Design of an Intelligent Controller to Stabilize a Non-Minimum Phase Plant Using PI Control Scheme

W. Saman Marasinghe∗ and Dr. Sudath R. Munasinghe†
∗Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka.
Email: saman@millenniumit.com
†Department of Electronic and Telecommunication Engineering, University of Moratuwa, Moratuwa, Sri Lanka.
Email: rohan@ent.mrt.ac.lk

Abstract—Designing an intelligent controller to stabilize a non-minimum phase plants is proposed. This intelligent controller is designed to learn the control action principles from the data obtained using other methods of automatic or manual control, e.g., a proportional-integral (PI) controller. It is well known that non-minimum phase systems present difficulty for applying control strategies because they have an initial inverse response to step input in the opposite direction in the steady state. This project we consider the non-identical reverse dynamics as well as the mechanical oscillation effect in non-minimum phase plant modeling. Also, we have dealt with random disturbances and random changes, in that the randomness appears in the instance of occurrence, magnitude, and sign of these events. It makes the simulation in this paper more realistic, and that the results more trustworthy. Since, it is extremely difficult to provide a hard stability proof for intelligent controllers, the training data for this intelligent controller is taken from a prolonged simulation using few PI controllers that are specific to different operating power levels to demonstrate the stability.

 NOMENCLATURE

| Power level [%] | p |
| Feed water flow [kg/s] | u |
| Steam flow [kg/s] | v |
| Water level [mm] | y |
| Reference water level [mm] | r |
| Level error [mm] | e_l = r - y |
| Flow error [kg/s] | e_f = v - u |
| Rated steam flow at power p [kg/s] | v_p |

I. INTRODUCTION

A discrete system is said to be a non-minimum phase process if at least one of the zeros of the transfer function is located outside the unit circle. These kinds of processes are common in industrial applications and they are characterized by their inverse response. It is well known that non-minimum phase systems present difficulty for applying control strategies because they have an initial inverse response to step input in the opposite direction in the steady state. The presence of unstable zero in a process transfer function is thus identified as being responsible for its difficult dynamic behavior; it is also the source of a considerable amount of difficulty in controller design. Another aspect of controlling a process with unstable zero is the instability problem, which arises in order to achieve high performance when the controller contains an inverse of the process model. U Tube Steam Generator (UTSG) of nuclear power plant is also a non-minimum phase plant, because its behavior contains some non-minimum phase dynamics. Design an Intelligent Controller to Stabilizing a water level of the UTSG while using the non identical reverse dynamics and considering the random changes and disturbances is the primary objective of this project.

Generally two types of nuclear power plants are used. Those are boiling-water reactors (BWRs), and pressurized-water reactors (PWRs). In the BWR, the water heated by the reactor core turns directly into steam in the reactor vessel and is then used to power the turbine-generator. In the PWR, the water passing through the reactor core is kept under pressure so that it does not turn to steam at all, it remains liquid. Steam to drive the turbine is generated in a separate piece of equipment called a steam generator. Mostly these steam generators (UT Tube Steam Generators) are used in PWRs. Nuclear power plants generate electricity by driving the armature coupled to a steam turbine. The steam is generated by the UTSG. The water level of the UTSG should be maintained within safe limits. A too high of a water level produces wet steam that could damage the turbine blades, therefore, the turbine trips off. On the other hand, too low of water-level causes poor cooling of the nuclear reactor, therefore, the reactor trips off. In both cases the power plant shuts down unintentionally. The water level regulation of UTSG is a very difficult control problem, and it is one of the major reasons for unintended shutdowns of nuclear power plants. The difficulty arises due to reasons such as non-linearity of dynamics, non-minimum phase dynamics (also known as reverse dynamics), and unreliable sensor feedback (at low power) [1]. Our main objective is to design a intelligent controller while overcoming these obstacles.

It is very difficult to model the water level of the UTSG by thermodynamics. But Irving [2] developed a simplified linear
dynamic model, in which the model parameters change as the operating point changes. He also specified those parameters for five specific operating points. Irvings model is the most popular UTSG model in control research, and it assumes that the reverse dynamics of feed-water and steam to be identical.

Kim [7] introduced a new model saying, that the non-minimum phase dynamics of feed-water and steam should not necessarily be identical, and distinguished the two effects introducing two more model parameters to the standard Irvings model. The Kims model seems more general than Irvings model and it can be converted to Irvings model by equating non-minimum phase parameters.

Most of UTSG controller designs have considered identical reverse dynamics for feed water and steam flows to simplify modeling. This is a good approximation, yet not necessarily true in general. The results so far reported on UTSG controller performance have been limited to a rejection of a single pre-determined steam disturbance, or a single level tracking without the presence of disturbances. These scenarios are ideal, whereas in real practice the reference level changes and disturbances occur independently, and frequently overlap each other so that the controller stability becomes a worried concern. In this project we consider the reverse dynamics separately and presence of disturbances of non-minimum phase plant modeling. It is extremely difficult to provide a hard stability proof for intelligent controllers. Therefore the training data for this intelligent controller is taken from a prolonged simulation using few PI controllers that are specific to different operating power levels to demonstrate the stability.

The data-driven intelligent controller was constructed in this project according to the Kims model which the reverse dynamics of the feed water and steam are not identical. The intelligent controller which we designed, has 5 inputs, 80 TS fuzzy rules, 532 parameters, and 1 overall output, and it was trained by the ANFIS hybrid algorithm [9]. The trained intelligent controller was used to regulate the UTSG water-level under the conditions similar to actual UTSG operations, where steam disturbances and reference level changes occur randomly, while overlapping each other. The 10 hour simulation of the plant under these conditions provides convincing results about the stability of the fuzzy controller, and the low RMS error of the water-level verifies its capability.

II. NON-MINIMUM PHASE PLANT

A discrete system is said to be a non-minimum phase process if at least one of the zeros of the transfer function is located out-side the unit circle. Non-minimum phase dynamics of the UTSG known as swell and shrink behaviors and those can be describe as follows. Swelling behavior refers to a temporary increase in water level in response to a reduction of liquid water mass in the steam generator. Swelling is momentarily observed when steam flow rate undergoes a sudden increment \((v \rightarrow v + \delta v)\) (Fig. 2(a)(d)) or feed-water flow rate undergoes a sudden drop \((u \rightarrow u - \delta u)\). Shrinking behavior is the exact opposite of swelling, and it refers to a temporary decrease in the water level, against an increase of the liquid water mass in the steam generator (Fig. 2(a)(c)). These behaviors, though they last momentarily, are in exact opposition of the response one would expect upon the nature of steam or feed-water flow changes introduced to the system. Because of this kind of reverse behaviors UTSG consider as non minimum phase plant. In [3] and [5], the mechanical oscillation effect (Fig.2e) has been neglected. These assumptions are helpful to simplify non minimum phase plant modeling, however, at an expense of loosing credibility to represent actual plants. In this paper, we consider mechanical oscillation effect as well as non identical reverse dynamics for steam and feed-water, therefore, to make the model more accurate in representing actual non minimum phase plants.

The descriptive picture of the UTSG is given in Fig 1. The heat generated at the nuclear reactor is taken away by forced-circulated water in the primary inlet. Since the radioactive particles can be contained in that water, in the PWRs, primary circuit is isolated from the rest of the system. Primary circuit has an inverted u-tube bundle submerged in the water column of the steam generator, where the heat transfer takes place from primary circuit to secondary circuit that makes secondary circuit water reach the state of bulk-boiling. The generated steam of the secondary circuit (with more than 99.9% dryness) is sent to the turbine, which is coupled to an armature to generate electricity.

As shown in Fig. 1, the water level \(y\) of the UTSG should be maintained within its lower and upper limits. Failure to maintain water level would lead to serious consequences including unintended plant shutdowns and system damages

1) If low water-level exposes the u-tubes, the heat transfer from the primary circuit to the secondary circuit will not take place efficiently. Consequently, primary circuit builds up heat within itself, which causes the reactor to trip off.

![Fig. 1. The U-Tube Steam Generator](image-url)
2) If the water level rises too high, the steam will contain more moisture (dryness less than 99.9%). And, the wet steam may damage the turbine blades, therefore, turbine trips off.

Eventhough UTSG dynamics shows high nonlinearity and non-minimum phase behavior, the water level can be approximately represent the following linearized model for a given power level [7].

\[
Y(s) = \frac{G_1 (U(s) - V(s))}{1+\tau_1 s} - \frac{G_2 V(s)}{1+\tau_2 s} - \frac{G_3}{1+\tau_3 s} - \frac{G_4}{s^2 + 2\pi s + T^2} U(s)
\]

where the four terms on the RHS are; mass capacity effect, non-minimum phase effect of feed-water, non-minimum phase effects of steam, and the effect of mechanical oscillation, in that order. The model parameters of (1), i.e., \(\{G_1, G_2, G_3, G_4, \tau_1, \tau_2, \tau_3, T\}\) are given in Table I for a generic plant. These parameters were originally published by Irving [2] for an ideal plant. Here we consider the reverse dynamics of feed water and steam independently and the mechanical oscillations too (1). So the approximations are more general than the other models. Therefore these results are more realistic and accurate to non minimum phase plant.

Figure 2 graphically illustrates UTSG dynamics given in (1) when the plant operates at 50% of its rated power. The reverse dynamics due to feed-water change \(\Delta u\) and steam flow change \(\Delta v\) are shown in Fig. 2(c) and (d) by → A, and ← B, respectively. In this paper, we consider mechanical oscillation effect as well as non identical reverse dynamics for steam and feed-water, therefore, to make the model more accurate in representing actual UTSG plants.

III. PI CONTROL SCHEME

A Proportional-Integral controller or PI is a standard feedback loop component in industrial control applications. It measures an "output" of a process and controls an "input", with a goal of maintaining the output at a target value, which is called the "set point". An example of a PI application is the control of a process temperature, although it can be used to control any measurable variable which can be affected by manipulating some other process variable. For example, it can be used to control pressure, flow rate, chemical composition, force, speed or a number of other variables. Automobile cruise control is an example of an application area outside of the process industries.

The basic idea is that the controller reads a sensor. Then it subtracts the measurement from a desired "set point" to determine an "error". The error is then treated in three different ways simultaneously:

1) Proportional - To handle the present, the error is multiplied by a negative proportional constant P, and sent to the output. P represents the band over which a controller’s output is proportional to the error of the system.

2) Integral - To handle the past, the error is integrated (or averaged, or summed) over a period of time, and then

Fig. 2. UTSG dynamics at 50% of the rated power. All graphs show the deviation from its steady state value at the specified power level. Plant excitations in (a) \(\Delta u = \Delta v = 6.6 \text{ [kg/s]}\), is 1% of the plant flow rates at 50% rated power.
multiplied by a constant I, and added to the proportional output. I represents the steady state error of the system. Using the Proportional term alone it is not possible to reach a steady set point temperature. Real world processes are not perfect and are subject to a number of environmental variables. As these variables are often constant they can be measured and compensated for.

Using two PI controllers, one for level error control and another one for the flow error control, it is possible to regulate the water level at all specific power levels given in Table I. The structure of the two PI controllers is shown in Fig. 3, whereas the PI control law of the two controllers is given by

\[
\Delta u^l(t) = k_p^l e^l(t) + k_i^l \int_0^t e^l(t) dt \tag{2}
\]

\[
\Delta u^f(t) = k_p^f e^f(t) + k_i^f \int_0^t e^f(t) dt \tag{3}
\]

And, the water-level adjustment (increment or decrement) is determined by summing the outputs of the two controllers as follows

\[
\Delta u(t) = \Delta u^l(t) + \Delta u^f(t) \tag{4}
\]

IV. DESIGN THE INTELLIGENT CONTROLLER

According to above PI control structure, a prolonged simulation of the UTSG plant was carried out under the conditions given in Table II. Starting from the beginning of the simulation, the reference water-level was intentionally changed in 5[mm] steps in every 300[s] intervals, over the entire range of 100[mm] → 120[mm] → 80[mm] → 100[mm], which takes 5100[s] in total. Then, another two hours was given for random reference changes. At all times, random steam disturbance were introduced according to the specifications given in table II.

\[
\Delta u(t) = \Delta u^l(t) + \Delta u^f(t) \tag{4}
\]

Fig. 3. Two PI controller system to regulate UTSG water-level

The total duration of one simulation epoch is therefore 12300[s]. This simulation was iteratively carried out while intuitively tuning PI parameters of the level error controller, i.e. \(k_p^l\) and \(k_i^l\), and the near-optimal settings shown in table III were found. The PI gains for flow error controller, i.e. \(k_p^f\) and \(k_i^f\) were set to acceptable values and kept unchanged for the sake of simplicity, and reduced dimensionality in intuitive tuning of the control gains. There are six random number sequences in this simulation, i.e. three for water level change (decision, magnitude, and sign), and three for steam disturbances (occurrence, magnitude, and sign), which were not maintained constant in repetitive simulations during gain tuning process. Therefore, every iteration generates a different random number sequence. We argue that it does not affect the gain tuning process because the simulation duration is sufficiently long (12300[s]) that the RMS error of the water level would not be affected as a result of not using constant random number sequences in each iteration. On the other hand, using different random number sequences in successive simulation epochs is essential for a generalization of the tuning process. Following data have been obtained under the control of the tuned PI controllers.

1) level error, \(e^l = r - y\)
2) percentage flow error, \(e^f\% = e^f / v_p \times 100\)
3) accumulated level error, \(\int e^l dt\)
4) accumulated flow error, \(\int e^f dt\)
5) percentage change in flow rate, \(\Delta u\% = \Delta u / v_p\)

Intelligent control has emerged as an alternative or complement to conventional control strategies in many engineering areas, especially in robotics. Intelligent control theory usually provides nonlinear controllers that are capable of performing various nonlinear control actions. If the parameters of the intelligent controllers are chosen appropriately, it is also possible for them to work for uncertain nonlinear systems. In addition, intelligent controllers are capable of handling many complex situations such as some control systems with large uncertainties in process parameters and/or systems structures, as well as some ill-modeled or linguistically described physical systems.

The difficulty of modeling and control of UTSG water-level inspired researchers to investigate data-driven techniques such as fuzzy and adaptive learning systems. Fuzzy reasoning can be used to interpret uncertain, incomplete data in order
TABLE I

<table>
<thead>
<tr>
<th>p[%]</th>
<th>G_1</th>
<th>G_2</th>
<th>G_3</th>
<th>G_4</th>
<th>T</th>
<th>ω_p</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.058</td>
<td>7.704</td>
<td>9.63</td>
<td>0.181</td>
<td>41.9</td>
<td>38.72</td>
</tr>
<tr>
<td>15</td>
<td>0.058</td>
<td>3.568</td>
<td>4.45</td>
<td>0.226</td>
<td>26.3</td>
<td>17.20</td>
</tr>
<tr>
<td>30</td>
<td>0.058</td>
<td>1.464</td>
<td>1.83</td>
<td>0.310</td>
<td>3.6</td>
<td>43.40</td>
</tr>
<tr>
<td>50</td>
<td>0.058</td>
<td>0.840</td>
<td>1.05</td>
<td>0.215</td>
<td>34.8</td>
<td>2.88</td>
</tr>
<tr>
<td>100</td>
<td>0.058</td>
<td>0.376</td>
<td>0.47</td>
<td>0.105</td>
<td>28.6</td>
<td>2.72</td>
</tr>
</tbody>
</table>

Fig. 4. Takagi-Sugeno type adaptive neuro-fuzzy system

then

$$\Delta u_i = \alpha_i e^l + \beta_i e^f + \gamma_i \int e^l dt + \epsilon_i \int e^f dt + \rho_i p + \sigma_i$$

$$w_i = \mu_{e^l}(A_j) \times \mu_{e^f}(B_k) \times \mu_{e^l}(C_l) \times \mu_{e^f}(D_m) \times \mu_p(E_n)$$

The output of the TS fuzzy system would be the weighted sum of the individual rule outputs as given by

$$\Delta u = \frac{\sum_{i=1}^{N} w_i \Delta u_i}{\sum_{i=1}^{N} w_i}$$

where \(N = j \times k \times l \times m \times n = 80\) is the number of rules as \(j = k = l = m = 2\) and \(n = 5\) are the number of fuzzy labels used for corresponding inputs. The fuzzy system output for \(\Delta u\) is compared with the corresponding PI controller output for \(\Delta u\) and the mismatch is used to adapt the consequent parameters \(\alpha_i, \beta_i, \gamma_i, \epsilon_i, \rho_i, \sigma_i\) and premise parameters \(a_{mf}, b_{mf}, c_{mf}, d_{mf}; m \in \{A_j, B_k, C_l, D_m\}; j, k, l, m \in \{1, 2\}\), that specify the trapezoidal membership functions of \(A_j, B_k, C_l, D_m\). This adaptation algorithm was proposed by Jang [9] which uses least squares estimates of the consequent parameters, and gradient based error back-propagation [10] for adapting premise parameters [6]. We have used two membership functions each for the first four inputs, which were initialized by grid-partitioning of the training data, whereas the fifth input (i.e. power) was assigned five membership functions, which were hand-crafted so that they have unity membership at the respective singleton values of power. Therefore, the number of rules is limited to 80, which is a manageable size for the neuro-fuzzy controller. In the premise part the 13 trapezoidal membership functions require 52 parameters, whereas in the consequent part the 80 rules need 480 coefficients.

V. RESULTS

For a prolonged duration of 10 hours, trained intelligent controller demonstrated comparable performance to PI controller under the plant simulation conditions described in Table II. The trained intelligent controller is able to track random changes in reference water-level satisfactorily, while rejecting random steam disturbances. The initial trapezoidal membership functions of the neuro-fuzzy network are \(A_1[-26.76, -17.58, -3.809, 5.37], A_2[-3.809, 5.37, 19.14, 28.32], B_1[-57.73, -38.5, -9.657, 9.574], B_2[-9.657, 9.574, 38.42, 57.65], C_1[-842.8, -5596, -1348, 1484], C_2[-1348, 1484, 5731, 8563], D_1[-1324, -889.2, -237.3, 197.2], D_2[-237.3, 197.2, 197.2, 237.3],\)
TABLE IV
RMS ERROR WITH PI CONTROL AND NEURO-FUZZY CONTROL

<table>
<thead>
<tr>
<th>p[%]</th>
<th>PI control[mm]</th>
<th>Fuzzy control[mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2.47</td>
<td>2.27</td>
</tr>
<tr>
<td>15</td>
<td>1.96</td>
<td>1.77</td>
</tr>
<tr>
<td>30</td>
<td>0.82</td>
<td>0.78</td>
</tr>
<tr>
<td>50</td>
<td>1.15</td>
<td>1.19</td>
</tr>
<tr>
<td>100</td>
<td>1.35</td>
<td>1.29</td>
</tr>
</tbody>
</table>

849, 1284], E₁[-11.63, -2.125, 12.13, 21.63], E₂[12.13, 21.63, 35.88, 45.38], E₃[35.88, 45.38, 59.63, 69.13], E₄[59.63, 69.13, 83.38, 92.88], and E₅[83.38, 92.88, 107.1, 116.6], and all consequent parameters were zeros. The initial membership functions were generated by grid-partitioning of training data, and it has zero membership value on either side of the training data distribution. This causes problems if an input variable swings beyond the range that grid-partitioning has already specified by looking at the training data. Under such a situation control action may fail, and plant may be destabilized. To eliminate this problem, the initial membership functions were stretched on either side as follows: A₁[-95.61, -86.44, -3.809, 5.37], A₂[-3.809, 5.37, 87.99, 97.17], B₁[-201.9, -182.7, -9.657, 9.574], B₂[-9.657, 9.574, 183.7, 201.9], C₁[-95.61, -26840, -1348, 1484], C₂[-1348, 1484, 26970, 29800], D₁[-4584, -4149, -237.3, 197.2], D₂[-237.3, 197.2, 4108, 4543]. The membership functions for power level were hand-crafted as E₁[1, 0, 8, 12], E₂[8, 12, 18, 27], E₃[18, 27, 33, 47], E₄[33, 47, 53, 97], and E₅[93, 97, 100, 101] using the known power levels. The boldface represents the modified premise parameters. Then, the neuro-fuzzy controller was trained for 15 epochs that showed a negligible error right from the beginning. The premise parameters of the trained fuzzy controller were rule1[4.704, 0.5, 0.1742, 1.742, -3.787×10⁻⁵, -7.57310⁻⁶], rule2[1.549, 0.1452, 0.1106, 0.5531, -0.2274, -0.01516], rule3[0.3962, 0.04859, 1.832, 0.2618, -0.1893, -0.006311] ... rule80[0.1951, 0.004271, 0.1389, 0.0697, 0.0553, 0.000553]. The RMS errors of UTSG water-level control under PI and fuzzy controllers are listed in Table IV.

The neuro-fuzzy controller is trained to learn the principle of PI control. The training data are produced by five specific PI controllers at each power level. Plant dynamics and PI gains undergo significant variations on the operating power. Therefore, neuro-fuzzy controller learns the PI control, as it sees through the actions of all five PI controllers. Therefore it may perform differently (better or worse) compared to PI controller at a specific power level. However, according to the results, the trained fuzzy controller performance is significantly comparable to the PI controller at all power levels.

VI. CONCLUSION

Intelligent controller has been developed to stabilize a non-minimum phase plant (UTSG) using PI control scheme. In comparison with other reported results, steam disturbances have not been included in [5], whereas [3] and [4] have simulated only a single change in reference water-level, and a single steam disturbance that are perfectly isolated in the timeline. It is, however, important to test how a UTSG controller performs when these events overlap, which is more likely the realistic situation. In our work, we have dealt with random disturbances and random changes in the reference water level, in that the randomness appears in the instance of occurrence, magnitude, and sign of these events. It makes the simulation in this paper more realistic, and that the results more trustworthy. This intelligent controller delivered satisfactory performance at all specific power levels. System stability was demonstrated by carrying out simulations for prolonged durations under the conditions similar to real system operations. This proposed intelligent controller can be trained off-line for any UTSG plant, given the actual data.

ACKNOWLEDGMENT

I would like to thank Dr. Sudath R. Munasinghe at the Electronic and Telecommunication Department, University of Moratuwa, Sri Lanka, for his advise and guidance for this work.

REFERENCES