Robust optimization-based multi-loop PID controller tuning: A new tool and an industrial example

Michael Harmse*, Richard Hughes, Rainer Dittmar*****
Harpreet Singh* and Shabroz Gill*

**IPCOSAptitude Ltd., Cambridge, United Kingdom*
(e-mail: info@ipcosaptitude.com)
***SABIC UK Petrochemichals Ltd., Wilton, Middlesbrough, United Kingdom*
**** West Coast University of Applied Sciences, Heide, Germany*
(e-mail: dittmar@fh-westkueste.de)

Abstract: Modern process plants are highly integrated and as a result, decentralized PID control loops are often strongly interactive. The currently used sequential tuning approach is not only time consuming, but does also not achieve optimal performance of the inherently multivariable control system. This paper describes a method and a software tool which allows a control engineer to calculate optimal PID controller settings for multiloop systems. It is based on the identification of a state space model of the multivariable system, and it uses constrained nonlinear optimization techniques to find the controller parameters. The solution is tailored to the specific control system and PID algorithm to be used. The methodology has been successfully applied in several industrial advanced control projects. The tuning results which have been achieved for interacting PID control loops in the stabilizing section of an industrial Gasoline Treatment Unit at SABIC Petrochemicals are presented.

Keywords: PID controller tuning, multi-loop control, decentralized control system design, nonlinear optimization, genetic algorithm, multivariable system identification.

1. INTRODUCTION

One of the most important challenges facing the process industry today is optimizing the operation of complex units, without compromising the safety and integrity of the process equipment. Process complexity has increased significantly over the past two decades due to increased level of heat integration and use of recycle streams. In addition, the need for increased process flexibility to deal with changing raw materials and alternate energy sources, as well as the need to adapt quickly to fluctuating throughput and quality targets, often means that the process dynamics will vary significantly over time and with operating point. The basic control layer of process plants almost always consists of a large number of decentralized SISO PID controllers, although this approach is intrinsically inadequate for multivariable processes. Due to the situation described above, the interactions between these controllers are becoming more important, and tuning these control loops for good performance and adequate robustness is a challenging task.

The industrial practice of PID controller tuning is still dominated by manual trial-and-error tuning. If tuning rules are used at all, it's the "classical" ones like Ziegler-Nichols or Chien-Hrones-Reswick which are based on simplified first order plus dead time (FOPDT) process models and do not consider stability robustness issues, therefore often being not adequate in modern process units with more complex dynamics and nonlinearity. In addition, many tuning rules assume that all PID controller equations work as described in the textbooks, when in fact there is substantial variation between the different vendors. In contrast, different PID controller structures result due to use of either the parallel or the serial form, using the control error or the PV by the Proportional (P) and Derivative (D) terms, and many other quirks like alternative implementations of the derivative filters. Tuning SISO PID controllers in a multivariable environment is usually done in a time-consuming sequential and iterative way, starting with the most important loops, and heuristic detuning in case the interactions are significant.

For a long time, vendors of automation systems such as Distributed Control Systems (DCS) and Programmable Logic Controllers (PLC) have been offering PID self-tuning functionality (tuning on demand). Unfortunately, they have only found limited application. This is also true for model based PID controller tuning software provided by the same or third-party vendors. Moreover, in most cases these tools are restricted to single loop tuning applications, and do not support multi-loop tuning (Li et al., 2006, Espinosa Oviedo et al., 2006 and Zhu, 2004).

The design of interacting PID controllers in a multivariable environment is not a new topic in the process control literature. At least three research directions can be identified: (1) reduction of controller interactions by proper MV-CV pairing, (2) design of decoupling networks and (3) consideration of MIMO interactions in decentralized controller tuning. In this paper, only the third direction is relevant. Several methods have been developed, Luyben's

BLT method being the most popular one (Monica et al., 1988). Here, the individual PI loops are first tuned by the Ziegler-Nichols rules independently. Then, a detuning factor is calculated which assures a certain stability margin for the controlled MIMO system. All individual controller gains are divided by this factor, and the reset times are multiplied by it. The price to be paid for the reduced interaction is a more sluggish behaviour of PI loops. Other methods include the sequential loop closing approach (Hovd and Skogestad, 1994), the independent design method (Hovd and Skogestad, 1993) and the multivariable generalization of the relay-feedback self-tuning method (Halevi et al. 1997). For a discussion of these methods the reader is referred to (Chen and Seborg, 2003).

This paper introduces a new method and a software tool "AptiTune™" for the calculation of optimum PID controller settings in a multivariable system (multivariable loop tuning). The method consists of several steps. First, a set of Finite Impulse Response (FIR) models of the open-loop MIMO plant is being identified and approximated by a reduced-order state space model. In a second step, optimal parameters for the decentralized PID controllers are calculated using constrained optimization. Finally, the setpoint tracking, disturbance rejection and noise attenuation behaviour of the controlled system is simulated.

It was the aim of the development to come up with a software tool which is based on recent identification and control developments, but which does not require in-depth knowledge of identification and control theory by the average user. Furthermore, the optimization solution is tailored to the specific target automation system, e.g. the particular DCS or PLC which is used for control purposes.

The remainder of the paper is organized as follows: In section 2, the identification and optimal tuning methods will be described together with the "AptiTune™" software tool. Section 3 presents some results of multiloop tuning in the stabilizer section of an industrial Gasoline Treatment Unit (GTU). The retuning of the PID controllers was one of the first steps of an advanced control project, which also included the design and commisioning of an MPC controller.

2. METHOD AND TOOL FOR MULTILOOP TUNING

2.1 Identification of the MIMO process model

The first step of model based multiloop tuning is to develop a dynamic model of the multivariable process with n inputs and n outputs, the outputs (u_i) and process variables (y_i) of the PID controllers shown in Fig. 1.

Our preferred approach is to switch all PID controllers to be tuned into manual mode whenever possible and to perform a series of output steps of different duration and amplitude. According to our experience, four to six steps with duration varying between 10% and 100% of the desired closed-loop settling time are usually sufficient. If a test signal generator is available, PRBS (pseudo-random binary sequence) or GBN

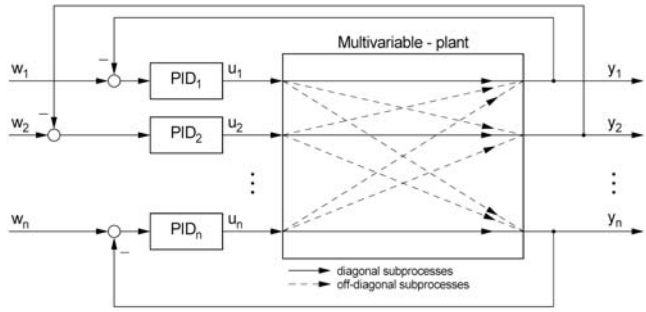


Fig. 1: Decentralized multiloop PID control system

(generalized binary noise), then an automated test may be used as an alternative. Both types of plant tests can be performed in sequential or in time-saving simultanous mode.

If one or more PID controllers cannot be switched to manual mode, then the loop can be kept in automatic mode and multiple setpoint steps can be made. The Projection Method described in (Forsell and Ljung, 2000) can then be used.

After pre-processing the raw test data (detection/rejection of outliers, filtering, decimation, cutting out periods of bad data etc.), the parameters of a MIMO FIR model

$$\vec{g}_{ij} = \left[g_{ij}(0), g_{ij}(1), g_{ij}(2), \ldots, g_{ij}(n_M) \right] \qquad i, j = 1...n \quad (1)$$

are estimated by least squares regression. The user should specify a-priori knowledge such as zero gain, known dead time or integrating behaviour of subprocesses. Although FIR models are estimated, the results are presented as Finite Step Response (FSR) models for easier visualization and understanding. The "AptiTune™" software tool also supports the import of FSR models created by identification tools from MPC packages, but also allows the user to specify a transfer function matrix.

In the next step, the MIMO FIR model is approximated by a linear state-space model of the form

$$\begin{aligned} \dot{x}(t) &= A\, x(t) + B\, u(t) \\ y(t) &= C\, x(t) \end{aligned} \qquad (2)$$

This approximation is not based on the raw or preprocessed plant test data, but on a model-to-model fit. To remove noise and cycles from the FIR model, it can first be smoothed using a central average filter. The state-space model is constructed using the singular value decomposition (SVD) model reduction technique (Maciejowski, 1989). While creating the state-space model, the diagonal model curves are given more preference than the off-diagonal models. As a result, diagonal models normally have higher order than the off-diagonal ones and consequently fit the original FIR model curves more accurately. The step responses calculated based on the stat-space models are graphically displayed.

If it is possible to do a closed-loop step test (or if historical data contain a clear SP step), a practical way of validating the process model is to simulate the closed loop behaviour of the control system with the actual PID controller parameters currently entered on the DCS , and to compare the simulation results with plant data. If the observed responses are similar to the simulated responses, then we can conclude that the model is sufficiently accurate for loop tuning purposes.

2.2 Calculation of optimal PID controller parameters

The PID controller parameters (controller gains $K_{c,i}$, reset times $T_{r,i}$ and derivative time constants $T_{d,i}$) are calculated solving numerically the nonlinear constrained optimization problem

$$\begin{aligned} &\min_{K_{Pi},T_{Ni},T_{Vi}} \quad J \\ &g_j(K_{Pi},T_{Ni},T_{Vi}) \le 0 \quad i=1...n,\ j=1...m \end{aligned} \tag{3}$$

where J denotes the objective function and g_j are constraints. The objective function J is a weighted sum of three terms $J = J_1 + \alpha J_2 + \beta J_3$ which assess different aspects of the control loop performance. The first part $J_1 = \int_0^{T_f} | y(t) - y_r(t) | dt$ refers to the Integrated Absolute Error (IAE) criterion for setpoint tracking. Here, the error is defined as the difference between the PV and a user-defined first order reference trajectory $y_r(t)$ connecting the actual PV and the setpoint. By specifying the time constant of the trajectory, the user can affect the speed of the response to setpoint changes. The second part $J_2 = \int_0^{T_f} | w(t) - y(t) | dt$ denotes the IAE for an input step disturbance. Finally, the third term $J_3 = \int_0^{T_f} |\Delta u(t)|(t)$ reflects the control effort. By setting the weighting coefficients α and β, the user can balance a compromise between the different performance objectives. Another design parameter allows the user to weight the performance of the n SISO control loops against the necessary degree of decoupling between them.

For each control loop, the user can specify one or more inequality constraints $g_j \le 0$ from the following list: maximum OP deviation after setpoint changes, maximum overshoot, minimum damping ratio, maximal noise amplification, process gain and deadtime margins, maximum/minimum limits of the controller parameters. For level buffering controllers, the maximum setpoint deviation and the minimum return time after a level disturbance can be specified. By careful specification of the constraints, the user can tailor the tuning to process-specific requirements.

For starting the numerical optimization, initial controller parameter values have to be selected. For this purpose, the user can choose to use the actual DCS values or values calculated by the Cohen-Coon tuning rule (for individual controller tuning assuming a SISO model). The degree of difficulty of the nonlinear constrained optimization problem depends on the number of controllers involved, the order of the process model, and the number and nature of the inequality constraints. In general, non-convexity and local minima can occur. Therefore, several search algorithms have been implemented, including a brute force global search in the entire parameter space, a genetic algorithm (both intended for initialization), and a generalized gradient algorithm (Vlachos et al., 2000).

In contrast to some PID controller tuning software available mainly for teaching and training purposes, the "AptiTune™" tool not only calculates "generic" PID controller parameters, but parameters for a specific realization of the PID controller equation for specific commercial control system hardware. The user can select between different control algorithms of widespread DCS systems such as Honeywell, Emerson DeltaV, Foxboro I/A, ABB and several others. For example, six different versions of the PID algorithms are available for the Emerson DCS, for which the optimization results may be quite different. Optimal controller parameters can of course also be calculated for P, I only and PI controllers.

After the optimizer has converged and optimal controller parameters have been found, the design process will be finished by simulation of the dynamic behaviour of the control system. It is useful to study different scenarios: setpoint tracking, input disturbance rejection, and noise attenuation. The "integrity" of the controlled MIMO system should be studied as well, i.e. the behaviour of the controlled system if one or more controllers are in manual mode, or if components like final actuators fail. Finally, the robustness against plant-model mismatch should be evaluated. For this purpose, robustness plots such as in Fig. 2 are helpful. It shows the purple stability limit in a process gain ratio/dead time graph, and the stability region for the minimum required combined gain and deadtime margins as the red polygon (left hand plot, lower left hand corner).

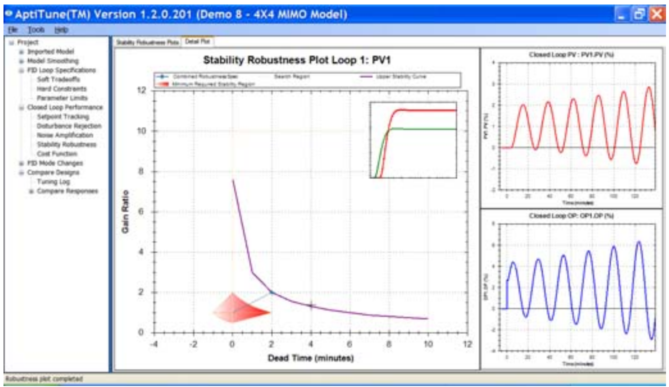


Fig. 2: Robustness plot

3. INDUSTRIAL EXAMPLE

The method and software tool described above have been used successfully in a number of advanced control projects. A good example is the stabilizing section of a GTU process, where improving the PID controller tuning was a prerequisite for successful MPC design and implementation.

The Process and Instrumentation Diagram for the GTU process is shown in Fig. 3. Although the overall system is (8x8), it was possible to decompose it into a (2x2) system on

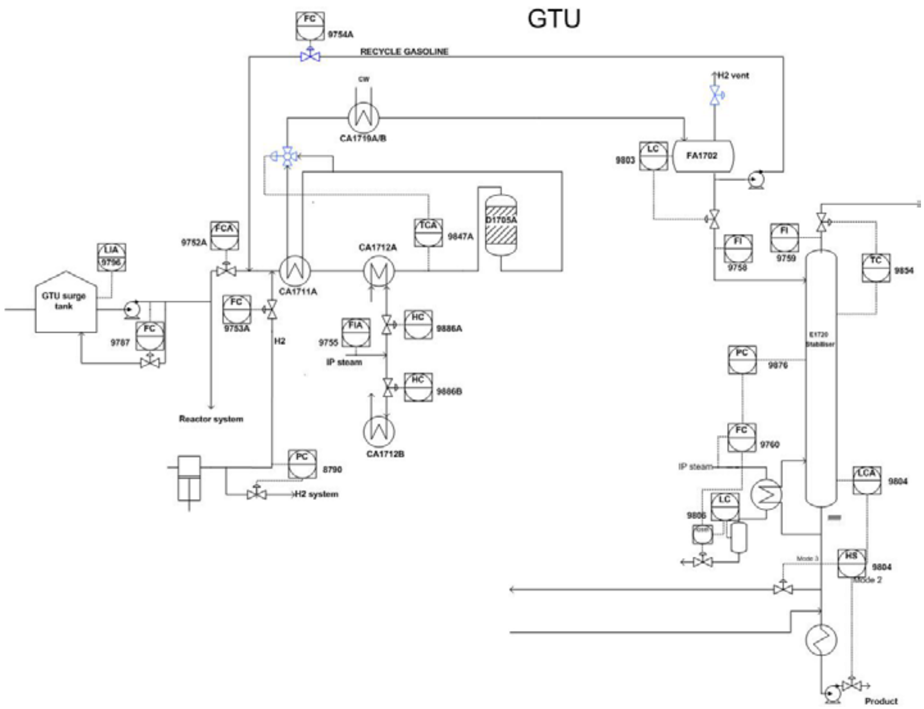


Fig. 3: P&I diagram of the GTU process

the E20 column and (6x6) for the rest of the unit. Due to limited space, the results for the (2x2) system are presented.

The objective of the E20 stabilizer column (on the right hand side of the P&ID) is to remove hydrogen and methane dissolved in the petrol (mostly C5) stream. The column is essentially a degassing drum with trays for improved separation. The current PID control scheme is somewhat unconventional in that the PID loop pairing is the "wrong" way round:

- A tray temperature close to the top of the column is controlled (even though the product specification is on the bottoms stream) by using the overhead vent valve for temperature control. There is no reflux drum, and feed comes in close to the top, providing the internal reflux stream.
- The column pressure controller cascades to the steam flow SP on the reboiler.

One reason for the unconventional PID loop pairing is that controlling pressure with the reboiler duty ensures that the pressure is less likely to go high and lift the safety valve. If the PID was paired the other way around, there is a chance for the pressure to go high if the overhead valve saturates before the column runs out of reboiler duty. Since the degree of interaction between the two loops is quite strong, the pressure loop will go unstable if either the feed drum LIC or the TIC approached unstable (even though there is nothing wrong with the tuning of the PIC). The drum level controller sets the valve position directly without the benefit of a cascaded flow controller. Any change in either upstream or downstream pressure affect the feed flow PV, which then affects the temperature and pressure in the E20 column. Due to the heavy coupling between the three loops, it is not really feasible to tune one loop without due consideration of the remaining two control loops.

Best results were obtained by step testing condensate flow setpoint and vent valve, fitting a 2x2 model and then calculating moderately fast but well damped tuning that takes the strong off-diagonal interaction into account. We tuned it for fast SP tracking but high robustness margins. The standard deviations of the two PVs are now substantially better than before and that both loops are tracking SP well.

The gain for the tray temperature vs. vent valve varies very substantially depending on operating point (from almost zero when the top of the column is perfectly pure, to almost infinite when the column goes rapidly off-spec). This value represents the best model we could identify by keeping the tray temperature in the correct range with the column pressure at the nominal operating point. The dynamic model is shown in Fig. 4.

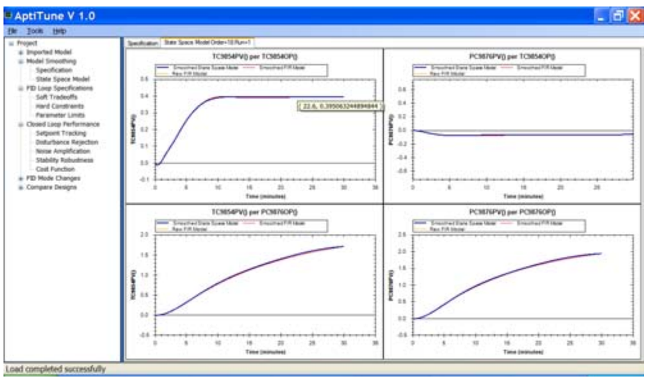


Fig. 4: Step responses of the dynamic model
(Legend: TC.PV=Tray 6 Temperature; PC=Column Pressure; TC.OP=Vent Valve; PC.OP=Steam Flow SP)

Note that the total state-space model order is 18. Fig. 5 shows the optimized PV responses for a step change in both SPs:

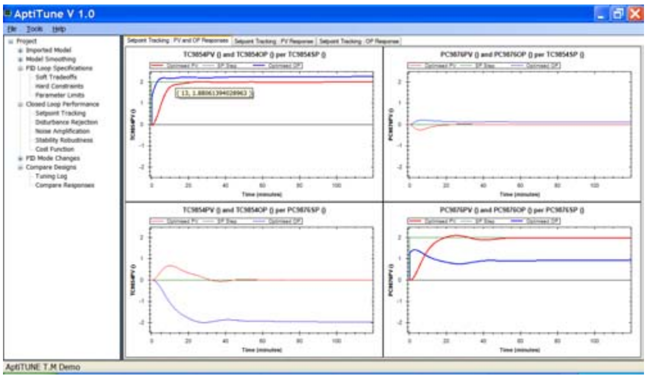


Fig. 5: Response of the 2x2 system to SP step changes
(Legend: PV is shown in red, OP in blue)

Note that PV overshoot is very low and that damping is exceptionally good. The OP value for the vent has a peak value that is almost the same as the steady state value, and the loop has about the same rise time in closed loop as compared to open loop (a speed-up factor of about 1x). The pressure loop is about 2x faster in closed loop compared to open loop. The load disturbance response is shown in Fig. 6:

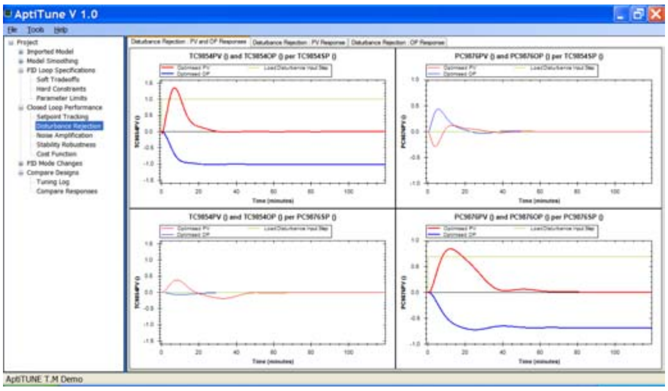


Fig. 6: Response of the 2x2 system to load disturbances

Note that the PV and OP responses have very good damping with peak OP values that are very similar to the steady state values. This will ensure exceptionally good damping on the actual process unit even when the process gain varies significantly. Gain and dead time (stability) margins are very good, see Fig. 7:

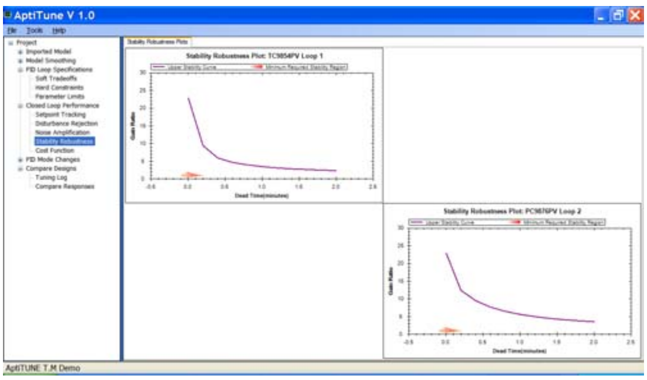


Fig. 7: Robustness plot (gain and deadtime stability margins)

A sensitivity analysis on the tray temperature loop shows that the gain of the process will have to increase by 3x AND the dead time will have to increase by another 12 seconds before the damping of the loops is unacceptable. Instability sets in at a process gain increase of more than 20x. A dead time error of more than 2 minutes is needed to reach instability, and there is no process mechanism for this to occur while the TIC is in the active range. These margins are exceptionally safe.

In order to compare the performance of the loops before and after retuning, we collected a week of normal operating data before we arrived on site, and one week of normal operating data after the re-tuning work was concluded. From these large data sets, we then calculated histograms to show the distribution of the control error (SP-PV). For process reasons, we wanted the loops to be robust and to be able to withstand changes in process dynamics. As a result, some loops were intentionally slowed down, and of course, their probability distributions will be wider than before. However, this compromise is all for a good cause as it will ensure that the loops remain operational for the years to come.

The performance of the pressure control loop PCA9876 is compared in Fig. 8:

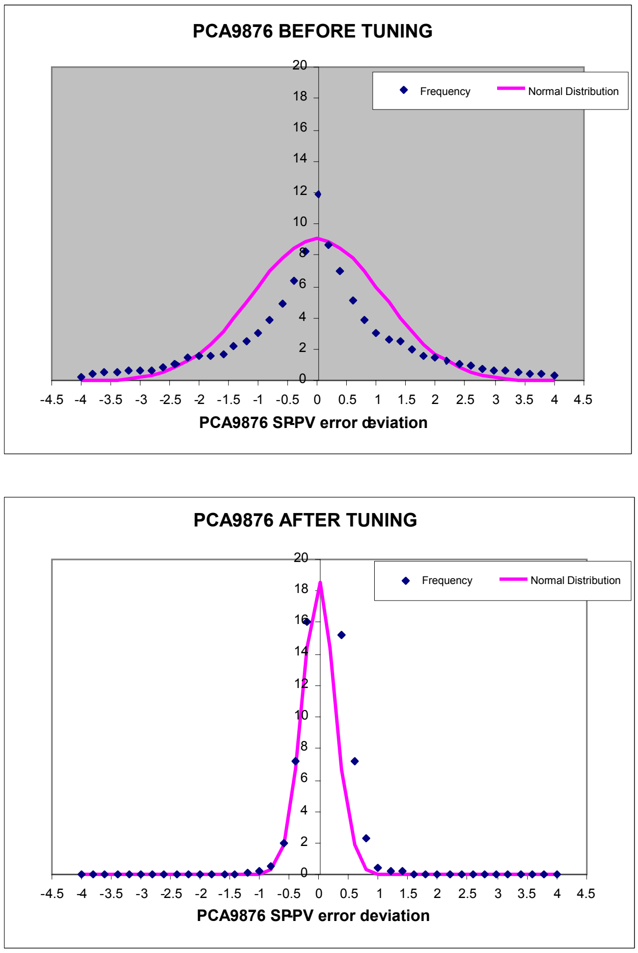
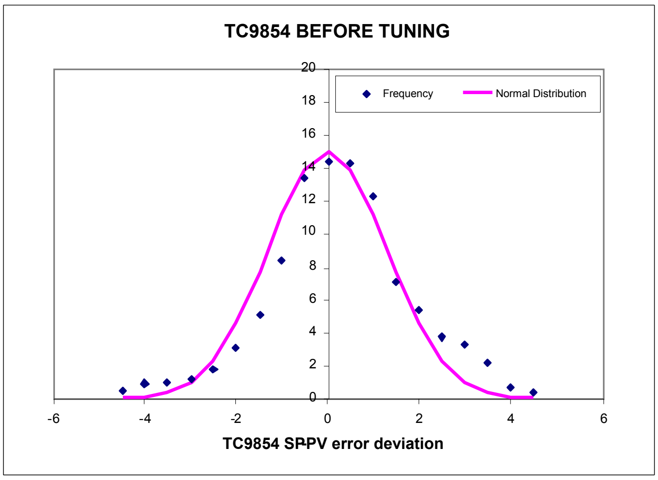


Fig. 8: Control error histograms for PCA9876 before and after retuning

It is clear from the two histograms shown above that the variability in the PV has reduced by about a factor of 4x. This is a big improvement in performance, yet this could be accomplished without compromising the robustness characteristics of the loop. The pink trace shows the best fit for a normal (Gaussian) distribution to the data, assuming a zero mean value. The estimated standard deviation σ reduced from 1.1 to 0.28.

The histograms for temperature loop TC9854 are compared in Fig. 9:



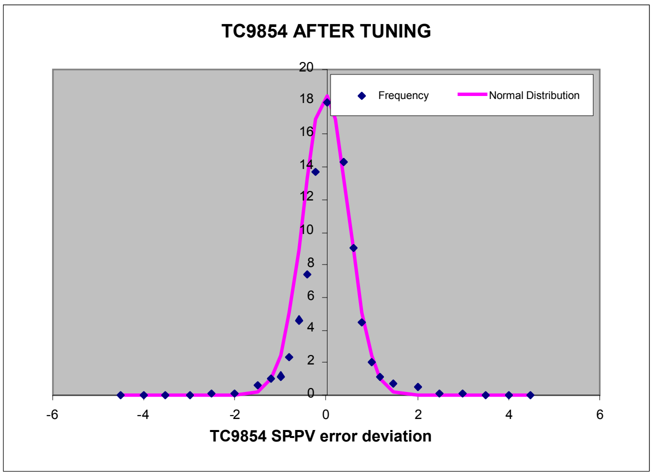


Fig. 9: Control error histograms for TC9854 before and after retuning

The standard deviation reduced from 1.3 down to 0.5, so by a factor of almost 3x. To be honest, these good results are partly due to the very slow initial tuning of some loops we found at the beginning of the project.

4. CONCLUSIONS

The following conclusions can be made. A MIMO model-based approach can be used to successfully tune multiple PID loops that interact strongly. If the open loop model is moderately accurate, then "one-shot" tuning is achievable and the simulated and observed OP and PV responses will be almost identical. The use of sufficiently large gain and dead time robustness margins ensures that the loop will remain stable and well damped even if the process is strongly nonlinear. This also helps to protect us against inaccurate model identification results. The ability to impose hard constraints on damping ratio, maximum PV overshoot, and the maximum OP value means we can ensure that the final design is safe from a process point of view. PID tuning rules cannot achieve this.

REFERENCES

Chen, D., Seborg, D.E. (2003). Design of decentralized PI control systems based on Nyquist stability analysis. *Journal of Process Control* 13(1), 27 – 39.

Espinosa Oviedo, J.J., Boelen, T., van Overschee, P. (2006). Robust advanced PID control. *IEEE Control Systems Magazine* 26(1), 15 – 19.

Forssell, U., Ljung, L. (2000). A projection method for closed-loop identification. *IEEE Trans. On Automatic Control* 45(11) 2101 – 2106.

Halevi, Y., Palmor, Z.J., Efrati, T. (1997). Automated tuning of decentralized PID controllers for MIMO processes. *Journal of Process Control* 7(2), 119 – 128.

Hovd, M., Skogestad, S. (1993). Improved independent design of robust decentralized controllers. *Journal of Process Control* 3(1), 43 – 51.

Hovd, M., Skogestad, S. (1994). Sequential design of decentralized controllers. *Automatica* 30(10), 1601 – 1607.

Li, Y., Ang, K.H., Chong, G. (2006). Patents, software and hardware for PID control. *IEEE Control Systems Magazine* 26(1), 42 – 54.

Maciejowski, J.M. (1989). *Multivariable Feedback Design*. Addison-Wesley

Monica, T.J., Yu, C.-C., Luyben, W.L. (1988). Improved multiloop single-input, single-output (SISO) controllers for multivariable processes. *Ind. Eng. Chem. Res.* 27, 969 – 973.

Vlachos, C., Williams, D., Gomm, J.B. (2000). Genetic approach to decentralized PI controller tuning for multivariable processes. *IEE Proceedings. Control Theory and Applications* 146(1), 58 – 64.

Zhu, Y. (2004). Robust PID tuning using closed-loop identification. Preprints of the International Symposium on Advanced Control of Chemical Processes ADCHEM 2004. vol. 1, 165 – 170