Implementation of Neural Model Predictive Control in Continuous Stirred Tank Reactor System

Ankit Kumar, Mohit Bajaj, Paras Nath Verma

Abstract—This paper analyses the usage of Discrete-time Linear Model Predictive Control in controlling a Continuous Stirred Tank Reactor System based on neural network. Two different schemes of Model Predictive control are present. To begin with, a basic Model Predictive Control based on Generalized Predictive Control is used and then a Model Predictive Control approach based on neural network. Simulation results have been included which demonstrate the performance of both controllers when used to control Single-Input Single-Output Continuous Stirred Tank Reactor System and the performance when Neural Network based Model Predictive Control is applied. In order to obtain a more accurate process description the neural model is trained with data from simulation of a phenomenological model and afterwards, is employed with actual plant data. Here, this strategy permits to carry out the training without to introduce disturbance in the real plant. Artificial Neural networks of different sizes are trained. The performance of a model predictive control based on the neural model is evaluated for disturbance rejection. The achieved results allow us to conclude that the developed neural model predictive control is adequate to control effectively.

Index Terms—Artificial Neural Network(ANN), Continuous stirred tank system(CSTR), Model Predictive Control(MPC), Plant, Simulation, States of process, weight vector.

---

1 INTRODUCTION

From the last few decades, An Artificial Neural Networks (ANN) have brought the valuable tools for modelling, identification and control of non linear systems. It have been implemented in control system developments for a wide variety of non-linear complex systems, mainly when a physical model, that represents adequately the real behaviour of the process, is not available. ANN shows very important advantages regarding other technologies when they are applied to nonlinear systems. This carry out a parallel and distributed processing of the information and are having the ability not to affected or harmed by the faults. Control systems that use ANN can be easily implemented, can operate simultaneously with qualitative and quantitative data, may be applied to systems with several inputs and outputs and are arbitrarily able to approximate non-linear functions[1]. From the starting of ninety's the ANN have been implemented in control area. Such characteristics allow it to design a simple network structure, in order to avoid extensive and unnecessary computations and facilitate the application on real time control.

In recent years, the requirements for the quality of automatic control in the process industries increased significantly due to the increased complexity of the plants and sharper specifications of product quality. At the same time, the available computation power increased to a very high level. As a result, computer models that are becoming computationally expensive became applicable even to rather complex problems. Model-based control techniques were developed to obtain better control. Model predictive control was introduced successfully in several industrial plants. A good advantage of such control schemes is the ability to handle constraints of actuated variables and internal variables. In most applications of model predictive techniques, a linear model is used to forecast the process behaviour over the horizon of interest [2],[3].

As most real processes show a nonlinear behaviour, some work was done to extend predictive control techniques to incorporate nonlinear models [4],[5],[6]. The most expensive part of the realization of a nonlinear predictive control scheme is the derivation of the mathematical model. In various cases it is not possible to obtain a suitable physically founded process model due to the complexity of the underlying processes or the lack of knowledge of critical parameters (as, e.g., temperature- and pressure-dependent mass transfer coefficients or viscosities) of the models. A promising way to overcome these problems is to use neural networks as nonlinear black-box models of the dynamic behaviour of the process [7],[8],[9]. Such neural network

- *Ankit Kumar is currently pursuing masters degree program in Control & Instrumentation engineering in MNNIT,Allahabad,India, Mob-09450746384. E-mail: kumarszannat@gmail.com*
- *Co-Author Mohit Bajaj is currently pursuing masters degree program in Power Electronics & Asic Design engineering in MNNIT Allhabad,India, Mob-09793892610. E-mail: thebestbajaj@gmail.com*
- *Co-Author Paras Nath Verma is currently pursuing masters degree program in Control & Instrumentation engineering in MNNIT Allhabad,India, Mob-09410475885. E-mail: paras84ece@gmail.com*

models may be derived from measured input output data of the plant. In many practical cases, however, conventional controllers (P/PI/PID-controllers) are already in use at the plant which stabilize the plant and provide some basic, sometimes sluggish control. As we demonstrate, measurements of input output variables of the plant operated with the linear controller may provide very good training data for the neural network. The approach is more practical (here plant is always under automatic control) and more effective than using experiments without control (open-loop). Model Predictive Control (MPC) has made a significant impact on modern control engineering. It has found a wide range of applications in the process, chemical and food processing industries.

Model Predictive Control refers to a specific procedure in controller design from which many kinds of algorithms can be developed for different systems, it may linear or nonlinear, discrete or continuous. The main difference in the various methods of MPC is mainly the way the control problem is formulated. One of the most popular methods of MPC is Generalized Predictive Control (GPC). GPC was developed by Clarke [10]. The idea of GPC is to calculate future control signals in such a way that it minimizes a cost function defined over a prediction horizon. GPC is capable of controlling processes with variable dead-time, unstable and non-minimum phase systems. In this work, Discrete-time Model Predictive Control (DMPC) is used to control the concentration of a nonlinear Continuous Stirred Tank Reactor (CSTR) System in MATLAB, Simulink environment. At first, Model Predictive Control based on Generalized Predictive Control [11] which is a restricted model approach, is employed. Then a different approach using neural network is used. Artificial Neural Network (ANN) when used with DMPC[4] have many benefits such as, the number of terms used in the optimization problem can be reduced to a fraction of that required by the basic procedure, allows substantial improvements in feasibility [7], two explicit tuning parameters can be used for tuning the closed loop performance with ease and For Multi-Input and Multi-Output (MIMO) configuration both of these tuning parameters can be selected independently for each input. Finally, simulation results are given to demonstrate the performance achieved when both approaches are applied to Single-Input and Single-Output (SISO) nonlinear CSTR System. Also, the DMPC using Neural Network approach can be applied to MIMO nonlinear CSTR System.

2 Model Predictive control

Model Predictive Control(MPC), is advanced method of process control that has been in use in the process industries such as chemical plants and oil refineries since the 1980. Model predictive controllers based on dynamic models of the process, most often models are obtained by system identification. The models used in Model Predictive Control are generally intended to represent the behaviour of complex dynamical systems. The additional difficulty of the Model Predictive Control algorithm is not generally needed to provide adequate control of less complex systems, which are generally controlled well by PID controllers. General dynamic characteristics that are complex for 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 independent variables. Often in a chemical process, independent variables that can be adjusted by the controller are often either the set points of regulatory PID controllers (pressure, flow, temperature, etc.) or the final control element (valves, dampers, etc.). MPC uses the current plant measurements, the current varying state of process, the Model Predictive Control models, and the process variable targets and limits to calculate future changes in the dependent variables. Such changes are calculated to hold the dependent variables close to target while honoring constraints on both independent and dependent variables.

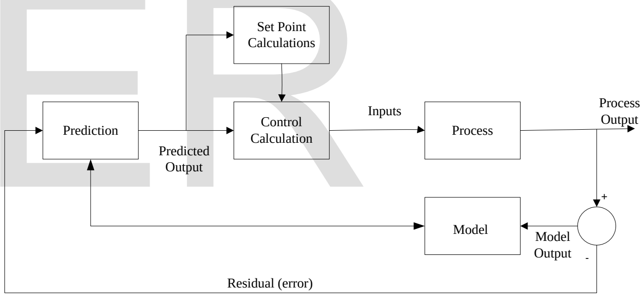

Fig. 1. Block Diagram of Model Predictive Control

The Model Predictive Control typically sends out only the prime change in each independent variable to be applied, and repeats the calculation when the next variation is required. While many real processes are not linear, often, they can be considered to be approximately linear over a small operating range. When linear models are not sufficiently accurate to represent the real process non linearities, several approaches may be used. The nonlinear model can be in the form of an empirical data fit (e.g. artificial neural networks) or a high-fidelity dynamic model based on fundamental mass and energy balance system. The nonlinear model can be linearized to derive a Kalman filter or specify a model for linear MPC. MPC is based on iterative, finite horizon optimization of the plant model. At time t, the current plant state can be sampled and a cost minimizing control strategy is computed (via a numerical minimization algorithm for a

ISSN 2229-5518

relatively short time horizon in the future: $[t, t+T]$. Specifically, the online calculation is used to explore state trajectories that emanate from the current state and find (via the solution of Euler-Lagrange equations) a cost-minimizing control strategy until time $[t+T]$.Only the first step of the control strategy is implemented and then the current(plant) state is sampled again and the calculations are repeated starting from the now current state, obtaining a new control and new predicted state path, and prediction horizon continuously keeps being shifted forward and for this reason MPC is also called receding horizon control[12]. MPC is a multivariable control algorithm which uses:

1. an internal dynamic model of the process
2.a history of past control moves and
3.an optimization cost function J over the receding prediction horizon,

To calculate the optimum control moves. The optimization cost function for MPC is given by:

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

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

Nonlinear Model Predictive Control, is a variant of model predictive control (MPC) that is characterized by the use of nonlinear system models in the prediction[13]. Similar to linear MPC, NMPC requires the iterative solution of optimal control problems on a finite prediction horizon. Although, such problems are convex in linear Model Predictive Control, in nonlinear Model Predictive Control these are not convex further. This brings challenges for both, Nonlinear Model Predictive Control stability theory and numerical solution. The numerical solution of the NMPC optimal control problems is typically based on direct optimal control methods using Newton-type optimization schemes, here, in one of the variants: direct single shooting, direct and multiple shooting methods, or direct collocation.

Nonlinear Model Predictive Control algorithms typically exploit the fact that consecutive optimal control problems are similar to each. This permits to initialize the Newton-type solution procedure efficiently by a suitably shifted guess from the previously computed optimal solution, saving a lot of amount of computation time. Nonlinear Model Predictive Control is continuously applied to more and more applications with high sampling rates, e.g., in the automotive industry etc, even when the states of the process are distributed in space (Distributed parameter systems)[14].

Objectives of the Model Predictive Control:
1. Present violations of input and output constraints.
2. Drive some output variables to their optimal set points , while maintaining other outputs within specified range.
3. Prevent excessive change of input variables.
4. Control as many process variables , when a sensor / actuator is not available.

In equally constraints on input and output variables, i.e. upper and lower limits ,can be included in both calculations –SP calculation , control calculations . This is a unique feature of MPC , and suitable for MIMO control problems. Optimizing function objective may include:
1.Maximising a profit function
2. Minimizing a cost function
3.Maximising production rate.

Optimum values of SP are changed due to varying process constraint changes:
1.Variation in process condition
2.Equipment
3.Instrumentation
4.Economic data prices and optimal cost.

In MPC , SPs are calculated each time with control calculation: Determine a sequence of control moves (manipulated value), such that the predicted response is close to the SP in an optimal manner. Control calculations are based on optimizing an objective function, based on a set of P predicted outputs. Determines M values of input

$$\hat{y}(k+1), \qquad i = 1, 2, ..., P$$

$$And,\ u(k+i-1), \qquad i=1,2,............M$$

Input is held constant after M moves , control horizon whereas P is prediction horizon. In Reducing Horizon approach, Although a sequence of M control moves is calculated at each sampling instant, after new

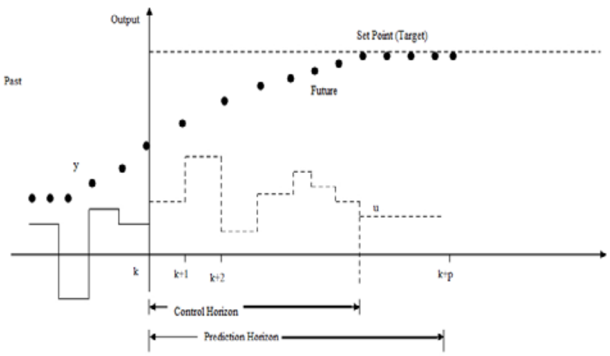

Fig. 2. Discrete Model Predictive Control scheme

measurement become available, only the first input move is implemented and this procedure is repeated at each sampling instant.

2.1 Predictions For SISO Models

In industrial applications of MPC, it has been a practice to have discrete –time, linear ,empirical models i.e. difference equation form or step response model. Step response models represent stable processes with unusual dynamic behaviour , which may not be described by simple T.F. model. But they may require large member of model parameters. Let step response model for a SISO process:

$$y(\mathrm{k}+1) = y_0 + \sum_{i=1}^{N-1} S_i \Delta u(\mathrm{k}-\mathrm{i}+1) + S_N u(\mathrm{k}-\mathrm{N}+1) \quad (2)$$

$\Delta u(\mathrm{k}-\mathrm{i}+1) =$ change in manipulated output over the instant

$= u(\mathrm{k}-\mathrm{i}+1) - u(\mathrm{k}-\mathrm{i})$

Where, y and u are deviation variables. Model parameters are

S_1 to S_N ; $30 \le N \le 120$ and $y(0) = y_0$ (may be zero)

if k is current sampling instant.

$$\hat{y}(\mathrm{k}+1) = \sum_{i=1}^{N-1} S_i \Delta u(\mathrm{k}-\mathrm{i}+1) + S_N u(\mathrm{k}-\mathrm{N}+1) \quad (3)$$

$$= S_1 \Delta u(\mathrm{k}) + \sum_{i=2}^{N-1} S_i \Delta u(\mathrm{k}-\mathrm{i}+1) + S_N u(\mathrm{k}-\mathrm{N}+1) \quad (4)$$

effect of current control action — effect of past control action

I term shows effect of current input because:

$\Delta u(\mathrm{k}) = u(\mathrm{k}) - u(\mathrm{k}-1)$

II term shows the effect of past input, u(i) , i < k

for 2 -step ahead prediction:

k=k′+1 and

$$\hat{y}(\mathrm{k}'+1) = \sum_{i=1}^{N-1} S_i \Delta(\mathrm{k}'-\mathrm{i}+2) + S_N u(\mathrm{k}'-\mathrm{N}+2) \quad (5)$$

valid for all positive k′

without loss of generality , we replace k by k and expand RHS:

$$\hat{y}(\mathrm{k}+2) = S_1 \Delta u(\mathrm{k}+1) + S_2 \Delta u(\mathrm{k}) + \sum_{i=3}^{N-1} S_i \Delta u(\mathrm{k}'-i+2) + S_N u(\mathrm{k}-N+2) \quad (6)$$

on RHS : I term - effect future control action

II term - effect current control action

III & IV terms - effect past control action.

An analogous derivation , by extension , provides an expression for j -step ahead prediction , j being arbitrary positive integer

$$\hat{y}(\mathrm{k}+j) = \sum_{i=1}^{j} S_i \Delta u(\mathrm{k}+j-1) + \sum_{i=j+1}^{N-1} S_i \Delta u(\mathrm{k}+j-1) + S_N u(\mathrm{k}+j-N) \quad (7)$$

effect of current and future control action — effect of past control actions or predicted unforced response

$$\hat{y}^0(\mathrm{k}+j) = \sum_{i=j+1}^{N-1} S_i \Delta u(\mathrm{k}+j-1) + S_N u(\mathrm{k}+j-N) \quad (8)$$

And, $$\hat{y}(\mathrm{k}+j) = \sum_{i=1}^{j} S_i \Delta u(\mathrm{k}+j-1) + \hat{y}^0(\mathrm{k}+j) \quad (9)$$

2.2 Extension Of MPC Calculation S, Based On Multiple Predictions:

Assume

$$\hat{\mathrm{Y}}(\mathrm{k}+1) = [\hat{y}(\mathrm{k}+1) \quad \hat{y}(\mathrm{k}+2) \quad \hat{y}(\mathrm{k}+\mathrm{P})]^{\mathrm{T}} \quad (10)$$

vector of predicted responses for next P sampling instants.

$$\hat{\mathrm{Y}}^0(\mathrm{k}+1) = [\hat{y}^0(\mathrm{k}+1) \quad \hat{y}^0(\mathrm{k}+2) \quad \hat{y}^0(\mathrm{k}+\mathrm{P})]^{\mathrm{T}} \quad (11)$$

$$\Delta U(\mathrm{k}+1) = [\Delta u(\mathrm{k}) \quad \Delta u(\mathrm{k}+1) \quad \Delta u(\mathrm{k}+\mathrm{M}-1)]^{\mathrm{T}} \quad (12)$$

Control horizon M and prediction horizon P are design parameters. In general $M \le P$, $P \le N + M$. So that

$$\hat{\mathrm{Y}}(\mathrm{k}+1) = S \Delta U(\mathrm{k}) + \hat{\mathrm{Y}}^0(\mathrm{k}+1) \quad (13)$$

Where,

$$S = \begin{bmatrix} S_1 & 0 & 0 & - & 0 \\ S_2 & S_1 & - & - & - \\ - & - & - & - & - \\ S_M & - & - & - & S_1 \\ S_{M+1} & - & - & - & S_2 \\ - & - & - & - & - \\ S_P & S_{P-1} & - & - & S_{P-M+1} \end{bmatrix}$$

Equation (13) do not make use of latest measurement. Since cumulative effects of model inaccuracy and unmeasured disturbances may lead to inaccurate prediction. To improve prediction , latest measurement can be used with strategy termed as output feedback. Add a bias connection , b (k+j) so that , corrected prediction:

$$\tilde{y}(\mathrm{k}+j) = \hat{y}(\mathrm{k}+j) + \mathrm{b}(\mathrm{k}+j) \quad (14)$$

$\hat{y}(\mathrm{k}+j)$: incorrect prediction

And, $$\mathrm{b}(\mathrm{k}+j) = y(\mathrm{k}) - \hat{y}(\mathrm{k}) \quad (15)$$

= residual /estimated disturbance

By substituting in (14):

$$\hat{\mathrm{Y}}(\mathrm{k}+1) = S \Delta U(\mathrm{k}) + \hat{\mathrm{Y}}^0(\mathrm{k}+1) + [y(\mathrm{k}) - \hat{y}(\mathrm{k})]\mathrm{I} \quad (16)$$

where I is P dimensional unity column vector. and , thus a correction is made for all P Prediction

$$\tilde{\mathrm{Y}}(\mathrm{k}+1) = [\tilde{y}(\mathrm{k}+1) \quad \tilde{y}(\mathrm{k}+2) \quad \tilde{y}(\mathrm{k}+p)] \quad (17)$$

3 Neural Model Predictive Control For Continuous Stirred Tank Reactor (CSTR) System

In this section a procedure for constructing a neural network model predictive controller for the control problem is presented. Here we adopt a procedure in which the controller is trained directly to minimize the cost for a training data set, without having to compute the optimal MPC control signals by off-line optimizations.

The controller is represented as

$$\Delta u(\mathrm{k}) = f(I(\mathrm{k}); w) \quad (18)$$

Where $f(I(\mathrm{k}); w)$ is a function approximator, $I(\mathrm{k})$ denotes the information which is available to the controller at time instant k, and w denotes a vector of approximator parameters (neural network weights). If complete state information is assumed, i.e., $I(\mathrm{k}) = I_{MPC}(\mathrm{k})$, the controller (18) can be considered as a functional approximation of the optimal MPC strategy. The approach studied here is, however, not restricted to controllers with full state information, and typically the set $I(\mathrm{k})$ is taken to consist of a number of past inputs $u(\mathrm{k}-i)$ and outputs $y(\mathrm{k}-i)$ as well as information about the set point or reference trajectory $y_r(\mathrm{k}-i)$ [15].

Set the random value for $I(\mathrm{k})$.

Note: Besides allowing for controllers of reduced complexity the controller structure may be fixed as well by imposing a structure on the mapping $f_N(.)$. For example, assuming that the information has the decomposition

$$I(\mathrm{k}) = [I_1(\mathrm{k}), I_2(\mathrm{k}), \ldots\ldots, I_r(\mathrm{k})] \quad (19)$$

A decentralized controller :

$\Delta u_i(\mathrm{k}) = f_{N,i}(I_i(\mathrm{k}), w_i),\ \ i = 1, \ldots\ldots, r$ is obtained by requiring that the controller has the structure

$$f_N(I(\mathrm{k}), w) = [\, f_{N,1}^{\mathrm{T}}(I_1(\mathrm{k}), w_1),\ f_{N,2}^{\mathrm{T}}(I_2(\mathrm{k}), w_2), \ldots\ldots\ldots,\ f_{N,r}^{\mathrm{T}}(I_r(\mathrm{k}), w_r)]^{\mathrm{T}} \quad (20)$$

For determination of controller parameters w in such a way that the control law (18) minimizes the cost it is required that the cost is minimized for a set of training data,

$$V^{(m)}(\mathrm{k}) = \{x^{(m)}(\mathrm{k}), u^{(m)}(\mathrm{k}-1), y_r^{(m)}(\mathrm{k}+1), \ldots., y_r^{(m)}(\mathrm{k}+N)\},$$
$$\text{where } m = 1, 2, \ldots., M \quad (21)$$

Using the control strategy (18), the system evolution for the
initial state x(m)(k) is given by

$$x^{(m)}(i+1) = g(x^{(m)}(i), u^{(m)}(i)) \quad (22)$$

$$\Delta u^{(m)}(i) = f_N(I(\mathrm{i}), w) \quad (23)$$

$$y^{(m)}(\mathrm{i}) = h(x^{(\mathrm{m})}(\mathrm{i})), \quad i = k, k+1, \ldots. \quad (24)$$

Define the associated cost associated with the training data (21),

$$J_N^{(m)}(w) = \sum_{i=k}^{k+N-1} [(y^{(m)}(i+1) - y_r^{(m)}(i+1))^{\mathrm{T}} Q (y^{(m)}(i+1) - y_r^{(m)}(i+1))$$
$$+\Delta u^{(m)}(i)^{\mathrm{T}} R \Delta u^{(m)}(i)] + q_N(x^{(m)}(k+N)) \quad (25)$$

The training of the function approximator (18) now consists of solving the nonlinear least-squares optimization problem

$$\min_w \sum_{m=1}^{M} J_N^{(m)}(w)$$

subject to the constraints

$$f_x(x^{(m)}(i+1)) \leq 0 \quad (26)$$

$$f_u(u^{(m)}(i)) \leq 0 \quad (27)$$

$$f_{\Delta}(\Delta u^{(m)}(i)) \leq 0, \quad i = k, k+1, \ldots., k+N-1 \quad (28)$$

4 PERFORMANCE OF THE CONTROL SYSTEM

The dynamic model of the Continuous Stirred Tank Reactor system is given by

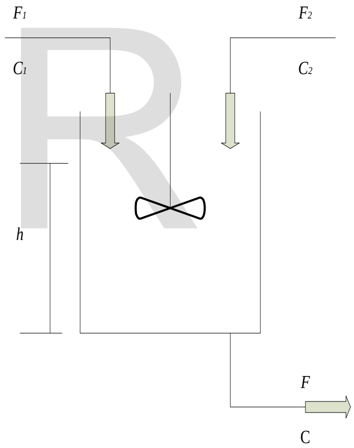

Fig. 3. Representation of Continuous Stirred Tank Reactor

$$\frac{dh(t)}{dt} = F_1(t) + F_2(t) - 0.2\sqrt{h(t)} \quad (29)$$

$$\frac{dC(t)}{dt} = (C_1 - C(t))\frac{F_1(t)}{h(t)} + (C_2 - C(t))\frac{F_2(t)}{h(t)} - \frac{k_1 C(t)}{(1 + k_2 C(t))^2} \quad (30)$$

Where $h(t)$ is the liquid level, $C(t)$ is the product concentration at the output of the process, $F_1(t)$ is the flow rate of the concentrated feed C_1, and $F_2(t)$ is the flow rate of the diluted feed C_2. The input concentrations are set to $C_1 = 24.9$ and $C_2 = 0.1$. The constants associated with the rate of consumption are $k_1 = 1$ and $k_2 = 1$.

The objective of the controller is to maintain the product concentration by adjusting the flow $F_1(t)$. To simplify the demonstration, set $F_2(t) = 0.1$. The level of the tank h(t) is not controlled for this experiment.

4.1 Simulation Results

The values of the all parameters are mentioned in the above section and all the simulation results are based on the prescribed values of the parameter.

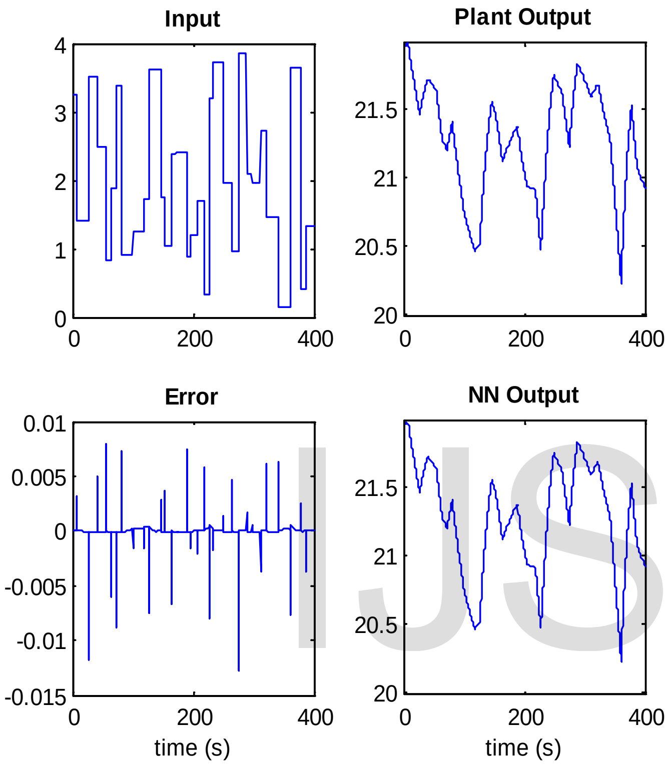

Fig. 4. Validation data for Neural Network Predictive Control

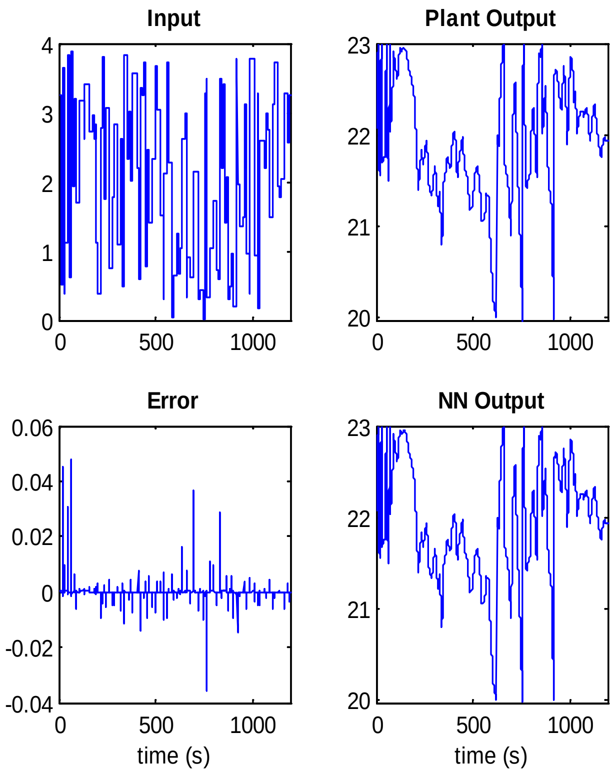

Fig. 5. Training data for Neural Network Predictive Control

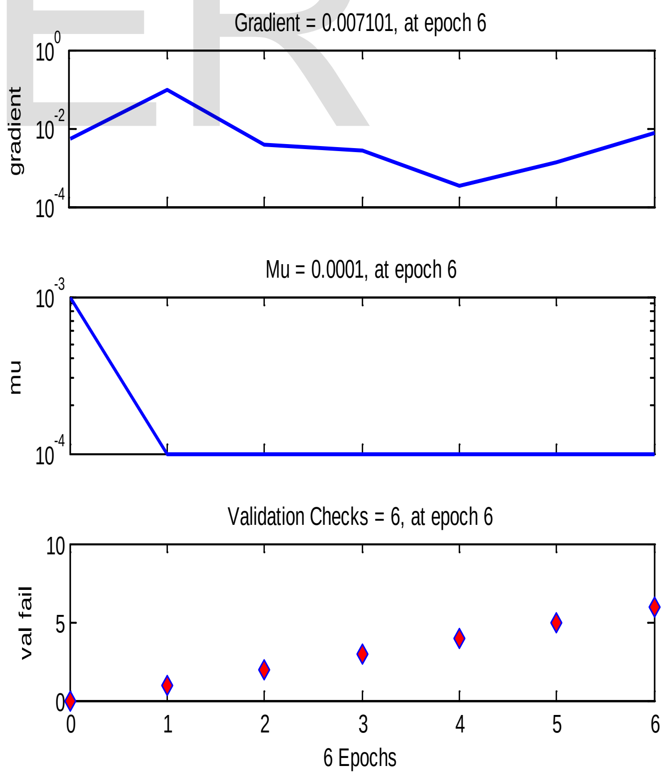

Fig. 6. Training States

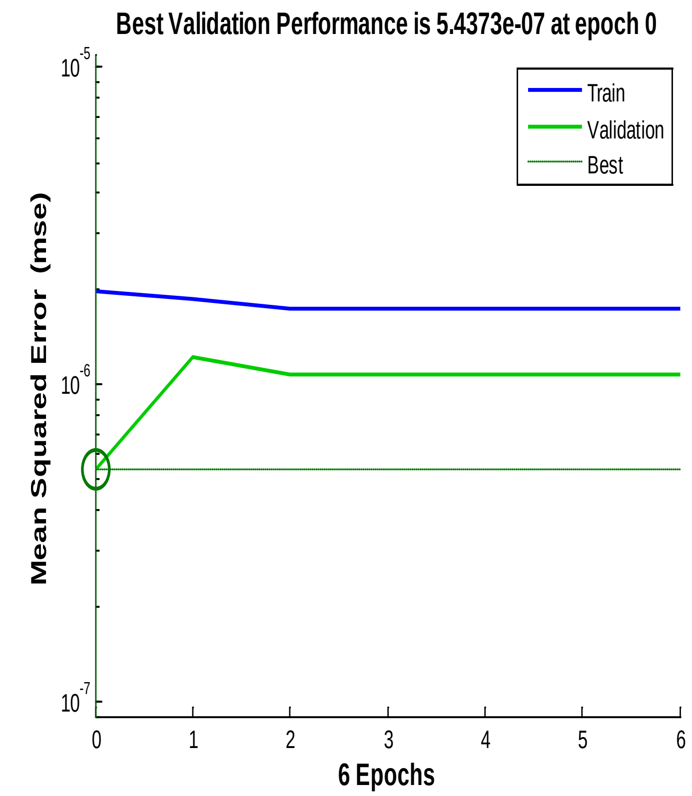

Fig. 8. Training Performance

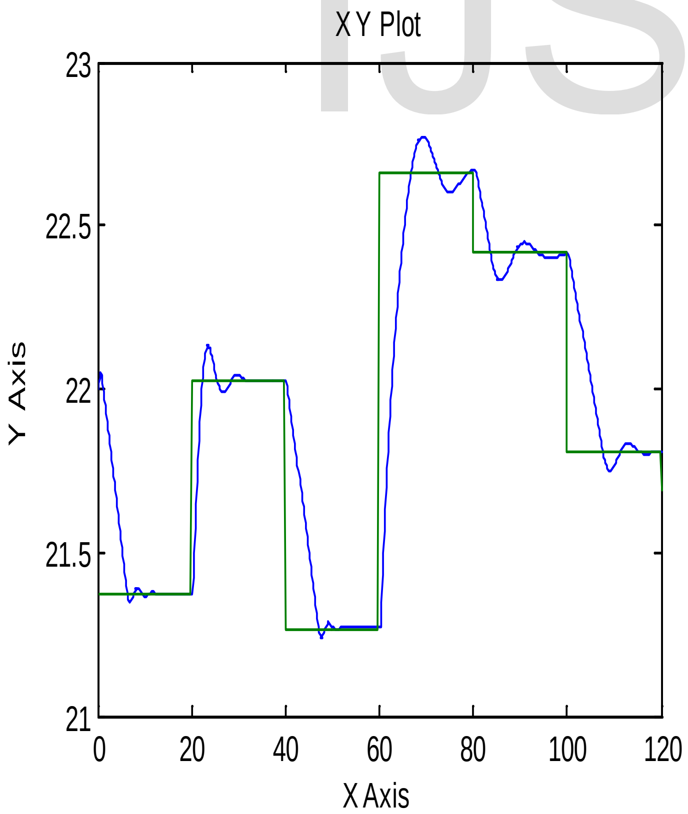

Fig. 9. Output of the system

5 Conclusion

The most significant aspect of this paper is the reduction in computational time. The model predictive control method involves highly mathematical computations. What is more, predictors based on artificial neural networks significantly increase computational demands of the MPC controllers. Nevertheless, the neural network based Model Prectiction Control provides very interesting way how, to reduce computational costs, because the training times of networks are incredibly short. This kind of artificial neural network could be promising for on-line adaption of the predictor in case of dynamic systems.

REFERENCES

[1] Graciela I. Suárez, Oscar A. Ortiz, Pablo M. Aballay, Nelson H. Aros, "Adaptive Neural Model Predictive Control for the Grape Juice Concentration Process" pp .978-1-4244-5697-0/10

[2] C. Cutler and B. Ramaker, "Dynamic Matrix Control: A Computer Control Algorithm," *Pror. 1980 Joint Autoniatic Coritrol Conference.*

[3] C.E. Garcia and M. Morari, "Internal Model Control. I . A Unifying Review and Some New Results," *hid. Eng. Chem. Process Des. Dei:* 2 1. pp.308-323, 1982

[4] T. Peterson. E. Hemandez. Y. Arkun, and F.J. Schork, "Nonlinear Predictive Control of a Semi Batch Polymerisation Reactor by an Extended DMC."*P roc. I989 American Control Conference*, pp. 1534- 1539. 1989.

[5] Liu, G.P., "*Nonlinear identification and control- A neural network approach*," Springer - Verlag London Ltd., London..

[6] D.D. Brengel and W.D. Seider. "MultistepNonlinear Predictive Control," *lid. Clzem. Eng. Res. 28*, pp. 18 12- 1822. 1989.

[7] J. Thibault and B.P.A. Grandjean. "Neural Networks in Process Control: A Survey," *Advanced Control* of *Chemical Processes*. K. Najim and E. Dufour. eds.. IFAC Symposium Series No. 8, pp. 251.260.

[8] B. Khan and J. A. Rossiter, "Triple mode or laguerre mpc: acomparison,"American Control Conference, 2011.

[9] J MacMuiTay and D Himmelblau, "Identitication of a Packed Distillation Column for Control via Artificial Neural Network "*proceedings of the American Control* ,San Francisco CA. pp 1455-1459,1992

[10] D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictivecontrol-part 1: The basic algorithm," *Automatica*, 137-148 vol.23, 1987.

[11] Tan, K.K., Lee, T.H., Hunag, S.N. & Leu, F.M., "PID control design based on a GPC approach," *Industrial & Engineering Chemistry Research* 41(8), 2002.

[12] Michael Nikolaou," Model predictive controllers: A critical synthesis of theory and industrial needs", *Advances in Chemical Engineering, Academic Press*, 2001, Volume 26, Pages 131-204

[13] An excellent overview of the state of the art (in 2008) is given in the proceedings of the two large international workshops on NMPC, by Zheng and Allgower (2000) and by Findeisen, Allgöwer, and Biegler(2006).

[14] M.R. García; C. Vilas, L.O. Santos, A.A. Alonso (2012). "A Robust Multi-Model Predictive Controller for Distributed

Parameter Systems". *Journal of ProcessControl* 22 (1): 60–71. doi:10.1016/j.jprocont. 2011.10.008

[15] Bernt M. A, kesson, Hannu T. Toivonen," A neural network model predictive controller" *2006 Elsevier Ltd* .doi:10.1016/j.jprocont.2006