

Optimization of a Temperature Control Loop using Self Tuning Regulator

K. Prabhu,
Kongu Engineering College,
Perundurai,

V. Murali Bhaskaran, PhD.
Professor,
Paavai College of Engineering, Namakkal

ABSTRACT

Continuous Stirred Tank Reactor (CSTR) plays an important role in the process industries. It helps for maintaining the temperature of the liquid in the reactors. This paper deals with the comparison of adaptive control and conventional PID control in CSTR process. In the adaptive control, Model Reference Adaptive Control (MRAC) and Self-Tuning Regulator (STR) methods are used. The Recursive Least-Square algorithm (RLS) gives the process parameters and Minimum Degree Pole Placement (MDPP) gives the controller parameters and is used to obtain the Control law. This paper illustrates how well the MDDP and RLS algorithms work. The S-function simulation is made using MATLAB codes and the results were analysed. Simulation results shows that the closed loop response of adaptive control has a better performance, compared with the conventional PID controller. This adaptive control method is applied to the Continuous Stirred Tank Reactor for maintaining the liquid temperature inside the reactor.

Keywords: Continuous Stirred Tank Reactor, Model reference adaptive control, Minimum Degree Pole Placement, PID controller, Recursive Least-Square algorithm, Self-Tuning Regulator.

1. INTRODUCTION

Most process especially chemical processes are nonlinear in some aspects. The process varies with load or can change with time, which is non-stationary. The above aspects require the settings of the controller revised and also we need a control mechanism that will improve the response affected by some disturbances. In the last decades, a vast development has been made in the field of adaptive control techniques. A brief review of various methods is presented here. Two important approaches made to design the control structure are Conventional PID, and MRAC [1]. A key problem in the design of systems with PID is to find the controller gain values. This is normally done based on the tuning of controllers. It is very difficult to guarantee the stability since the gain values get changed due to the process dynamics. Next, it is decided to propose the method called adaptive control. This method automatically detects the changes due to environmental conditions and readjusts the control mode settings, thereby optimizing the response of control loop. In that, MRAC is used to maintain the liquid temperature of a continuous stirred tank reactor. The model reference adaptive control (MRAC) was originally proposed by Whitaker at MIT. He considered a problem where the specifications were given in terms of a reference model which tells how the process output ideally should respond to the command signal. The following parameter adjustment mechanism, called the "MIT-rule", was used in the Whitaker's original MRAC.

This method performs well when compared to Conventional PID [2]. The MRAS attempts to adjust the parameters so that the correlation between the error e and the sensitivity derivatives becomes zero. The MRAC called a direct scheme because the regulator parameters are updated directly. But the problem in the MIT-rule is that, it will perform well if the parameter k is small. The allowable size depends on the magnitude of the reference signal. Consequently, it is not possible to give fixed limits which guarantee stability. The MIT-rule can thus give an unstable closed-loop system. Modified adjustment rules can be obtained using stability theory. All schemes discussed so far are called direct methods because the adjustment rules tell directly how the controller parameters should be updated. In this work, a new algorithm for STR is presented here. The proposed method called self-tuning regulator [3], is a different scheme where process parameters are updated and the controller parameters are obtained from the solution of a design problem. Here no need to find the adjustment mechanism. It is complicated procedure to determine the adjustment rules. STR opens the way to give a significant simplification of the control algorithm because the design calculations are eliminated and the regulator parameters are updated directly. Hence in STR, the result shows the best improvement in the transient response characteristics. The proposed method is computationally efficient and easy to implement. As a result, STR method has a wide number of applications when the plant exhibits non-linear behavior and when the plant is unknown. This gives the controller to maintain a desired point of performance in spite of any noise or fluctuation in the process. The study of STR is described below.

2. PROCESS MODELING AND TRANSFER FUNCTION

The process taken in to account is CSTR. The mathematical model of the reactor comes from in the reactor, which is in term cooled by a coolant that flows through a jacket around the reactor. The jacket is assumed to be perfectly mixed. Heat transfer takes place through the reactor wall into jacket. The main objective is to maintain the temperature of the reacting mixture at desired value. The manipulated variable is the coolant temperature. The CSTR [4] has three input signals: C_{Af} -Concentration of feed stream, T_f -Inlet feed stream temperature, T_j -Jacket coolant temperature. The two output signals: C_A -Concentration of A in reactor tank, T -Reactor Temperature. Assume the Reactor Parameter's value for the process.

The CSTR [4] has three input signals: C_{Af} -Concentration of feed stream, T_f -Inlet feed stream temperature, T_j -Jacket

coolant temperature. The two output signals: C_A - Concentration of A in reactor tank, T -Reactor Temperature. Assume the Reactor Parameter's value for the process.

Table 1. Reactor Parameter's value

Description	Reactor parameters	Nominal value
Flow rate and volume	$F/V, \text{hr}^{-1}$	4
Reaction rate constant	K_o, hr^{-1}	15×10^{12}
Heat of reaction	$(-\Delta H), \text{BTU/lbmol}$	40000
Activation energy	$E, \text{BTU/lbmol}$	33500
Liquid density and Specific heat	$\rho C_p, \text{BTU/ft}^3$	54.65
Inlet feed stream temperature	$T_f, ^\circ\text{C}$	70
Concentration of feed stream	$C_{Af}, \text{lbmol/ft}^3$	0.132
Enthalpy	UA/V	122.1
Jacket coolant temperature	$T_j, ^\circ\text{C}$	60

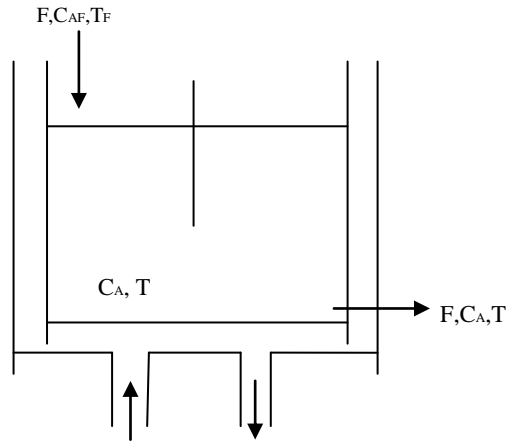


Fig 1: Continuous Stirred Tank Reactor with Cooling Jacket

2.1 Overall material balance & Linearization of Dynamic Equation

The CSTR is modeled using energy conservation principles,

Rate of accumulation = Rate of energy flow in – Rate of energy flow out.

The mass balance equations of CSTR are

$$F_2(C_A, T) = \frac{dT}{dt} = \frac{F}{V}(T_f - T) - \left(\frac{\Delta H}{\rho C_p} \right) K_o e^{\left(\frac{-E}{RT} \right)} C_A - \left(\frac{UA}{V \rho C_p} \right) (T - T_j) \quad (1)$$

The simplified transfer function of the CSTR temperature process can be found from the mass balance equations after modelling for the given reactor parameter values.

$$G_p(s) = \frac{0.13s + 4.79}{s^2 + 8.38s + 13.4} \quad (2)$$

The next step is to define the transfer function of model. The basic second order model of the form:

$$G_m(s) = \frac{\omega_n^2}{s^2 + 2\omega_n \xi s + \omega_n^2} \quad (3)$$

The desired specifications are defined in order to design the controller. The required specifications for the temperature control a maximum Overshoot (Mp) is 2% and a settling time (Ts) is less than 3 seconds are selected.

Hence the transfer function of the model is given:

$$G_m(s) = \frac{3.56}{s^2 + 2.67s + 3.56} \quad (4)$$

The discrete time transfer function of the CSTR temperature process ($T_j = 60^\circ\text{C}$) find from continuous time transfer function of the plant by using Mat lab command.

$$G_p(z) = \frac{0.89z^3 + 1.1z^2 + 0.15z}{z^3 - 1.2z^2 + 0.07z + 0.545} \quad (5)$$

The plant model is validated by means of an RLS algorithm described in the next section.

The transfer function of the reference model (desired) in discrete form

$$G_m(z) = \frac{0.1513z^3 + 0.1871z^2 + 0.0255z}{z^3 - 0.9z^2 + 0.15z + 0.114} \quad (6)$$

By using an MRAC method, the control law is designed for the above process transfer function and the reference model. Then the simulation studies are made in the later section.

3. MODEL REFERENCE ADAPTIVE CONTROLLER

Model reference adaptive controller (MRAC) is a controller used to force the actual process to behave like same model process. MRAC systems adapt the parameters of a normal control system to achieve this match between model and process. The Model Reference Adaptive Controller shows in Fig 2.

The reference model defines the desired performance characteristics of the process being controlled. The adaptation law uses the error between the process and the model output, the process output and input signal to vary the parameters of the control system. These parameters are varied so as to minimize the error between the process and the reference model.

The adaptation law attempts to find a set of parameters that minimize the error between the plant and the model outputs. To do this, the parameters of the controller are incrementally adjusted until the error has reduced to zero. A number of adaptation laws have been developed to date.

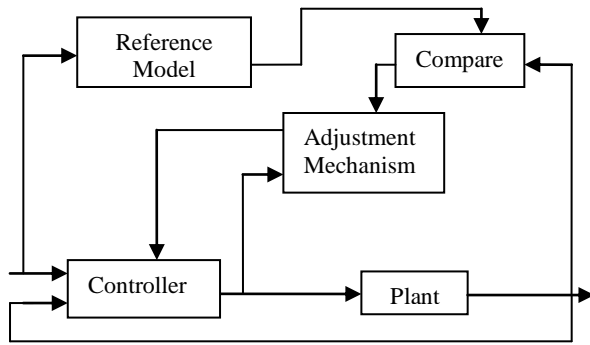


Fig 2: Block diagram of Model Reference Adaptive Controller

The two main types are the gradient and the Lyapunov approach and we have use gradient approach.

3.1 Adaptive Mechanism

It is used to adjust the parameters in the control law. Adaptation law [5] searches for the parameters such that the response of the plant which should be same as the reference model. It is designed to guarantee the stability of the control system as well as convergence of tracking error to zero. The MIT rule is the original approach to model reference adaptive control.

The MIT rule is

$$\frac{d\theta}{dt} = -\gamma e \frac{\partial e}{\partial \theta} \quad (7)$$

3.2 Adaptive MIT (AMIT) Algorithm

Based on a priori knowledge, the process is modelled as second order. The transfer function of the interactive thermal process can be represented by equation

$$\frac{Y}{U} = \frac{K}{s^2 + a_1 s + a_2} \quad (8)$$

Where K , a_1 and a_2 are positive and are the process parameters.

The AMIT control law is given by

$$U = \theta_1 U_c - \theta_2 Y \quad (9)$$

The closed-loop transfer function related to the output and input with the AMIT controller in the loop is given by equation

$$\frac{Y}{U_c} = \frac{K\theta_1}{s^2 + a_1 s + (a_2 + K\theta_2)} \quad (10)$$

Where U_c is the command signal (reference input). The controller parameters are updated by the adaptation mechanism such that the process output follows the model output equation.

$$\frac{Y_M}{U_c} = \frac{K_M}{s^2 + A_1 s + A_2} \quad (11)$$

where K_M , A_1 and A_2 are the reference model parameters. To apply the AMIT controller, the sensitivity derivatives are obtained by calculating the partial derivatives of

modelling error with respect to the Controller Parameters. The Process Parameters K , a_1 & a_2 are not known.

An approximation based on the observation:

$s^2 + a_1 s + (a_2 + K\theta_2) = s^2 + A_1 s + A_2$ is applied for perfect model following. Then,

$$\frac{\partial e}{\partial \theta_1} = \frac{K}{s^2 + A_1 s + A_2} U_c \quad (12)$$

$$\frac{\partial e}{\partial \theta_2} = -Y \frac{K}{s^2 + A_1 s + A_2} \quad (13)$$

The controller parameters θ_1 and θ_2 are

$$\theta_1 = -\frac{\gamma}{s} e \frac{K}{s^2 + A_1 s + A_2} U_c \quad (14)$$

$$\theta_2 = \frac{\gamma}{s} e \frac{K}{s^2 + A_1 s + A_2} Y \quad (15)$$

Where, $\gamma' = \gamma K$

Hence, it is a good way to compensate for known nonlinearities. It reacts quickly to changing conditions. The simulation studies are made in the later section. The proposed method called self tuning regulator is discussed in the next section.

4. SELF-TUNING REGULATOR

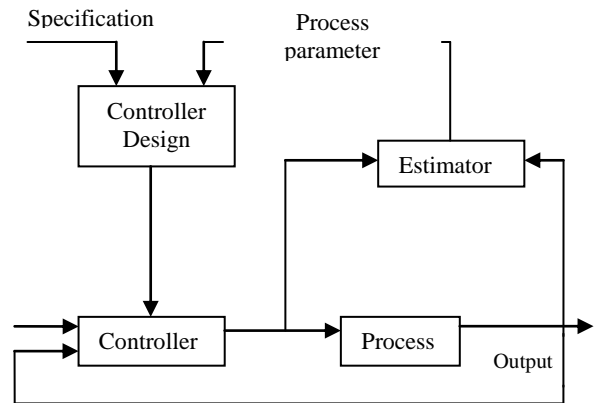


Fig 3: Block diagram of Self Tuning Regulator

The Self-Tuning Regulator [6] (STR) attempts to automate several of these tasks. The block diagram of a self-tuning regulator is shown in Fig.3. The STR is composed of two loops. The inner loop contains the process and an ordinary feedback controller.

The outer loop consists of a recursive parameter estimator and design calculations. The "Estimator" in the block represents an estimation of the process parameters using least-squares methods. The "Controller Design" in the block represents the design of the controller for the system. The "Controller" is to calculate the control law with the controller parameters. The system can be viewed as an automation of processing the modelling, estimation and design, in which the process model and the control design

are updated at each sampling interval. Sometimes the STR algorithm can be simplified by reparameterizing and directly estimating the controller parameters, not the process parameters alone. It is flexible that the STR scheme can be implemented by different choices of the underlying design and estimation methods. Simulation studies for the estimator and the control algorithms are mainly undertaken by using MATLAB S-function.

4.1 S- function Simulation

An S-function (System-functions) is a computer language description of a dynamic system. S-functions can be written using MATLAB or C. It is specifically structured to simulate the results. S-functions use a special calling syntax that enables you to interact with Simulink equation solvers. This interaction is very similar to the interaction that takes place between the solvers and built-in Simulink blocks. The form of an S-function is very general and can accommodate continuous, discrete, and hybrid systems. As a result, nearly all Simulink models can be described as S-functions. It is used for estimation and controller design.

4.2 Scope of work

- 1) Introduce RLS algorithm and pole placement algorithm.
- 2) Use of S-function program for estimation and controller design.
- 3) Analyse the stability of adaptive control system and give the comparison between adaptive controller and conventional controller.

The process parameters are obtained by using RLS algorithm if the process is unknown. Then the controller parameters are obtained by using pole placement technique. In STR, the design calculations are eliminated and the regulator parameters are updated directly. But in MRAC, finding the adjustment rules and mechanism is very complicated. Due to the above reason, the STR method is preferred.

5. RECURSIVE LEAST-SQUARE ALGORITHM

The least-square method is used to estimate an unknown process. But the process is known already by means of a modelling studied above. It is validated by using RLS algorithm. The algorithm is demonstrated in this section. The principle of RLS[7] is that the unknown parameters of a model is chosen by minimizing the sum of the square of the difference between the actually observed and the analytically predicted output values with possible weighting that measure the degree of precision. The process is described by the single-input, single output (SISO) system.

$$A(z^{-1})y(t) = B(z^{-1})(u(t-d_0) + v(t-d_0)) \quad (16)$$

In vector form, $y(k) = \phi^T(k)\theta(k)$

Here, y is the output, u is the input of the system, and v is a disturbance. The disturbance can enter the system in many ways. Here, it enters at the process input. Recursive estimation algorithm is desirable. It saves the computation time by using the results obtained at time $(t-1)$ to get the estimates at time t .

The recursive least-square estimator is given by

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k) - \phi^T(k)\hat{\theta}(k-1)] \quad (17)$$

$$K(k) = P(k-1)\phi(k)(I + \phi^T(k)P(k-1)\phi(k))^{-1} \quad (18)$$

$$P(k) = P(k-1) - P(k-1)\phi(k)[I + \phi^T(k)P(k-1)\phi(k)]^{-1}\phi^T(k)P(k-1) \quad (18.a)$$

The RLS algorithm above is interpreted intuitively. The estimate $\hat{\theta}(k)$ is obtained by adding a weighted rediction error term $y(k) - \phi^T(k)\hat{\theta}(k-1)$ to the previous estimate $\hat{\theta}(k-1)$. The elements of the vector $K(k)$ are weighting factors that tell how the correction and the previous estimates should be combined. Notice that $P(k)$ can be made arbitrarily close and by choosing P_0 sufficiently large. Here, the system to be controlled is identified by means of RLS algorithm. Hence unknown parameters are estimated using RLS algorithm. The estimated parameters are $\beta_0, \beta_1, \beta_2, \alpha_1, \alpha_2, \alpha_3$. Here, six parameters are estimated. The RLS estimator is simulated by using S- function under simulink. The S- function block is added and it is defined by an M-file S-function code into a simulink model, and excites the plant to be estimated. In the final stage, the RLS estimation algorithm is simulated and the plant model is obtained as shown in equation (6) in discrete form. The estimation process is completed and the parameters are identified. The next step is to design the control algorithm by using a MDPP method.

6. POLE PLACEMENT DESIGN

As shown in Fig.4, a general linear controller can be described by

$$R(z^{-1})u(t) = T(z^{-1})u_c(t) - S(z^{-1})y(t) \quad (19)$$

where $R(z^{-1})$, $S(z^{-1})$ and $T(z^{-1})$ are polynomials in the back shift operator z^{-1} . This controller [8] consists of a feed forward with the transfer operator and a feedback with the transfer operator. It thus has two degrees of freedom.

The block diagram of the closed-loop system is illustrated in the following Fig.4.

The closed-loop characteristic polynomial (also called a Diophantine equation) is thus

$$A_c = AR + BS \quad (20)$$

The key idea of this design method is to specify the desired closed-loop characteristic polynomial A_c , which is a pole placement problem. R and S can then be solved. The solution that gives a controller of lowest degree is the minimum-degree solution.

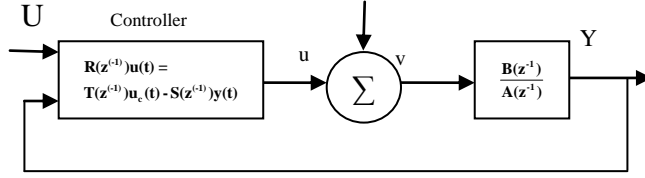


Fig 4: Closed Loop System with a General Linear Controller

polynomials R and S. Other conditions must be introduced to calculate the polynomial T in the controller.

The Diophantine equation then reduces to

$$AR' + B'S = A_m A_0 = A_c' \quad (21)$$

where A_0 is called the observer polynomial. Introducing the factorizations of B, B_m and A_c gives

$$T = B_m' A_0 \quad (22)$$

For a controller to be causal it must satisfy $\deg S \leq \deg R$ and $\deg T \leq \deg R$. Since $\deg A > \deg B$, there is always a minimum degree solution of (13) such that $\deg S < \deg A$. Thus the causality conditions can be derived from (21) and (22) as [10]:

$$\deg A_c \geq 2 \deg A - 1 \quad (23)$$

$$\deg A_m - \deg B_m \geq \deg A - \deg B = d_0 \quad (24)$$

In the discrete-time case, it is reasonable to choose R, S and T to have the same degree, such that there is no extra delay in the controller. From this the control algorithm can be derived.

6.1 Pole Placement Algorithm

The simulation steps of STR [9] are as follows:

Step1: Estimate the coefficients of the polynomials A and B using RLS method.

Step2: Use the polynomials A and B estimated in step 1, apply pole placement method. The polynomials R, S and T of the controller are obtained by solving Diophantine equation and compute the control action.

6.2 Identification of process parameters

From (6)

$$G_p(z^{-1}) = \frac{(0.89 + 1.1z^{-1} + 0.15z^{-2})}{(1 - 1.2z^{-1} + 0.07z^{-2} + 0.545z^{-3})} \quad (25)$$

Where $d=0, m=2, n=3$

Total number of unknown parameters [10] to be estimated = $n+m+1 = 2+3+1 = 6$

6.3 Identification of controller parameters

The reference model is specified as

From (6) $a_{m1} = 0.9, a_{m2} = 0.15, a_{m3} = 0.114$

Step 1: Estimate the co-efficient of polynomials A and B

From (6) $b_0 = 0.89, b_1 = 1.1, b_2 = 0.15$

From the above information β can be calculated as

$$\beta = \frac{1 + a_{m1} + a_{m2} + a_{m3}}{b_0 + b_1 + b_2} = 0.170$$

$$B_m = \beta B = 0.170(0.89z^3 + 1.1z^2 + 0.15z)$$

Step 2: The polynomials R, S and T of the controller are obtained by solving Diophantine equation.

$$A_c = AR + BS = A_0 A_m \quad (26)$$

$$z^3 - 1.2z^2 + 0.07z + 0.545)R + (0.89z^3 + 1.1z^2 + 0.15z)S$$

$$= (z + 0.8)(z^3 - 0.9z^2 + 0.15z + 0.114) \quad (27)$$

$$R(z) = z^2 + r_1z + r_2 = z^2 - 0.29z + 0.21 \quad (28)$$

$$S(z) = S_0z^2 + S_1z + S_2 = -0.1z^2 - 0.4z + 0.12 \quad (29)$$

$$T = \beta A_0 \quad (30)$$

Step 3: Compute the control law from the equation

$$Ru(t) = Tu_c(t) - Sy(t) \quad (31)$$

$$u(t) = 0.29u(t-1) - 0.21u(t-2) + 0.170u_c(t-1) + 0.136u_c(t) + 0.1y(t) + 0.4y(t-1) - 0.12y(t-2) \quad (32)$$

7. SIMULATION OF PID AND ADAPTIVE CONTROL SYSTEMS

7.1 Conventional PID Controller

We have defined the plant we need to develop a standard controller [11] to compare with the adaptive controller [12]. Controller setting is done using Ziegler-Nicholas technique and the best controller parameters are found to be $K_c=10, t_i=1$ and $t_d=1$. This model has the step input signal, PID controller and transfer function of process plant.

7.2 Self Tuning Regulator

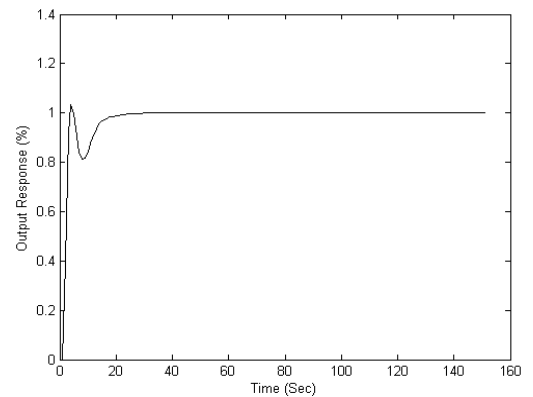


Fig 5: Output Responses of the Control Signal

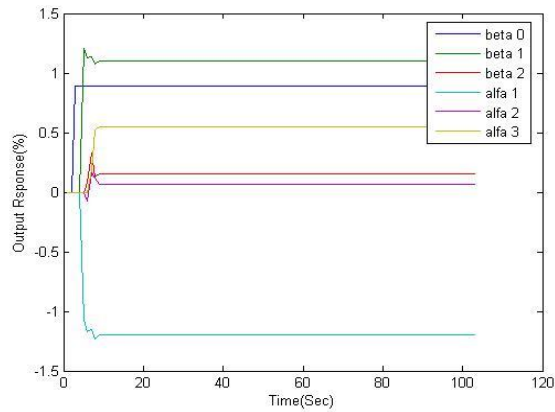


Fig 6: Output Responses of the Estimator

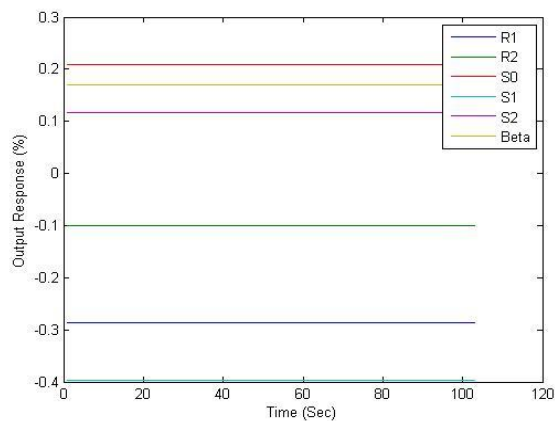


Fig 7: Output Responses of the Controller Parameters

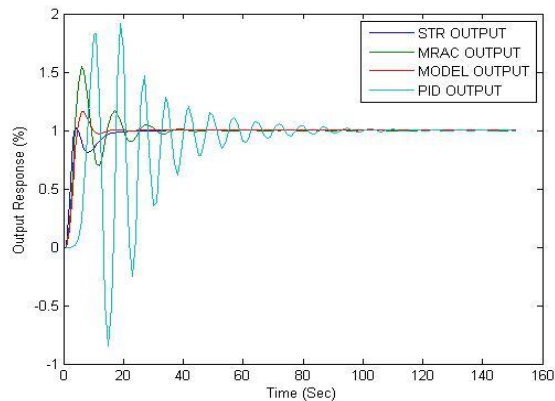


Fig 8: Output Responses of the STR, MRAC and PID Controller

8. RESULTS AND DISCUSSION

Fig.8 shows the plant output responses of the three controllers. The conventional PID becomes unstable and it shows oscillatory response. MRAC controller has less oscillation in the initial point and less settling time when compared to conventional PID controller. But the STR has no oscillations and minimum settling time when compared to both controllers. Thus the STR is better for this process, when compared to the MRAC and Conventional PID controller. Hence, in STR, the controller is automatically

tuned and optimizes the closed loop system in an efficient way. Hence the control loop performance is improved in an adaptive way [13].

Table 2. Comparison of Time Domain Parameters for the PID, MRAC and STR

Time Domain Parameters	PID	MRAC	STR
Dead Time (t_d) (sec)	7	2.7	2.2
Rise Time (t_r) (sec)	8.1	4	3.5
Peak Time (t_p) (sec)	10	6	4
Settling Time (t_s) (sec)	145	78	35
Maximum Overshoot (M_p) (%)	91	55	3.5

9. CONCLUSION AND FUTURE SCOPE

As compared to conventional controllers, Adaptive Control method is very effective to handle the situations where the parameter variations and environmental changes occur. They are demonstrated clearly in the results discussed above. In PID, it is to find the controller gain values. This is normally done based on the tuning of controllers. It is very difficult to guarantee the stability since the gain values get changed due to the process dynamics. In MRAC, the MIT rule is developed and it will perform well if the parameter k is small. The allowable size depends on the magnitude of the reference signal. Consequently, it is not possible to give fixed limits which guarantee stability. The MIT-rule can thus give an unstable closed-loop system. In STR [14], controller is designed such that the controllers parameters get varied automatically even for the variations in the characteristics of the process. STR gives a significant simplification of the control algorithm because the design calculations are eliminated and the regulator parameters are updated directly. Hence in STR, the result shows the best improvement in the transient response characteristics. In this process, assume the disturbance is the coolant water temperature. The controller parameters (5 parameters) like R_1 , R_2 , S_0 , S_1 , S_2 , varied automatically even the coolant water temperature changes to a new value. The number of controller parameters depends on the order of the process. That is, the controller gets adapted even when the temperature of the coolant water changes. As a result, the STR method has given vast improvements in the closed loop performance of the CSTR process by maintaining a constant temperature of the reacting mixture. The reason for using STR is, the controller parameters are updated directly without having an adaptive mechanism [15]. Hence the proposed method is computationally efficient and easy to implement when compared to the other two methods. This method has the capability of high speed of response [16] and achieving the desired output very quickly. This type of time domain specifications are identified shown in Table.2 in order to show the good performance of the closed loop of CSTR process. This type of technique is specifically applied to CSTR, Distillation Column, Boilers [17] and all other unit operations. This can be further extended to check the analysis of robustness of the system by introduce a noise contamination to the process.

10. REFERENCES

- [1] Missula Jagath Vallabhai, Pankaj Swarnkar, D.M. Deshpande, "Comparative Analysis of PI Control and Model Reference Adaptive Control Based Vector Control Strategy for Induction Motor Drive", *International Journal of Engineering Research and Applications (IJERA)*, Vol. 2, Issue 3, pp.2059-2070, May-Jun 2012.
- [2] Rajesh Singla and Rahul Upadhyay, "Analysis of CSTR Temperature Control with Adaptive and PID Controller(A Comparative study)", *IACSIT International Journal of Engineering and Technology*, vol. 2, No. 5, pp. 453-458, October 2010.
- [3] Middleton, R.H. and Goodwin, G.C., "Design issues in Adaptive control", *IEEE Transactions on Automatic Control*, vol.33, issue.1, pp. 50-58, January 1988.
- [4] George Stephanopoulos, "Chemical Process Control", Prentice-Hall of India Private Limited, India, 2008.
- [5] Middleton R.H., Goodwin, G.C., "Design issues in adaptive control", *IEEE transaction on automatic control*, vol. 33, no. 1, pp. 50-58, 1988.
- [6] Astrom.K.J and B.Wittenmark, "On Self-tuning Regulators", *Automatica*, vol .9, no. 2, pp. 185-199, 1973.
- [7] Honghong Duan, Jie Jia, Ruifeng Ding, "Two-stage recursive least squares parameter estimation algorithm for output error models", *International Journal of Mathematical and Computer Modeling*, vol. 55, no. 3, pp. 1151-1159, 2012.
- [8] D. Vasanthi, B. Pranavamoorthy, N. Pappa, "Design of a self-tuning regulator for temperature control of a Polymerization reactor", *ISA Transactions*, vol. 51, Issue 1, pp.22-29, January 2012.
- [9] Glan Devadhas.G, Pushpakumar.S, Muruga Prasad, S.V, "Intelligent Computation of Controller using Optimization Technique for a Non-linear Chemical Process", *International Journal of Research and Reviews in Soft and Intelligent Computing (IJRRSIC)*, vol. 1, no.3 pp. 49-55, September 2011.
- [10] Jiabo Zhanga, Feng Ding and Yang Shi, "Self Tuning control based on multi-innovation stochastic gradient parameter estimation", *International Journal of Systems and Control Letters*, vol. 58, no.1, pp. 69-75, 2009.
- [11] Luyben, W. L, "Process Modeling, Simulation and Control for Chemical Engineers", McGraw-Hill, New York, 1989.
- [12] Karl J.Astrom and Bjorn Wittenmark, "Adaptive Control", Second Edition, Pearson Education Private Limited, Singapore, pp. 185-245, 1995.
- [13] M. Krstic, I. Kanellakopoulos, "Nonlinear and Control Design", New York: Wiley and Sons, 1995.
- [14] Wellstead.P.E and D.L.Prager, "Multivariable Pole-Assignment Self-tuning Regulator", *Proceedings of the Institution of Electrical Engineers*, vol. 126, issue.8, pp. 781-787, August 1979.
- [15] Astrom, K.J. and P. Eykhoff, "System Identification-A Survey", *Automatica*, vol.50, no. 7, pp. 123, 1971.
- [16] Anderson, B.D.O. and R.M. Johnstone, "Adaptive Systems and Time Varying Plants", *International Journal of Control*, vol. 37, no. 2, pp. 367-377, 1983.
- [17] Chalam, V.V., "Adaptive Control Systems: Techniques and Applications", Marcel Dekker, New York, 1987.

AUTHOR'S PROFILE

K.Prabhu was born in Tamilnadu INDIA in 1980. He obtained his Engineering degree in Electrical & Electronics in 2002 from the Bharathiyar University, Coimbatore and his M.E. degree in Applied Electronics in 2006 from Anna University, Chennai. Currently, he is working as an Assistant Professor. His research interest includes control system, adaptive & process control systems. He published many papers in journals & conferences.

Dr. V. Murali Bhaskaran is a Principal of Paavai College of Engineering, NH-7, Pachal, Namakkal, Taminadu, India. He obtained his B.E. Degree in Computer Science and Engineering from Bharathidasan University in the year 1989, M.E. degree in Computer Science and Engineering from Bharathiar University in the year 2000 and secured University III rank and Ph.D in Computer Science and Engineering from Bharathiar University in the year 2008. He has published 4 papers in Journals and 12 papers in Conferences both at National and International level. He is a life member of ISTE, CSI and ACS. Also he is a fellow member in IE (FIE). His areas of interest include Computer Architecture, Computer Networks, Network Security, Information Security, etc.,