An Expert System to Perform On-Line Controller Tuning

Jonathan Litt

The expert system described in this article tunes a proportional-integral-derivative (PID) controller on-line for a single-input-single-output multiple-lag process with dead time. The expert system examines features of each transient response and the corresponding controller parameters. It determines a new set of controller gains to obtain a more desirable time response. This technique can be used to determine and implement a different set of PID gains for each operating regime and, once in steady state, the system can be used to find optimal parameters for load disturbance rejection. The expert system can be applied to any system of the specified form (aerospace, industrial, etc.) and can be expanded to include additional process models.

Introduction

Proportional-integral-derivative (PID) controllers (sometimes called three-mode or three-term controllers) are quite popular. The PID controller or controllers made up of a subset of the terms (P, PI, PD) are used in most industrial applications [1] and in some multivariable (modern) applications, such as [2]. The PID controller has no standard form and it can be represented in several ways [3]. A form common in industrial applications is used here, namely

$$G_C(s) = K_C[1/(T_I s)+1+T_D s].$$

In this form, K_C represents the proportional term, T_I is the reset time (reciprocal of the integral), and T_D is the rate time or derivative term.

In many circumstances, the ability to tune a controller to meet closed loop time domain specifications is considered more of an art than a science. In cases where the controlled system is hard to model accurately or where it is unmodeled, as in many industrial situations, the tuning might be done by someone with an intuitive feel for the process and not necessarily a theoretical understanding of it. It is common for each plant operator to have his own preference for the best type of transient response and operators will often change each other's controller settings from shift to shift. In such situations, one speaks of the optimal response as the transient which *looks* the best to the operator. This optimality is not defined in any mathematical sense. On the other hand, some mathematically optimal PID parameters produce transients which are unacceptable from a practical point of view [4]. The operators consider such features as period and overshoot to be the determining factors in an acceptable response. This is strictly a time domain approach. In industrial applications, the dissatisfaction with controller performance is so high among plant operators that over 50% of process control loops are run in manual (open loop) mode [5]. In modern control applications, many of which are optimal designs using frequency domain techniques, tuning is still a concern even though it is often described as an iterative design approach (for example, see [6] or [7]). There are still cases, however, where the classical terminology is applied to modern problems, as in [8] or [9].

The original algorithmic time domain procedure for the adjustment of PID controllers regulating industrial processes was presented by Ziegler and Nichols [10]. Using the fact that a process with dead time under the appropriate proportional-only control, K_U, will respond with a sustained oscillation of period P_U to a step change in input [11], they determined a set of general formulas for controller gains which give what they considered optimal step responses in the majority of industrial processes-transients with the overshoot of the second peak one quarter that of the first. The Ziegler-Nichols controller parameters are determined using the equations, $K_C = 0.6K_U$, $T_I = P_U/2$, $T_D = P_U/8$. The step response using these parameters settles rapidly but overshoot may be unacceptably high.

Descriptions of several expert systems which perform PID controller tuning have appeared in the literature. Probably the most well known of these is summarized in [12] where the PID parameters are adjusted to give a faster response as long as the maximum allowable values for overshoot and damping are not exceeded. The expert system described in [13] tries to achieve the minimum integrated error response by identifying system parameters, then computing and implementing the optimal gains based on them. The system parameters are recalculated during each transient and the controller gains are updated. This procedure is repeated until convergence. In [14]-[16], the expert systems are used to choose between different numerical algorithms for control or controller tuning, based on the operator's objectives. The expert system reported on in [17] uses meta-rules to identify the class of open loop system to which the controlled plant belongs. Then it applies PI tuning rules appropriate for that type of system to meet specifications on maximum overshoot, maximum undershoot, and damping ratio. In [18], a fuzzy logic PID controller is described which continuously adjusts the controller parameters about their nominal values to improve every portion of the response.

The work described in the current paper was inspired by the tuning maps of Doris Wills [19]. Wills' original work addressed the tuning of a PID controller for a process described by the following with $\tau_1 = \tau_2 = 10T$:

$$H(s)=\exp(-Ts)(\tau_1 s+1)^{-1}(\tau_2 s+1)^{-1}. \qquad (1)$$

A desired transient response was chosen for each map based on such features as Ziegler-Nichols-type response [10], critical damping, or minimum Integral Time Absolute Error (ITAE) [20]. For given values of T_I and T_D, the transient of the desired form along with the value of K_C which achieved it were plotted in the appropriate region of the T_D/P_U-P_U/T_I plane. The normalization of the axes by the ultimate period, P_U, made them dimensionless and therefore general to a class of systems of the type in (1). Given a plant which may be approximated by (1), and its experimentally-

Presented at the 1990 American Control Conference, San Diego, CA, May 22-25, 1990. The author is with the Propulsion Directorate, U.S. Army Aviation Systems Command, Lewis Research Center, Cleveland, OH 44135.

0272-1708/91/0400-0018$01.00© 1991 IEEE

determined ultimate period, the PID parameters for the desired response can be selected directly from the appropriate map.

The Expert System

In this work the tuning map idea was extended from specified response curves plotted in the T_D/P_U-P_U/T_I plane to response surfaces plotted in the T_D/P_U-P_U/T_I-K_C space (Fig. 1). Each set of transients in the figure consists of ten responses with T_I and T_D fixed while K_C varies from 1.0 to 19.0 in increments of 2.0. The transient sets are displayed obliquely with K_C increasing from top to bottom. The horizontal axis for each set is marked in intervals of P_U s. The sets of responses are positioned appropriately in the T_D/P_U-P_U/T_I plane, presented on a log-log scale.

Tuning rules were extracted directly from this three-dimensional map. These rules, which are used with a forward-chaining inference engine, tune the PID controller parameters based on the important features of the response. The expert system looks at the following features: 1) percent overshoot, 2) ratio of the overshoot of the second peak to that of the first, 3) period of oscillation, 4) rise time, and 5) damping. The features' names are self-explanatory except for damping which is defined as

$$\text{damping} = (P_2\text{-}V_1)/(P_1\text{-}V_1)$$

where P_1, P_2, and V_2 are, respectively, the values of the first and second peaks and the first valley of the step response. The damping is larger if the response is highly oscillatory and smaller if the response settles quickly. The process model used here is of the form in (1) with $\tau_1 = \tau_2 = 10$ and $T = 1$.

Looking at the map in Fig. 1, one can infer general rules about how the changes in the controller parameters affect the features of the response. The observations are as follows:

1) Increasing K_C decreases period and vice versa.

2) Increasing K_C increases overshoot and vice versa.

3) Increasing K_C decreases rise time and vice versa.

4) Increasing K_C increases damping and vice versa.

5) Decreasing T_I increases overshoot ratio and vice versa.

6) Decreasing T_I increases damping and vice versa.

7) Decreasing T_I decreases stability and vice versa.

8) Increasing T_I decreases overshoot and vice versa.

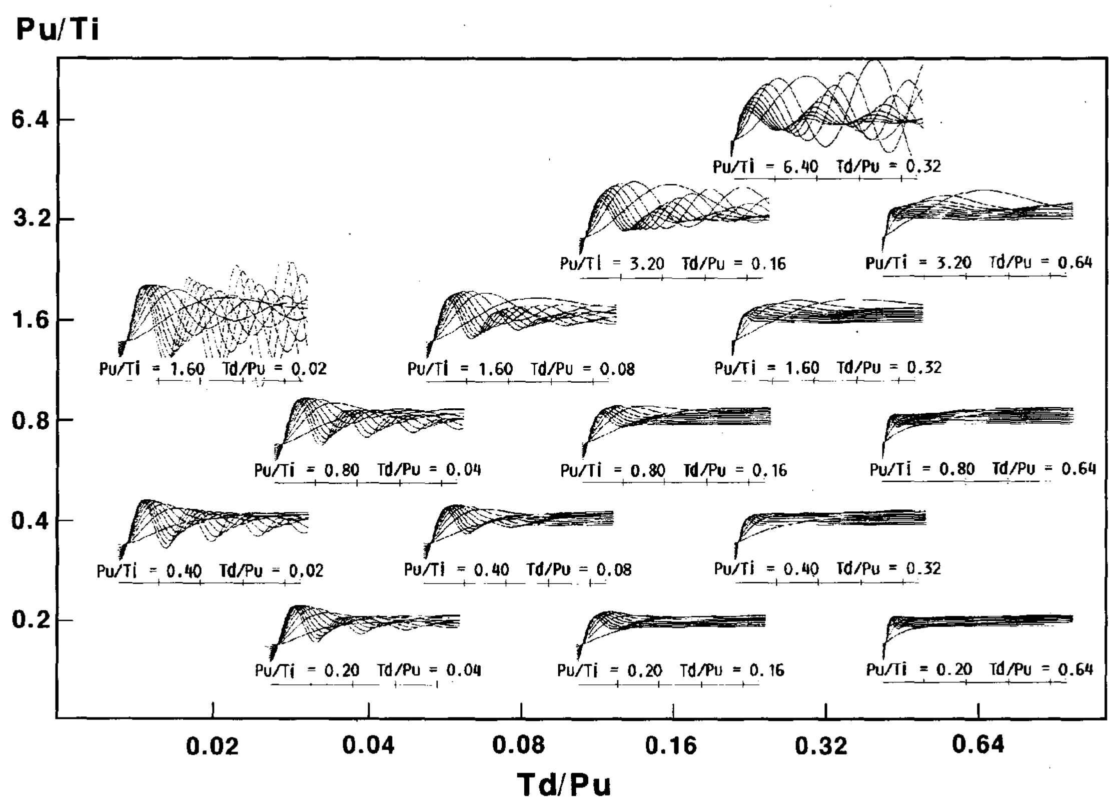

Fig. 1. Three-dimensional tuning map from which general tuning rules were extracted.

9) Increasing T_D increases stability and vice versa.

10) Increasing T_D decreases rise time and vice versa.

The inputs to the expert system are 1), a desired numeric value for each of the features (for instance, desired overshoot is 10%, desired damping is 0.5, etc.), 2) a numeric weight for each feature indicating relative importance to the user, 3) maximum acceptable overshoot, and 4) initial PID values. Each transient is given a numeric score which is the sum of the normalized error in each feature multiplied by the feature's respective weight as introduced in [21]. The normalized error E is computed as the absolute value of the difference between the actual feature value and the desired feature value divided by the desired feature value. Thus, the error in each feature is scaled so an error of ±100% has a value of 1.0 independent of units.

As the transient occurs, an on-line pattern extractor monitors it and determines numerical values for each feature based on observation. Easily identified landmarks of the response such as peaks and valleys are combined to produce the more complicated patterns describing the response. Noise filtering can be used to smooth the response if the features become obscured.

After the transient settles, the expert system analyzes the numerical values representing the patterns which describe the response. Based on the errors in the features, the expert system determines which way the tuning parameters must be altered. Since there will almost always be a conflict with respect to the direction of parameter adjustment when tuning to attain several desired features at once, the PID terms are modified in the direction which will have the most impact. To determine which direction this is, the score for each feature is multiplied by +1 or -1 to indicate an increase or decrease in the value of the particular tuning term. The signed scores associated with each controller term are then added and the sign of the sum determines the direction of change for that term. In the situations where more than one tuning parameter affects a feature, the feature's entire weighted error is assigned to the term which, in decreasing order of importance, will 1) reduce overshoot if overshoot is close to or above the maximum allowed, 2) have the smallest impact on other features, and 3) improve stability.

Once the direction of adjustment for each term is inferred, the change is implemented. A fibonacci sequence is used to determine the amount of change. A fibonacci sequence is an ordered set of numbers defined recursively as $f(n) = f(n-1) + f(n-2)$ for $n \geq 3$. An example is $\{f(n)\} = \{1, 1, 2, 3, 5, 8, 13, ...\}$. The amount of change in each controller parameter is the reciprocal of $f(n)$ times the value of the term times the sign of the sum of weighted errors associated with that controller parameter. The general equation for updating a term is

$$\text{term}(n) = \text{term}(n-1) \cdot \left[1 + f(n)^{-1} \operatorname{sgn}\left(\sum_{i=1}^{R} s_i w_i E_i \right) \right] \quad (2)$$

$$\text{where } \operatorname{sgn}(x) = \begin{cases} -1, & x < 0 \\ 0, & x = 0 \\ 1, & x > 0 \end{cases}.$$

R is the number of features assigned to the particular term, s_i is the sign of the weighted error for the ith feature, determined by the expert system, indicating whether to increase or decrease the term, w_i is the weight assigned by the user to the ith feature, and E_i is the normalized error of the feature. Each of the three terms uses its own fibonacci sequence. Thus, since all three tuning parameters might not be updated on each iteration, the value of n in can be different for each term. This technique of adjusting the controller gains will never allow the values to become negative and will cause the values to converge after a finite number of trials.

The feature values, weighted errors, total score, fibonacci numbers, and PID parameters for each response are stored in a frame [22]. A frame is a data structure which consists of slots, facets, and values. Each frame represents one transient and each of the slots it contains represents an important piece of information about the transient, for example, a feature. Each slot has associated facets which contain the name of a specific piece of information describing the slot and a corresponding value. Fig. 2 shows a frame with all of the slots, facets, and values filled.

	TRANSIENT #0001	
percent overshoot	value	56.94
	weighted error	140.82
damping	value	0.51
	weighted error	0.55
rise time	value	2.86
	weighted error	0.43
period	value	17.99
	weighted error	0.80
overshoot ratio	value	0.26
	weighted error	0.71
total score	value	143.30
K_C	value	15.0
	Fibonacci term	2
T_I	value	6.75
	Fibonacci term	2
T_D	value	1.69
	Fibonacci term	2

Fig. 2. Example of a frame used to store transient data.

The transient which receives the lowest score is considered the best by the expert system. This information is not used until the controller terms have converged, however. Since the expert system is emulating a human tuner, it bases its adjustments on differences between the most recent response and the desired response. The justification for this strategy is that the rules contain the information needed to achieve a minimum score and they will find it after enough iterations. Thus, it is possible that tuning may actually cause a higher score from one transient to the next by making too large a change and overshooting the optimal set of parameters in the (K_C, T_I, T_D) space, for instance. Since the step size is successively reduced, the rules should cause the score to approach a minimum in the long run.

If the change in the value of each PID gain is small compared to the respective gain's total value and the score is relatively constant, the expert system will invoke special tuning rules designed to move the parameters out of the region of convergence where the score has attained a local minimum. Several of these rules are implemented. Each one covers a specific situation where a coarse adjustment of the parameters might be required, and each resets the fibonacci sequences to the beginning. These rules are necessarily conservative to keep the closed loop system stable. If none of the special rules applies, the parameters are set to those of the lowest-scoring transient, the fibonacci sequences are incremented from those of the lowest-scoring transient to reduce the step size, and the tuning process continues. When the convergence of the parameters coincides with the advent of the lowest scoring response and no local minimum rule applies, the tuning ceases. The final step response is truly optimal as it is close to the operator's specifications from a visual point of view while, at the same time, it minimizes the objective function.

Example

The open-loop system used in the example was of the type in (1) with $\tau_1 = \tau_2 = 10.0$ s and the dead time approximated by a first order lag with a time constant of 1.0 s. Euler integration with a time step of 0.01 s was used to evolve the system through time. These two approximations (a lag to represent dead time and numerical integration) introduce small differences between the simulation and the ideal system in (1) which help verify the robustness of the rules. In the example, a Ziegler-Nichols-type response was desired. Thus the desired feature values and their weights were specified as in Table I with a maximum overshoot value of 45% allowed.

Table I
Desired Feature Values and User-Specified Weights

FEATURE	DESIRED VALUE	WEIGHT
percent overshoot	10	30.0
overshoot height ratio	0.25	20.0
damping	0.5	20.0
period	10.0 s	1.0
rise time	2.0 s	1.0

It can be seen from the weights asociated with rise time and period that these features were not considered as significant as the others in the quality of the response while the overshoot value was considered very important.

The initial controller values were chosen as the Ziegler-Nichols parameters of $K_C = 15.0$, $T_I = 6.75$, and $T_D = 1.69$, calculated from $P_U = 13.5$ and $K_U = 25$. These values gave a response of the desired shape but with unacceptably large overshoot. Fig. 3 shows plots of every fifth response beginning with the first, originating at the left center, and finishing with the 111th, originating in the center bottom. These selected transients are representative of those obtained during the entire tuning process.

Fig. 4 displays the score and each feature value plotted versus transient number. The responses coinciding with the large jumps in score and parameter values usually indicate where a local minimum rule was applied. After 108 transients, the response curves converged to almost exactly the desired form. Only damping was significantly different from the user-specified value-0.26 rather than 0.50.

Conclusions

The expert PID tuner is able to emulate a human PID tuner using a strictly classical,

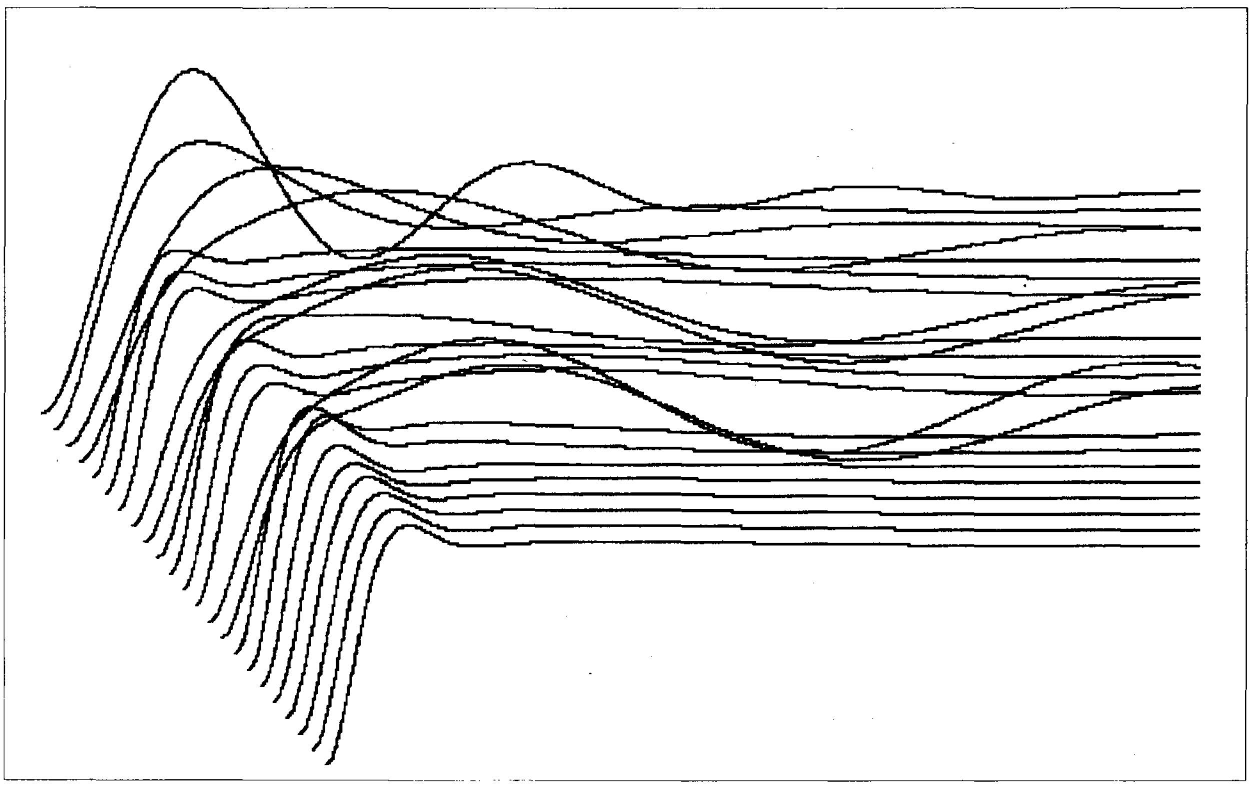

Fig. 3. Selected transient responses from the controller tuning procedure. Transient number increases from upper left to lower right.

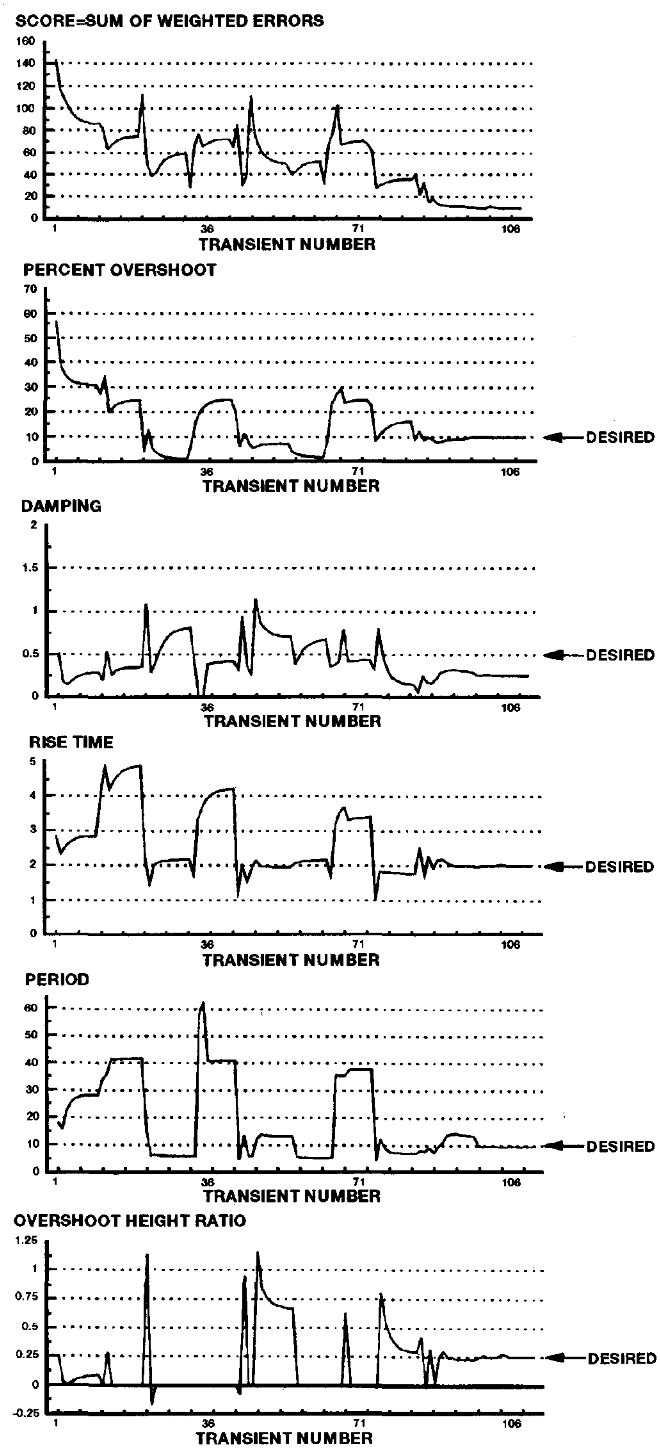

Fig. 4. Transient response score and feature values versus transient number.

time domain approach. It works well for the examples tried, including the example presented here, although it may take many transients to converge. The rules are purposely vague enough that they do not fully model every nuance of the tuning map. This allows the rules to work reasonably well with systems similar to but not exactly of the form used to develop the map. In this way the expert system has wider applicability and is thus more useful.

In this heuristic approach to tuning, there is no guarantee that the parameters will converge such that the score is a global minimum. The tuning can be only as good as the rules and knowledge about the system being controlled. The expert system would benefit from the inclusion of additional of rules for taking the parameters out of the area of a local minimum if it suspected that one was found.

The objective function is formulated in such a way that the operator's preferred response shape is approximated as the parameters are optimized. This combines the pragmatic approach of the human tuner with the mathematical foundation of a cost measure to provide an optimal response curve in a practical as well as numerical sense.

Future Work

Fuzzy algorithms [23] are not used here but controller tuning is an excellent application for them. The convergence of the tuning parameters could be sped up significantly if rules were included which determine how much to adjust the value of a term based on the magnitude of the error in the feature.

An on-line transient stability check such as in [24] or [25] would also be an important addition. If the expert tuner were to be implemented on a real process, even though the tuning is done so as to improve stability whenever possible, an on-line stability monitor would be a necessary safety feature.

One reason that plant operators find dissatisfaction with closed-loop control is that on their highly nonlinear processes, a set of PID gains, optimal in one operating range, produces an unacceptable response in another. This expert system can be easily extended to find different sets of PID gains for different operating regimes. The only change required is that all transient data must be grouped according to operating region so that there will be many small groups of frames rather than one large one. As long as the rules, which are general to begin with, are fairly accurate over the whole operating range, the only time the grouping would play a role in the tuning is when the lowest-scoring transient for a particular region is required. This is essentially the idea behind gain scheduling in many modern applications.

Once a process is in steady state, closed loop control is used to provide disturbance rejection. Tuning maps have been developed for this application [26]. The procedure used to determine the rules for this expert system could be easily utilized to find rules for tuning PID controllers to handle loads.

The normalization of the error, although intended as an equalizing device, can cause problems in certain cases. If a 1% overshoot is requested and 2% is attained, the component of the score due to the error is the same as in the case where a 20% overshoot is requested and a 40% overshoot results. It is unlikely that a 100% error in the first case would be considered significant. This sensitivity issue should be addressed.

The reciprocated fibonacci sequence which was used to adjust the controller parameters was chosen arbitrarily because it is a convenient, convergent sequence. It is not the only sequence which could have been used. One problem with using a sequence of this kind, however, is that it causes the PID parameters to converge after a certain number of iterations rather than at a point of minimum score for the transient. Often the two correspond but sometimes they do not. The results might be improved if the step size were decreased based on the change in the score, as in a gradient algorithm.

References

[1] J. Martin, A. B. Corripio, C. L. Smith, "How to select controller modes and tuning parameters from simple process models," *ISA Trans.*, Vol. 15, pp. 314-319, 1976.

[2] A. H. Jones and B. Porter, "Design of adaptive digital set-point tracking PI controllers incorporating recursive step-response matrix identifiers for gas turbines," in *Proc. AIAA Guidance, Navigation and Control Conf.*, Williamsburg, VA, Aug. 18-20, 1986, pap. 86-2156.

[3] J. P. Gerry, "A comparison of PID control algorithms," *Control Eng.*, pp. 102-105, Mar. 1987.

[4] M. Fu, A. W. Olbrot, and M. P. Polis, "Comments on "Optimal gain for proportional-integral-derivative feedback," *IEEE Control Syst. Mag.*, Vol. 9, pp. 100-101, Jan. 1989.

[5] N. Andreiev, "A new dimension: A self-tuning controller that continually optimizes PID constants," *Control Eng.*, pp. 84-85, Aug. 1981.

[6] R. S. McEwen and D. P. Looze, "Quadratic weight adjustment for the enhancement of feedback properties," in *Proc. 1982 Amer. Control Conf.*, Arlington, VA, June 14-16, 1982.

[7] C. A. Harvey and G. Stein, "Quadratic weights for asymptotic regulator properties," *IEEE Trans. Auto. Control*, Vol. 23, pp. 378-387, June 1978.

[8] T. T. Jussila and H. N. Koivo, "Tuning of multivariable PI-controllers for unknown delay-differential systems," *IEEE Trans. Auto. Control*, Vol. 32, pp. 364-368, Apr. 1987.

[9] H. N. Koivo and S. Pohjolainen, "Tuning of multivariable PI-controllers for unknown systems with input delay," *Automatica*, Vol. 21, pp. 81-91, Jan 1985.

[10] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Trans. A.S.M.E.*, Vol. 64, pp. 759-768, 1942.

[11] F. G. Shinskey, *Process Control Systems.* New York: McGraw-Hill, 1988.

[12] T. W. Kraus and T. J. Myron, "Self-tuning PID controller uses pattern recognition approach," *Control Eng.*, pp. 106-111, June 1984.

[13] R. Devanathan *et al.*, "An expert PID controller," in *Proc. IEEE Int. Workshop on Artificial Intelligence for Industrial Applications*, 1988.

[14] K. J. Åström, "Auto-tuning, adaptation and expert control," in *Proc. 1985 Amer. Control Conf.*, Boston, MA, June 19-21, 1985.

[15] K-E Årzén, "Use of expert systems in closed loop feedback control," in *Proc. 1986 Amer. Control Conf.*, Seattle, WA, June 18-20, 1986.

[16] J. Lieslehto, J. T. Tanttu, and H. N. Koivo, "An expert system for tuning PID controllers," in *Proc. IFAC Computer Aided Design in Control Systems*, Beijing, PRC, 1988.

[17] B. Porter, A. H. Jones, and C. B. McKeown, "Real-time expert tuners for PI controllers," *IEE Proc. D*, Vol. 134, pp. 260-263, July 1987.

[18] S. Tzafestas and N. P. Papanikolopoulos, "Incremental fuzzy expert PID control," *IEEE Trans. Indus. Electron.*, Vol. 37, pp. 365-371, Oct. 1990.

[19] D. M. Wills, "Tuning maps for three-mode controllers," *Control Eng.*, pp. 104-108, Apr. 1962.

[20] A. M. Lopez, J. A. Miller, C. L. Smith, and P. W. Murrill, "Tuning controllers with error-integral criteria," *Instrumentation Technol.*, pp. 57-62, Nov. 1967.

[21] J. Litt, "An expert system for adaptive PID tuning based on pattern recognition techniques," in *Proc. 1986 ISA North Coast Conf.*, Cleveland, OH, May 19-22, 1986.

[22] P. H. Winston and B. K. P. Horn, *LISP*. Reading, MA: Addison-Wesley, 1984.

[23] L. A. Zadeh, "Fuzzy algorithms," *Inform. Control*, Vol. 12, pp. 94-102, Feb. 1968.

[24] L. L. Scalzott and C. F. Lorenzo, "Practical stability criterion and its application to digital simulation," NASA Tech. Note D-4203, Dec. 1967.

[25] J. M. G. Sá da Costa and N. Munro, "On-line transient stability analysis in large power systems by pattern recognition techniques," in *Proc. IEE Control 85*, Cambridge, U. K., July 9-11, 1985.

[26] D. M. Wills, "A guide to controller tuning," *Control Eng.*, pp. 93-95, Aug. 1962.

Jonathan Litt received the B.S. and M.S. degrees in systems and control engineering from Case Western Reserve University, Cleveland, OH, in 1983 in 1985, respectively. Since 1985 he has been an Aerospace Engineer with the Propulsion Directorate, U.S. Army Aviation Systems Command, NASA Lewis Research Center, Cleveland, OH. His research interests are in the area of artificial intelligence applied to controller reconfiguration and controller tuning as well as linear system theory and multivariable controller tuning.

Axelby Outstanding Paper Award

Every year, the Control Systems Society awards up to three outstanding paper awards to authors of papers published in *IEEE Transactions on Automatic Control* during the two preceding calendar years. The award is named after George S. Axelby, founding editor of the *Transactions.* We are now soliciting nominations for this year's award, from papers published in the *IEEE Transactions on Automatic Control* from January 1989 through December 1990 (volumes 34 and 35). This is an open invitation to nominate papers for consideration. The deadline for nominations is May 15, 1991. Nominations should be sent to: Prof. Michael Sain, Axelby Award Chairman, Dept. of Electrical and Computer Engineering, University of Notre Dame, Notre Dame, IN 46556; phone 219-239-6538.