ControlMaster CM10, CM30, CM50 and CMF310
Universal process controllers
Getting the best performance from Autotune

Contents
1 Introduction ...................................................... 1
2 How autotune works ........................................ 1
3 How ControlMaster autotune works ................. 1
4 Tuning a process ............................................. 2
   4.1 Cascade ....................................................... 2
   4.2 Gain scheduling ............................................... 3
   4.3 Split output ..................................................... 3

1 Introduction
This publication describes how to use the autotune facility and advise how best it can be applied to the different template types available on ControlMaster.

Autotune is a user-activated feature that enables the controller’s PID parameters to be set automatically using an 'at set-point' algorithm. This means that the autotune must be performed on a stable process variable that is close to setpoint.

2 How autotune works
Autotune works by modifying the control output and monitoring the process response to that change in order to calculate the optimum PID settings required to control the process. The modification in the control output creates an oscillation in the process by introducing a step change to the output; autotune then monitors the response of the process value to that change. The step change can be 3 %, 5 %, 10 % or 25 %.

Alternatively, there is a maximum option where the controller can be configured to change the output by the maximum amount in both directions from the process set point. For best results, select the largest initial output step the process can tolerate.

3 How ControlMaster autotune works
ControlMaster’s autotune works by cycling the process in a symmetrical fashion. This means that following selection of the output step value, ControlMaster performs both an upward and downward step change to that value. For example, if the step change is set at 10 % and the output is set at 20 % in manual mode, ControlMaster first performs an upward step to 30 %, then drops the output down to 10 %, monitoring the effect on the process in both positions. For this to function correctly, ensure that the control output setting can tolerate a step change of this value in both directions.

Referring to Fig. 3.1, the autotune sequence is as follows:
1. Set the first step value and dynamics required. For best results, select the largest initial output step size that can be tolerated by the process.
2. Ensure the control type is set to PID.
3. Start Autotune from the Operator menu.
4. ControlMaster monitors noise \( A \) and calculates a hysteresis value.
5. User-defined initial step in the output \( B \). When the process exceeds the hysteresis value the output is stepped down.
6. ControlMaster adjusts output amplitude automatically \( C \) to keep PV disturbance to minimum required.
7. When consistent oscillation is established \( D \), the Autotune process is stopped. Optimum settings are calculated from the process dynamics monitored.
4 Tuning a process

When tuning a process, consider how the autotune functions work; the following steps detail how to use the autotune function:

1. Access the Loop Control Page in ControlMaster’s configuration level and ensure:
   a. Control Type is set to ‘PID’
   b. Autotune mode is set to ‘ON’

2. Select the first step size best suited to the process.

3. Exit configuration mode and return to normal operating mode.

4. Select manual control mode.

5. Adjust the control output to a value that allows the step change selected at step 2 to be performed in both positive and negative directions.

6. Check that the process variable is stable and, for best results, close to the setpoint.

7. Press the menu key and select ‘Autotune’ to start autotune.

Most loops can be autotuned in this very simple fashion, however there are some exceptions – see Sections 4.1 to 4.3.

4.1 Cascade

A cascade loop has 2 inter-dependent control loops (master and slave) and both can be autotuned following these procedures. However, it is important to note that where cascade control is used, the slave loop must be tuned correctly first, followed by the master loop. This means running 2 separate autotune processes.

**Fig. 4.1 Cascade with remote setpoint**
4.2 Gain scheduling
Gain scheduling is used to improve non-linear process control. In gain scheduling, ControlMaster uses different PID settings based on the value of a user-selected reference signal. 3 different settings for the PID parameters can be used, within 3 ranges set by the user-defined parameters that are expressed in the engineering range of the gain scheduling reference signal. When the value of the reference signal passes one of these limits, the next set of PID parameters are switched in.

When running autotune, 3 tuning cycles or ‘sets’ are performed (see Fig. 4.2):

- Set 1 – run when the gain scheduling reference signal is less than limit 1. Autotune calculates the PID parameters for Set 1 and copies those values into Sets 2 and 3 until their tune has been performed.
- Set 2 – run when the gain scheduling reference signal exceeds limit 1 but not limit 2. Autotune calculates the the PID parameters for Set 2 but does not modify those already calculated for Set 1.
- Set 3 – run when the gain scheduling reference signal exceeds limit 2. Autotune calculates the PID parameters for Set 3 but does not modify those already calculated for Sets 1 and 2.

All of these autotunes must be performed in the standard way with a stable process variable that is close to the setpoint.

In normal operation, when the gain scheduling reference signal exceeds any of the pre-defined limits, ControlMaster selects the relevant PID parameters to use automatically.

![Fig. 4.2 Gain scheduling](image)

4.3 Split output
Split output divides the PID output into 2 separate outputs, enabling heat/cool and other applications requiring dual outputs to be controlled. The linear relationship between the input from the PID algorithm and the 2 outputs is configured using Min./Max. input/output values as shown in Fig. 4.3.

![Fig. 4.3 Split output](image)

A common problem encountered in applications requiring a split output (for example, heat/cool) is the large differential in the way the 2 outputs react. This means that a single set of PID parameters may not provide sufficient control. The solution is gain scheduling. However, 2 sets of PID parameters are used, one for the heat portion of the output and 1 for the cool portion of the output. This is configured in exactly the same way as the standard gain scheduling, but the control output is the gain scheduling reference source.