PID_TUNE: A PID AUTOTUNING SOFTWARE TOOL ON UNICOS CPC

R. Martí, B. Bradu, E. Blanco Vinuela, CERN, Geneva, Switzerland
L. Frutos, R. Mazaeda, C. Prada, University of Valladolid, Valladolid, Spain

Abstract

PID (Proportional, Integral and Derivative) is the most used feedback control algorithm in the process control industry. Despite its age and thanks to its simplicity in terms of deployment and its efficiency in most of industrial processes, this technique still has a bright future. The major challenge in using PID control is to find the optimal set of parameters to tune the controller. This may be a complex task as it mostly depends on the dynamics of the process being controlled. In this paper a tool able to provide the engineers a set of PID parameters in an automated way is proposed. The tool offers auto-tuning methods, both in open and close loop, and others can be added as it is designed to be flexible. The tool is fully integrated in the framework UNICOS (CERN Standard Control framework) and can be used to tune multiple controllers at the same time, directly from the supervision layer.

INTRODUCTION

The largest majority of controllers found at the regulatory layer in the process industry are of PID (Proportional, Integral and Derivative) type. At CERN, around 4900 PID controllers are employed on the LHC cryogenic control system, 870 are employed on cooling and ventilation installations. In total, more than 8000 PID controllers are used at CERN in control systems based on the UNICOS (Unified Industrial Control System) framework [1]. All these controllers are implemented in more than 300 different PLCs (Programmable Logic Controller). Therefore, a tool to automatically tune the PID parameters would be a valuable asset in terms of controller robustness and hence in improving uptime and operation of the CERN industrial plants.

Although the PID controller has only three parameters, it is not straight forward, without a systematic procedure, to find suitable parameters. The main goal of this work is to provide an auto-tuning tool where different methods are available. The initial implemented methods were selected to cope with real industrial process scenarios. Two main features are searched: easiness of use, to allow operators to perform the task, and flexibility to extend the tool with new methods to cope with complex processes.

The article is organised as follows: first, the real implementation of the tool is depicted, second, the different tuning methods included in the tool are described including their suitability on different processes. Then, real industrial use cases with different auto-tuning methods employed are depicted along with a performance comparison between them. Finally, the paper concludes with the most important ideas and future work.

TOOL IMPLEMENTATION

The PID structure taken into consideration to design the auto-tuning tool is the ISA (International Society of Automation) version used in the UNICOS framework. The control law can be seen in the formula (1)

$$K_c \cdot (e(t) + \frac{1}{T_i} \cdot \int e(t)dt + T_d \cdot \frac{d}{dt}e(t)) \qquad (1)$$

The tool is fully integrated in the UNICOS framework at the supervision layer, the WinCC OA SCADA (Supervisory Control and Data Acquisition) from SIEMENS. It has been implemented in the WinCC OA control scripting language based on ANSI-C. The PID algorithm remains untouched in the PLC (control layer) and only its parameters are set once the tuning process is validated. The architecture and data flow is depicted in Figure 1.

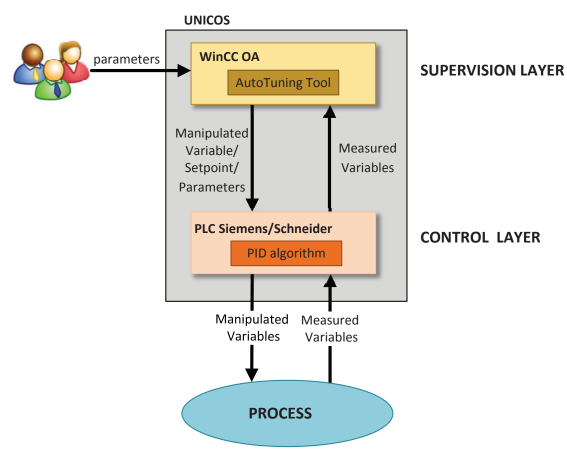

Figure 1: PID auto-tuning tool architecture.

The human interaction is made through dedicated interfaces embedded in the UNICOS controller faceplate. The user can select the method, configure its options, initiate the auto-tuning, evaluate the behaviour and ultimately set the new parameters to the controller. An example of such interface is shown in the Figure 2.

AUTOTUNING: BASIC CONCEPTS AND METHODS

The PID auto-tuning concept refers to the capability to automatically compute the parameters of a PID connected to the real plant. Many different approaches are available. This work focused on developing an auto-tuning tool, PID_TUNE, launched by the operators or the control engineers on their own initiative, performing the chosen tuning algorithm and

ISBN 978-3-95450-148-9

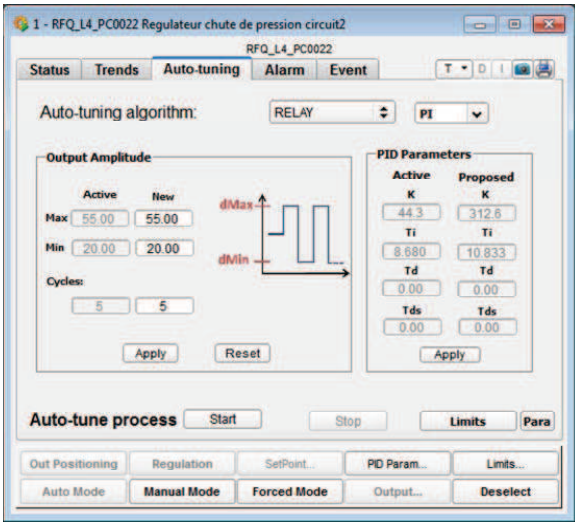

Figure 2: UNICOS auto-tuning user interface: Relay method.

presenting the results to the user who ultimately decides to apply them to the controller.

It is of general consensus that any skilled professional can tune a control loop better than any auto-tuning mechanism, however this task may be very time consuming. However the number of loops and the availability of experts can condition this assessment to the point that an auto-tuning tool can be of great help for the day to day operation of the plants.

The behaviour of an auto-tuning algorithm is based on what an expert would do when tuning that loop by his own. First a phase of observation, then in function of the control objectives selection of the method and finally applying the PID parameters. The first phase includes process data taking and this is done by stimulate the process by either moving the manipulated variable or the controller setpoint. Evidently this settles a first trade off of stimulating enough the process to get meaningful experimental data versus creating a process perturbation which can be fatal for the process itself. To make this feasible, several constraints must be taken into account: e.g. controlled and measured variable limitations, noise.

The selection of a method depends on the control needs which, in turn, depends on the analysis of the control problem. There is no a general rule although the methods available in the tool can cope with the majority of the processes as they are rather universal. Three methods has been included: Relay, SIMC and IFT.

Relay Method

The relay auto-tuning method was proposed to automate the Ziegler-Nichols ultimate cycling tuning method [2], where the critical gain (K_{cr}) and period (P_{cr}) are determined with a proportional-only regulator. The controller gain is gradually increased until an oscillation is obtained. The procedure concept is rather simple but difficult to automate ensuring the safety of the installation as the oscillation amplitude must be always kept under control. To overcome possible instabilities the relay incorporates a hysteresis [3].

The relay is connected in the feedback control loop instead of the PID controller (Fig. 3). It alternates the output between 2 values when the difference between the set-point and the controlled variable changes sign. This is the way it obtains a limited cycle oscillation.

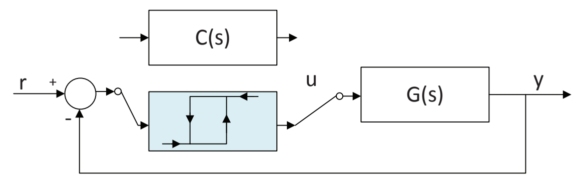

Figure 3: Relay tuning.

Some practical issues must be addressed when implementing this method. First, the measurement noise may give errors in detection of peaks and zero-crossing, thus again a hysteresis is used. It is a common practise to start the method when a steady state [4], of the process is identified to facilitate the sustained oscillations although the tool includes an algorithm to automatically identify steady state [5].

The advantage of this method is the simplicity of its use. The user only has to introduce the amplitude of the hysteresis and the number of cycles to detect sustained oscillations (Fig. 2). The method fits in processes with unknown dynamics.

SIMC

Another well known group of auto-tuning methods is based on plant models [6]. Basically, two different steps compose the methodology, first a process identification phase and then the application of the tuning rules. Obviously this approach is heavily constrained by the availability of the process model.

The developed tool allows users to identify a first-order model by applying a step on the manipulated variable. In addition to this, a steady-state automatic identification algorithm has been included to fully automate the identification phase. Then the tool suggest the best parameters according to well known tabulated values.

The SIMC method was chosen due to its advantages regarding others model based techniques [7]. The main, the simplicity in terms of parameterisation: SIMC has just a single tuning parameter, the desired performance. Its selection is a trade-off between performance ("tight" control) and robustness ("smooth" control). Therefore, the operators or control engineers can easily choose what kind of response they desire in the feedback control loop. The method is applicable to stable and low order systems with not complex dynamics.

ISBN 978-3-95450-148-9

IFT: Iterative Feedback Tuning

The previous two methods have some notably drawbacks: on one side unpleasant disturbances are introduced to the process and, on the other, the need of a model of the plant. Also the algorithms need to disable the PID controller while the auto-tuning process is running. These justified the inclusion of a new method in the tool that ideally does not require the knowledge of any explicit model of the plant and its execution would not disturb much the nominal loop dynamics.

The IFT [8] is a technique inspired by the iterative parametric optimisation approach. It is entirely working in closed-loop and making random but contained perturbations in the loop to find the optimal parameters. It computes a cost function to minimise the current value of the measured value and a desired first order response.

The IFT algorithm is particularly suitable for the industrial use as it fulfils the requirements of being a model-free and closed-loop procedure which does not demand a very severe modification of the nominal conditions. The algorithm has the capability of rejecting efficiently disturbances, crucial for the regulatory control widely deployed in process industries. The IFT tuning algorithm included in the tool has been developed to ease the use of the IFT method, low parameterisation but still offers a minimum set of practical safeguards constraining the freedom of the underlying algorithm during the tuning procedure.

CASE STUDIES

In order to commission and validate the auto-tuning tool together with its algorithms, the tool was deployed in the LHC (Large Hadron Collider) cryogenics simulator. The simulator provides the same UNICOS engineering environment (SCADA and PLCs) as in any production system but connected to a simulated process. The second phase was to deploy it to a real case with complex dynamics, a cooling plant of the LHC.

Auto-tuning on the LHC Cryogenic Simulator

This real-time simulator is able to imitate the real transients of the LHC cryogenics and it embeds a identical version of the real control system. A flow control loop has been selected as candidate to perform the auto-tuning. This PI based control loop regulates the amount of gaseous helium circulating in LHC to maintain the thermal shielding of superconducting magnets around 80 K. It was originally too sluggish, generating flow oscillations and overshoots when disturbed by the circuit pressure changes.

Al three methods (Relay, SIMC and IFT) were applied on this control loop to find its PI parameters. The parameters are compared with the original ones (Table 1).

A significant pressure disturbance was applied in the simulator (t = 2 min) in order to compare the controller performance with the different parameter sets (Figure 4). The three auto-tuning parameter sets are resembling and then provide a controller similar response, but in any case faster and more stable than the original one.

Table 1: Auto-tuning Results on the Cryogenic Simulator

Tuning	Kc	T_i	Overshoot	Oscillation
Original	1	200	22 %	6 %
Relay	9	12	0.5 %	1 %
SIMC	4.5	7	1 %	1.3 %
IFT	13	11	0.5 %	0.5 %

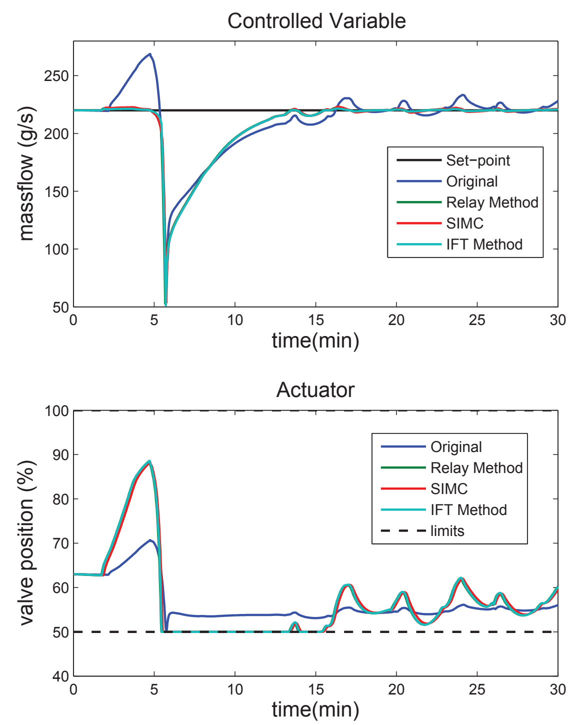

Figure 4: PI controllers performance in simulation.

Auto-tuning on a LHC Cooling Plant

As the first real test on a production system, the tool was used on a chilled water production unit providing water at 5 $°C$ for LHC. The operation team observed instabilities on the condenser output temperature of the chiller at 25 $°C$. The tuning of the associated PI controller was troublesome because this process is always disturbed and unstable due to a noticeable delay time. The temperature deviation to operate the chiller should be less than 1 $°C$ and indeed it was oscillating by around 2 C with the valve moving wildly between 0 % and 100 %. The relay method was the first method used. This allowed to stabilise the temperature with a reasonable control effort on the valve. However there were important disturbances all the time, so the temperature was still oscillating too much (Fig. 5). Then, an auto-tuning in close loop using the IFT method was applied as disturbance

ISBN 978-3-95450-148-9

rejection is one of its features. The results were satisfactory regarding the operation requirements (Fig. 6).

The different tuning parameters found are compared in Table 2 with the corresponding oscillations measured on the controlled temperature ($|\Delta y|$) and on the valve ($|\Delta u|$).

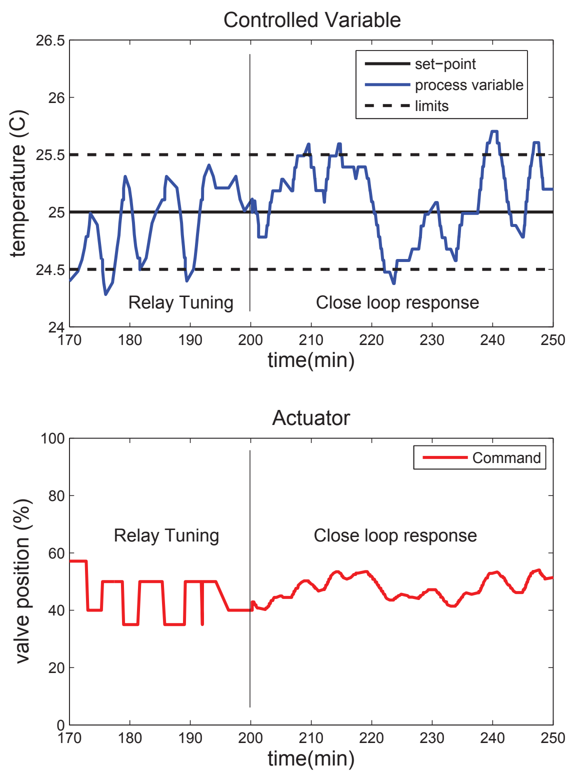

Figure 5: Relay method results on a LHC cooling unit.

Table 2: Auto-tuning Results on a LHC Cooling Unit.

Tuning	Kc	T_i	$\lvert\Delta y\rvert$	$\lvert\Delta u\rvert$
Original	8	0.5	2 °C	100 %
Relay	8.3	311	1.3 °C	10 %
IFT	50.9	5666	0.5 °C	20 %

CONCLUSIONS

An auto-tuning tool for PID controllers is presented in this paper. It is fully integrated, at the supervision level, in the CERN control framework (UNICOS). The tool does not imply any PLC control logic modification. This tool allows operators and control engineers to apply different auto-tuning methods on the regulation loops depending on the process and on the operational constraints. Three initial methods have been implemented and tested: Relay, SIMC and IFT. The tool has been validated in a real scale simulator and then applied in a production system, a LHC cooling plant. It has demonstrated its usability and its performance where the proposed control loops have been properly tuned. In the coming months, this auto-tuning tool will be deployed on all CERN control systems using UNICOS. The tool is

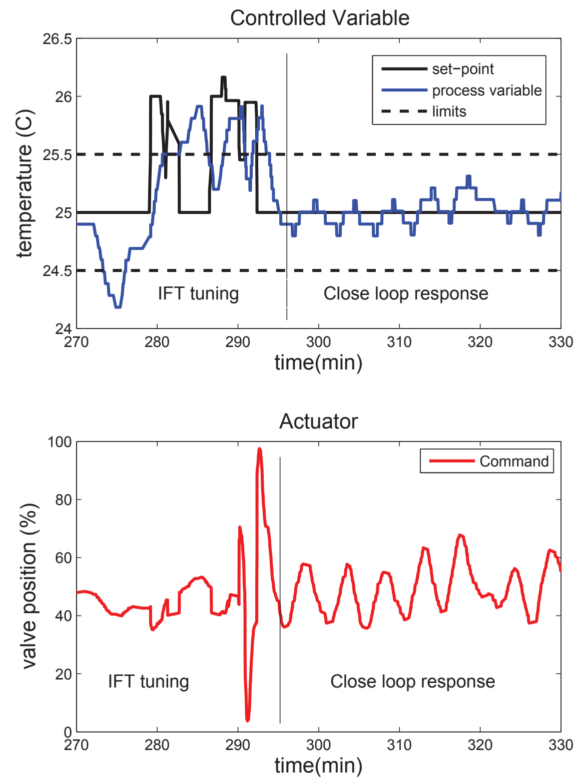

Figure 6: IFT method results on a LHC cooling unit.

designed in a flexible and open way so in the future, other methods could be easily added.

REFERENCES

[1] Ph. Gayet and R. Barillère. "UNICOS a framework to build industry like control systems: Principles Methodology". 10th ICALEPCS Int. Conf. on Accelerator and Large Expt. Physics Control Systemns. Geneva (Switzerland), 2005.

[2] J.G. Ziegler and N.B. Nichols. "Optimum setting for automatic controllers".Transactions of the ASME,64, 1942.

[3] K. Aström and T. Hägglund. "Automatic tuning of simple regulators on phase and amplitude margins".Automatica, 20, 1984.

[4] S. Dormido and F. Morilla. "Auto-tuning and Antiwindup methods in PID controllers".UNED Press, 1995.

[5] R. Russell Rhinehart."Automated Steady and Transient State Identification in Noisy Processes". American Control Conference (ACC), Washington (USA), 2013.

[6] A. O'Dwyer. "Handbook of PI and PID Controller Tuning Rules".Imperial College Press, 2006.

[7] Sigurd Skogestad and Chriss Grimholt. "The SIMC Method for Smooth PID Controller Tuning". PID Control in the Third Millennium.Springer. 2012.

[8] H. Hjalmarsson, M. Gevers and O. Lequin. "Iterative feedback tuning: Theory and applications". IEEE Control Systems Magazine 18(4),1998.

ISBN 978-3-95450-148-9