Fachhochschule Köln
Cologne University of Applied Sciences

OPTIMIZATION OF SUBSTRATE FEED IN A BIOGAS PLANT USING NON-LINEAR MODEL PREDICTIVE CONTROL

Master Thesis presented to

COLOGNE UNIVERSITY OF APPLIED SCIENCES

Campus Gummersbach

In the degree Program *Automation and IT*

Written by : *Ashwin Kumar Venkatesan*

First Examiner : *Prof. Dr. Michael Bongards*

Second Examiner : *Prof. Dr. Frithjof Klasen*

Gummersbach, December 2012

ACKNOWLEDGEMENTS

My profound gratitude for Daniel Gaida for his advice and his continual support throughout my thesis and for the knowledge I gained when working with him.

I thank Prof Dr. Michael Bongards for offering me the opportunity to pursue this thesis and providing all the necessary facilities.

I thank all my Professors in Cologne University of Applied sciences who helped me gain the necessary knowledge required to understand this thesis.

I thank my friends especially Esli, Naren and Prawyn for their continual support and encouragement throughout my thesis. I also thank the other members of the GECOC team for their support.

Finally, I thank all my family members who have been instrumental in supporting me through their concern in every stage of my work.

CONTENTS

ABSTRACT

Biogas is a very good source of clean energy. The potential of this becoming an important energy source is high and this makes the need for optimizing the plant very important. Since the Biogas production process is highly complex, nonlinear and slow, it is very difficult to determine the optimal substrate feed for the plant. Nonlinear Model Predictive Control has been shown to be a useful tool to control the biogas process. Already in (Luis Sousa Brito, 2011) this strategy has been implemented to the Biogas plant using a model developed for the Digester, called Anaerobic Digestion Model No. 1(ADM1). This thesis aims to further develop the control strategy to the biogas plant by decoupling the sampling time and the Control horizon. The experiments performed in this thesis reveal the possibility of the application of this strategy to the plant and the problems encountered in the implementation of this strategy to a complex process as a Biogas plant. It is also shown, that it is possible to implement this strategy to control and optimize a real biogas plant in the near future.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

TOE	- Tons of Oil Equivalents
CMAES	- Covariance Matrix Adaptation Evolutionary Strategy
NMPC	- Nonlinear Model Predictive Control
MPC	- Model Predictive Control.
GUI	- Graphical User Interface.
CHP	- Combined Heat and Power
DE	- Differential Evolution
GA	- Genetic Algorithm
GECO-C	- Gummersbacher Environmental Computing Center
NMPC	- Nonlinear Model Predictive Control
PSO	- Particle Swarm Optimization Toolbox
QIHNMPC	- Quasi-infinite Horizon Approach to NMPC

1. INTRODUCTION

Biogas plants are ideal friends of nature and man. In the modern day where technology and comfort are the leaders of man, to use energy without polluting nature is very difficult. The limited availability of fossil fuels and the damage caused by pollution has pushed mankind to search for alternatives sources of fuel which are environmental friendly.

Each year 590-880 million tons of methane is released into the atmosphere (Kossmann & Pönitz, 2011). 90% of this is through Microbial activity (Decomposition of Biomass) and the rest through burning of Fossils fuels. Methane CH4 is second largest contributor to global warming. Although it remains in the atmosphere for a short time, it traps 21 times more heat than Carbon Dioxide.

The sources of a Biogas plant are the agricultural and domestic waste. In developing countries such as India and China there is a tremendous production of biological waste and when waste left unused or left uncared for, becomes a problem for the society in terms of pollution, smell, space and hygiene. In these countries an efficient Biogas technology has an attractive future. In developed countries stringent environmental measures have made biogas technology an attractive option.

The few advantages of Biogas production are energy for cooking, heating and sometimes fuel for vehicles, efficient management of waste, conversion of waste into high quality fertilizer and reduction in the amount of methane.

Higher production demand and need for more efficient process has motivated the idea of application of Non linear model predictive control to the biogas plant. This thesis has been divided into 5 chapters.

Starting with the current chapter: The Introduction- which states the need for this thesis.

Chapter 2 explains about the basics of the biogas plants, principle and its working. Chapter 3 deals with the basic principle of MPC NMPC and the mathematical formulation of NMPC, its advantages and disadvantages.

Chapter 4 discusses the optimization methodology used in this thesis. It gives the basics of CMAES and its application on the Biogas plant. A detailed mathematical formulation is given with the emphasis on simplicity and easy understanding. However the explanation is not application specific. Chapter 5 describes the Biogas toolbox. Although for a detailed explanation the user is advised to refer Matlab Help.

Chapter 6 describes the application of NMPC strategy to a virtual biogas plant. This is done through the Matlab Biogas Toolbox.

Chapter 7 deals with the experimental results obtained by simulation. The relationship between the variables is explained in detail, with the special focus on the effect of the sampling time on population size, number of generations, control horizon, prediction horizon. The reliability of the implementation and the problems associated with are discussed.

To deal with the problems, two suggestions have been suggested. This thesis can be considered as a starting point for more future research.

2. BIOGAS PLANTS

2.1. PRINCIPLE

The principle of the Biogas plant is simple. The slurry or organic waste when treated with bacteria is decomposed and methane gas is produced. Thus the produced methane gas can be used for heat and electricity generation. Below the process in described in detail.

The Biogas production takes place in four distinct steps.

- Hydrolysis
- Acidogeneis
- Acetogenesis
- Methanogenesis

HYDROLYSIS:

In this stage anaerobic bacteria convert or breakdown high molecular matter like proteins, carbohydrates, fats and cellulose to simple substances like amino acids, monosaccharide, amino acids, fatty acids and water. Polymers are broken down to monomers. This makes the substrate more soluble in water.

ACIDOGENESIS:

The substrate mixture is treated with acid forming bacteria which consumes the oxygen in the digester during the conversion process. Thus this step provides an anaerobic environment for the methane producing bacteria. This process produces acids, alcohols, ketones and gases.

ACETOGENESIS:

The acid forming bacteria produce the initial products for methane formation from organic acids. During this step hydrogen is consumed by the bacteria and maintaining a stable temperature is very important. At the end of this process Acetic acid, carbon Dioxide and Hydrogen are formed.

METHANOGENESIS:

This is the last and vital step. 90% of Methane is formed during this process 70 % from acetic acid (Refer (Biogas Production Process, 2009)). Therefore acetogenesis is a vital step in order for this step to function properly. Acetic acid, carbon dioxide and Hydrogen are converted to methane, carbon dioxide and water.

Steps/Process	Process	Bacteria	Output
I	Hydrolysis	Anaerobic hydrolysis Bacteria	Monosaccharides, amino acids and fatty acids
II	Acetogenesis	Acid formers	Organic acids and CO_2
III	Acetogenesis	Acetic acid forming bacteria	Acetic acid, CO_2 and Hydrogen
IV	Methanogenesis	Methane producing bacteria	Methane, CO_2 and water

Table 2-1 - Tabulation of Biogas production process (Biogas Production Process, 2009)

There are some fundamental conditions that must be satisfied for the process to function properly - (Biogas Production Process, 2009),

- Anaerobic environment- Environment free of Oxygen.
- Humidity – Bacteria can feed and live only in moist conditions.
- Temperature – The optimum temperature for all bacteria is 25-40^{o} Celsius.

- Fermentation period – The biogas production is different at different stages of the process. Therefore this has to be monitored properly.

2.2. CLASSIFICATION

Biogas plants can be classified in different ways

- With reference to the way in which the substrates are being loaded a biogas plant can be classified into (refer (Флюид . Fluid, 2002)) but for a more detailed explanation refer (various-types-biogas-plants)
  - Continuous load types
  - Batch Biogas plants
- By the way in which the Biogas is collected (refer (Флюид . Fluid, 2002))
  - Fixed Dome Plants
  - Floating Dome Plants
  - Horizontal and Vertical biogas plants
  - Underground and above ground plants
  - Steel, concrete and masonry digesters

2.3. PARTS OF A BIOGAS PLANT

A Biogas plant configuration depends on the design and various other factors. Therefore to describe each part used by each type of plant would not be appropriate, instead the most common and important parts are described below. For a more thorough explanation refer to (Parts of a Biogas plant or Bio-gas Generator, 2012).

- Foundation
- Substrate Mixer
- Inlet Chamber
- Digester
- Dome

- Outlet Chamber
- Gas Outlet pipe and Valves
- CHP Cogeneration Units

FOUNDATION:

The foundation is the fundamental part of the plant which incorporates most of the plants' processes and is made of concrete and brick ballast. It is absolutely essential that there may be no seepage or leakage of external substances or temperature into the plant.

SUBSTRATE MIXER:

The Substrate mixer is a tank in which the all substrates and water are mixed. Water is an essential component in the degradation process. For example cow dung must be mixed with water in the ratio 1:1. A mixer is equipped with a propeller for mixing and chopping and in order to transport this mixture, a pump is also connected to the mixer. For some substrates a higher temperature might be a pre requisite, therefore a pre heater may be present.

INLET CHAMBER:

The inlet is a simple tank above the ground in which the substrates can be poured easily.

DIGESTER:

A Digester is a tank where the actual process of anaerobic digestion actually takes place. The tank is well insulated from outside influences. Biogas production is extremely sensitive to temperature. A sudden change or an increased temperature can damage the biogas production.

There are different types of digesters (Types of Biogas Digesters and Plants, 2012) based on the climate conditions and costs. There are many types of Biogas plants, the most important types are described below:

- Fixed-Dome plants
- Floating-drum plants
- Balloon plants

- Low cost Polyethylene Digester

DOME:

This is a gas chamber above the digester to collect the gas produced by the bacteria. Usually this is in a hemispherical shape and the more gas produced the more pressure on the substrate. From the dome the gas is the pumped to the Cogeneration units.

OUTLET CHAMBER:

The substrate after the anaerobic process is transferred or pumped into the storage tank or the outlet chamber. The substrate is later disposed later. This storage tank is usually bigger than the digesters and is usually above the ground for easy disposal and

GAS OUTLET PIPE AND VALVES:

The gas outlet pipe connects the dome and the point of application. That is the Dome and Cogeneration units. A valve controls the flow of biogas.

CHP CO-GENERATION UNITS:

CHP stands for Combined Heat and Power Generation unit. The biogas in the gas chamber is used to generate electricity and the heat from the generator is used for local heating. Since both heat and electricity are generated at the same time, it is called co generation unit. This process is preferred than a decentralized Electricity generation and heating, since this is more efficient.

2.4. ADVANTAGES AND DISADVANTAGES

The *advantages* are

- Production of clean energy.
- Conversion of organic waste into high quality fertilizer. (Kossmann & Pönitz, 2011).
- A method of waste management. Ejection of methane into the atmosphere from agricultural can be avoided Methane contributes 22 times more to greenhouse effect than CO_2.
- An additional income source.
- Independent functioning of localities, since they need not depend on the power transmitted through longer distances from Power Plants using Coal and Nuclear sources.

The *disadvantages* are

- The only disadvantage is the cost involved in installing a biogas plant. Efforts have to be taken to reduce the cost.

2.5. SUBSTRATE SOURCES

The common sources for Biogas production are shown below in Table 2 along with their ability to produce Biogas. Clearly from the table one can infer that Methane (CH_4), Carbon Dioxide (CO_2) and Water (H_2O) are three important products in Biogas.

However the presence of Hydrogen Sulphide creates a corrosive environment. Therefore there are certain restrictions in the direct application of Biogas to the household applications. But there are easy and safe measures which can be taken to prevent the risks.

In the table the methane content in the agri food industry is the highest, however the Hydrogen Sulphide content is also higher. Therefore a careful selection in the sources should be in order to increase Methane production and to keep H_2S production within a safe level.

The next table (Table 3) shows the world wide sources available for Biogas production and the amount which can be valued. Clearly Agricultural wastes occupy the top position.

COMPONENTS	SCALE	HOUSEHOLD WASTE	WASTEWATER TREATMENT PLANTS SLUDGE	AGRICULTURAL WASTES	WASTE OF AGRIFOOD INDUSTRY
CH_4	% vol	50-60	60-75	60-75	68
CO_2		38-34	33-19	33-19	26
N_2		5-0	1-0	1-0	-
O_2		1-0	< 0,5	< 0,5	-
H_2O		6 (à 40 ° C)	6 (à 40 ° C)	6 (à 40 ° C)	6 (à 40 ° C)
Total		100	100	100	100
H_2S	mg/m^3	100 - 900	1000 - 4000	3000 – 10 000	400
NH_3		-	-	50 - 100	-
Aromatic		0 - 200	-	-	-
Organochlorinated/ organofluorated		100-800	-	-	

Table 2-3- Various Components present inside various sources (Naskeo Environnement, 2009)

WORLD WIDE BIOGAS RESOURCES	PRODUCED BIOGAS (TOE/YEAR)	BIOGAS WHICH CAN BE VALUED (TOE/YEAR)
Urban and industrial solid waste	750	60 to 100
Urban and industrial waste water	50	40 to 50
Agricultural by-products	1000	40 to 150
TOTAL	1800	140 to 300
Biogas/worldwide consumption of natural gas	100%	8% to 17%

Table 2-4- Consumption of Biogas (Naskeo Environnement, 2009)

The Table 4 below shows the different types of agricultural sources used for Biogas production. A higher methane potential does not mean that the Fermenter is supplied with that particular substrate only; usually the availability of the substrate depends on the climatic and economic conditions. Therefore a situation often arises where multiple substrates have to be used in combination in order to obtain the maximum production and lower cost. This calls for careful consideration on the amount of the substrate being used. For this reason a Biogas plant operator operates the plant based on the experience.

Matter	Methane potential (m3 CH4/Ton of raw material)
Liquid bovine manure	20
Contents of paunch	30
Bovine manure	40
Potato pulps	50
Brewery waste	75
Shearing of lawn	125
Corn residues	150
Lubricate from slaughter-house	180
Molasses	230
Used grease	250
Cereal waste	300

Table 2-5- Methane Potential of Various substrates (Naskeo Environnement, 2009)

When Biogas is compared with the Non renewable resource such as Natural gas, the amount of methane it contains is much higher. But the availability of Natural gas is limited. The only factor which pushes Biogas production forward is the fact that it is clean, safe, and cheap.

TYPES OF GASES	BIOGAS 1 HOUSEHOLD WASTE	BIOGAS 2 AGRIFOOD INDUSTRY	NATURAL GAS
Composition	60% CH4	68% CH4	97,0% CH4
	33 % CO2	26 % CO2	2,2% C2
	1% N2	1% N2	0,3% C3
	0% O2	0% O2	0,1% C4+
	6% H2O	5 % H2O	0,4% N2

Table 2-6 – Types of Biogas (Naskeo Environnement, 2009)

2.6. NEED FOR CONTROL AND OPTIMIZATION

As discussed in the previous section (2.1) the Biogas production is a complex process. Temperature, pressure, Solid content, pH etc are some of the few factors which require careful monitoring. A control strategy has to be developed in order to monitor all these factors. Moreover the availability of a substrate and the cost of procurement constantly vary as per demand and climate, therefore in order to maintain higher Biogas Production - optimization of substrate feed plays a vital role.*One methodology* is to separate the control and optimization i.e. to use optimization methods to compute an optimal substrate feed to the plant and maintaining the constraints separately, as done in (Wolf, McLoone, & Bongards, 2009). However this is open loop control and this does not consider the disturbances occurring in the system. Although the input substrate is optimal, if a disturbance occurs then the system is not forced to the steady state.

Another methodology to enhance the previous methodology is the use of closed loop optimal control. An optimal input is computed based on the current state of the system, the constraints and the Input is applied to the system in accordance with Non linear Model Predictive control strategy which is explained in the next Chapter. The proven success of MPC in many industrial applications and the advantages of using Non linear MPC encourage a user to apply this methodology to a highly Non-linear system such as a Biogas Plant.

3. NON-LINEAR MODEL PREDICTIVE CONTROL

3.1. MPC PRINCIPLE:

Model Predictive control or Receding Horizon Control is the predecessor of Non Linear Model Predictive control. In order to understand NMPC one must understand MPC. The MPC tries to solve an open loop optimal control problem. There are 3 important variables to be understood.

- Control Horizon
- Prediction Horizon
- Sampling Time-δ

The controller tries to control a process by keeping the process constraints and limits satisfied. At every sampling instant data is collected by the controller. Then the controller, with the help of the current available data and a model of the plant, predicts an output over a time period called prediction horizon. Then for this predicted output, it predicts an input control sequence over the control horizon. This control sequence is applied to the system over a time period of Sampling time the sampling period.

The diagram shown below describes the working of MPC. Assume that the controller has to provide the plant with an input. The controller is provided with the plant constraints, objective function, optimization method and a model of the plant. The controller now receives data of the plant at time t=0. 'x' and 'u' are the previous state and input to the plant. Based on the constraints and objective function the controller predicts the required output over a period of prediction horizon indicated by T_p.

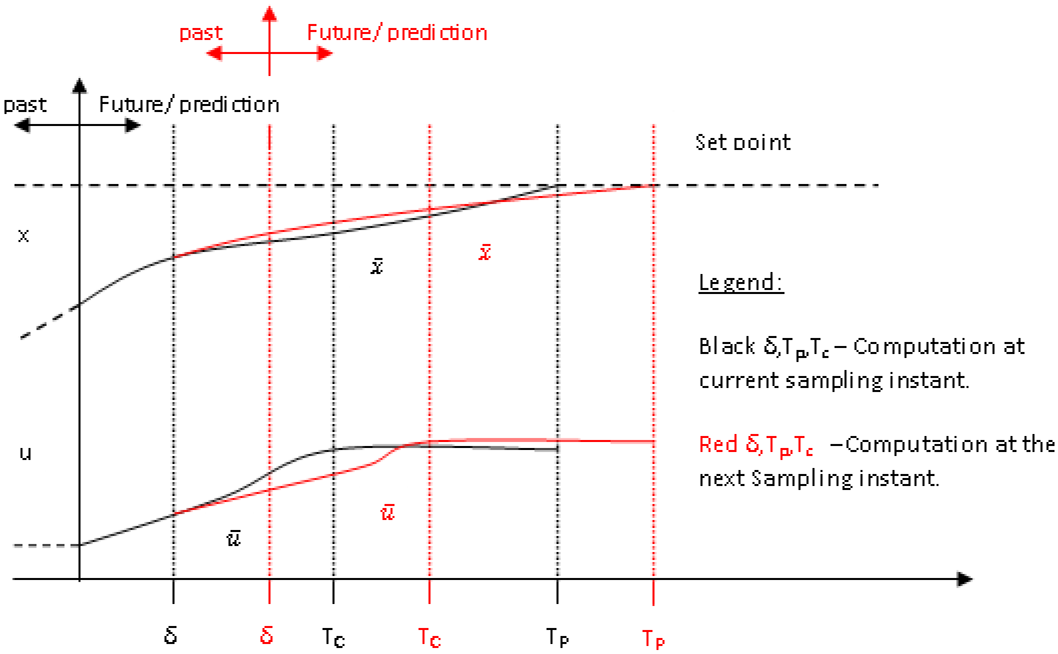


Figure 3-1 – Model Predictive Control Strategy

It then calls the optimization method to compute the corresponding input for the predicted output by using a linear model of the plant. $\bar{x}$ and $\bar{u}$ are the predicted states and inputs. However the input is computed over a period of Control horizon indicated by T_c. The input remains constant from T_c to T_p with the constant value equal to the input value at T_c. Therefore the input is predicted is such a way that in the period of 0-T_p the predicted output is followed by the plant.

Ideally one would like to have an infinite prediction Horizon so that the plant follows the desired trajectory. But this is not possible for two reasons. One the model used here is a linear model, therefore there is a definitely a model mismatch. Two the plant is functioning in the practical world, hence it is subjected to disturbances which creates a variation in the predicted output and the real output. Three the computation power required by the optimization method for an infinite horizon increases as the prediction horizon increases.

At the next sampling instant new measurements are taken and the computation is done again. This is indicated by the color red. Stability and robustness depend on the Sampling time, Prediction and Control Horizons.

The inclusion of constraints into the control setting is a great advantage when compared to the traditional control strategies. Also a linear model is used to approximate the response of the plant, which along with constraints gives us a region in space where the plant can be operated. Although MPC was designed for many applications, the extent to which a linear model can compensate for the non linearities is limited; therefore another method is required to compensate this defect.

NMPC strategy compensates for the non linearities in the plant. According to (Allgöwer, Findeisen, & Nagy, 2004) the main factors that encourage the application of NMPC are

- Most of the processes are inherently non linear
- Higher quality demands.
- Tighter environmental regulations.
- Increasing productivity demands.

3.2. NONLINEAR MODEL PREDICTIVE CONTROL:

In general all practical processes are inherently nonlinear. Linear MPC's use linear models to predict the system's output. This means that better control of a process depends on how accurate the model is and how flexible the controller can be, so as to accommodate the model non linearities. Even if the model is made highly complex, we can only go thus far to model the non linear dynamics of the system. Today higher product quality specifications and tougher regulations push a controller to operate the process on the boundary of stable/optimal regions. This type of control is possible with Nonlinear Models.

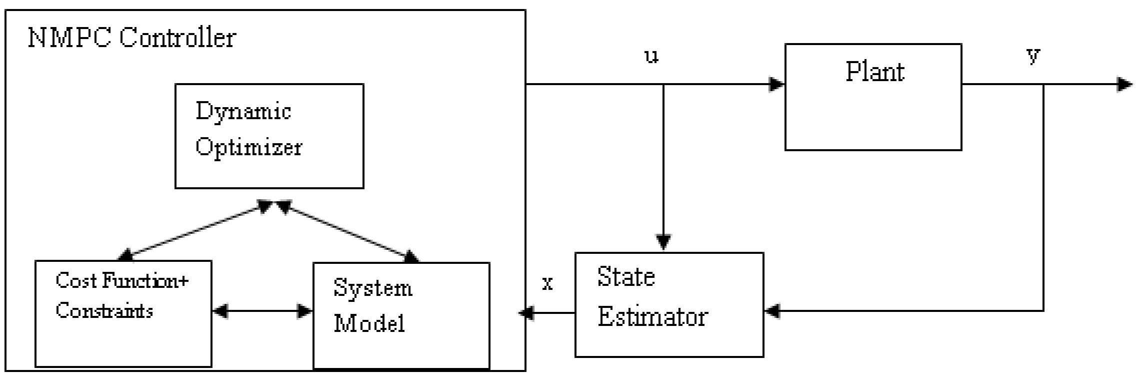


Figure 3-2 – Basic NMPC control loop (Allgöwer, Findeisen, & Nagy, 2004)

Shown above in figure 2 is the basic NMPC control loop. The dynamic optimizer uses the system model, the current state of the plant and the constraints to compute the cost function for each value of possible input. Based on the best cost function an input value is chosen (u) and applied to the plant.

The state estimator computes the current state of the plant from the output 'y'. The system model is a mathematical representation through differential equations. These differential equations contain the variables for the states computed by the state estimator and the output can be calculated. Once the desired output trajectory is computed the possible input values are computed using this model. Then the input values are taken by the optimization algorithm to compute the fitness value and the input with the best fitness value is chosen for the plant.

1. The state of the system or plant is measured.

2. An optimal input $(\bar{u}(t) \dots \bar{u}(t+T_c))$ is computed by minimizing a given cost function (F(x,u)) over a certain prediction horizon(T_p) in the future.

3. Apply the calculated input until the sampling instant Sampling time.

4. Repeat step 1.

3.2.1. MATHEMATICAL FORMULATION OF NMPC

Shown below is the mathematical formulation of NMPC for a continuous system (Allgöwer, Findeisen, & Nagy, 2004). Consider a continuous system represented by

$$\dot{x}(t) = f\big(x(t), u(t)\big), x(0) = x_0$$

With input and state constraints,

$$u(t) \in U, \forall t \geq 0,$$

$$x(t) \in X, \forall t \geq 0.$$

Here $x(t) \in R^n$ and $u(t) \in R^m$ denote the vector of states and inputs respectively.

Problem 1: To find $\min_{\bar{u}(.)} J(x(t), \bar{u}(.))$

The system is described by,

$$\dot{\bar{x}}(\tau) = f\big(\bar{x}(\tau), \bar{u}(\tau)\big),$$

With $\bar{x}(t) = x(t)$ and

$$\bar{u}(\tau) \in U, \forall \tau \in [t, t+T_c],$$

$$\bar{u}(\tau) \in \bar{u}(t + T_c), \forall \tau \in \left[t + T_c, t + T_p\right],$$

$$\bar{x}(\tau) \in X, \forall \tau \in \left[t, t + T_p\right],$$

The equations indicate that $\bar{u}$ is varied from t to $t+T_c$ and is left constant until $t+T_p$ with the value at $t+T_c$. But$\bar{x}$, the state of the system changes throughout $t+T_p$ as the input is still applied but not changed.

The cost functional can be represented

$$J\big(x(t), \bar{u}(.)\big) \coloneqq \int_{t}^{t+T_p} F\big(\bar{x}(\tau), \bar{u}(\tau)\big) d\tau \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (1)$$

The Cost functional J can be represented by the stage cost F which specifies performance- is represented by,

$$F(x, u) = (x - x_s)^T Q (x - x_s) + (u - u_s)^T R (u - u_s)$$

Where x_s and u_s are the desired trajectory of state and the input.

Let $\bar{u}^*\big(.\,; x(t)\big) : \left[t, t + T_p\right] \to U$, be the optimal solution of (1) and the open loop optimal problem is solved repeatedly at $t_j = j\delta$ with $j=0,1,2\dots$

Then the input given to the system is written as

$$u(t) \coloneqq \bar{u}^*\left(t; x\left(t_j\right)\right),$$

Therefore the nominal closed loop system is given by,

$$\dot{x}(t) \coloneqq f(x(t), \bar{u}^*\left(t; x\left(t_j\right)\right)),$$

The optimal cost of (1) as a function of state is given by a value function V,

$$V(x) = J(x, \bar{u}^*(.\,; x)),$$

This function plays a vital role in the stability analysis of NMPC as it serves as a Lyapunov function candidate.

3.2.2. PROPERTIES OF NMPC:

If the cost over T_p is reduced by the optimization algorithm, then it would be even better if the cost were to be optimized with T_p and T_c at infinity. However the computation would become extremely difficult with this setting and therefore finite horizons are used.

Therefore a shorter horizon pertains to lower cost. But when using a finite horizon the predicted open loop values and the actual states and inputs differ significantly, even without model mismatch and external disturbances. This difference causes two problems.

- The goal to minimize the performance objective over infinite horizon of the closed loop by a feed back is not achieved.
- There is no guarantee that the closed loop system would be stable.

According to (Findeisen & Allgöwer, 2005) the key characteristics of NMPC are:

- NMPC allows the use of a nonlinear model for prediction.
- XNMPC allows the explicit consideration of state and input constraints.
- In NMPC a specified performance criteria is minimized on-line.
- In NMPC the predicted behavior is in general different from the closed loop behavior.
- The on-line solution of an open-loop optimal control problem is necessary for the application of NMPC.
- To perform the prediction the system states must be measured or estimated.

3.2.2.1. STABILITY

One of the main concerns in using NMPC is the compromise forced because of the finite horizons as the predicted values and the real values of the state and input are not the same. In normal sense one would like to have a strategy to attain closed loop stability irrespective of the choice of performance parameters in the cost and to get the current scheme as close as possible to the infinite horizon NMPC scheme.

"An NMPC strategy that achieves closed loop stability independent of the choice of the performance parameters is usually referred to as NMPC approach with *guaranteed stability."* (Findeisen & Allgöwer, 2005). In order to implement guaranteed stability for finite horizon implementations additional constraints are added to the equations. These constraints are not based on Performance specifications but only to assure the guaranteed stability. One approach to this issue is Quasi Infinite Horizon Approach (Q1HNMPC) but the reader can better understand it from (Nicolao, Magni, & Scattolini, 2000). But to implement these schemes in practical systems is highly difficult (Luis Sousa Brito, 2011).

3.2.2.2. EFFECT OF SAMPLING TIME

In this thesis the parameter sampling time or the sampling time is added to the NMPC algorithm and its effects are studied. The main purpose of sampling time is to smoothen the reaction curve of the plant. Also having a short sampling time increases the response time of the plant to disturbances.

One can argue that having a short control horizon instead of having a separate parameter for sampling time may also work in the same way. But having a short control horizon means forcing the controller to compute inputs of short duration(T_c) for corresponding outputs of longer duration (T_p). Thus during the optimization run the controller is forced to apply larger changes in the input in order to produce the same predicted output. When a large input step is involve there is always a chance that this might result in a bang-bang control.

At the same time care must be taken to counter act the disturbances to plant. Since the biogas production is a very slow process the effect of disturbances may be realized late. Therefore the disturbances should be counteracted as early as possible.

To solve this paradox a new variable called the sampling time is added to algorithm. The controller computes the input for the system with a prediction horizon and control horizon as before, the only change however is the fact that the input is applied for a period of sampling time (with $\delta<T_c<T_p$). This means the controller will not be forced to compute a larger step and the state of the system is monitored and controlled for every ‘δ’ days. During each δ the whole process is done again.

3.3. NMPC STRATEGY TO BIOGAS PLANT

As discussed in the previous chapter the Biogas plant is a complex and highly non linear process which requires careful control and optimization of the substrate feed. Also the Biogas process is a slow process for which the period for control may extend for days. This means that a reliable strategy should be established which can predict the effect of input on the Fermenter beforehand. Also one cannot rely on the experience of the operator, because this creates dependability on the particular operator and increases his work load.

The use of Non linear model to approximate the function of a Fermenter and the application of prediction makes NMPC an ideal choice to a Biogas Plant. If this technology is perfected, then the labor required can be reduced, profits can be increased and reliability can also be increased.

4. OPTIMIZATION

4.1. INTRODUCTION

CMAES stands for **C**ovariance **M**atrix **A**daptation **E**volution **S**trategy. Normally optimization methods depend on the differentiability of the objective function to find desired result. However modern day problems are so complex that the objective function is often non differentiable and non convex in nature and there it becomes difficult to find a global optima.

Also when the state of a system is evaluated through a simulation, explicit calculation of the status is not possible. Therefore an approximate differentiable model is formulated. However the model and real system are not properly matched and usually the global optima does not coincide with global optima of the real system. (Hoshimura, 2007).

At this point (Hansen, 2011) proposed an Evolutionary Algorithm which uses Normal distribution to find search points without using derivatives. However this method can be applied to unconstrained Optimization (Hoshimura, 2007).

Methods like Particle Swarm Optimization, Genetic Algorithm and CMAES find a near optimal solution in a practical time rather than finding an exact solution in an impractical period. The CMAES methodology uses Multivariate normal distribution to generate its search points and updates its mean and covariance matrix to move the distribution and finding the optimal solution.

4.2. FUNDAMENTALS:

Before proceeding to the main explanation, it is better to go through the fundamental concepts based on which CMAES formulates its theory.

- **EIGEN DECOMPOSITION OF A POSITIVE DEFINITE MATRIX:**

  Eigen Decomposition is the factorization of a matrix into a canonical form, in which the matrix is represented in terms of its Eigen values and Eigen vectors. The Eigen Decomposition of a Matrix C,

$$C = BD^2B^T$$

  C- Positive Definite Matrix

  B- Is an orthogonal matrix of Eigen Vectors.

  D- A Diagonal matrix with square roots of eigenvalues of C. diag(d_1,d_2,…d_n).

- **THE MULTIVARIATE NORMAL DISTRIBUTION:**

  According to (http://en.wikipedia.org) a random vector is said to be *k*-variate normally distributed if every linear combination of its *k* components has a univariate normal distribution. The multivariate normal distribution is often used to describe, at least approximately, any set of (possibly) correlated real-valued random variables each of which clusters around a mean value.

$$\begin{aligned} \boldsymbol{\mathcal{N}}(\boldsymbol{m},\boldsymbol{C}) &\sim \boldsymbol{m} + \boldsymbol{\mathcal{N}}(\mathbf{0},\boldsymbol{C}) \\ &\sim \boldsymbol{m} + \boldsymbol{C}^{\mathbf{1/2}}\boldsymbol{\mathcal{N}}(\mathbf{0},\boldsymbol{I}) \\ &\sim \boldsymbol{m} + \boldsymbol{BD}\ \boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{\mathcal{N}}(\mathbf{0},\boldsymbol{I}) \\ &\sim \boldsymbol{m} + \boldsymbol{BD}\ \boldsymbol{\mathcal{N}}(\mathbf{0},\boldsymbol{I}) \end{aligned}$$

  ~ denotes equality in distribution

  $\boldsymbol{\mathcal{N}}(\mathbf{0},\boldsymbol{I}), \boldsymbol{\mathcal{N}}(\mathbf{0},\boldsymbol{C})$-represents Multivariate distribution.

  $\boldsymbol{m}-$ Mean Value of search distribution.

  B- Is an orthogonal matrix of Eigen Vectors.

  C- Positive Definite Matrix

D- A Diagonal matrix with square roots of eigenvalues of C. diag(d_1,d_2,…d_n)

- **RANDOMIZED BLACK BOX OPTIMIZATION:**

  A black box is a device that can be described or studied only by its input, output and transfer characteristics without any knowledge about the internal working of the system. In this case we expect to minimize an objective function. The objective is to find one or more search points (Candidate Solutions), where we want to minimize an objective function (or cost function or fitness function).

- **HESSIAN AND COVARIANCE MATRICES:**

  Covariance is a measure of how to variables change together. A covariance matrix is a measure of how a set of variables change with each other. The final objective of covariance matrix adaptation is to closely approximate the contour lines of the objective function f. A Hessian matrix is a matrix of second order differentials.

4.3. ANALYSIS

This section describes the algorithm and the basis for understanding the CMAES methodology. This section is structured into 4 parts which efficiently modularizes the concept to enable easier understanding. Also refer Figure 4-1.

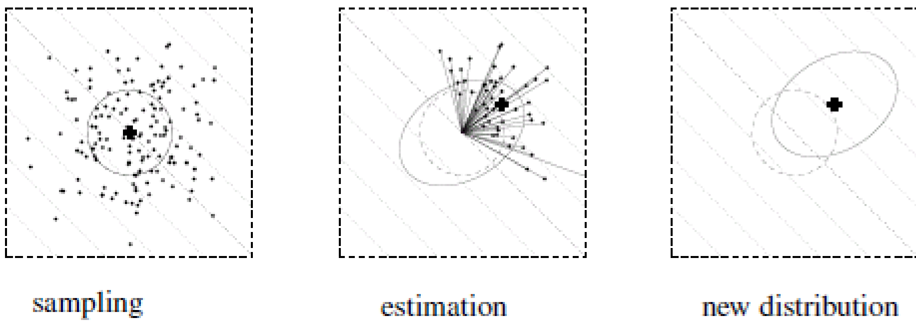


Figure 4-1 CMAES Step by Step Process

- **SAMPLING:**

  Sampling is the process of moving the mean and choosing new points in the search space using a normal distribution. The new search points are chosen with the mean, the step size and a multivariate normal distribution.

$$x_k^{(g+1)} \sim \quad m^{(g)} + \sigma^{(g)} \mathcal{N}\left(0, C^{(g)}\right) \, for \; k = 1, \dots . \lambda$$

$\mathcal{N}\left(0, C^{(g)}\right) -$ *is a multivariate normal distribution with zero mean and*

Covariance matrix $C^{(g)}$.

$x_k^{(g+1)} \epsilon \mathbb{R}^n -$ *is the kth individual from* $g+1$ *th generation*

$m^{(g)} \epsilon \mathbb{R}^n$, *mean value from the search distribution of generation* g.

$n \epsilon \mathbb{R}^n$, *Search space dimension.*

$\sigma^{(g)} \epsilon \mathbb{R}_+$, *step size at generation* g.

$C^{(g)} \epsilon \mathbb{R}^{n*n}$, *Covariance matrix at generation* g.

$\lambda \geq 2$, *Population size or sample size or number of off spring.*

- **SELECTION AND RECOMBINATION:**

The mean is calculated from the selected points in the search space. i.e. a weighted sum of points are used to calculate the mean. Selection involves choosing of points with appropriate weight and recombination involves calculation of the mean.

$$m^{(g+1)} = \sum_{i=1}^{\mu} w_i x_{i:\lambda}^{(g+1)}$$

$$\sum_{i=1}^{\mu} w_i = 1, w_1 \geq w_2 \geq \cdots \geq w_\mu > 0$$

$\mu \leq \lambda$ *is the parent population size. i.e. the number of selected points*

$w_{i=1\ldots\mu} \epsilon \mathbb{R}_+$, *positive weight coefficients for recombination*

$w_{i=1\ldots\mu} = \frac{1}{\mu},$

The above equation calculates the mean value of μ *selected points.*

$x_{i:\lambda}^{(g+1)}$, *i − th best individual out of* $x_1^{(g+1)}, \ldots, x_\lambda^{(g+1)}$. *the index i: λ denotes the index of the i − th ranked individual and*

$f\left(x_{1:\lambda}^{(g+1)}\right) \leq f\left(x_{2:\lambda}^{(g+1)}\right) \leq \cdots \leq f\left(x_{\lambda:\lambda}^{(g+1)}\right)$, *and f is the objective function to be minimized.*

- **MATRIX ADAPTATION:**

This section describes the updating the covariance matrix. The covariance matrix does not consider the sign of the step; rather it considers the magnitude or the scale.

$$C^{(g+1)} = \left(1 - c_1 - c_\mu\right) C^{(g)} + c_1 P_c^{(g+1)} P_c^{(g+1)^T} + c_\mu \sum_{i=1}^{\mu} w_i \, y_{i:\lambda}^{(g+1)} (y_{i:\lambda}^{(g+1)})^T$$

$$c_1 \approx \frac{2}{n^2}$$

$$c_\mu \approx \min\left(\frac{\mu_{eff}}{n^2}, 1 - c_1\right).$$

$\left(1-c_1-c_\mu\right)$ - is the learning rate for the covariance matrix adaptation.

$$y_{i:\lambda}^{(g+1)} = \frac{x_{i:\lambda}^{(g+1)} - m^{(g)}}{\sigma^{(g)}}$$

$$\sigma = Step\ Size$$

$$P_c^{(g+1)} = Evolution\ path\ at\ generation\ g+1.$$

$$\mu_{eff} = The\ variance\ effective\ election\ mass.$$

$$P_c^{(g+1)} P_c^{(g+1)^T} = Rank\ one\ Update$$

$$\sum_{i=1}^{\mu} w_i\, y_{i:\lambda}^{(g+1)} (y_{i:\lambda}^{(g+1)})^T = Rank - \mu\ update$$

To achieve fast search, i.e. competitive performance, the population size λ must be small. But with a small population size, it is not possible to get a reliable estimator for a good covariance matrix. As a result one must use the information from previous generation. So the number of generations must be increased.

The rank-μ-update is called so because the sum of outer products is of rank min (μ,n)(with probability one).

The right most summand is of rank one and adds the most probable term for y_{g+1} into the covariance matrix. Therefore the probability to generate y_{g+1} in the next generation increases. This update is called rank one update.

- **STEP SIZE VARIATION:**

The covariance matrix adaptation does vary the step size explicitly. Since negative values are not permitted in the covariance matrix, only the scale is varied by fading out the old information via $1 - c_1 - c_\mu$. Therefore there is a need to introduce another factor which controls the step size explicitly. To vary the step size $\sigma^{(g)}$ we use the evolution path $p_\sigma^{(g+1)}$. The evolution path is the sum of successive steps which have been done in the procedure. This procedure can be applied independent of the covariance matrix and is called *cumulative path length control, cumulative step-size control or cumulative step length adaptation.* The length of the path is decided based on the following:

- If the evolution path is short, and if the single steps cancel each other out, then the step size should be decreased.
- If the evolution path is long and the single steps are pointing to similar direction, then the same distance can be covered by fewer but longer step sizes in the same direction. Therefore the step size should be increased.
- The desired situation is when the steps are perpendicular in expectation.

Shown below is the formula for the evolution path, which is used for the calculation of the step size.

$$p_\sigma^{(g+1)} = (1 - c_\sigma)p_\sigma^{(g)} + \sqrt{c_\sigma(2 - c_\sigma)\mu_{eff}}\, C^{\left(g^{-\frac{1}{2}}\right)}\left(\frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}}\right)$$

The step size formula is shown below,

$$\ln \sigma^{(g+1)} = \ln \sigma^{(g)} + \left(\frac{c_\sigma}{d_\sigma E\|\mathcal{N}(\mathbf{0},\boldsymbol{I})\|}\right)\left(\left\|p_\sigma^{(g+1)}\right\| - E\|\mathcal{N}(\mathbf{0},\boldsymbol{I})\|\right).$$

$Where\ , d_\sigma \approx Damping\ parameter, scales\ the\ change\ magnitude\ of\ \ln \sigma^{(g)}.$

$E\|\mathcal{N}(\mathbf{0},\boldsymbol{I})\| -\ Expectation\ of\ the\ Euclidean\ norm\ of$

$\mathcal{N}(\mathbf{0},\boldsymbol{I})\ Normal\ Distributed\ random\ vector.$

4.4. ADVANTAGES AND DISADVANTAGES

Advantages of CMAES are

- Can be applied to non-differentiable problems, which means CMAES can be applied in places where steepest descent method or the quasi Newton method have been rendered useless.
- Shows better performance than methods like GA and PSO etc… since CMAES takes second order derivative into account approximately by using covariance matrix (Hoshimura, 2007).

Disadvantages are:

- When applied to easy problems which are solved immediately by normal optimization methods, CMAES takes longer time and therefore higher cost for calculating. This is mainly due to the generation of search points and evaluation of each point and then finding a solution using a covariance matrix, which requires larger time for computation.

4.5. CMAES FOR NMPC

In order to use CMAES to the NMPC strategy certain calibrations have to be done. While the NMPC strategy is ideal the application of CMAES into it brings certain difficulties. For example the CMAES computes solutions in a discrete space while the NMPC implementation requires a continuous input. The processing power and calculation time is also a major concern. All the required changes are explained in Chapter 6.

5. BIOGAS TOOLBOX

5.1. SUNDERHOOK

The plant used in this thesis is called Sunderhook Biogas Plant. This plant is located at Sunderhook 8, 48599 Gronau (North Rhine-Westphalia), Germany. A virtual model of this plant is constructed using Matlab and this model is used for further study and analysis.

The plant description and characteristics are shown below. The Sunderhook Plant consists of two Fermenters and two CHPs. There are 5 types of substrates which are loaded to the plant.

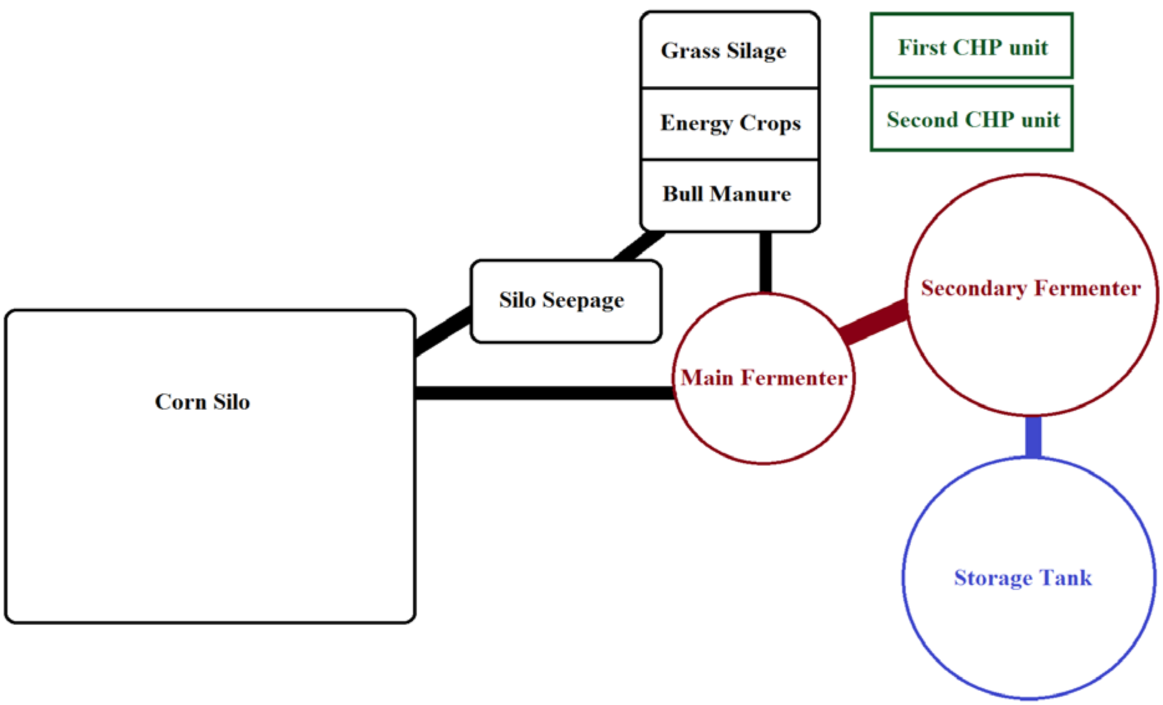


Figure 5-1 – Configuration of the Sunderhook plant (Luis Sousa Brito, 2011)

The main Fermenter is shorter in Diameter and Higher in Height when compared with the second Fermenter. There are 5 silos (i.e. a pit or other airtight structure in which green crops are compressed and stored as silage. (Silo)) for storing the substrates. The sizes of the silos are designed according to the utilization of the substrates in the fermentation process as shown in figure 3 (Luis Sousa Brito, 2011).

The process variable measurements from the Biogas plant are recorded in MS Access Database and are provided through online remote access. Also the entire production system works on a WinCC-based control system.

The data obtained from the plant are used to for two purposes.

- Controlling the plant
- Further analysis and Optimization.

Since the whole digestion process is very slow, the process measurements are taken every two hours. The measured process variables are

- Gas volume & Composition.
- Temperature of the main and secondary Fermenters.
- The gas temperatures at the inlet, after the scrubber and after re-heater.
- The amount of electricity produced by the two CHP units.
- Overall power consumption of the system.
- PH values, dry matter concentrations and the operational work time of pumps and transport equipments (e.g. transport of substrates and digestrate discard).

5.2. BIOGAS PLANT- SIMULINK MODEL

This section gives a basic idea of the model and its construction. Shown below in figure 5 is the actual configuration of the Sunderhook plant. Since the real plant has already been discussed in the previous section, the reader is required to make a comparative study between the real and the virtual plant to understand the structure.

The blocks are modularized for better operation, control and understanding. For example in the model shown below the blocks used are

- Substrate Mixer - Options for the type and source of Volume Flow of substrate are chosen here. However the numbers of substrates are chosen elsewhere.
- ADM1 -Anaerobic Digestion Model- This block was designed by (Batstone, et al., 2002) and this simulates the non linear behavior of the digester. This block (See figure 4) uses 37 dimension states and has a number of differential equations to simulate the non linearity. Therefore naturally this is very complex. This block simulates an ideal digester at 41 deg Celsius.

  INPUT:
  - Qin - Input Substrate Flow

  OUTPUT:
  - Gas - Biogas amount [m^3/d] .
  - Gas[%] - Biogas concentrations [%].
  - Qe - Substrate Flow Output
  - Int.vars - Internal variables.
  - pH - pH values obtained during the process.

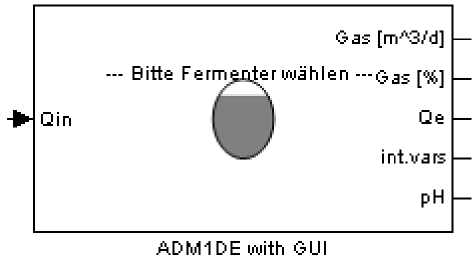


Figure 5-2 - ADM1 Block

- Storage Tank - To store the substrate for disposal.
- Pump - To pump substrate between Fermenter 2 to Fermenter 1.
- Sensors - To Obtain measurements.
- BHKW - CHP Combined Heat and Power Generation Unit.
- Energy Analyzer - To analyze the energy produced.

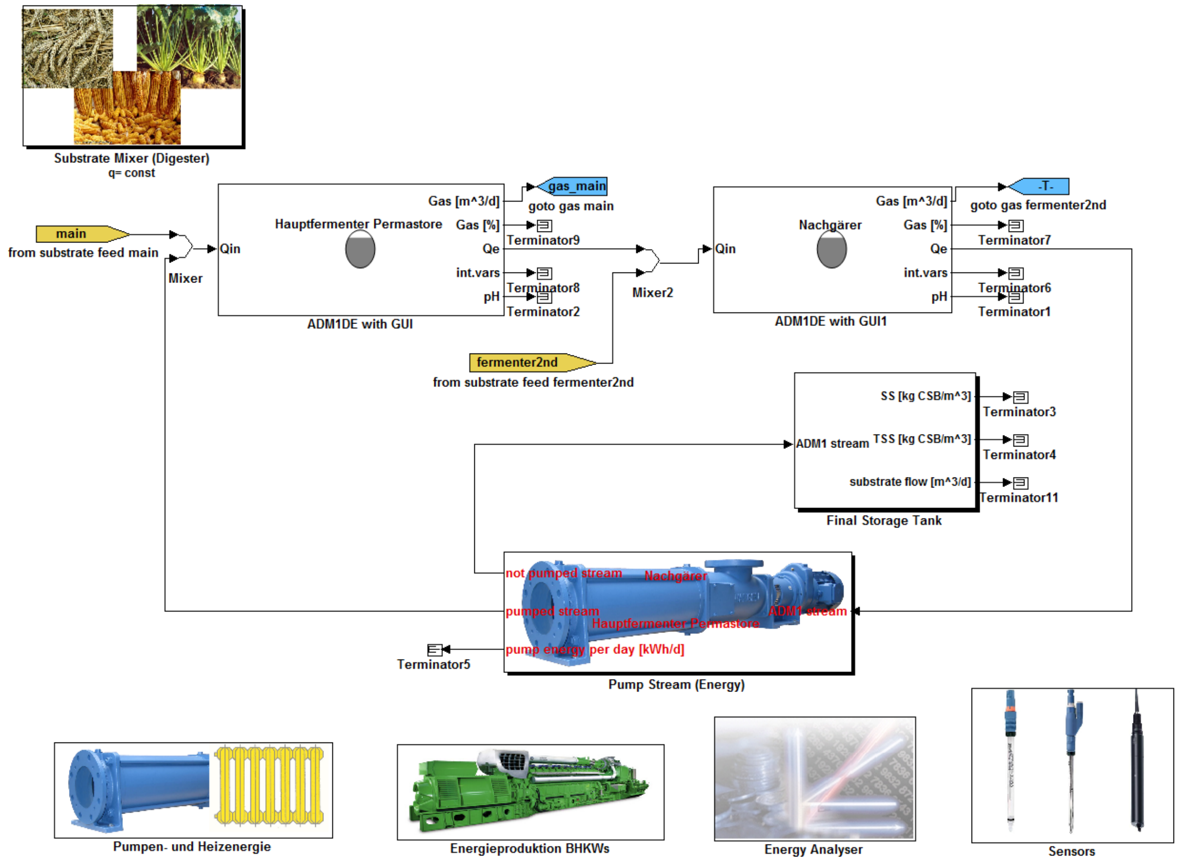


Figure 5-3 - Simulink Model for Sunderhook plant (Luis Sousa Brito, 2011)

5.3. CONSTRUCTION OF THE MODEL

There are four steps involved in creating the model.

- Creating configuration files needed for simulation
- Creating files containing simulation data
- Defining lower and upper bounds for input parameters
- Create the simulation model.

Each step can be executed with the help of a GUI. To make it easier the GUI shown below encapsulated all the other GUIs. The user can access all the other required GUIs from this GUI. This GUI can be called through the Matlab command line by:

- *Gui_biogas_plant_modelling*

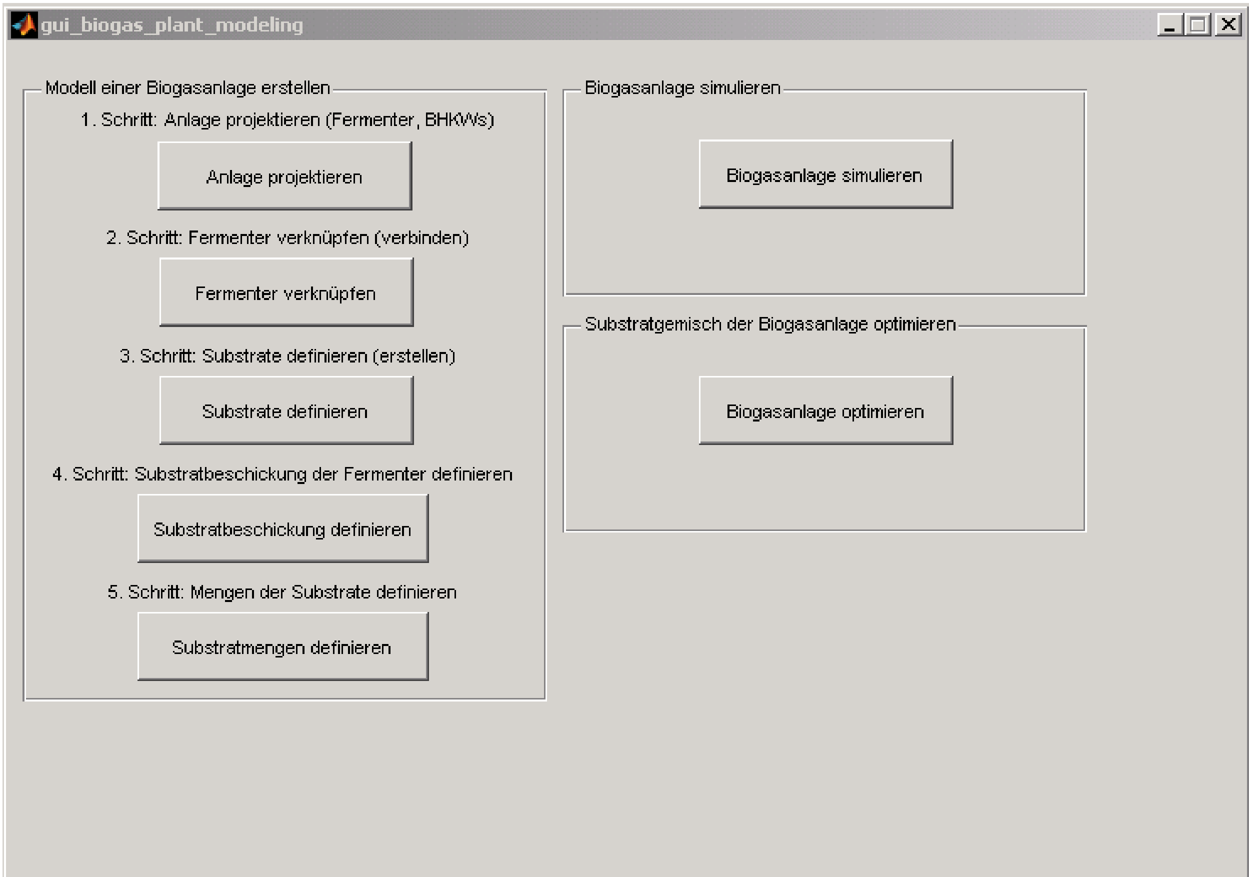


Figure 5-4 GUI showing the encapsulation of all steps involve in Model creation

1. CREATING CONFIGURATION FILES:

Here the plant and substrates are defined. Common parameters required for the plant are the Fermenter size, Temperature, Volume etc… (As in figure 5-5)Common parameters required for the substrate are Chemical Composition, Name, temperature, Solid content etc… (As in figure 5-6) Multiple Fermenters and Substrates can also be defined.

The configuration files can also created using the following GUI shown in figure 5-5, the GUI can be called from Matlab command line through.

- *gui_plant*

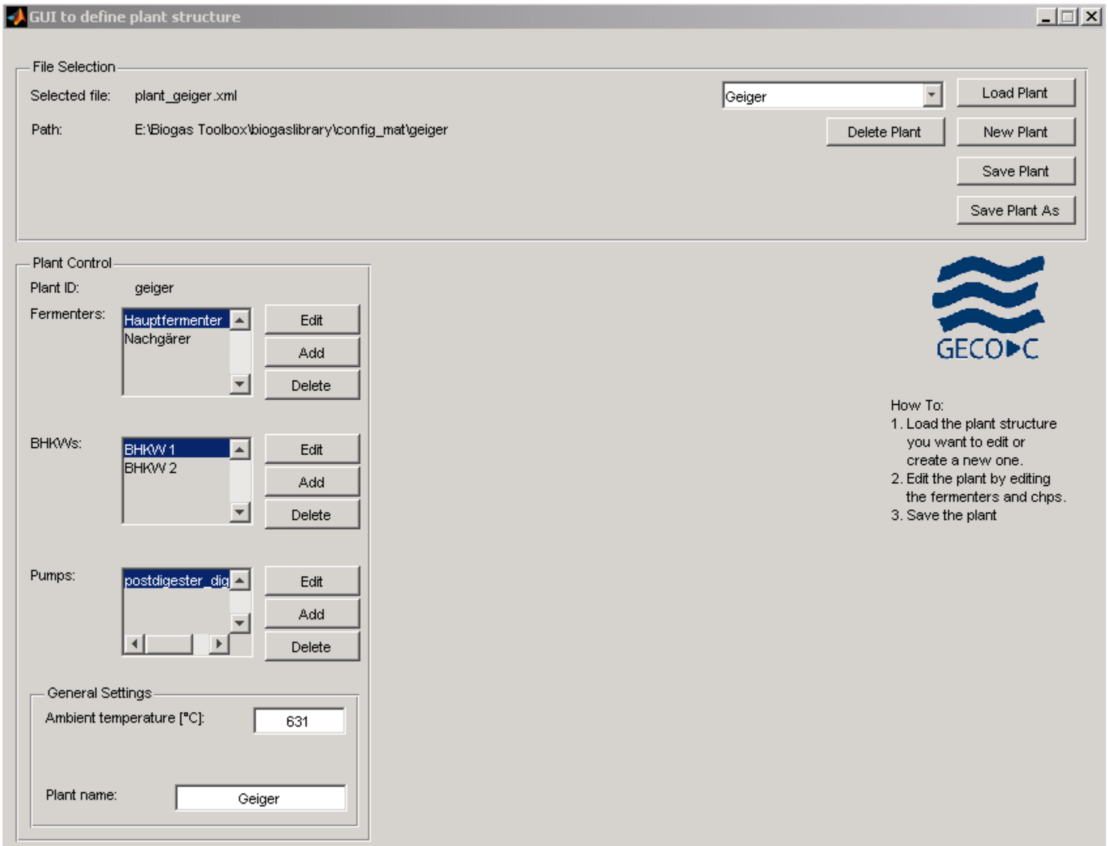


Figure 5-5 GUI Plant to define plant configuration

- *gui_substrate*

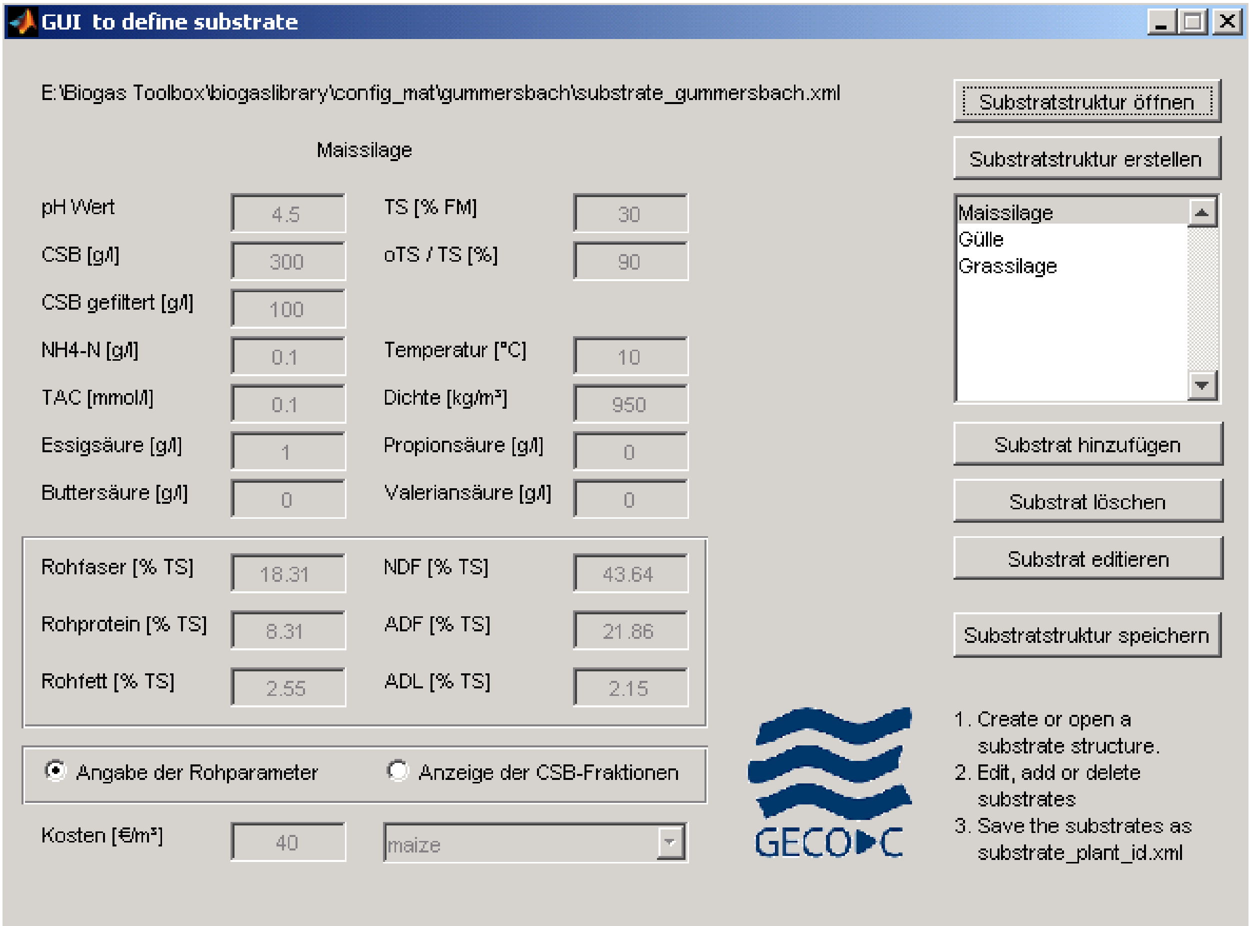


Figure 5-6 GUI defining substrate characteristics

Also the configuration of the plant i.e. the connection between Fermenters and between substrates and Fermenters can also be set here. These files are the 'Network' files which are indicated in the name of the file. The command line call is shown below:

- *gui_substrate_network*
- *gui_plant_network*

2. CREATING FILES CONTAINING SIMULATION DATA:

These files contain the amount of substrate pumped into each Fermenter and the amount pumped between Fermenter. This can be done with the help of the GUIs as shown in figure 5-7.

- *set_input_stream*

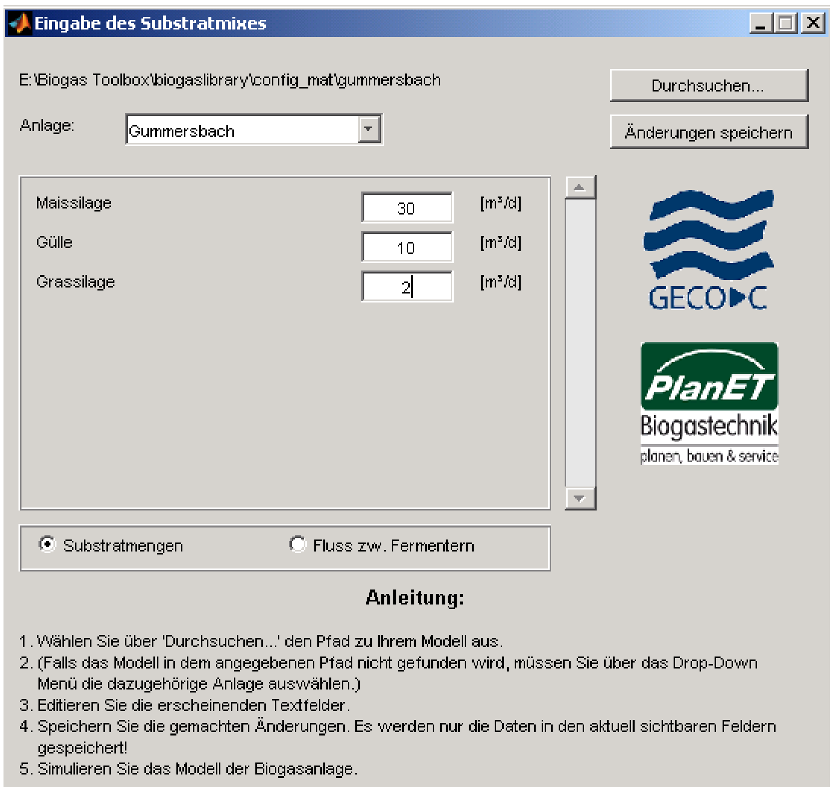


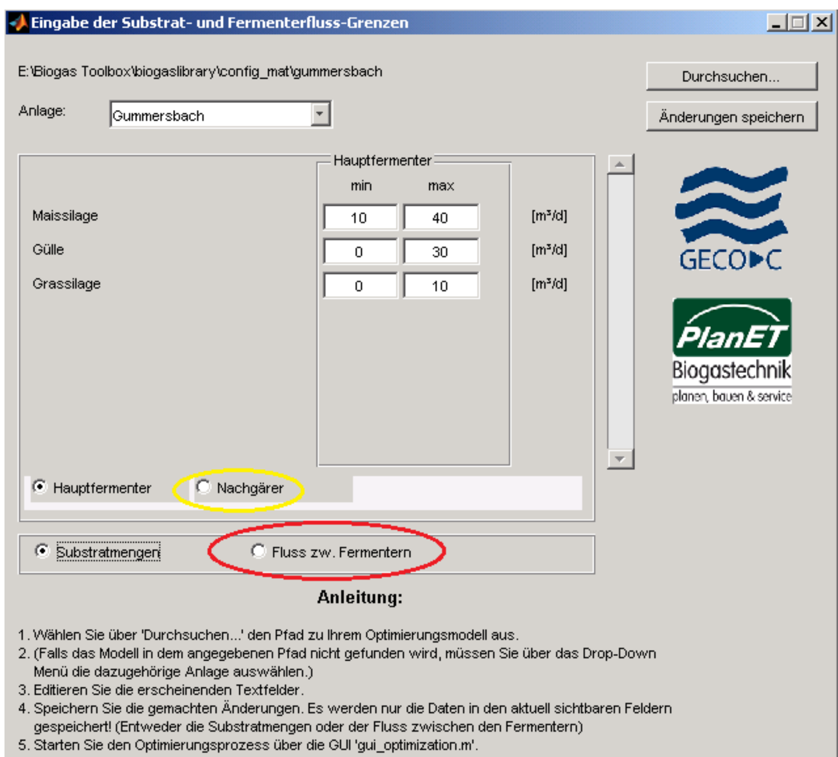
Figure 5-7 GUI for defining the substrate mixes

3. DEFINING LOWER AND UPPER BOUNDS FOR INPUT PARAMETERS:

As the name suggests, the minimum and maximum value of the substrate flow into each Fermenter is set here. Also the files for Digester state are created here. As shown in figure 5-8 the required plant is chosen. The connections between the plants are automatically loaded as per the settings specified by the previous GUIs.

The minimum and maximum values can be set for each Fermenter. In figure below the minimum and maximum values of each substrate to the main Fermenter(Fermenter I-Hauptfermenter) is shown. In order to access the next Fermenter the option circled in yellow should be accessed. Similarly the option circled in red is chosen for the setting the minimum and maximum values between Fermenter. This GUI can be called from the Matlab command line through,

- *set_input_stream_min_max*



-

Figure 5-8 GUI for defining substrate and Fermenter boundaries

4. CREATE THE SIMULATION MODEL

The simulation model is created with the help of Simulink. All the necessary blocks are provided by Biogas Toolbox. For easy understanding the functions are modularized for each block. After placing the necessary blocks in the right position (As shown in Figure 5-3), the blocks are referenced to the files created in the previous steps.

Table 6 shows the files created during the above steps.

Shown below in figure 6 is an example GUI for the creation of configuration files for the Fermenter and Energy production units. Dimensions, Capacity and Temperature can be set with this GUI.

File	Explanation
Plant_plant_id Substrate_plant_id	The Dimension, Composition,Temperature etc…
plant_network_min_sunderhook plant_network_max_sunderhook	Total Amount of substrate that should flow into and between the Fermenters
substrate_network_min_sunderhook substrate_network_max_sunderhook	Amount of each substrate flowing into each Fermenter.
digester_state_min_sunderhook digester_state_max_sunderhook	A digester state if described by 37 states/dimensions.
fitness_params_sunderhook	The fitness coefficients required for the fitness function.

Table 5-1 – Various files used for the simulation model

6. NMPC OPTIMIZATION- SETUP, ALGORITHM AND IMPLEMENTATION

The NMPC optimization involves the use of CMAES methodology to compute an optimal input to the model described in the previous sections with due considerations on constraints, boundaries and the fitness function. To summarize the NMPC code must obtain settings from the user and use the given state of the Plant, the constraints, the boundaries and finding a maximum value of fitness to obtain an optimal sequence of inputs over the control horizon using various optimization methods. And then force that input onto the model over δ and obtain the results.

6.1. ALGORITHM

The NMPC procedure discussed in this thesis follows the steps as shown below(Refer (Gaida D. , Wolf, Baeck, & Bongards, 2012))

1. Initial Settings:

- Let
  - $m \in \mathbb{N}^+$ number of available substrates
  - $n \in \mathbb{N}^+$ dimension of the modeled state vector.
- Let
  - $T_c \in \mathbb{R}^+$ The control Horizon
  - $T_p \in \mathbb{R}^+$ The prediction Horizon.
  - With $0 \leq T_c \leq T_p$.
- The Anaerobic Digestion Model 1 has 37 Dimensions therefore n=37.

- Set the Boundaries
  - LB ϵ $\mathbb{R}^m$ The Lower Boundary
  - UB ϵ $\mathbb{R}^m$ The Upper Boundary
  - With LB≤UB.
- Set Optimal Substrate feed at k=0, $u_{opt,0}$ ϵ $\mathbb{R}^m$ to the current substrate feed of the biogas plant.

2. NMPC loop:

I. For k=1,2,3…Estimate the current operating state of the real plant $\hat{x}_{k-1} \epsilon \mathbb{R}^m$.

II. Define substrate feed boundaries lb ϵ $\mathbb{R}^m$, ub ϵ $\mathbb{R}^m$ such that

a. lb:= max((1-c). $u_{opt,k-1}$,LB)

b. ub:= max((1+c). $u_{opt,k-1}$,UB)

a & b ensure that the lb and ub are within the safe operating limits of the plant which is LB and UB.

c. c ϵ (0,1) satisfying lb ≤ ub .

III. Find Optimal substrate feed $u_{opt,k}$ by minimizing an objective function.

$$f: \mathbb{R}^n * \mathbb{R}^m \rightarrow \mathbb{R}$$

$$u_{opt\,,k} \coloneqq arg \min_{u\epsilon R^m} F\left(x\left(T_p\right), u\right)$$

$$With\ respect\ to\ lb \leq u \leq ub$$

$$x^{'}(\tau) = f_{ADM1}\left(x(\tau), u(\tau)\right)$$

$$x(0) = \hat{x}_{k-1}$$

$$u(\tau) = u = const, \forall\tau\ \in [0, T_p]$$

IV. Apply the optimal substrate feed $u_{opt\,,k}$ calculated over the control horizon to the real plant until the next sampling instant δ.

V. End FOR-Go to Step I.

6.2. OPTIMIZATION

The optimization of the substrate feed involves in finding an optimal operating point i.e. an optimal equilibrium. This is defined or decided by the fitness function. The optimal equilibrium can be said to dependent on the following:

1. The amount of each substrate fed into each Fermenter.
2. The sludge pumped between the Fermenters.
3. The steady state vectors of the Fermenters.
4. The parameters of the digestion process inside the Fermenters.

Optimality Criterions which are to be followed:

1. To find an optimal Equilibrium so that it corresponds to an optimal operation of the Biogas plant. All the factors pertaining to the equilibrium can be included in the fitness function.
2. Calibrating the plant involves calculating the internal state of the plant. Values such as the pH, methane concentration etc… is given and with this information the internal state of the plant has to be calculated.
3. Controlling the plant is similar to finding an optimal equilibrium. Here the algorithm must ensure that when we start at a critical state the final state of the plant should be in the stable region.
4. Linear and non linear constraints are given for each parameter and the optimization process has to consider all the constraints in calculating the optimal value. Some parameters may contradict each other i.e. that is increasing one parameter may push another parameter out of the limits. Therefore maintaining all the parameters within limits is an optimization problem in itself.

    However all these factors are taken care of by the toolbox and one need not separate between the criterions.

6.2.1. FITNESS FUNCTION

In any given experiment there should be a way or method to rate or measure the quality of the output. In this toolbox we make use of a fitness function which is used to determine the quality of the output of the biogas plant.

We cannot use the biogas production or the substrate costs alone as a measurement factor since many factors have to be considered and kept within limits. Therefore we use a function called *Fitness function* which takes into account the pH value, the heating costs, biogas production, substrate cost etc… and adds appropriate weights and gives a value. In this implementation the lower the fitness function the better is the result. In chapter 7 the experiments conducted mainly focus on reducing this fitness value.

6.3. SETUP

The NMPC setup deals with the creation of all the files necessary for the working of the NMPC optimization tool. Shown below in figure 6-1 are the files required by the NMPC. These files contain the limits like the Maximum and minimum value of substrate flow into the Fermenters, between the Fermenters etc…

To optimize the substrate feed the upper and lower bounds should be set. This can be set by the following:

- *set_input_input_stream_min_max*

This GUI thought same as the one used before for the construction of the model, the limits set here are used for optimization rather than control. Also in order to optimiza a plant we need the following

- A digester state

Initial state of the Plant.

- Parameters inside the Digester.

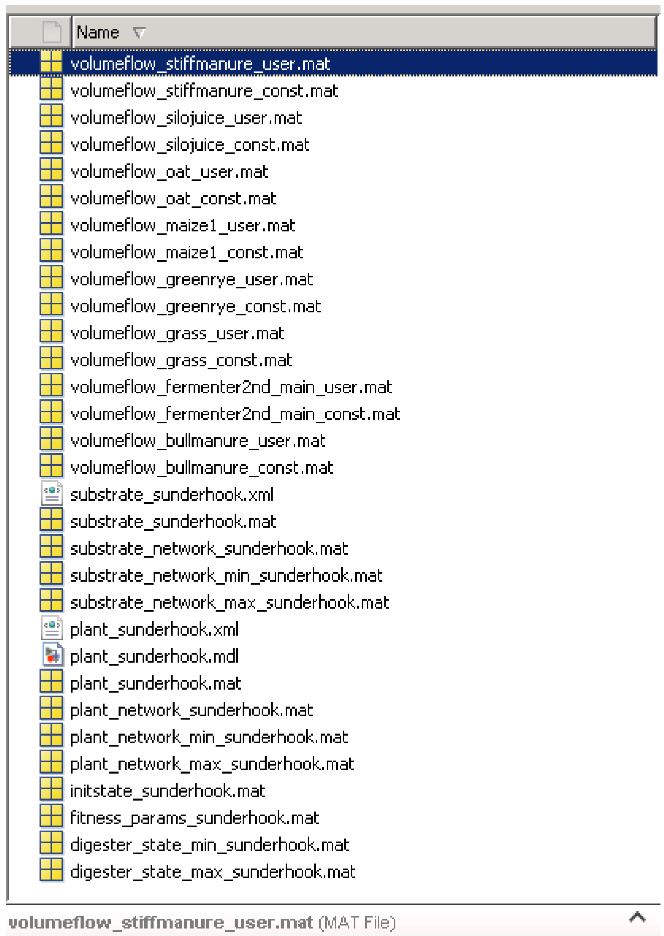


Figure 6-1 - Files involved in the simulation process

The command below creates a Digester state file with default minimum and maximum values for the digester states.

- *createDigesterStateMinMax('plant name')*

Similarly the following commands create an Initial state file and a file with default parameters for the digester.

  - *createNewInitstatefile(' plant name')*
  - *createADM1paramsfile(' plant name')*

CREATING THE VOLUME FLOW FILES:

The Volume flow files state the amount of substrate flowing into the Fermenter initially. Depending on the type of flow they are classified as ‘const’ and ‘user’. If the type of volume flow is set to ‘const’ then a constant flow is used throughout the process. But if it is set to user then a user defined flow is used i.e. values returned by the optimization option. The files are created during runtime when the optimization process computes a suitable input, this input is written in these files. To learn more refer (Matlab help).

6.4. NMPC SIMULATION

The proposed methodology is explained in this section. The NMPC implementation is divided into modular form. For easier understanding a flow chart is shown below. After the variables are initialized and loaded, the system is simulated for a period of 500 days in order to achieve a steady state. This steady state is essential because it should be certain that the current state of the plant is not influenced by any disturbances. Therefore a period of 500 days will ensure that a dynamic equilibrium is reached.

After this the NMPC strategy is applied through a FOR loop. In the loop the optimization methodology is called to compute the optimal input value. The optimization algorithm in this case CMAES computes the best possible input sequence which would yield maximum biogas production and reduced cost. A fitness function is usually used to indicate the best result. This fitness function takes into account the heating cost, electricity production, gas distribution etc… The cost function described in section 3.2.1 equation (1) is the same as the fitness function.

The fitness function replaces the cost function because of the fact that center of concern is only the final optimal state and not the way to reach the entire steady state. Therefore the integral $\int_{t}^{t+T_p} F(\bar{x}(\tau), \bar{u}(\tau))d\tau$ is not used instead the fitness is calculated at the end of the prediction horizon represented by T_p.

After the fitness values are calculated at the end of prediction horizon by the optimization strategy, the best value of input for which the fitness value was the best is chosen and applied to the plant for a period of δ.

Then the next iteration begins. At this juncture the plant state at Tp of the previous iteration is not used but the state at the end of δ is used, meaning that the plant was simulated for a period of δ. Similar process is carried out. At the beginning of each iteration, the Substrate optimization boundaries (‘lb’ and ‘ub’) are changed according the change value given.

At the end of the FOR loop the last state and input to the plant is saved into the files. After the FOR loop the plant is simulated for another 500 days to find whether the current system state leads to a stable state or not.

This is last step is important because we can know whether the controller always leaves the system in safe region or not.

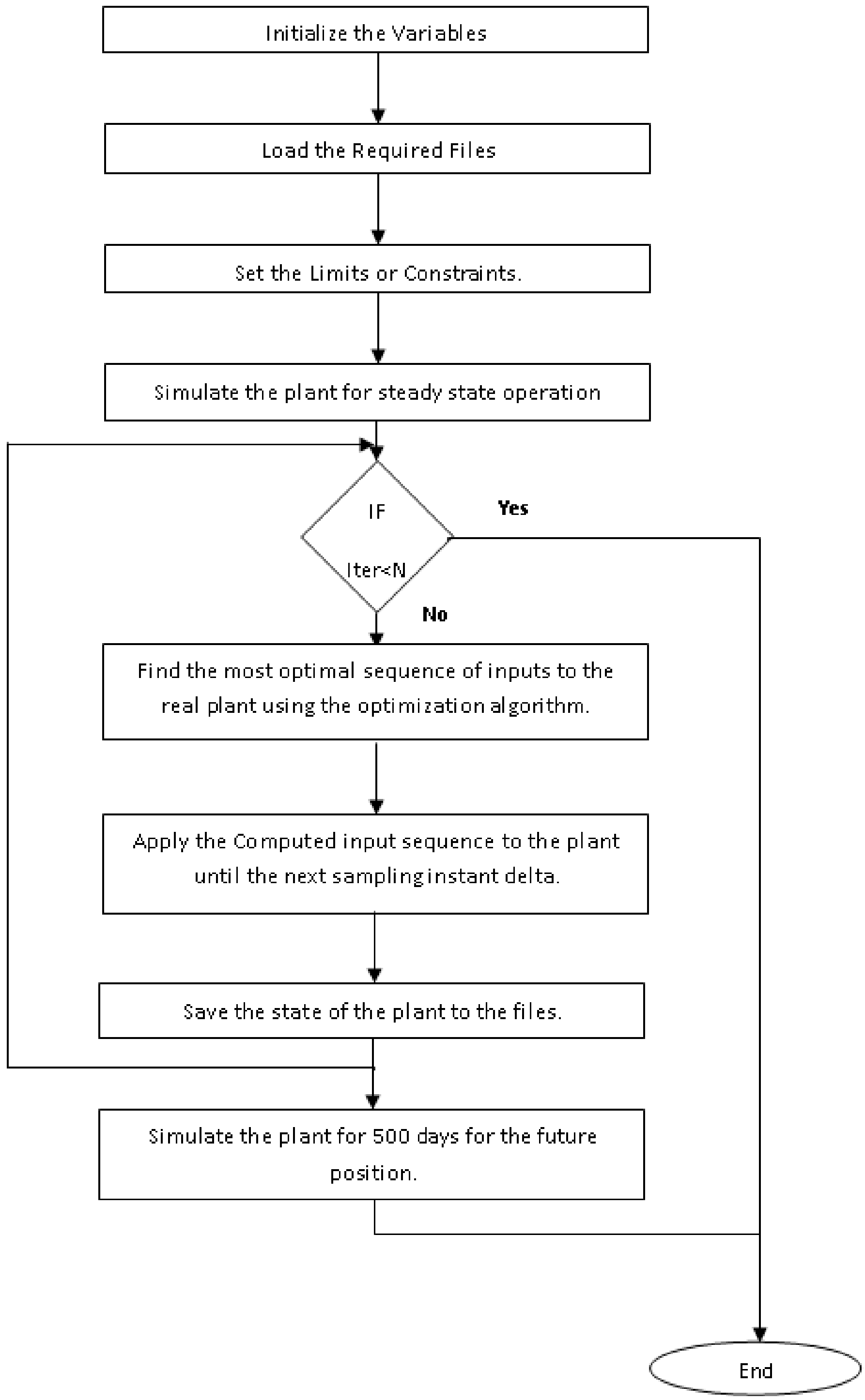


Figure 6-2 - Flow Chart describing the implementation of NMPC strategy

6.5. CMAES IN NMPC

Analysis of CMAES in a theoretical aspect alone will not enable the user to understand its implementation in NMPC. CMAES is an optimization methodology which chooses the best possible output in accordance with the constraints and then calculating the fitness function, the output for the best of value is then used. The possible inputs for the desired output are calculated before hand and CMAES searches in the input space for the best possible fitness value.

There are 3 parameters which are of paramount importance in using CMAES. The population size defines the number of dimensions involved (individuals) i.e. these individuals search in the defined space for an optimal maxima or minima depending on the requirement. Then the best value obtained by each individual is saved and the best value of all individuals is involved in calculating a mean which is weighted sum of the best values of all individual.

The number of generations is the number of times the mean is moved i.e. the number of generation defines how many times the covariance matrix (4.3) is updated.

Finally one might consider the fact that the input ‘u’ to the plant is a continuous function and when calculated through CMAES, it involves infinite dimensions i.e. one can pick out infinite values between two values in a continuous function. To calculate all these values would be highly impossible. Therefore the NMPC works out a strategy in such a way that the input sequence is piece wise constant function over δ i.e. the input is a constant value until δ and is changed to another constant value for the next δ. This means that the input sequence calculated over the control horizon would be a piece wise constant function. Therefore the calculated input will have discrete no. of steps or values in the form of a vector.

The length of the vector is decided by the value of the control horizon and the δ the sampling time. For example if T_c=6 and δ =2 the input vector calculated at every δ is of length 3 and the first value of the vector is applied to the plant.

$$u(t) \coloneqq \bar{u}^* \left(t; x\left(t_j\right)\right),$$

$$\bar{u}^*(\alpha) = \bar{u}^*(t), \forall \alpha \in [t, t+\delta],$$

6.6. OPTIMIZATION SPECIFICATIONS

It is very important to know the right optimization specifications. The two main parameters are Population size and No. of generations. These optimization parameters combined with the controller parameters determine the type of results. Since the input is a vector, the no. of simulations done per input value is equally divided among the input values. The no. of simulations is the product of Population size and No. of generations. The no. of steps or input vector length is the modulus of Control Horizon/ δ. Therefore if similar results are required then the number of simulations per input should be at least equal. This means that the complexity of calculation increases as the no. of steps increases.

Shown below is an example table (Table 7). Clearly Case 2, 3, 4 should give similar results as the no. of simulations per input is the same. Case 1 should have the worst result as the no. of simulations per input is low and Case 4 should have the best fitness value.

Case No.	*Population Size*	*No. of Generations*	*No. of simulations*	*δ*	*Control Horizon*	*Input Vector length (number of steps)*	*No. of Simulations/Input*
1	**4**	**2**	**8**	**2**	**6**	**3**	**2**
2	**4**	**2**	**8**	**6**	**6**	**1**	**8**
3	**4**	**6**	**24**	**2**	**6**	**3**	**8**
4	**12**	**2**	**24**	**2**	**6**	**3**	**8**
5	**4**	**6**	**24**	**6**	**6**	**1**	**24**

Table 6-1 – Dependency of fitness value on the number of Simulations

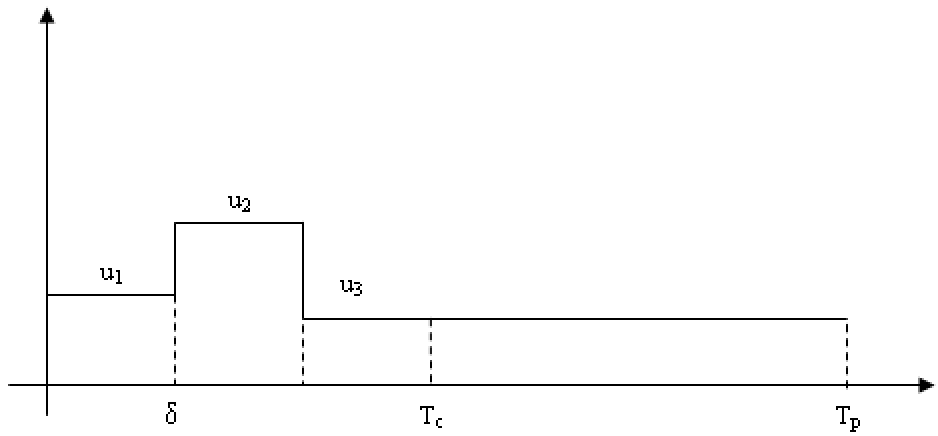


Figure 6-3 - Discretization of Input sequence for CMAES algorithm

CMAES does not differentiate between the position of inputs i.e. the optimization algorithm does not know whether the input value calculated is for the first value in the input vector or the second. Shown above in figure 6-4 the three steps are shown. The three steps are calculated independently by the CMAES algorithm irrespective of their position.

So far all the necessary concepts have been discussed and analyzed. In the next chapter the actual results obtained during the simulation of the model are discussed.

7. EXPERIMENTS

7.1. SETUP

The experiments are done in a completely virtualized model of a biogas plant. The reason behind using a virtual model is the model can be analyzed and tested in a safe and secure environment before it is applied to the real plant. The biogas model used in this experiment is shown in figure 7-1. The model consists of a nonlinear system, a dynamic optimizer, constraints and cost Functions as shown again in figure**.**

The setup can be considered as a black box model in which we know the input, output and transfer characteristics but we consider nothing about the internal working of the system. A biogas plant can be considered as a multiple input and single output system where the input are the substrate volume flows and the output is the Biogas production or in this case the variable for measuring the efficiency is the fitness variable. Our only goal is to get a better fitness.

A fitness value which is more to the negative side is considered as a better fitness i.e. the more negative the fitness is the better the biogas production is.

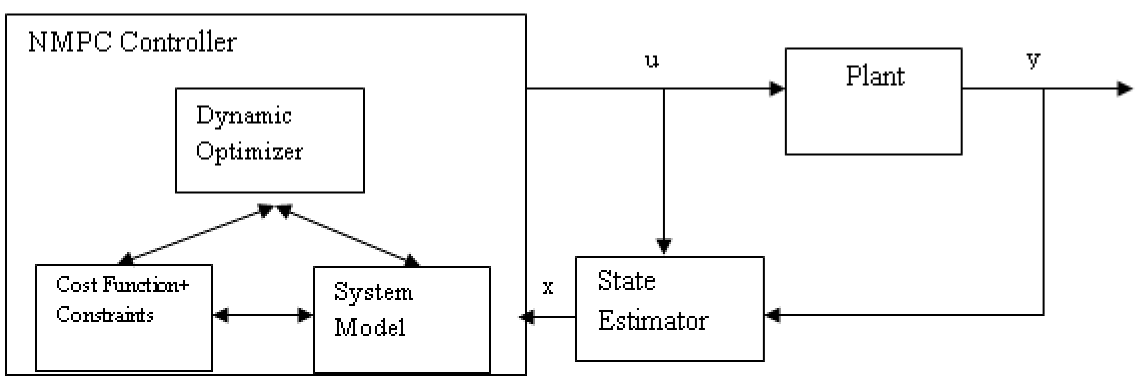


Figure 7-1- Overview of Implementation of NMPC strategy

Manipulated Variables	Substrate	Minimum (m^3/day)	Maximum (m^3/day)	Control Variables
Fitness Function	Maize	20	40	Fitness Function
	Green rye	0	0	
	Oat	0	0	
	Bull Manure	10	30	
	Grass	0	0	
	Stiff manure	0	5	
	Silo juice	1	3	

Table 7-1- A list of Substrates

Shown in the above Table 7-1 are the various manipulated and control variables. The NMPC strategy computes these variables in accordance with the constraints and the fitness function. The Flow between the Fermenters is kept constant i.e. the minimum and maximum values are the same. When this is the case then the corresponding variable is not involved in the optimization process. An optimum value for the variable will not be computed if the minimum and maximum values are the same.

7.1.1. COMPUTER CONFIGURATION

The experiments are conducted in an Intel Core 2 Quad, 2.5 GHz, 4 GB RAM, windows 7 and Matlab R2012b. Though, a real plant is not used in the simulation, the ADM1 model acts as the plant and the model. The model acts as a black box in which we do not consider the transfer characteristics, rather we consider the output from the model.

It is best practice to create a back up of the original files, particularly the initstate file (initial state file). This is because as the simulation is performed on the plant the initial state of the plant is continuously overwritten.

A working model of the plant is copied into a folder and all the changes required for the optimization and control methodology are created. The code is written in such a way that the files are copied into a new folder each time a new simulation begins; thereby no damage can be done to the original files.

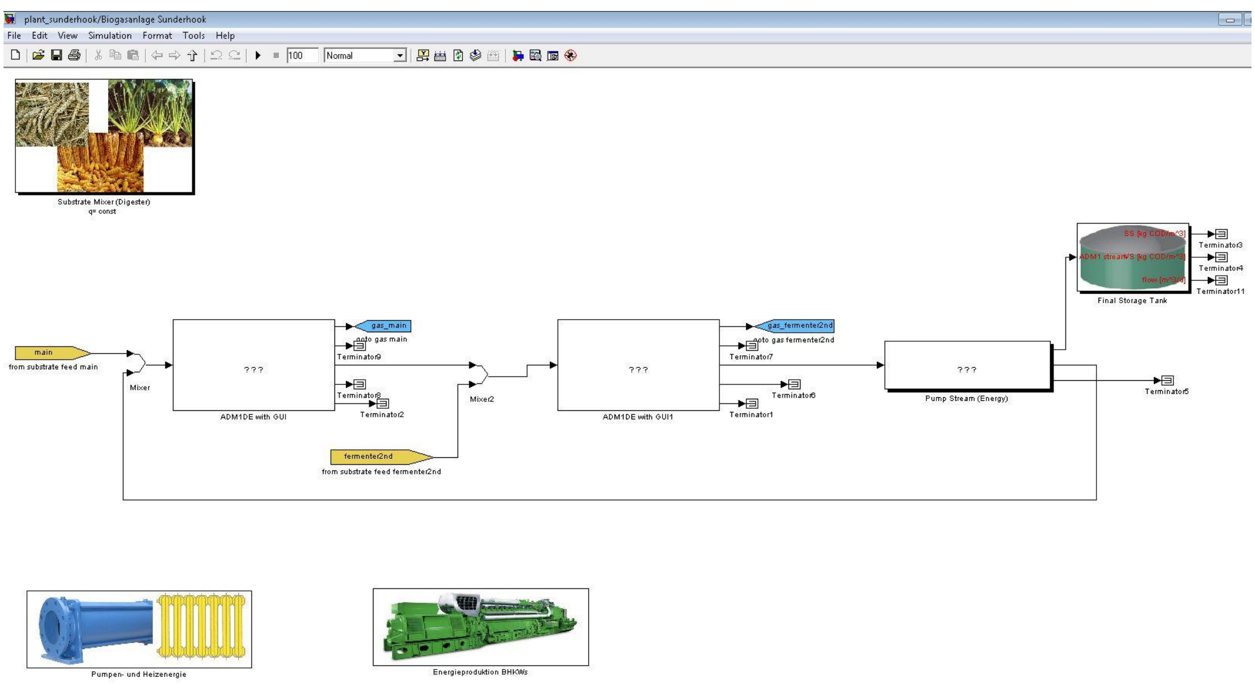


Figure 7-2- Overall Model as Constructed in MATLAB

7.2. INITIAL ANALYSIS

This section gives an idea how the NMPC strategy is analyzed i.e. what are the important aspects that should be analyzed.

.The output can be analyzed in many ways. For example shown below in Figure 7-3 are the changes in substrates through the NMPC strategy. In the next Figure 7-5 the corresponding change in Fitness values are shown. As discussed in the previous chapters the fitness value should be as low as possible and the change in substrate should be as smooth as possible. The x axis denotes the simulation time in days and the y axis denotes the Volume flow of the substrate in cubic meters. In Figure 7-3 one can observe that the number steps seen in the volume flow corresponds to the number of iterations. The substrates grass, green rye and oat are not fed to Fermenter denoted by a ‘0 m^3/day’ volume flow.

Also the amount of change in each substrate corresponds to ‘0.05 %’ as specified in the parameters section before starting this simulation. The fitness value plot shown below in Figure 7-4 follows a decreasing trend. This indicates a good response from the plant. However this will not be the case always.

For Number of steps (Number of Steps) = 1 the trend of fitness throughout the simulation is shown above in Figure 7-4. Shown below in Figure 7-5 is the trend of fitness for Number of steps=3. One can observe that the trend is not decreasing in nature; however the final value of fitness is lower than the initial value. The real problem is explained in detail in 7.3.2

Therefore what is important is the final fitness. The plant fitness has to get better at the end as the control strategy is implemented.

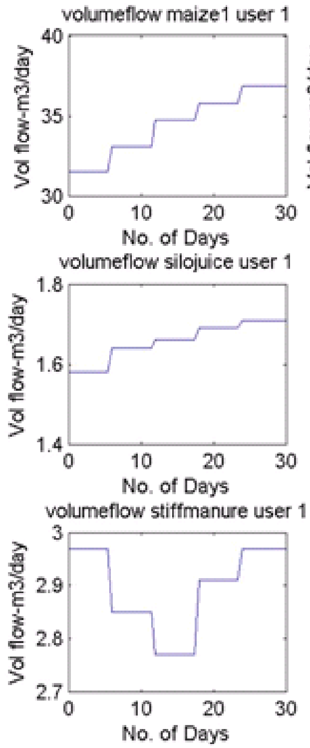


Figure 7-3- Variation of all Substrates with respect to the Simulation time. Parameters are Population size = 8, number of generations = 2, N = 5, T_c = 6 days, T_p = 100 days and δ = 6 days.

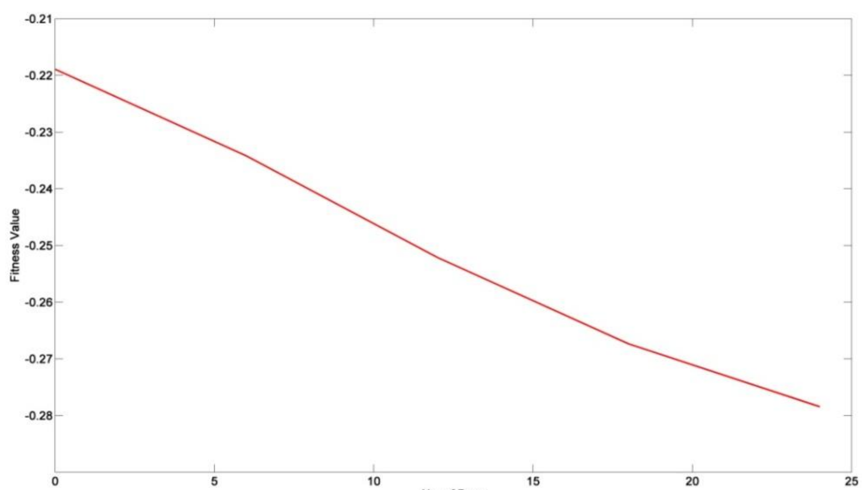


Figure 7-4- Variation of Fitness Value with respect to the simulation time for settings of Figure 7-3.

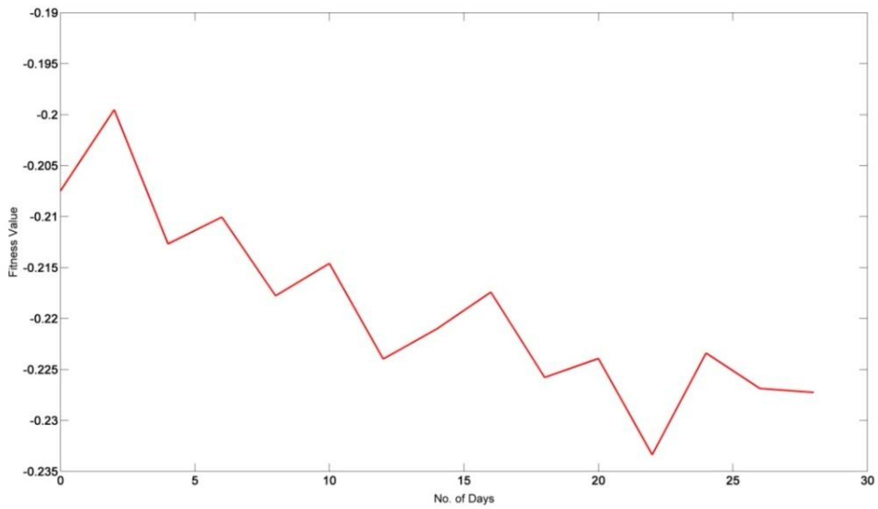


Figure 7-5- Variation of Fitness with respect to simulation time for the same settings as before but with number of steps =3 i.e. δ =2.

7.2.1. SAMPLING TIME AND FITNESS

Shown below in Figure 7-6 is the variation of fitness with respect to values of substrates. The x axis represents the substrate maize, the y axis represents Bull manure and the y axis represents Silo juice. The color bar represents the value of fitness. The more the color is towards the blue spectrum the better is the fitness value.

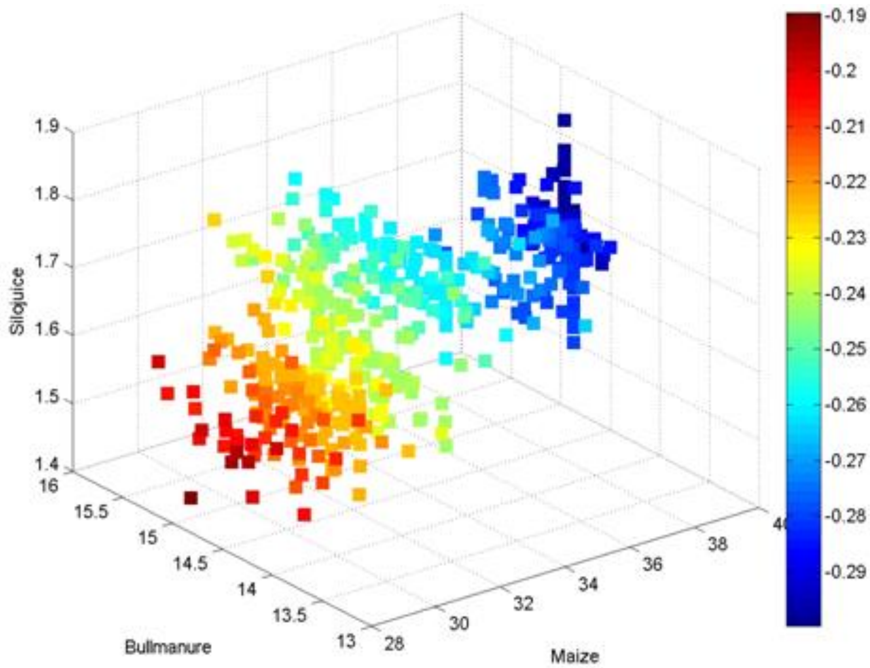


Figure 7-6 Variation of Maize, Bull manure and Silo juice with respect to fitness for Population size=20, No. of generation=6, T_c=6, T_p=100 and N=5, Number of steps =1.

In the above Figure 7-6 one can deduce that for higher value of maize the fitness value tends to get better. That is whenever the value of maize is high the biogas production has been high. And the other substrates do not have that much an effect on the fitness value.

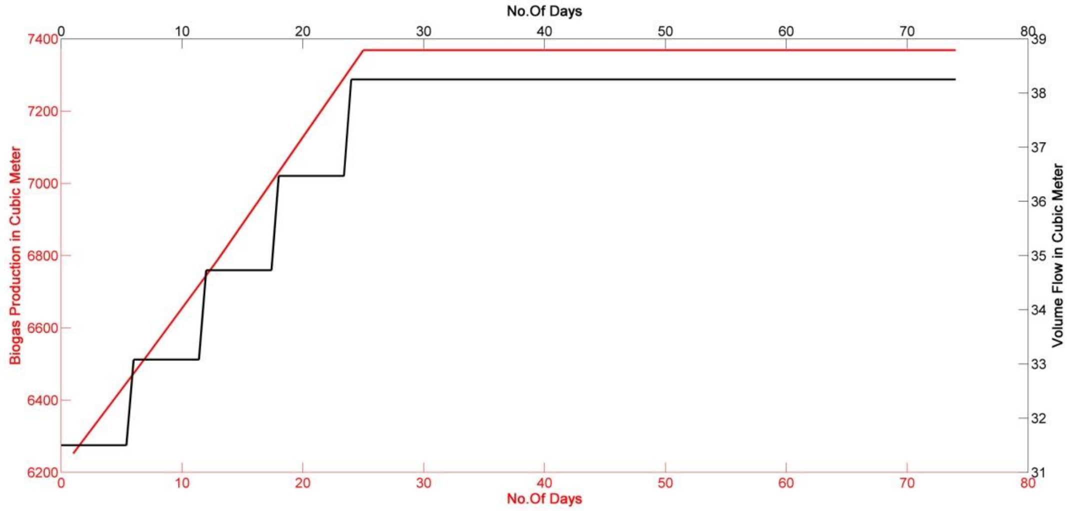


Figure 7-7- Comparison of Volume Flow of Maize and Biogas Production with respect to simulation time for the same setting as Figure 7-6.

The Figure 7-7 above shows the comparison between Biogas production and changes in volume flow of substrate: maize. The red line indicates the biogas production and the blue line indicates changes in maize. This proves the previous statement that the biogas production is greatly affected by Maize. There is a direct relation between Volume flows of Maize to the Biogas production until a certain limit (as defined in the parameters).

Shown below in Figure 7-8 is the variation of silo juice with the biogas production. The red line indicates the biogas production and the blue line indicates the variation in Silo juice in cubic meter. Even here there is a direct proportion between the change in substrate and the biogas production. However further analysis with different case reveal that maize has a greater effect than others.

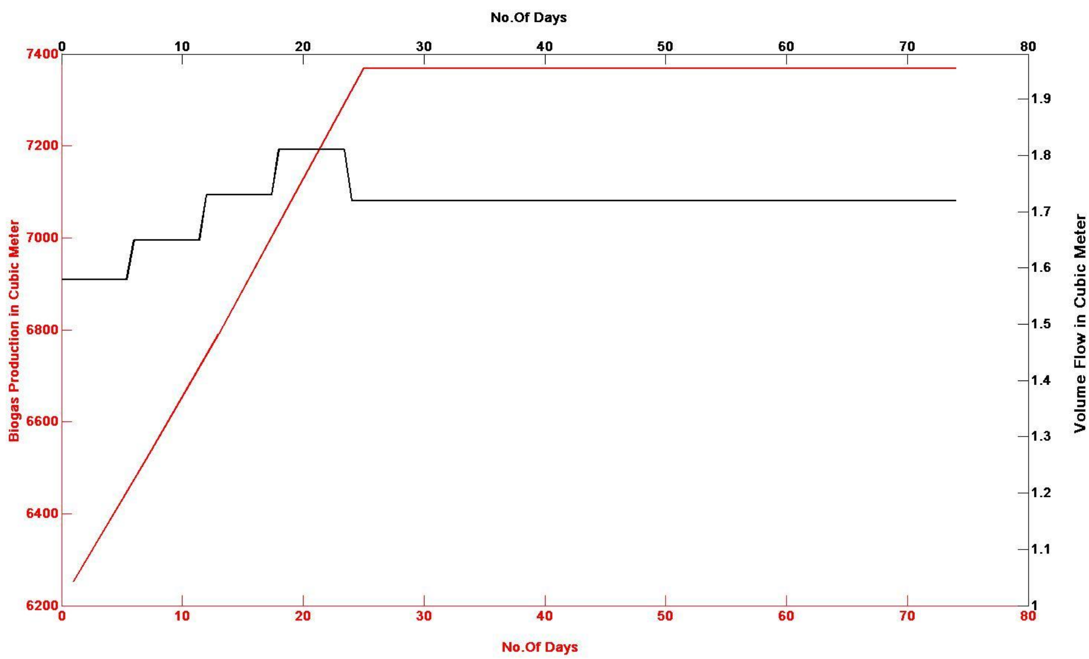


Figure 7-8- Comparison of Volume Flow of Silo Juice and Biogas Production with respect to simulation time for the same setting as Figure 7-6.

Shown below in the Figure 7-9 is the variation of 3 substrates namely Maize, Bull manure and Silo juice with fitness. The fitness is represented in colors. The more the color is towards the blue spectrum the better is the fitness. However this graph represents the first step among the two inputs calculated by the CMAES algorithm. We can observe that better fitness values are found at various other combinations other than that of higher values of maize.

However in Figure 7-10 we can see the second step calculated by the CMAES algorithm and this step seems to indicate a clear differentiation between higher fitness values and higher volume flows of Maize. In order to explain this variation from the figure representing number of steps =1 one must refer to (6.6) where an implementation of the two steps are shown. This can be explained due to the fact that during the optimization runs only the last step is applied to the plant for a considerably longer period i.e. the last step or value is applied to the plant from the control horizon until the prediction horizon.

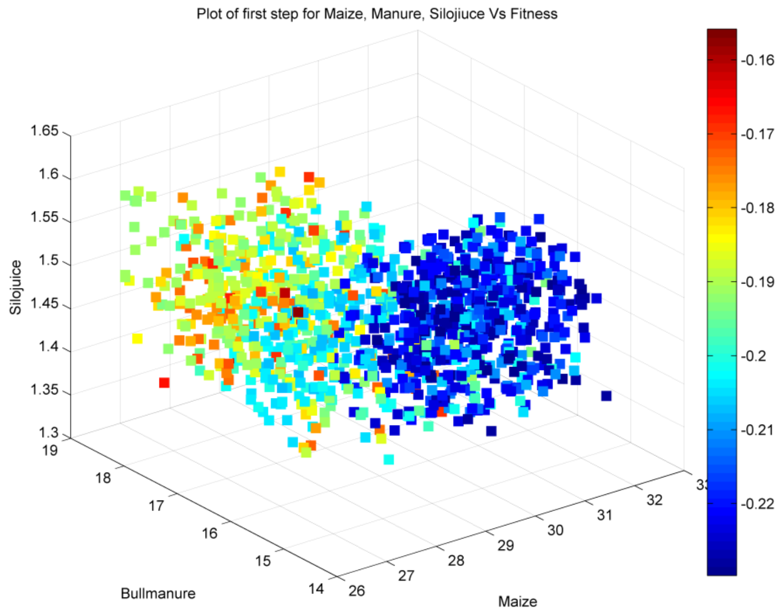


Figure 7-9- Variation of Fitness for the first step for substrate Maize, Bull Manure and Silo juice with respect to Fitness for Number of steps =2. All the other settings are the same.

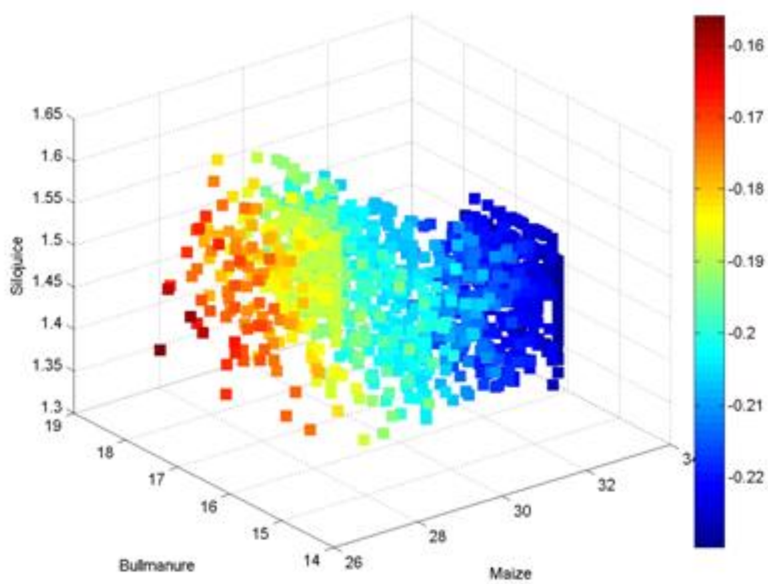


Figure 7-10- Variation for the second step for settings same as the previous Figure (7-10)

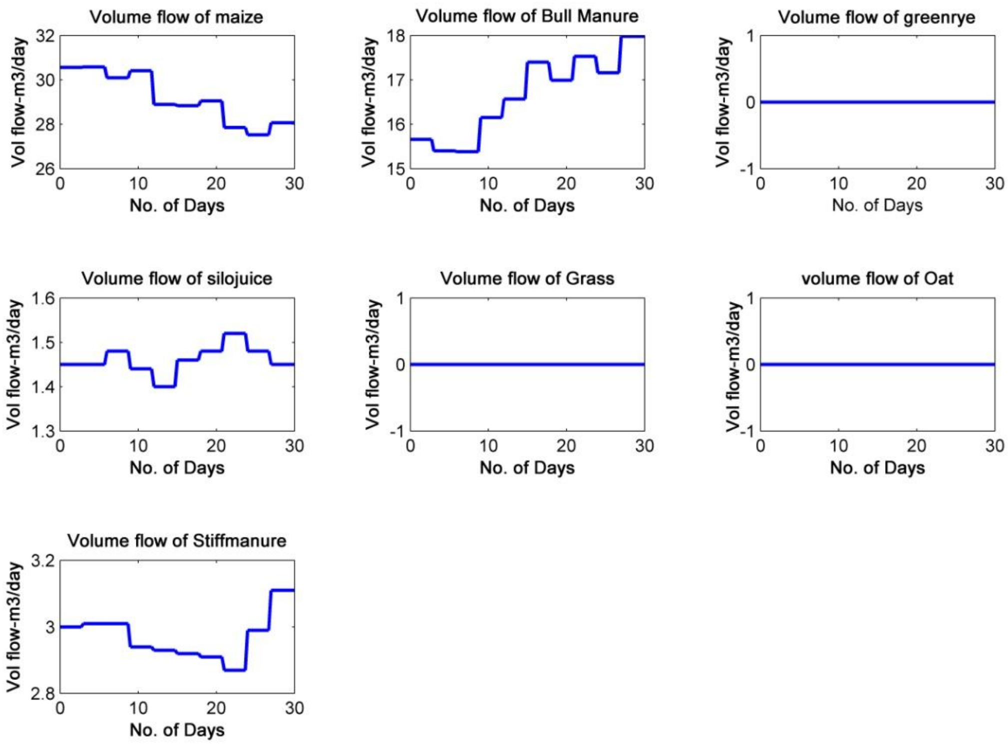


Figure 7-11- Variation of Each substrate with respect to Simulation time for Number of steps = 2.

Therefore one can say that for the second step the fitness should look much better i.e. the distinction between the higher values and lower values of fitness will be clear when the second step values are observed. However the real plant behavior differs significantly from the behavior of the simulated plant i.e. the fitness obtained during the optimization run is different from the fitness obtained from the real plant simulation as only the first step is applied. This is the reason why the cases with number of steps = 1 gives better fitness values than the cases with number of steps =2. The explanation of this problem is in 7.3.2

7.2.2. FITNESS AT THE END OF FINAL PREDICTION HORIZON

During the simulation of the plant the optimization algorithm stores the best input values for each iteration in ‘volumeflow_substrate_id_user_1’ (substrate_id denotes each substrate) files. In order to do know or analyze the simulation trajectory, we can use the user_1 files as input, the initial state before the NMPC implementation and simulate the model without the usage of optimization algorithm in order to get the trajectory of the output. This is equivalent to the previous method because the plant starts at the same initial state and the same inputs are applied at the same instants. One advantage of this simulation is that fact that we can obtain the value of the plant at the end of the prediction horizon of the last iteration.

One might question the significance of this value, since the prediction horizon is used inside the strategy and not explicitly on the plant. The values of fitness stored during the NMPC strategy are the fitness obtained over δ and not at the end of prediction horizon. Since the plant is simulated for a fixed period of time, during the final iteration the last value calculated by the optimization algorithm is applied and this value is kept constant until the end. If this value forces the plant to a stable state, then the value at the end of the simulation can be used, however if it does not lead to a stable state, one cannot conclude immediately that the setting is bad. There is also a possibility that with prolonged implementation of NMPC strategy the state might come back to the stable region. Therefore one of the best ways to check a scenario is at the end of the prediction horizon of the last iteration, especially if the number of steps involved is more than one. The optimization algorithm tries to bring the plant to a steady state at the end of the prediction horizon. So if we compare or analyze the values at the period corresponding to the prediction horizon of the last iteration, the results would provide a better insight.

7.3. PROBLEMS

7.3.1. INITIAL STATE PROBLEM

Color	Control Horizon	Prediction Horizon	Population size	Number of Generations	δ
Red	6	100	20	6	2
Blue	6	100	20	6	3
Green	6	100	20	6	6

Table 7-2- Settings used in figure Figure 7-12

The value of each substrate is set to a constant for a period of 200 days and then the values obtained from optimization are fed into the plant. These values are obtained from the usr_1 files in which during the NMPC strategy the optimal input values fed into the plants are stored. The constant values which are given initially to the plant are obtained from the steady state configuration of the plant. Therefore in accordance with the above theory the trajectory of any plant configuration for the first 200 days should be the same if the said changes are made to each of them. After the feeding the values calculated by the optimization algorithm the trajectory of the output may or may not differ. However the results obtained from the model show otherwise. Shown below in Figure 7-12 are the three scenarios for which simulations were done. The settings for the scenarios are shown in Table 7-2

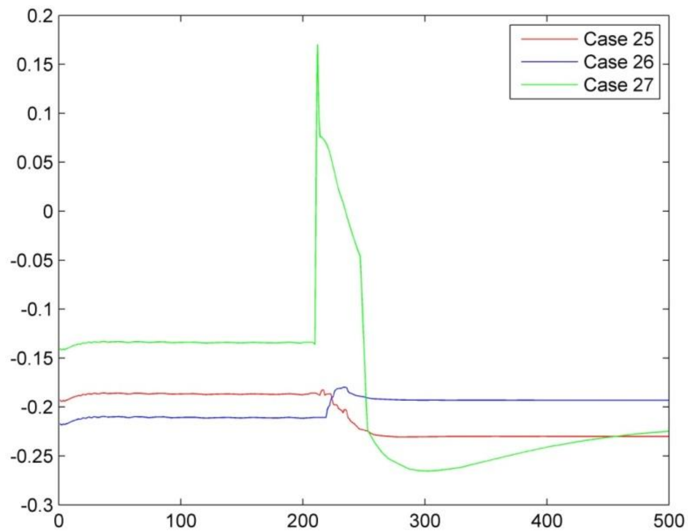


Figure 7-12- Simulation of the plant with the values calculates previously the optimization algorithm

As we can see from the above table and Figure 7-12 three cases with the same settings but with different sampling times are compared. Here we expect to find that the fitness for δ less than control horizon should be the best and the fitness for δ equal to the control horizon should be the worst. However for reasons stated in section 7.3.2 the plots show otherwise.

Also according to the previous explanation all the three scenarios should have the same trajectory for period of 200 days. However from the graph we can observe that none of the curves are similar to one another for the initial period of 200 days. Further analysis proved that there is a hidden influence of the user file on the plant which is yet to be deciphered exactly. In other words there is *bug* in the strategy which should be corrected so as to analyze the scenarios through this method.

However this bug will not affect the NMPC strategy because these are only written and not read by the optimization method. Therefore only the analysis is affected and not the experiment.

It is interesting to note that the hypothesis that a for the given population size and number of generations the fitness for the case with δ equal to the control horizon has the best fitness If we observe the green line which pertains to case where number of steps =1 the fitness trajectory starts at a fitness of around -0.13 and ends at ~= -0.225. Although the trajectory for the scenario with number of steps=3 has a lower fitness the starting fitness is very close. Therefore we can conclude that with number of steps =1 the fitness improves on a larger scale.

SOLUTION:

The solution to the above problem was found to be in a .dll file. The problem in the coding was corrected and as a result the initial state coincides correctly.

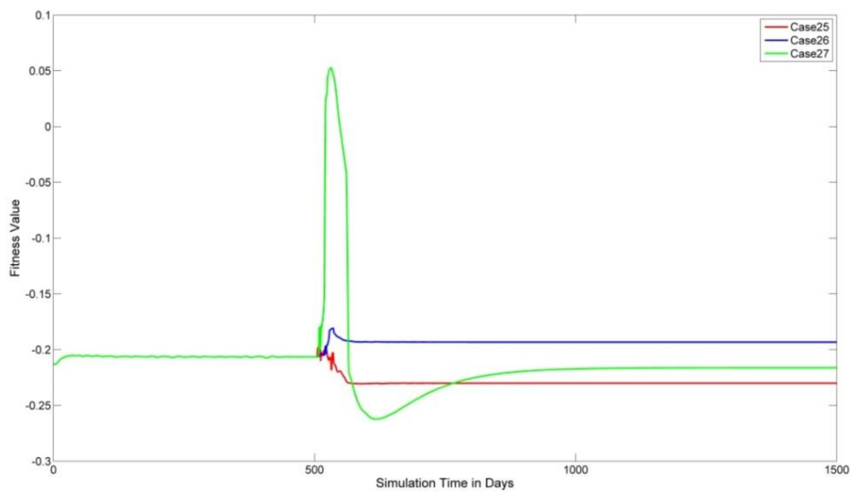


Figure 7-13 The Trajectory after the Correction in DLL file

7.3.2. THE OPTIMIZATION PROBLEM

Although the implementation of δ was successful the results do not show effective improvement. Shown below are comparisons of substrate changes employed by two scenarios. One with number of steps = 1 (Assume first scenario) and one with number of steps =2 (Assume second scenario) and number of steps =3 (Third Scenario). It is interesting to observe the path of the substrate maize for the first scenario which shows a steady increase in the volume flow (red). On the contrary the second scenario has an irregular trend. On the first look it might appear that the optimization algorithm has chosen the wrong values for the second scenario. However deeper analysis proves that the optimization algorithm is not entirely at fault.

During the second scenario i.e. for number of steps =2 only the first step is applied to the plant. During the optimization process the optimization algorithm continuously tries to improve the cost benefit ratio. In the previous implementation there was only one step possible (i.e. number of steps =1) in the control horizon. However in the second scenario there are two. Since during the optimization process the second step is applied to the model during the period between the control horizon and prediction horizon the importance of the second step is far higher than the first. Naturally the algorithm emphasizes a better value for the second step. The value of δ is too small when compared with the prediction horizon and the length of each step is equal to δ. Therefore the first step is used to reduce the cost of substrate i.e. a lower value of first step is used in combination with a higher value for the second step for Maize. Since the period for which this first step is applied is far less when compared to the second step, this tradeoff is acceptable by the optimization method.

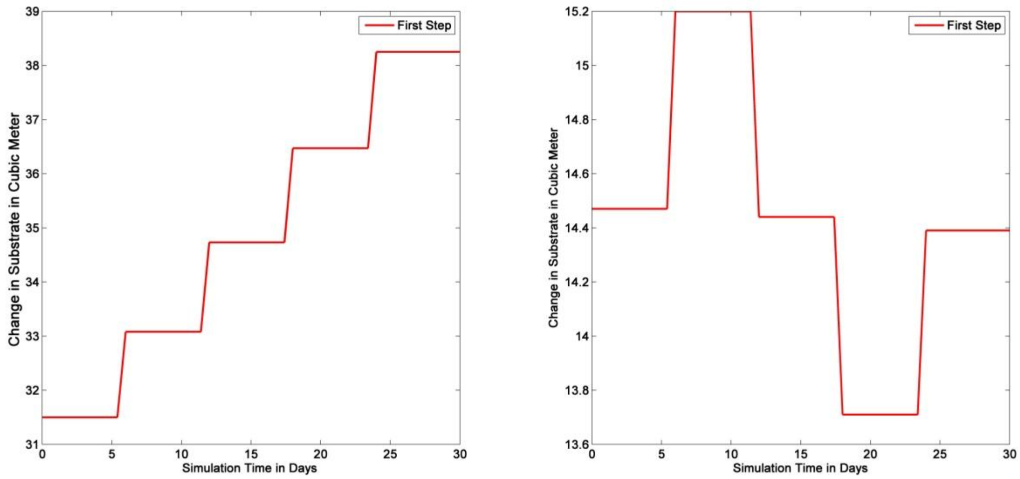


Figure 7-14- Substrate Vs Simulation Time for Maize (Left) and Bull Manure (right) for Number of Steps =1

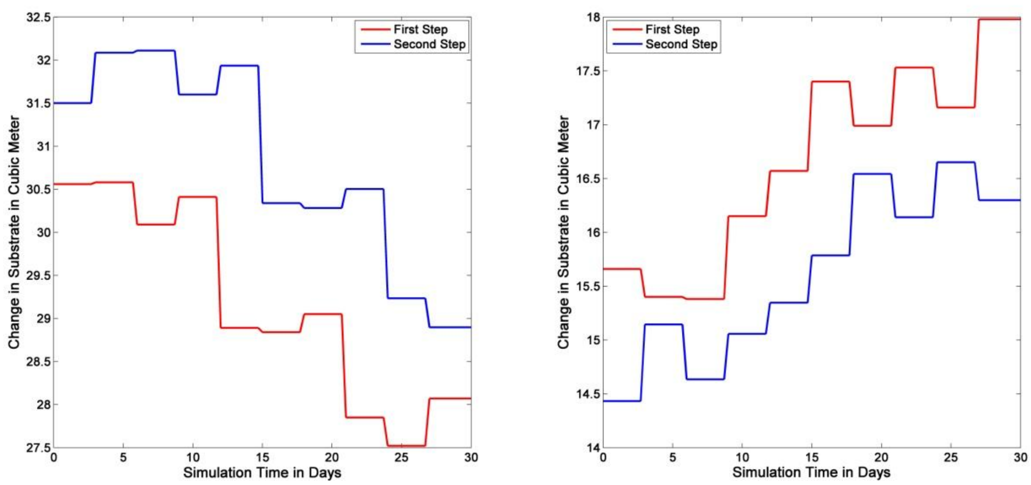


Figure 7-15- Substrate Vs Simulation Time for Maize (Left) and Bull Manure (right) for Number of Steps =2.

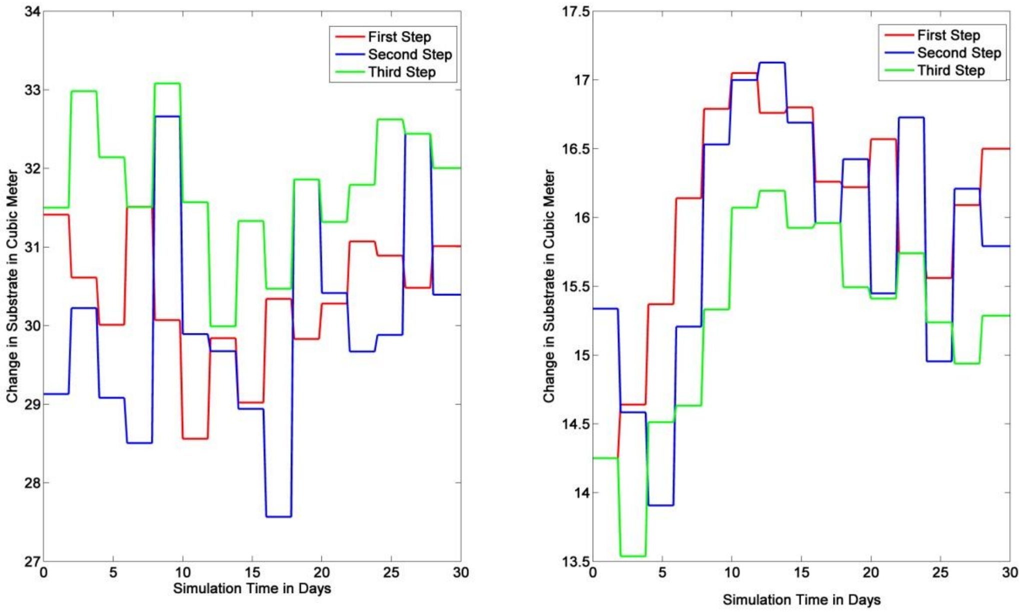


Figure 7-16- Substrate Vs Simulation Time for Maize (Left) and Bull Manure (right) for Number of Steps =3.

Shown above in Figure 7-14 is the volume flow of substrates Maize (left) and Bull manure (Right) for number of steps =1. Here we observe a steady increase in the Maize value and this corresponds to a higher biogas production. In the next figure that is Figure 7-15 the red curve of Maize (left) has a steady decreasing trend. However the second step which was calculated by the optimization algorithm is also shown and this is constantly above the first step.

One might argue that second step could have been even higher, however the constraints given to the plant restricts for that big a change. However the differences seen here correspond to 0.05% of the substrate value. Had the second step been applied to the plant the results would have been different. This is indicated clearly in the previous section where the effect of three substrates on fitness is shown. For the second step Figure 7-10 the fitness value is clearly better for higher values of maize. Because of the problem associated with this implementation, and the absence of a penalty term to control the gap, this strategy does not work for sampling time less than the Control horizon.

The Figure 7-16 shows the variation of substrate for number of steps=3 and clearly the variation is even higher. However, here also the third step is maintained at a higher value (green line). This problem would adversely affect all the following experiments and results.

7.4. EXPERIMENT 1

This experiment discusses on the effect of Control Horizon, Prediction horizon and Number of Steps on fitness. Table 7-3 contains the parameters which are used for this experiment. The parameters whose values have been bold and italicized are those which are varied. However the control horizon, prediction horizon and change value are set constant. The number of iterations depends on the value of δ i.e. the number of days for the NMPC strategy is applied to the plant is set to a constant value. The number of steps calculated in the control horizon is the modulus of control horizon and δ.

No.	*Group*	*Variable Name*	*Value*
1	NMPC Parameters	Control Horizon(days)	6
2		Prediction Horizon(days)	100
3		δ(days)	***2,3,6***
4		No. of iterations	15,10,5
5	Optimization Parameters	Population size	***8,15,20***
6		No. of Generations	***2,4,6***
7		Change	0.05

Table 7-3- Settings used for the First Experiment

7.4.1. SIMILAR FITNESS EXPECTATION:

As discussed in section 6.6 to achieve similar fitness values the number of simulations per calculated input should be the same. Shown below in Figure 7-17 and Figure 7-18 are the comparison between the fitness values of cases with similar number of simulations per input set. An input set is defined as the values to be calculated by the optimization algorithm and it includes all the substrates. For number of steps = 1 the input set is seven which is the number of substrates and for number of steps =2 there are double the number of inputs, since there is a second step which needs to be calculated. Therefore when the population size is 8, number of generations is 2 and the number of steps is one, then there are sixteen simulations done by the optimization algorithm in order to find optimal substrate feed values for the set of substrates. However if the number of steps is two then the optimization algorithm is required to compute inputs of 2 sets, each corresponding to each step. Therefore in order to achieve similar fitness the population size and number of generations should be increased accordingly. Ideally it should be similar but we tend to get minor differences between the cases, this is because of the problem explained in Section 7.3.2.

No. of Steps	**~16 Simulations per Input Set**			**~32 Simulations per Input Set**		
	Population Size	*Number of Generations*	*Case No*	*Population Size*	*Number of Generations*	*Case No*
1	8	2	3	8	4	12
2	8	4	11	15	4	14
	20	2	8			

Table 7-4 Tabulation comparing the cases which ought to have similar fitness

In the above table the cases where we expect similar are grouped according to the number of steps implemented. The cases with 16 simulations per input set are grouped in the left hand side and the cases with 32 simulations per input set are grouped in the right hand side. There are however more cases which have been simulated, the case seen above are the most relevant.

Shown in Figure 7-17 is the fitness trajectory for cases with 16 simulations per input set. All the scenarios are fed with the same initial state for an initial period of five hundred days and that is the reason why we see the same trajectory for all the scenarios in the first 500 days. Then the inputs computed by the optimization algorithm are applied. When we consider the end of the simulation time, the fitness values at this juncture for Case 3 and 11are the best and for Case 8 they are the worst. However these results denote the biogas production output where the last value computed by the optimization algorithm for the last iteration, is applied to the plant for a period of 900 days. Since the prediction horizon for these scenarios are one hundred days and the input computed to the plant are for a period of 30 days, it would be prudent to check the results at the end of the last prediction horizon (500+30+100= 630th day). From the graph it is clear that Case 3 has the best fitness (the conclusion is obtained on visual observation) and Case 8 has the worst. For number of steps =2 the fitness for case 8 gets worse. It is interesting to see that although Case 3 has best fitness, we observe an initial overshoot in fitness, however for the other two cases it is not so. This is because for number of steps =1 the system is forced to set a high input to the plant and as a result of this high value the fitness overshoots initially. This is not observed for the other two cases where number of steps = 2. Therefore we can conclude that for a sampling time less than the control horizon the overshoot is reduced. Although for reasons stated in section 7.3.2 the strategy does not work in the way we expect, the reduction in overshoot is a good sign.

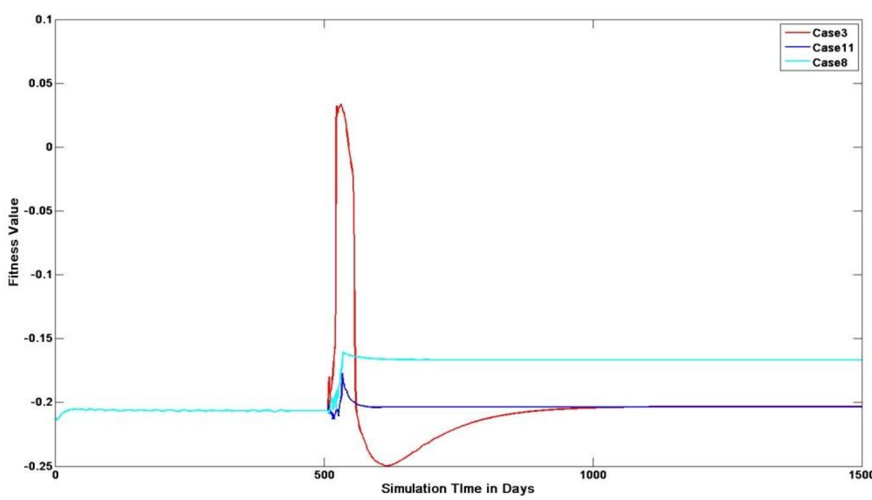


Figure 7-17- Fitness trajectory for scenarios with approximately 16 simulations per input set.

Similarly in Figure 7-18 we observe the variation of fitness for 32 simulations per input set. Case 12 is observed to have the best fitness and the Cases 14 has the worst. Here the number of simulation per input set has been increased to two fold; therefore we expect to observe better results.

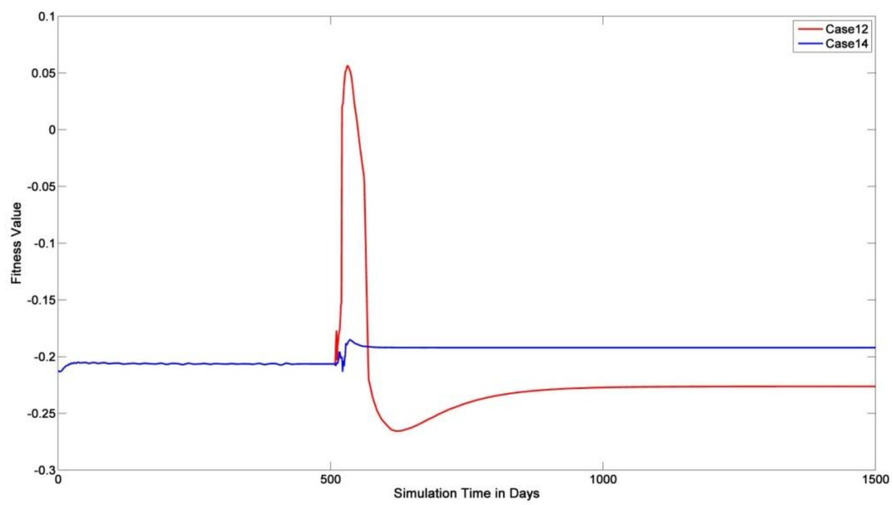


Figure 7-18- Fitness trajectory for scenarios with approximately 32 simulations per input set.

There is no change in the presence of an over shoot for number of steps =1 and the results for number of steps =2 seems to have a smooth transition, although the fitness gets worse. These results also support our previous conclusions.

7.4.2. FITNESS VARIATION WITH OPTIMIZATION PARAMETERS

Shown below in the Figure 7-19 is effect of sampling time (δ) and Number of generations, population size on Fitness. The color bar denotes the fitness values; the better the fitness the more the color is towards the blue spectrum. The x axes denote Number of generations, y axis denotes δ and the z axis denotes fitness. In Figure 7-19 we can see a general trend of fitness getting better for number of steps equal to one and as the number of steps increases to 2(δ=3) the fitness gets worse (proving the previous hypothesis in 7.3.2). This indicates again that the fitness gets better as the number of steps decreases.

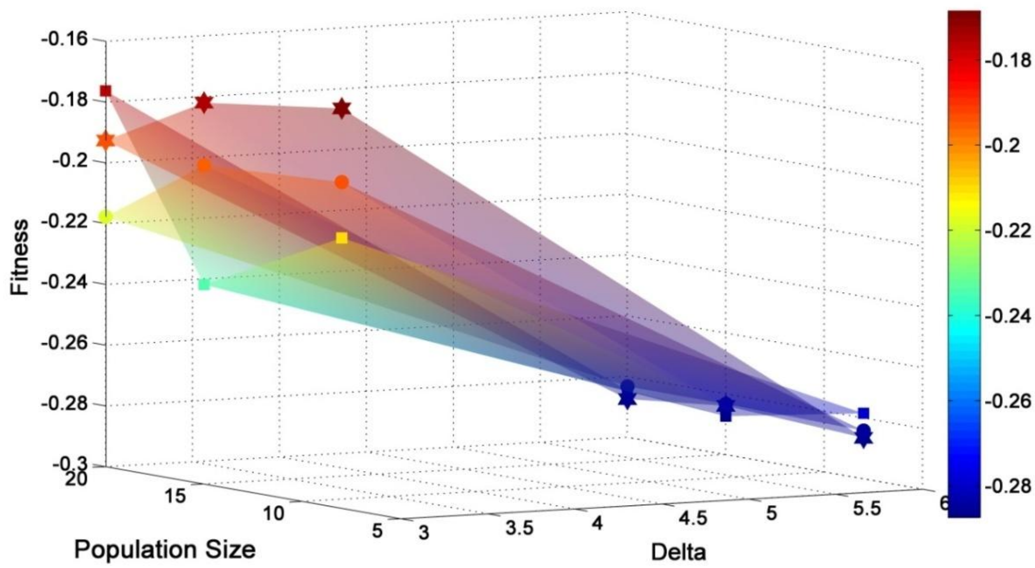


Figure 7-19- Effect of Optimization Parameters on Fitness values.

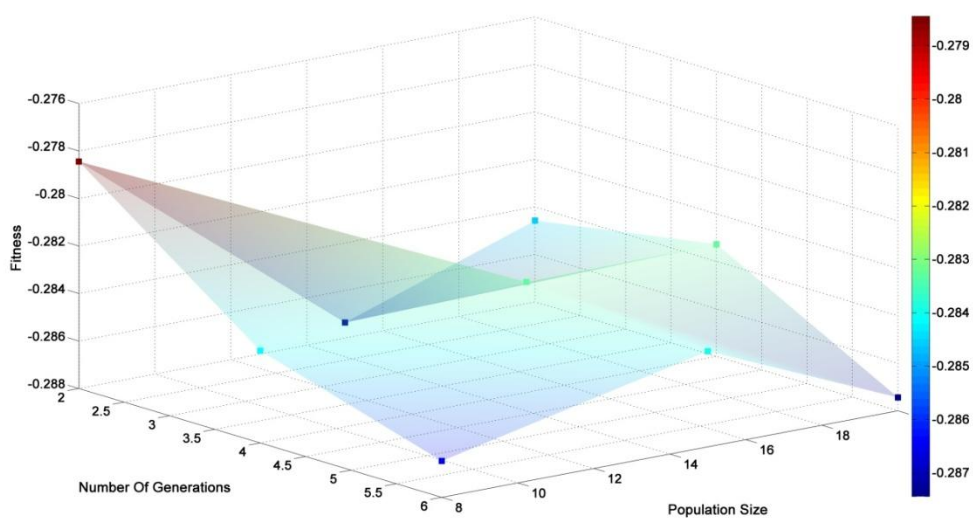


Figure 7-20 Variation of Fitness Value for number of steps =1

For number of steps =1 the fitness does not vary much. For number of steps =1 the fitness values are much closer to each other. In Figure 7-19 we observe a trend in fitness, decreasing as the number of generation's increases. However the previous result is contrary to this.

However for number of steps = 2 the fitness value varies greatly, especially for number of generations. As the number of generations increases we expect a better fitness. This trend is seen in all the Values of population size for the number of generations= 4 and 6. However the surface denoting the number of generations = 2 follows the trend only for population size= 8,15. For population size of 20 the result gets worse than the other surfaces. This indicates that as the optimization efficiency increases the difference in gap between the steps calculated by the optimization algorithm also increases, thereby producing bad fitness values (refer section THE OPTIMIZATION PROBLEM). To conclude the setup works better for number of steps =1 when compared with number of steps =2.

7.5. EXPERIMENT 2

This experiment shows the variation of Fitness value with respect to the variation in Control Horizon, Prediction horizon and δ.

No.	*Group*	*Variable Name*	*Value*
1	NMPC Parameters	Control Horizon (days)	***6,12,18***
2		Prediction Horizon(days)	***100,150,200***
3		δ(days)	***2,3,6***
4		No. of steps	1,2,3
5		No. of iterations	36/δ
6	Optimization Parameters	Population size	20
7		No. of Generations	2
8		Change	0.05

Table 7-5- Settings used in Experiment 2

The above table contains the parameters which are used for this experiment. The parameters whose values have been bold and italicized are those which are varied. However the population size, number of generations and change value are set constant. The number of iterations depends on the value of δ i.e. the number of days for the NMPC strategy is applied to the plant is set to a constant value. The number of steps calculated in the control horizon is the modulus of control horizon and δ.

7.5.1. FITNESS WITH CONTROL PARAMETERS

The Plot below is a 3 Dimensional plot showing the variations of fitness with respect to control horizon, prediction horizon and number of steps =1. Here the variable δ is not used since many values of δ are used in accordance with the variation of the control horizon. Therefore the number of steps: which is the modulus of Control horizon and δ is used, since they are varied between 1 and 2.

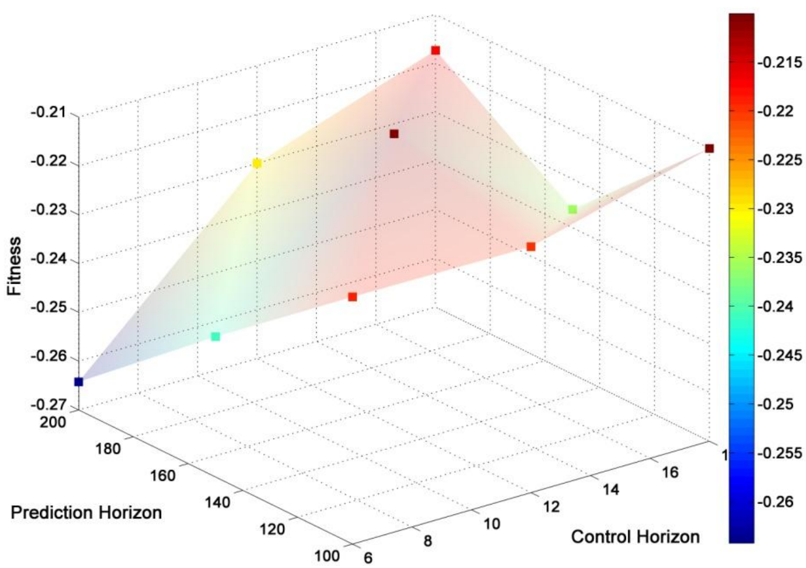


Figure 7-21 Variation of Fitness with control Parameters for Number of steps =1

In the above Figure 7-21 the color bar represents the fitness value obtained after a 1500 day simulation of the model using the sequences of input values computed by the optimization algorithm and the more the color is towards the blue spectrum the better is the fitness value. The x axis represents the Control horizon, the y axis represents Prediction horizon and z axis represents the fitness. There is a trend in fitness getting better as the prediction horizon increases and control horizon decreases. The odd value in fitness for prediction horizon of 150 and control horizon of 18 can be excluded as there is a general trend of decrease in all the other cases.

The overall trend in the graph is obvious although the case for prediction Horizon of 150 and control horizon of 18 does not correspond to the trend. There is a general decrease in fitness as the prediction horizon increases and control horizon decreases. This is an expected behavior. As control horizon decreases the controller must feed the changes in volume flow of the substrates more quickly. As the prediction horizon increases the system has more time to attain steady state.

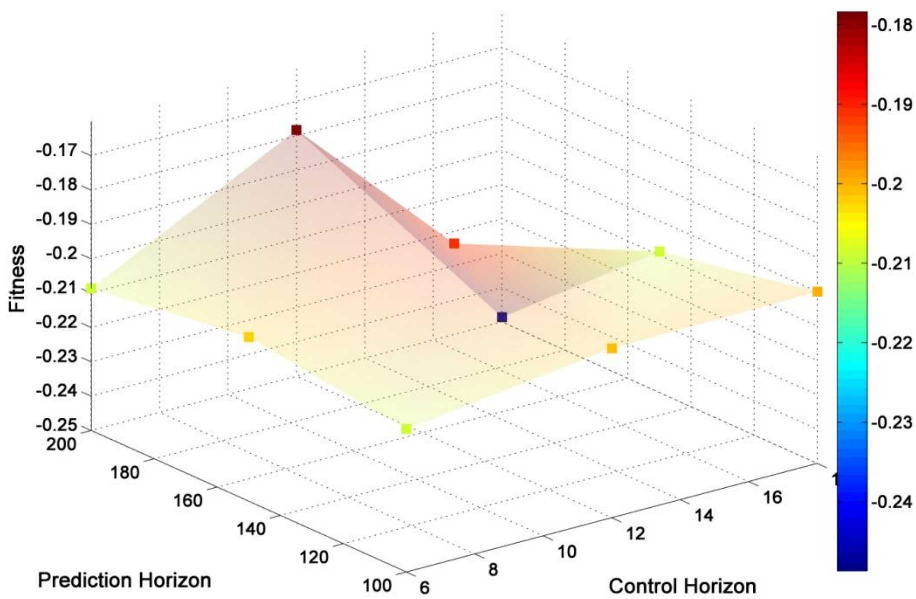


Figure 7-22- Variation of fitness with prediction horizon, control horizon for number of steps = 2

Shown above in Figure 7-22 is the variation of fitness values with prediction horizon and control horizon. As we can see there is no trend that can be determined from this figure. Therefore we can only conclude that for number of steps = 2 the strategy does not work properly as for the reasons explained in the section 7.3.2.

To conclude, this experiment proves that for number of steps =1 the NMPC strategy works fine however for number of steps =2 we get ambiguous results as we see no definite trend.

7.6. EXPERIMENT 3

This section deals with the reliability of the model. The model is simulated with the same settings for 3 times and checked whether the same results are obtained during each of the simulation. This is important because it should be certain that the model behaves in the same way irrespective of the time i.e. a time invariant system. Shown below in Table 7-6 are the settings for the simulations conducted during this experiment.

No.	Parameter	Value
1	Control Horizon(days)	6
2	Prediction Horizon(days)	100
3	δ(days)	***3,6***
4	Number Of Iterations	***10,5***
5	Population Size	8
6	Number of Generations	2

Table 7-6 Setting Used in Experiment 3

It is necessary to find out whether the system behaves in the same way for various number of steps calculated in the control horizon i.e. for various values of δ but for the same values of control horizon.

7.6.1. RELIABILITY FOR NUMBER OF STEPS =1

In the table shown above the control horizon is kept constant and δ the sampling time is varied, this way the number of steps calculated by the optimization algorithm is varied.

The Table 7-7 below shows the fitness obtained during the simulation runs. The fitness obtained is different during each run; this is because the CMAES algorithm is a Non deterministic Evolutionary algorithm i.e. each time it is not certain that the algorithm proceeds in the same way, however the difference in fitness values are very small.

Sl. no	Population Size	Number of Generations	δ(days)	Control Horizon(days)	Prediction Horizon(days)	Total Simulation time(days)	Fitness
1	8	2	6	6	100	30	-0.2350
							-0.2305
							-0.2340

Table 7-7- Settings for number of step=1 for the reliability test

Shown below in table is the error analysis of the fitness values obtained. The box plot of the error values are shown in figure and the upper bound and lower bound of the error values are denoted by the top and lower edges of the rectangle. The y axis denotes the Fitness value. There is only one outlier which can be disregarded. The mean value is approximately -0.2331.

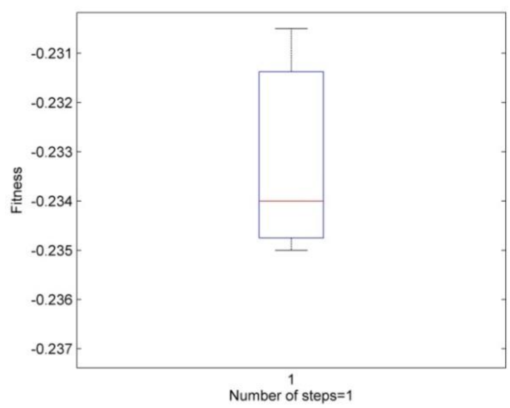


Figure 7-23- Box Plot for the Reliability Test for number of steps = 1

The Figure 7-23 shows the variation of maize for 3 cases with the same settings. The x axis denotes the Simulation period and the y axis represents the fitness. Since the influence of maize on fitness is very high the trajectory follows a similar path.

However the effect of the non deterministic method is compensated by the combination of various substrates. This in turn ensures the reliability of the strategy. Shown below are the changes in all the substrates' volume flow in Figure 7-24.

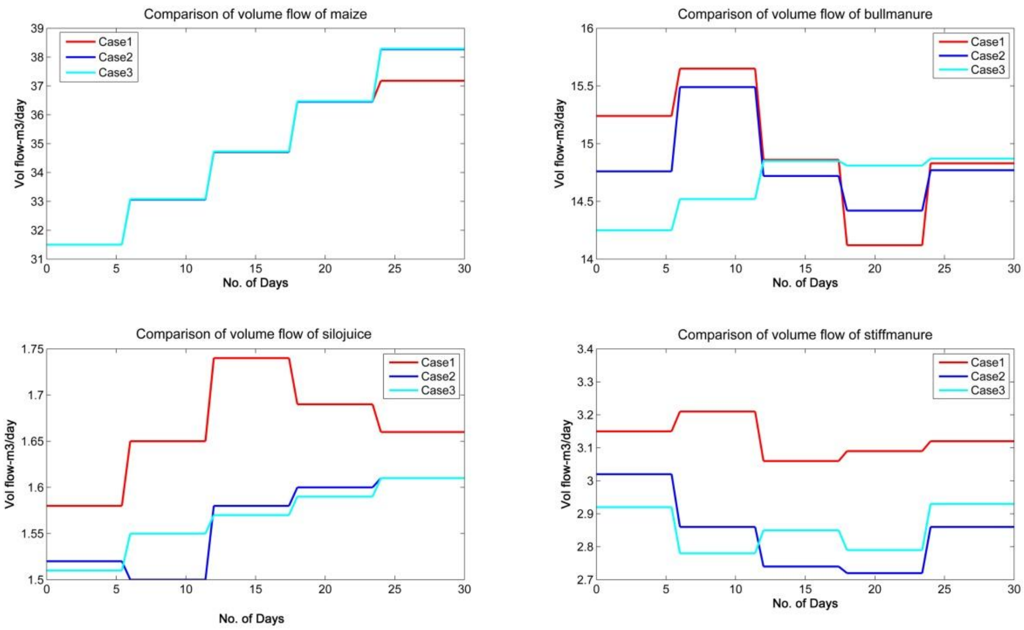


Figure 7-24 Substrate Changes Comparison between three runs with same settings and number of steps =1. Shown above are Maize (Top Left), Bull Manure (Top Right), Silo juice (Bottom Left) and Stiff Manure (Bottom right).

In all of the above figures the number of steps corresponds to the number of iterations and the total simulation time corresponds to the product of number of iterations and δ value. In this case δ = 6 days which is denoted by a change for every 6 days and number of iterations = 5 which is denoted by the number of steps. Therefore the simulation period is 30 days. The trend in maize is similar for all the three trials and this is denoted in the box plot shown above. This indicates that the reliability for this setting is good.

7.6.2. RELIABILITY FOR NUMBER OF STEPS = 2

The Table 7-8 below shows the settings of simulations done for two steps. The box plot is shown in Figure 7-25. The reliability for this case is not as good as the reliability for the scenario with number of steps=1. The problems explained in Section 7.3.2, the increasing complexity with the increase in the number of steps, the inadequate population size and number of generations affect adversely the reliability.

Sl. no	Population Size	Number of Generations	δ	Control Horizon	Prediction Horizon	Total Simulation time	Fitness
1	8	2	6	6	100	30	-0.2202
							-0.1804
							-0.2372

Table 7-8- Settings for number of steps = 2 for the reliability Test

The Figure 7-25 shows the box plot show the variation of each substrate between different runs for number of steps =2 and The Figure 7-25 shows the variation of maize (Top right), Silo juice (top left), bull manure (Bottom left) and stiff manure (Bottom Right).

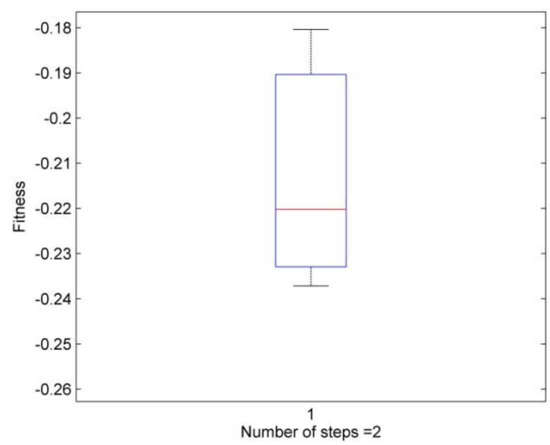


Figure 7-25- Box Plot for the Reliability Test for number of steps = 2

Optimization algorithm plays a vital role in deciding the final result of the strategy. The decision of the best values to be applied to the plant is taken by the algorithm. The CMAES algorithm used in this thesis continuously tries to increase the biogas production and cost effectiveness. However higher biogas production does not always mean higher profits. The costs of substrates are also taken into account. Therefore a tradeoff exists between the cost and biogas production and this is defined in the fitness function. Shown below in the diagram is the normal implementation of NMPC strategy with only one step between the control horizons. Here the sampling time δ and control horizon are one and the same. In order to improve or equip the strategy we introduce a new variable δ.

The main reason of introducing δ is to approximate the step sizes. In figure we can see the implementation and expectation of the result when δ is included. The ambiguous results obtained in section 7.4 and section 7.5 can be explained due to the fact that CMAES is not aware of the fact that only the first step is applied to the plant and the rest.

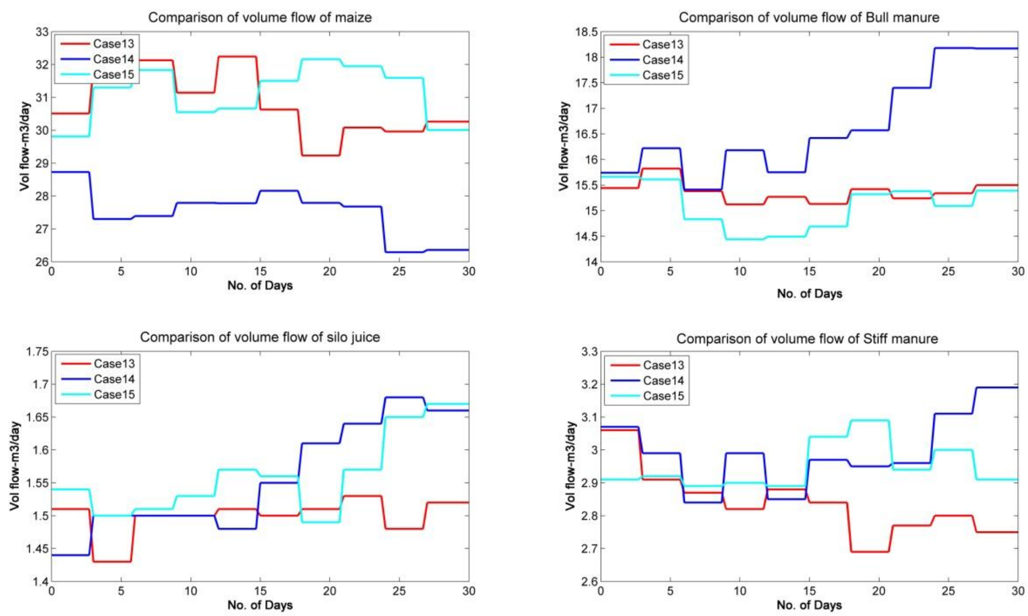


Figure 7-26- Substrate Changes Comparison between three runs with same settings and number of steps =2. Shown above are Maize (Top Left), Bull Manure (Top Right), Silo Juice (Bottom Left) and Stiff Manure (Bottom right).

When CMAES computes, for example say two steps within the control horizon i.e. δ is half as control horizon, CMAES tries to improve the fitness function by reducing the cost of substrates required. As explained in the previous section (7.3.2) the last step before the control horizon is the step which is maintained constant throughout the period between control and prediction horizon. Obviously this step would have higher influence that the first step since the period in which the first step is applied is much less when compared with the prediction horizon.

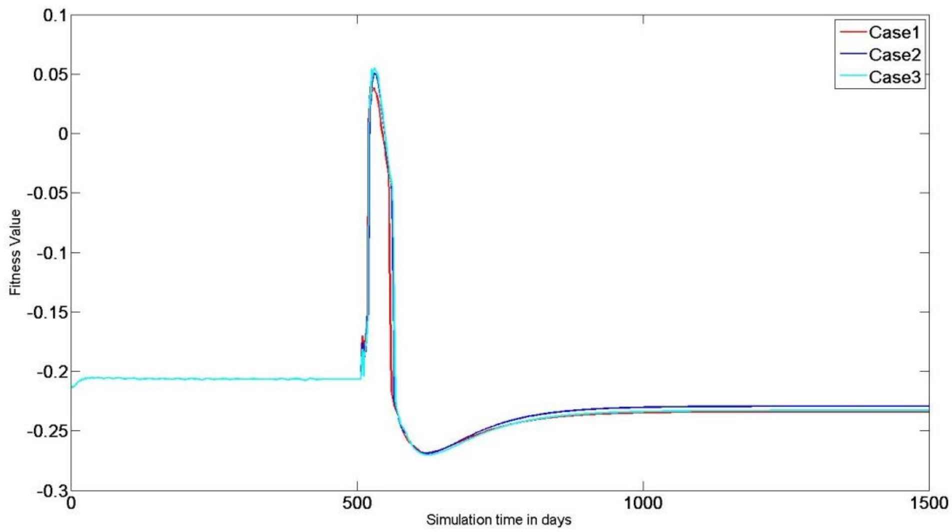


Figure 7-27 Variation in Fitness trajectory for the same settings for all the three cases for number of steps =1

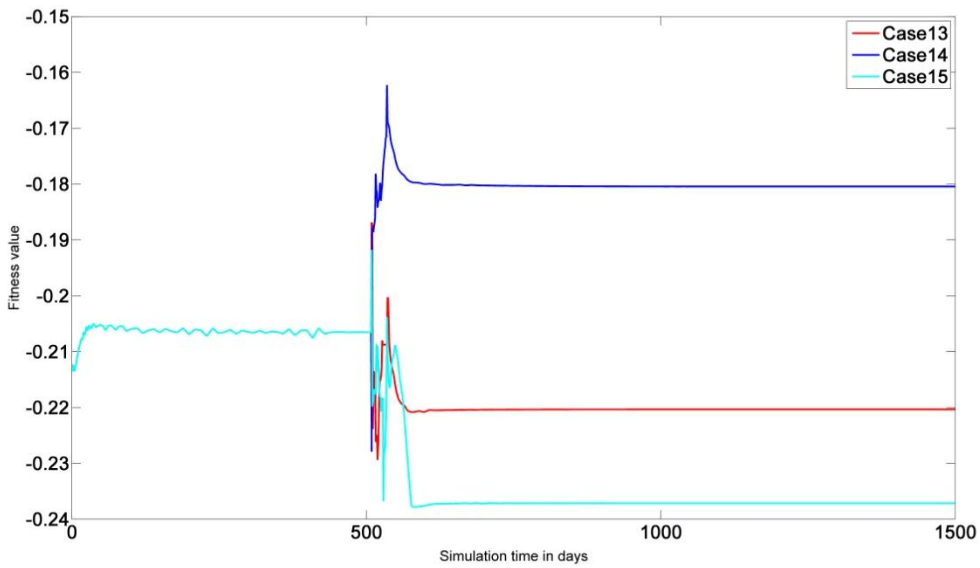


Figure 7-28 Variation in Fitness values for the same settings for all the three cases for number of steps =2

From the above two figures we can conclude, this experiment shows that the reliability of the strategy decreases as the number of steps increases

7.7. SUMMARY AND SOLUTIONS

The setup was tested and analyzed for the variation of fitness value with respect to the optimization and control parameters. It was found that for number of steps = 1 the setup provides a better result when compared the case with number of steps = 2. This is shown clearly with experiment I and II where we get ambiguous results as the number of steps increases. However the idea in itself was not wrong, the implementation had a glitch. These problems were analyzed and in order to solve these problems two solutions are discussed. The principle for which is explained below.

7.7.1. FIRST METHOD

One solution can be the introduction of a penalty term which effectively controls the gap between the first and the second step i.e. a variable which controls the amount of variation present between the steps. Since the final step has the most influence on the simulated plant, it cannot be changed. Therefore the only solution which would be adopted by the optimization algorithm will be to move the first step closer to the second step. This is indicated in the figure below, it merely indicates the concept rather than the actual implementation.

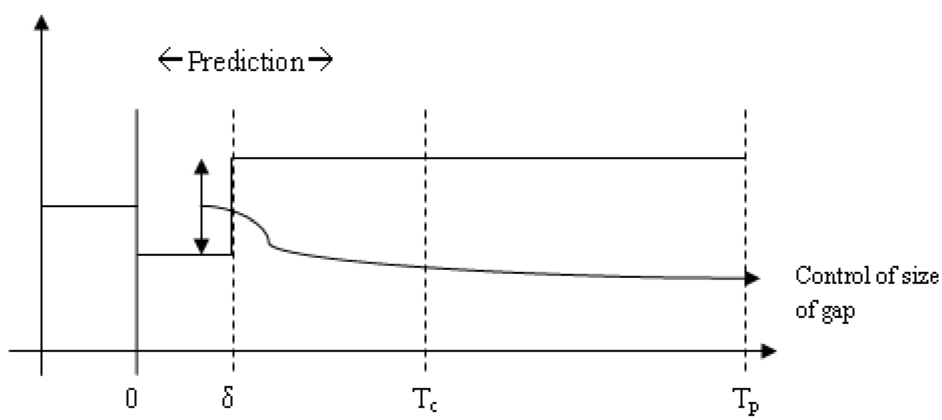


Figure 7-29- First Solution to control the Gap between two steps

7.7.2. SECOND METHOD

Another solution would be to implement a smoothening mechanism i.e. to calculate a single step over the control horizon but implement this step in parts over δ. For example if number of steps required are 3 then we split the calculated change in substrate into three and implement each part over δ refer Figure 7-30 .This way the curve would move towards the desired area.

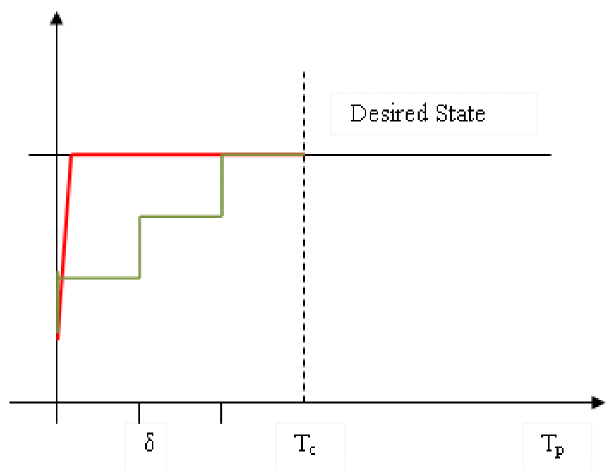


Figure 7-30- Second solution- To introduce an approximation mechanism to smoothen the step.

However when analyzing the above two methods there are advantages and disadvantages. While the second method encourages simplicity in computing power and implementation the former involves the same complexity as before. However with the first method the optimization method has an opportunity to decrease the substrate for a short period (i.e. δ) and then increase that is, suppose we use the second method there can be only a decrease or an increase in the substrate from the current value, so if we use the first method we can decrease the substrate below the current value (for the first step) and then increase the value.

The first method is required when there is a sudden disturbance in the plant. Therefore it is possible to get better results with implementation of either of the two methods. To conclude the NMPC strategy works better for number of steps = 1, however it is necessary in the practical implementation that we implement more steps. Therefore it is necessary that further research be carried out in order for the betterment of this method.

CONCLUSION

This thesis can be seen as a test phase for the biogas toolbox. The thesis shows that NMPC methodology can be implemented to a biogas plant. The toolbox has been analyzed and tested under various conditions in order to understand the response of the NMPC strategy.

The complexities and difficulties in the analysis indicate that the choosing of substrate mixes was of paramount importance for the effective operation of the plant. The inclusion of a fitness function and its role in evaluating the cost, inclusion of environmental factors and production was proven to be very important. This means that not only can biogas plant be optimized for more production and costs; it can also be made to consider environmental factors. This is very useful considering the importance placed on pollution and green energy.

In the experiments above the results for sampling time equal to control horizon are very good. This proves that the NMPC strategy is a viable solution to a Biogas plant. The experiments conducted reveal that without decoupling of the control horizon and sampling time, there is an initial over shoot in the fitness. However in using the sampling time as a separate parameter the overshoot is considerably reduced. This is one of the expected benefits of the decoupling process.

The reliability test conducted in the last experiment effectively concluded that the reliability of the strategy is better for sampling time equal to control horizon and worse for sampling time less than control horizon. This problem was also analyzed and explained in detail . The presence of ambiguous results for higher values of number of steps is explained in the previous sections. However, experiencing these problems first hand during this thesis, some suggestions are already given. To implement and test the viability of these solutions can be an interesting research topic in the future.

Also this thesis stresses the importance of computation power employed. Since the ADM1 is a highly complex model and NMPC is a highly complex strategy a considerable time is taken for computation. A faster optimization run is necessary in order to make this strategy realizable. Also in reality one might want to implement higher number of steps. For number of steps =2, population size of 20 and number of generations =4 the time taken was 5 hours. This should be reduced in the future. All the above analysis proves that with the opportunity of further research and development the NMPC strategy will be a viable and practical solution to a biogas plant.

BIBLIOGRAPHY

[1] Allgöwer, F., Findeisen, R., & Nagy, Z. K. (2004). Nonlinear Model Predictive Control:From Theory to application. *35* (3).

[2] Batstone, D. J., Keller, J., Angelidaki, I., Kalyuzhnyi, S. V., Pavlostathis, S. G., Rozzi, A., et al. (2002). The IWA Anaerobic Digestion Model No 1 (ADM1). *Water Science and Technology* (pp. 65-73). St. Lucia: IWA Publishing.

[3] *Biogas Production Process*. (2009). (C. PaterVis.com) Retrieved 10 1, 2012, from http://www.patervis.com: http://www.patervis.com/process.html

[4] Findeisen, R., & Allgöwer, F. (2005). *An Introduction to Nonlinear Model Predictive Control.* Prague: 16th IFAC World Congress.

[5] Gaida, D., Brito, A. L., Wolf, C., Bäck, T., Bongards, M., & McLoone, S. (2011). Optimal Control of Biogas Plants using Nonlinear Model Predictive Control. Trinity College Dublin: IEEE Editorial.

[6] Gaida, D., Wolf, C., Baeck, T., & Bongards, M. (2012). Nonlinear Model Predictive Substrate Feed Control of Biogas Plants. *IEEE Editorial* .

[7] Gaida, D., Wolf, C., Meyer, C., Stuhlsatz, A., Lippel, J., Bäck, T., et al. (2011). State Estimation for Anaerobic Digesters using the ADM1. *A journal on International Association on Water pollution Research.*

[8] Hansen, N. (2011). *The CMA Evolution Strategy: A Tutorial.*

[9] Hoshimura, K. (2007). *Covariance Matrix Adaptation Evolution Strategy for constrained optimization problem.* Kyoto University, Applied Mathematics and Physics. Kyoto: Graduate School of Informatics.

[10] *http://en.wikipedia.org.* (n.d.). Retrieved 10 12, 2012, from Multivariate normal distribution: http://en.wikipedia.org/wiki/Multivariate_normal_distribution

[11] Kossmann, W., & Pönitz, U. (2011). Biogas Digest. *1*.

[12] Luis Sousa Brito, A. (2011). *Optimization of full-scale Biogas Plants using Nonlinear Model Predictive Control(Master Thesis).* Gummersbach.

[13] *Naskeo Environnement*. (2009). Retrieved 10 12, 12, from http://www.biogaz-europe.com/: http://www.biogas-renewable-energy.info/

[14] Nicolao, G. D., Magni, L., & Scattolini, R. (2000). Stability and Robustness of non linear receding horizon control. 3-23.

[15] *Parts of a Biogas plant or Bio-gas Generator*. (2012). (Bright Hub Inc) Retrieved 9 30, 2012, from http://www.brighthub.com: http://www.brighthub.com/environment/renewable-energy/articles/23762.aspx

[16] *Silo.* (n.d.). Retrieved 10 12, 2012, from http://en.wikipedia.org: http://en.wikipedia.org/wiki/Silo

[17] *Types of Biogas Digesters and Plants*. (2012, 8 30). (Energypedia) Retrieved 9 30, 2012, from https://energypedia.info: https://energypedia.info/index.php/Types_of_Biogas_Digesters_and_Plants

[18] *various-types-biogas-plants*. (n.d.). (Indiastudychannel) Retrieved August 28, 2012, from http://www.indiastudychannel.com: http://www.indiastudychannel.com/resources/117400-various-types-biogas-plants.aspx

[19] Wolf, C., McLoone, S., & Bongards, M. (2009, 12). Biogas Plant Control and Optimization Using Computational Intelligence Methods. *Oldenbourg Wissenschaftsverlag, (DOI 10.1524/auto.2009.0809),* , pp. 638-49.

[20] Флюид . Fluid. (2002). *Different Types of Biogas plants*. (WordPress & the Atahualpa Theme by BytesForAll) Retrieved 9 30, 2012, from http://www.fluid-biogas.com: http://www.fluid-biogas.com/?page_id=131&lang=en

APPENDIX A

Shown below is table which explains the purpose of each script used in the NMPC strategy as in the order in which they are called in the main file.

No.	Function/Scripts		Explanation
1	NMPC_InitializationInputParams		Initialize Input parameters
2	load_biogas_mat_files		Load all the initialized parameters(plant's data)
3	NMPC_createDigesterStateMinMax		Load the last initial state from File.
4	NMPC_load_SubstrateFlow NMPC_load_FermenterFlow		Sets the 'lb' and 'ub' of the substrate and Fermenter to the current volume flow and creates the 'Volumeflow_substrate /Fermenter_const' files.
5	biogasM.optimization.popBiogas		Define the population and parameters for optimization
6	NMPC_ctrl_strgy_change		This returns the change value for Fermenter and substrate flow.
7	NMPC_ctrl_strgy_SubstrateFlow_MinMax NMPC_ctrl_strgy_FermenterFlow_MinMax		This modifies the substrate and Fermenter flow in accordance with the change value
8	NMPC_simBiogasPlantExtended		Simulate the plant for a period of 500 days to achieve steady state.
	Begin *FOR* Loop		
9	1.	findOptimalEquilibrium	Find an optimal input using the specified optimization method.
	2.	NMPC_simBiogasPlantExtended	Apply the calculated input over a period of δ
	3.	NMPC_save_ctrl_strgy_FermenterFlow NMPC_save_ctrl_strgy_SubstrateFlow	This sets the substrate and Fermenter flow to the file and sets the substrate and Fermenter min/max to the current value.

	4.	NMPC_ctrl_strgy_FermenterFlow_MinMax NMPC_ctrl_strgy_SubstrateFlow_MinMax	The value of substrate and Fermenter flow in accordance with change
	End *FOR* **Loop**		
10	NMPC_simBiogasPlantExtended		Simulate the plant for 500 days to extrapolate the plant stability
11	NMPC_save_SimOptmimData		Save all necessary data.

APPENDIX B

Shown below is the settings used for the first experiment. The first experiment involves the variation of fitness value with respect to Population size, Number of Generations and Sampling time.

No.	T_p	T_c	No. of steps	Population size	No. of Generations	Change	Fitness Value
1.	100	6	3	30	2	0.05	-0.2069175
2.			2				-0.20502721
3.			1				-0.20487693
4.		12	3				-0.18739801
5.			2				-0.26364147
6.			1				-0.21507109
7.		18	3				-0.23437093
8.			2				-0.16681376
9.			1				-0.21397907
10.	150	6	3				-0.15158212
11.			2				-0.20355057
12.			1				-0.22761347
13.		12	3				-0.15770984
14.			2				-0.1921228
15.			1				-0.21799986
16.		18	3				-0.18539528
17.			2				-0.24629128
18.			1				-0.21949243
19.	200	6	3				-0.18586195
20.			2				-0.16945653
21.			1				-0.21340174

22.		12	3				-0.19686716
23.			2				-0.19116991
24.			1				-0.2090693
25.		18	3				-0.23000682
26.			2				-0.19321112
27.			1				-0.21751032

APPENDIX C

The table contains the settings used for the second experiment. The second experiment deals with the variation in Fitness value with respect to Control Horizon, Prediction Horizon and Sampling time.

No.	T_p	T_c	No. of steps	Population size	No. of Generations	Change	Fitness Value
1.	100	6	3	8	2	0.05	-0.22093284
2.			2				-0.20828169
3.			1				-0.22082549
4.			3	15			-0.24154729
5.			2				-0.20042107
6.			1				-0.22103434
7.			3	20			-0.19345462
8.			2				-0.20000791
9.			1				-0.2114154
10			3	8	4		-0.14570692
11			2				-0.20220896
12			1				-0.24253304
13			3	15			-0.17621969
14			2				-0.19314689
15			1				-0.21183895
16			3	20			-0.21003725
17			2				-0.20890148
18			1				-0.23725602
19			3	8	8		-0.2150619
20			2				-0.20851087
21			1				-0.26535203

22			3	15			-0.1917891
23			2				-0.18131813
24			1				-0.2313361
25			3	20			-0.20730186
26			2				-0.24869865
27			1				-0.21872078

DECLARATION

I hereby declare that I have independently written the thesis submitted by me. I have indicated all my passages, whose exact wording or analogy have been derived from published or unpublished works of others. I have indicated all sources and other means used for this thesis. Neither the exact content nor the major segments of the thesis have yet been submitted to an examination authority.

Gummersbach