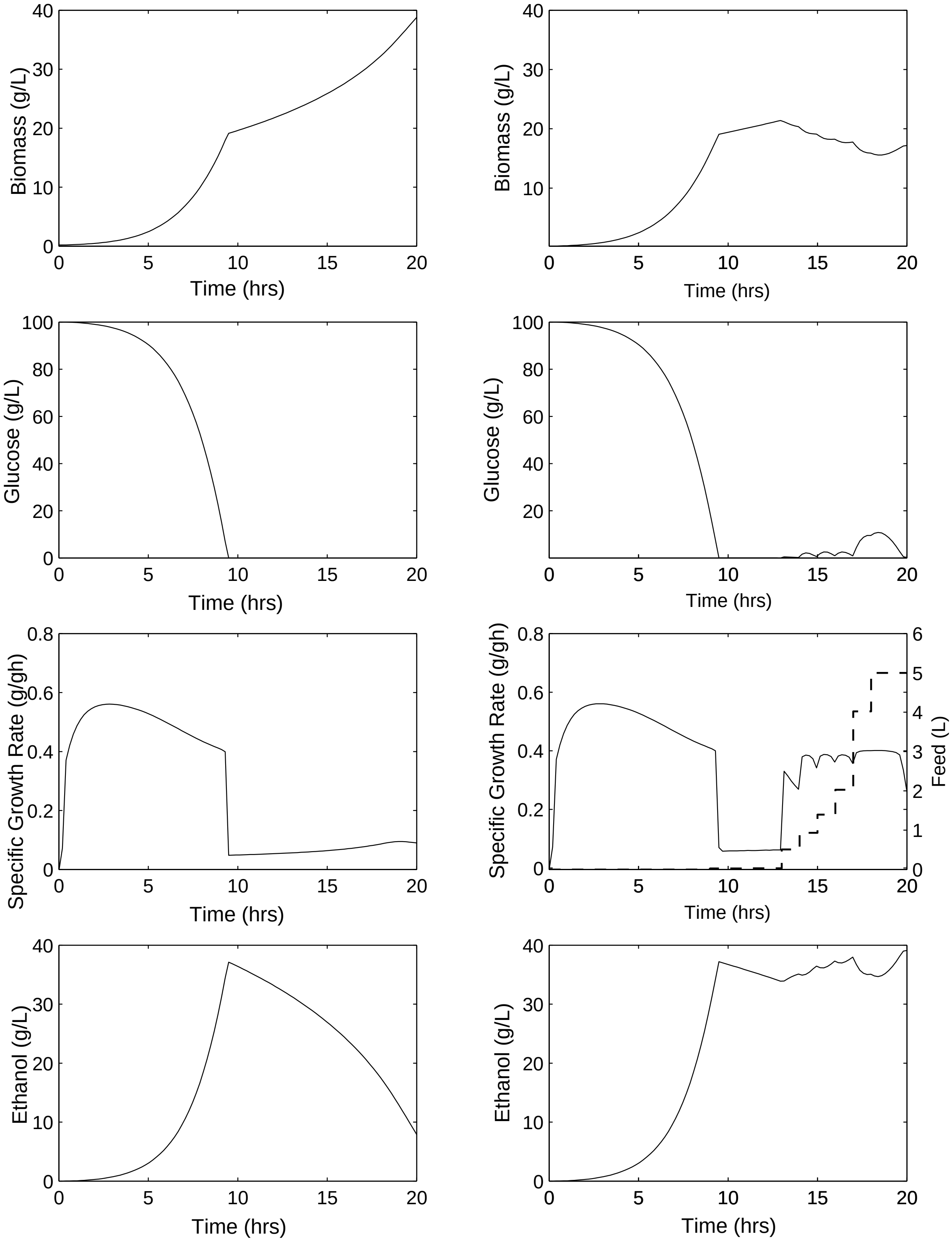


Figure 2.5: The plots show the state and specific growth rate profiles(—) in batch growth conditions (left) and when an input(- - -) is applied for the fed-batch case (right).
$(X_0 = 0.2\ g/L,\ S_0 = 100\ g/L,\ E_0 = 0\ g/L,\ V_f = 20\ L, S_i = 100\ g/L)$

The series of plots on the right, describe the concentration profiles for the same variables but with the fermentation proceeding in a fed-batch mode with a certain input profile as the feed with substrate concentration in the feed as 100 g/L. In this case, the specific growth follows a trend similar to that observed during the batch growth in the initial part of the fermentation. However, once feed is introduced during the course of the fed-batch the specific growth increases due to the glucose concentration in the feed. As the glucose from the feed is utilized, the specific growth rate reduces. The added feed also ensures that ethanol is not consumed as a substrate and this is reflected in the increase in ethanol concentration during the course of feed addition. Thus the model is able to capture the dynamics of sequential growth on glucose and ethanol in both batch, as well as fed-batch mode.

Unlike the model considered above, models that have been previously used to describe fed-batch baker's yeast fermentation [56],[59] do not include a detailed representation of the cellular level growth process. The growth processes in these models are described using a lumped, unstructured representation in which the biological components are described by a single variable, the specific growth rate, μ. The representative equations, as reported in [56] are,

$$\begin{aligned}
\frac{dX}{dt} &= \mu X - \frac{F}{V}X && \text{(2-19)}\\
\frac{dS}{dt} &= -\sigma X + \frac{F}{V}(S_i - S) && \text{(2-20)}\\
\frac{dP}{dt} &= \pi X - \frac{F}{V}P && \text{(2-21)}\\
\frac{dV}{dt} &= F && \text{(2-22)}\\
\mu &= \frac{0.408S}{0.22 + S}e^{-0.028P} && \text{(2-23)}\\
\pi &= \frac{S}{0.44 + S}e^{-0.015P} && \text{(2-24)}\\
\sigma &= \mu/0.1 && \text{(2-25)}
\end{aligned}$$

Here, μ, π and, σ are empirical relations describing the specific growth rate, substrate (S) consumption rate and, the product (P) formation rate respectively. The model simulation results for batch growth with the same initial conditions as used in the simulation for Figure 2.3 are shown in Figure 2.6. It is observed that the initial part of the batch fermentation proceeds in a similar

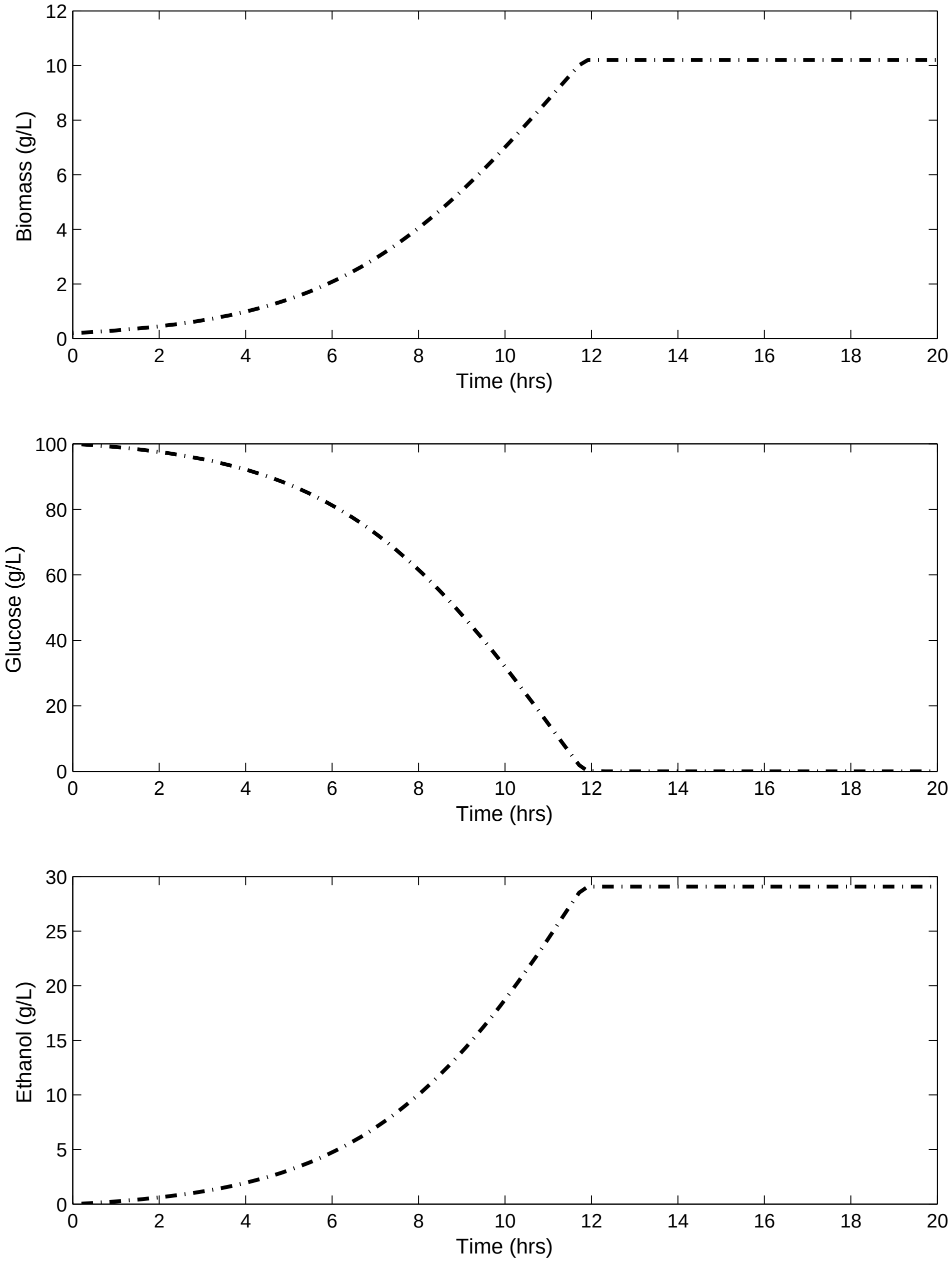


Figure 2.6: Plot showing growth behavior described using a lumped parameter approach for the biological model. ($X_0 = 0.2\ g/L,\ S_0 = 100\ g/L,\ E_0 = 0\ g/L$)

manner, with both the biomass and ethanol growing in an exponential manner. However, once the initial amount of glucose is completely consumed, the ethanol concentration remains constant and the model does not capture the biomass growth on ethanol. Thus biomass concentration remains constant, once the glucose is completely consumed. The comparison of the specific growth rates for the two models is shown in Figure 2.7. It can be seen that the lumped parameter model fails to account for the biomass growth on ethanol as the specific growth rate falls to zero as the glucose is completely consumed. This is due to the fact that the model structure is unable to capture the growth on ethanol and this drawback is a result of the simplified description of the cellular physiology.

2.2 Results from Modeling

In this chapter, some of the benefits of having a mathematical model for fed-batch bioreactor operation were presented. The uniqueness of the multi-scale modeling philosophy, and its applicability to the fed-batch bioreactor problem at hand, was discussed. The bioreactor model was presented in detail and its open-loop behavior was discussed. It was found that the model can capture the sequential growth of biomass on glucose and ethanol respectively. The characteristics of the specific growth were also analyzed for growth on both glucose as well as ethanol. Comparison between the multi-scale model and a lumped model were also presented. It was found that the conventional four state macroscopic reactor model with lumped biological representation, was unable to capture the sequential growth dynamics.

In order to satisfy a performance objective, one needs to determine the optimal substrate policy. Tools from optimal control theory are utilized in conjunction with the dynamic model described in this chapter to obtain the input profile that will maximize the performance objective, the end of batch ethanol concentration. The next chapter addresses the design of a controller to track the optimal ethanol trajectory in presence of unmodeled disturbances.

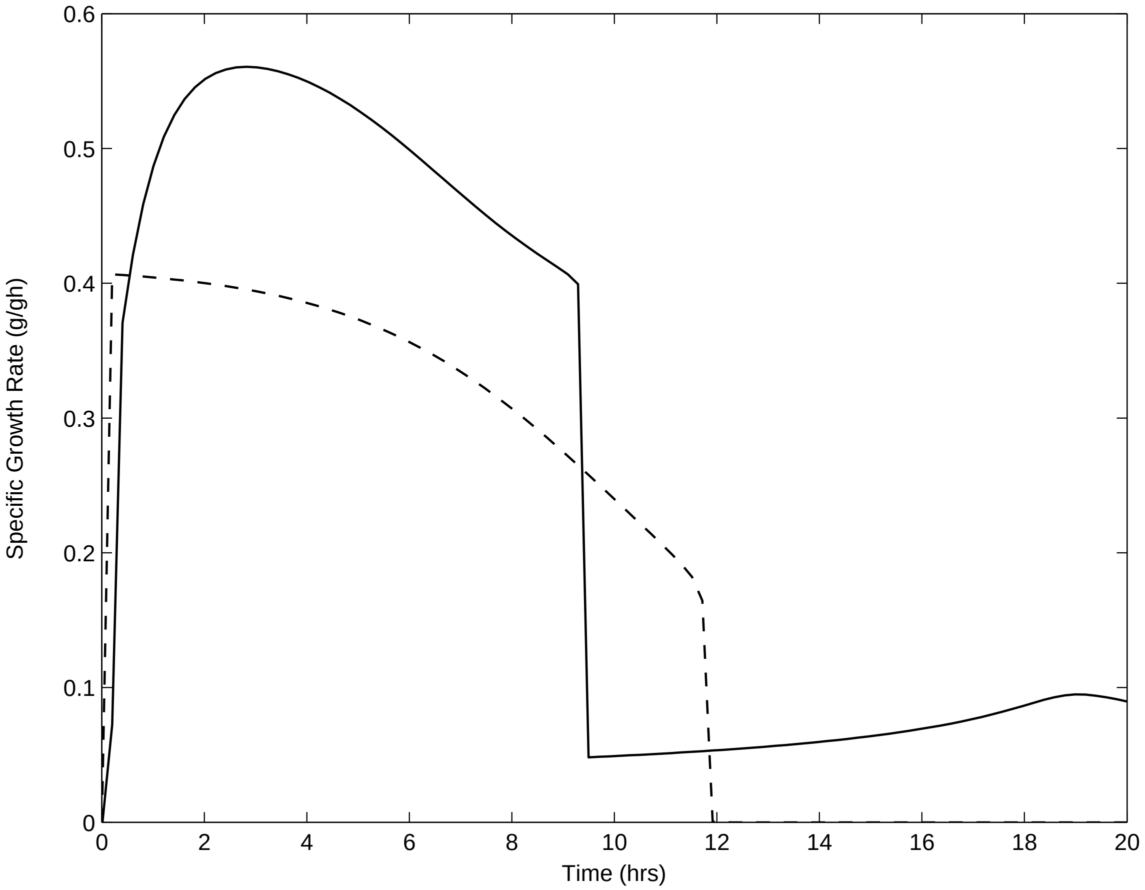


Figure 2.7: Comparison of the specific growth rate obtained from a lumped model (- - -) and the multi-scale model (—) considered in this work [3]

3.0 CONTROL

This chapter focuses on optimization and controller synthesis for the multi-scale, fed-batch bioreactor considered in this work and described by the model in Section 2.1. First, the open-loop optimal control problem is discussed and the off-line optimal substrate feeding policy for the system is determined. Next, the design of a controller to track the optimal profile on-line in a closed-loop is considered, and a discussion of the disturbance compensation strategies employed for the constrained optimization problem concludes the chapter.

3.1 Optimal Control

3.1.1 Motivation for optimal control

In the case of a fed-batch bioreactor, one goal is to maximize an appropriate performance objective. Towards achieving this goal, it is important to note that decisions made regarding the input during the course of the batch play an important role on the objective function. As an example, consider a hypothetical case study with two simulated runs for a fed-batch bioreactor starting with the same initial conditions and with the same total amount of feed. The bioreactor model used for the simulations was the same as that described in Section 2.1. Maximization of the ethanol concentration E, at the end of the batch was used as the performance measure. The only difference between the two case-studies is the time at which the feed is introduced in each of the simulations. The input substrate feed and the state variable profiles obtained for both of the simulated runs are shown in Figure 3.1. The plots on the left are generated with the glucose feed sent into the reactor during the latter part of the batch run, whereas those on the right, show the effect of the feed supplied during the earlier stages of the fed-batch operation. The ethanol concentration at the end of the batch is 31 g/L and 25 g/L respectively, thus demonstrating the importance of the feeding strategy for fed-batch fermentations. Hence it is required to determine the input profile that would maximize the end of batch ethanol concentration. This forms the conceptual basis for the optimal control problem.

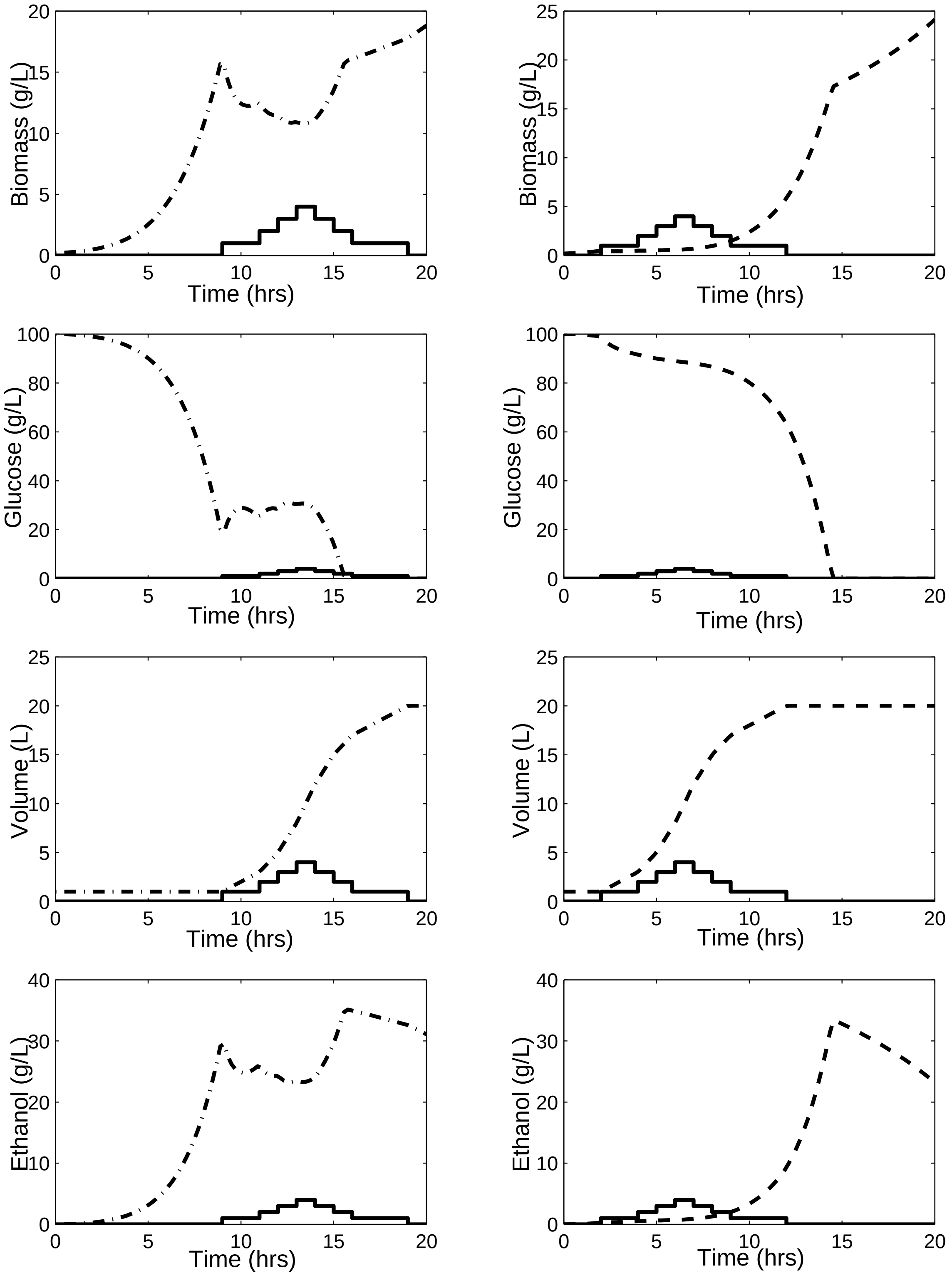


Figure 3.1: Fed-Batch simulation comparison of identical feed profiles (—) added late (left) and early (right). State variable profiles ($---$) are shown.
($X_0 = 0.2\ g/L, S_0 = 100\ g/L, V_0 = 1\ L, E_0 = 0\ g/L\ and\ S_i = 100\ g/L$)

3.1.2 Brief theory on optimal control

The general formulation of the optimal control problem is now presented. Let us consider that the system dynamics are described by,

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \qquad \text{for } \ t \geq t_0 \text{ and } t_0 \text{ is fixed} \tag{3-1}$$

In this equation, $\mathbf{x}(t)$ and $\mathbf{u}(t)$ are vector valued state and input respectively and, t_0 is the initial time. Associated with the process operation is an objective function that needs to be maximized and the general formulation for the objective function is given as,

$$\mathbf{J}(t_0) = \phi(\mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} \mathbf{L}(\mathbf{x}, \mathbf{u}, t)\ dt \tag{3-2}$$

The function ϕ accounts for the contribution of the final state, whereas $\mathbf{L}$, accounts for the path dependence in the objective function with t_f as the final time of operation. In its simplest form, the optimal control problem concerns with the maximization of the objective function $\mathbf{J}$, subject to the the system dynamics described by $\mathbf{f}$. However, due to physical limitations, additional constraint equations need to be included in the control problem. Constraints are usually present in the form of bounds on the state and/or manipulated variables. The detailed solution technique can be found in [51], and the main steps in the solution procedure are presented here. Initially, the necessary condition for $\mathbf{u}$ to be optimal is that it should maximize the Hamiltonian, which is obtained by appending the system dynamics to the objective function using a co-state variable. This is mathematically described as,

$$\mathcal{H}(\mathbf{x}, \mathbf{u}, t) = \mathbf{L}(\mathbf{x}, \mathbf{u}, t) + \lambda^T \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{3-3}$$

In this equation, λ is the co-state variable and is used to incorporate the system dynamics into the objective function. The original optimal control problem is then transformed into a two-point boundary value problem, as the differential equations for the state and the new co-state variables have boundary conditions defined at t_0 and at t_f, respectively.

Different approaches have been used for the solution of optimal control problems for fed-batch fermentors and these can be broadly divided into two distinct categories, one based on the Pontryagin's maximum principle [57], [59] and the other based on nonlinear optimization techniques [52], [53], [56]. A brief literature review on the application of these techniques for the control of fed-batch ethanol fermentation has been presented in Section 1.3.2 and for completeness, the two techniques are discussed briefly, below.

Pontryagin's Maximum Principle: Pontryagin and co-workers [50] showed that if the input profile $\mathbf{u}(t)$, for the optimal control problem is constrained, then the necessary condition for $\mathbf{u}(t)$ to be optimal, *i.e.* $\partial\mathcal{H}/\partial\mathbf{u} = 0$, is replaced by the more general relation,

$$\mathcal{H}(\mathbf{x}^\star, \mathbf{u}^\star, \lambda^\star, t) \quad \leq \quad \mathcal{H}(\mathbf{x}^\star, \mathbf{u}^\star + \delta\mathbf{u}, \lambda^\star, t), \quad \text{for all admissible } \delta\mathbf{u} \tag{3-4}$$

In this equation $\star$ denotes optimal quantities. The essence of the maximum principle is that the, *"Hamiltonian must be minimized over all admissible* $\mathbf{u}$ *for optimal values of the state and co-state"* [63]. On applying Pontryagin's Maximum Principle to the fed-batch fermentation problem, for optimizing the amount of ethanol at the end of the batch, the Hamiltonian simplifies to,

$$\mathcal{H} = \lambda^T \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{3-5}$$

The system equations describing the macroscopic reactor (equations 2-14 through 2-18) are given by $\mathbf{f}$. Since the control input, *i.e.* feed rate F, appears linearly in the system equations, the Hamiltonian is linear in the control variable and is therefore given as,

$$\mathcal{H} = \mathbf{l}(\mathbf{x}, \lambda)u \tag{3-6}$$

In this equation, $\mathbf{l}$ is a linear function. Now, if at some point during the process operation, $\mathbf{l} < \mathbf{0}$, then in order to maximize the Hamiltonian, $\mathbf{u}$ will be given by its lower bound, $\mathbf{u}_{min}$. In a similar manner, in regions where $\mathbf{l} > \mathbf{0}$, $\mathbf{u}$ will have the value $\mathbf{u}_{max}$. This shifting of the control input between its upper and lower bounds is known as a bang-bang control problem. On the other hand, the singular control problem arises when $\mathbf{l} = \mathbf{0}$ in some region. In this case, both the Hamiltonian and its first and higher derivatives are zero. The maximum principle then does not give any information about finding the optimal control input. As a result, part of the optimal trajectory may be singular, and switching times between the singular and non-singular regions need to be determined. Differential equations for both the state and co-state variables then need to be solved in the different regions, and the solution has been reported to be highly sensitive to the choice of initial conditions and the initial guess values used for the solution of the optimization problem [56]. In addition, when constraints are present on the control variable and/or the system state variables, the resulting problem poses sufficient numerical and computational challenges. The applicability of the maximum principle is therefore limited to low order problems, or problems with fairly low complexity in the process model.

Nonlinear Optimization Techniques: As an alternative to Pontryagin's maximum principle, the original optimal control problem can be converted into a Nonlinear Programming (NLP) problem. The mathematical basis for this method of solution is to make the original problem computationally tractable by using a discrete approximation to the original continuous input profile. In this method, the fed-batch interval (t_0, t_f) is partitioned into sub-intervals and the control variable is approximated by suitable functions, such as, piecewise constant functions, within each interval. The resulting problem is thus reduced to a mathematical programming problem that can be solved using standard optimization algorithms. Figure 3.2, adapted from [64], shows the controls discretized using piecewise constant functions. The switching times on the X axis are pre-assigned whereas, the heights of the piecewise constant functions on the Y axis are decision variables for the optimization algorithm.

3.2 Optimal Control Results

For the optimal control of the fed-batch ethanol fermentation considered in this work, the state variable, $\mathbf{x}$ and the system equations $\mathbf{f}$, are described by the multi-scale model represented by equations 2-14 through 2-18, so that,

$$\mathbf{x} = [X\ S\ V\ DOT\ E]^T \tag{3-7}$$

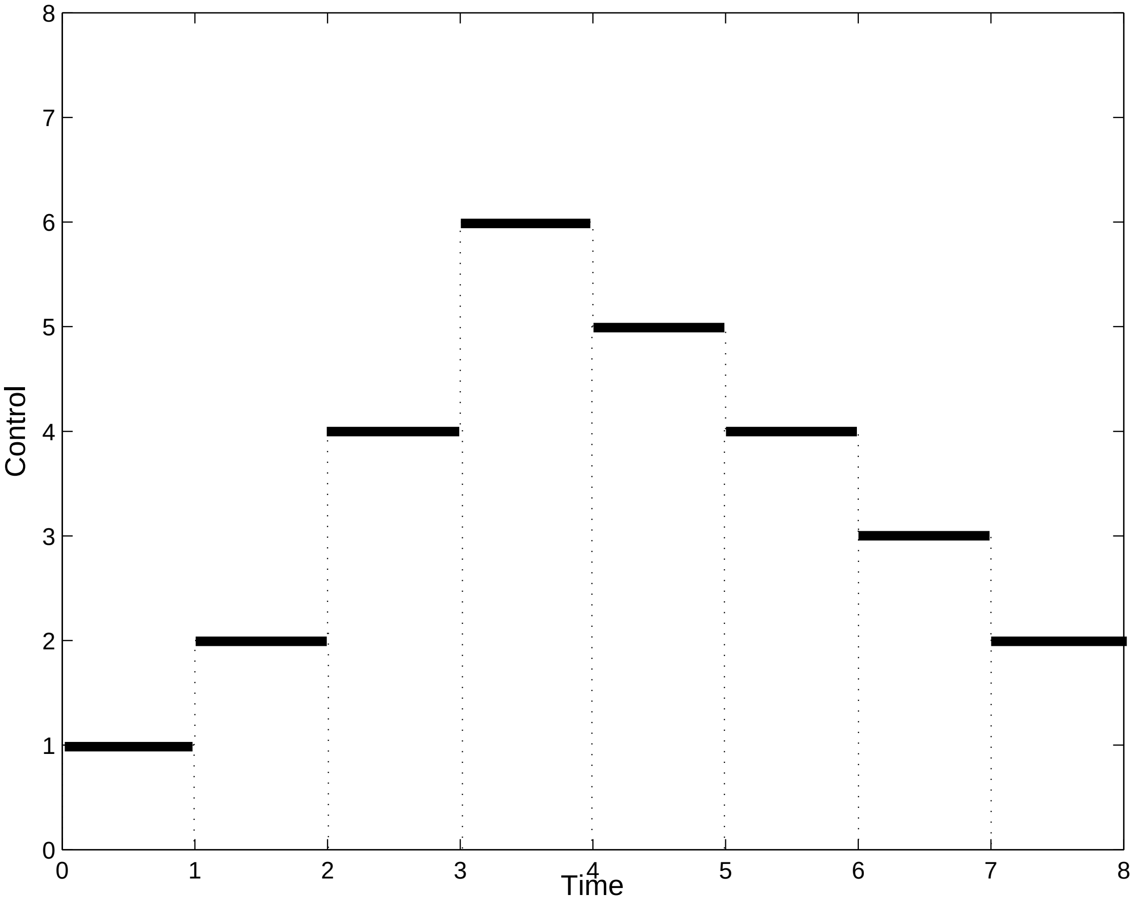


Figure 3.2: Plot showing the control input discretized using piecewise constant functions (adapted from [64])

where, X is the biomass, S is the substrate, V is the reactor volume, DOT is the dissolved oxygen tension, and E represents the ethanol concentration. The feed-rate F, serves as the control input $\mathbf{u}$ and the objective function for the optimal control problem is the maximization of ethanol concentration at the end of the batch given as,

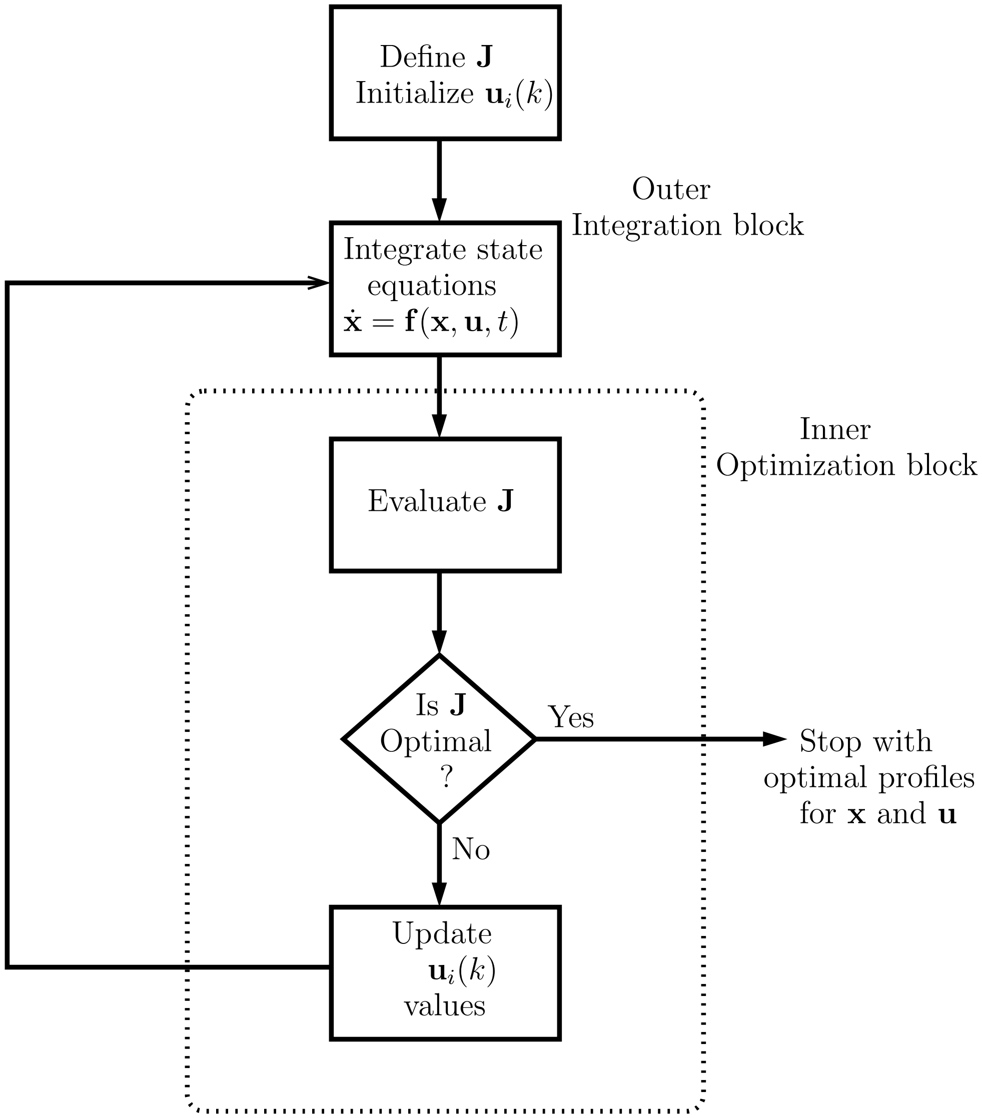
$$\min_{u(t)} \mathbf{J}[\mathbf{u}(t), \mathbf{x}(t)]$$

$$\mathbf{J} = E(t_f) \tag{3-8}$$

In addition it is assumed that the system is constrained so that the reactor volume at the end of the batch is restricted to 20 L, which is an end-point constraint. Further, the feed-rate cannot be negative and the feed rate over any interval cannot exceed the free reactor volume at the start of the interval, so that,

$$\begin{array}{ccccc} 0 & \leq & u(k) & \leq & 20 - V(k) \\ 0 & \leq & \sum_{k=t_0}^{t_f} u(k) & \leq & 20 - V_0 \end{array} \tag{3-9}$$

In these equations, $V(k)$ is the volume at the time k, and V_0 is the initial volume. Due to the system nonlinearities and constraints on the state and input, a nonlinear optimization technique was used to obtain the optimal input sequence. The final batch time was chosen as 20 hours (for comparison with [52],[56]). The entire batch was split into equal intervals of one hour duration. The optimal control was then discretized into 20 piece-wise constant segments, with values $u_i(k)$ for $k = 0, 1, 2, ...19$. Here, i is the iteration value during the course of the optimization, whereas k represents the time instant at which the input is applied. Thus the value of i increases as the optimization proceeds and the objective function is calculated for each iteration until it converges to its maximum or minimum value. The goal of the optimization procedure is to find the values of $u(k)$ for all k, which give an optimal value of the performance index, $\mathbf{J}$. The general algorithm for the solution procedure is described in Figure 3.3 adapted from [65]. The solution is composed of two distinct blocks, an optimization block and an integrator block. At the beginning, a starting guess value for the input is provided. Based on this input, the system differential equations are integrated and the value of the objective function is determined. The optimization algorithm uses a convergence criterion to determine the optimal value, failing which, new input values are



Define $\mathbf{J}$
Initialize $\mathbf{u}_i(k)$
Outer
Integration block
Integrate state equations
$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t)$
Inner
Optimization block
Evaluate $\mathbf{J}$
Is $\mathbf{J}$ Optimal ?
Yes
Stop with optimal profiles for $\mathbf{x}$ and $\mathbf{u}$
No
Update $\mathbf{u}_i(k)$ values

Figure 3.3: Algorithm for the solution of the optimal control problem

selected based on the gradient of the objective function at that point, and the sequence continues. The input profile for which the objective function converges to a maximum value, is the optimum profile. For the numerical integration of the state equations, the `ode45` routine in MATLAB (© 2002, The Mathworks, Natick, MA) was used, and the optimization was carried out using the MATLAB routine `fmincon`. The results of the optimization strategy are shown in Figure 3.4. The optimal control policy for the fed-batch fermentation shows two distinct regions. Initially, it is seen that the fermentation proceeds in a batch mode as the value of the substrate feed rate is zero; substrate addition then follows the batch period. The batch growth continues until all of the initial substrate, with which the fermentation process starts, is completely utilized. At this point, if the batch growth were to continue, *i.e.* if additional substrate was not fed in the reactor, biomass growth would continue on ethanol as a substrate, which is undesirable. Thus, in order to prevent this consumption of ethanol, glucose feed is sent into the reactor, and this is utilized for the production of ethanol, in preference to biomass.

This approach leads to the optimal glucose feeding profile for maximizing the end of batch ethanol concentration. For the initial conditions considered in this work, the profile is composed of an initial batch growth *i.e.* no feed addition and then the fed-batch growth. This is qualitatively similar to the optimal profile obtained by Pushpavanam *et al.* [56], and by Modak and Lim [59]. The off-line optimum profile provides the reference trajectory that the fed-batch operation must follow, in order to maximize the end of batch ethanol concentration in the absence of disturbances. The optimal control policy represents the best performance that can be obtained from the system for the given set of initial and feed conditions.

3.3 Model Predictive Control

As is seen from Section 1.3.2, most of the control literature for yeast cultures focuses on an open-loop operation owing to their highly nonlinear and inherently difficult dynamic behavior [6]. Optimization is carried out off-line, and the reactor is fed with the determined optimal feed profile. Once the batch proceeds there is no provision to account for disturbances occuring during the batch.

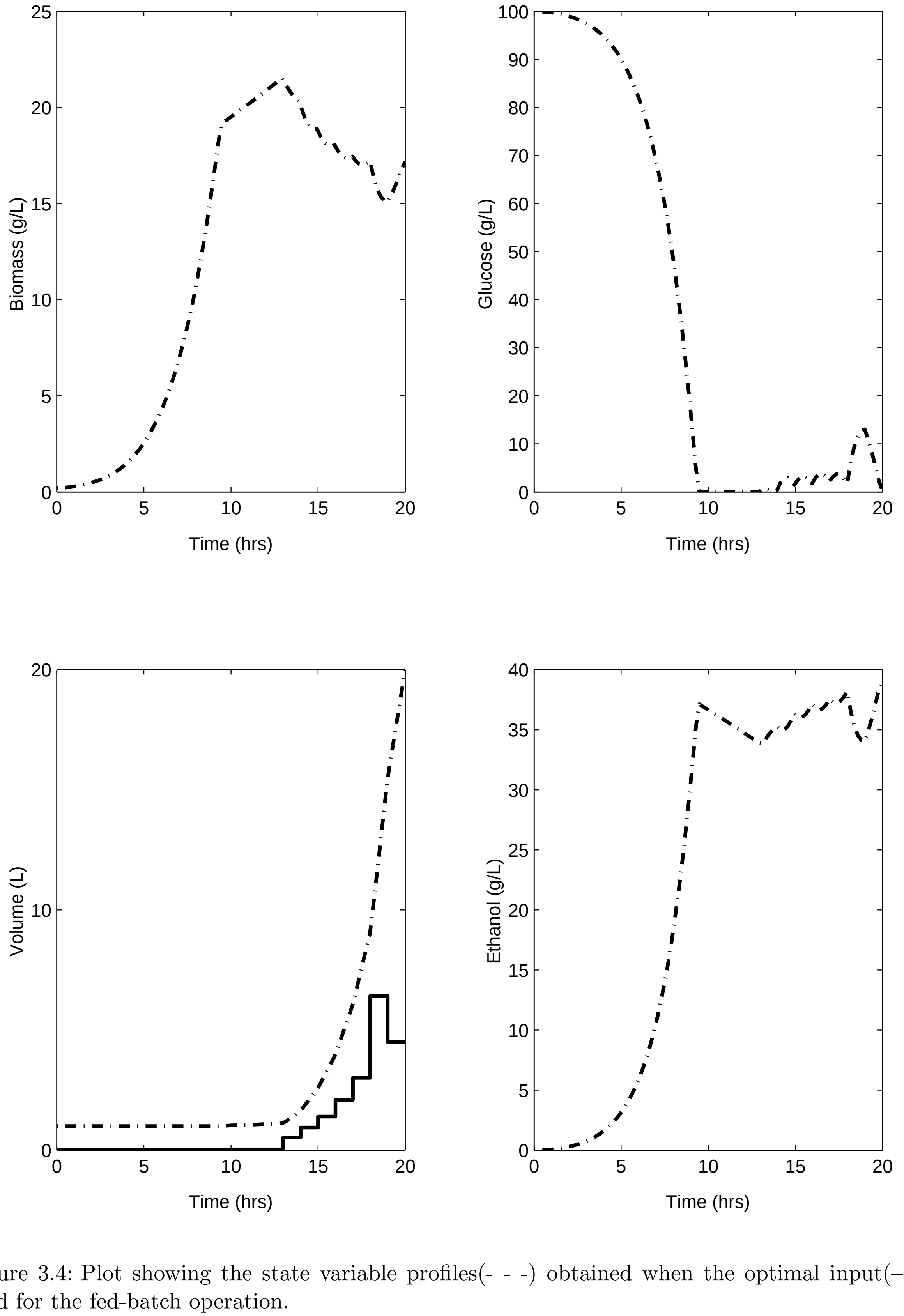


Figure 3.4: Plot showing the state variable profiles(- - -) obtained when the optimal input(–) is used for the fed-batch operation.
$(X_0 = 0.2\ g/L, S_0 = 100\ g/L, V_0 = 1\ L, E_0 = 0\ g/L\ and\ S_i = 100\ g/L)$

This problem of implementing the optimal policy on-line and ensuring that the system follows the optimal trajectory in the presence of disturbances has received limited prior consideration [52]. Attention is thus focused on designing a controller to track the optimal policy, and on approaches for disturbance compensation for the closed-loop control problem.

The control of fed-batch fermentors and polymerization reactor systems is similar in that both of these systems are operated in a dynamic, fed-batch mode and are highly nonlinear. As an example, Peterson and Arkun consider free radical polymerization of methyl methacrylate in fed-batch mode for the control of the reactor temperature and the number averaged molecular weight (NAMW) of the polymer [66]. They employ a nonlinear Dynamic Matrix Control (DMC) algorithm and the time-varying disturbances are calculated such that the output of the extended linear model matches the output from the nonlinear model at all future sampling times. Constraints were not included in their formulation, and for controller design, the prediction and control horizons were both set a value of one. In this work, the control formulation is treated in a slightly different manner.

Conventional Model Predictive Control (MPC), [67],[68],[69],[70],[71] which employs a receding horizon framework, is geared more towards continuous processes [72]. In contrast to continuous MPC, the control horizon in batch processes shrinks as the batch nears completion. Thus, the Shrinking Horizon Model Predictive Control (SHMPC) formulation is utilized in this study as it is more suited to batch processes where the available window for control shrinks as the end of batch nears. The SHMPC approach has been utilized by Liotta *et.al.* [73] for the control of particle size in a semi-batch emulsion polymerization reactor and by Thomas and co-workers for the quality control of composite laminates [74]. It was hypothesized that these concepts which have proved successful for control of polymerization reactors could be applied for the fed-batch bioreactor problem considered in this work.

An added advantage of using a predictive controller is the fact that, for fed-batch fermentors, decisions made during the course of the process operation have a significant impact on the final product quality, as demonstrated in Figure 3.1. In such a scenario, the model predictive control algorithm proves beneficial, as it can predict the effect of past input moves on the future system behavior. Since process feedback is incorporated into the algorithm at every time step, predicted future constraint violations can be accounted for, and subsequent input moves can be chosen so as to prevent these violations from taking place. The model predictive control algorithm is also well known for its explicit on-line handling of manipulated variable constraints. On the other hand, constraint handling is not easily incorporated with classical feedback techniques; furthermore classical feedback controllers respond only after the disturbance manifests itself in the process output, thereby limiting their applicability for constrained fed-batch control problems.

3.3.1 Model Predictive Control: The Basic Algorithm

In order to understand the Model Predictive Control (MPC) algorithm, consider Figure 3.5. The figure and the notation used in the description are adapted from [75]. The first part of the MPC algorithm is the specification of the reference trajectory, which may be as simple as a step change to a new setpoint or, as is common in batch processes, a trajectory that the system must follow. At the present time, k, the reference trajectory has a value $r(k)$.

Also at k, consider the predicted process output over a future prediction horizon, p. A suitable controller model of the process is used to obtain the projected behavior of the output over the prediction horizon by simulating the effects of past inputs applied to the actual process (value $y^*(k)$ at the current time).

The same controller model is used to calculate a sequence of m current and future manipulated variable moves, in order to satisfy some specified objective function. Here, m is the move horizon. A common objective function is to minimize the sum of squared deviations of the predicted controlled variable values from a time-varying reference trajectory, over the prediction horizon, based on system information available at the current time k. The minimization also takes into account constraints that may be present on the state, output, or the manipulated variable. Conceptually, the problem is similar to constructing and utilizing the inverse of the controller model to determine the sequence of m moves that most closely achieve the specified output behavior over the predicted horizon [75].

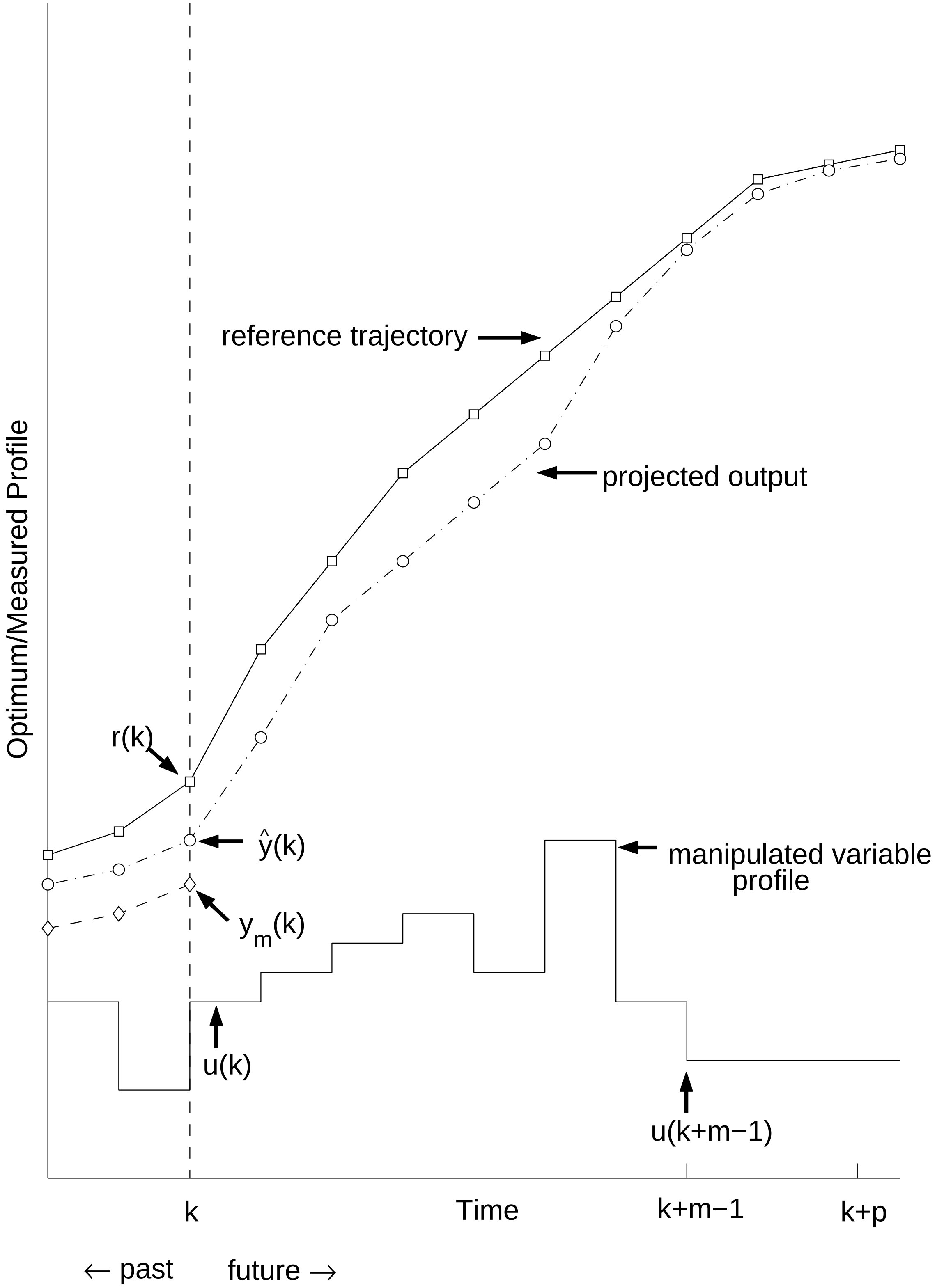


Figure 3.5: MPC algorithm schematic (adapted from [75])

However, due to unmodeled disturbances and modeling errors, there may be deviations between the actual observed output, $y_m(k)$ and the predicted output behavior. Due to this deviation, the computed future manipulated variable moves may no longer be appropriate and hence, only the first of the computed manipulated variable moves, $\Delta u(k)$, is implemented on the actual process. The error $d(k) = y_m(k) - \hat{y}(k)$, is calculated and is used to update the future measurements. At the next time instant, $k + 1$, the process measurement is again taken and the horizon is shifted forward by one step. Based on this new horizon and using the updated system information, the optimization is carried out again, and the process continues. Since, the horizon recedes, at the next time step, this is also known as a receding horizon control problem. However, in the case of batch systems where the final time of the process operation is specified, the available prediction horizon, and the window of opportunity for control, shrinks as the batch nears completion. Consequently, the value of the prediction horizon in the control algorithm, successively decreases as the end of batch nears. This is known as the SHMPC problem [4].

3.3.2 Linear and Nonlinear Model Predictive Control

Since its inception, Model Predictive Control has had considerable success in industry. In a recent survey, Qin and Badgwell report that MPC has found applications in diverse areas such as chemicals, food processing, aerospace, pulp and paper, and the automotive industries [76]. A number of comprehensive reviews on MPC have also appeared in literature [30], [71], [77]. MPC algorithms have been classified most generally on the basis of the type of controller model used. In the most general implementation of MPC, the model is a nonlinear function of the system states, manipulated inputs and outputs. The objective function and system constraints could be nonlinear as well, and this particular implementation is known as Nonlinear Model Predictive Control (NLMPC). Linear Model Predictive Control (LMPC) refers to the particular case when the controller model is linear. A quadratic objective function is commonly used, which is one of the most convenient forms for the on-line optimization problem; the input and output constraints, if present, are generally linear in the decision variables. However, as pointed out by Biegler and Rawlings, the constrained LMPC problem is no longer linear [78].

In the case of LMPC, the most commonly used model form is the finite convolution model. This convolution model is based on either step or impulse response models in which case the output is represented as:

$$\begin{aligned}
y(k) &= \sum_{i=0}^{M} h(i)u(k-i) && \text{Impulse Response Model,} \\
y(k) &= \sum_{i=0}^{M} s(i)\Delta u(k-i) && \text{Step Response Model, \quad where,} \\
\Delta u(k) &= u(k) - u(k-1)
\end{aligned}$$

Here the parameters $h(i)$ and $s(i)$ are the impulse and step response coefficients, which are usually obtained from plant data. The model memory, M, denotes the window over which past inputs affect the output significantly. Other model forms that have been used are the state-space format and the discrete transfer function format, and a description of these model forms can be found in [75].

3.4 Model Predictive Control: Bioreactor Case Study

In this section, the application of the Model Predictive Control for the fed-batch fermentation problem considered in this work is discussed. Considering NLMPC, a nonlinear optimization problem needs to be solved at every time step. The general methodology is to convert the objective function and the constraints into a nonlinear programming problem, and this approach has been used for the control of a CSTR by Eaton and Rawlings [79]. However, NLMPC is computationally demanding and the optimization problem may not be strictly convex, so that the solution may converge to local minima, or if it does converge to the global minima, may take a long time to do so [71]. This makes the on-line implementation of NLMPC a non-trivial task. Instead of NLMPC, the Nonlinear Quadratic Dynamic Matrix Control (NLQDMC) algorithm [5] is considered in this work and the notation used in this section is adapted from [80]. In NLQDMC the optimization (shown later in this section) is reduced to solving a quadratic program (QP) at every time step. A quadratic program is inherently convex and is thus mathematically more tractable as compared to general nonlinear optimization problems. Consequently, a global solution to the optimization problem can be assured within a specified tolerance at every time-step and this makes on-line implementation for fed-batch fermentation control possible.

3.4.1 Model Construction

The original bioreactor model, discussed in Section 2.1 is a continuous time, nonlinear model. In order to design a suitable controller for this process, a controller model is required to predict the process behavior. This prediction of the process output is composed of three parts:

Effect of Past Manipulated Variables on the Predicted Outputs: The effect of past manipulated variables on the predicted output, $y^*(k+1), y^*(k+2), ..., y^*(k+p)$, is defined as the value of the output, assuming no manipulated variable changes occur in the future. In the context of the bioreactor case study, it amounts to determining the ethanol trajectory over the prediction horizon, assuming no further changes in the substrate feed rate take place. This contribution was calculated as follows: the original process model in the form of continuous time nonlinear differential equations can be represented as:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}) \qquad (3\text{-}10)\end{aligned}$$

At the current time k, the plant measurement, $\mathbf{y}_m(k)$, the state vector at time k based on information at time $k-1$, $\mathbf{x}(k|k-1)$, and the past manipulated variable, $u(k-1)$, are known.

For $i = 1, 2, ...p$, where p is the prediction horizon, the nonlinear model is numerically integrated, with the input constant, *i.e.* $u(k+i-1) = u(k-1)$, to obtain values of the state variables, and hence the output variable. Since this is a fed-batch system with a fixed final time, the value of p is the number of time steps from the current time to the end of the batch.

Effects of Future Manipulated Variables on the Predicted Outputs: In order to calculate the effect of future manipulated moves on the predicted outputs, a step response model was used. The original nonlinear model in equation 3-10 was linearized at every time step of the controller calculation. Thus, based on $\mathbf{x}(k|k-1)$ and $u(k-1)$, the linear model is given as,

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \mathbf{A}_k\hat{\mathbf{x}} + \mathbf{B}_k\mathbf{u} \\ \hat{\mathbf{y}} &= \mathbf{C}_k\hat{\mathbf{x}} \end{aligned} \tag{3-11}$$

where,

$$\begin{aligned} \mathbf{A}_k &= \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right)\Big|_{\mathbf{x}=\hat{\mathbf{x}}(k|k-1),u=u(k-1)} \\ \mathbf{B}_k &= \left(\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right)\Big|_{\mathbf{x}=\hat{\mathbf{x}}(k|k-1),u=u(k-1)} \\ \mathbf{C}_k &= \left(\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right)\Big|_{\mathbf{x}=\hat{\mathbf{x}}(k|k-1)} \end{aligned}$$

This is in contrast to approaches where linearization is carried out at only one point and the resulting linear model is used for further calculations. Due to the dynamic nature of the fed-batch operation, there is no steady state, and hence the process behavior cannot be represented accurately by linearizing around a single operating point. Another factor that contributed towards linearizing the model at every time-step was the fact that the system is subject to abrupt changes, such as when the original amount of substrate is completely consumed, or when feed is introduced into the reactor. Thus the system nonlinearities and dynamics necessitated linearization at every time step. Analytical linearization of the original nonlinear model was carried out off-line to obtain a function that enabled the calculation of the linear model on-line based on the values of the estimated state and past input variables. A comparison between the nonlinear and linear models for the optimal input is shown in Figure 3.6. It is found that the linear model, based on a dynamic linearization, adequately represents the process. The next step is to discretize the linear model to obtain the discrete-time model in state-space format,

$$\begin{aligned} \hat{\mathbf{x}}_{j+1} &= \Phi_k\hat{\mathbf{x}}_j + \Gamma_k\mathbf{u}_j \\ \mathbf{y}_j &= \mathbf{C}_k\hat{\mathbf{x}}_j \end{aligned} \tag{3-12}$$

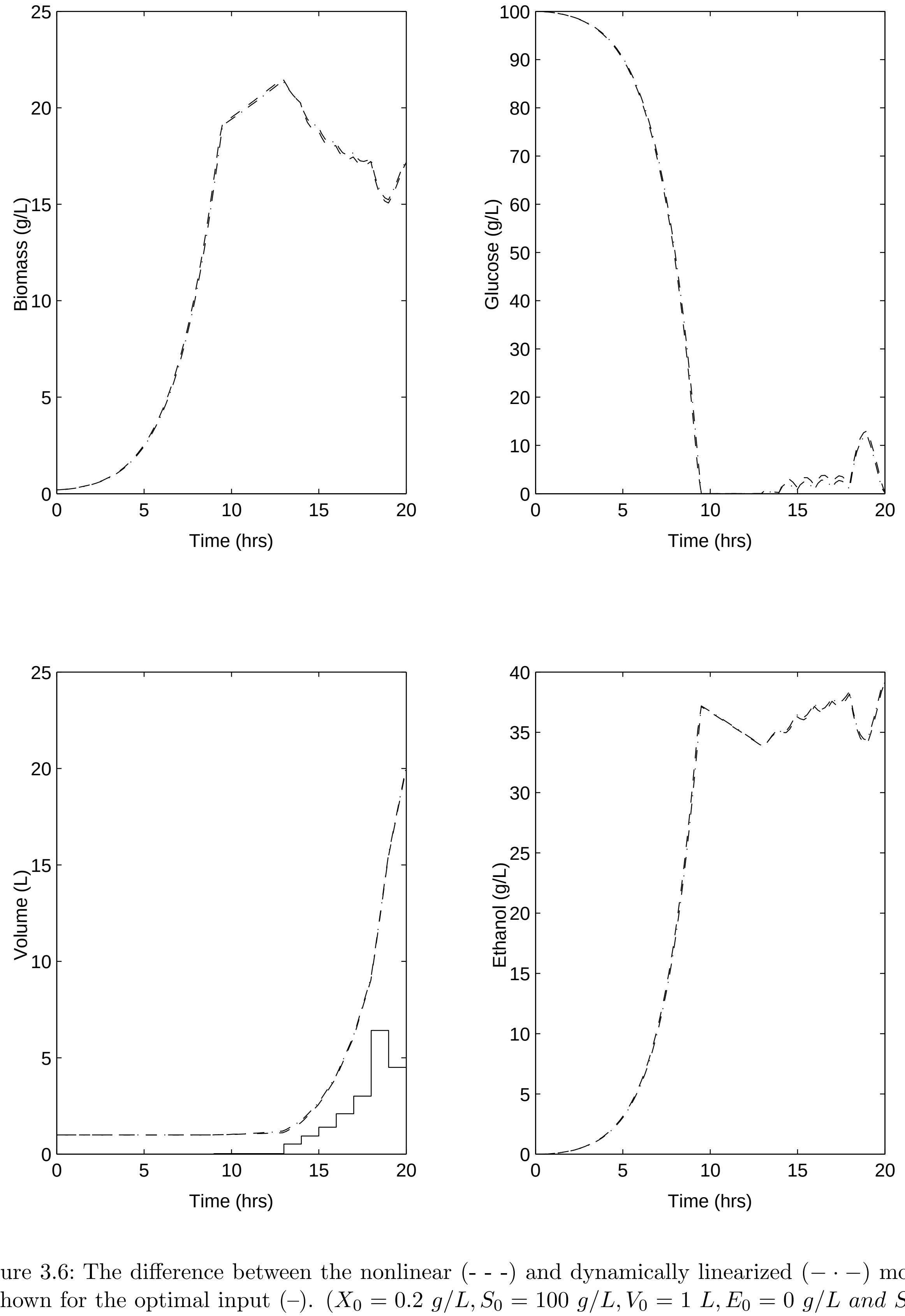


Figure 3.6: The difference between the nonlinear (- - -) and dynamically linearized ($- \cdot -$) model is shown for the optimal input (–). ($X_0 = 0.2\ g/L, S_0 = 100\ g/L, V_0 = 1\ L, E_0 = 0\ g/L\ and\ S_i = 100\ g/L$)

In this equation, Φ_k and Γ_k are the discrete time equivalents of matrices $\mathbf{A}_k$ and $\mathbf{B}_k$ respectively. Once the discretized model was obtained, step response coefficients for the system were calculated. The calculations for discretization and the step response coefficients were performed using the functions `c2dmp` and `ss2step`, in MATLAB (© 2002, The Mathworks, Natick, MA). Since the step response coefficients are used to calculate the effect of the future manipulated variable moves, only p coefficients are required. These step responses were calculated from,

$$\mathbf{S}_{i,k} = \sum_{j=1}^{i} \mathbf{C}_k \Phi_k^{j-1} \Gamma_k \quad (i = 1, 2, ...p) \tag{3-13}$$

In this equation, $\mathbf{S}_{i,k}$ represents the step response coefficient i time steps in the future, calculated from the current time k. Using the step response coefficients, the dynamic system matrix was obtained. This matrix is known as the step response matrix and at a particular time instant k, it is of the form,

$$\mathcal{S}_k^u = \begin{bmatrix} S_1^u & 0 & 0 & \cdots & 0 \\ S_2^u & S_1^u & 0 & \cdots & 0 \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ S_m^u & S_{m-1}^u & S_{m-2}^u & \cdots & S_1^u \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ S_p^u & S_{p-1}^u & S_{p-2}^u & \cdots & S_{p-m+1}^u \end{bmatrix} \tag{3-14}$$

The step response matrix has p rows and m columns. The first column represents the step response co-efficients extending until the prediction horizon, and hence p rows in the step response matrix. The second column represents the next instant and it has the coefficients shifted down by one element. Thus the m columns extend until the control horizon with the system step response coefficients appropriately shifted down in order.

Since the model linearization was carried out at every time step, the discretization, step-responses, and the dynamic matrix, were calculated at every time step as well. In MPC $m \leq p$ always, so that the controller determines the next m moves only. Thus the vector of current and future manipulated variables is given as,

$$\Delta\mathcal{U}(k|k) = [\ \Delta u(k|k)\ \Delta u(k+1|k)\ ....\ \Delta u(k+m-1|k)\ ]^T \tag{3-15}$$

All subsequent input changes are set to zero, so that,

$$\Delta u(k+m|k) = \Delta u(k+m+1|k) = ... = \Delta u(k+p-1|k) = 0 \tag{3-16}$$

Effects of Future Disturbances on the Predicted Outputs: The unmodeled effects at the current sampling time are computed as the difference between the plant measurement obtained from the actual nonlinear model, $y_m(k)$, and the model prediction, $\hat{y}(k)$. In the absence of any further information about these disturbances at future times, it is assumed that the predicted values of the disturbances are equal to the current values.

$$d(k+1) = d(k+2) = ... = d(k+p-1) = d(k) = y_m(k) - \hat{y}(k) \tag{3-17}$$

With these three contributions, the p step ahead, predicted output $\hat{\mathcal{Y}}$, is given as,

$$\hat{\mathcal{Y}}(k+1|k) = \mathcal{Y}^*(k|k) + \mathcal{D}(k|k) + \mathcal{S}_k^u \Delta\mathcal{U}(k|k) \tag{3-18}$$

where, $\mathcal{Y}^*$ represents the contribution of past inputs, $\mathcal{S}_k^u \Delta\mathcal{U}(k|k)$ represents the effect of future manipulated variables on the output, and $\mathcal{D}$ represents the unmodeled disturbance vector, of length p. In matrix form, the predicted output is given as,

$$\begin{bmatrix} \hat{y}(k+1|k) \\ \hat{y}(k+2|k) \\ . \\ . \\ \hat{y}(k+m|k) \\ . \\ . \\ \hat{y}(k+p|k) \end{bmatrix} = \begin{bmatrix} y^*(k+1|k) \\ y^*(k+2|k) \\ . \\ . \\ y^*(k+m|k) \\ . \\ . \\ y^*(k+p|k) \end{bmatrix} + \begin{bmatrix} d(k) \\ d(k) \\ . \\ . \\ d(k) \\ . \\ . \\ d(k) \end{bmatrix}$$

$$+ \begin{bmatrix} S_1^u & 0 & 0 & \cdots & 0 \\ S_2^u & S_1^u & 0 & \cdots & 0 \\ . & . & . & \cdots & . \\ . & . & . & \cdots & . \\ S_m^u & S_{m-1}^u & S_{m-2}^u & \cdots & S_1^u \\ . & . & . & \cdots & . \\ . & . & . & \cdots & . \\ S_p^u & S_{p-1}^u & S_{p-2}^u & \cdots & S_{p-m+1}^u \end{bmatrix} \begin{bmatrix} \Delta u(k+1|k) \\ \Delta u(k+2|k) \\ . \\ . \\ \Delta u(k+m|k) \\ . \\ . \\ \Delta u(k+p|k) \end{bmatrix}$$

3.4.2 Controller Synthesis

After formulating the controller model, the essential components of the closed-loop control algorithm are considered. These are the reference trajectory that needs to be tracked, the objective function for the on-line optimization, and its associated constraints.

Specification of the Reference Trajectory: The optimal input that was generated in Section 3.2 was used to generate the reference trajectory for ethanol that has to be followed in the closed-loop format. The reference trajectory was also specified with the same step size as was used in the discretization of the controller model. This reference trajectory, $\mathcal{R}(k|k)$ for ethanol, with the optimal input is shown in Figure 3.7.

Objective Function: The 2-norm squared objective function was used in this work and is of the following form,

$$\min_{\Delta\mathcal{U}(k|k)} \left\{ ||\Gamma_y \left[\hat{\mathcal{Y}}(k+1|k) - \mathcal{R}(k+1|k) \right] ||_2^2 + ||\Gamma_u \Delta\mathcal{U}(k|k)||_2^2 \right\} \tag{3-19}$$

In this equation, the first term denotes the projected deviations of the controlled variable, the ethanol trajectory ($\mathcal{Y}$) from the time-varying, ethanol reference trajectory ($\mathcal{R}$) whereas, the second term penalizes the manipulated variable moves ($\Delta\mathcal{U}$). The quadratic criterion penalizes large deviations more than smaller ones, so that the output remains close to its reference trajectory. It may so happen that the manipulated variable moves that make the output follow the reference trajectory may be too severe and may not be physically realizable due to actuator saturation or system constraints. Thus, weights are included in the objective function so that the relative importance of the trajectory tracking and the control effort can be decided by the values of the (usually) diagonal weights, Γ_y and Γ_u, respectively. Larger values of Γ_u relative to Γ_y penalize changes in the input. This results in lower values of $\Delta\mathcal{U}$, and as a result the reference trajectory is not followed as closely. On the other hand, larger relative values of Γ_y result in good tracking characteristics at the cost of potentially large changes in the input.

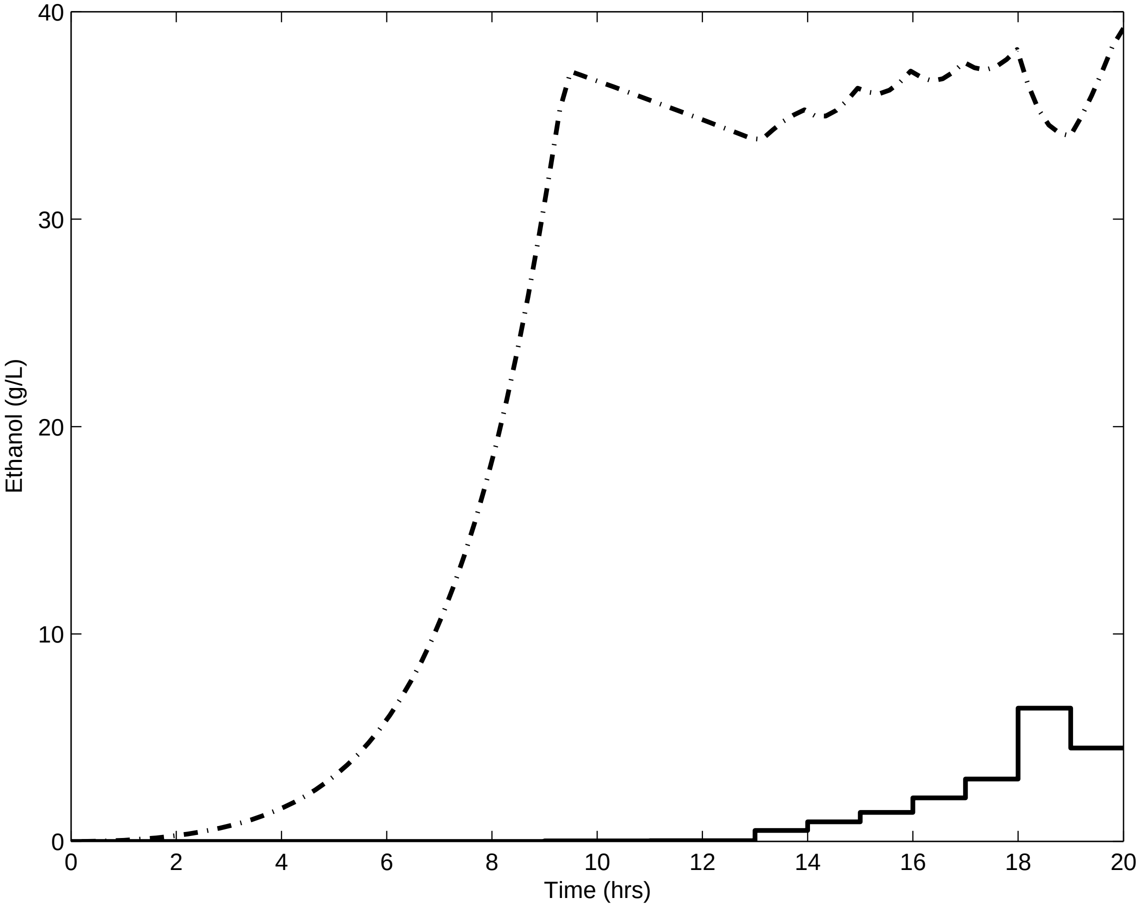


Figure 3.7: Plot showing the ethanol reference trajectory $(- \cdot -)$ and the optimal input profile $(- - -)$. $(X_0 = 0.2\ g/L, S_0 = 100\ g/L, V_0 = 1\ L, E_0 = 0\ g/L\ and\ S_i = 100\ g/L)$

Constraints: Constraints play a very important role in the quadratic programming problem solved at every time step of the controller calculation. Since the reactor is constrained to a fixed final volume, the sum of the manipulated variables *i.e.* the total substrate feed cannot exceed the final reactor volume bound of 20 L. Consider any point k during the reactor operation. The reactor volume at k is known to be $V(k)$, so that $(20 - V(k))$ represents the free volume available in the reactor. Since m future control intervals are considered in the MPC algorithm, this free volume represents the upper bound for the feed addition over the control horizon. This manipulated variable constraint for future time intervals can be formulated as m linear inequalities [80] resulting in,

$$u_{min}(l|k) \leq u(k-1) + \sum_{j=0}^{l} \Delta u(k+j|k) \leq u_{max}(l|k) \quad \text{for } l = 0, 1, ..., m-1 \tag{3-20}$$

In this equation, $\Delta u(k+j|k)$ is a scalar and represents a series of future manipulated variable moves for l, $u(k-1)$ is the past value of the manipulated variable. The lower bound on the input u_{min} is the flow rate at the previous time step; this guarantees a non-decreasing volume. The upper bound on the input, u_{max}, is given as:

$$\begin{aligned} u_{max}(l|k) &= \frac{20 - V(k+l|k)}{(p-l)\Delta t} \quad \text{for } l = 0, 1, ..., m-1 \\ V(k+l|k) &= V(k) + \sum_{j=0}^{l} \Delta u(k+j|k) \end{aligned} \tag{3-21}$$

Similarly, constraints on the manipulated variable rate can be formulated as m inequalities resulting in,

$$\Delta u_{min}(l|k) \leq \sum_{j=0}^{l} \Delta u(k+j|k) \leq \Delta u_{max}(l|k) \quad \text{for } l = 0, 1, ..., m-1 \tag{3-22}$$

Here Δu_{min} ensures a non-negative feed rate into the reactor whereas Δu_{max} ensures that a single input or the sum of all input additions do not lead to overflow of the reactor.

The manipulated variable magnitude and rate constraints can be then combined into a single convenient expression given as,

$$\begin{aligned}
\mathcal{C}^u \Delta\mathcal{U}(k|k) &\geq \mathcal{C}(k+1|k) \qquad \text{where,} \qquad\qquad (3\text{-}23)\\
\mathcal{C}^u &= \begin{bmatrix} -I_L \\ I_L \\ -I \\ I \end{bmatrix}\\
I_L &= \begin{bmatrix} I & 0 & \cdots & 0 \\ I & I & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ I & I & \cdots & I \end{bmatrix} \qquad \text{and,}\\
\mathcal{C}(k+1|k) &= \begin{bmatrix} u(k-1) - u_{max}(k|k) \\ \cdot \\ \cdot \\ u(k-1) - u_{max}(k+m-1|k) \\ u_{min}(k|k) - u(k-1) \\ \cdot \\ \cdot \\ u_{min}(k+m-1|k) - u(k-1) \\ -\Delta u_{max}(k|k) \\ \cdot \\ \cdot \\ -\Delta u_{max}(k+m-1|k) \\ \Delta u_{min}(k|k) \\ \cdot \\ \cdot \\ \Delta u_{min}(k+m-1|k) \end{bmatrix}
\end{aligned}$$

In this equation, $\mathcal{C}$ represents all the error vectors on the constraints *i.e.* deviations of u and Δu from their upper and lower bounds and is a function of the time instant k. I_L is a lower triangular matrix, whereas $\mathcal{C}^u$ contains all the matrices on the Left Hand Side of the inequalities.

Control Problem as a Quadratic Program: Attention is now focused on formulating the optimization problem as a QP problem to be solved at every time step. The generalized optimization problem and its associated constraints are,

$$\begin{aligned}
\min_{\Delta\mathcal{U}(k|k)} &\left\{ ||\Gamma_y \left[\hat{\mathcal{Y}}(k+1|k) - \mathcal{R}(k+1|k) \right] ||_2^2 + ||\Gamma_u \Delta\mathcal{U}(k|k)||_2^2 \right\} \\
\text{s.t. } &\hat{\mathcal{Y}}(k+1|k) = \mathcal{Y}^*(k|k) + \mathcal{S}_k^u \Delta\mathcal{U}(k|k) + \mathcal{D}(k|k) \qquad (3\text{-}24)\\
&\mathcal{C}^u \Delta\mathcal{U}(k|k) \geq \mathcal{C}(k+1|k)
\end{aligned}$$

where,

$$\mathcal{R}(k+1|k) = \begin{bmatrix} r(k+1|k) \\ r(k+2|k) \\ \cdot \\ \cdot \\ r(k+p|k) \end{bmatrix} \tag{3-25}$$

is the vector of reference trajectories.

By substituting the prediction equation into the objective function and writing the problem in the standard QP formulation, the optimization problem becomes,

$$\min_{\Delta\mathcal{U}(k|k)} \Delta\mathcal{U}(k|k)^T \mathcal{H}_k^u \Delta\mathcal{U}(k|k) - \mathcal{G}(k+1|k)^T \Delta\mathcal{U}(k|k) \tag{3-26}$$
$$\text{s.t.} \;\; \mathcal{C}^u \Delta\mathcal{U}(k|k) \geq \mathcal{C}(k+1|k)$$

where, the Hessian for the QP is,

$$\mathcal{H}_k^u = \mathcal{S}_k^{uT} \Gamma_y^T \Gamma_y \mathcal{S}_k^u + \Gamma_u^T \Gamma_u \tag{3-27}$$

and the gradient vector is,

$$\mathcal{G}(k+1|k) = 2\mathcal{S}_k^{uT} \Gamma_y^T \Gamma_y E_p(k+1|k) \tag{3-28}$$

where,

$$E_p(k+1|k) = \begin{bmatrix} e(k+1) \\ e(k+2) \\ \cdot \\ \cdot \\ e(k+p) \end{bmatrix} \tag{3-29}$$

$$= \mathcal{R}(k+1|k) - [\mathcal{Y}^*(k|k) + \mathcal{D}(k|k)] \tag{3-30}$$

3.4.3 Simulation Results

The NLQDMC algorithm was implemented in MATLAB (© 2002, The Mathworks, Natick, MA) and the QP was solved on-line, at every controller time step using the function `quadprog`. The results of the closed-loop control are as shown in Figure 3.8. The parameters that were used in the controller design were: a control horizon m of unity; a sample time of 1 hr; a manipulated variable move weight $\Gamma_u = 2$; and a setpoint tracking weight $\Gamma_y = 3$. The objective value from the open-loop

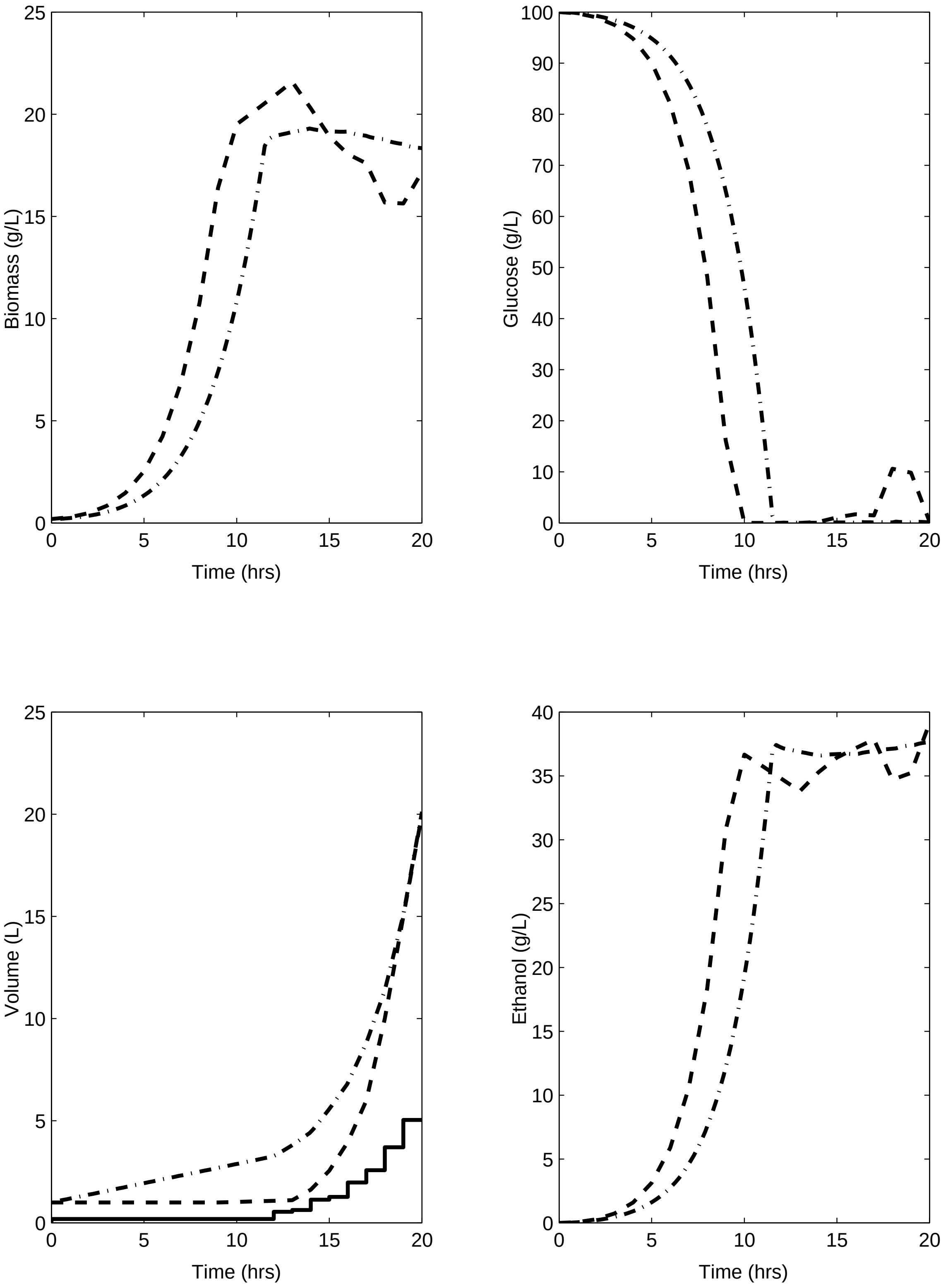


Figure 3.8: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($---$), and the closed-loop input profile(—) under nominal conditions. ($m = 1, \Gamma_u = 2\ \Gamma_y = 3$)

optimal control policy is 39.17 g/L, whereas that from the closed-loop NLQDMC scheme is 37.6 g/L. The objective function value obtained is thus considerably less (4%) than that obtained from the open-loop optimization. Although the volume constraint is satisfied, the controller does not track the reference trajectory satisfactorily. Further, it was observed from a number of simulations, that the controller performance was highly sensitive to the tuning parameters, Γ_u and Γ_y. However, improvement in tracking performance was obtained for different tuning parameters. Since the volume constraint was formulated as an inequality constraint, the closed-loop solution occasionally led to a volume which was less than the final volume constraint of 20 L, so that the total amount of ethanol at the end of the batch, $E_f \times V_f$, was reduced.

Consequently, in order to attain a final volume as close to 20 L as possible, the objective function for the optimization problem was modified to include a term that penalized deviations from the final reactor volume. The modified objective function is given as,

$$\min_{\Delta\mathcal{U}(k|k)} \left\{ ||\Gamma_y \left[\hat{\mathcal{Y}}(k+1|k) - \mathcal{R}(k+1|k)\right]||_2^2 + ||\Gamma_u \Delta\mathcal{U}(k|k)||_2^2 + ||\Gamma_f \left[V_f - 20.0\right]||_2^2 \right\} \tag{3-31}$$

In this equation, Γ_f is the weighting factor that penalizes deviations from the final volume, whereas V_f is the projected final volume at the end of the batch. Inclusion of this additional term in the objective function led to a considerable increase in performance and this can be seen in Figure 3.9. The sampling time was 1 hr, and the weighting factors used were, $\Gamma_y = 3$, $\Gamma_u = 10$, and $\Gamma_f = 3$. The control horizon was set equal to the prediction horizon, *i.e.* $m = p$. The ethanol concentration at the end of the batch was 38.91 g/L, which is very close to (within 0.6% of) the open-loop objective function value of 39.17 g/L. The reactor volume constraint was also satisfied. A comparison of the input profiles obtained from the closed-loop SHMPC and the off-line open-loop optimization is shown in Figure 3.10. It is observed that the closed-loop SHMPC profile differs from the open-loop input profile at the beginning of, and at the very end of the batch. There is an initial feed addition of 0.25 L in SHMPC, and at the end of the batch the changes in the feed rate are not as abrupt as those observed in the open-loop input profile. The reason for the initial addition could be attributed to the fact that, at the start of the batch when no feed addition has taken place, there is a considerable difference between the reference trajectory and the projected output trajectory. In order to account for this difference, the controller introduces the glucose feed. But as the batch proceeds, better predictions become available and the controller balances out the initial feed addition and does an admirable job of tracking the reference trajectory.

Another interesting feature observed in the simulations was the role played by the control horizon on the final performance. For a value of $m = 1$, in Figure 3.11, the control horizon is limited to the next step. Consequently, the controller calculates an input profile valid only for the next step, and this myopic view results in continuous feed addition. As a result, the final concentration of ethanol at the end of the batch is limited to 33.6 g/L. As the control horizon length increases, the controller generates input moves that are valid over a larger time horizon in the future as compared to $m = 1$. As a result, there is better control over the feed addition, the initial feed addition is reduced, and the amount of ethanol obtained at the end of the batch progressively increases. In Figure 3.12, with $m = 5$, 36.534 g/L of ethanol is obtained at the end of the batch, whereas in Figure 3.13, for $m = 9$ the amount of feed added initially is considerably reduced and results in a final ethanol concentration of 38.81 g/L which is very close to 38.91 g/L obtained when $m = p$.

Controller Performance for Disturbance Rejection: In order to test the performance of the controller for disturbance rejection, a +10% change in the glucose feed concentration was made. The sample time, and the weighting factors for the objective function, were the same as those used in the nominal case, and the results are shown in Figure 3.14. It is observed that the controller performance deteriorates, and attains a final ethanol concentration of only 25.6 g/L. Due to the increase in glucose concentration in the feed, the controller initially sends a reduced volume of feed into the reactor. However, as the batch nears completion, the controller adds an increased amount of feed in order to avoid violating the final volume constraint. As a result of this delayed feed addition, the concentration of ethanol is significantly reduced due to the dilution effect of the added volume. At the same time, considerable amount of glucose remains unutilized at the end of the batch which is not an efficient utilization of the available substrate.

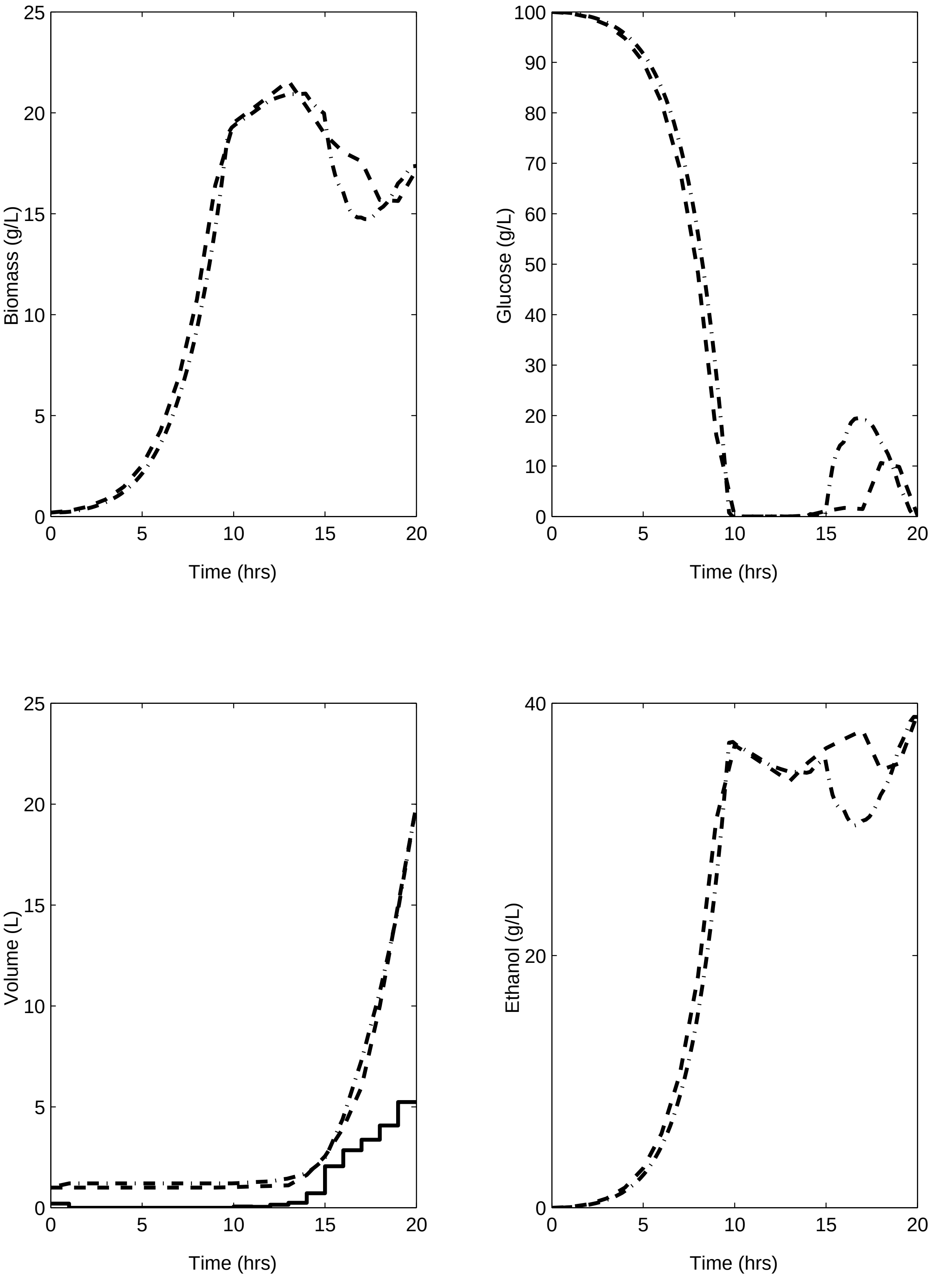


Figure 3.9: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($- - -$), and the closed-loop input profile(—) under nominal conditions for prediction horizon. ($m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 3$)

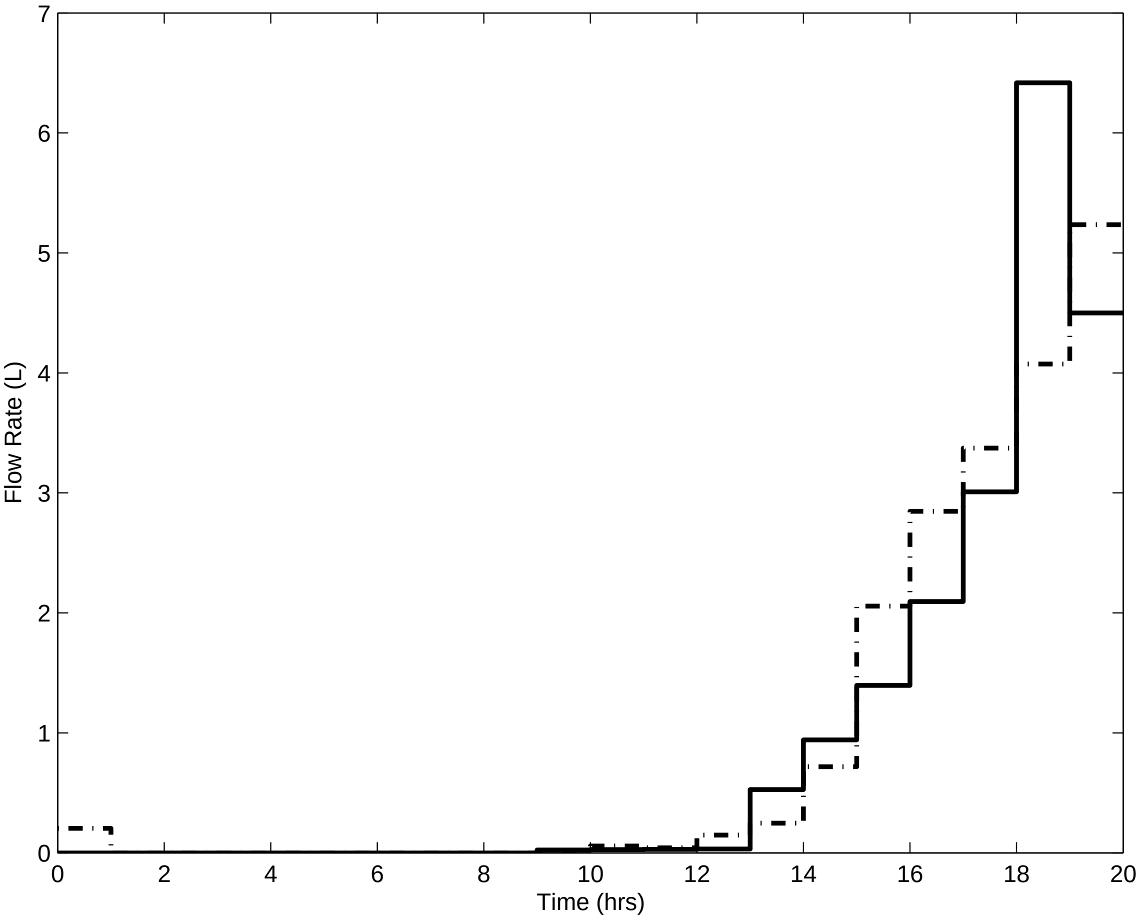


Figure 3.10: Plot of the input profiles obtained using closed loop NLQDMC($- \cdot -$) and the open-loop input profile(—) under nominal conditions for prediction horizon. ($m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 3$)

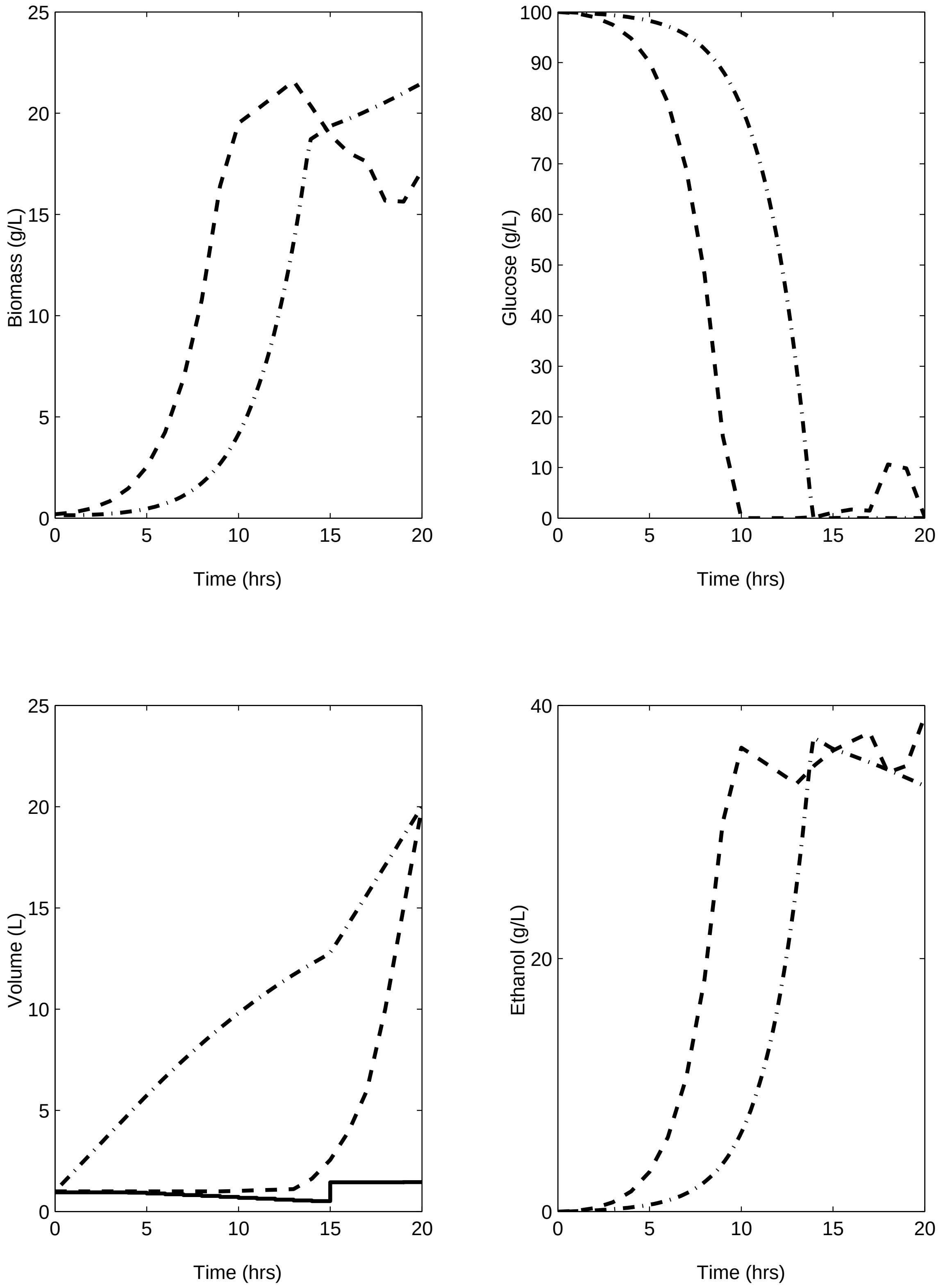


Figure 3.11: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($---$), and the closed-loop input profile(—) under nominal conditions for prediction horizon. ($m = 1,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 3$)

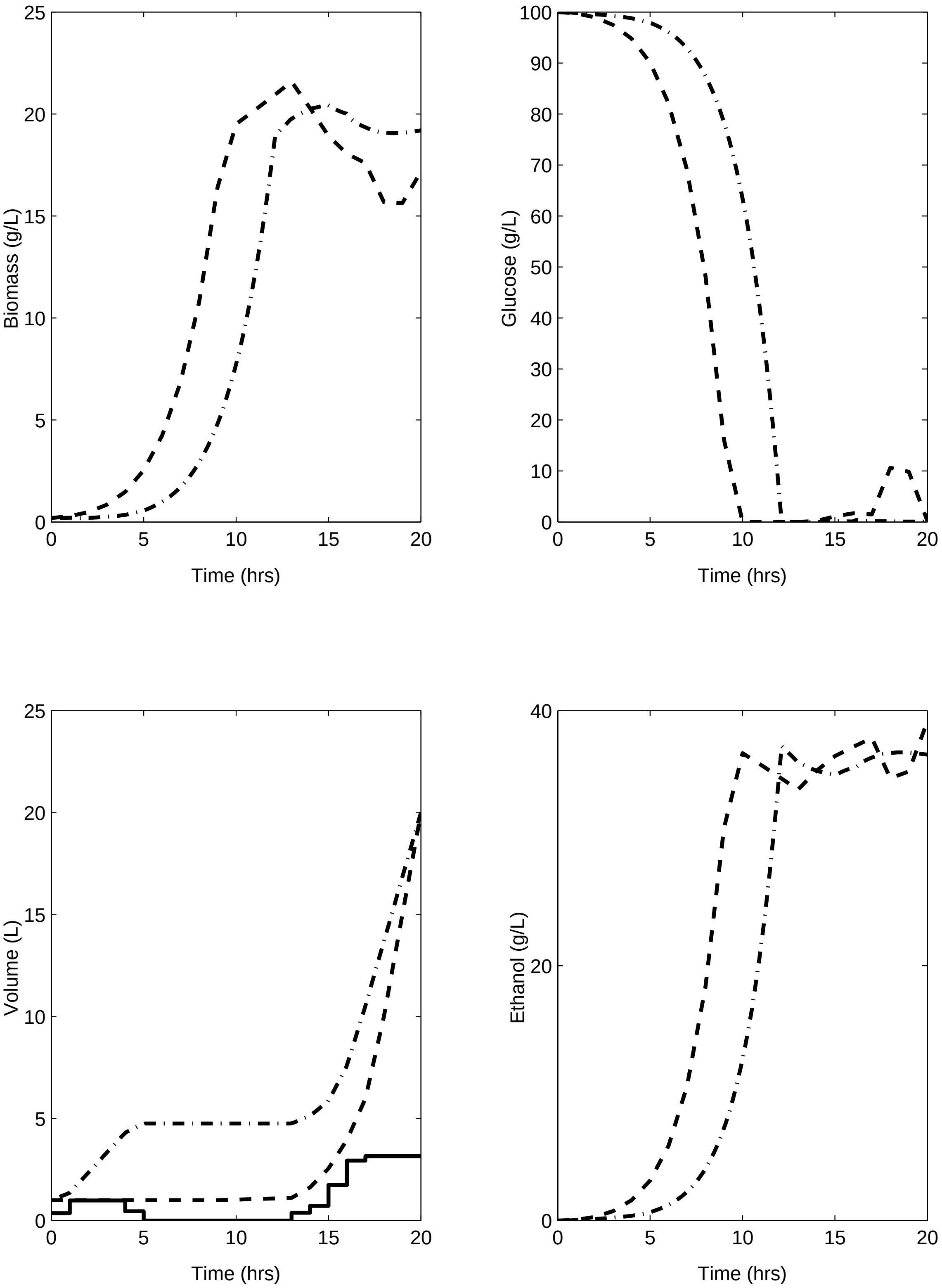


Figure 3.12: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($- - -$), and the closed-loop input profile(—) under nominal conditions using prediction horizon. ($m = 5,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 3$)

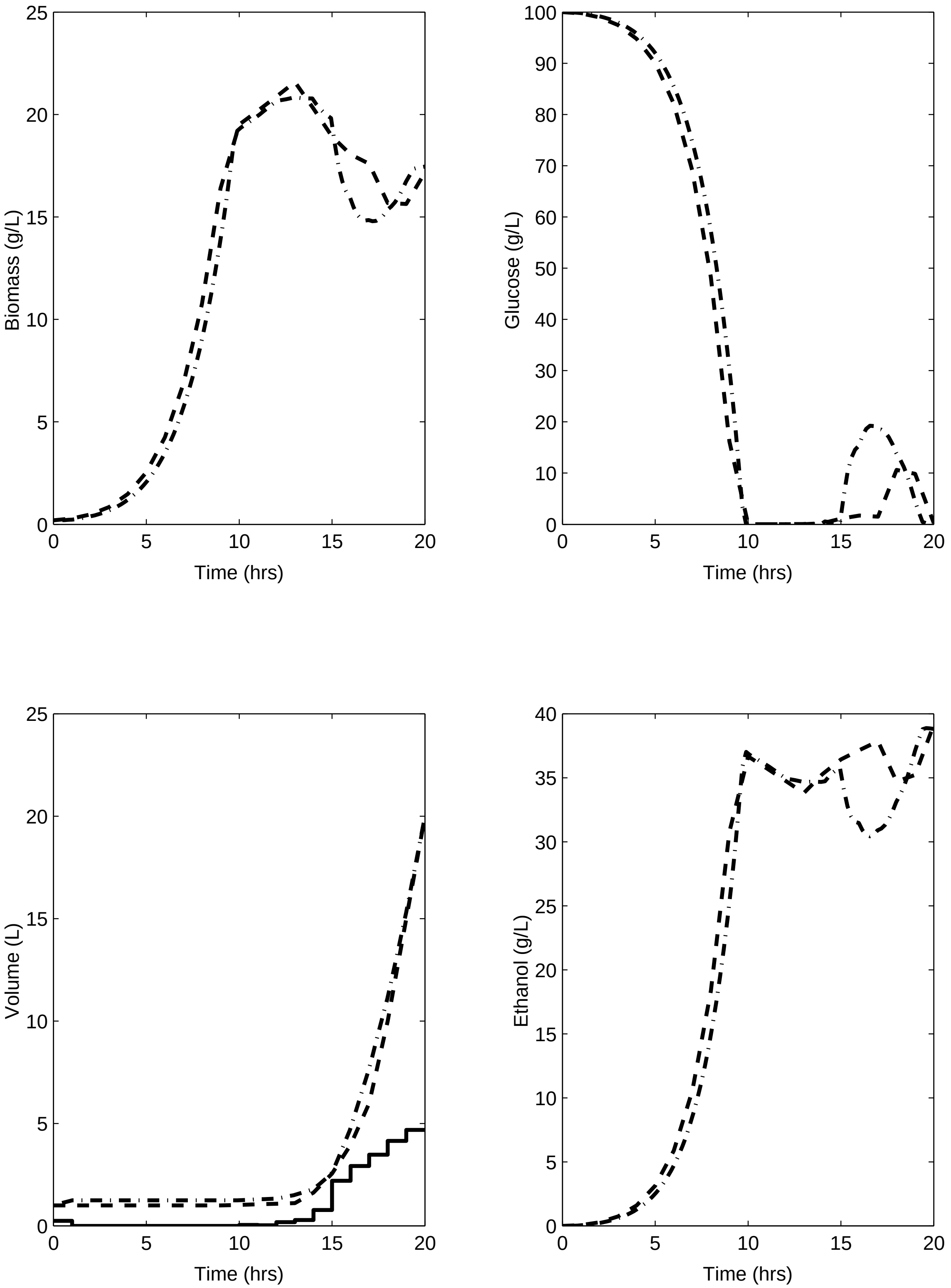


Figure 3.13: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($---$), and the closed-loop input profile(—) under nominal conditions using prediction horizon.($m = 9,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 3$)

Due to the feed disturbance, the original reference trajectory is no longer optimal, and this results in the decreased ethanol concentration at the end of the batch. In order to correct this, re-optimization can be carried out, to generate a new reference trajectory, thereby taking into account the increased amount of glucose within the feed. Generally, disturbances occuring in the plant are not known. With the knowledge of the initial pulse of the added feed at the start of the batch, and the system measurement at a certain point during the course of the batch, the initial feed concentration that causes the system to diverge from the reference trajectory can be calculated. Since this back-calculation of the feed concentration involves only the numerical integration of the system model, the new (disturbance) feed concentration can be readily obtained. In order to generate the new reference trajectory, optimization is carried out for the remainder of the batch, keeping the initial trajectory the same as the one calculated up to the measurement time point in the batch, where deviation of the system from the reference trajectory is detected. For the +10% disturbance considered in this work, the system information until 4 hrs was utilized. With the knowledge of the volume of feed added until 4 hrs, the glucose concentration in the feed (which in this case is 110 g/L) was back-calculated. Based on this increased feed glucose concentration, the optimal control policy was recalculated for the remainder of the batch *i.e.* from 4 hrs to 20 hrs. This new optimal input was used to obtain the modified ethanol reference trajectory which showed that the new final ethanol concentration was 42.19 g/L. For the SHMPC controller, the new ethanol reference trajectory was utilized for time 4 hrs onward. The results of this apprach are shown in Figure 3.15. The controller tracks the new reference trajectory and returns a final ethanol concentration of (40.17 g/L) which is within 5% of the re-optimized value of 42.19 g/L. Further, this closed-loop value is significantly (36%) better than the value of 25.6 g/L obtained with no re-optimization. It also out-performs the open-loop optimal value of 39.17 g/L and the closed-loop value of 38.91 g/L for the nominal case with feed glucose concentration of 100 g/L.

Limitations of the current approach: It was observed that the system performance was highly sensitive to controller tuning *i.e.* selection of the weights Γ_y, Γ_u, and Γ_f. In the current case,

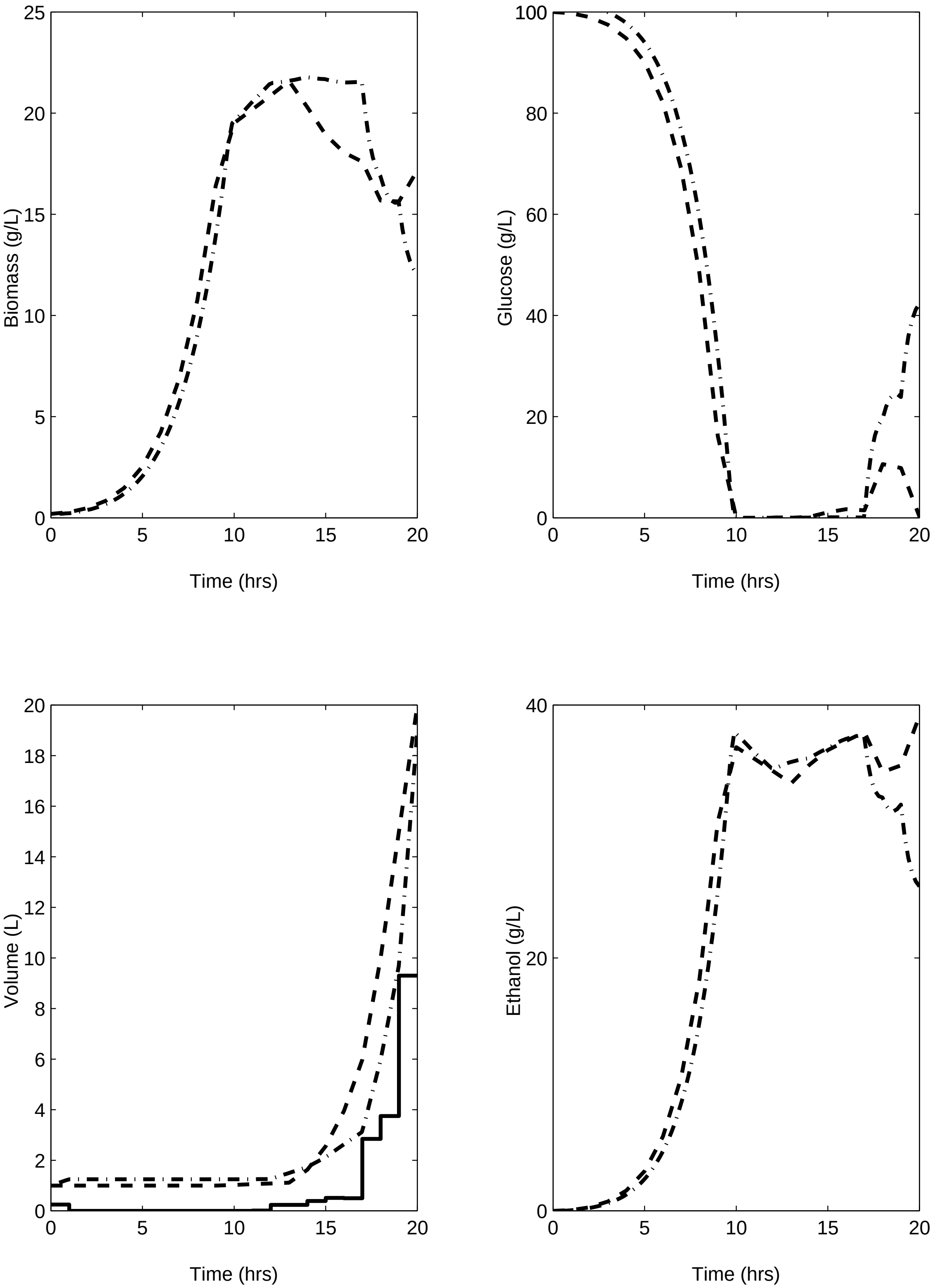


Figure 3.14: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($- - -$), and the closed-loop input profile(—) for a +10 % change in feed concentration. ($S_i = 110\ g/L,\ m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 3$)

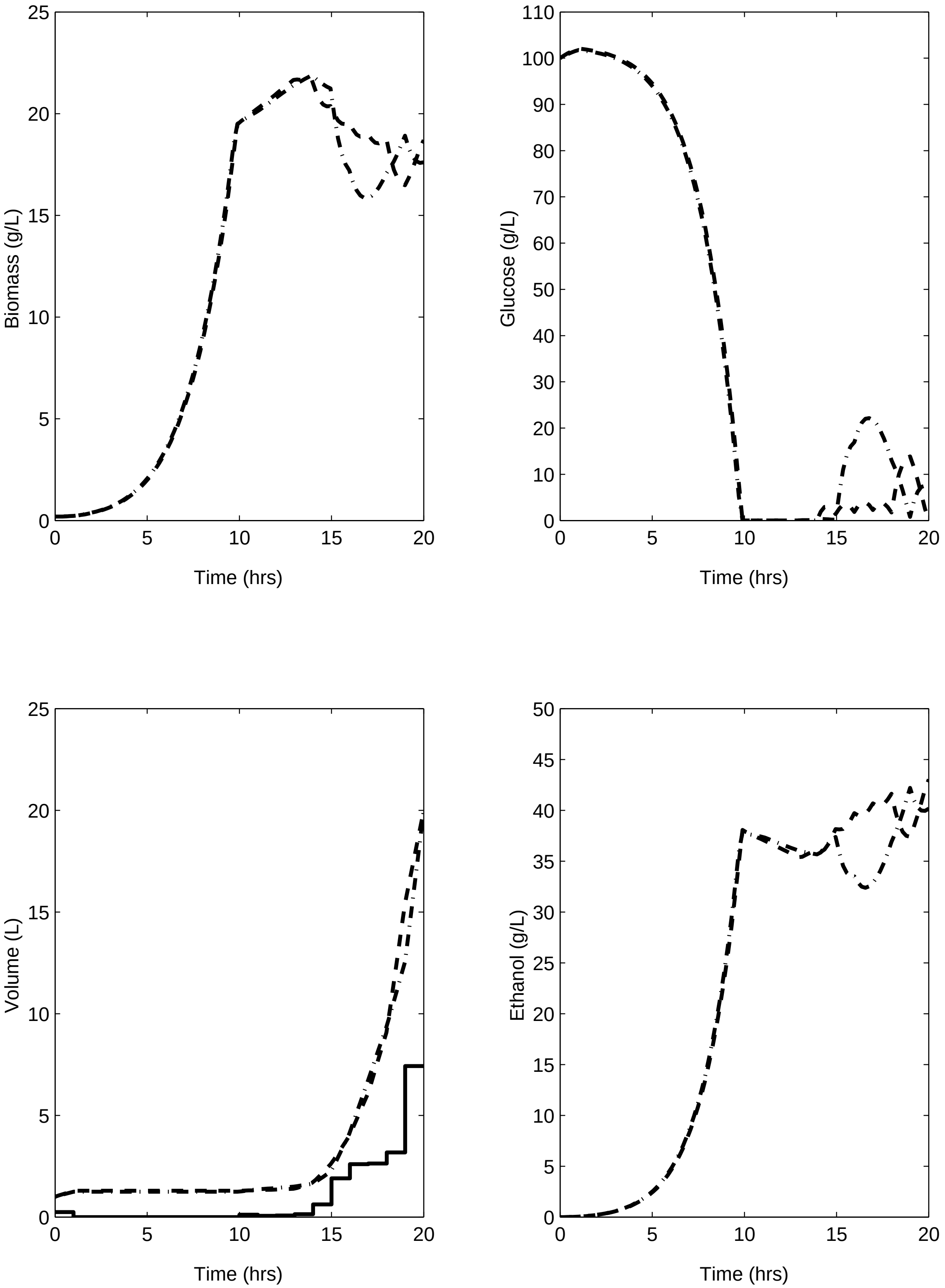


Figure 3.15: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($- - -$), and the closed-loop input profile(—) for a +10 % change in feed concentration and re-optimization of reference trajectory. ($S_i = 110\ g/L,\ m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 3$)

the controller tuned for the nominal case returned a final ethanol concentration of 40.17 g/L which is good but is slightly off when compared to 42.19 g/L which is the value from the re-optimized trajectory. However, with controller re-tuning, the closed-loop ethanol concentration with re-optimization increased to 41.82 g/L. This led to the conclusion that the controller tuning is disturbance specific, *i.e.* one needs to know *a priori* the nature of the disturbance before tuning the controller. However, accurate information about expected disturbances is not commonly available in an industrial setting.

Due to the final volume constraint of 20 L and the presence of the feed concentration disturbance, a large amount of feed is added to the system at the end of the batch when the sampling time is as large as 1 hour. Thus it was hypothesized that if the sampling time was reduced, the controller could eliminate large feed additions at the end of the batch, and this could lead to a more efficient utilization of available glucose. With this objective, the SHMPC scheme was now applied with a sampling time of 0.5 hours. Figure 3.16 shows the nominal controller performance. The weights used were, $\Gamma_y = 3$, $\Gamma_u = 10$, and $\Gamma_f = 2$. The tracking performance shows improvement, and the amount of ethanol obtained at the end of the batch is 38.98 g/L, which is very close to the open-loop optimal value of 39.17 g/L. This controller also shows excellent disturbance rejection without the need for re-optimization as shown in Figure 3.17. The final volume constraint of 20 L is satisfied, and ethanol obtained is 42.5 g/L, which is reasonably close to the open-loop optimal value of 43.17 g/L when the feed concentration is 110 g/L. Thus, reducing the sample time considerably improves tracking and disturbance compensation characteristics of the controller.

This performance improvement was attributed to the improved output trajectory prediction due to the reduced sampling time. In Figure 3.18, the actual process behavior with input profile for the first 10 hours of operation is compared with the predicted profiles from the two different sampling times evaluated at time $t = 10\ hr$, which was selected because it is the time when feed is first introduced into the reactor and the operation changes from the batch to the fed-batch mode. Another comparison at 15 hours is also shown in Figure 3.19. The predicted output behavior with 0.5 hr sampling time matches the actual ethanol profile more closely than the predicted output behavior for the 1 *hr* sampling time at both 10 and 15 hours. Thus, better future projections for the ethanol profile lead to an improvement in the closed-loop controller performance.

In this section, the design and performance of a controller was evaluated for reference tracking and disturbance rejection in a closed-loop. It was found that the tuning parameters played a significant role in controller design, and the final volume constraint was not satisfied when deviations from the final volume were not penalized. Once the objective function was modified to account for final volume constraint violations, admirable nominal performance was achieved. However, in the presence of a +10% feed concentration disturbance, re-optimization of the reference trajectory needed to be carried out to attain the fed-batch target. Finally, it was observed that more rapid sampling considerably improved both the nominal and disturbance compensation performance of the controller. The reduced sampling time also eliminated the need for on-line re-optimization and de-sensitized the system performance from controller tuning effects.

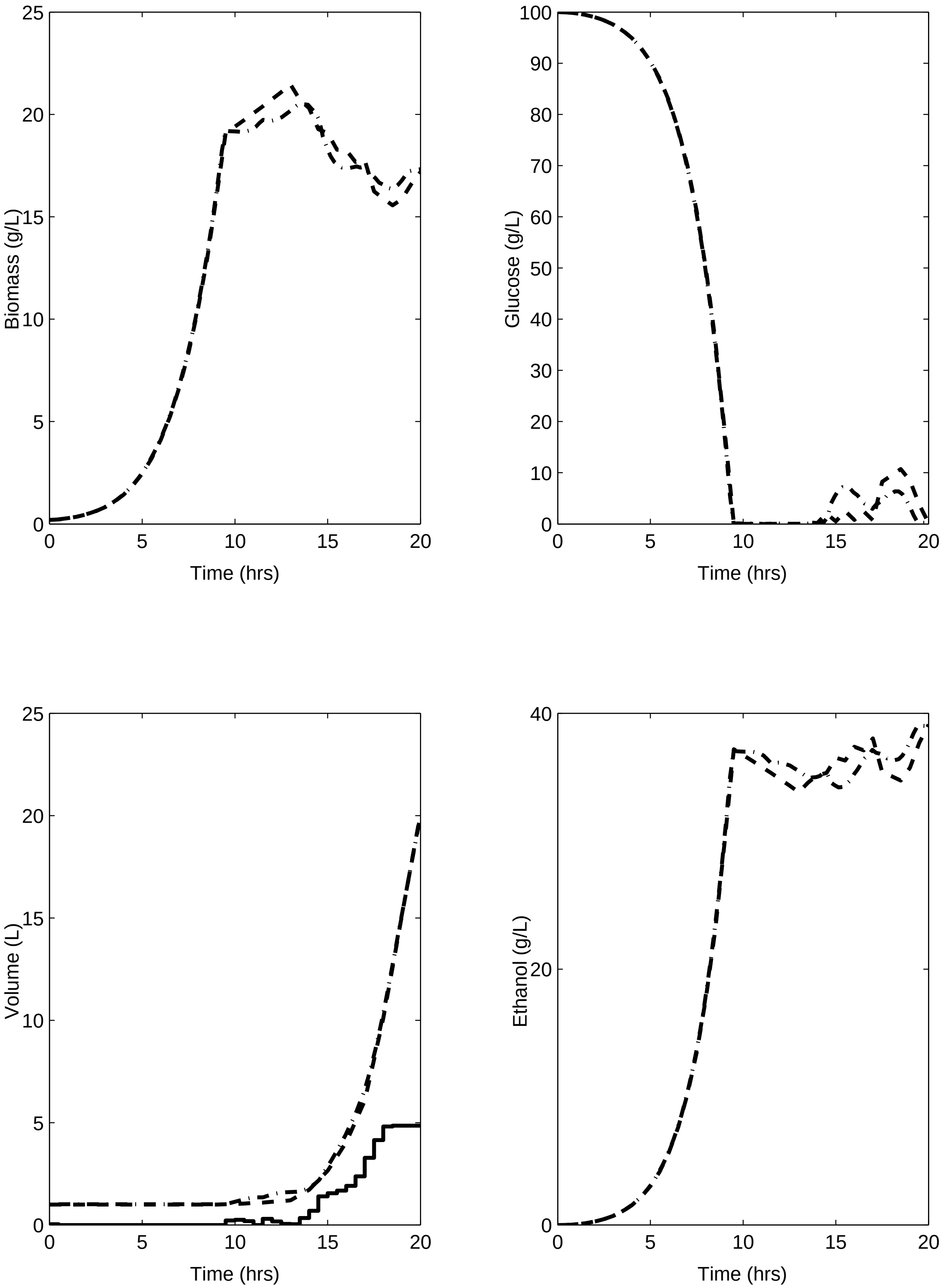


Figure 3.16: Plot of the state trajectories obtained using closed loop NLQDMC($- \cdot -$) compared with the open-loop, optimal trajectory ($---$), and the closed-loop input profile(—) under nominal conditions and a sampling time of 0.5 hours.
$(m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 2)$

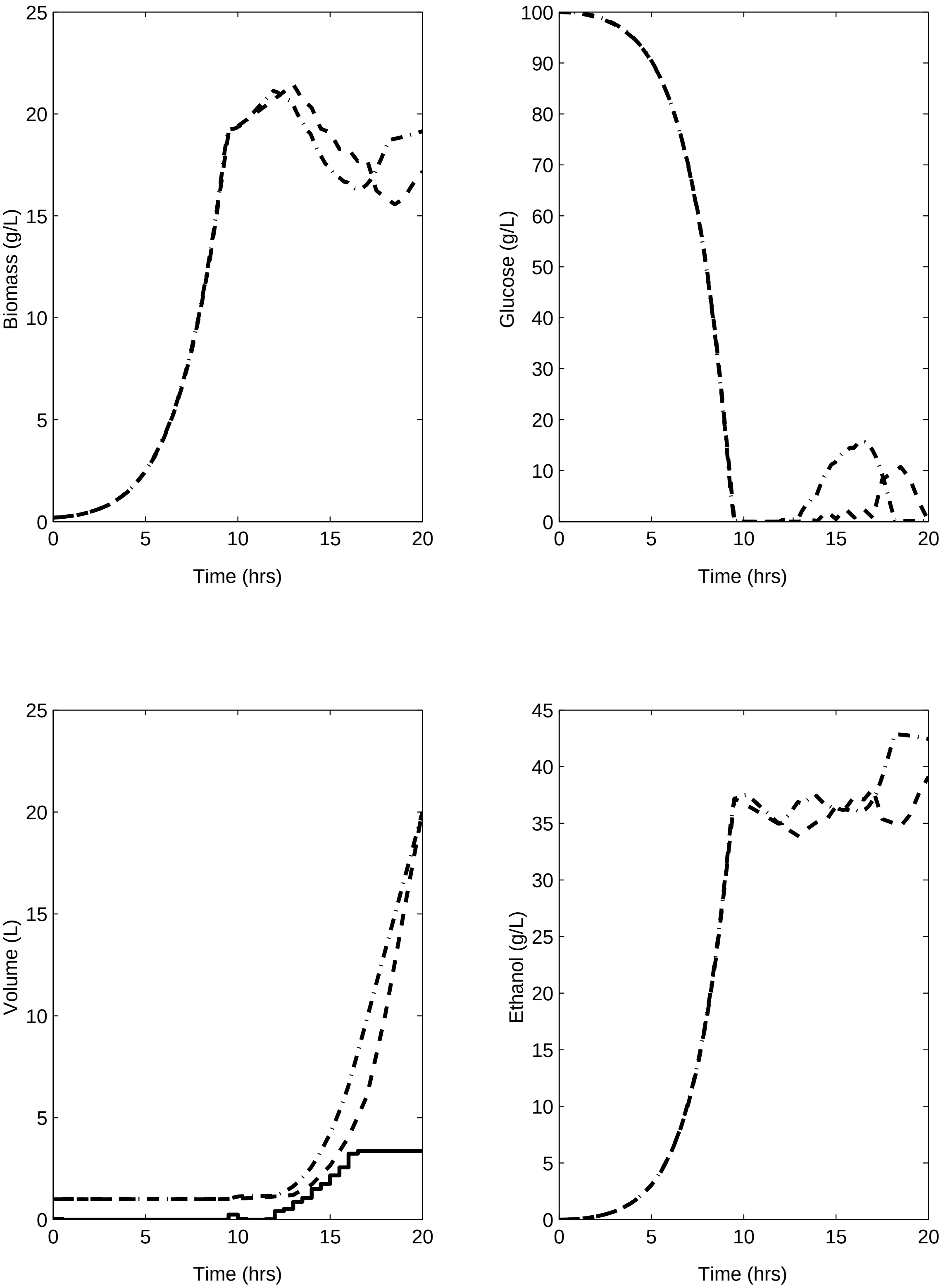


Figure 3.17: Plot of the state trajectories obtained using closed loop NLQDMC(− · −) compared with the open-loop, optimal trajectory (− − −), and the closed-loop input profile(—) for a +10 % change in feed cncentration and a sampling time of 0.5 hours. ($S_i = 110\ g/L,\ m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 2$)

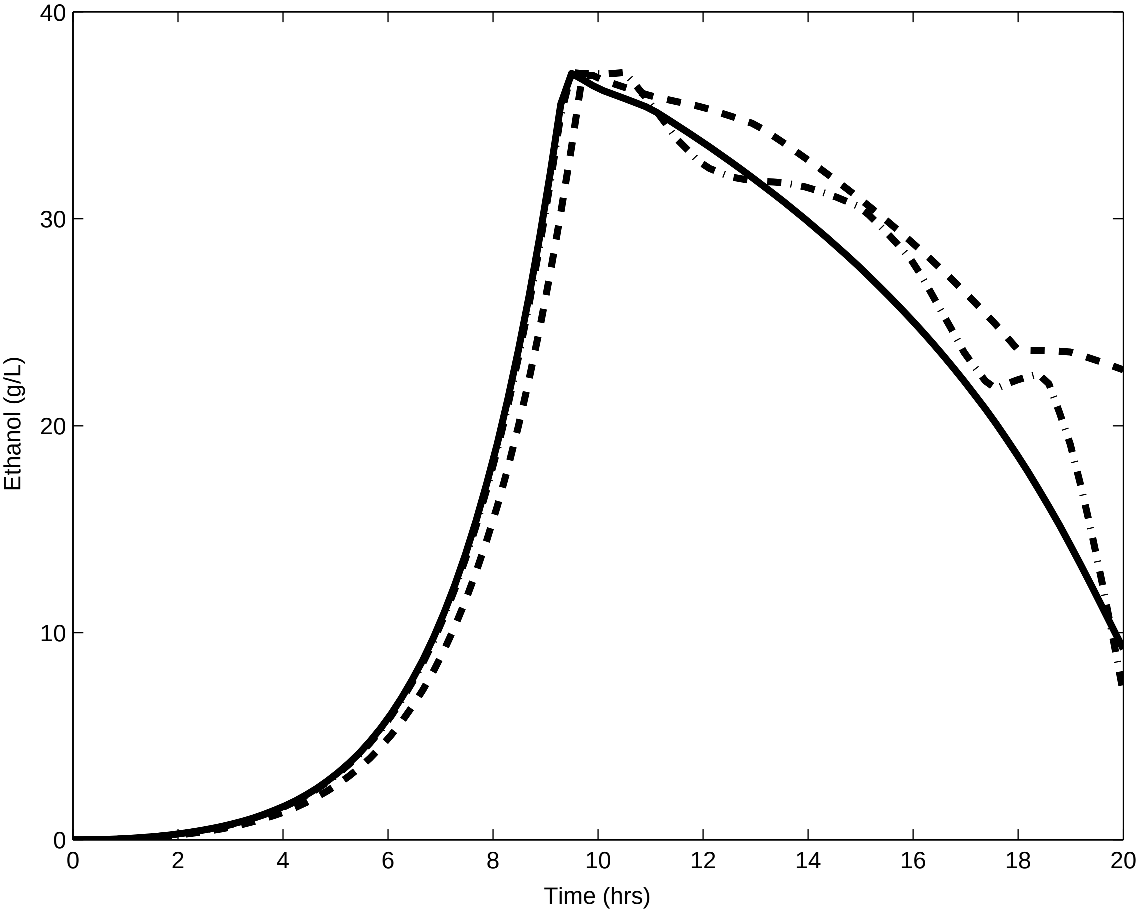


Figure 3.18: Plot of the ethanol trajectories obtained using closed loop NLQDMC for the first 10 hrs at sampling times of 1 hr ($---$), and 0.5 hrs ($-\cdot-$) compared with the optimal trajectory (—) for the first 10 hrs. ($S_i = 100\ g/L,\ m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 2$)

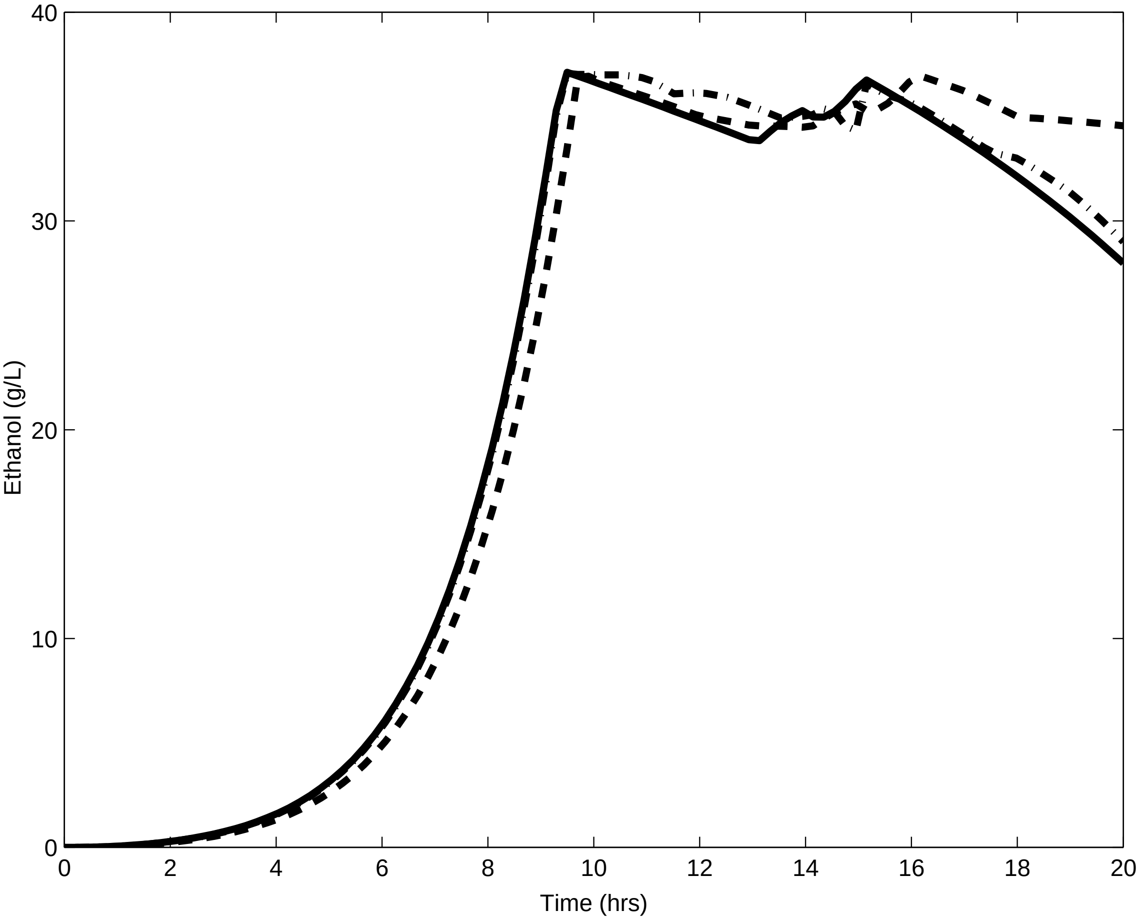


Figure 3.19: Plot of the ethanol trajectories obtained using closed loop NLQDMC for the first 15 hrs at sampling times of 1 hr ($---$), and 0.5 hrs ($-\cdot-$) compared with the optimal trajectory (—) for the first 15 hrs. ($S_i = 100\ g/L,\ m = p,\ \Gamma_y = 3,\ \Gamma_u = 10,\ \Gamma_f = 2$)

4.0 SUMMARY AND FUTURE WORK

4.1 Summary

In this work, the problem of fed-batch fermentation control has been addressed. The bioreactor operation is highly nonlinear at both a microscopic cellular scale and a macroscopic reactor scale; with reactions at these scales occuring at different times, the bioreactor system is multi-scale both temporally and spatially. A model structure that recognizes the multi-scale nature of the system, was utilized to describe the fermentation of *Saccharomyces cerevisiae* on glucose to produce ethanol as a product. It was shown that the model can capture the sequential utilization of glucose and ethanol as substrates, and it can represent the dynamics of the system in both batch and fed-batch mode of operation.

The fed-batch mode of operation is also complex due to its dynamic nature. The set-points change continually over the batch, so that there is no single steady state operating point about which the process can be controlled. The controller design philosophy recognizes this dynamic nature and both the open-loop and the closed-loop control strategies are considered in this work, thus presenting a complete treatment of the controller design for the fed-batch fermentation control problem. An open loop optimal control policy was first determined in order to maximize the ethanol concentration at the end of the batch, with the substrate feed rate as the manipulated variable. In the next step, a nominal controller based on the NLQDMC strategy was designed to track the reference trajectory determined from the open-loop control. The controller design also included explicit constraints on both the state and manipulated variables and performed an excellent job of tracking the reference trajectory, while attaining the end of batch ethanol concentration. The SHMPC framework was also used in the closed-loop controller design and its utility for the fed-batch fermentation control problem was demonstrated. The performance of the controller was also evaluated in the presence of a +10% disturbance in the feed concentration. The effect of the volume constraint on the final ethanol concentration is best demonstrated with the concentration disturbance. It was found that re-optimization to obtain a new reference trajectory significantly improved the disturbance rejection performance of the controller. As an alternative to the re-optimization strategy, the use of a reduced sampling time of 0.5 *hrs.* improved both the nominal and the disturbance rejection performance of the controller without controller retuning.

4.2 Recommendations for Future Work

There are a number of directions in which future research on this work can proceed. The first direction relates to the open-loop optimization problem that is carried out off-line to generate the reference trajectory for the objective of maximizing the end of batch ethanol concentration. At this point, this calculation takes about 3 *hrs* on an `AMD-Athlon 1-Ghz` processor. By discretizing both the state and the manipulated variables using the technique of orthogonal collocation on finite elements [79],[81], the time required for the off-line optimization could be reduced. As a result faster analysis of the effect of initial conditions and nature of the objective function, on the open-loop optimal profile could be carried out.

The other direction for future work relates to the simultaneous on-line re-optimization of the reference trajectory. In the closed-loop operation it was observed that disturbances can have a severe impact on system performance. The utility of the re-optimization approach was demonstrated at one time-point. The goal is to make this on-line re-optimization a part of the closed-loop control algorithm, so that the closed-loop reference trajectory could be recalculated in the face of disturbances. This could significantly improve the disturbance rejection characteristics of the controller. A faster calculation of the open-loop optimization profile and hence the reference trajectory could also have an important role on the success of the re-optimization problem.

Another important direction that needs to be explored is the estimation problem. Currently, the closed-loop controller design is based only on an ethanol measurement. With the help of an online estimator such as the Extended Kalman Filter (EKF) [82] a current estimate of the complete system state could be obtained, based on measurements as and when they become available. Design of such an estimator in parallel with the closed-loop controller design could then be carried out, and the results of using a full state for the closed-loop controller design could be evaluated.

Finally, it would be interesting to test the controller design experimentally on a lab-scale bioreactor, and evaluate the controller performance. Controller performance in the presence of model/plant mismatch is also an important problem that needs to be addressed.

APPENDIX A

APPENDIX A   BIOREACTOR MODEL SIMULATION CODE

MATLAB (©2002, The MathWorks) MEX-file

```
#define S_FUNCTION_NAME a
/*
 * Need to include simstruc.h for the definition of the SimStruct and
 * its associated macro definitions.
 */

#include ”simstruc.h”
#include ”stdio.h”
#include ”math.h”

/*
 * mdlInitializeSizes - initialize the sizes array
 *
 * The sizes array is used by SIMULINK to determine the S-function block's
 * characteristics (number of inputs, outputs, states, etc.).
 *
 * The direct feedthrough flag can be either 1=yes or 0=no. It should be
 * set to 1 if the input, ”u”, is used is the mdlOutput function. Setting this
 * to 0 is akin to making a promise that ”u” will not be used in the mdlOutput
 * function. If you break the promise, then unpredictable results will occur.
 */
static void mdlInitializeSizes(SimStruct *S)
{
    ssSetNumContStates(    S,  5);   /* number of continuous states          */
    ssSetNumDiscStates(    S,  0);  /* number of discrete states            */
    ssSetNumInputs(        S,  1);  /* number of inputs                     */
    ssSetNumOutputs(       S,  5);  /* number of outputs                    */
    ssSetDirectFeedThrough(S,  0);  /* direct feedthrough flag              */
    ssSetNumSampleTimes(   S,  1);  /* number of sample times               */
    ssSetNumSFcnParams(    S,  0);  /* number of input arguments            */
    ssSetNumRWork(         S,  0);  /* number of real work vector elements   */
    ssSetNumIWork(         S,  0);  /* number of integer work vector elements*/
    ssSetNumPWork(         S,  0);  /* number of pointer work vector elements*/
}

/*
 * mdlInitializeSampleTimes - initialize the sample times array
 *
 * This function is used to specify the sample time(s) for your S-function.
 * You must register the same number of sample times as specified in
 * ssSetNumSampleTimes. If you specify that you have no sample times, then
 * the S-function is assumed to have one inherited sample time.
 *
 * The sample times are specified period, offset pairs. The valid pairs are:
 *
 *   [CONTINUOUS_SAMPLE_TIME   0     ]  : Continuous sample time.
 *   [period                   offset]  : Discrete sample time where
 *                                        period > 0.0 and offset < period or
 *                                        equal to 0.0.
 *   [VARIABLE_SAMPLE_TIME     0     ]  : Variable step discrete sample time
 *                                        where mdlGetTimeOfNextVarHit is
 *                                        called to get the time of the next
 *                                        sample hit.
 *
 *  or you can specify that the sample time is inherited from the driving
 *  block in which case the S-function can have only one sample time:
 *    [INHERITED_SAMPLE_TIME     0     ]
 */
static void mdlInitializeSampleTimes(SimStruct *S)
{
   ssSetSampleTime(S, 0, CONTINUOUS_SAMPLE_TIME);
   ssSetOffsetTime(S, 0, 0.0);
}
/*
 * mdlInitializeConditions - initialize the states
 *
 * In this function, you should initialize the continuous and discrete
 * states for your S-function block.  The initial states are placed
 * in the x0 variable.  You can also perform any other initialization
 * activities that your S-function may require.
 */
static void mdlInitializeConditions(double *x0, SimStruct *S)

{
        x0[0] = 0.2;     /* Biomass concentration in g/L         */
        x0[1] = 100.0;   /* Substrate concentration in g/L       */
        x0[2] = 1.00;    /* Reactor Volume in L                  */
        x0[3] = 100.0;   /* Dissolved Oxygen Tension in %        */
        x0[4] = 0.0;     /* Ethanol concentration in g/L         */
}

/*
 * mdlOutputs - compute the outputs
 *
 * In this function, you compute the outputs of your S-function
 * block.  The outputs are placed in the y variable.
 */

static void mdlOutputs(double *y, double *x, double *u, SimStruct *S, int tid)
{
        y[0] = x[0];                    /* Biomass concentration in g/L         */
        y[1] = x[1];                    /* Substrate concentration in g/L       */
        y[2] = x[2];                    /* Reactor Volume in L                  */
        y[3] = x[3];                    /* Dissolved Oxygen Tension in %        */
        y[4] = x[4];                    /* Ethanol concentration in g/L         */
}

/*
 * mdlUpdate - perform action at major integration time step
 *
 * This function is called once for every major integration time step.
 * Discrete states are typically updated here, but this function is useful
 * for performing any tasks that should only take place once per integration
 * step.
 */

static void mdlUpdate(double *x, double *u, SimStruct *S, int tid)
{
}

/*
 * mdlDerivatives - compute the derivatives
 *
 * In this function, you compute the S-function block's derivatives.
 * The derivatives are placed in the dx variable.
 */

static void mdlDerivatives(double *dx, double *x, double *u, SimStruct *S, int tid)
{
        /*Constants used in the program */

        double qOcap = 0.3;
        double ki    = 10.00;
        double qSmax = 2.4;
        double Ks    = 0.12;
        double qm    = 0.01;
        double Ke    = 0.1;
        double Cx    = 0.0384;
        double Cs    = 0.0333;
        double Ce    = 0.0435;
        double Yxsox = 0.5;
        double Yxsf  = 0.15;
        double Yxe   = 0.72;
        double Yos   = 1.067;
        double Yoe   = 2.087;
        double Yes   = 0.51;
        double F0    = 0.08;
        double SFR   = 0.25;
        double Fmax  = 0.21;
        double Si    = 100.00;
        double KLa   = 600.00;
        double Ko    = 1.0;
        double tL    = 0.75;

        /* Definition of other variables */

        double qOmax;
        double F;
        double qS;
        double qSox;
        double qSoxan;
        double qSoxen;
        double qOs;
        double red;
        double qSfan;
        double qSfen;
        double qEp;
        double qEen;
        double qO;
        double My;
        double qo2;
        double qSf;
        double qEan;
        double qEc;
        double qcomp;

        int i;

        /* Current simulation time*/

        double tt = ssGetT(S);
        /* Calculation Algorithm          */

        qOmax   = qOcap/(1.00+x[4]/ki)*x[3]/(x[3]+Ko);
        qS      = qSmax*x[1]/(x[1]+Ks)*(1.00-exp(-tt/tL));
        qSox    = qS;
        qSoxan  = (qSox - qm)*Yxsox*Cx/Cs;
        qSoxen  = qSox - qSoxan;
        qOs     = qSoxen*Yos;
        red = 1.00;

        /*mexPrintf(”qOs is %lf : qOmax is %lf : ”,qOs,qOmax);*/

        if (qOs > qOmax)
            {
                red = qOmax/qOs;
                qOs = qOmax;
            }

        /*mexPrintf(”t = %lf: red is %lf :\n”,tt,red);*/

        qSoxen = red*qSoxen;
        qSoxan = red*qSoxan;
        qSox    = qSoxan + qSoxen;
        qSf     = qS - qSox;
        qSfan   = qSf*Yxsf*Cx/Cs;
        qSfen   = qSf - qSfan;
        qEp     = qSfen*Yes;
        qEen    = (qOmax - qOs)/Yoe*x[4]/(x[4]+Ke);

        for(i=1;i<=35;i++)
        {
                qEan = qEc*Yxe*Cx/Ce*x[4]/(x[4]+Ke);
                qEc  = qEen + qEan;
        }

        qO      = qOs + qEen*Yoe;
        My      = (qSox - qm)*Yxsox + qSf*Yxsf + qEc*Yxe;

        /*F       = F0*(exp(SFR*tt));
        if (F > Fmax)
                F = Fmax;*/

        F = u[0];

        dx[0]   = x[0]*(-F/x[2] + My);
        dx[1]   = F/x[2]*(Si - x[1]) - qS*x[0];
        dx[2]   = F;
        dx[3]   = KLa*(100-x[3])-qO*x[0]*14000.00;
        dx[4]   = x[0]*(qEp - qEc) - F/x[2]*x[4];
        qo2     = qO/32*1000.00;

}

#if 0  /* Change to a 1 if VARIABLE_SAMPLE_TIME has been specified */
/*
 * mdlGetTimeOfNextVarHit - Get the time of the next variable sample time hit
 *
 * This function is called one for every major integration time step.
 * It must return time of next hit by using ssSetTNext. The time of the
 * next hit must be greater than ssGetT(S).
 */
#define MDL_GET_TIME_OF_NEXT_VAR_HIT
static void mdlGetTimeOfNextVarHit(SimStruct *S)
{
    /* ssSetTNext(S, <timeOfNextHit>); */
}
#endif

/*
 * mdlTerminate - called when the simulation is terminated.
 *
 * In this function, you should perform any actions that are necessary
 * at the termination of a simulation.  For example, if memory was allocated
 * in mdlInitializeConditions, this is the place to free it.
 */

static void mdlTerminate(SimStruct *S)
{
}

#ifdef  MATLAB_MEX_FILE    /* Is this file being compiled as a MEX-file? */
#include "simulink.c"      /* MEX-file interface mechanism */
#else
#include "cg_sfun.h"       /* Code generation registration function */
#endif
```

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Massachussets Institute of Technology - Summer Sessions . Downstream processing. URL: http://web.mit.edu/professional/summer/courses/biology/20.45s.html, 1999.

[2] M. Morari and E. Zafiriou. *Robust Process Control*. Prentice-Hall, Englewood Cliffs, NJ, 1989.

[3] H. T. B. Pham, G. Larsson, and S-O. Enfors. Growth and energy metabolism in aerobic fed-batch cultures of *Saccharomyces cerevisiae*: Simulation and model verification. *Biotech. Bioeng.*, 60:474–482, 1998.

[4] B. Joseph and F. W. Hanratty. Predictive control of quality in a batch manufacturing process using artificial neural networks. *Ind. Eng. Chem. Res.*, pages 1951–1961, 1993.

[5] C. E. García. Quadratic dynamic matrix control of nonlinear processes. An application to a batch reaction process. In *Proc. AIChE Annual Meeting*, San Francisco, CA, 1984.

[6] R. Berber. Control of batch reactors : A review. *Trans. Inst. Chem. Eng.*, 74(Part A):3–20, 1996.

[7] J. E. Bailey. Toward a science of metabolic engineering. *Science*, 252:1668–1675, 1991.

[8] G. Stephanopoulos and J. J. Vallino. Network rigidity and metabolic engineering in metabolic overproduction. *Science*, 252:1675–1681, 1991.

[9] Zofia Verwater-Lukszo. A practical approach to recipe improvement and optimization in the batch processing industry. *Comput. Ind.*, 36:279–300, 1998.

[10] H. W. Blanch and D. S. Clark. *Biochemical Engineering*. Marcel Dekker Inc., 1996.

[11] J. E. Bailey and D. F. Ollis. *Biochemical Engineering Fundamentals*. McGraw–Hill, New York, Second edition, 1986.

[12] Karl-Heinz Bellgardt. *Bioreaction Engineering Modeling and Control*, chapter 2: Bioprocess Models, pages 44–105. Springer, July 2000.

[13] M. L. Shuler and F. Kargi. *Bioprocess Engineering Basic Concepts*. Prentice Hall, 1992.

[14] M. H. Bassett, P. Dave, F. J. Doyle III, G. K. Kudva, J. F. Pekny, and G. V. Reklaitis. Perspecties on model based integration of process operations. *Comput. Chem. Eng.*, 20(6):821–844, 1996.

[15] W. Marquardt. Towards a process modeling methodology. In *Methods of Model Based Process Control*, pages 3–40. Kluwer Academic Publishers, 1995.

[16] J. W. Jeong, J. Snay, and M. M. Ataai. A mathematical model for examining growth and sporulation processes of *Bacillus subtilis*. *Biotech. Bioeng.*, 35:160–184, 1990.

[17] M. M. Domach, S. K. Leung, R. E. Cahn, G. G. Cocks, and M. L. Shuler. Computer model for glucose-limited growth of a single cell of *Escherichia coli B/r-A*. *Biotechnology and Bioengineering*, 26:203–216, 1984.

[18] D. E. Steinmeyer and M. L. Shuler. Structured model for *Saccharomyces Cerevisiae*. *Chem. Eng. Sci.*, 44:2017–2030, 1989.

[19] P. A. Wisnewski, F. J. Doyle III, and F. Kayihan. Fundamental continuous pulp digester model for simulation and control. *AIChE J.*, 43:3175–3192, 1997.

[20] A. Aoyama, F.J. Doyle III, and V. Venkatasubramanian. Control-affine fuzzy neural network approach for nonlinear process control. *J. Proc. Cont.*, 5:375–386, 1995.

[21] P. Agrawal, G. Koshy, and M. Ramseier. An algorithm for operating a fed-batch fermenter at optimum specfic-growth rate. *Biotech. Bioeng.*, 33:115–125, 1989.

[22] R. S. Parker and F. J. Doyle III. Optimal control of a continuous bioreactor using an empirical non-linear model. *Ind. Eng. Chem. Res.*, 40:1939–1951, 2001.

[23] F. J. Doyle, B. A. Ogunnaike, and R. K. Pearson. Nonlinear model-based control using second order Volterra models. *Automatica*, (31):697–714, 1995.

[24] B. Maner and F.J. Doyle III. Simulated polymerization control using autoregressive plus Volterra–based MPC. *AIChE J.*, 43:1763–1784, 1997.

[25] L. Ljung. *System Identification: Theory for the User*. P T R Prentice Hall, Englewood Cliffs, NJ, 1987.

[26] R. S. Parker. *Model-Based Analysis and Control for Biosystems*. PhD thesis, Department of Chemical Engineering, University of Delaware, 1999.

[27] A. Tholudur and W. F. Ramirez. Optimization of fed-batch bioreactors using neural network parameter function models. *Biotech. Prog.*, 12:302–309, 1996.

[28] S. F. de Azevedo, B. Dahm, and F. R. Oliveira. Hybrid modelling of biochemical processes: A comparison with the conventional approach. *Comput. Chem. Eng.*, (Suppl.):S751–S756, 1997.

[29] D. Psichogios and L. H. Ungar. A hybrid neural network–first principles approach to process modeling. *AIChE J.*, 38:1499–1511, 1992.

[30] M. A. Henson. Nonlinear model predictive control: Current status and future directions. *Comput. Chem. Eng.*, 23:187–202, 1998.

[31] H. M. Tsuchiya, A. G. Fredrickson, and R. Aris. Dynamics of microbial cell populations. *Adv. Chem. Eng.*, 6:125, 1966.

[32] A. G. Fredrickson, R. D. Megee III, and H. M. Tsuchiya. Mathematical models for fermentation processes. *Adv. Appl. Microbiol.*, 13:419, 1970.

[33] J. Monod. The growth of bacterial cultures. *Annu. Rev. Microbiol.*, 3:371, 1949.

[34] R. D. Yang and A. E. Humphrey. Dynamic and steady state studies of phenol biodegradation in pure and mixed cultures. *Biotech. Bioeng.*, 17:1211, 1975.

[35] F. M. Williams. A model of cell growth dynamics. *J. Theor. Biol.*, 15:190, 1967.

[36] S. W. Peretti and J. E. Bailey. Mechanistically detailed model of cellular metabolism for glucose-limited growth of *Escherichia coli B/r-A*. *Biotech. Bioeng.*, 28:1672–1689, 1985.

[37] N. V. Mantzaris, F. Srienc, and P. Daoutidis. Nonlinear productivity control using a multi-staged cell population balance model. *Chem. Eng. Sci.*, 57:1–14, 2002.

[38] A. M. Zamamiri, Y. Zhang, M. A. Henson, and M. A. Hjortsø. Dynamics analysis of an age distribution model of oscillating yeast cultures. *Chem. Eng. Sci.*, 57:2169–2181, 2002.

[39] H. Y. Wang, C. L. Cooney, and D. I. C. Wang. Computer aided baker's yeast fermentations. *Biotech. Bioeng.*, 19:69–86, 1977.

[40] J. V. Straight and D. Ramkrishna. Cybernetic modeling and regulation of metabolic pathways. growth on complementary nutrients. *Biotech. Prog.*, 10:574–587, 1994.

[41] J. Varner and D. Ramkrishna. Metabolic engineering from a cybernetic perspective-I. theoretical preliminaries. *Biotech. Progr.*, 15:407–425, 1999.

[42] D.S. Kompala. *Bacterial Growth on Multiple Substrates. Experimental Verification of Cybernetic Models.* PhD thesis, Purdue University, 1984.

[43] P. S. Dhurjati and R. J. Leipold. *Computer Control of Fermentation Processes*, chapter 8: Biological Modeling, pages 207–221. CRC Press, Boca Raton, Florida, 1990.

[44] D. S. Kompala, D. Ramkrishna, N. B. Jansen, and G. T. Tsao. Investigation of bacterial growth on mixed substrates: Experimental evaluation of cybernetic models. *Biotech. Bioeng.*, 28:1044–1055, 1986.

[45] D. S. Kompala. Cybernetic modeling of spontaneous oscillations in continuous cultures of *Saccharomyces cerevisiae*. *J. Biotech.*, 71:267–274, 1999.

[46] S. Skogestad and M. Morari. Understanding the dynamic behavior of distillation columns. *Ind. Eng. Chem. Res*, 27(10):1848–1862, 1988.

[47] L. S. Balasubramhanya and F. J. Doyle III. Nonlinear control of a high-purity distillation column using a traveling wave model. *AIChE J.*, 43(3):703–714, 1997.

[48] Y. K. Rani and R. V. S. Rao. Control of fermenters- a review. *Bioproc. Eng.*, 21:77–88, 1999.

[49] A. Johnson. The control of fed–batch fermentation processes – A survey. *Automatica*, 23:691–75, 1987.

[50] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. *The Mathematical Theory of Optimal Processes.* Interscience Publishing, 1962.

[51] A. E. Bryson and Y. C. Ho. *Applied Optimal Control.* Hemisphere Publishing Corporation, New York, 1975.

[52] E. Balsa-Canto, J. R. Banga, A. A. Alonso, and V. S. Vassiliadis. Efficient optimal control of bioprocess using second-order information. *Ind. Eng. Chem. Res.*, 39:4287–4295, 2000.

[53] R. Luus and D. Hennessy. Optimization of fed-batch reactors by the Luus-Jaakola optimization procedure. *Ind. Eng. Chem. Res.*, 38:1948–1955, 1998.

[54] R. Luus. Application of dynamic programming to differential algebraic process systems. *Comp. Chem. Eng.*, 17(4):373–377, 1993.

[55] R. Luus and O. Rosen. Application of dynamic programming to final state constrained optimal cntrol. *Ind. Eng. Chem. Res.*, 30(7):1525–1530, 1991.

[56] S. Pushpavanam, S. Rao, and I. Khan. Optimization of a biochemical fed-batch reactor using sequential quadratic programming. *Ind. Eng. Chem. Res.*, 38:1998–2004, 1999.

[57] J. Hong. Optimal substrate feeding policy for a fed batch fermentation with substrate and product inhibition kinetics. *Biotech. Bioeng.*, 28:1421–1431, 1986.

[58] C-T. Chen and C. Hwang. Optimal on-off control for fed-batch fermentation processes. *Ind. Eng. Chem. Res.*, 29:1869–1875, 1990.

[59] J. M. Modak and H. C. Lim. Feedback optimization of fed-batch fermentation. *Biotech. Bioeng.*, 30:528–540, 1987.

[60] S. J. Parulekar and H. C. Lim. Modeling, optimization, and control of semi-batch bioreactors. *Adv. Biochem. Eng. Biotech.*, 32:207–258, 1985.

[61] S. Palanki, C. Kravaris, and H. Y. Wang. Synthesis of state feedback laws for end-point optimization in batch processes. *Chem. Eng. Sci.*, 48(1):135–152, 1993.

[62] D. B. Hodge and M. N. Karim. Modeling and advanced control of recombinant *Zymomonas mobilis* fed-batch fermentation. *Biotech. Progr.*, 18:572–579, 2002.

[63] F. L. Lewis and V. L. Syrmos. *Optimal Control.* Wiley Interscience, second edition edition, 1995.

[64] R. Martin and K. L. Teo. *Optimal Control of Drug Administration in Cancer Chemotherapy.* World Scientific, River Edge, NJ, 1994.

[65] P. L. Lee, R. B. Newell, and I. T. Cameron, editors. *Process Control and Management.* Chapman Hall, New York, first edition, 1998.

[66] T. Peterson, E. Hernández, Y. Arkun, and F. J. Schork. A nonlinear DMC algorithm and its application to a semi-batch polymerization reactor. *Chem. Eng. Sci.*, 47(4):737–753, 1992.

[67] C. E. Garcı́a, D. M. Prett, and M. Morari. Model predictive control: Theory and practice - A survey. *Automatica*, 25(3):335–348, 1989.

[68] N. L. Ricker. Model predictive control with state estimation. *Ind. Eng. Chem. Res.*, 29:374–382, 1990.

[69] N. L. Ricker. Model predictive control: State of the art. In Y. Arkun and W. H. Ray, editors, *Proceedings of the Fourth International Conference on Chemical Process Control*, pages 271–296, Padre Island, TX, 1991.

[70] M. Morari and J. H. Lee. Model predictive control: The good, the bad, and the ugly. In Y. Arkun and W. H. Ray, editors, *Proceedings of the Fourth International Conference on Chemical Process Control*, pages 419–444, Padre Island, TX, 1991.

[71] M. Morari and J. H. Lee. Model predictive control: Past, present and future. In *PSE ESCAPE-7 Symposium*, Trondheim, Norway, 1997.

[72] D. Bonvin, B. Srinivasan, and D. Ruppen. Dynamic optimization in the batch chemical industry. In *Proceedings of CPC VI*, AIChE Symposia Series, pages 283–307. CACHE Corporation, 2001.

[73] V. Liotta, C. Georgakis, and M. S. El-Aasser. Real-time estimation and control of particle size in semi-batch emulsion polymerization. In *Proceedings of the American Control Conference*, pages 1172–1176, Albuquerque, NM, 1997.

[74] M. M. Thomas, B. Joseph, and J. L. Kardos. Batch chemical process quality control applied to curing of composite materials. *AIChE J.*, 43(10):2535–2545, October 1997.

[75] B. A. Ogunnaike and W.H. Ray. *Process Dynamics, Modeling, and Control.* Oxford University Press, 1994.

[76] S. J. Qin and T. A. Badgwell. An overview of industrial model predictive control technology. In *Proceedings of the Fifth International Conference on Chemical Process Control*, Tahoe City, CA, 1996.

[77] J. B. Rawlings, E. S. Meadows, and K. R. Muske. Nonlinear model predictive control: A tutorial and survey. In *IFAC Symposium on Advanced Control of Chemical Processes*, pages 203–214, Kyoto, Japan, 1994.

[78] L. T. Biegler and J. B. Rawlings. Optimization approaches to nonlinear model predictive control. In Y. Arkun and W.H. Ray, editors, *Proceedings of the Fourth International Conference on Chemical Process Control*, pages 543–571, Padre Island, TX, 1991.

[79] J. W. Eaton and J. B. Rawlings. Feedback control of chemical processes using on-line optimization techniques. *Comput. Chem. Eng.*, 14(4/5):469–479, 1990.

[80] M. Morari, J.H. Lee, C.E. Garcíá, and D.M. Prett. *Model Predictive Control.* In Preparation, 2002.

[81] J. E. Cuthrell and L.T. Biegler. On the optimization of differential algebraic process systems. *AIChE J.*, 33(8):1257–12700, 1987.

[82] G. Gattu and E. Zafiriou. Nonlinear quadratic dynamic matrix control with state estimation. *Ind. Eng. Chem. Res.*, 31:1096–1104, 1992.