IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DESIGN OF MODEL PREDICTIVE CONTROL ON CONTINUOUS STIRRED TANK REACTOR

[1]Shivanshu Nilesh Patil, [2]Onkar Madhukar Gobare, [3]Shreyas Narayan Dhadge, [4]Prof. Shashikant Rangnathrao Kale

Department of Instrumentation & Control,
AISSMS's Institute of Information Technology,
Pune-411006, India

Abstract: The continuous stirred tank reactor (CSTR) is a batch reactor equipped with an impeller or other mixing device to ensure efficient mixing. The article aims to provide steps and successful implementation of a Single input- Single output (SISO) strategy for modelling of CSTR on DeltaV DCS system. We have implemented batch and continuous strategy for CSTR using PID and Model Predictive Control (MPC). The results obtained by both techniques are compared further. By implementing this loop, we will be able to understand continuous and batch processes better. The results achieved through MPC works better than PID. Multiple input-Multiple output (MIMO) systems handling and constraints handling capacity of MPC is reliable than PID due to advanced algorithms and modelling of the plant.

***Index Terms* – CSTR, MPC, Emerson DeltaV, DCS, modelling, chemical engineering.**

I. INTRODUCTION

This project aims to design and control the Continuous Stirred Tank Reactor (CSTR) loop based on an advanced control strategy that is needed in process and chemical industries. Since every industry has adopted advanced techniques for control of nonlinear processes, the CSTR is considered the most nonlinear system, with critical process parameters like temperature, level, and flow. CSTR are reaction vessels in which reagents, reactants, and often solvents are added while the product(s) of the reaction is simultaneously removed. Tank reactors are powerful tools for continuous chemical processing in this manner. In this project we have interfaced a pilot plant like industrial CSTR, to Emerson DeltaV DCS system. Using DCS we have implemented Model Predictive Control strategy to control pressure parameter and to understand the working interacting parameters and loops in the nonlinear system.

CSTR is specifically considered a non-linear system, which exhibits highly dynamic behavior [1,2,3,4]. CSTR has led to design of different control mechanisms for handling temperature and concentration effectively. CSTR has various transfer function nonlinear mode equations which may carried out exothermic or isothermal process, thus CSTR is treated as crucial system and exhibits dynamic behavior [5,6,7,8,9]. In chemical industries, the CSTR is most widely used chemical reactor. To have more exposure towards control of nonlinear systems, many advanced control strategies have been introduced in process industries [1,2,3,4,10,11]. CSTR is a main element in the chemical industries. CSTR is backbone of all process control industries, it performs an important role in controlling the temperature of the reaction for the design purpose [5,12]. CSTR shows highly nonlinear properties and because of linear modelling and controlling is difficult in the case of CSTR [13,14].

PID and MPC are the most widely used control strategies in chemical industries [1,2,3]. The article [4] focuses on a PID type control strategy with Feedback, Feed-Forward, and Cascade type configurations implemented on temperature loop in CSTR. CSTR has led to the design of different control mechanisms for handling temperature and concentration thus, a control mechanism that would make accurate changes, hence PID controller is a correct controller to be used to control such process in a large scale [6]. The optimized GA-PID controller shows good transient response compared to that of the conventional PID controller [6]. The article [4] propose feedback type PID configuration to control temperature in CSTR.

The implementation of nonlinear PID using local linear state-space models of a nonlinear system was introduced in [2]. The article [6] focuses on implementation of PID using GA which results in good optimization and advanced control over system. In article [9] proposed method of designing PID using SOPTDZ. Thus, the time domain parameters of CSTR with GA optimized PID controller are analyzed. It is observed that the designed controller provides a better response to the CSTR plant. The article [2] proposed design method for nonlinear PID using local PID controllers on the support of local state space models representing nonlinear system. The overall output of local PID controllers is put together to take global output and give it to nonlinear PID [2].

The article [4] proposed simulations-based temperature control loops of CSTR using DCS. The purpose of the DCS is to provide easy optimization, and advance control over the system. In DCS, advance features like MPC, Fuzzy Logic are implementable and easily handled due to storage optimization. Using DCS, designing of the model predictive technique for level control of a system has been done in [15]. In article [16], the experiment suggests implementation of control technique on a nonlinear process using DeltaV DCS. The salt mixing setup used in this experiment exhibits MIMO and dynamic behavior which is perfect system to practice advance control strategies.

The article [1] presents sets of the model applied to the model predictive controller (MPC) for advanced control on CSTR. Using multiple sets of models based on MPC can make the process switch smoothly over the multiple operating safe points, making the process more efficient. MPC with a quite acceptable operating point can handle a nonlinear CSTR system smoothly [1,2]. Rather than traditional linear MPC, a non-linear MPC has always shown exceptional results and excellent performances [1], [3,11]. The article [1] suggests MPC control on the base of mixture distribution i.e., series of parallel models, to CSTR. The article [3] presents a NMPC algorithm based on stochastic differential equations (SDE) can trace desirable temperature setpoint during operation while attains stability. MPC has adaptive ability to handle outputs and varying inputs due to it optimization, disturbance handling [11].

MPC algorithm provides more reliability and fast adaptivity with comparable computation complexity to control CSTR process over a linear MPC [1]. The computation complexity is reduced in F-NMPC based model introduced in [2] compared to A-NMPC. A Nonlinear Model Predictive Control (N-MPC) introduced in [3] demonstrates absolute control over temperature parameter in CSTR. The Distributed Control System (DCS) can help in easy monitoring and optimize computation of MPC [4]. MPC performance can be enhanced along with more disturbance predictions [11]. The article [5,12,17] presents traditional way for modelling and controlling of CSTR, but as time passed away many advance control strategies comes into the picture like MPC to have a smooth and efficient command on CSTR.

The article [10] has designed a model-based controller using PDC technique via T-S fuzzy bilinear model. This model is most trendy in process industries for implementation of nonlinear MPC due to its fuzzy rules. The parameters that can be controlled under fuzzy control are Fuzzification, Knowledge Base, Fuzzy reasoning, Defuzzification [5]. Fuzzy adaptive controllers are designed with the unique backstepping method for the nonlinear MIMO ambiguous system, which resulted effectively on CSTR system with less lag in output [8]. The article [7] proposed two control strategies, optimization through ICA and gain scheduling through fuzzy logic. SSE and IAE criteria help in better outcome by which fuzzy structure implemented get more flexibility and enhanced behavior. The fuzzy based study concluded that all fuzzy observers correctly traced the fault, despite disturbances [13]. Therefore, monitoring algorithms need to identify faults quickly and accurately to implement necessary correction actions to return the system to desired handling conditions [5,12,13].

II. CONTINUOUS STIRRED TANK REACTOR INSIGHT

2.1 P&ID of the CSTR plant:

The pilot plant shown in the Fig. 1 is divided into four sections. The P&ID highlights the main loops required for controlling CSTR in Fig. 2 using DCS as well as local PID control. It also contains safety interlocks issued to the control valves CV1 and CV2 as bypass route to avoid any damage to the control valves during failure of signal.

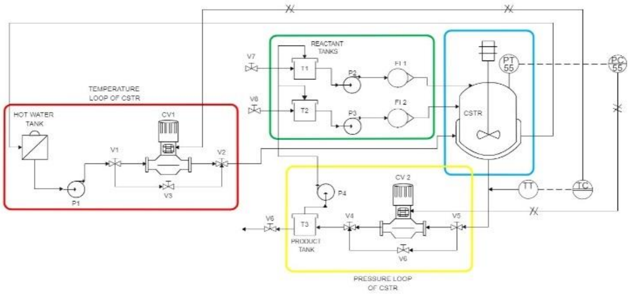

Figure 1. P&ID of Continuous Stirred Tank Reactor.

2.2 CSTR pilot plant:

Figure. 2 shows the working continuous stirred tank reactor pilot plant available in the department of Instrumentation & Control, AISSMS's Institute of Information Technology, Pune.

Figure. 2 Continuous Stirred Tank Reactor Pilot Plant.

In chemical and process industries, CSTRs are considered nonlinear systems because of their highly interacting parameters and fast dynamics. A CSTR is a reactor that incorporates an impeller or agitator for mixing purposes. Using CSTR, two or more reactants can be mixed so that their temperatures are isothermal throughout the reactor. By creating a temperature-controlled environment, the CSTR improves the quality of the final product. CSTR is useful batch as well as continuous processing of products.

2.3 Standard Operating Procedure for CSTR:

1. Make sure the hot water tank and reactant tanks are filled with the required level and drain out the reactor if any liquid is present.
2. Let the temperature and pressure inside the reactor stabilize and turn ON the heater of the hot water tank.
3. Check the bypass valves (V3 & V6) on both the CV1 and CV2 are closed and line valves (V1, V2, V4, V5) are opened.
4. Make sure the compressor is filled with the air for smooth control valve action and set the regulator of the I/P at the preferred value.
5. Turn ON the pump P2 and P3 to charge the reactants one by one in the reactor and fill the vessel with the appropriate level.
6. Release the hot water by turning ON the pump P1 into the outer jacket of the reactor.
7. Simultaneously turn ON the agitator for proper mixing and to maintain the temperature and pressure inside the reactor.
8. Monitor the temperature and pressure inside the reactor continuously to achieve the desired conditions.
9. After the desired conditions are attained, drain out the product in the product tank and let the reactor parameters stabilize for the next batch.

2.4 Input/Output Counts:

Table 1. I/O Counts for CSTR.

I/O TYPES	I/O REQUIRED	I/O AVAILABE
Analog Inputs	1	3
Analog Outputs	1	2
Digital Inputs	0	3
Digital Outputs	4	5

III. METHODOLOGY

3.1 Interfacing of CSTR plant with DeltaV DCS:

Figure 3. I/O's connection at DCS coming from loop

Figure 4. I/O's connection at loop coming from DCS.

3.1.1 Digital I/O Connections:

For triggering pumps on/off we use digital supply/output. Discrete output cards available in DCS for discrete control. Discrete means true or false, 1 or 0 etc. Here 1 corresponds to +24V and 0 means 0V. we take 24V supply from SMPS to use discrete output cards. Do cards is of 8 channels, each channel contains 2 slots. In first slot of each channel, we provide +24V supply and from 2nd slot we connect to the field device such as pumps. Here common ground is available for all discrete outputs, we connect ground of DCS to the common ground of DCS.

3.1.2 Analog I/O Connections:

For analog output, we directly connect positive terminal of AO card to positive terminal of CV and negative terminal of AO card to Negative of CV. Here, AO is also 8 channel module which includes 2 slots each. 1st slot of each channel is positive line and 2nd one is negative line. 4-20mA output signal is used to regulate or to control opening and closing of control valve. In analog input connection, we done it with 2 wire loop wired connection. In this loop connection, positive terminal of transmitter is connected to the negative terminal of AI card and negative terminal of transmitter is connected to positive terminal of AI card. Signal coming from transmitter is in 4-20mA form.

3.1.3 Batch Process for CSTR:

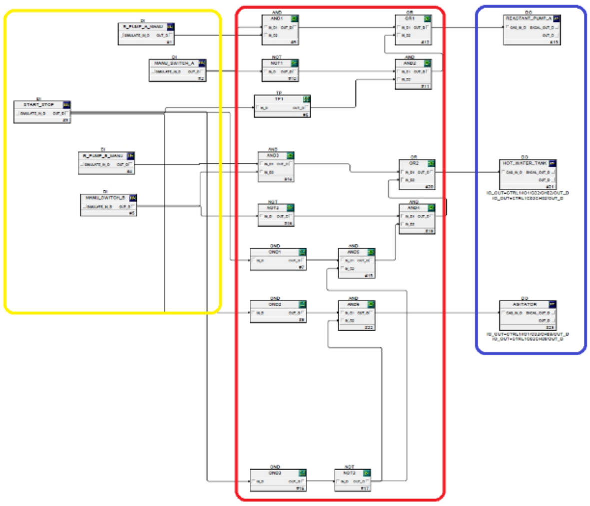

Figure 5. Batch Program of CSTR on Control Studio.

In DeltaV DCS, the logic development and FBD designing is done on Control Studio. The Figure. 5 shows the batch process created for CSTR to run for required amount of time. In this program manual override for all pumps is provided in case if needed. The yellow box contains the master start/stop button for batch-run and manual override switches for pumps. The red box contains the interlocks and timer blocks used for running the pumps for specified time. The blue box contains the discrete outputs i.e., pumps and agitator. The detailed information of blocks in Fig. 5 is shown in Table 2. below.

Table 2. Batch Program FBD blocks detail.

BLOCK NAME	PARAMETERS
START_STOP	Master button for batch run.
MANU_SWITCH_A & MANU_SWITCH_B	Manual shift switch for pump 2 & 3.
R_PUMP_A_MANU & R_PUMP_B_MANU	Manual start/stop button for pump 2 & 3.
OND	On Delay timer blocks
TP	Timer pulse block
HOT_WATER_TANK	Pump 1 (DO)
REACTANT_PUMP_A	Pump 2 & 3 (DO)
AGITATOR	Agitator (DO)

The manual override for pumps has been provided an effective interlock where the operator must first switch from auto to manual mode through button MANU_SWITCH_A & then it gives access for making motor start/stop through button R_PUMP_A_MANU. When in manual mode the operator won't be able to run the auto batch process due to interlocks assigned in the middle section using AND OR & NOT blocks.

3.1.4 PID control on CSTR:

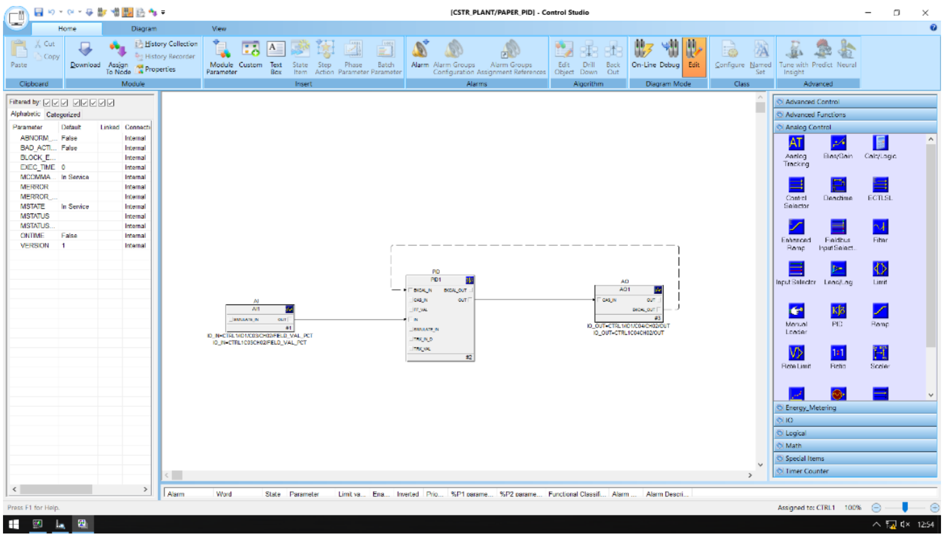

Figure 6. PID control strategy for pressure loop of CSTR.

To understand the nonlinear systems dynamic behavior and interaction between parameters PID control strategies is also implemented. In the Figure. 6 the PID control strategy for CSTR is shown with three blocks namely AI1, PID1, & AO1. The parameters considered here are pressure inside the reactor and the product line control valve i.e., Pressure loop of CSTR. The tuning of the PID done through trail and estimated error method and tuned in a such way that desired response is achieved.

Table 3. PID parameters values

PARAMETERS	VALUES
K_P (Gain)	1
K_I (Reset)	500
K_D (Rate)	0.02

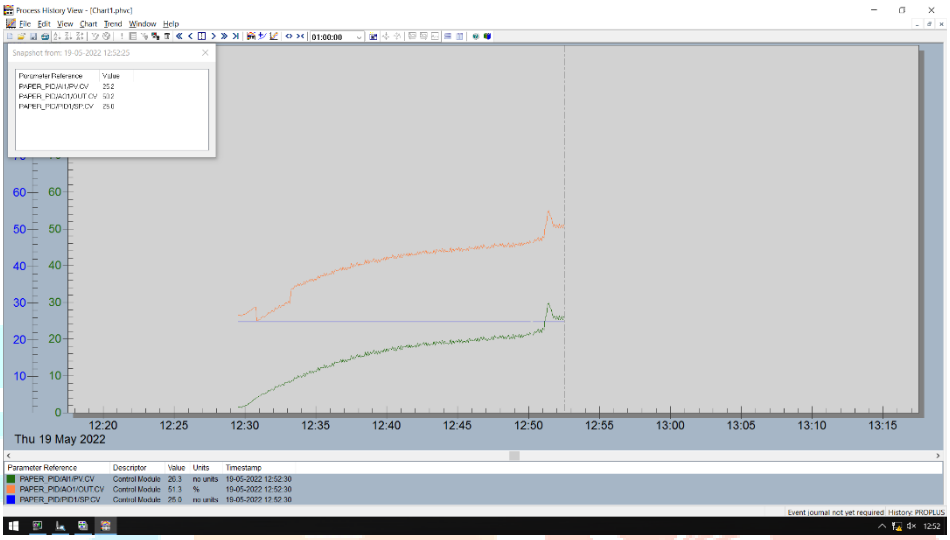

Figure 7. PID response on Process History View.

In Figure.7 PID response of the pressure loop of CSTR is shown. The graph plot is of pressure (PT) versus the control valve 2 (CV 2). Based on the response, PV takes a long time and a lot of tuning before reaching SP. PIDs are useful only for SISO systems, while in the process industry we mostly deal with MIMO, nonlinear systems. For non-linear systems, they employ cascaded PIDs that must be continuously tuned to ensure setpoints are maintained. The ideal response of PID can achieve SP in minimum time and less tuning is required.

IV. Model Predictive Control Strategy on CSTR.

The advance control strategy implementation is applied to enhance the output and performance of the plant. Generally, PID control action is employed on plants which surpassed by advanced control strategies like MPC, Neural Networks, Fuzzy Logic, etc. by using advanced controllers like DCS. MPC is model based control strategy. It uses model to predict future values of outputs and optimize control action which takes system to desired response in less time. The DeltaV Model Predictive Control (MPC) Function block allows interactive processes to be controlled within measurable operating constraints while accounting for process interaction and measurable disturbances.

This function block replaces standard PID control that utilizes feedforward, decoupling networks, and override for multivariable control. We define all parameters (manipulated, disturbance, constraints, and controlled) in the MPC function block for the desired process using Control Studio. The DeltaV Predict implements multivariable, model predictive control in DeltaV environments. Within DeltaV Predict we can run an automated test on your process after downloading the module containing the MPC function block. Data automatically gathered by the Continuous Historian during testing is used by the Predict application to create a step response model of your process. The model generated gets assigned to the controller with updated matrix used by the MPC function block.

4.1 Stages for implementation of MPC on DeltaV DCS:

Software's required for the modelling in Emerson DeltaV system are Control Studio, DeltaV Predict and MPC Operate. The modelling is recognized by the MPC block automatically as it is done through data driven step test. Following sections include steps to implement modelling using data driven technique on Emerson DeltaV system.

4.1.1 Designing a program on Control Studio:

1. Open the Control Studio.
2. Drag and drop the Analog input and output blocks from I/O Library & MPC function block from Advanced control Strategy library to development area.
3. Connect the 'OUT' parameter of the AI block to 'CNTRL1' parameter of the MPC block (input) and connect the 'MNPLT1' parameter of the MPC block to 'CAS_IN' parameter of the AO block (output).
4. Connect the 'BKCAL_OUT' parameter of the AO block to 'BKCAL_IN' parameter of the MPC block as a feedback path.

5. Assign the tags to the AI & AO block by selecting them and right click. Then select ‘Assign I/O Tag’ and choose appropriate channel tags which will be used for the modelling process. Label the blocks for better understanding.
6. Now save the program and click on ‘Assign to Node’ on top left-corner to assign the program to the controller which is going to look after the process.
7. Download the program.

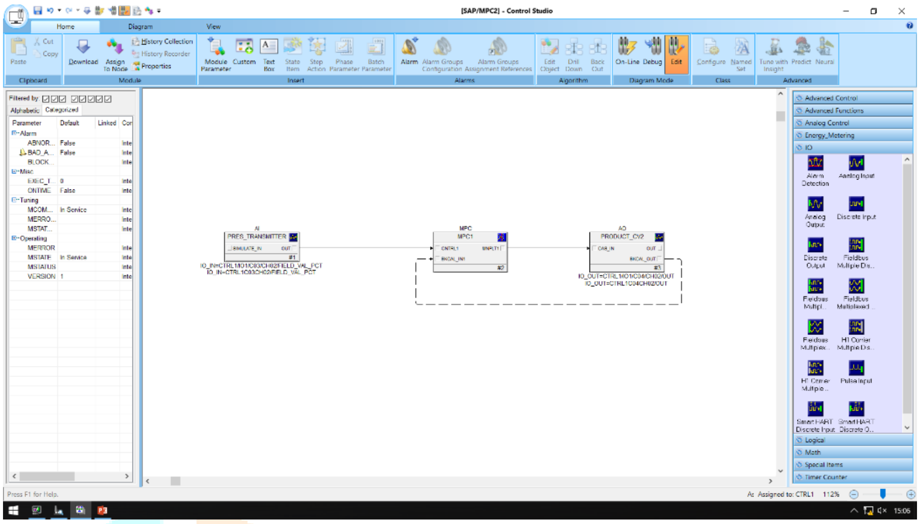

Figure 8. MPC program on Control Studio.

The Figure. 8 shows the program created on the Control Studio for MPC. The modelling is doe after the block is assigned to the controller through the module program. In this program the blocks utilised are AI (PT), MPC1, AO (CV 2).

4.1.2 Data driven Modelling on DeltaV Predict:

1. Open the DeltaV Predict application.
2. Open File & browse the module created in the control studio & select the MPC block used.
3. The selected MPC block will appear on the ‘Models’ window on the left side of the screen.
4. Verify the block used one more time and then go to ‘Test Process’ section in the bottom of the screen. Now, we must define some test criteria like Step size, steady-state time, and cycles according to our requirements
5. Set the ‘Step Size (%) MNPLT1’ to ‘20’.
6. Set the ‘Time to Steady-State’ to ‘500’ seconds. This will initiate the cycles of steps in given period.
7. Set the ‘Cycles’ to ‘5’.
8. Make sure the controlled variable ‘CV’ of system is biased with certain value so that the test doesn’t fail because of constraints applied on the output parameter.
9. To add bias value, go to ‘Operation’ window then click on the MNPLT1 and set the value to ‘xyz’ and click ‘OK’.
10. Click on the ‘Test’ and Wait for the time displayed for test cycle to complete.
11. After the timeout, go to ‘Autogeneration’ section besides and check the ‘CNTRL1’ checkbox in integrating section and then click on ‘Autogenerate’ to create the model of the data collected.
12. After step 9, the model created will be visible in ‘Models’ window. The model generated gets assigned to the MPC block automatically.

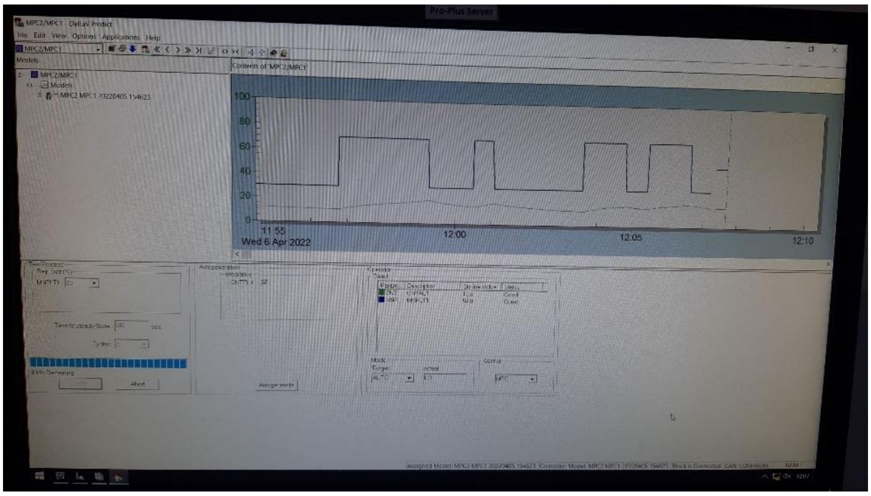

Figure 9. MPC Test Cycle on DeltaV Predict.

The Figure. 9 illustrates the data driven test cycle done on DeltaV Predict to get the Model based on the system response. The above graph containing blue and green line indicates the MNPLT1 (PV) and CNTRL1 (CV 2) respectively. After the test cycle is completed, the model is automatically assigned to the MPC in the Control Studio. Once the test is done the model created will be available in the Left window. The Figure. 10 displays the Model created on DeltaV Predict of the corresponding pressure loop of CSTR system.

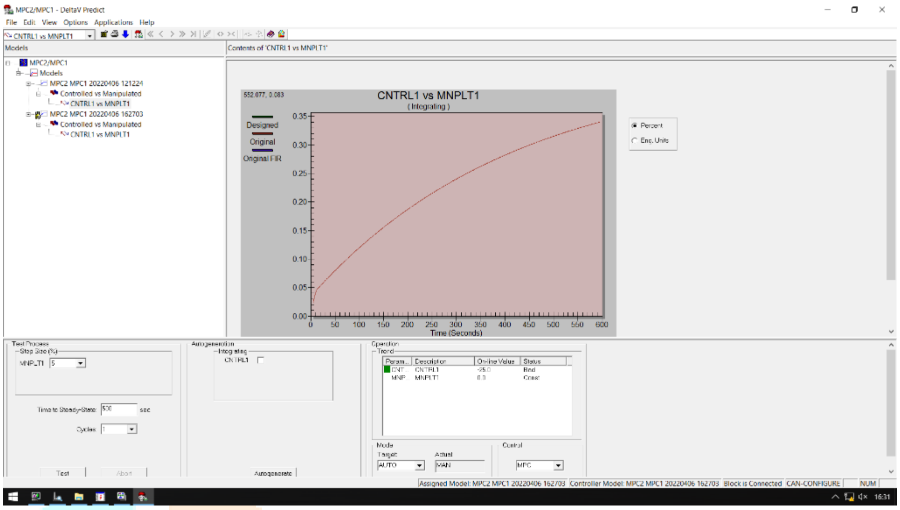

Figure 10. Model created on DeltaV Predict after test cycle.

4.1.3 Model handling using MPC Operate:

1. Go back to Control Studio and open the program containing MPC block. Click on save and download the program. After that click on 'Online'.
2. Now the program is online, select MPC block and click on 'MPC Operate' visible on the right corner of window.
3. On MPC operate we have multiple tools and settings available to operate the MPC block. The main screen contains trends showcasing all parameters i.e., setpoint, process variable, controlled variable, and sliders to change the values as well.
4. Choose the 'Setpoint' parameter and change the value to 'xyz' and observe the trends.

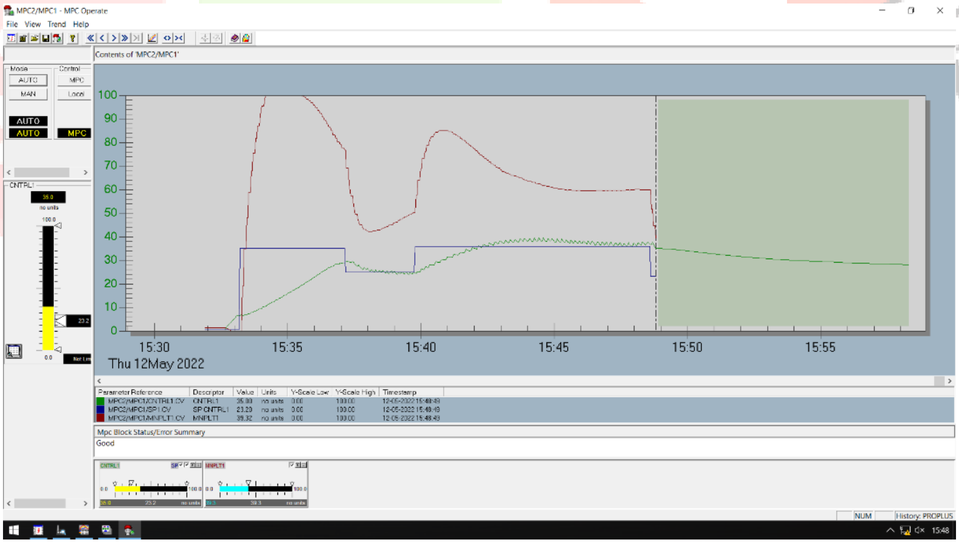

Figure 10. MPC response on MPC Operate.

The Figure. 11 displays the response achieved by MPC on pressure loop of CSTR. In the graph above the red and green lines describe the behaviour of MNPLT1 (Pressure) VS CNTRL1 (Control Valve 2). The blue line indicates the setpoint given to the MPC. The X-axis and Y-axis of the shows the time and percentage scale, respectively. The light green colour section in the graph is the future trajectory or predicted response calculated by the model.

V. Results

After comparing the PID and MPC responses in terms of their graphs plotted against the same parameters in Figure. 7 & Figure. 11 respectively, MPC control action is more dependable and does a quick action than PID. In PID action, if a setpoint is adjusted, tuning of parameters is required again. The system will take time to achieve different values which is not the case in MPC action. As illustrated in Figure. 11, the setpoint is changed several times, and the time taken by the controller to achieve the desired response is noticeably short. The main advantage of MPC action is that it represents future readings or rather tells how the system will respond. The setpoint tracking is another advantage of MPC over PID that can be seen in Figure. 11 which is represented in the green colour section. The setpoints given in both the cases are the same but due to mathematically modelling, the MPC has an advantage over

PID through setpoint tracking and it can achieve the desired response efficiently and quickly. In MPC, the constraints are managed better in both inputs and outputs which is otherwise in PID where constraints are only give to outputs.

Table 4. Comparison of PID and MPC operation.

PID	MPC
SP = 25	SP = 25
Time required to achieve SP = 15 to 20 min	Time required to achieve SP = 3 to 5 min

From the Table 4. it can be observed MPC is more reliable than PID for nonlinear system. The setpoints given for both the controllers are same yet PID takes more time to achieve desired response as compared to MPC. Because as soon as the setpoint is changed in the PID, appropriate tuning of parameters is required. That's not the case in MPC because of modelling it is easy to handle the multiple setpoint changes.

VI. Conclusion

VII. This paper concludes that the nonlinear systems in industries require advanced control techniques like MPC to manage the parameters and constraints. In MPC, after the setpoint is changed the process variable immediately controls the controlled variable to achieve desired response in less time. The MPC can also increase productivity due to fast response and runs the system efficiently i.e., total utilization of the system is done. For MIMO systems, the disturbances are managed very well by MPC whereas in the case of PID strategy, cascading of controllers is done which is complicated for tuning as well as designing. Theoretically, several PID actions have been introduced to control the MIMO systems efficiently, but it is implementable to all systems.

VIII. Acknowledgement

In order to complete this research and project, we would like to express our sincere appreciation to AISSMS's Institute of Information Technology and the Emerson Centre of Excellence Laboratory for providing us with infrastructural facilities.

References

[1] J. Pekaf and V. Havlena, "CONTROL OF CSTR USING MODEL PREDICTIVE CONTROLLER BASED ON MIXTURE DISTRIBUTION," 2004. [Online]. Available: www.elsevier.comllocate/lfac
[2] J. Prakash and K. Srinivasan, "Design of nonlinear PID controller and nonlinear model predictive controller for a continuous stirred tank reactor," ISA Transactions, vol. 48, no. 3, pp. 273–282, 2009, doi: 10.1016/j.isatra.2009.02.001.
[3] M. R. Wahlgreen et al., "Nonlinear model predictive control for an exothermic reaction in an adiabatic CSTR," in IFAC-PapersOnLine, 2020, vol. 53, no. 1, pp. 500–505. doi: 10.1016/j.ifacol.2020.06.084.
[4] M. Ponni Bala, E. B. Priyanka, K. Prabhu, P. Bharathi Priya, T. Bhuvana, and V. Cibi Raja, "Real-time performance analysis of temperature process using continuous stirred tank reactor," in 2021 International Conference on Emerging Smart Computing and Informatics, ESCI 2021, Mar. 2021, pp. 647–650. doi: 10.1109/ESCI50559.2021.9396877.
[5] M. Nazar Ismael, A. T. Nawaf, and M. A. Abdullah, "Continuous Stirred Tank Reactor using the Dual/Valve Position-Cascade Control System," 2020.
[6] Prabhu K, "Pid Controller Design For Cstr Plant Using Genetic Algorithm Based Pid Controller", [Online]. Available: www.ijstr.org
[7] M. F. ad Rahmat, A. M. Yazdani, M. A. Movahed, and S. Mahmoudzadeh, "Temperature control of a continuous stirred tank reactor by means of two different intelligent strategies," International Journal on Smart Sensing and Intelligent Systems, vol. 4, no. 2, pp. 244–267, 2011, doi: 10.21307/ijssis-2017-438.
[8] L. P. Xin, B. Yu, L. Zhao, and J. Yu, "Adaptive fuzzy backstepping control for a two continuous stirred tank reactors process based on dynamic surface control approach," Applied Mathematics and Computation, vol. 377, Jul. 2020, doi: 10.1016/j.amc.2020.125138.
[9] D. Krishna, K. Suryanarayana, G. Aparna, and R. Padma Sree, "Tuning of PID Controllers for Continuous Stirred Tank Reactors," Indian Chemical Engineer, vol. 54, no. 3, pp. 157–179, 2012, doi: 10.1080/00194506.2012.766005.
[10] M. Hamdy and I. Hamdan, "Robust fuzzy output feedback controller for affine nonlinear systems via T-S fuzzy bilinear model: CSTR benchmark," ISA Transactions, vol. 57, pp. 85–92, 2015, doi: 10.1016/j.isatra.2014.11.012.
[11] M. Pervez and T. Kamal, "Comparative Study of Modern Control Techniques for Optimal Dynamic Nonlinear Process Control," Nov. 2020. doi: 10.1109/INMIC50486.2020.9318161.
[12] V. and R. O. P. and K. V. Kumar Utsav and Sharma, "MPC-Based Temperature Control of CSTR Process and Its Comparison with PID," in Advances in Electrical and Computer Technologies, 2020, pp. 1109–1115.
[13] M. Delbari, K. Salahshoor, and B. Moshiri, "Adaptive generalized predictive control and model reference adaptive control for CSTR Reactor," in 2010 International Conference on Intelligent Control and Information Processing, 2010, pp. 165–169. doi: 10.1109/ICICIP.2010.5565238.
[14] H. Ballesteros-Moncada, E. J. Herrera-López, and J. Anzurez-Marín, "Fuzzy model-based observers for fault detection in CSTR," ISA Transactions, vol. 59, pp. 325–333, 2015, doi: https://doi.org/10.1016/j.isatra.2015.10.006.
[15] C. A. Rietz and D. K. Rollins, "Implementation of a predictive modeling technique on a DCS," in Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No.98CH36207), 1998, vol. 5, pp. 2951–2955 vol.5. doi: 10.1109/ACC.1998.688399.

[16] B. R. Y. Oung, J. H. van der Lee, and W. Y. Svrcek, "A NONLINEAR, MULTI-INPUT, MULTI-OUTPUT Process Control Laboratory Experiment," 2006.

[17] P. B. Sistu and B. W. Bequette, "Nonlinear predictive control of uncertain processes: Application to a CSTR," AIChE Journal, vol. 37, no. 11, pp. 1711–1723, 1991, doi: https://doi.org/10.1002/aic.690371114.

[18] M. Nazar Ismael, "Continuous Stirred Tank Reactor (CSTR) using Mamandi fuzzy control."