Design of Self-Tuning Regulator for Non-Linear Unstable System

Reshmi R, Gnanasoundharam J, K Kotteeswaran R, reshmiramanath@gmail.com

Abstract — A considerable amount of controller design problems are addressed by the researchers while designing controllers for non-linear system. Classical control technique such as feedback and feedforward are used to tune the system to obtain a satisfactory response. This work proposes an adaptive control scheme based on Self-Tuning Regulator (STR), to combat with the uncertainties and difficulties in the control behavior of non-linear systems. This work implements the integration of Recursive Least Square (RLS) and Minimum Degree Pole Placement (MDPP) algorithm. The Recursive Least Square method is used to estimate the plant parameters and the Minimum Degree Pole Placement is used to calculate control parameters and the control law is obtained. The proposed control algorithm is applied to non-linear unstable system. The experimental work is implemented using MATLAB/Simulink and the obtained results are analyzed in detail. The obtained simulation results illustrate how well the MDPP and RLS algorithm works.

Index Terms— Self Tuning Regulator, Recursive Least Square, Minimum Degree Pole Placement

I. INTRODUCTION

PROCESS Control is a combination of control engineering and chemical engineering that is used to monitor, regulate or control the dynamics of a plant or a process, which cannot be achieved by manually. This is widely implemented in food process, oil refining, hydrometallurgical and power generating industries. Industrial process often involves automated control to reduce the variation of certain characteristics which affect the safety. Process operating under a control regime operate in some form of closed loop control. Most industrial processes are controlled by classical PID regulators due to its simple structure, optimal performance and robustness over wide range of operating conditions.

In 1950 research were carried out to design autopilots for high performance aircraft. Ordinary feedback control does not operate well for the entire control regime, which lead to development of adaptive control. In general, “to adapt” means to change a behavior to conform to new circumstances. Thus, an adaptive controller is a controller that can modify its behavior in response to varying dynamics and disturbance of the process. Ordinary feedback also attempts to reduce the effect of disturbance and plant uncertainty. But adaptive system is any physical system that has been designed with an adaptive viewpoint. State Space and stability control theory brought major development in control theory that contributed to development of adaptive control. An Adaptive control system consists of two closed loops, one is normal feedback with the plant and other loop is parameter adjustment loop. Usually there are four types of adaptive control schemes: Self-tuning regulator, model reference adaptive control, gain scheduling and dual control.

Self-Tuning Regulator is a method where the process parameters are updated and control law is attained. This method provides a simplified control algorithm as it eliminates the design calculations and process parameters are updated directly. Thus, the Self-Tuning Regulator (STR) offers a better and improved transient response. The main advantage with this method is that it easy to implements and is computationally effective. The Self-Tuning Regulator is described in detail in the section below.

II. SELF-TUNING REGULATOR

Self-tuning system is capable of optimizing its own internal running parameters in order to maximize or minimize the objective function (Astrom et al.). Self-tuning system exhibits non-linear, adaptive mechanism. Self-tuning system has been a trademark in the aerospace industry, as this kind of feedback is necessary to produce an optimal multivariable control of linear process. The block diagram of self-tuning regulator is shown in figure 1. The self-tuning regulator consists of two loops. The inner loop comprises of the process and an ordinary feedback controller. The outer loop comprises the Recursive parameter estimator.

![Block diagram of Self-Tuning Regulator](image)

The block named “Estimator” denotes an on-line estimation of process parameters using recursive least square algorithms. The block named “Controller Design” represents an on-line solution to design problem for systems with known parameter or with estimated parameter. The block named “Controller” is used to calculate the control action with the controller parameters are computed by its proceeding block. The system can be viewed as an automation of processing, modelling,
estimating and designing, in which the process model and control design are updated at each sampling interval.

A. Estimation Algorithm

It is important to estimate the process parameter on-line in adaptive control. For an adaptive control system, the adaptive mechanism is based on identifying the system first. A self-tuning regulator explicitly includes a recursive parameter estimator. The process parameters estimation is a part of system identification. System identification can also be said as selection of model, structure, experiment design, parameter estimation and validation.

Recursive Least Estimation Algorithm

The least-square method is commonly used in system identification. The principle is that the unknown parameters of a mathematical model should be 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 measures the degree of precision (Plakket et al.). In adaptive control system the observations are obtained sequentially in real time. Recursive estimation algorithm is desirable. It saves the computation time by using the results obtained at time \( t-1 \) to get the estimate at time \( t \). Hence, the recursive least square (RLS) estimation method is used in this section.

It is assumed that the process is described by single-input, single output (SISO) system.

\[
A(z^{-1})y(t) = B(z^{-1})(u(t - d_x) + v(t - d_y)) \tag{2.1}
\]

Where

\[
A(z^{-1}) = 1 + a_1 z^{-1} + \ldots + a_n z^{-n}
\]

\[
B(z^{-1}) = b_1 + b_2 z^{-1} + \ldots + b_n z^{-n}
\]

With \( m = n - d_x \) \( y \) 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 assumed that \( v \) enters at process input. The recursive least-square estimator is given by

\[
\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k) - \hat{\theta}(k)^T \hat{\theta}(k-1)] \tag{2.2}
\]

\[
K(k) = P(k-1) \hat{\theta}(k-1) [I + \hat{\theta}(k)^T P(k-1) \hat{\theta}(k)]^{-1} \tag{2.3}
\]

\[
P(k) = P(k-1) - P(k-1) \hat{\theta}(k-1) [I + \hat{\theta}(k)^T P(k-1) \hat{\theta}(k)]^{-1} \hat{\theta}(k)^T P(k-1) \hat{\theta}(k-1) \tag{2.4}
\]

The RLS algorithm above can be interpreted instinctively. The estimate \( \hat{\theta}(k) \) is obtained by adding weighted prediction error term \( \hat{\theta}(k)^T \hat{\theta}(k-1) \) to the previous estimate \( \hat{\theta}(k-1) \). The term \( \hat{\theta}(k)^T \hat{\theta}(k-1) \) can be viewed as the value of \( y \) at time \( k \) predicted by the model with the previous estimates \( \hat{\theta}(k-1) \).

The unknown parameters \( \beta_0, \beta_1, \beta_2, \ldots, \beta_n \) and \( a_0 \) are estimated by using recursive least square algorithm. 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 recursive least square estimation algorithm is simulated and the plant model is obtained in the discrete form. The estimation process completed and parameters are identified.

B. Control Algorithm

A general linear controller can be described as

\[
R(z^{-1})u(t) = T(z^{-1})u(t) - S(z^{-1})y(t) \tag{2.5}
\]

Where \( R(z^{-1}), S(z^{-1}) \) and \( T(z^{-1}) \) are polynomials in the back-shift operator \( z^{-1} \). This controller consists of a feedback with transfer operator \( \frac{R(z^{-1})}{S(z^{-1})} \) and a feedback with the transfer operator \( \frac{R(z^{-1})}{T(z^{-1})} \). It thus has two degrees of freedom. The block diagram of the closed loop system is shown in Fig 2.

![Fig. 2. Block diagram of a general Linear Controller](image)

The closed loop characteristics polynomial is represented as

\[
A_C = A_R + B_S \tag{2.6}
\]

The key idea of the controller design is to specify the desired closed loop characteristic polynomial \( A_C \) as a design parameter. By solving the Diophantine equation, the polynomials \( R \) and \( S \) are obtained. The closed loop characteristics polynomial \( A_C \) determines the property and the performance of the closed system. The Diophantine equation always has solution if the polynomial \( A \) and \( B \) are co-prime.

The Diophantine equation can be simplified as

\[
A_R + B_S = A \tag{2.7}
\]

To have a control law that is causal in the discrete-time case, we must impose the following conditions upon the polynomials in the control law

\[
deg S \leq deg R \tag{2.8}
\]

\[
deg T \leq deg R \tag{2.9}
\]

The condition \( deg S \leq deg R \) implies that

\[
deg A_R \geq 2 deg A - 1 \tag{2.10}
\]

The condition \( deg T \leq deg R \) implies that

\[
deg A_m - deg B \geq deg A - deg B = d_0 \tag{2.11}
\]

III. RESULTS AND SIMULATION

A. Simulation of RLS Estimator

To estimator the process parameters on-line, the recursive least square estimator is used. Adaptive mechanisms are based on identifying the system first and the process parameter estimation is portion of system identification. The RLS estimator presented is simulated by using S-Function under Simulink in this work.

The plant to be estimated is in the general form as below:

\[
G(z^{-1}) = z^{-d} \frac{N(z^{-1})}{D(z^{-1})} \tag{3.1}
\]

where \( d \) is the time delay, \( n \geq m + d \)

\[
N(z^{-1}) = \beta_0 + \beta_1 z^{-1} + \ldots + \beta_m z^{-m}, \deg [N(z^{-1})] = m
\]
The estimation block diagram is shown in the following figure.

Assuming that \( u(k) \) and \( y(k) \) are the input and the output of the plant, respectively, we can write the plant as much as below

\[
\begin{align*}
D(z^{-1}) &= a_0 + a_1 z^{-1} + \cdots + a_n z^{-n}, \deg[D(z^{-1})] = n
\end{align*}
\]

The estimation block diagram is shown in the following figure.

Assuming that \( u(k) \) and \( y(k) \) are the input and the output of the plant, respectively, we can write the plant as much as below

\[
y(k) = \beta_0 u(k-d) + \cdots + \beta_m u(k-d-m) - a_1 y(k-1) - \cdots - a_n y(k-n)
\]

Or in the form of vector

\[
y(k) = \varphi^T(k) \theta(k)
\]

### B. Simulation of MDPP Controller

The MDPP control law is simulated using S-Function. The degree of the polynomial of the process model and reference model is given and system will be estimated automatically. The steps for the simulation of Minimum Degree Pole Placement (MDPP) controller are mentioned below.

**Data:** \( B_m / A_m \) is closed-loop pulse transfer operator which gives the desired reference model and \( A_n \) is the desired polynomial.

**Step 1:** Using the RLS technique the coefficients of polynomials \( A \) and \( B \) in equation are evaluated.

**Step 2:** the MDPP method is applied to the polynomials \( A \) and \( B \) estimated in step 1. solving the Diophantine equation find the polynomials \( R, S \) and \( T \) of the controller.

**Step 3:** Finally, the control action is computed from equation

\[
Ru(t) = Tu(t) - Sy(t)
\]

Step 3 is repeated at each sampling period.

First, we have considered a third order non-linear system whose transfer function is

\[
G_p(z) = \frac{0.9 z^2 + 1.3 z + 0.23}{z^3 - 1.2 z^2 + 0.05 z + 0.495}
\]

\( G_p(z) \) could be rewritten as

\[
G_p(z) = G_p(z^{-1}) = \frac{0.9 z^{-1} + 1.3 z^{-2} + 0.23 z^{-3}}{1 - 1.2 z^{-1} + 0.05 z^{-2} + 0.495 z^{-3}}
\]

For the above-mentioned transfer function \( d=1, m=2 \) and \( n=3 \). Therefore, the total number of unknown parameters need to be estimated is 6. The parameters estimated using projection algorithm is mentioned below.

The reference model is specified as \( B_m / A_n \), where

\[
A_n = z^3 - 0.85 z^2 + 0.24 z + 0.01
\]

The estimation block diagram is shown in the following figure.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>( \beta_0 )</th>
<th>( \beta_1 )</th>
<th>( \beta_2 )</th>
<th>( a_1 )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Value</td>
<td>0.9</td>
<td>1.3</td>
<td>0.23</td>
<td>-1.20</td>
<td>0.05</td>
<td>0.496</td>
</tr>
<tr>
<td>Time=10s</td>
<td>0.9</td>
<td>1.2</td>
<td>0.21</td>
<td>-1.21</td>
<td>0.05</td>
<td>0.445</td>
</tr>
<tr>
<td>Time=50s</td>
<td>0.9</td>
<td>1.3</td>
<td>0.23</td>
<td>-1.21</td>
<td>0.05</td>
<td>0.496</td>
</tr>
</tbody>
</table>

Solving the Diophantine equation, we get the \( R, S, \) and \( T \) polynomials

\[
R(z) = z^2 - 0.2708 z - 0.3962
\]

\[
S(z) = 2.932 z^2 - 0.5232 z - 0.1482
\]

\[
T(z) = 0.1846 z^2 - 0.0738 z - 0.0590
\]

Finally, we acquire the control law as

\[
u(t) = 0.1846 u_c(t) - 0.0738 (t-1) - 0.059 (t-2) + 2.932 y(t) + 0.5232 y(t-1) - 0.1482 y(t-2)
\]

\[-0.2708 u(t-1) - 0.3962 u(t-2)
\]
Here we consider a third order non-linear unstable system (Anggi et al.) whose discretized transfer function is

\[ G_p(z) = \frac{0.339z^2 + 0.011z + 0.002}{z^2 - 2.35z + 1.764z - 0.427} \]  

(3.12)

\[ G_S(z) = G_p(z^{-1}) = \frac{0.339z^{-1} + 0.011z^{-2} + 0.002z^{-3}}{1 - 2.35z^{-1} + 1.764z^{-2} - 0.427z^{-3}} \]  

(3.13)

For the above-mentioned transfer function \(d=1, m=2\) and \(n=3\). Therefore, the total number of unknown parameters need to be estimated is 6. The parameters estimated using projection algorithm is mentioned below.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>(\beta_0)</th>
<th>(\beta_1)</th>
<th>(\beta_2)</th>
<th>(\alpha_1)</th>
<th>(\alpha_2)</th>
<th>(\alpha_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Value</td>
<td>0.339</td>
<td>0.011</td>
<td>0.002</td>
<td>-2.35</td>
<td>1.764</td>
<td>0.427</td>
</tr>
<tr>
<td>Time=50s</td>
<td>0.339</td>
<td>0.011</td>
<td>0.002</td>
<td>-2.34</td>
<td>1.765</td>
<td>0.426</td>
</tr>
<tr>
<td>Time=100s</td>
<td>0.339</td>
<td>0.011</td>
<td>0.002</td>
<td>-2.33</td>
<td>1.766</td>
<td>0.425</td>
</tr>
</tbody>
</table>

Finally, we acquire the control law as

\[ u(t) = 0.125u(t-1) - 0.051(t-1) - 0.048(t-2) + 1.452y(t-1) - 2.303y(t-1) - 0.9082y(t-2) - 0.5403u(t-1) - 0.3081u(t-2) \]  

(3.18)

IV. CONCLUSION

When compared to conventional controller, the adaptive controllers can effectively handle the situations such as parameter variation and environmental changes. In self-tuning regulator (STR), controller is designed such that controller parameters are automatically varied even for small variation. The Self-Tuning Regulator provides a way for simplifying of control algorithm because design calculations are eliminated and the parameters of the regulator are directly updated. The number of controller parameters to be estimated depends on the order of the process. Thus, it adapts even when the process parameters changes. The results obtained shows that the Self-Tuning mechanism provides an improved transient response and there is no oscillation in the obtained output. The settling time is also minimum when compared to other controllers. As a result, the Self-Tuning Regulator technique...
has an improved closed loop performance when compared to conventional controllers. Hence Self-Tuning Regulator technique is computationally effective and can be implemented easily. Self-Tuning Regulator can be applied to various nonlinear systems such as Distillation column, Level Control, CSTR, Boiler and other systems as well.

REFERENCES
