Performance Analysis of First-Order Plus Dead-Time Processes Using Generalized Predictive Control

A Thesis submitted for partial fulfillment of the requirements
for the degree of Master of Technology in
Electronics and Insrumentation Engineering

Submitted by
Jahagirdar Ankush Chandrakant
(213EC3215)

Under the Guidance of
Prof. Tarun Kumar Dan
Associate Professor

Department of Electronics & Communication Engineering,
National Institute of Technology, Rourkela-769008 (Odisha)
May 2015

Department of Electronics & Communication Engineering,
National Institute of Technology, Rourkela-769008,
Odisha, India

CERTIFICATE

This is to certify that the thesis entitled, "**Performance Analysis of First-Order Plus Dead-Time Processes Using Generalized Predictive Control**", submitted by **Jahagirdar Ankush Chandrakant** in partial fulfillment of the requirements for the award of Master of Technology Degree in **Electronics and Instrumentation Engineering** at National Institute of Technology, Rourkela is an authentic work carried out by him under my supervision. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university/ institute for the award of any Degree or Diploma.

Place: Rouekela
Date:

Prof. Tarun Kumar Dan
Associate Professor,
Dept. of Electronics & Communication Engg.
NIT Rourkela

ACKNOWLEDGEMENT

"Success is not final, failure is not fatal: it is the courage to continue that counts."

-Winston S. Churchill

So true, that this line encouraged me throughout the project. I want to express my sincere thanks and gratitude to **Prof. Tarun Kumar Dan**, who supervised and guided me on every step. Without his guidance, I could never have completed my project in this stipulated period of time. His support, guidance, motivation and encouragement resulted in this accomplishment. I would also like to thank all the professors and members of the Department of Electronics and Communication Engineering for their generous help in various ways for the completion of the thesis. I also extend my thanks to my parents and the fellow students for their friendly co-operation.

Jahagirdar Ankush Chandrakant

ABSTRACT

Generalized Predictive Control (GPC) is a part of a family of Model Predictive Controllers, which predicts the future output using the concept of Receding Horizon. A cost function is then optimized based on the predicted error and optimized control sequence is calculated. The standard GPC algorithm, proposed in 1987, finds many applications. It has several modifications and extensions to incorporate adaptiveness and constraints. Though GPC is inherently discrete, current research trend is development of continuous GPC algorithms. GPC controller has four tuning parameters, control weight, minimum and maximum prediction horizon and control horizon. This thesis tries to address the effect of these tuning parameters on the performance of a First-Order Plus Dead-Time (FOPDT) process. The performance was measured in terms of three important dynamic characteristics- settling time, rise time and peak overshoot. Lower values of control weight and higher values of prediction horizon gave better performance with regards to these criteria. The observations from these simulations were used as guidelines for tuning the GPC controller for a tank level control system. Single tank system is an example of an FOPDT process. Due to the non-linearity in the process model, it was found to perform better under adaptive GPC (AGPC) algorithm. Simulations were also performed to take the input constraints into consideration. Then, disturbance rejection and robustness behaviour of the adaptive constrained GPC (ACGPC) controller was studied. It was found that, GPC gives many configurable parameters to deal with different control problems. The performance was found to be better than conventional PI controller.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Model Predictive Control (MPC) is a family of controllers that contain following common concepts:

- Output of the process is predicted using a mathematical model, known as prediction model.
- The reference trajectory is known beforehand.
- A control sequence is calculated by solving an optimization problem. Constraints can also be incorporated at this step.
- Receding strategy i.e. at each sampling instant, only the first control signal of the calculated sequence is applied.

1.1 Principle of MPC

Figure 1.1 shows the principle of MPC, which is based on receding horizon strategy.

1. Future output ($\hat{y}$) are predicted over some time length (N), konwn as the prediction horizon. This is achieved using a process model, which is generally locally linearized.
2. The control sequence (u) is obtained as a solution of a constrained optimization problem, which consists of a cost function. The cost function comprises of predicted output sequence ($\hat{y}$), future reference trajectory (w).
3. Although the control action is predicted for some horizon in future, only the first element of the sequence is applied to the process. At next sampling time the procedure is repeated. This is known as the Receding Horizon concept.

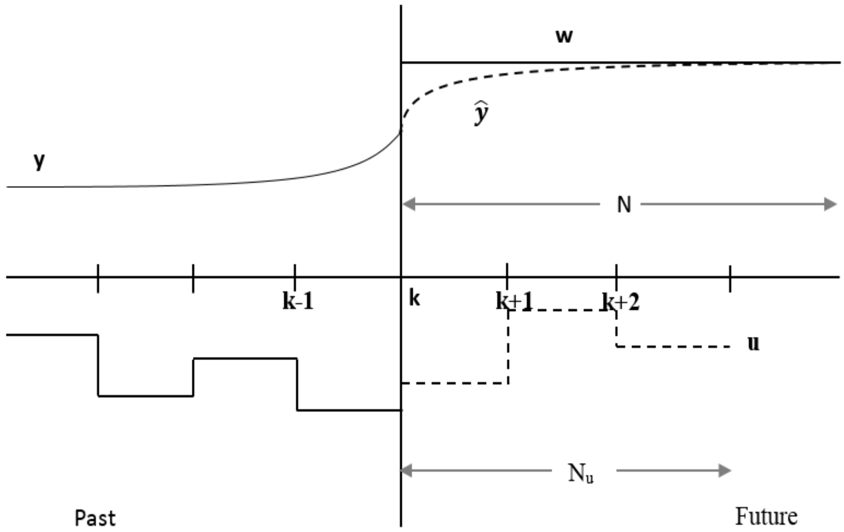

Figure 1.1: Concept of Receding Horizon

1.2 Prediction Model

The process model used in a predictive control should capture the process dynamics, should be preferably linear, and should be able to predict future output based on past and present input-output data. The model which is used for Generalized Predictive Control is based on the discrete-time transfer function of the process.

$$A(z^{-1})y(n) = B(z^{-1})u(n-1) \tag{1.1}$$

This representation is also applicable to unstable processes, unlike other representations like step response model.
Incorporating the noise in eq(6.7) we obtain a CARIMA model.

$$A(z^{-1})y(n) = B(z^{-1})u(n-1) + C(z^{-1})\frac{\xi(n)}{\Delta} \tag{1.2}$$

which can be written as,

$$\Delta A(z^{-1})y(n) = B(z^{-1})\Delta u(n-1) + C(z^{-1})\xi(n) \tag{1.3}$$

Here, $\xi(n)$ represents white noise and $C(z^{-1})$ represents a design polynomial.

1.3 Optimization Function

The standard optimization cost function used in GPC contains quadratic terms of predicted future error (between predicted output and future reference) and control increments calculated over a finite horizon in the future.

$$J = \sum_{i=N_1}^{N_2} [P(z^{-1})\hat{y}(n+i) - w(n+i)]^2 + \lambda \sum_{i=1}^{N_u} [\Delta u(n+i-1)]^2 \qquad (1.4)$$

N_1 is called minimum prediction horizon, N_2 is called as maximum prediction horizon, and N_u is known as control horizon. The time interval from N_1 to N_2 represent the future time interval over which the output is supposed to follow the reference trajectory. If d is the time delay of the process, the value of N_2 should be at least $d+1$. N_2 should be large enough to cover the important dynamics of the process response curve. Value of N_u helps to reduce the computational load and thus the time taken to calculate the control sequence. λ is known as control weight. More the value of λ, lesser the control effort is allowed to be.

1.4 Advantages

GPC, and MPC in general, has following advantages when compared with convention control strategies:

- MPC is inherently discrete in nature.
- Predictive control algorithms are robust and the constraints can be easily incorporated.
- They usually outperform conventional PID controllers and are applicable to non-minimum phase, open-loop unstable, time delay and multivariable processes.

1.5 Disadvantages

Besides the advantages above, GPC has following disadvantages:

- GPC is computationally very demanding. It requires several complex mathematical operations like matrix formation, inversion and multiplication.
- Due to computational complexity, GPC can't be applied to hard real-time processes, which have stringent time restrictions.
- It is difficult to implement the GPC algorithm using an embedded processor, as the memory requirement may be higher.

Chapter 2

Literature Review

Introduced in 1987 by Clarke, Mohtadi and Tuffs, GPC is a widely used member of MPC family. Initial work including standard GPC algorithm, its adaptive nature, application of constraints was done by Clarke and his colleagues. Recent trends in the topic show different extensions, modifications and applications of standard GPC algorithm. Also different configurations like cascade GPC, feed-forward GPC are now being studied.

- D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control-Part I. The basic algorithm," *Automatica*, vol. 23, no. 2, pp. 137-148, Mar. 1987. [1]
  This paper proposes the standard GPC algorithm. Solving Diophantine equation in two unknown polynomials is an important part of the algorithm. This paper gives an iterative method for solving such equations.

- D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized Predictive Control-Part II Extensions and interpretations," *Automatica*, vol. 23, no. 2, pp. 149-160, Mar. 1987.
  This is second part of the previous paper and talks about self-tuning performance of GPC. The simulation examples show the effect of under and over-parameterization. It proposes extensions of GPC to deal with the noise part.

- D. W. Clarke and C. Mohtadi, "Properties of generalized predictive control," *Automatica*, vol. 25, no. 6, pp. 859-875, Nov. 1989. [2]
  Clarke and Mohtadi talk about different properties of Generalized Predictive Control in this paper. The special focus is on the stability based on choices of prediction horizon.

- C. Bordons and E. F. Camacho, "A generalized predictive controller for a wide class of industrial processes," *IEEE Transactions on Control System Technology*, vol. 6, no. 3, pp. 372-387, May 1998. [3]
  This paper proposes an approximated method of the standard GPC algorithm so as to reduce the computational time required. The method is applicable only to first-order and dead-time processes. The simulation results from the paper show that the approximated method takes very less

computation time and the performance is comparable with the standard GPC.

- J. L. Garriga and M. Soroush, "Model Predictive Control Tuning Methods: A Review," *Industrial & Engineering Chemistry Research*, vol. 49, pp.3505-3515, 2010. [4]
  It gives review of various tuning methods proposed upto 2010 for tuning of model predictive controllers, including DMC and GPC.

Various modifications, extensions and analysis studies of basic GPC algorithm are available in literature.

- M. Xu, S. Li, and W. Cai, "Cascade generalized predictive control strategy for boiler drum level," *ISA Trans.*, vol. 44, pp. 399411, 2005. [5]

- A. Krolikowski and D. Jerzy, "Self-tuning generalized predictive control with input constraints," *Int. J. Appl. Math. Comput. Sci.*, vol. 11, no. 2, pp. 459479, 2001. [6]

- P. Sarhadi, K. Salahshoor, and A. Khaki-Sedigh, "Robustness analysis and tuning of generalized predictive control using frequency domain approaches, *Appl. Math. Model.*, vol. 36, no. 12, pp. 61676185, 2012. [7]

- M. Short, "Input Constrained Adaptaive GPC for Simple Industrial Plant," *18th Int. Conf. Autom. Comput.*, 2012. [8]

Other than this fundamental literature, which gives the basic theory on GPC, there are various papers which show performance of GPC in specific applications.

- P. Alkorta, O. Barambones, J. Cortajarena, and A. Zubizarreta, "Efficient multivariable generalized predictive control for sensorless induction motor drives," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 9, pp. 5126-5134, 2014. [9]

- H. Hapoglu, S. Karacan, Y. Cabbar, C. E. Balas, and M. Alpbaz, "Application of Multivariable Generalized Predictive Control To a Packed Distillation Column," *Chemical Engineering Communications*, vol. 174, no. 1, pp. 61-84, Aug. 1999. [10]

- Kay-Soon Low, Koon-Yong Chiun, and Keck-Voon Ling, "Evaluating generalized predictive control for a brushless DC drive," *IEEE Transactions on Power Electronics*, vol. 13, no. 6, pp. 1191-1198, 1998.

- Y. Shoukry, M. Watheq El-Kharashi, and S. Hammad, "An embedded implementation of the Generalized Predictive Control algorithm applied to automotive active suspension systems," *Elsevier, Computer & Electronics Engineering*, vol. 39, no. 2, pp. 512-529, Feb. 2013.

Chapter 3

Generalized Predictive Control

In its most simple form, a GPC controller represents a linear SISO controller. Consideration of disturbances is not taken into account. The model with which a process is described is known as CARIMA (Continuous Auto-Regressive Integrated Moving Average). This model is based on the discrete-time transfer function of the system.

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{n_b} z^{-n_b}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{n_a} z^{-n_a}} \tag{3.1}$$

If the system is affected with a dead-time, the first elements of the polynomial $B(z^{-1})$ are equal to zero.

3.1 Prediction

The CARIMA model mentioned in eq1.2 can be used to predict the future output of the process over certain time interval, known as Prediction Horizon. In order to obtain j-step ahead prediction, following Diophantine equation[1] with $C(z^{-1}) = 1$ needs to be solved.

$$1 = E_j(z^{-1})A(z^{-1})\Delta + z^{-j}F_j(z^{-1}) \tag{3.2}$$

Let,

$$\tilde{A}(z^{-1}) = A(z^{-1})\Delta = 1 + (1 - a_1)z^{-1} + (a_1 - a_2)z^{-2} + \cdots + a_{n_a} z^{-n_a} \tag{3.3}$$

$$1 = E_j(z^{-1})\tilde{A}(z^{-1}) + z^{-j}F_j(z^{-1}) \tag{3.4}$$

where, j takes values from N_1 to N_2. Clarke et al. [1] proposed a method to solve this Diophantine equation. Using this method, the polynomials E_j and F_j can be calculated iteratively. Once the these unknown polynomials were obtained, the system output was predicted using the following equation.

$$\begin{aligned}\hat{y}(n+j) = {} & E_j(z^{-1})B(z^{-1})\Delta u(n+j-1) \\ & + E_j(z^{-1})\xi(n+j) + F_j(z^{-1})y(n)\end{aligned}$$

Defining $G_j(z^{-1}) = E_j(z^{-1})B(z^{-1})$,

$$\begin{aligned}\hat{y}(n+j) &= G_j(z^{-1})\Delta u(n+j-1) \\ &+E_j(z^{-1})\xi(n+j) + F_j(z^{-1})y(n)\end{aligned}$$

Let $g_0, g_1, \cdots, g_{N-1}$ be the coefficients of $G_j(z^{-1})$. Then the predicted output can be divided into future and past values as below:

$$\hat{y}(n+j) = G\tilde{u} + f \tag{3.5}$$

where

$$G = \begin{bmatrix} g_0 & 0 & 0 & \cdots & 0 \\ g_1 & g_0 & 0 & \cdots & 0 \\ g_2 & g_1 & g_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{N_2-1} & g_{N_2-2} & g_{N_2-3} & \cdots & g_0 \end{bmatrix} \tag{3.6}$$

Here, N_1 is assumed to be 1. If higher value of N_1 needs to be considered, then the initial rows of the matrix G will become zero. Also $\tilde{u}$ is the calculated control sequence.

$$\tilde{u} = [\Delta u(n), \Delta u(n+1), \cdots, \Delta u(n+N_2-1)]^T \tag{3.7}$$

$$f = F_j(z^{-1})y(n) + G'(z^{-1})\Delta u(n-1) \tag{3.8}$$

where, $G'(z^{-1})$ is part of $G_j(z^{-1})$ which will consider only past values of control input u. The term $G\tilde{u}$ is called as the forced response, as it depends on future values of the control input and f is called as the free response, as it depends on only past values of the control input.

3.2 Optimization

The optimization function used in GPC is usually quadratic. This function is optimized with respect to the required control sequence. The values of the future control increments thus obtained are the optimal values. The equation for the cost function is as follows:

$$J = \sum_{i=N_1}^{N_2} [\hat{y}(n+i) - w(n+i)]^2 + \lambda \sum_{i=1}^{N_u} [\Delta u(n+i-1)]^2 \tag{3.9}$$

$$J = (G\tilde{u} + f - w)^T (G\tilde{u} + f - w) + \lambda \tilde{u}^T \tilde{u} \tag{3.10}$$

Taking the partial derivative of J with respect to $\tilde{u}$ and equating it to zero,

$$\begin{aligned}\frac{\partial J}{\partial \tilde{u}} &= 2G^T(G\tilde{u} + f - w) + 2\lambda\tilde{u} \\ 2(G^TG + \lambda I)\tilde{u} &= 2(w - f)G \\ \tilde{u} &= (G^TG + \lambda I)^{-1}(w - f)^T G\end{aligned}$$

Thus,

$$\tilde{u} = H^{-1}g \tag{3.11}$$

where, $H^{-1} = G^TG + \lambda I$ and $g = (f - w)^TG$.

This equation gives the whole trajectory of the future control increments. Only the present value, i.e. first row of matrix u is applied to the process and the whole algorithm is computed at time $t + 1$. This strategy is called as Receding Horizon Strategy.

Chapter 4

Extensions of GPC

GPC, in its standard form, is designed based on the transfer function of the process under control. But for the processes with modelling uncertainties or unknown parameter variations, desired performance may not be achieved using standard GPC. Also, for practical implementation, input and output constraints, like actuator saturation constraints, must be applied. Mathematical treatment of GPC allows to incorporate the adaptive parameter estimation algorithm and also various input-output constraints.

Figure 4.1 shows a block diagram of adaptive, constrained GPC. Before the prediction, the process parameters are estimated using a Recursive Least-Square (RLS) estimator, based on the past input-output data of the process. The constraints are applied at the optimization block. GPC uses quadratic optimization, thus the problem of finding optimized control sequence becomes the constrained optimization problem or quadratic programming problem.

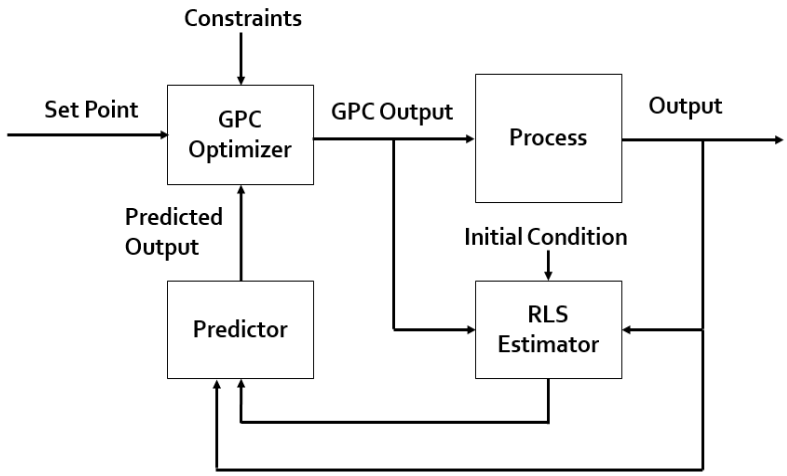

Figure 4.1: Block Diagram of Adaptive, Constrained GPC

4.1 Adaptive GPC

The biggest challenge in applications of control theory is to find out appropriate model of the process under control. The modelling uncertainties can not only degrade the performance of the overall control loop, but also make the system unstable. Thus robustness is important and the system should be able to adapt to the changes occurring in the process parameters.
In adaptive GPC, the modelling coefficients, i.e. polynomials $A(z^{-1})$ and $B(z^{-1})$, are on-line estimated continuously based on the real-time input-output data, and the control law is modified accordingly.
The CARIMA model is given by-

$$A(z^{-1})\Delta y(n) = B(z^{-1})\Delta u(n-1) + \xi(n)$$

Then,

$$\Delta y(n) = -A_1(z^{-1})\Delta y(n) + B(z^{-1})\Delta u(n-1) + \xi(n)$$

where,

$$A_1(z^{-1}) = A(z^{-1}) - 1$$

Denote the modelling parameters and data in the vector forms,

$$\theta = [a_1, a_2, \cdots, a_{n_a}, b_0, b_1, \cdots, b_{n_b}]^T$$
$$\phi(n) = [-\Delta y(n-1), \cdots, -\Delta y(n-n_a), \Delta u(n-1), \cdots, \Delta u(n-n_b-1)]^T$$

Then,

$$\Delta y(n) = \phi(n)^T\theta + \xi(n)$$

The iterative least square method can be used to estimate the parameter vector.

$$\hat{\theta}(n) = \hat{\theta}(n-1) + K(n)[\Delta y(n) - \phi(n)^T\hat{\theta}(n-1)] \tag{4.1}$$
$$K(n) = P(n-1)\phi(n)[\phi(n)^T P(n-1)\phi(n) + \mu]^{-1} \tag{4.2}$$
$$P(n) = \frac{1}{\mu}[I - K(n)\phi(n)^T]P(n-1) \tag{4.3}$$

where, $0 < \mu < 1$ is known as the forgetting factor, $K(n)$ is known as the weighting factor and $P(n)$ is a positive-definite covarince matrix. Generally value of μ is taken as $0.95 < \mu < 1$. The initial value of P is taken as $P(-1) = \alpha^2 I$, where α is a sufficiently large positive scalar. Whereas the initial value of θ depends on the available information about the process.

Algorithm

1. Based on the newly obtained input-output data, use the iterative formula to estimate the modelling parameters, so as to obtain $A(z^{-1})$ and $B(z^{-1})$.

2. Based on the obtained $A(z^{-1})$, solve the Diophantine equation (Eq 3.2).

3. Calculate G (Eq 3.6), H^{-1} and g (Eq 3.11), accordingly.

4. Calculate the optimized control sequence (Eq 3.11) and apply it to the process.

5. Repeat steps (1) to (4) at each time step.

4.2 Constrained GPC

The optimization part of GPC allows various input-output constraints to be incorporated within the GPC law. There are various types of constraints, amplitude and rate constraints on the process input are the most common type. For the control horizon $N_u = 1$, the constrained optimization problem becomes trivial. Whereas, as discussed in detail in [11], $N_u = 2$ is the easier case for applying amplitude and rate constraints. Both amplitude and rate constraints can be simultaneously applied, detailed account of which is given in [8]. A saturation function $sat(x, a, b)$ is defined as-

$$sat(x,a,b) = \begin{cases} a & \text{if x<a} \\ b & \text{if x>b} \\ x & \text{otherwise} \end{cases}$$

4.2.1 Rate Constraints

Rate constraints are the constraints on the rate of change of process input, i.e Δu. Let RC^+ and RC^- be the upper and lower rate constraints respectively. Then there are two cases in the quadratic optimization problem. These cases consider the value of control horizon $N_u = 2$.

When $\Delta u(n+1)$ is feasible
That is for $RC^- < \Delta u(n+1) < RC^+$. Then-

$$\Delta u_c(n) = sat(\Delta u(n), RC^-, RC^+) \tag{4.4}$$

When $\Delta u(n+1)$ is not feasible

$$\tilde{u}_c = \tilde{u} + H^{-1} L_2 e_2 \tag{4.5}$$

where, H is the Hessian matrix in eq 3.11, L_2 is the Lagrange multiplier, given by-

$$L_2 = \frac{(RC^- - \Delta u(n+1))}{H(2,2)}$$
$$e_2 = [0,1]^T$$

Algotithm

1. Calculate the unconstrained GPC control increment sequence $\tilde{u}$, using the GPC control law eq 3.11.

2. If $\Delta u(n+1)$ satisfies the constraints, set $\Delta u_c(n) = \Delta u(n)$.

3. If the constraints are not satisfied, re-optimise using eq 4.5.

4. Implement the control increment $\Delta u(n)$ by clipping it to appropriate value.

$$\Delta u_c(n) = sat(\Delta u(n), RC^-, RC^+)$$

5. Repeat steps (1)-(4) at next time step.

4.2.2 Amplitude Constraints

Amplitude constraints are nothing but saturation constraints. For example, if a control valve can provide maximum flow of $400LPH$, then it is an amplitude constraint which should be applied. Let AC^+ and AC^- be upper and lower amplitude constraints respectively. Following algorithm [11] shows how to implement amplitude constraints for $N_u = 2$. Let $u(n) = u(n-1) + \Delta u(n)$ be the control signal obtained by standard GPC law.

When $u(n+1)$ is feasible
That is for $AC^- < u(n+1) < AC^+$. Then-

$$u_c(n) = sat(u(n), AC^-, AC^+) \tag{4.6}$$

When $u(n+1)$ is not feasible
Let σ_1 and σ_2 be the sums of row 1 and row 2 of matrix H. Then-

$$\Delta u_c(n) = \Delta u(n) - \frac{\sigma_1}{\sigma_1 + \sigma_2}\{AC^- - u(n-1) - \Delta u(n) - \Delta u(n+1)\} \tag{4.7}$$

Algotithm

1. Calculate the unconstrained GPC control sequence u, using the GPC control law eq 3.11.

2. If $u(n+1)$ satisfies the constraints, set $\Delta u_c(n) = \Delta u(n)$ and go to step(4)

3. If the constraints are not satisfied, re-optimise using eq 4.7.

4. Implement the control signal $u(n)$ by clipping it to appropriate value.

$$u_c(n) = sat(u(n-1) + \Delta u(n), AC^-, AC^+)$$

5. Repeat steps (1)-(4) at next time step.

4.2.3 Simultaneous Constraints

To apply both amplitude and rate constraints, positive and negative slacks are defined as [8]-

$$\begin{aligned} S_p(n) &= min(AC^+ - u(n-1), RC^+) \\ S_n(n) &= max(AC^- - u(n-1), RC^-) \\ S_p(n+1) &= min(AC^+ - u(n-1), 2.RC^+) \\ S_n(n+1) &= max(AC^- - u(n-1), 2.RC^-) \end{aligned}$$

Algorithm

1. If $u(n+1)$ is feasible, check for $\Delta u(n+1)$.
   (a) If $\Delta u(n+1)$ is feasible, then-

   $$\Delta u_c(n) = sat(\Delta u(n), S_n(n), S_p k))$$

   (b) If $\Delta u(n+1)$ is not feasible, then calculate $\Delta u_c(n)$ by the Rate Constraints algorithm given above.

2. If $u(n+1)$ is not feasible, then calculate $u_c(n)$ by the Amplitude Constraints algorithm given above and compute the corrected control increments as-

   $$\begin{aligned} \Delta u_p(n) &= u_c(n) - u(n-1) \\ \Delta u_p(n+1) &= sat(u(n+1), AC^-, AC^+) - \Delta u_p(n) \end{aligned}$$

   Check for feasibility of $\Delta u_p(n+1)$.

   (a) If $\Delta u_p(n+1)$ is feasible, then-

   $$\Delta u_c(n) = sat(\Delta u_p(n), S_n(n), S_p k))$$

   (b) If $\Delta u_p(n+1)$ is not feasible, then calculate $\Delta u_c(n)$ by applying Rate Constraints algorithm given above to the control increment, which is unconstrained and uncorrected in amplitude. Redefine positive and negative slacks as-

   $$\begin{aligned} S_p(n) &= min(S_p(n+1) - RC^+, RC^+) \\ S_n(n) &= max(S_n(n+1) - RC^-, RC^-) \end{aligned}$$

   Then the optimal control increment is given by,

   $$\Delta u_c(n) = sat(\Delta u(n), S_n(n), S_p k))$$

Chapter 5

Effect of Tuning Parameters

GPC controller has four tuning parameters, minimum and maximum prediction horizon (N_1, N_2), control weight (λ) and control horizon (N_u). To study the effect of each tuning parameter on closed-loop system performance, an arbitrary transfer function $\frac{5e^{-10s}}{(150s+1)}$ was considered. The process was sampled at sampling time of $T_{samp} = 1sec$. The discrete-time transfer function was found to be $\frac{0.3333z^{-1}}{1-0.9933z^{-1}}z^{-10}$. While studying the effect of one tuning parameter, the other tuning parameters were kept constant (at $N_1 = 11, N_2 = 200, N_u = 2, \lambda = 0.1$). The performance was measured in terms of dynamic characteristics, like percentage peak overshoot M_p, rise time T_r and settling time T_s.

5.1 Prediction Horizon

Prediction horizon is the time length over which the GPC controller predicts the future output of the process. As such the value of the prediction horizon should be such that the process dynamics should be sufficiently covered. Thus larger prediction horizons should give stable output. But the computational requirements increase exponentially with increasing prediction horizon values. Thus optimum value for prediction horizon should be selected such that the response is relatively more stable and also the computational burden is not too much for the controller hardware to meet. With larger computational burden, there is also a possibility that the sampling time requirements are not met. To reduce some of this burden, minimum prediction horizon can also be introduced. For instance, if the process has some time delay t_d, the input does not affect the output upto time $t = t_d$. Thus, logically, the prediction should start after t_d. This can be achieved using minimum prediction horizon. Both minimum and maximum prediction horizon are configurable in GPC algorithm. The Value of maximum prediction horizon (N_2) can't be less than the time delay of the process, otherwise the output would always be zero. It should also be greater than the value of minimum prediction horizon. Thus-

$$N_2 > min(t_d, N_1) \tag{5.1}$$

$$1 < N_1 < N_2 \tag{5.2}$$

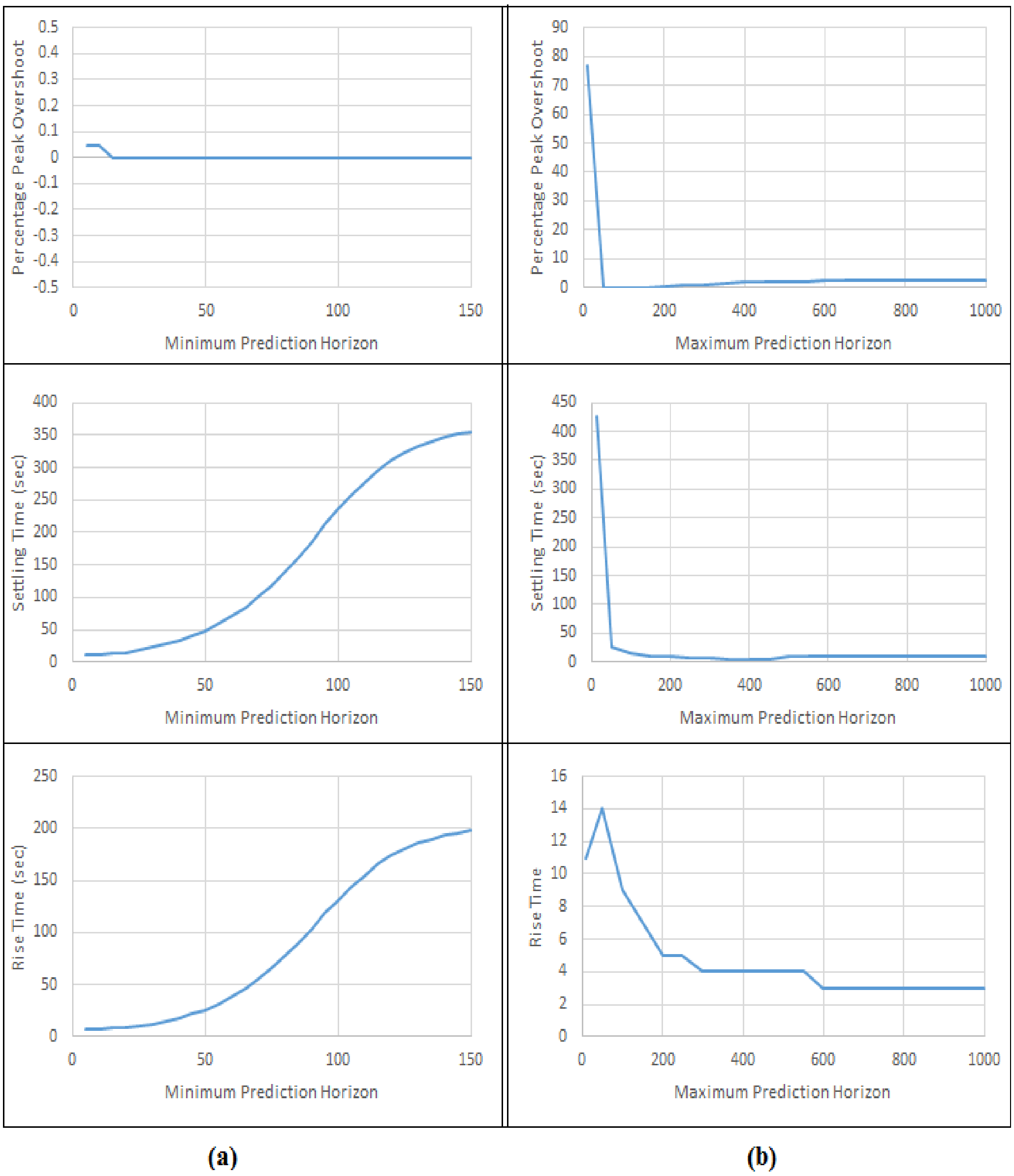

Figure 5.1: Effect of (a)Minimum Prediction Horizon & (b)Maximum Prediction Horizon

From the graphs it was noted that value of N_1 should be as minimum as possible and value of N_2 should be as large as possible. But the added computational burden limits these values. The performance criteria remained same for $N_1 = 1$ to $N_1 = 10$, which is the time delay of the process. Thus value of N_1 should be fixed at $N_1 = t_d$. Optimum value of N_2, considering the computational burden, was found to be around $N_2 = 160$.

5.2 Control Horizon

Control horizon is the length over which the optimized future control sequence is calculated. Its value can be any whole number. Its effect on different performance criteria can be observed from below graphs.

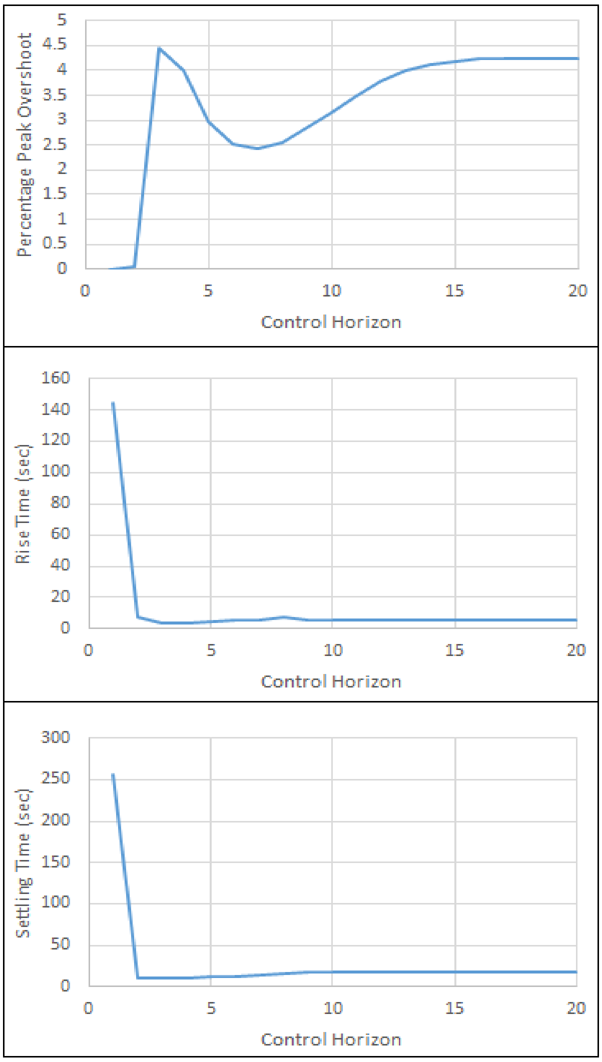

Figure 5.2: Effect of Control Horizon

It was observed that, the optimum value of N_u was 2. $N_u = 2$ not only gives the optimum performance, but it is also an easier case of solving and implementing the constrained optimization problem.

5.3 Control Weight

Control weight is the most important tuning parameter of all. It directly affects the control efforts. More the value of the control weight, lesser is the value of the controller output. From the graphs, it was observed that, increasing the value of λ increased the settling time T_s and rise time T_r, i.e the response became more sluggish, though the peak overshoot was reduced to 0%. Also, much lower values of λ gave latger values of peak overshoot. Thus the optimum value was found to be $\lambda = 0.1$

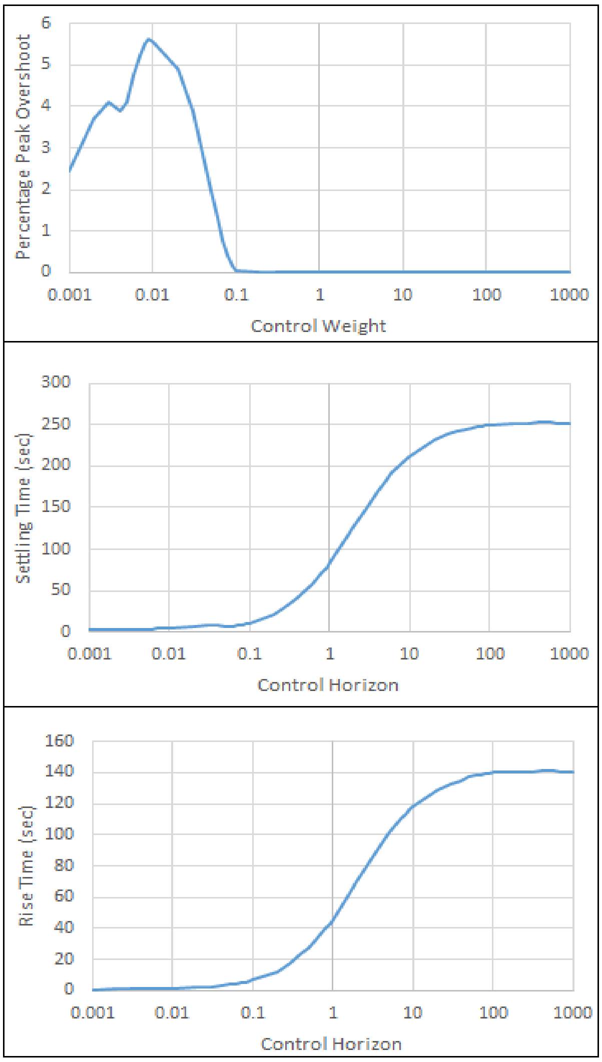

Figure 5.3: Effect of Control Weight

Based on the observations from these tuning simulations, GPC control algorithm was applied to a tank level control system, which is also a first order plus dead-time process. Next chapter gives the mathematical modelling of tank level system and extensive simulation study using GPC control.

Chapter 6

Single Tank Level System

Liquid level is a widely measured and controlled phenomenon in industry. With variable resistance provided by an outlet valve to the flow of liquid, tank level systems exhibit non-linear nature, as there exists square-root relationship between the level and outlet flow. A single tank level system is a first-order system, whereas various interacting and non-interacting combinations of multiple tanks are possible, which can be second order or higher based on number of tanks involved. This section includes mathematical model of single tank level system, followed by simulations of off-line GPC controller and adaptive GPC controller applied to the system.

6.1 Mathematical Modelling

Tank level system is a capacitive process. Together with an outlet valve which provides resistance to the flow of liquid, it exhibits first order behaviour with time constant equal to the product of capacitance and resistance.

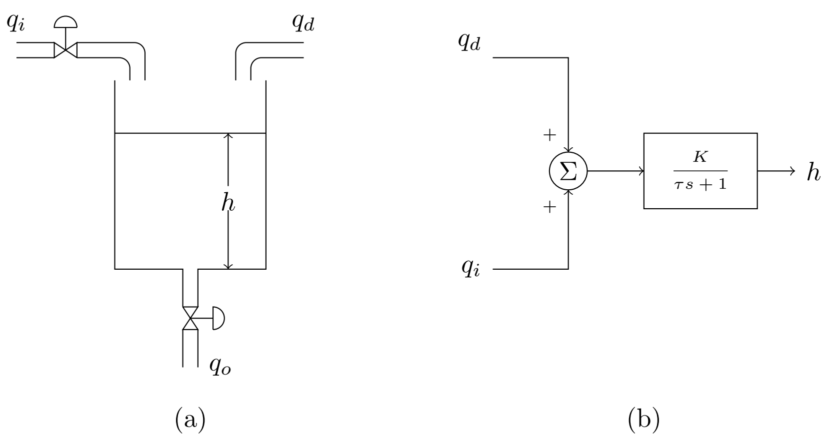

Figure 6.1: Single Tank Level System (a)Process Diagram, (b)Block Diagram

Figure 6.1 shows a tank level system with input flow q_i, outlet flow q_o, instantaneous level h, surface area of tank A, and surface area of outlet a. The mass balance equation for the system can be given as below.

$$A\frac{dh}{dt} = q_i - q_o$$

The output flow q_o can be calculated by applying Bernoulli's equation to the system,

$$h_1 + \frac{P_1}{\rho g} + \frac{v_1^2}{2g} = h_2 + \frac{P_2}{\rho g} + \frac{v_2^2}{2g}$$

where 1 is the point where the liquid surface is open to atmosphere in the tank, and 2 is the point at the outlet at bottom of the tank. Thus $(h_1 - h_2) = h$ is the liquid level in the tank, $P_1 = P_2 = P_{atm}$. The velocity of the liquid at the outlet is then given by-

$$v = \sqrt{2gh} \tag{6.1}$$

Eq 6.1 is known as Toricelli's equation. Thus theoretical value of outlet flow is given by-

$$q_{oth} = av = a\sqrt{2gh}$$

But the theoretical value of flow is never equal to its actual value. Introducing discharge coefficient C_d gives the expression for the outlet flow as-

$$q_o = C_d a\sqrt{2gh} = \beta\sqrt{h}$$

The output flow q_o has square-root relationship with level h. Thus,

$$\frac{dh}{dt} = \frac{1}{A}(q_i - \beta\sqrt{h}) \tag{6.2}$$

This equation gives continuous-time mathematical model of single tank level system in the form of a differential equation.

6.1.1 Linearization

Eq 6.2 gives a non-linear relationship between the input flow and tank level. System transfer function can not be derived from this equation. Even design of GPC controller for the system assumes a locally linearized model. Such a model can be obtained by Taylor series expansion of eq 6.2, around a steady-state (q_{is}, h_s). Linearizing the term $\beta\sqrt{h}$ using Taylor series expansion and neglecting the higher order terms gives,

$$A\frac{dh}{dt} + \frac{\beta}{2\sqrt{h_s}}h = q_i - \frac{\beta\sqrt{h_s}}{2} \tag{6.3}$$

Eq 6.3 is a linear equation. The effect of linearization can be observed from a plot of $\frac{dh}{dt}$ against h, i.e. rate of change of level versus the level. Figure 6.2 shows linearization around the steady-state $q_{is} = 330LPH, h_s = 36.79cm$. The input

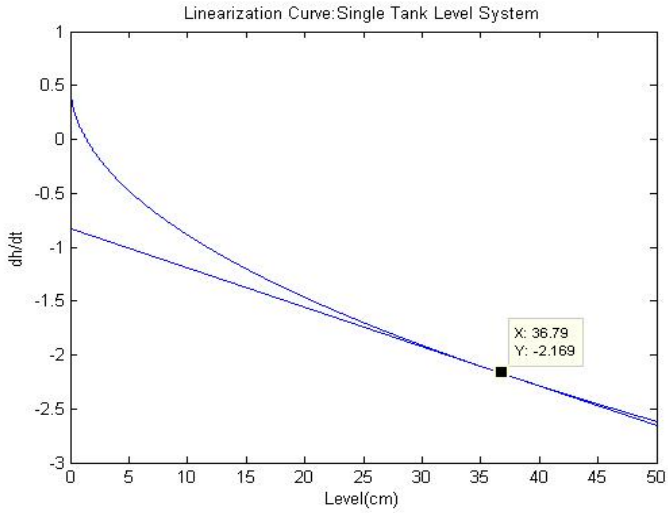

Figure 6.2: Linearization of Single Tank Level System

flow was kept constant at its steady-state value and the level was varied from 0 to $50cm$.
Following facts can be observed from the figure,

- As the level increases, the out flow also increases, which causes the rate of change of level to decrease.

- The plot of $\frac{dh}{dt}$ is non-linear for eq 6.2, whereas it is linear for eq 6.3.

- The two plots meet tangentially at the steady-state operating point, around which the model was linearized.

- The system is highly non-linear for lower values of steady-state height.

6.1.2 Transfer Function

To derive the transfer function, deviation variables need to be introduced. These variables give the deviation of input or output from its steady-state. At steady-state eq 6.3 becomes,

$$A\frac{dh_s}{dt} + \frac{\beta}{2\sqrt{h_s}}h_s = q_{is} - \frac{\beta\sqrt{h_s}}{2} \tag{6.4}$$

Subtracting eq 6.4 from eq 6.3 gives,

$$A\frac{d(h-h_s)}{dt} + \frac{\beta}{2\sqrt{h_s}}(h-h_s) = (q_i - q_{is}) \tag{6.5}$$

Let $h - h_s = H$ and $q_i - q_{is} = Q$, then eq 6.5 becomes,

$$A\frac{dH}{dt} + \frac{\beta}{2\sqrt{h_s}}H = Q \tag{6.6}$$

Taking Laplace transform and rearranging the terms gives system transfer function,

$$\frac{H(s)}{Q(s)} = \frac{K}{\tau s + 1} \tag{6.7}$$

where, $K = \dfrac{2\sqrt{h_s}}{\beta}$ and $\tau = AK$.

It can be observed that, the transfer function is not same for the entire range, but depends upon the steady-state condition. Also, practically, along with the first-order nature of the transfer function, there is an input delay (t_d), which accounts for time taken by the liquid flow to reach the tank after the input is applied. Thus the transfer function of the system is given by,

$$\frac{H(s)}{Q(s)} = \frac{K}{\tau s + 1}e^{-st_d} \tag{6.8}$$

6.2 Descretization

GPC is inherently discrete control algorithm. Thus the process model must be discretized before applying GPC. As the CARIMA model given in eq 1.2 is based on transfer function model of the system, locally linearized model 6.6 is considered for discretization. The selection of sampling time (T_{samp}) depends on process dynamics. Let the samplind time (T_{samp}) and time delay (t_d) be related such that $t_d = dT_{samp}$, where d is an integer. Introducing the time delay in eq 6.6 and using Euler's forward difference approximation for $T_{samp} = 1sec$, the difference equation can be written as,

$$H(n+1) - H(n) + \frac{\beta}{2\sqrt{h_s}}H(n) = \frac{1}{A}Q(n-d)$$
$$H(n+1) - H(n) + \frac{1}{\tau}H(n) = \frac{K}{\tau}Q(n-d)$$

Taking the z-transform of above equation, we get

$$\begin{aligned} zH(z) - H(z) + \frac{1}{\tau}H(z) &= \frac{K}{\tau}Q(z)z^{-d} \\ H(z)(z - 1 + \frac{1}{\tau}) &= \frac{K}{\tau}Q(z)z^{-d} \end{aligned}$$

$$\frac{H(z)}{Q(z)} = \frac{\frac{K}{\tau}z^{-d}}{(1 - z^{-1} + \frac{1}{\tau}z^{-1})}z^{-1}$$

$$\frac{H(z)}{Q(z)} = \frac{bz^{-1}}{1 - az^{-1}} z^{-d} \tag{6.9}$$

where, $b = \frac{K}{\tau}$ and $a = \frac{\tau - 1}{\tau}$. Eq 6.9 gives the discrete-time transfer function a first-order plus dead-time system.

6.3 Simulation Study

6.3.1 Tank Specifications

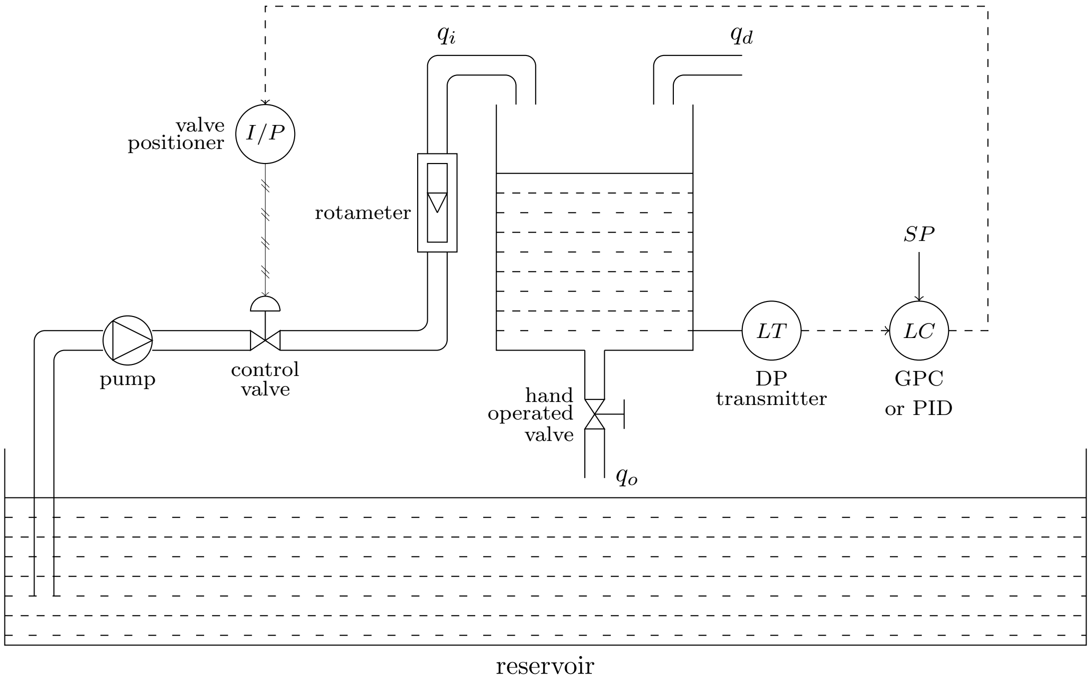

Figure 6.3: Tank Level Control System:Process Diagram

The specifications of the tank level system under study were taken from a laboratory set-up. The diameter of tank was $15cm$, diameter of the outlet pipe was $1.5cm$. The input flow q_i can be varied from $0LPH$ to $400LPH$. The input delay, that is time taken by the controller output to affect the tank level, was approximately $t_d = 10sec$. The final control element, in this case, was a control valve. A valve positioner was also used to convert $4 - 20mA$ signal from the controller into $3 - 15psi$. This pressure signal from the valve positioner controls the position of control valve stem. The control valve was assumed to be linear for simulation purposes. The maximum flow from the control valve was $400LPH = 111.1111cm^3/sec$ and the input to the I/P converter was $4 - 20mA$. Thus, the combined transfer function of the control valve and I/P converter was found to be-

$$\frac{111.1111 - 0}{20 - 4} = 6.9444cm^3 s^{-1} mA^{-1}$$

6.3.2 Process Response

As suggested by eq 6.7, the transfer function of the system depends on the steady-state conditions. Different step inputs were applied to the system and corresponding steady-state conditions and transfer function at that steady-state were calculated. The polynomials $A(z^{-1})$ and $B(z^{-1})$ were also calculated using the discretiztion technique discussed in Section 6.2. Following table shows the results.

Table 6.1: Dependence of Transfer Function on Steady-State

In Flow (q_i) (LPH)	**Steady-State Level (cm)**	**Transfer Function**	**$A(z^{-1})$**	**$B(z^{-1})$**	**d**
50	0.8879	$\frac{0.1278}{25.959s+1}e^{-10s}$	$1-0.9629z^{-1}$	0.00474	10
100	3.5516	$\frac{0.2557}{43.918s+1}e^{-10s}$	$1-0.9777z^{-1}$	0.00569	10
150	7.9911	$\frac{0.3835}{65.877s+1}e^{-10s}$	$1-0.9850z^{-1}$	0.00573	10
167.832	10	$\frac{0.4291}{73.702s+1}e^{-10s}$	$1-0.9866z^{-1}$	0.00574	10
200	14.2063	$\frac{0.5114}{87.8357s+1}e^{-10s}$	$1-0.9887z^{-1}$	0.00575	10
237.348	20	$\frac{0.6068}{104.23s+1}e^{-10s}$	$1-0.9904z^{-1}$	0.00576	10
250	22.191	$\frac{0.6391}{109.779s+1}e^{-10s}$	$1-0.9909z^{-1}$	0.00576	10
290.7	30	$\frac{0.7432}{127.659s+1}e^{-10s}$	$1-0.9922z^{-1}$	0.0577	10
300	31.9566	$\frac{0.7669}{131.738s+1}e^{-10s}$	$1-0.9924z^{-1}$	0.00577	10
335.628	40	$\frac{0.8581}{147.389s+1}e^{-10s}$	$1-0.9932z^{-1}$	0.00578	10
350	43.498	$\frac{0.8948}{153.697s+1}e^{-10s}$	$1-0.9935z^{-1}$	0.00578	10
375.192	50	$\frac{0.9592}{164.763s+1}e^{-10s}$	$1-0.9939z^{-1}$	0.00578	10

Eq 6.2 was simulated to obtain response of single tank level system with above specifications. Fig 6.4 shows response of the system to a step input of $q_i = 0LPH$ to $300LPH$, which was applied at $t = 0sec$.

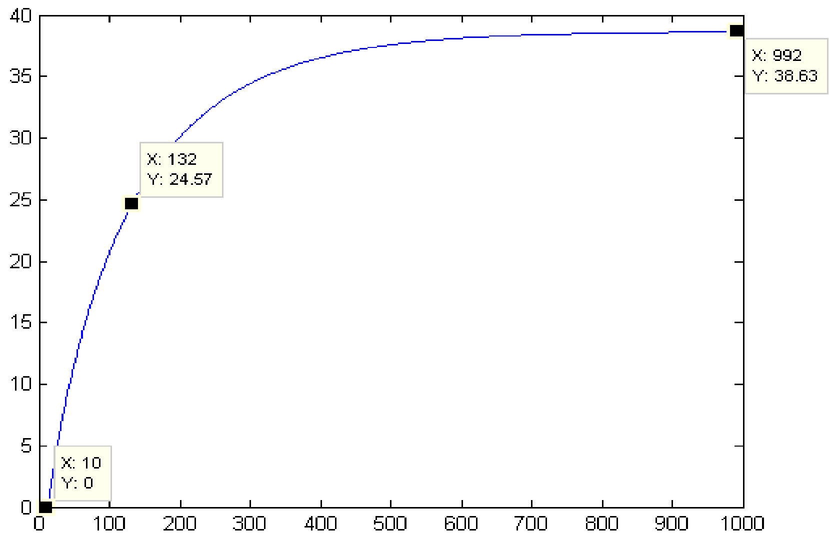

Figure 6.4: Step Response for $q_i = 0LPH$ to $300LPH$

Following facts were observed from the response-

- The response to the input flow q_i is a first-order response.
- The first-order response is characterized by gain and time-constant of the system, which are approximately,

$$\begin{aligned} gain &= \frac{38.63cm}{330LPH} = \frac{38.63cm}{91.67cm^3/sec} = 0.4214cm^{-2}sec \\ \tau &= 122sec \\ t_d &= 10sec \end{aligned}$$

6.3.3 PI Control

Level systems are conventionally controlled by proportional-integral (PI) controller. The derivative term is generally not used because, the oscillations (up-down motion) of stem of control valve around certain position introduces noise in the system. As the noise is a rapidly changing signal, its derivative is higher, which results in higher derivative action. This may cause the closed-loop system to become unstable. Table 6.1 suggests that, the transfer function depends on the steady-state conditions. In order to obtain the response of the system under PI control, the Zeigler-Nicholas Tuning parameters were obtained for following transfer functions corresponding to respective steady-state conditions.

Table 6.2: Zeigler-Nicholas Tuning Parameters

In Flow (q_i) **(LPH)**	**Steady-State Level (cm)**	**Transfer Function**	K_P	K_I
167.832	10	$\frac{2.9798}{73.702s+1}e^{-10s}$	2.226049	0.0066848
237.348	20	$\frac{4.2138}{104.23s+1}e^{-10s}$	2.226185	0.0066852
290.7	30	$\frac{5.1611}{127.659s+1}e^{-10s}$	2.226163	0.0066851
335.628	40	$\frac{5.9589}{147.389s+1}e^{-10s}$	2.226084	0.0066849
375.192	50	$\frac{6.6611}{164.763s+1}e^{-10s}$	2.226159	0.0066852

It should be noted from table 6.2 that, though the transfer functions are different for different steady-state conditions, the tuning parameters K_P and K_I are almost equal. The performance of the system under PI control, tuned with Zeigler-Nicholas tuning rules, was observed by applying different steps of set-point change to the closed-loop system.

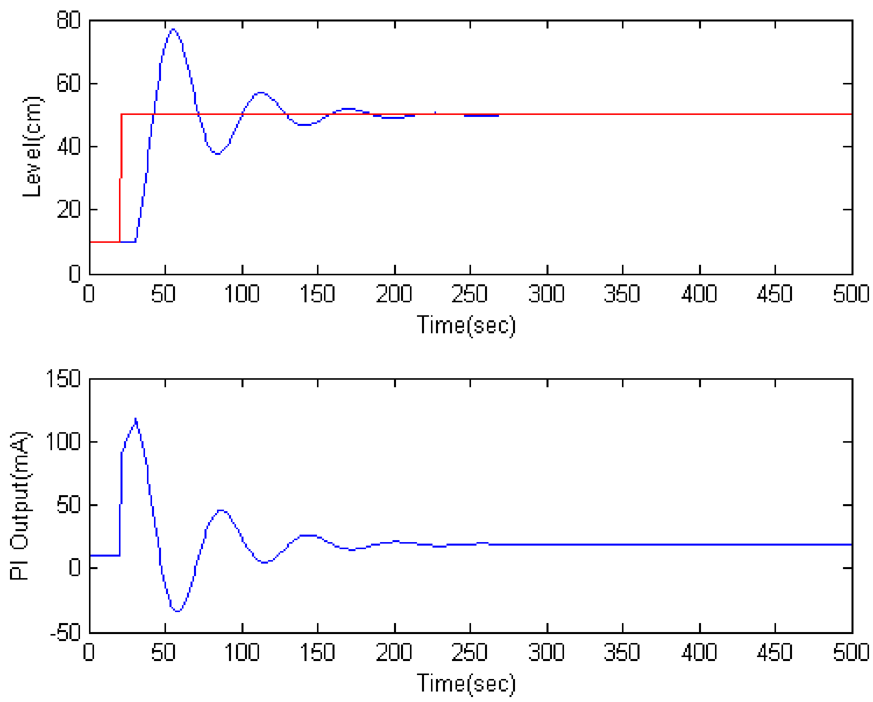

Figure 6.5: PI Control of Tank Level System (Set-point Step: 10-50cm)

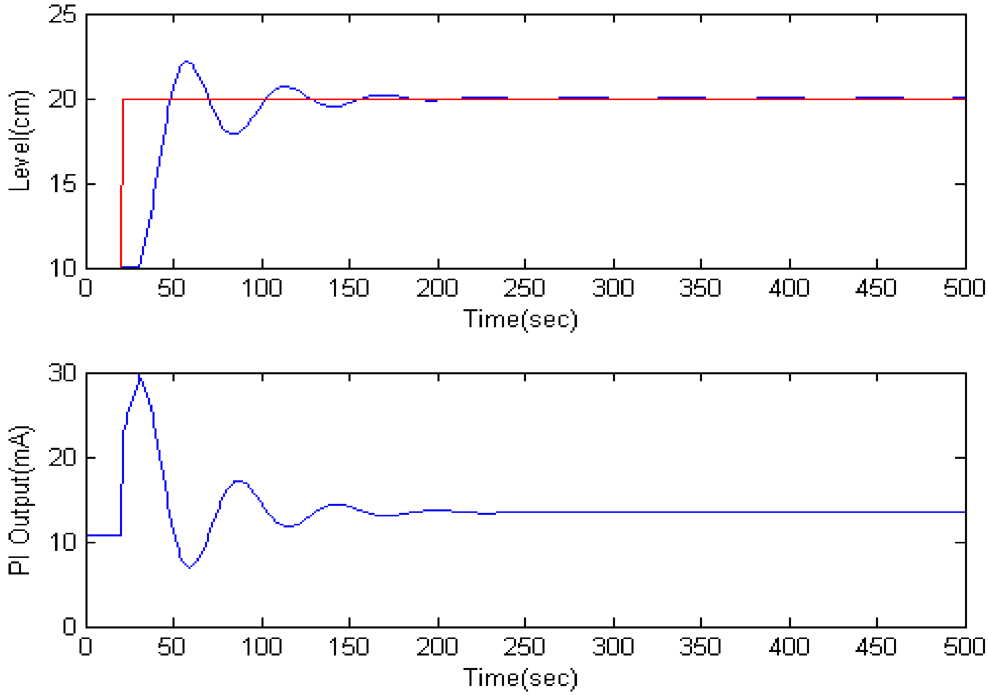

Figure 6.6: PI Control of Tank Level System (Set-point Step: 10-20cm)

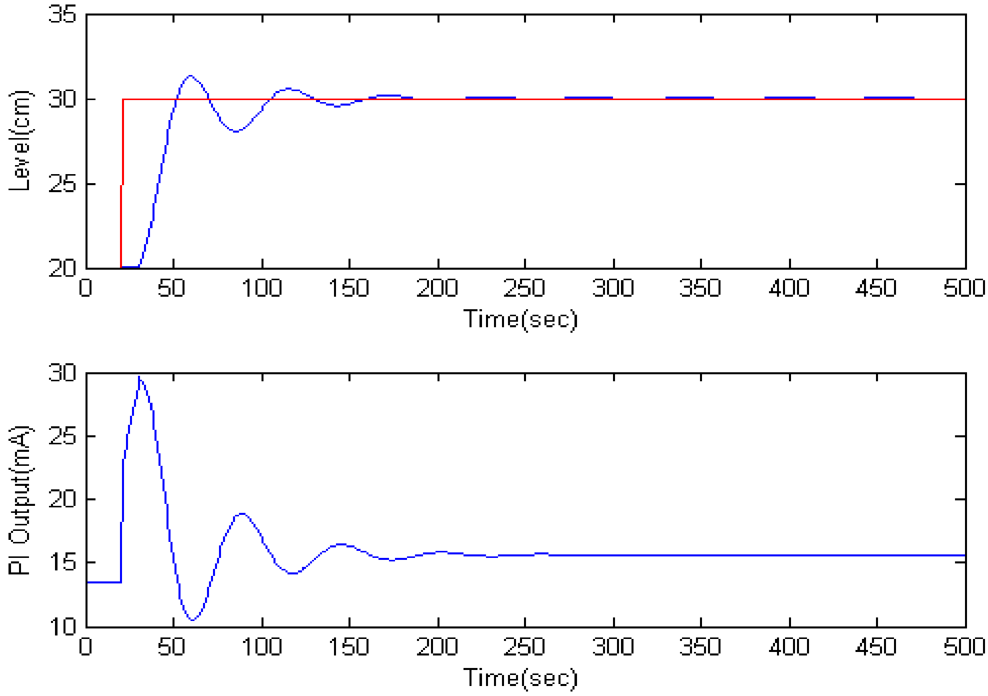

Figure 6.7: PI Control of Tank Level System (Set-point Step: 20-30cm)

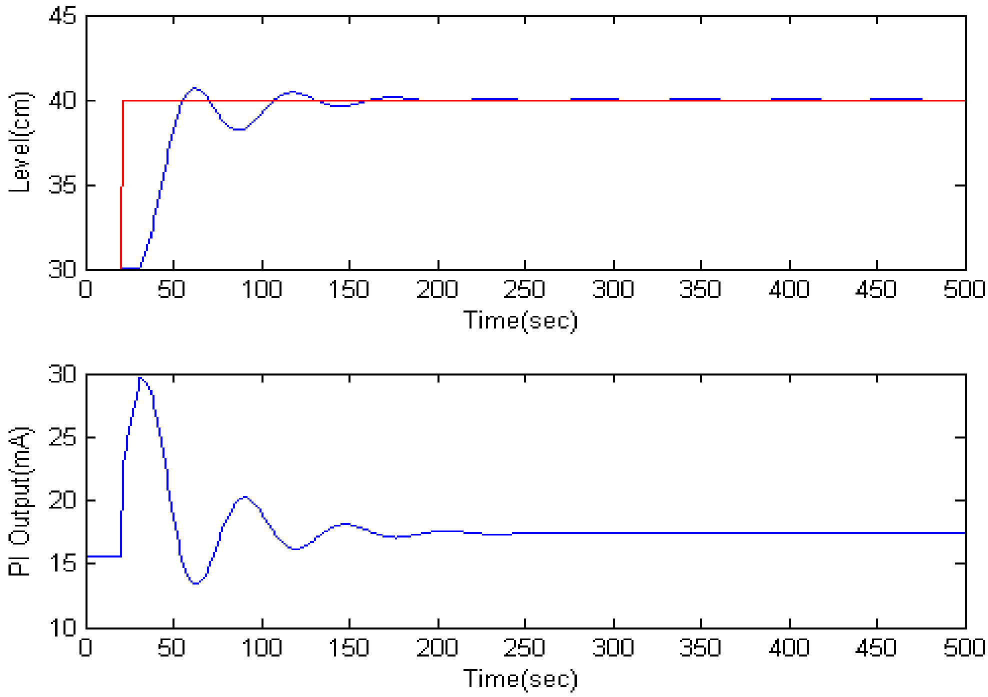

Figure 6.8: PI Control of Tank Level System (Set-point Step: 30-40cm)

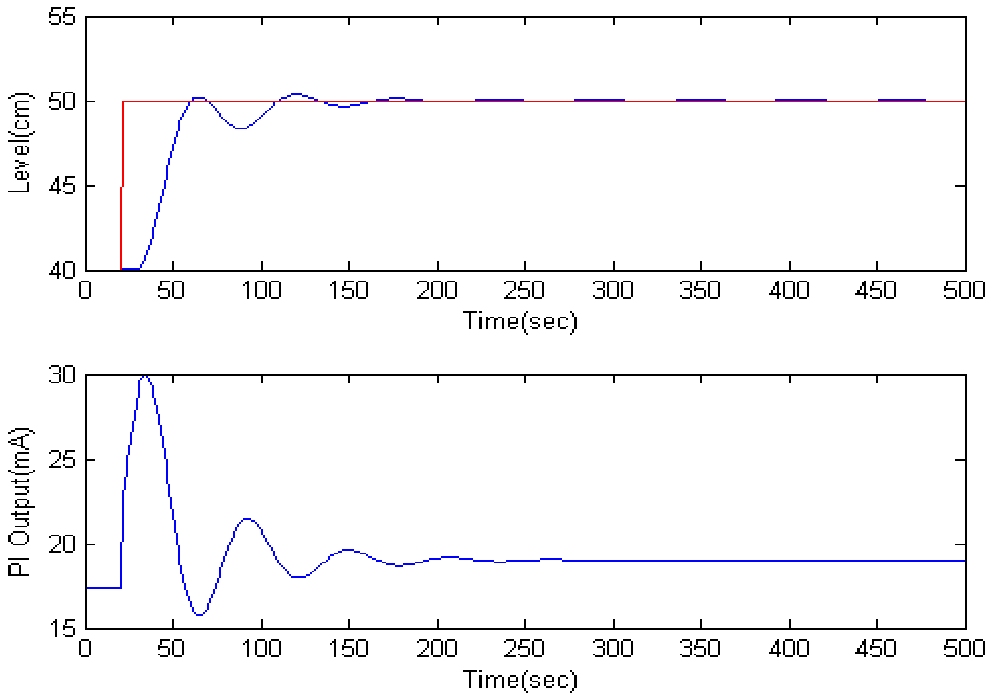

Figure 6.9: PI Control of Tank Level System (Set-point Step: 40-50cm)

6.3.4 Non-Adaptive GPC Control

Basic theory of non-adaptive GPC is discussed in Chapter 3. For the design of and implementation of this standard GPC algorithm, $A(z^{-1})$ and $B(z^{-1})$ polynomials need to be known. These polynomials can be obtained from discrete-time transfer function of the process. As the tank level system under consideration is non-linear, both continuous-time and discrete-time transfer functions depend on the steady-state conditions. Thus $A(z^{-1})$ and $B(z^{-1})$ polynomials were calculated for different steady-state conditions (Refer table 6.1). Based on these values, different step inputs were applied to the system and its performance is plotted. Based on the experience gained from tuning simulations presented in Chapter 5, the tuning parameters were selected $N_1 = 12, N_u = 2, \lambda = 0.1$. Value of N_2 was kept different due to different values of process time constant at different steady-states.

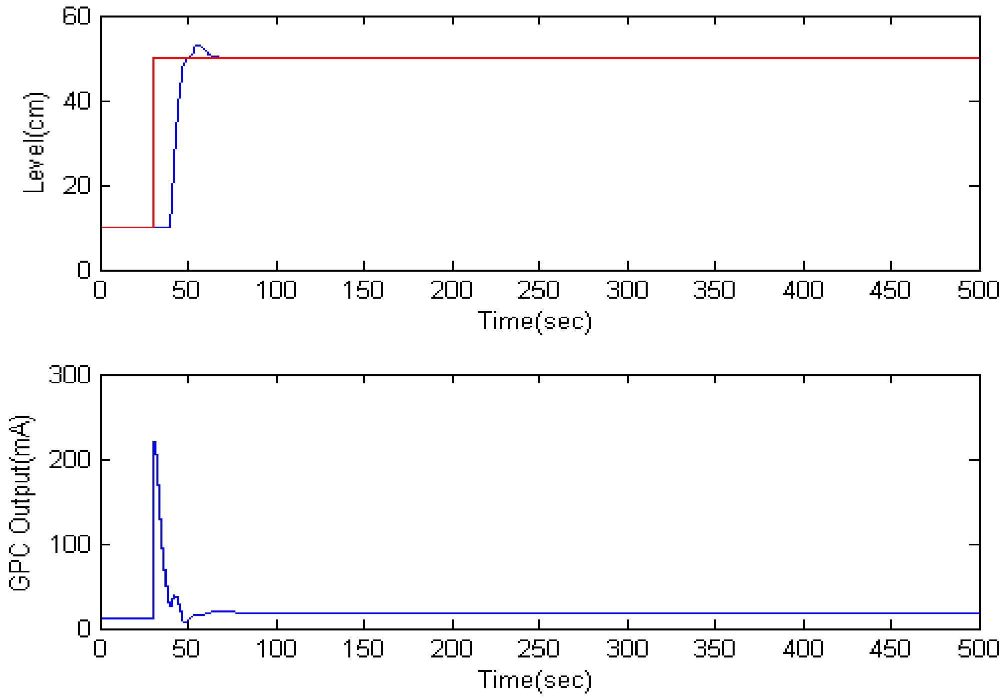

Figure 6.10: GPC Control of Tank Level System (Set-point Step: 10-50cm)($N_2 = 84$)

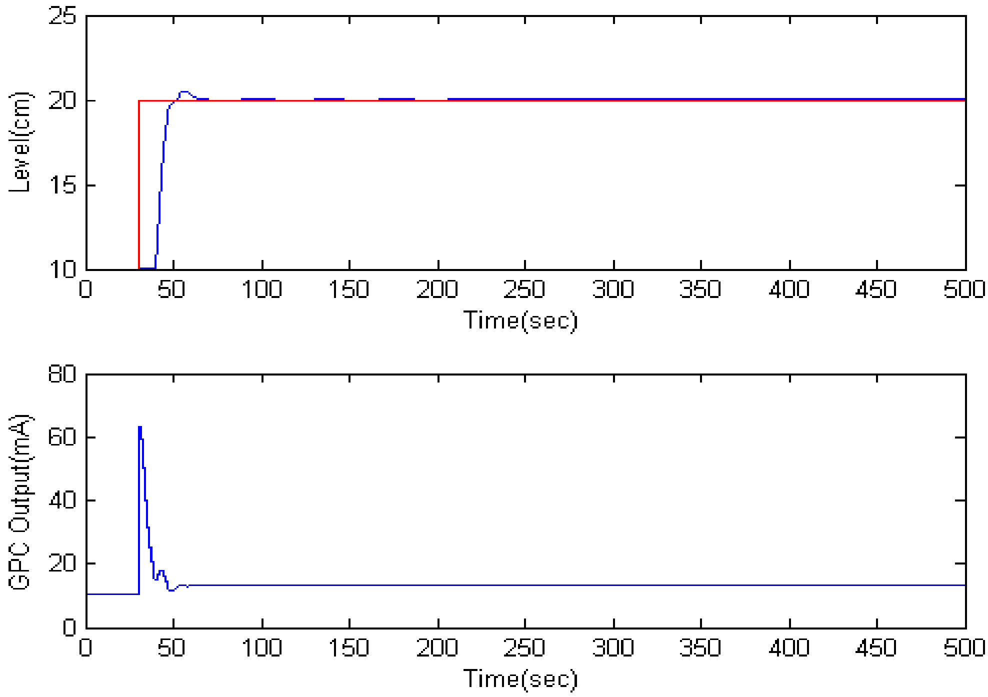

Figure 6.11: GPC Control of Tank Level System (Set-point Step: 10-20cm)($N_2 = 84$)

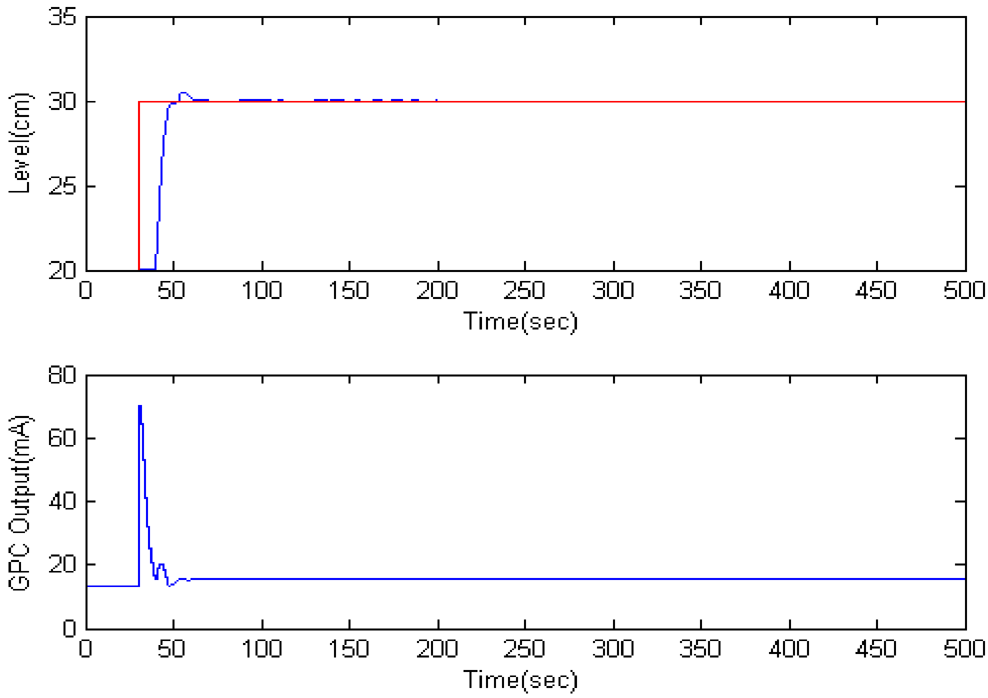

Figure 6.12: GPC Control of Tank Level System (Set-point Step: 20-30cm)($N_2 = 115$)

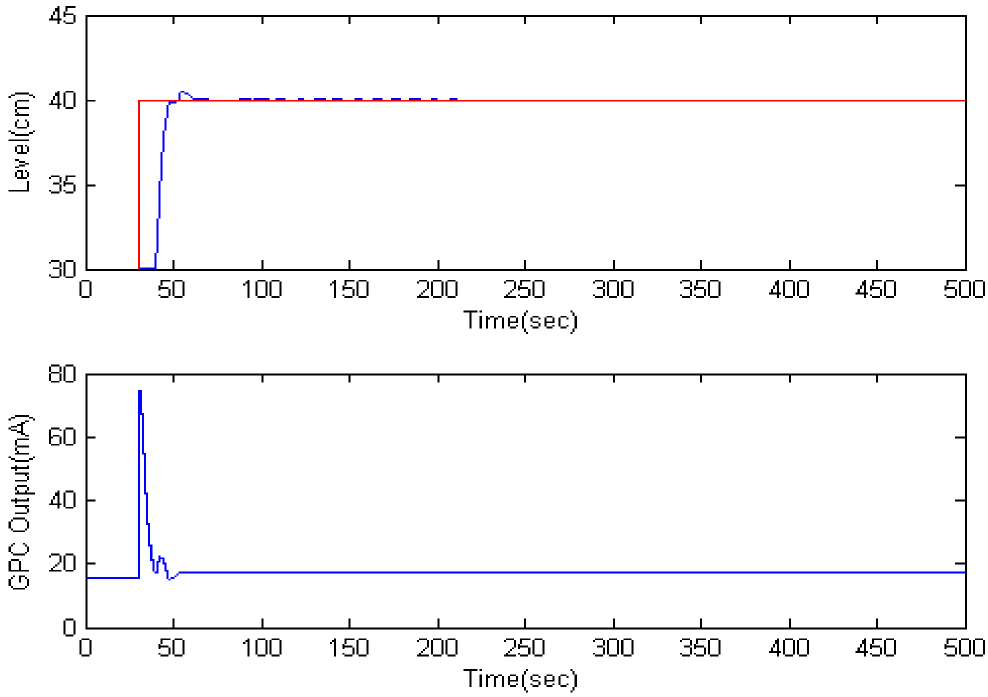

Figure 6.13: GPC Control of Tank Level System (Set-point Step: 30-40cm)($N_2 = 138$)

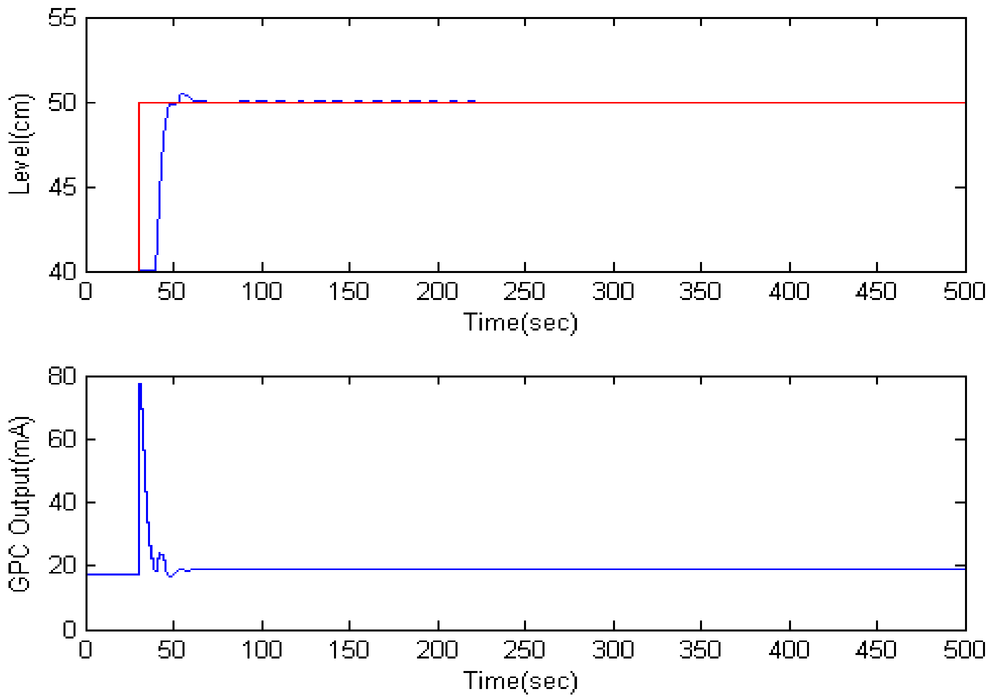

Figure 6.14: GPC Control of Tank Level System (Set-point Step: 40-50cm)($N_2 = 158$)

6.3.5 Adaptive GPC Control

Recursive Least Square (RLS) algorithm with fading memory is used in adaptive GPC technique to estimate the $A(z^{-1})$ and $B(z^{-1})$ polynomials of the process. AGPC deals with these changes in the process parameters of a tank level system and gives better results. In this subsection, AGPC was applied to the tank level system and its results are compared with PI and standard GPC control. The tuning parameters were selected as $N_1 = 12, N_u = 2, \lambda = 0.1$. N_2 was kept different at different set-point changes. In addition to these parameters, RLS algorithms uses two cofigurable parameters, the values for which were decided by trial-and-error as $\alpha = 0.1, \mu = 0.95$.

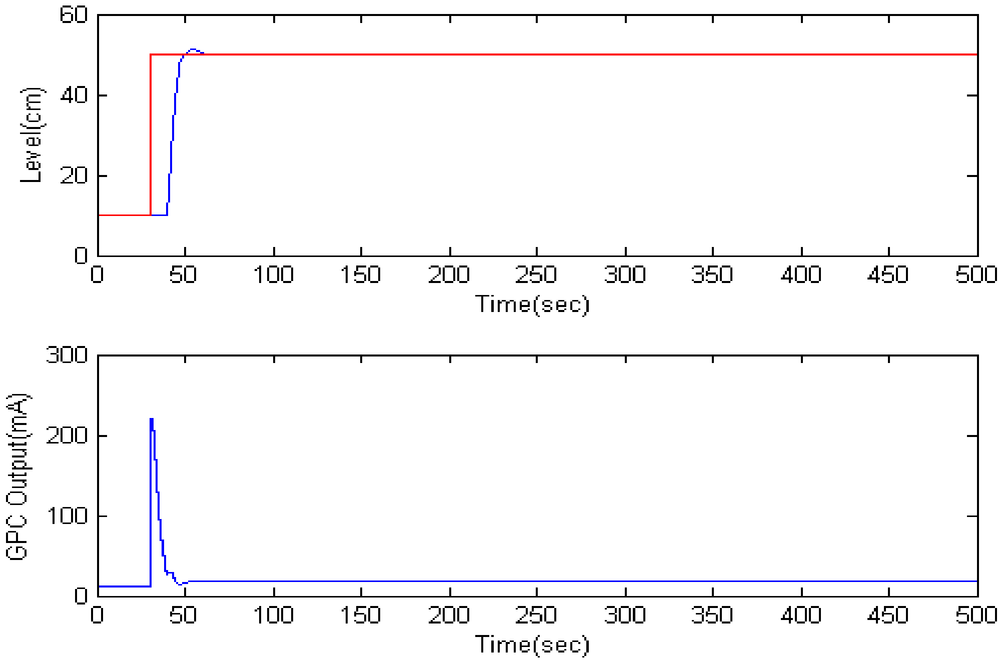

Figure 6.15: Adaptive GPC Control of Tank Level System (Set-point Step: 10-50cm)($N_2 = 84$)

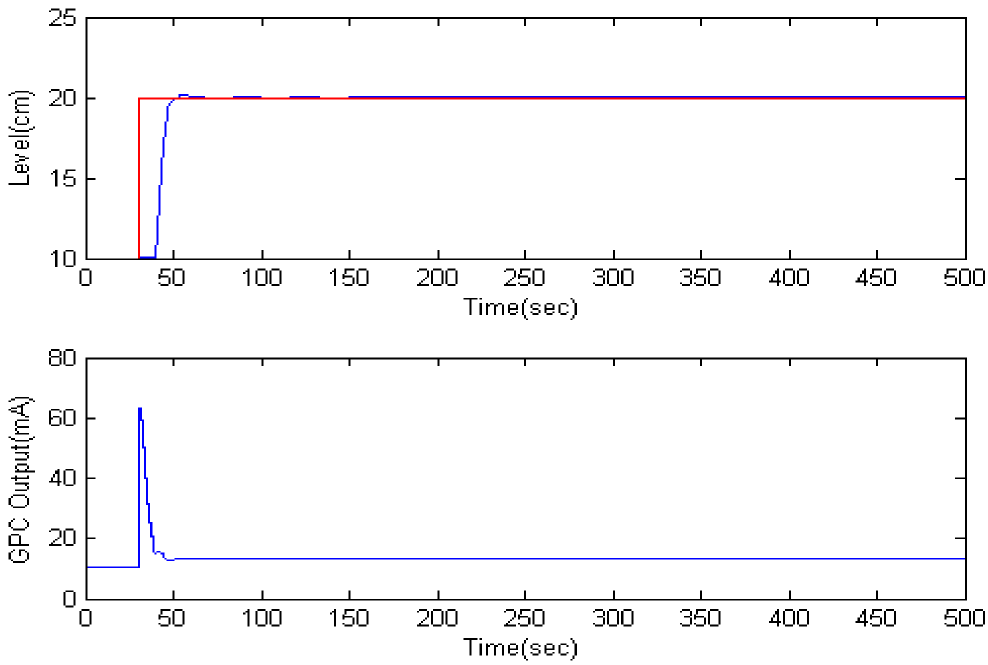

Figure 6.16: Adaptive GPC Control of Tank Level System (Set-point Step: 10-20cm)($N_2 = 84$)

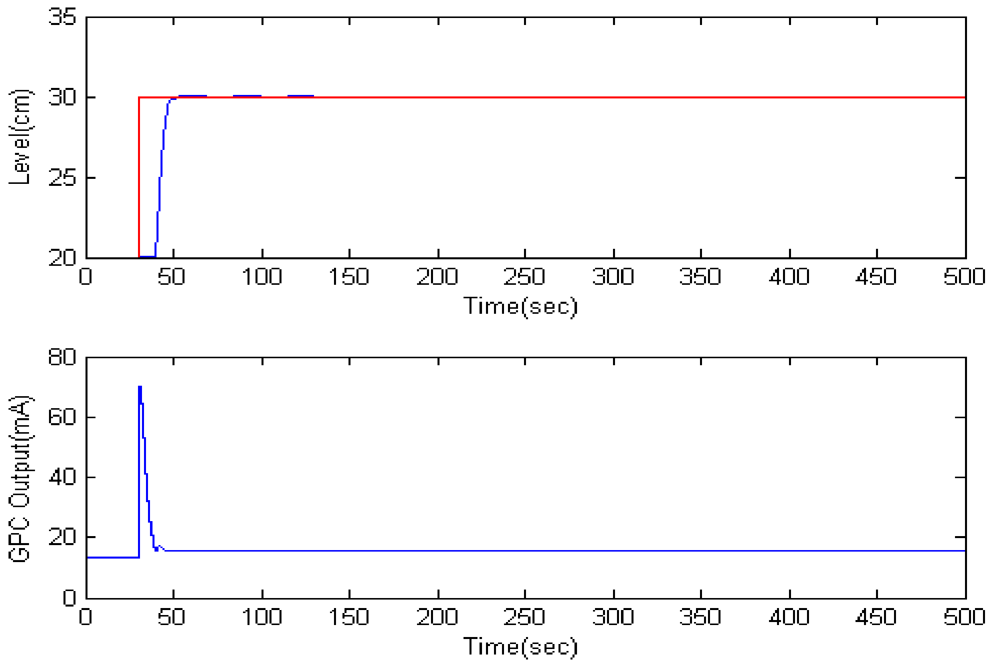

Figure 6.17: Adaptive GPC Control of Tank Level System (Set-point Step: 20-30cm)($N_2 = 115$)

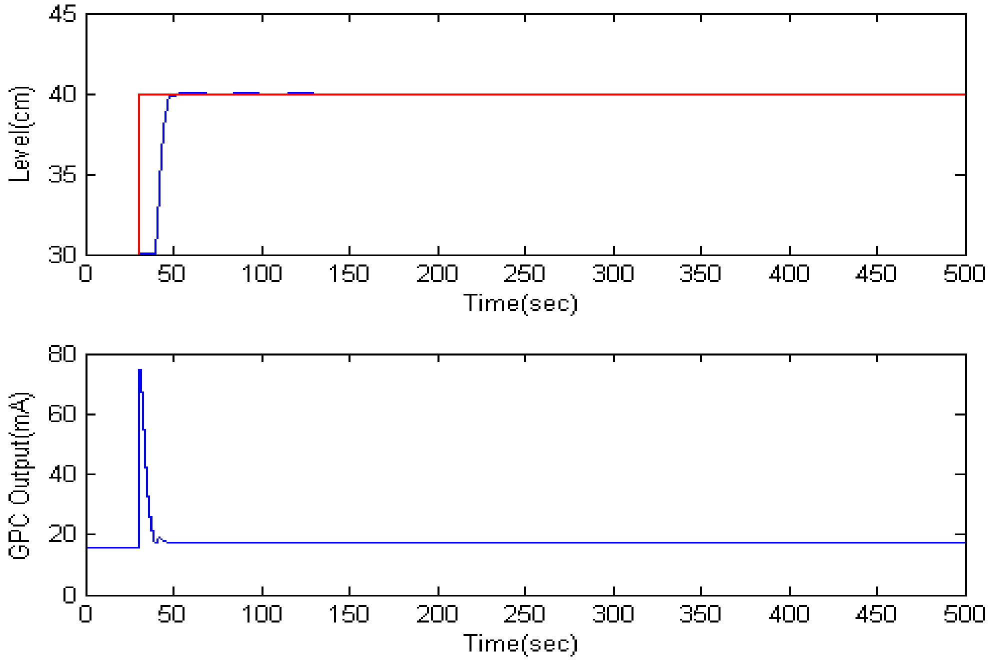

Figure 6.18: Adaptive GPC Control of Tank Level System (Set-point Step: 30-40cm)($N_2 = 138$)

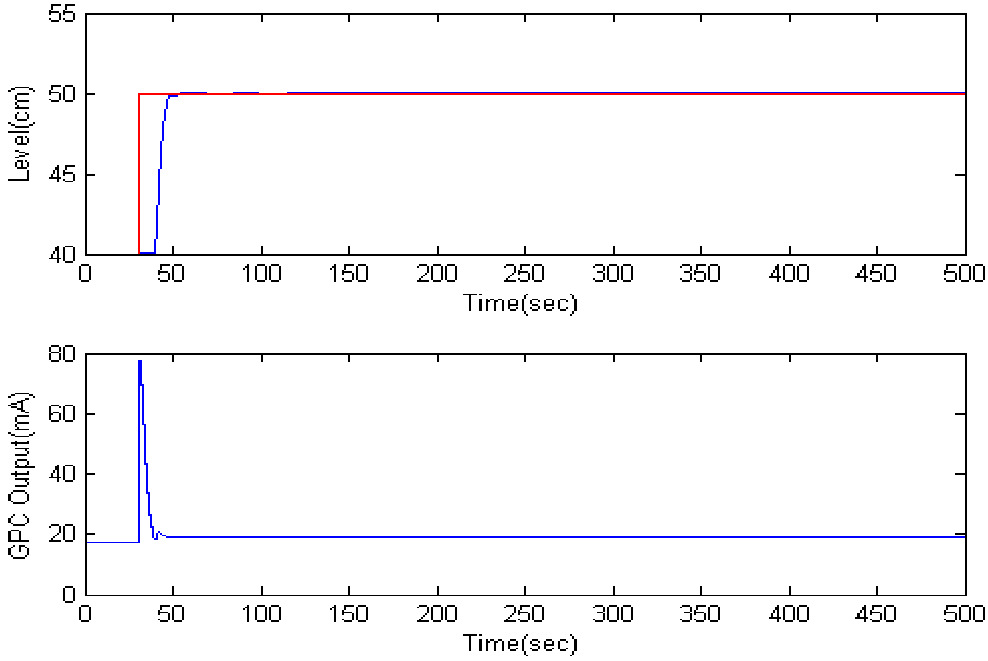

Figure 6.19: Adaptive GPC Control of Tank Level System (Set-point Step: 40-50cm)($N_2 = 158$)

6.3.6 Comparison of Different Control Schemes

The performance of diffenrent control schemes, PI, GPC and AGPC, can be compared in terms of percentage peak overshoot M_p, settling time T_r and settling time T_s. These values were calculated for each control scheme and for each step change applied. The values of the performance criteria were different for different step changes. General trend was that M_p and T_s decreased for upper ranges of operation, whereas the value of T_r decreased with an exception of PI control, in which the value of T_r rather increased.

Table 6.3: Comparison of Peak Overshoot under PI, GPC & AGPC control

Set-point Change	**M_p for PI (%)**	**Deviation for GPC (%)**	**Deviation for AGPC (%)**
10 to 50cm	66.6639	-91.47	-96.77
10 to 20cm	21.5281	-83.16	-95.02
20 to 30cm	12.9452	-69.95	-95.03
30 to 40cm	6.5901	-36.18	-93.72
40 to 50cm	3.4224	31.6	-91.73

Table 6.4: Comparison of Settling Time under PI, GPC & AGPC control

Set-point Change	**T_s for PI (sec)**	**Deviation for GPC (%)**	**Deviation for AGPC (%)**
10 to 50cm	184	-79.89	-84.24
10 to 20cm	133	-73.68	-81.95
20 to 30cm	135	-76.29	-83.70
30 to 40cm	136	-77.20	-85.29
40 to 50cm	137	-78.10	-86.13

Table 6.5: Comparison of Rise Time under PI, GPC & AGPC control

Set-point Change	**T_r for PI (sec)**	**Deviation for GPC (%)**	**Deviation for AGPC (%)**
10 to 50cm	9	-11.11	-11.11
10 to 20cm	13	-30.76	-30.76
20 to 30cm	16	-56.25	-56.25
30 to 40cm	18	-66.67	-66.67
40 to 50cm	21	-76.19	-76.19

The values of the performance criteria for GPC and AGPC control were calculated as the percentage deviation from the values under PI control. The results are tabulated in table 6.3, 6.4 and 6.5. It can be readily observed that AGPC performed much better in all regions of operation.

However, it was observed from the graphs that, the output of the controller, which was given as input to the process, was well beyond its saturation limits. If the standard industrial limit of $4-20mA$ is considered, the control valve, which is calibrated for this range, will saturate beyond $20mA$. Thus these control schemes, when applied practically, will not give desirable results.

Next chapter discusses the constraint handling capability of GPC. Two input constraints, namely amplitude constraint and rate constraint, were applied and the results were observed.

Chapter 7

Adaptive Constrained GPC

Though the performance of adaptive GPC was found to be better, there many practical problems that need to be dealt with. One such important problem is that of the constraints of the system. Constraints can't be incorporated directly in the classical PID controller. However, mathematical treatment of GPC allows the constraints to be incorporated within the control law itself. The constrained GPC was discussed in detail in Chapter 4. This section shows performance of adaptive constrained GPC (ACGPC) applied to a tank level system.

There are various types of constraints, but amplitude and rate constraints on the input are very important. As per the industrial standard, the output of a controller is kept between 4 to $20mA$. The current-to-pressure (I/P) converter is calibrated to give 3 to $15psi$ output for a corresponding 4 to $20mA$ signal. Correspondingly, the control valve is calibrated to give minimum flow for $3psi$ pressure and maximum flow for $15psi$ pressue. Thus 4 to $20mA$ limits on the controller are actually the saturation limits of the control valve. these limits are taken as the amplitude constraints on the GPC controller output. The rate constraints are the limits put on the rate of change of GPC output per second. These are taken arbitrarily as $-2mA$ to $+2mA$. Thus the constrained GPC ensures that the output of the GPC controller stays within these two limits. In other words, the constrained GPC output satisfies following inequalities, at each time step k:

$$4mA \leq u(k) \leq 20mA$$
$$-2mA \leq \Delta u(k) \leq +2mA$$

7.1 Set-point Tracking

Figures 7.1, 7.2, 7.3, 7.4 and 7.5 show the response of tank level system under ACGPC control for different step changes. The tuning parameters were selected as $N_1 = 12, N_u = 2, \lambda = 0.1$.

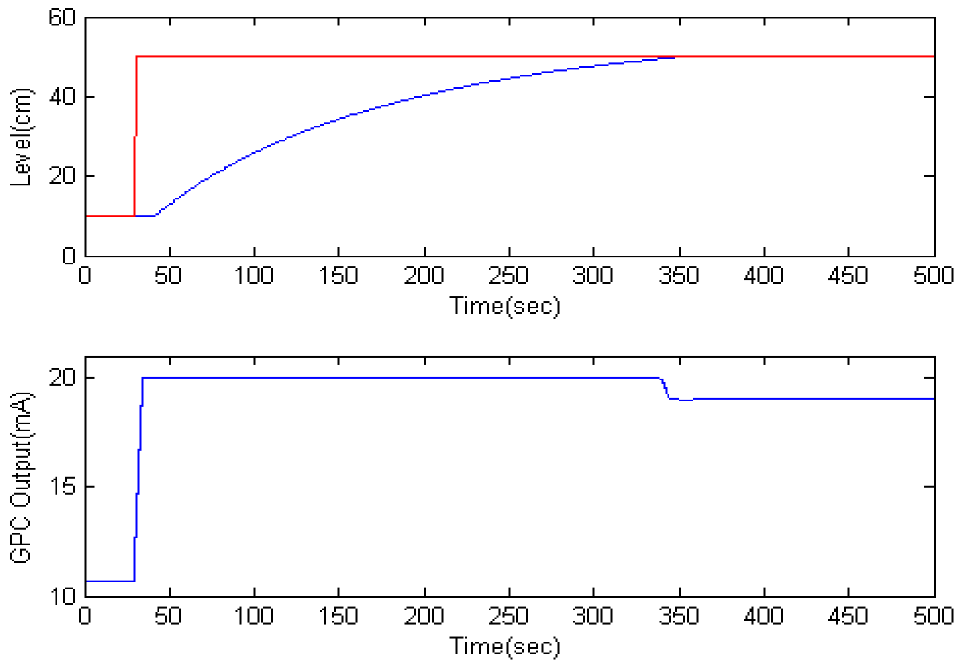

Figure 7.1: Adaptive Constrained GPC Control of Tank Level System (Set-point Change: 10-50cm)($N_2 = 84$)

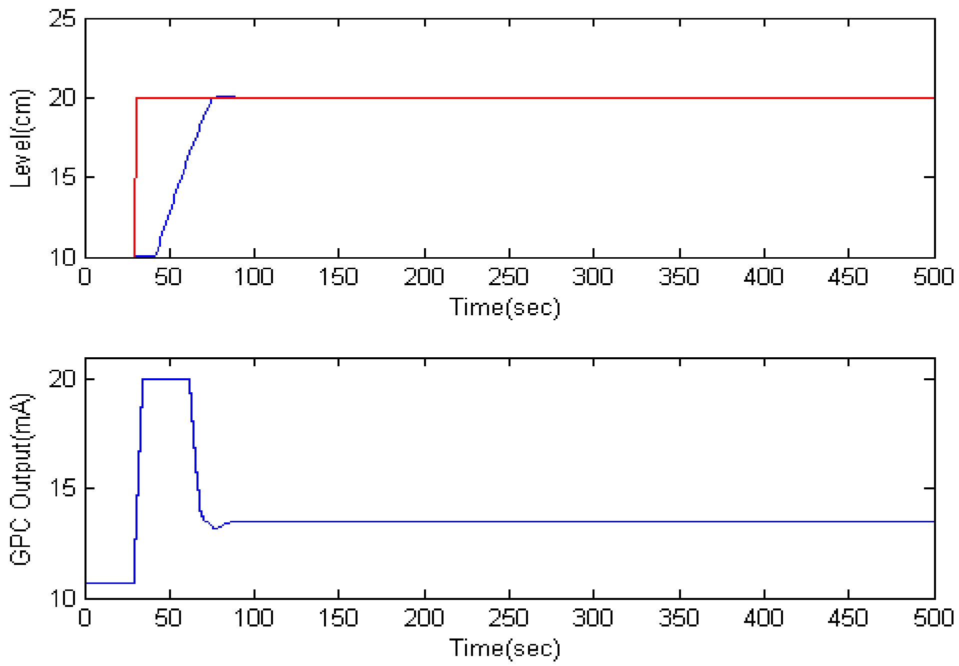

Figure 7.2: Adaptive Constrained GPC Control of Tank Level System (Step Change: 10-20cm)($N_2 = 84$)

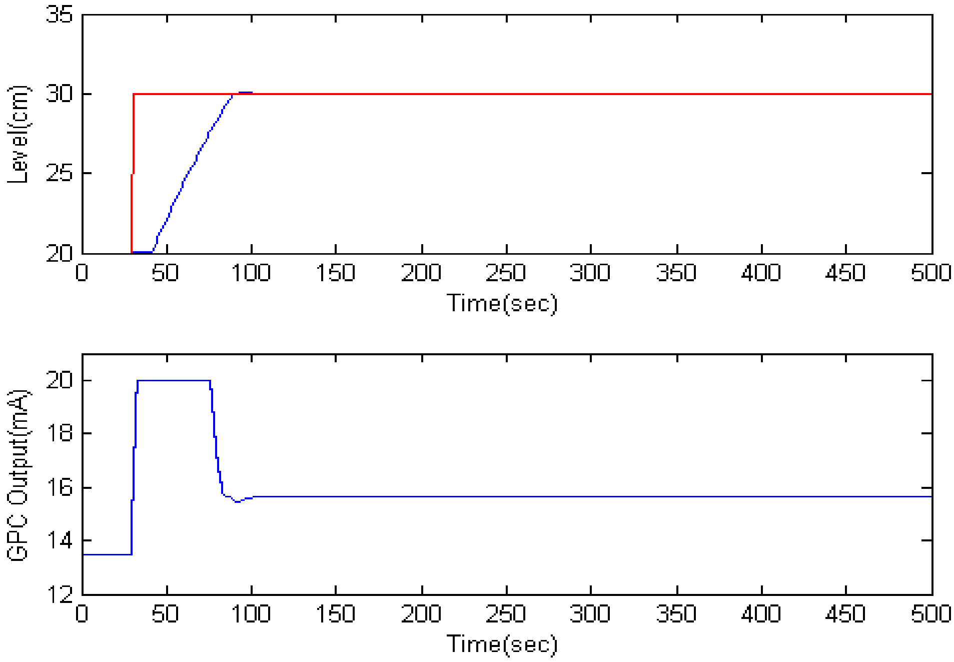

Figure 7.3: Adaptive Constrained GPC Control of Tank Level System (Step Change: 20-30cm)($N_2 = 115$)

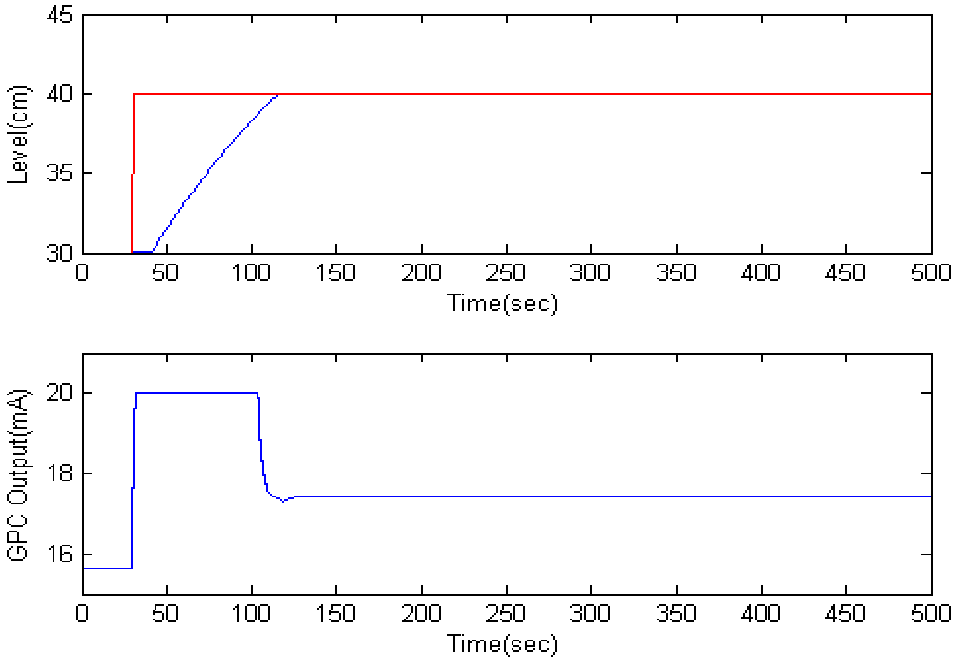

Figure 7.4: Adaptive Constrained GPC Control of Tank Level System (Step Change: 30-40cm)($N_2 = 138$)

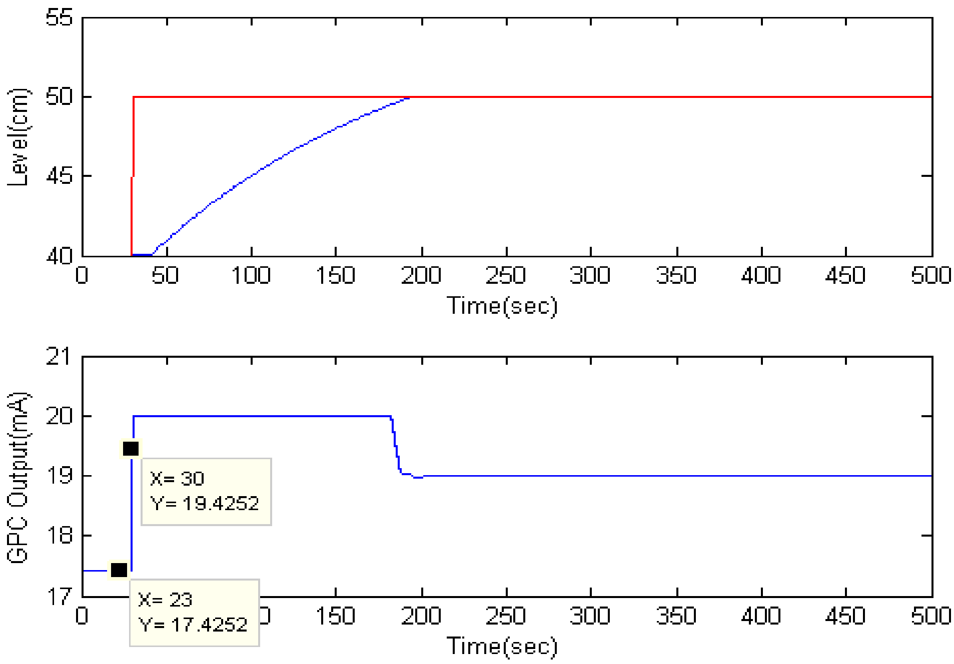

Figure 7.5: Adaptive Constrained GPC Control of Tank Level System (Step Change: 40-50cm)($N_2 = 158$)

The observations from above plotted graph are tabulated below.

Table 7.1: Performance under Adaptive Constrained GPC Control in Different Ranges of Operation

Set-point Change	**M$_p$ (%)**	**T$_s$ (sec)**	**T$_r$ (sec)**	**Max GPC Output (mA)**
10 to 50cm	0.0292	303	218	20
10 to 20cm	0.8719	46	26	20
20 to 30cm	0.5021	58	37	20
30 to 40cm	0.2873	84	60	20
40 to 50cm	0.1109	160	120	20

Though the results seem to be inferior than previously applied control schemes, in terms of the settling time and rise time, these results are more practically achievable. Whenever the controller output, in non-constrained case, goes beyond the limits of 4 to $20mA$, the control valve saturates. Thus besides the fact that the simulation results for non-constrained case are better, practically, the constrained GPC should give better results.

7.2 Disturbance Rejection

Till now, set-point tracking performance of the GPC was simulated and compared with classical PI control. Now the disturbance rejection performance of adaptive constrained GPC (ACGPC) is simulated. The disturbance under consideration is an input disturbance (Refer fig 6.1). The step-change in disturbance flow-rate of magnitude $50LPH$ was applied at $t = 30sec$. The tuning parameters were selected as $N_1 = 12, N_2 = 115, N_u = 2, \lambda = 0.1$.

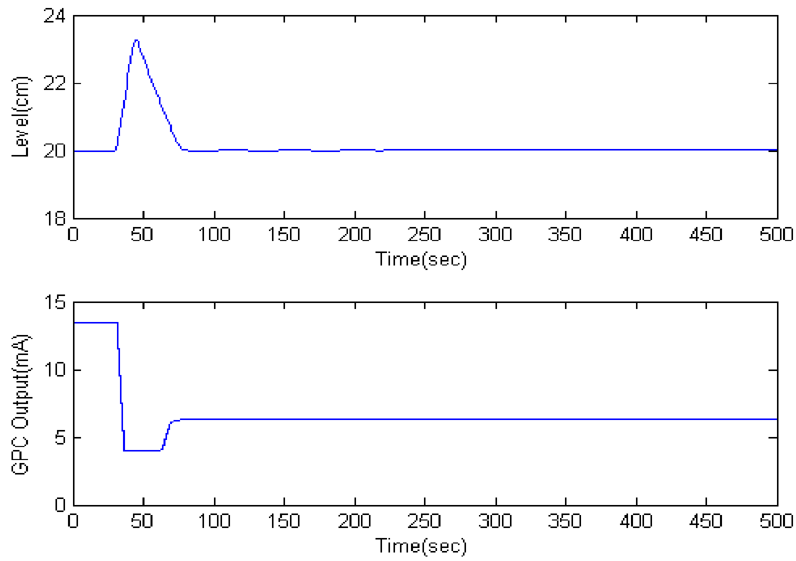

Figure 7.6: Disturbance Rejection under ACGPC Control of Tank Level System

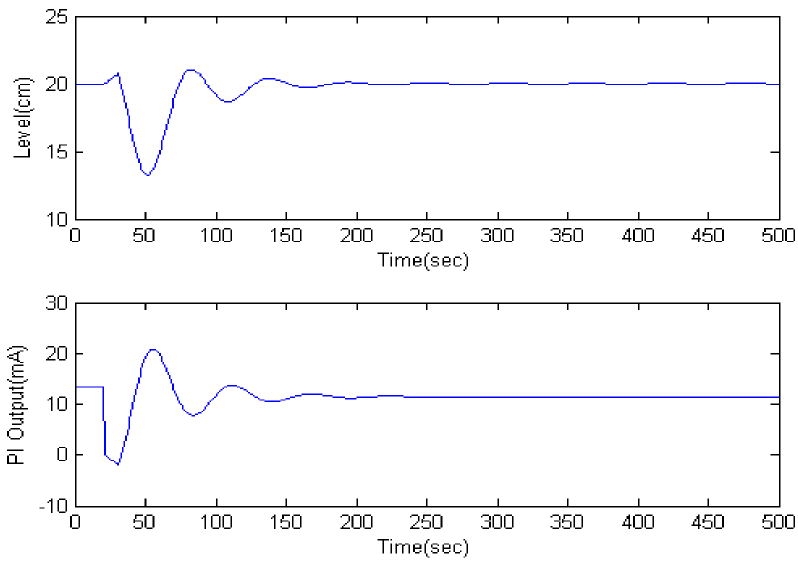

Figure 7.7: Disturbance Rejection under PI Control of Tank Level System

It can be clearly observed from these two plots that, adaptive GPC gave much better performance even when the constraints are imposed. The response of PI controller for disturbance rejection was found to be oscillatory.

7.3 Robustness

Adaptive GPC not only gives better performance under constantly changing values of process parameters, but it also gives a more robust design for the controller. The biggest problem faced in any control problem is of modelling. It is difficult to find out accurate mathematical model for any process. Thus there is always some uncertainty in the derived mathematical model, which is used for control purposes. Primary objective of control system design is to make the system robust to these modelling uncertainties as well as to the parameter variations, which are inherent to the process or occur over a period of time.

To check the robustness of adaptive constrained GPC for a tank level system under consideration, the initial values of modelling parameters, like time delay $t_d = 10$, process gain $K = 4.9798$ and time constant $\tau = 104.23$, at steady-state conditions of (237.348 LPH, 20cm), were varied and its effect on the performance criteria was observed. The under the adaptive GPC control, with tuning parameters set as $N_1 = 10, N_2 = 164, N_u = 2, \lambda = 10$, the performance criteria were found to be $M_p = 0\%, T_s = 192sec, T_r = 101sec$. The results are tabulated below.

It was observed that, higher values of control weight λ gave more robust performance. But, as concluded in Chapter 5 and 6, higher values of λ gives sluggish response. Thus there is a trade-off between the performance and the robustness.

Table 7.2: Robustness of ACGPC:Variations in Process Gain

Process Gain	**Process Gain Variation (%)**	**M_p Variation (%)**	**T_s Variation (%)**	**T_r Variation (%)**
2.9798	-40.16	0	-15.625	-16.83
3.9798	-20.08	0	-8.85	-9.9
4.9798	0	0	0	0
5.9798	20.08	0	10.93	9.9
6.9798	40.16	0	23.44	21.78

Table 7.3: Robustness of ACGPC:Variations in Process Time Constant

Time Constant (sec)	**Time Constant Variation (%)**	**M_p Variation (%)**	**T_s Variation (%)**	**T_r Variation (%)**
84.23	-19.19	0	-18.75	-18.81
94.23	-9.59	0	-8.85	-8.91
104.23	0	0	0	0
114.23	9.59	0	7.29	7.92
124.23	19.19	0	13.54	14.85

Table 7.4: Robustness of ACGPC:Variations in Process Time Delay

Time Delay (sec)	**Time Delay Variation (%)**	**M_p Variation (%)**	**T_s Variation (%)**	**T_r Variation (%)**
8	-20	0	9.89	8.91
10	0	0	0	0
12	20	0	-11.98	-9.9
14	40	0	-18.23	-12.87
16	60	0	-20.31	-10.89

Chapter 8

Conclusion

The effect of different GPC tuning parameters on performance of an arbitrary stable first-order plus dead-time (FOPDT) system was studied. The performance was measured in terms of settling time, peak overshoot and rise time. GPC has four tuning parameters: minimum and maximum prediction horizon, control weight and control horizon. The performance was observed to be better for lower values of the control weight and larger values of prediction horizon. Minimum prediction horizon reduced the computational load without any effect on the performance, because for its values less than the process time delay, the performance criteria remained constant.

From the observations made during the tuning simulations, GPC control was implemented and tuned for a single tank level control system, which is also a FOPDT process. Due to non-linear nature of the process, the transfer function was not same throughout the operating change. It was rather found to be dependent on the steady-state conditions. Thus the results obtained were different in different ranges of operation. However, comparison with the performance of conventional PI controller revealed that, the results obtained for adaptive GPC control were much better.

Though the results of AGPC were found to be better, practically the problem of actuator saturation is dominant and dictates the performance of the overall system. To take such constraints into account, performance of adaptive constrained GPC was studied. ACGPC was found to give practically acceptable results. The disturbance rejection and robustness under ACGPC was also found to be acceptable.

Overall, mathematical theory of GPC provides various extensions of the basic algorithm to incorporate adaptive structure, constraints and multivariable control. This makes GPC widely applicable in many areas. But computational burden of GPC restricts its use in hard real-time processes.

Bibliography

[1] D. Clarke, C. Mohtadi, and P. Tuffs, “Generalized predictive controlpart i. the basic algorithm,” *Automatica*, vol. 23, no. 1, pp. 137 – 148, Mar. 1987.

[2] D. Clarke and C. Mohtadi, “Properties of generalized predictive control,” *Automatica*, vol. 25, no. 6, pp. 859–875, Nov. 1989.

[3] C. Bordons and E. Camacho, “A generalized predictive controller for a wide class of industrial processes,” *Control Systems Technology, IEEE Transactions on*, vol. 6, no. 3, pp. 372–387, May 1998.

[4] J. L. Garriga and M. Soroush, “Model Predictive Control Tuning Methods: A Review,” *Industrial & Engineering Chemistry Research*, vol. 49, pp. 3505–3515, 2010.

[5] M. Xu, S. Li, and W. Cai, “Cascade generalized predictive control strategy for boiler drum level.” *ISA transactions*, vol. 44, pp. 399–411, 2005.

[6] A. Krolikowski and D. Jerzy, “Self-tuning generalized predictive control with input constraints,” *International Journal of Applied Mathematics and Computer Science*, vol. 11, no. 2, pp. 459–479, 2001.

[7] P. Sarhadi, K. Salahshoor, and A. Khaki-Sedigh, “Robustness analysis and tuning of generalized predictive control using frequency domain approaches,” *Applied Mathematical Modelling*, vol. 36, no. 12, pp. 6167–6185, 2012.

[8] M. Short, “Input Constrained Adaptaive GPC for Simple Industrial Plant,” *18th International Conference on Automation and Computing*, pp. 20–25, Sep. 2012.

[9] P. Alkorta, O. Barambones, J. Cortajarena, and A. Zubizarrreta, “Efficient multivariable generalized predictive control for sensorless induction motor drives,” *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 9, pp. 5126–5134, Sept 2014.

[10] H. Hapoglu, S. Karacan, Y. Cabbar, C. E. Balas, and M. Alpbaz, “Application of Multivariable Generalized Predictive Control To a Packed Distillation Column,” *Chemical Engineering Communications*, vol. 174, no. 1, pp. 61–84, Aug. 1999.

[11] T. Tsang and D. Clarke, "Generalised predictive control with input constraints," *Control Theory and Applications, IEE Proceedings D*, vol. 135, no. 6, pp. 451–460, Nov 1988.