ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2016

Model Predictive Control System for Steam Turbine Using MATLAB

G.Mallikarjuna Rao[1], Bikash Swain[2]

Professor, Dept. of EIE, LakiReddy Bali Reddy College of Engineering, Mylavaram, Andhra Pradesh, India[1]

Assistant Professor, Dept. of EIE, LakiReddy Bali Reddy College of Engineering, Mylavaram, Andhra Pradesh, India[2]

ABSTRACT: The traditional control strategy based on PID controllers may be unsatisfactory when dealing with processes with large time delay and constraints. This paper presents a model predictive controller (MPC) for a steam turbine in power plant. We tackle our problem by first formulating our control problem using Model Predictive Control. We used an augmented state model for our system, making the input to the system a new state variable and our new system input as the change in input power. For our MPC controller, we decided to use a simple, fourth-order linear model of the steam turbine. A simple model will help us shift our efforts from modeling to studying and characterize our MPC controller better. We evaluated our MPC Controller by varying the inputs to our model and observe the plant output.

KEYWORDS: Model Predictive Control, PID control, Steam turbine, fourth order linear model, plant output, Prediction horizon

.

I.INTRODUCTION

Model Predictive Control, or MPC, is an advanced method of process control rely on dynamic models of the process, most often linear empirical models obtained by system identification. Common dynamic characteristics that are difficult for traditional PID controllers include large time delays and high-order dynamics.MPC models predict the change in the dependent variables of the modelled system that will be caused by changes in the variables. Independent variables that cannot be adjusted by the controller are used as disturbances. Dependent variables in these processes are other measurements that represent either control objectives or process constraints. This environment led to the development, in industry, of a more general model based control methodology in which the dynamic optimization problem is solved on-line at each control execution. Process inputs are computed so as to optimize future plant behaviour over a time interval known as the prediction horizon. In the general case any desired objective function can be used. Plant dynamics are described by an explicit process model which can take, in principle, any required mathematical form.

II. BASIC CONCEPTS OF MODEL PREDICTIVE CONTROL

A. Objectives of MPC:

- Prevent violations of input and output constraints
- Drive some output variables to their optimal set points, while maintaining other outputs with in specified ranges
- Prevent excessive movement of the input variables.

B. **MPC Basic concepts:**

- Future values of output variables are predicted using a dynamic model of the process and current measurements.
- Unlike time delay compensation methods predictions are made more than one time delay ahead.
- The control calculations are based on both future predictions and current measurements.
- The manipulated variables, $\boldsymbol{u}(k)$, at the k-th sampling instantare calculated so that they minimize an objective function, J.

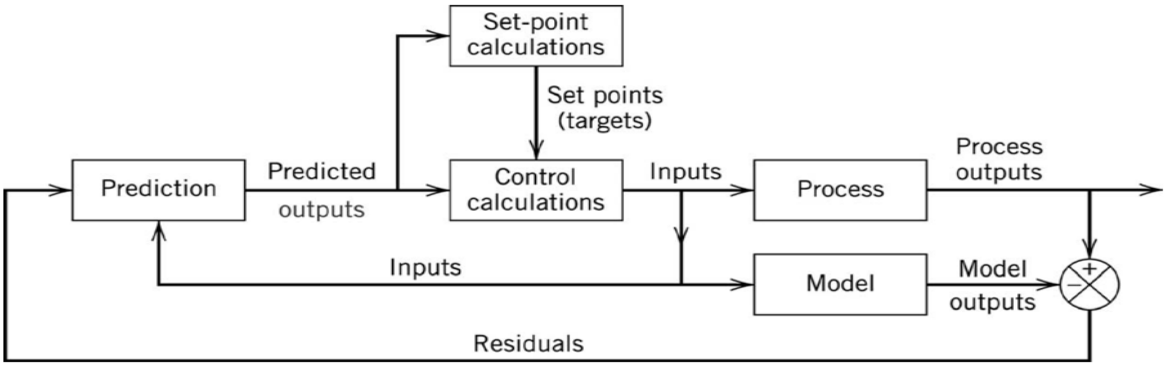
ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2016

- Minimizes the sum of squares of deviations between predicted future outputs and specific referencetrajectory.
- Inequality & equality constraints, measured disturbances are included in the control calculations.
- The calculated manipulated variables are implemented as set point for lower level control loops. (cf. cascade control).



MPC Block diagram

C. Basic Elements of MPC

- Reference Trajectory Specification
- Process Output Prediction (using Model)
- Control action sequence computation (programming problem)
- Error Prediction Update.

The MPC scheme makes use of the receding horizon principle, illustrated in Figure. At each sample, a finite horizon optimal control problem is solved overa fixed interval of time, the prediction horizon.

At the k^{th}sampling instant, the values of manipulated variables U at the next m sampling instants, {u(k),u(k+1),.....,u(k+M-1)} are calculated.

This set of *M*"control moves" is calculated so as to minimize the predicted deviations from the reference trajectory over the next *P* sampling instants while satisfying the constraints.

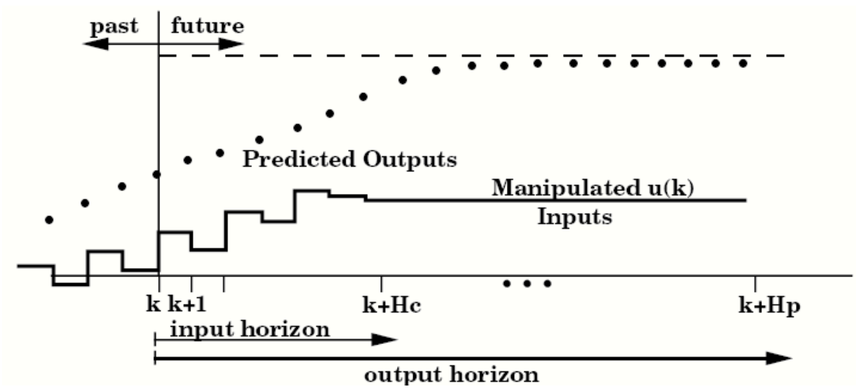
Typically, an LP or QP problem is solved at each sampling instant.

Terminology: *M* = control horizon Hc, *P* = prediction horizon Hp

Then the first "control move", ***u***(*k*), is implemented.

At the next sampling instant, *k*+1, the *M*-step control policy is re-calculated for the next *M* sampling instants, *k*+1 to *k*+*M*, and implement the first control move, ***u***(*k*+1).

Then Steps 1 and 2 are repeated for subsequent sampling instants.



ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2016

The figure shows the basic idea of predictive control. In this presentation of the basics, we confine ourselves to discussing the control of a single-input, single-output (SISO) plant. We assume a discrete-time setting, and that the current time is labeled as time step k, at the current time the plant output is y(k), and that the figure shows the previous history of the output trajectory. Also shown is a set point trajectory, which is the trajectory that the output should follow, ideally. The value of the set-point trajectory at any time t is denoted by s(t). Distinct from the set-point trajectory is the reference trajectory. This starts at the current output y(k), and defines an ideal trajectory along which the plant should return to the set-point trajectory, for instance after a disturbance occurs. The reference trajectory therefore defines an important aspect of the closed-loop behavior of the controlled plant. It is not necessary to insist that the plant should be driven back to the set-point trajectory as fast as possible, although that choice remains open. It is frequently assumed that the reference trajectory as fast as possible, although that choice remains open. It is frequently assumed that the reference trajectory approaches the set point exponentially, which we shall denote T_{ref} defining the speed of response. That is the current error is

$$\epsilon(k) = s(k) - y(k)$$

Then the reference trajectory is chosen such that the error i steps later, if the output followed it exactly, would be

$$\epsilon(k+i) = \exp(-iTs/Tref) * \epsilon(k) = \lambda i * \epsilon(k)$$

Where T_s is the sampling interval and $\lambda = \exp(-Ts/Tref)$.(note that $0<\lambda<1$). That is the reference trajectory is defined to be

$$r(k+i|k) = s(k+i) - \epsilon(k+i) = s(k+i) - \exp(-Ti/Ts) * \epsilon(k)$$

The notation r(k+i|k) indicates that the reference trajectory depends on the conditions at time k,in general. Alternative definitions of the reference trajectory are possible Fore.g.: a straight line from the current output which meets the set point trajectory after a specified time.

A predictive controller has an internal model which is used to predict the behavior depends on the assumed input trajectory ŭ(k+i|k) (i=0,1,…,Hp-1) that is too applied over the prediction horizon, and the idea is to select that input which promises best predicted behavior. We shall assume that internal model is linear ; this makes the calculation of the best input relatively straightforward. The notation ŭ rather than u here indicates that at time step k we only have a prediction of what the input at time k+i may be; the actual input at that time, u(k+i),will probably be different from ŭ(k+i|k).Note that we assume that we have the output measurement y(k) available when deciding, the value of the input u(k).This implies that our internal model must be strictly proper , namely that according to the model y(k) depends on the past inputs u(k-1),u(k-2), …, but not on the input u(k). In the simplest case we can try to choose the input trajectory such as to bring output at the end of the prediction horizon, namely at time k_Hp, to the required value r(k+Hp).

In this case we say, using the terminology of richalet, that we have a single coincidence point at time k+Hp. There are several input trajectories {ŭ(k|k),ŭ(k+1|k),…,ŭ(k+Hp-1|k)} which achieve this , and we could choose one of them , for example the one which requires smallest input energy. But is usually adequate, and in a fact preferable, to impose some simple structure of the input trajectory, parameterized by a smaller number of variables. The figure shows the input assumed to vary over the first three steps of the prediction horizon, but to remain constant thereafter:

$$\breve{u}(k|k) = \breve{u}(k+1|k) = \breve{u}(k+Hp-1|k).$$

In this case there is only one equation to be satisfied

$$\hat{y}(k+Hp|k) = r(k+Hp|k)$$

There is a unique solution. Once a future input trajectory has been chosen, only the first element of that trajectory is applied as the input signal to the plant. That is, we set u(k)=ŭ(k|k), where u(k) denotes the actual input signal applied. Then the whole cycle of output measurement is repeated, prediction, and input trajectory determination is repeated., one sampling interval later: a new output measurement y(k+1) is obtained ;a new reference

trajectoryr(k+i|k+1)(i=2,3,…,) is defined ; predictions are made over the horizon k+1+I,with i=1,2,…Hp; a new trajectory ǔ(k+1+i|k+1),with i=0,1,…,Hp-1) is chosen;
and finally the next input is applied to the plant :

u(k+1)=ǔ(k+1|k+1).

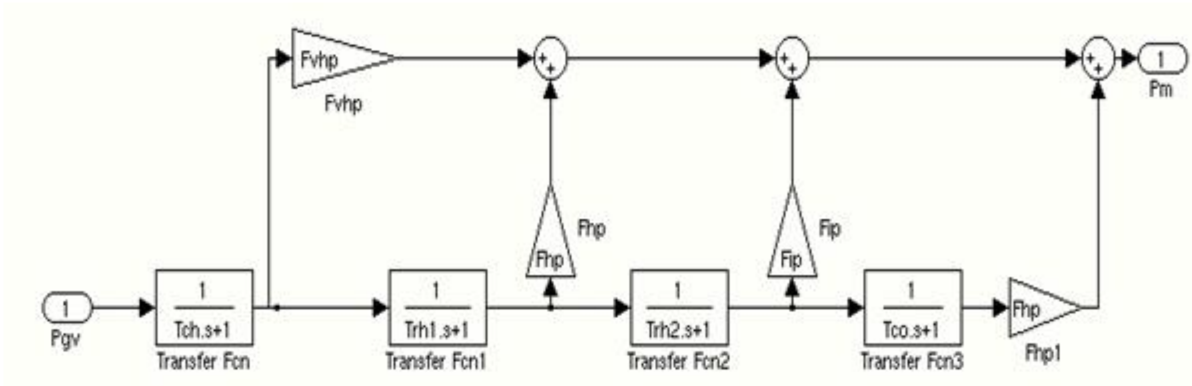
Since the horizon prediction remains of the same length as before, but slides along by one sampling interval at each step this way of controlling a plant is often called a receding horizon strategy.

III. IMPLEMENTATION OF MPC FOR STEAM TURBINE IN MATLAB

Steam turbines are mechanical devices that convert steam power into rotary mechanical motion. These devices require speed control to prevent physical damage to the system; trying to dramatically increase or decrease the speed at once expedites the wear and tear of the machine and could cause breakage. Uncontrolled acceleration in a steam turbine will make the turbine accelerate continuously and become unstable. Steam turbine failures are costly as producing a steam turbine necessitates high precision and requires high quality materials. Although there are some mechanisms in place to prevent uncontrolled acceleration in a steam turbine, a better approach is to regulate the amount of change in mechanical motion that the steam turbine can have at a given moment. The problem of governing the speed of the steam turbine is a control problem, and thus is the motivation for our goal: to design a MPC controller that regulates the acceleration of the steam turbine by limiting the amount of change in input speed the turbine can have.

We tackle our problem by first formulating our control problem using Model Predictive Control. We used an augmented state model for our system, making the input to the system a new state variable and our new system input as the change in input power. For our MPC controller, we decided to use a simple, fourth-order linear model of the steam turbine. A simple model will help us shift our efforts from modeling to studying and characterize our MPC controller better. We evaluated our MPC Controller by varying the inputs to our model and observe the plant output.
Steam turbines operate in the following manner:
The steam enters the turbine through the main throttle valve and flows to one or more governor valves.
The governor valves control the flow of steam into the high-pressure turbine section.
After reheat, the steam enters the intermediate pressure (IP) turbine through the reheat stop-intercept valves.
Valve position is varied to maintain or change the load.

We decided to model the turbine using a linear, fourth-order, double-reheat, and tandem compound steam turbine model. We chose a tandem model (the turbine is assembled along a single shaft) over a compound model (two generators over two shafts) because tandem compound is the type of most modern steam turbines.



Simulink model of controller
Tch-Time Constant of delay between control valves and high pressure turbines
Trh1-Time Constant of delay between control valves and the first reheater
Trh2 - Time Constant of delay between control valves and the second reheater
Tco - Time Constant of delay between the intermediate pressure and low pressure turbines
Input to Model -Pgv - Gate Value Power

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2016

Output from model -Pm - Mechanical Power
Fvhp -Very high pressure
Fhp - High Pressure
Fip -Intermediate Pressure

Steam turbines are mechanical devices that convert steam power into rotary mechanical motion. These devices require speed control to prevent physical damage to the system; trying to dramatically increase or decrease the speed at once expedites the wear and tear of the machine and could cause breakage. Uncontrolled acceleration in a steam turbine will make the turbine accelerate continuously and become unstable. Steam turbine failures are costly as producing a steam turbine necessitates high precision and requires high quality materials.

Although there are some mechanisms in place to prevent uncontrolled acceleration in a steam turbine, a better approach is to regulate the amount of change in mechanical motion that the steam turbine can have at a given moment. The problem of governing the speed of the steam turbine is a control problem, and thus is the motivation for our goal: to design a MPC controller that regulates the acceleration of the steam turbine by limiting the amount of change in input speed the turbine can have.

We tackle our problem by first formulating our control problem using Model Predictive Control. We used an augmented state model for our system, making the input to the system a new state variable and our new system input as the change in input power. For our MPC controller, we decided to use a simple, fourth-order linear model of the steam turbine. A simple model will help us shift our efforts from modeling to studying and characterize our MPC controller better.

We evaluated our MPC Controller by varying the inputs to our model and observe the plant output.
We modelled our steam turbine using the following time constants [3]:
Tch = 0.1; Trh1 = 5; Trh2 = 5; Tco = 3
and valve weights [3]:
Fvhp = .3; Fhp = .2; Fip = .3
Using Simulink Linear Analysis Tool in MATLAB, we obtain the transfer function of our system.

$$\frac{3S^3 + 2.6S^2 + 0.8533S + 0.1333}{S^4 + 10.73S^3 + 7.507S^2 + 1.747S + 0.1333}$$

Writing this in state space representation yields

$$A = \begin{bmatrix} 10 & 0.2 & 0 & 0 \\ -10 & -0.2 & 0 & 0 \\ 0 & 0.2 & 0.2 & 0 \\ 0 & 0 & 0 & 0.3333 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 3 \\ 0.4 \\ 0.6 \\ 0.0667 \end{bmatrix} \quad D = [0]$$

A. Control Goals

As previously stated, we want our controller to ensure that the change in input signal to thesystem stays within a certain range. We also want to track a reference input to the system. This will yield a system that follows a reference signal as well as ensures the input change does not become unreasonable. The importance of these two control objectives is weighted according to the importance of each objective.

Optimization:
We will first setup our optimization problem using the following objective function:

$$JN(x(0);U)= \sum_{k=0}^{N-1}(((y(k) - r(k))TQ(y(k) - r(k)) + uT(k)Ru(k)))$$

Where U is the sequences of inputs & Q; R are the weight matrices.
Our optimization problem thus becomes:

$$\text{Min } \sum_{k=0}^{N-1}(((y(k) - r(k))TQ(y(k) - r(k)) + uT(k)Ru(k)))$$

Such that
x(k + 1) = Ax(k) + Bu(k)
y(k) = Cx(k)
$\_\Delta U_{lowerbound} \leq \Delta U \leq U_{upperbound}$

The first quadratic term, with [y(k) - r(k)], is for reference tracking. We want the output of the plant to follow a specified input, and minimize on the difference of the actual output with the desired output. The second term, with _U,

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2016

allows us to minimize on input change for the reasons specified earlier. Q and R are the respective weight matrices for these two control objectives.

An optimization cost function J over the receding prediction horizon, to calculate the optimum control moves. The optimization cost function is given by:

$$J = \sum_{i=1}^{N} w_{x_i}(r_i - x_i)^2 + \sum_{i=1}^{N} w_{u_i} \Delta {u_i}^2$$

Without violating constraints (low/high limits) with:

x_i= *i* -th controlled variable (e.g. measured temperature)

r_i= *i* -th reference variable (e.g. required temperature)

u_i= *i* -th manipulated variable (e.g. control valve)

w_{x_i}= weighting coefficient reflecting the relative importance of x_i

w_{u_i}= weighting coefficient penalizing relative big changes in u_i

Graphical User Interface:

We implemented in GUI using MPC Tool box in MATLAB to facilitate the simulation and testing of our model. The following parameters can be changed

A. Simulation Parameters:

- Sampling Time
- Time Horizon
- Total Simulation Time

B. Model Parameters:

- Max Positive Input Change
- Min Positive Input Change

IV. SIMULATION

We decided to study our MPC controller by varying the following parameters:

- Time Horizon i.e. Prediction horizon
- Constraint on the Change in Input

A. Time Horizon:

We ran our simulation for 50 seconds with the constraint 0.1 ≤ΔU ≤0.1 and varied the time Horizon. Increasing the time horizon allows our MPC controller to look farther into the future and optimize for future changes. We would expect our plant output to be smoother as we increase the time horizon.

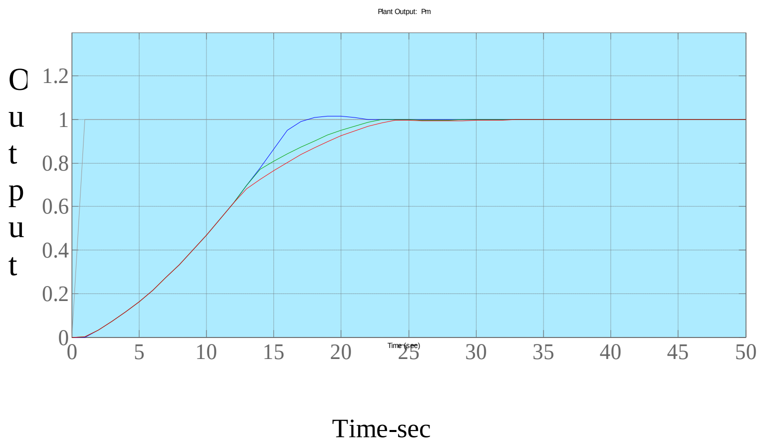


Fig-1 Plant Output when Time Horizon is constant and ΔU constraint is set between0.1 ≤ΔU ≤0.1

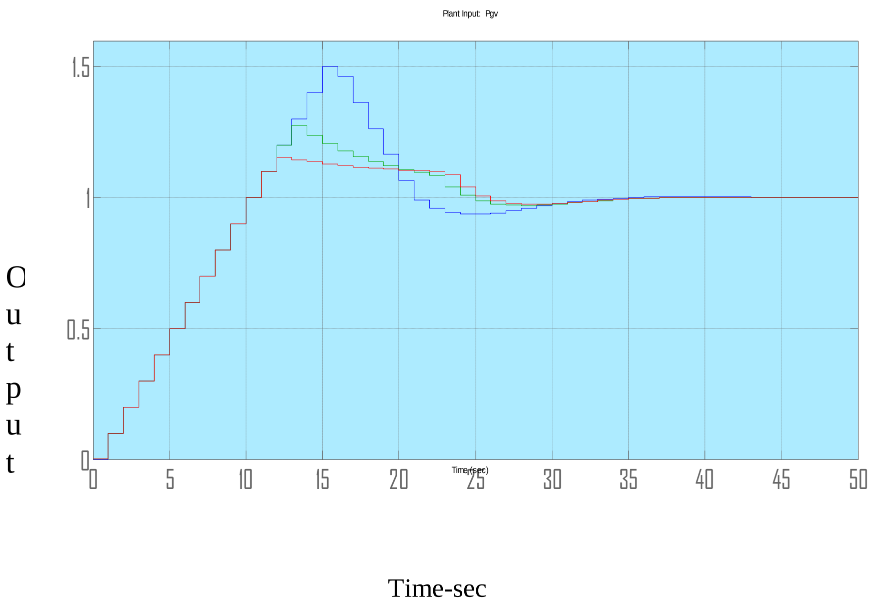
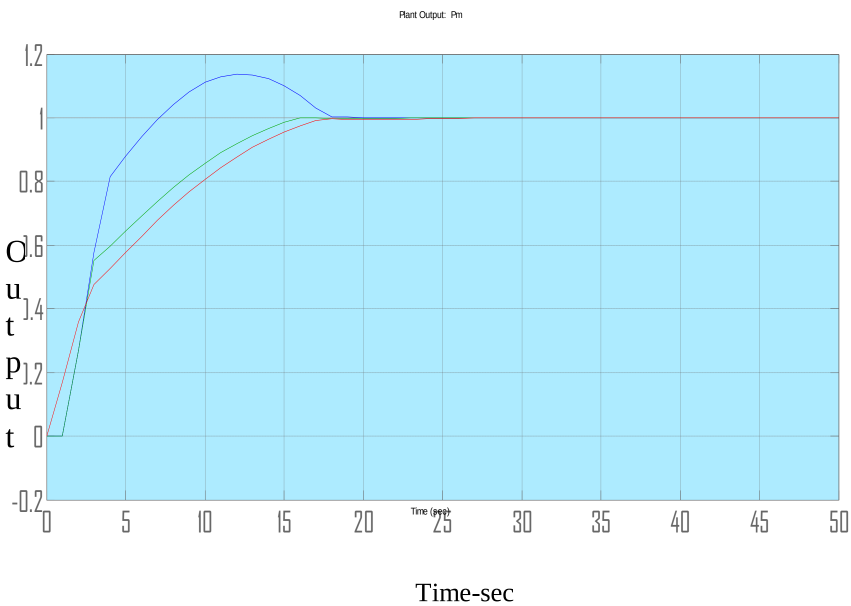


Fig-2 Plant Output when Time Horizon is varied and ΔU constraint is set between $0.1 \leq \Delta U \leq 0.1$

- Prediction horizon=2
-Prediction horizon=15
-Prediction horizon=40

B. Constraint on the Change in Input

We now change our input constraint from $0.1 \leq \Delta U \leq 0.1$ to $0.1 \Delta U \leq 0.5$ and vary the time horizon as previously done. When we relaxed the constraint on the Change in Input, we observe that the MPC controller takes advantage of this and increase the change in input to reach steady state faster.



ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2016

- Prediction horizon=2
-Prediction horizon=15
-Prediction horizon=40

V. RESULT AND DISCUSSION

Plant Output - Plant's output based on the control signal
Control Signal - Control signal from the MPC controller to the plant
Having an MPC Graphical User Interface such as this allows us to easily change parameters in the model and plant and observe the results. As expected, when the constraints on the change in input power are loosened, the performance of the controller increases. In Figure the control signal maxes out at the upper bound for ΔU for quite a long time for the shorter time horizons. By loosening that bound, the time for the output to reach steady-state decreases quite dramatically (again, just for the shorter time horizons). The longer time horizon is "smarter" in this case and knows it does not need that large of an input to reach the reference step value, but as we can see for the loosened upper bound, this also makes the settling time much slower for a longer time horizon. For a piecewise-linear reference signal with short periods of constant value, shorter time horizons work better at tracking the reference signal. Longer time horizons will make the controller want to go to the final value of the reference signal instead of following the earlier reference values. The "better" choice for a time horizon for a reference input such as this one depends solely on the application. With regards to the parameter changes in the plant, as can be seen from the plots, the controller produces the same result even for varied plants as long as the time horizon is sufficiently long. This can be seen by comparing Figures with Figure. For short time horizons (N=2), the settling time for different plants increases, because the controller cannot see far enough ahead to determine the differences it must account for. Comparing the instances where we varied the plant time constants versus when we varied the valve weighing factors, it can be seen that the results are almost identical.

VI. CONCLUSION

MPC technology has progressed steadily in the twenty two years since the firstIDCOM and DMC applications. Survey data reveal approximately 2200 applicationsto date, with a solid foundation in refining and petrochemicals, and significantpenetration into a wide range of application areas from chemicals to food processing.

An important observation is that industrial MPC controllers almost always useempirical dynamic models identified from test data. The impact of identificationtheory on process modeling is perhaps comparable to the impact of optimal controltheory on model predictive control. It is probably safe to say that MPC practice is one of the largest application areas of system identification. The current success of MPCtechnology may be due to carefully designed plant tests.Efforts towards integrating identification andcontrol design may bring significant benefits to industrial practice.

The future of MPC technology is bright, with all of the vendors surveyed herereporting significant applications in progress. Next-generation MPC technology islikely to include multiple objective functions, an infinite prediction horizon, nonlinearprocess models, better use of model uncertainty estimates, and better handling of illconditioning.

REFERENCES

[1] Model Predictive Control System Design and Implementation using Matlab by Liuping Wang.
[2] Michael Nikolaou, Model predictive controllers: A critical synthesis of theory and industrial needs,Advances in Chemical Engineering, Academic Press, 2001, Volume 26.
[3] Bequette B.W. "Process Control Modelling Design and Simulation" Prentice Hall Edition.
[4] Bemporad, A., Reducing conservativeness in predictive control of constrained systems with disturbances, Proceedings of the 37th IEEE Conference on Decision & Control, Dec. 1998.
[5] Modelling and Supervisory Control Design for a Combined Cycle Power Plant Salah g. abokhatwa andrezakatebi , industrial control centre, department of electrical and electronic engineering, university of Strathclyde, gl1qe, Glasgow, UK