Comparison of Series and Parallel Cascade using PI Controller

V. Dinesh Kumar, V. M. Sivakumar, D. Mahalakshmi, M. Thirumarimurugan

1 Department of Electronics and Instrumentation Engineering, Kumaraguru College of Technology, Coimbatore, India
2 Department of Chemical Engineering, Coimbatore Institute of Technology, Coimbatore, India.
3 Department of Chemical Engineering, Coimbatore Institute of Technology, Coimbatore, India.
4 Department of Chemical Engineering, Coimbatore Institute of Technology, Coimbatore, India.

Received 12 February 2015; accepted 20 March 2016; published 25 March 2016

Address For Correspondence:
V. M. Sivakumar, Department of Chemical Engineering, Coimbatore Institute of Technology, Coimbatore, India.
E-mail: vmsivakumarar@gmail.com

Copyright © 2016 by authors and American-Eurasian Network for Scientific Information (AENSI Publication).
This work is licensed under the Creative Commons Attribution International License (CC BY).
http://creativecommons.org/licenses/by/4.0/

ABSTRACT
A cascade control strategy can be used to reduce the possible disturbances and to enhance the performance of a control system. In this paper, comparison is made between series cascade system and the parallel type by using PI controller. Normally PI controller provides the better response in comparison with other conventional type because of the absence of derivative error. Simulation results for both series and parallel type is obtained and the result is compared using peak overshoot, settling time and steady state error by using MATLAB code. It is shown that the overshoot is minimized in parallel cascade systems which increases the performance of the process. In this paper, distillation column is taken as an application of cascade control.

KEYWORDS: Cascade systems; load disturbance; Series cascade systems; Parallel cascade control; PI Control; Distillation column.

INTRODUCTION
In process industries, cascade control is extensively used to reduce the effects of possible disturbances and to improve the dynamic performance of the closed-loop system. Cascade control was introduced many years ago by Franks and Worley[1]. The standard feedback control loop sometimes does not provide performance good enough for processes with long time delays and more disturbances. Cascade control loops is a common feature in the control industries for the control of temperature, flow and pressure loops.

In traditional series cascade control, both the manipulated variable and the disturbance affect the primary output through the intermediate (secondary) output. In case of parallel cascade systems, the manipulated and disturbance variables simultaneously affect primary and secondary outputs. Parallel cascade control was first considered by Luyben [2]. The cascade control structure is made of two loops: primary (outer) loop and secondary (inner) loop. In parallel cascade control, the secondary loop dynamics should be much earlier than the primary loop because the disturbances entering into the secondary loop should be discarded immediately so that it reduces the steady state error in the primary loop. The typical example of a parallel cascade control system is the overhead composition control of a distillation column cascaded onto the control of a tray temperature[3]. The reflux flow rate (manipulated variable) and the feed flow or composition (disturbance, d) have an effect on both the purity of the overhead product (primary output, $y_1$) and the tray temperature (intermediate output, $y_2$). The control purpose is to maintain the overhead composition at the setpoint. By controlling the tray temperature, the variation in the feed can be compensated sooner than it disturbs the product composition as shown in Fig 1. If a long delay exists in the primary loop, the cascade control may not give satisfactory closed loop responses to set point changes. To overcome this problem, many researchers use a dead time compensator scheme in the...
primary loop of the series cascade control system[4,5,6]. This paper shows how effective the parallel cascade control structure achieves the steady state response with the minimum overshoot in comparison with series cascade control structure.

Methodology:
In this paper, the comparison between series and parallel cascade control is done with the help of PI controller. Generally PI controller offers the minimum overshoot and the settling time and also there is no steady state error because of the absence of derivative effect. Therefore PI controller has more advantages in the terms of error with that of the PID controller. More often in many industries, where the steady state error is the major criteria, PI controller is used which improves the steady state accuracy by decreasing the steady state error.

Fig. 2 shows the parallel cascade control structure in which $G_{cp}$ and $G_{cs}$ are the primary and the secondary controller. Here $P_1$ and $P_2$ are the process transfer function of outer and the inner loop. First, the primary controller is in manual mode whereas the secondary controller is tuned based on the dynamic form of the secondary process. Then the secondary loop is in automatic mode and the primary controller is tuned using the dynamic model. In this paper, both the outer loop ($G_{cp}$) and the inner loop ($G_{cs}$) use the PI controller.

Series cascade control:
In series cascade control systems, (as shown in Fig. 3) the manipulated variable $r_2$ affects directly one intermediate variable ($y_2$) and this in-turn affects the primary controlled variable ($y_1$). The primary loop controls the controlled variable $y_1$ by manipulating the set point of the secondary controller $G_{cs}$. Thus we have the same controlled variable and set point as like a single feedback loop but the control valve has been augmented by an inner control loop. The disturbance $P_{d2}$ is rejected by the secondary loop before they affect the full process, and thus response is quicker and the impact on $y_1$ is less. The primary loop is necessary to handle the other disturbances, such as $P_{d1}$ that always exist.

The process transfer function for the outer and the inner loop which is given by,

$$P_1 = P_{d1} = \frac{e^{-4s}}{(20s + 1)}$$

$$P_2 = P_{d2} = \frac{1}{(10s + 1)}$$

$P_1$ and $P_2$ are outer loop and inner loop process transfer function whereas $P_{d1}$ and $P_{d2}$ are outer loop and inner loop disturbance transfer function. By using MATLAB, the series cascade control structure is designed as shown in Fig. 3 and the simulation results are found.

Parallel cascade control:
In parallel cascade system both the manipulated variable and the disturbances affect the primary and secondary outputs through parallel action. Disturbance rejection is the major concern in the design of process control systems. In the absence of disturbances, a process will remain at the steady state, and the control is not necessary [11,12]. Several control algorithms were proposed to improve disturbance-rejection capability. Two controllers can be designed separately for servo purposes and disturbance rejection[13,14,15]. This type of approach increases the complexity of control system design by designing two controllers for a single input output pair [16,17]. In order to overcome the load disturbance problems, a combination of primary and secondary measurements, a simpler approach can be taken[18,19]. For this purpose parallel cascade control structure is used, in which the primary controller is designed for the servo purpose and the secondary controller is designed for the disturbance-rejection purpose[20]. The general block diagram of a parallel cascade control structure has been shown in fig (2).

The process transfer function for the outer and the inner loop which is given by,

$$P_1 = P_{d1} = \frac{e^{-4s}}{(20s + 1)}$$

$$P_2 = P_{d2} = \frac{1}{(10s + 1)}$$
P₁ and P₂ are outer loop and inner loop process transfer function whereas P₁d₁ and P₂d₂ are outer loop and inner loop disturbance transfer function. By using MATLAB, the parallel cascade control structure is designed as shown in Fig. 4 and the simulation results are found.

RESULTS AND DISCUSSION

In this paper, specification parameters using series and parallel cascade control strategies have been exhaustively explored. The performance indices obtained are quite straightforward and simple. The results of the output response to a step response in a cascade loop using a described simulink models for series and parallel cascade control are presented in Fig. 5. Both the series and the parallel cascade controller is tuned using the Cohen-Coon (open loop) method. The tuned values for both series and parallel cascade are in Table 1.

The tuned values for K_p and K_i are used to simulate the response of both series and parallel cascade. The results of the output response for a step input using the described simulink models for a series and parallel cascade by conventional PI controller is presented in Fig.5. It is observed from the result that the designed parallel cascade controller possesses better tracking capability and less overshoot. The results are further compared quantitatively in Table 2. Table 2 lists the control parameters for both the cascade controller (series and parallel).

![Fig. 1: An overhead composition control in the distillation column](image1)

**Fig. 1:** An overhead composition control in the distillation column

![Fig. 2: Parallel cascade control](image2)

**Fig. 2:** Parallel cascade control

![Fig. 3: Simulink model of Series cascade control](image3)

**Fig. 3:** Simulink model of Series cascade control
Fig. 4: Simulink model of Series cascade control

Fig. 5: Simulation results of series and parallel cascade controller

Table 1: Open loop PI tuning values for both series and parallel cascade controller

<table>
<thead>
<tr>
<th>Controller</th>
<th>K_p</th>
<th>K_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary loop</td>
<td>0.277</td>
<td>0.033</td>
</tr>
<tr>
<td>Secondary loop</td>
<td>63.48</td>
<td>98.72</td>
</tr>
</tbody>
</table>

Table 2: Control parameters obtained from the obtained response

<table>
<thead>
<tr>
<th>Cascade Controller Parameters</th>
<th>Rise time ($t_r$) in sec</th>
<th>Settling time ($t_s$) in sec</th>
<th>Overshoot ($M_p$) in %</th>
<th>Tracking capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parallel</td>
<td>40</td>
<td>231</td>
<td>5</td>
<td>Good</td>
</tr>
<tr>
<td>Series</td>
<td>55</td>
<td>267</td>
<td>86</td>
<td>Better</td>
</tr>
</tbody>
</table>

Conclusion:
Parallel cascade controller, provides the good tracking capability with very minimum overshoot (5%) and rise time (40 sec). with regards to overshoot, parallel cascade shows the least value (5%) and hence the best behavior is provided.

REFERENCES