CHAPTER 4

FUZZY SLIDING MODE AND SIMULATED ANNEALING
BASED PI CONTROLLER TUNING

4.1 GENERAL

In this chapter, tuning of a PI controller for pH process and heat exchanger is carried out with fuzzy sliding mode and simulated annealing based PI controller tuning. Fuzzy sliding mode is designed to enhance the robustness of the system and reduce the influences of the time-varying disturbances. Also, simulated annealing is implemented to find the global optimum solution. The simulation results show that the proposed controller outperforms the classical tuning in stability, convergence and robustness.

4.2 OVERVIEW OF FUZZY SLIDING MODE BASED PI CONTROLLER TUNING

In control theory, SMC is a nonlinear control method that alters the dynamics of a nonlinear system by application of a discontinuous control signal that forces the system to "slide" along a cross-section of the system's normal behavior. The state-feedback control law is not a continuous function of time. Instead, it can switch from one continuous structure to another based on the current position in the state space. The multiple control structures are designed so that the trajectories always move toward an adjacent region with a different control structure. So the ultimate trajectory will not exist entirely within one control structure. Instead, it will slide along the boundaries of the
control structures. The motion of the system as it slides along these boundaries is called a sliding mode. The geometrical locus consisting of the boundaries is called the sliding (hyper) surface. In the context of modern control theory, any variable structure system, like a system under SMC, may be viewed as a special case of a hybrid dynamical system, as the system flows through a continuous state space. It also moves through different discrete control modes. The main strength of sliding mode control is its robustness.

One application of sliding mode controllers is the controlling and tuning of pH neutralization process and heat exchanger. Sliding mode control must be applied with more care than other forms of nonlinear control that have more moderate control action. In particular, because of delays and other imperfections, the hard sliding mode control action can lead to chatter, energy loss, plant damage, and excitation of unmodeled dynamics. The performance of the proposed method has been evaluated through its application in nonlinear processes that are frequently used as benchmarks of nonlinear process control strategies.

4.2.1 Fuzzy Sliding Mode based PI Controller Tuning for pH Process and Heat Exchanger with Bypass And Sensor

In an SMC system, the control commands are adequately designed such that the states will move towards the desired sliding plane. Once the states reach the sliding surface, the system possesses some invariance properties, such as robustness, order reduction and disturbance rejection. The first step to design a sliding mode control is to determine the sliding hyper plane with desired dynamics of the corresponding sliding motion. And the next step is to design the control input so that the state trajectories are driven and attracted towards the sliding hyper plane.
Chattering is defined as the occasional unwanted vibrations occurring in a process. However, the most disadvantage of using SMC is the chattering phenomenon. Because of discontinuous switching control applied to the plant, chattering always appears as a source to excite the un-modeled high frequency dynamics of the controlled system. One commonly used method to eliminate the chattering is to replace the relay control by a saturating approximation. Another method is to apply fuzzy logic to the SMC system such that a smooth and reasonable hitting control can be generated to reduce the chattering. The tuning of PI controller is often used as a benchmark for all kinds of controllers.

The fuzzy sliding mode based PI controller tuning for pH process is shown in Figure 4.1. Sliding mode control with fuzzy logic control techniques is employed for improving the robustness and performance of nonlinear systems with uncertainty in this process. The pH process block can be replaced by a heat exchanger with bypass and sensor in a heat exchanger process.

![Figure 4.1 Fuzzy sliding mode based PI controller tuning for pH process](image)

Thus fuzzy sliding mode based controller tuning is used to eliminate the chattering in the process completely and to obtain a better output with no overshoot. The obtained results are described in the next section.
4.3 FUZZY SLIDING MODE BASED PI CONTROLLER TUNING RESULTS

The tuning of a pH process as well as heat exchanger with bypass and sensor is carried out using this technique. Here sliding mode control for obtaining the optimal tuning of the processes is simulated using MATLAB.

4.3.1 Simulation Results for pH Process

It is found that the development of sliding mode fuzzy logic makes the tuning of pH process more efficient. By proper tuning, the obtained proportional and integral gain values are 0.8 and 5 respectively. The input membership functions for error and change in error for proportional gain ($K_p$) are shown in Figure 4.2 and Figure 4.3 respectively. Their corresponding output membership function and surface plot are shown in Figure 4.4 and Figure 4.5. Initially sliding gain value of 0.8 is chosen. Then fuzzy logic rules are framed. Seven membership functions are chosen for obtaining proper gain values.

![Membership Function Plots](image)

Figure 4.2 Input membership function plot for error ($K_p$)
Figure 4.3 Input membership function plot for change in error ($K_p$)

Figure 4.4 Output membership function plot ($K_p$)

Figure 4.5 Surface function plot ($K_p$)
Similarly the input functions for error and change in error for integral gain \( (K_i) \) are shown in Figure 4.6 and Figure 4.7 respectively. Their corresponding output membership function and surface plot are shown in Figure 4.8 and Figure 4.9. Seven triangular membership functions (NB, NM, NS, ZO, PS, PM, PB) are chosen for accurate tuning in this process.

![Membership Function Plots](image1)

**Figure 4.6 Input membership function plot for error \((K_i)\)**

![Membership Function Plots](image2)

**Figure 4.7 Input membership function plot for change in error \((K_i)\)**
The development of fuzzy sliding mode based PI controller tuning makes the pH process more efficient by completely eliminating the overshoot. Servo and regulatory responses of pH using sliding mode fuzzy logic control are shown in Figure 4.10 and Figure 4.11. In Figure 4.11, for regulatory system, overshoot occurs and then it becomes stable. The simulation results are tabulated and shown in Table 4.1 and Table 4.2. It is seen from the Tables 4.1 and 4.2, that using fuzzy sliding mode based PI controller tuning for a servo pH process, the overshoot is completely eliminated.
4.3.2 Simulation Results for a Heat Exchanger with Bypass and Sensor

Then tuning has been extended to heat exchanger with bypass and sensor. The output responses (Servo, Regulatory) of heat exchanger using fuzzy sliding mode based PI controller tuning are shown in Figure 4.12 and Figure 4.13. As seen in Figures 4.12 and 4.13, a small undershoot initially occurs due to the system transfer function.
The obtained proportional and integral gain values are 0.92 and 5.6 respectively. The simulation are tabulated and shown in Table 4.3 and Table 4.4 respectively. It is seen from the tables, that in fuzzy sliding mode based PI controller tuned servo process, the overshoot is completely eliminated.
4.4 OVERVIEW OF SIMULATED ANNEALING

SA is a generic, probabilistic and metaheuristic for the global optimization problem. It is often used when the search space is discrete. For certain problems, simulated annealing may be more efficient than exhaustive enumeration; provided that the goal is merely to find an acceptably good solution in a fixed amount of time, rather than the best possible solution. The name and inspiration come from annealing in metallurgy, a technique involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects.

By analogy with this physical process, each step of the SA algorithm replaces the current solution by a random "nearby" solution, chosen with a probability that depends both on the difference between the corresponding function values and also on a global parameter (called the temperature).

4.4.1 The Basic Iteration

At each step, the SA heuristic considers some neighboring state s' of the current state s, and probabilistically decides between moving the system to state s' or staying in state s. These probabilities ultimately lead the system to move to states of lower energy. Typically this step is repeated until the system reaches a state that is good enough for the application, or until a given computation budget has been exhausted.

4.4.2 The Neighbors of a State

The neighbors of a state are new states of the problem that are produced after altering the given state in some particular way. In the traveling salesman problem, each state is typically defined as a particular permutation
of the cities to be visited. The neighbors of some particular permutation are
the permutations that are produced by interchanging a pair of adjacent cities.
The action taken to alter the solution in order to find neighboring solutions is
called "move" and different "moves" give different neighbors. These moves
usually result in minimal alterations of the solution.

4.4.3 Acceptance Probabilities

The probability of making the transition from the current state \( s \) to a
candidate new state \( s' \) depends on the energies \( e = E(s) \) and \( e' = E(s') \) of the
two states, and on a global time-varying parameter called the temperature.

4.4.4 The Annealing Schedule

The effect of cooling schedule on the performance of simulated
annealing is shown in Figure 4.14 and 4.15. The problem is to rearrange the
pixels of an image so as to minimize a certain potential energy function,
which causes similar colors to attract at short range and repel at a slightly
larger distance. These images obtained with a fast cooling schedule is shown
in Figure 4.14 and a slow cooling schedule is shown in Figure 4.15, producing
results similar to amorphous and crystalline solids, respectively.

![Figure 4.14 Fast cooling schedules](image-url)
For any given finite problem, the time required to ensure a significant probability of success will usually exceed the time required for a complete search of the solution space.

4.4.5 Pseudocode

The pseudocode presents the simulated annealing heuristic. It starts from an initial state and continues to a maximum number of steps.

4.4.6 Selecting the Parameters

In order to apply the SA method to a specific problem, the following parameters are to be specified. The state space, the energy (goal) function, the acceptance probability function, and initial temperature. These choices can have a significant impact on the method's effectiveness.

4.4.7 Diameter of the Search Graph

An essential requirement for the neighbor function is that it must provide a sufficiently short path from the initial state to any state which may be the global optimum.
4.4.8 Transition Probabilities

For each edge (s, s') of the search graph, it is necessary to define a transition probability, which is the probability that the SA algorithm will move to state s' when its current state is s.

4.4.9 Acceptance Probabilities

In practice, it is common to use the same acceptance function for many problems. As a result, the transition probabilities of the simulated annealing algorithm do not correspond to the transitions of the analogous physical system.

4.4.10 Efficient Candidate Generation

When choosing the candidate generator neighbor, it must be considered that after a few iterations of the SA algorithm, the current state is expected to have much lower energy than a random state.

4.4.11 Barrier Avoidance

When choosing the candidate generator neighbor it must also try to reduce the number of "deep" local minima-states (or sets of connected states) that have much lower energy than all its neighboring states. Such "closed catchment basins" of the energy function may trap the SA algorithm with high probability and for a very long time.

4.4.12 Cooling Schedule

The relaxation time (the time one must wait for the equilibrium to be restored after a change in temperature) strongly depends on the "topography" of the energy function and on the current temperature. In the SA
algorithm, the relaxation time also depends on the candidate generator, in a very complicated way. All these parameters are usually provided as black box functions to the SA algorithm. Therefore, in practice the ideal cooling rate cannot be determined beforehand, and should be empirically adjusted for each problem.

4.4.13 Restarts

Sometimes it is better to move back to a solution that was significantly better rather than always moving from the current state. This process is called restarting of simulated annealing.

4.5 SA-BASED TUNING FOR PI CONTROLLER

Simulated Annealing is motivated by an analogy to annealing in solids. Normally this algorithm simulates the cooling process, by gradually lowering the temperature of the system until it converges to a steady frozen state. Hence simulated algorithm is applied to optimization problems. Simulated Annealing is used to search for feasible solutions and converge to an optimal solution.

This section gives a strategy based on simulated annealing for the optimal tuning of a PI controller.

The steps involved for the tuning process are as follows:

Step 1 : Initialize – Start with a random initial placement. Initialize a very high “temperature”.

Step 2 : Move – Perturb the placement through a defined move.

Step 3 : Calculate score – Calculate the change in the score due to the move made.
Step 4 : Choose – Depending on the change in score, accept or reject the move. The probability of acceptance depending on the current “temperature”.

Step 5 : Update and repeat – Update the temperature value by lowering the temperature.

Go back to Step 2.

The process is iterated until “Freezing Point” is reached.

4.5.1 Simulation Results for a pH Process

Here SA-based PI parameter tuning for obtaining the optimal design of the pH process is simulated. Optimal PI settings are computed by means of optimization based on the algorithm. Servo and regulatory responses of simulated annealing tuned PI control for pH process are shown in Figure 4.16 and in Figure 4.17. The simulation results for servo and regulatory system are tabulated in Table 4.1 and Table 4.2 respectively.

It is seen that SA tuned PI controller is better than fuzzy sliding mode based PI tuned controller in various aspects like delay time, peak time, settling time and rise time. But it has an overshoot of about of about 7.9 % for servo process whereas no overshoot in fuzzy sliding mode based PI tuned process. Overshoot is only about 7.5 % for regulatory pH process.

The number of iterations used in this algorithm is 1000. The best fitness value obtained is 751.15. Also using SA tuning, the controller parameters obtained are $K_p = 1.1542$ and $K_i = 2.4627$. 
Figure 4.16 Servo response of SA tuned pH process

Figure 4.17 Regulatory response of SA tuned pH process
Figure 4.18 Output responses of ISE, IAE, ITAE using SA for a pH process (Servo)

Figure 4.19 Output responses of ISE, IAE, ITAE using SA for a pH process (Regulatory)
The different errors are integral square error, integral absolute error and integral time absolute error. For any of the possible criteria, the best response corresponds to the minimum value of the chosen criterion. These error responses are used for the analysis of the controller output performance. Output Responses of ISE, IAE, ITAE using SA are shown in Figure 4.18 and in Figure 4.19. ISE, IAE, ITAE using SA for servo and regulatory pH process are about 101.7, 755.3, 966.3 and 141.4, 787.5, 1002.8. Thus controller tuning with minimum ISE can be employed in this process.

**Table 4.1** Comparison of fuzzy sliding mode and SA based PI controller tuning for a pH process (Servo)

<table>
<thead>
<tr>
<th>Tuning Methods</th>
<th>Delay Time (sec)</th>
<th>Rise Time (sec)</th>
<th>Peak Time (sec)</th>
<th>Settling Time (sec)</th>
<th>Peak Overshoot (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy Sliding Mode</td>
<td>3.5</td>
<td>7.0</td>
<td>7.0</td>
<td>5.0</td>
<td>0</td>
</tr>
<tr>
<td>SA</td>
<td>0.9</td>
<td>1.75</td>
<td>2.5</td>
<td>2.4</td>
<td>7.9</td>
</tr>
</tbody>
</table>

**Table 4.2** Comparison of fuzzy sliding mode and SA based PI controller tuning for a pH process (Regulatory)

<table>
<thead>
<tr>
<th>Tuning Methods</th>
<th>Delay Time (sec)</th>
<th>Rise Time (sec)</th>
<th>Peak Time (sec)</th>
<th>Settling Time (sec)</th>
<th>Peak Overshoot (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy Sliding Mode</td>
<td>3.33</td>
<td>6.66</td>
<td>26.5</td>
<td>24.66</td>
<td>8.4</td>
</tr>
<tr>
<td>SA</td>
<td>0.64</td>
<td>1.28</td>
<td>24.33</td>
<td>24.12</td>
<td>7.5</td>
</tr>
</tbody>
</table>
4.5.2 Simulation Results for a Heat Exchanger with Bypass and Sensor

SA-based PI controller tuning for obtaining the optimal values of heat exchanger is simulated. Servo and regulatory responses of simulated annealing tuned PI control for heat exchanger are shown in Figure 4.20 and Figure 4.21. Results obtained are tabulated in Table 4.3 and Table 4.4 respectively. It is seen that SA tuned PI controller is better than fuzzy logic based PI controller tuning in various aspects like delay time, peak time, rise time and peak overshoot. But as seen from Table 4.3, for a servo heat exchanger process the settling time is about 13.75 sec, whereas in fuzzy logic based PI controller tuning it is only 13.5 sec.

Using SA, the controller parameters obtained after simulation are $K_p = 7.7148$ and $K_i = 1.1104$. Thus the development of SA tuned PI control makes the process more efficient than that of using fuzzy logic based PI controller tuned process.

Figure 4.20 Servo response of SA tuned heat exchanger process
Figure 4.21 Regulatory response of SA tuned heat exchanger process

The different errors ISE, IAE and ITAE are used for the analysis of the controller output performance. Output responses of ISE, IAE, ITAE using SA for servo and regulatory process are shown in Figure 4.22 and Figure 4.23. ISE, IAE, ITAE using SA for servo and regulatory system are 8.6809, 10.7106, 66.5054 and 9.4565, 12.4451, 102.84. Thus controller tuning with minimum ISE can be employed in this process.

Figure 4.22 Output responses of ISE, IAE, ITAE using SA (Servo)
Figure 4.23 Output responses of ISE, IAE, ITAE using SA (Regulatory)

Table 4.3 Comparison of fuzzy sliding mode and SA based PI controller tuning for a heat exchanger (Servo)

<table>
<thead>
<tr>
<th>Tuning Methods</th>
<th>Delay Time (sec)</th>
<th>Rise Time (sec)</th>
<th>Peak Time (sec)</th>
<th>Settling Time (sec)</th>
<th>Peak Overshoot (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy Sliding Mode</td>
<td>4.5</td>
<td>9.0</td>
<td>11.0</td>
<td>13.5</td>
<td>0</td>
</tr>
<tr>
<td>SA</td>
<td>1.4</td>
<td>2.8</td>
<td>2.0</td>
<td>13.75</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4.4 Comparison of fuzzy sliding mode and SA based PI controller tuning for a heat exchanger (Regulatory)

<table>
<thead>
<tr>
<th>Tuning Methods</th>
<th>Delay Time (sec)</th>
<th>Rise Time (sec)</th>
<th>Peak Time (sec)</th>
<th>Settling Time (sec)</th>
<th>Peak Overshoot (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy Sliding Mode</td>
<td>6.1</td>
<td>12.2</td>
<td>11.0</td>
<td>14.1</td>
<td>0.5</td>
</tr>
<tr>
<td>SA</td>
<td>1.32</td>
<td>2.64</td>
<td>3.175</td>
<td>3.75</td>
<td>0</td>
</tr>
</tbody>
</table>
4.6 SUMMARY

It has been described in this chapter, about the tuning of a pH process as well as heat exchanger with bypass and sensor using fuzzy logic based PI controller tuning and SA based PI controller tuning. Initially the tuning of pH process was developed using fuzzy logic based PI controller tuning. The best advantage of using this method is to avoid chattering. Also the overshoot is completely eliminated for servo pH process. The rise time, peak time, delay time and settling time are more using this tuning method. In order to minimize the rise time, peak time, delay time and settling time, another recently developed tuning method, SA which has been tested through extensive simulation is proposed. Using SA-based tuning approach, the obtained rise time, peak time and delay time are less than fuzzy logic based PI controller tuned method.

Finally the investigation in this chapter reveals that SA-based tuned method is better than fuzzy logic based PI controller tuned method.