

Simulation Studies of the SI PID Control Autotuning

David C. Sheats

Scientific Instruments
Inc. West Palm Beach,
FL 33407

dsheats@scientificinstru-
ments.com

Zvi S. Roth

Florida Atlantic
University, Boca Raton,
FL 33431

rothz@fau.edu

Jon K. Burford

Florida Atlantic
University, Boca
Raton, FL 33431

jon@caperion.net

Joseph W. Snyder

Scientific Instruments
Inc. West Palm Beach,
FL 33407

jsnyder@scientificinstru-
ments.com

ABSTRACT

The Scientific Instruments, Inc. (SI) feedback control PID tuning is based on a sequence of P control closed loop step response tests utilizing the measured peak overshoot values and the time between the first peak and first dip of the step response. This autotuning technique, developed empirically at Scientific Instruments, Inc. for temperature control of systems operating in the cryogenic regime (i.e. - below 100K), and its associated set of experiment results were described in detail in an earlier paper [1].

This paper focuses on simulation studies designed to assess more deeply the performance of the SI method, explore its validity for a large range of system parameters and configurations.

The main finding of these studies is that this novel PID control tuning method appears to be limited in scope to processes characterized by one slow and exactly two fast time constants. The algorithm fails to produce satisfactory results if the number of short-duration modes exceeds two. The simulations explore corrective means to extend the SI PID autotuning method to a larger set of process control scenarios.

Keywords

PID, Temperature control, Autotuning, Cryogenic, SI method.

1. MODEL ASSUMPTIONS

A typical application of temperature control in a cryogenic environment would be that of a heater (a strip heater wrapped around a metal block or a cartridge heater inserted into the block) and a temperature sensor mounted on the block. The block is cooled at all times through contact with either a cryogenic gas or a refrigeration stage. The control objective is to bring the block to thermal equilibrium at a user-specified temperature and maintain it at that level.

The process that relates the block temperature (in Kelvin) to the heater power (in Watts) is typically modeled, for sufficiently small set-point changes, and without counting the controller's dynamics, as a third-order linear system. This model must take into account the heat capacity of the heater, block and sensor, and the thermal resistances of the heater-block connection,

heater-sensor connection and heater-cryogenic bath path. These capacities and resistances translate into three time constants in an all-pole transfer function [2]. Typically, one of the three process time constants is large (i.e. in the order of magnitude of tens or hundreds of seconds), whereas the other time constants are fast (i.e. in the order of magnitude of a few seconds to a fraction of a second). Indeed, the block's time constant is often orders of magnitude larger than the time constants of the heater and the thermometer.

In this paper we shall adopt the simplest analog form of the PID control algorithm:

$$U(s) = P\left(1 + \frac{1}{Is} + Ds\right)E(s) \quad (1.1)$$

where $U(s)$ is the Laplace Transform of the heater power $u(t)$ (measured in [W]) and $E(s)$ is the Laplace Transform of the error signal $e(t)$. The error $e(t)$ is the difference between the set-point temperature T_{ref} and the temperature measured by the sensor $T(t)$. The control constants P , I and D are the proportional, integral and derivative control gains.

Cryogenic environment heating (or cooling) is typically characterized by the following nonlinear phenomena:

a) Temperature dependent specific heat of the block - If we represent the small-signal reduced process open-loop transfer function, after removing all fast time constants, as

$$\frac{T(s)}{U(s)} \cong \frac{K}{1 + s\tau}$$

then both the DC gain K and the time constant τ are functions of the temperature. Typically, both increase with the temperature. Assuming that all fast time constants contribute an aggregate pure time delay L to the model:

$$\frac{T(s)}{U(s)} \cong \frac{K}{1 + s\tau} e^{-sL} \quad (1.2)$$

then L is also a function of T , typically an increasing function. Experimental curves of $K(T)$, $\tau(T)$ and $L(T)$ for the SI M9700 process were presented in [1].

b) Amplifier saturation - The heater's power $u(t)$ cannot exceed a given maximum power U_m . Furthermore, heater power is always non-negative. In the closed-loop system this means that if the temperature overshoots beyond the commanded temperature the cooling down is done in open-loop by the cryogenic surroundings. There is no active

control action in the negative direction. The following equation summarizes these actuator constraints:

$$0 \leq u(t) \leq U_m$$

c) Temperature-dependent Holding Power - If the amount of set-point change r is positive (r being equal to the desired temperature minus the initial temperature) the controller applies a heating signal from $t=0$. If $r < 0$, there is initially a zero control signal, but near steady-state a nonzero control signal resumes. If the PI (or PID) controller keeps the closed-loop system stable at a fixed temperature of $T_\infty = T_0 + r$ then there exists a steady-state "holding" heater power u_∞ that is an increasing function of T .

An issuing of a new set-point implies the resetting of the clock to $t=0$. The "present" temperature becomes the "initial" temperature. The present integral control memory (i.e. the summation of all error terms from $t=0$ to the present time t) is always the beginning "holding power." This term is only reset to zero if the control action is terminated and then restarted.

PID control references listed in [1] and elsewhere are geared towards systems such as described by (1.2). Fast time constants are normally part of the process' "un-modeled dynamics" making it convenient for control researchers to work with an aggregated pure-time delay L . The very essence of PID process control is a "black box" view of the process: "A feedback control is found based on a single or a few tests, without knowing the detailed model parameters and without attempting to identify the process' mathematical model." As will be pointed in Section IX the simulation studies for the SI PID control method require however an alternate equation to (1.2) in which the fast time constants are explicitly taken into account. Let τ_1 denote the slow time constant of the open-loop process, and let $\tau_{f1}, \tau_{f2}, \dots, \tau_{fN}$, denote N fast time constants. Without loss of generality (in the simulations) we shall replace all individual fast time constants with a single multiple pole of multiplicity N located at $s = -1/\tau_f$. N is an integer greater or equal to 2.

$$\begin{aligned} \frac{T(s)}{U(s)} &\cong \frac{K}{(1 + s\tau_1)(1 + s\tau_{f1}) \dots (1 + s\tau_{fN})} \\ &\approx \frac{K}{(1 + s\tau_1)(1 + s\tau_f)^N} \end{aligned} \quad (1.3)$$

Cryogenic temperature control processes typically have $N=2$ for the number of fast time constants. The SI PID autotuning method has been implemented successfully on over hundred systems worldwide. In one isolated case the tuning method provided suboptimal performance when the user attempted to connect two DC power supplies in series to power the control heater. The two power supplies and the thermometer all constituted three fast time constants ($N=3$). The simulation studies also address system performance for different N values.

2. CONTROL NEEDS

A basic challenge for many PID tuning methods, whether based on a single or finitely many real-time tests, is the long testing time. A test based on open-loop step response (such as Ziegler-Nichols first tuning method) necessitates waiting until steady-

state is attained, to measure the open-loop gain. Real-time measurement of the slow time constant may or may not require the attainment of steady-state. A tuning test based on closed-loop oscillation is even more time consuming. Ziegler-Nichols P control oscillation experiment, in which P is gradually increased until self oscillations arise, requires the interactive observation of multiple oscillation periods to verify that indeed these are sustained, rather than decaying. Astrom's relay experiment [3] is faster (data collection time can be as short as one oscillation period) but it requires a symmetric relay (i.e. symmetric positive and negative control actions). The control is unidirectional in most temperature control applications done at the cryogenic regime.

In closed-loop P control there is a set of measured step response features that can be observed relatively quickly and accurately. Peaks and dips of the signal tend to be very recognizable – such peak values and their time of occurrence can be measured in a fully automated manner.

The SI autotuning experiments measures two specific closed-loop P control step response features – the first peak (overshoot) and next dip (undershoot). Three measured parameters associated with these peak and dip parts of the response play a role in the tuning process: the amount of peak overshoot OS (or alternatively the absolute output temperature peak T_{peak}), the gain P_c at which a specific prescribed peak overshoot value (or a specific prescribed absolute peak) is obtained, and the time difference ΔT between the first peak time and the first dip.

In [1] it was demonstrated that such information can be used effectively to tune the PID control parameters in a fast, repeatable and fully automated manner, for all small-signal control applications.

There are no "perfect" PID values for a given control environment. Near a specific temperature range there exist many combinations of PID values that may produce acceptable results. Users' needs may also vary widely. Some users may only be concerned with stable control over a long period of time at a single temperature, while others may be changing temperature often and desire rapid change. Some users may be very concerned that the temperature does not overshoot the desired set-point at all, while others are willing to accept overshoot for increased speed of response.

For most cryogenic temperature control applications, a closed-loop step response which features some moderate peak overshoot (i.e. 15% or less) is considered acceptable. Keeping in mind that the control action is unidirectional, there is zero control signal whenever the output temperature overshoots above the set-point. A closed-loop under-damped step response has negative slope segments in which the cooling down is governed by open-loop cryogenic bath influence. Whatever the cooling time is, there is no sense in requiring that the peak time be significantly shorter than the cooling time.

3. SI PID AUTOTUNING PROCESS

The iterative step-by-step PID tuning procedure, summarized below, was discovered through exhaustive experimentation. It is an empirical result. The simulation studies reported in this paper are in part aimed at obtaining better insight to eventually allow for Control Systems theoretical explanation of this particular method, and related class of methods.

The step-by-step auto-tuning procedure presented in [1] is detailed below, with some additional modifications:

Step 0) The system starts at a default pre-programmed set of PID parameters. In the case of multiple tuning procedures (say, due to a move to a new temperature range) the system's initial PID settings is made the same as the last used set of PID parameters.

Step 1) Allow the system to stabilize at any temperature. [Due to the continuing cooling action of the cryogenic bath, there is a need to apply a control heater signal equal to the system's holding power at such temperature. This requires keeping the controller's I term at a nonzero constant value. We typically stabilize the system at a temperature near which the control actions are to be taken].

Step 2) A set-point command of 2K above the existing set-point is issued using "P only" control. [The controller's I term is held constant, at the value of Step 1. No further integration action is taken at any step of the tuning process. The D control term is set to zero]

Step 3) The output peak $T_{peak,1}$ at the first peak time is measured and the initial value of $P=P_1$ is recorded. [At this point there is no need to measure the peak time].

Step 4) If peak temperature $T_{peak,1}$ is less than 3.2K above the beginning temperature the P term is adjusted up by 50% (that is, $P_2=1.5P_1$) and step response test is repeated. If the temperature peak is larger than 3.2K, the P term is adjusted down by 50% (that is $P_2=0.5P_1$) and the experiment is repeated. In either case the system is reset to its original set-point prior to starting the second test. If peak temperature $T_{peak,1}$ equals 3.2K, we jump to the end of Step 6 by setting $P=P_1$ and allowing the response to continue to the first dip following the first peak.

Step 5) The peak overshoot at the second trial $T_{peak,2}$ is measured and if different than 3.2K a linear interpolation between the two trials' observed peaks is carried to determine what P value is expected to produce a peak of $T_{peak,3}=3.2K$.

$$P_3 = P_1 + \frac{T_{peak,3} - T_{peak,1}}{T_{peak,2} - T_{peak,1}} \cdot (P_2 - P_1) \quad (3.1)$$

The value P_3 becomes P for the subsequent test step, and it is also taken to be the control value of P for the PID tuning. [Application of $P=P_3$ may not actually produce an exact 3.2K peak, as overshoots vs. P does not obey a linear model]. If $T_{peak,2}=3.2K$ we go to Step 6, setting $P=P_3=P_2$ and allow the response to continue to the first dip before being reset to the original set-point.

Step 6) Using the P value obtained at Step 5, a third trial [i.e. a 2K set-point shift] is performed, and the time ΔT between the first peak and the first dip [i.e. occurring after half cycle of the decaying oscillations] is recorded.

Step 7) The controller's I term is set to the oscillation period of Step 6: $I = 2 \cdot \Delta T$.

Step 8) The controller's D term is set to $D=I/4$.

For a simulated illustration of the SI PID auto tuning steps see the upper right window of Figure 5.1.

Comments: (a) The empirical search that historically led to the above tuning procedure involved a study of a large number of step response records each containing a complete step response all the way to steady-state. These records revealed the special role played by a step P control response that has peak overshoot of around 60%. The actual SI tuning test however relies on

absolute peak values without reaching a step response steady-state and without knowing what this steady-state level is. It is known from elementary classical control theory that a step command applied to a type-zero linear process creates an output steady-state error. A search for a peak response of 3.2K (which is ideally a 60% overshoot of a 2K set-point), is more readily implemented, but is not expected to exactly match an actual 60% overshoot. The simulation studies of the SI method that we discuss here explore both the tuning method based on measured overshoot and steady-state values as well as the accelerated method based on measured peak values. (b) The linear interpolation process (3.1) done between the first two peak values predicts a 3.2K peak, but in reality the peak response obtained in trial 3 is only in the vicinity of this desired value. The auto-tuning performance seems fairly robust to such procedural approximations. (c) The above tuning procedure, done with a set-point change of 2K, was checked experimentally to work successfully for set-point changes as large as 10K. There are applications in which even larger set-point changes are sometimes issued, but in these cases the heater quickly reaches its upper saturation limit and stays there for a period as the temperature approaches the desired level. "Large-control-signal" simulations taking into account model nonlinearities are not discussed in this paper. (d) The D term has been "traditionally" set to $D=I/4$, and as such yield a double real controller zero, however the simulation tools presented in this paper easily allow exploration of this issue in a little more detail. (For example the lower right window of Figure 5.1 shows the closed-loop step response with D set to $D=I/7$, while keeping the same P and I values).

4. SIMULATION GOALS

The practical success story of the SI PID auto-tuning method raises four fundamental questions:

- 1) Can a theoretical justification to the SI method, stemming from control systems theory, be found?
- 2) Is the SI PID tuning technique limited in scope only to cryogenic temperature control systems, or is it relevant to other industrial process control applications?
- 3) Can the SI PID tuning method be improved or extended?
- 4) What circumstances can cause the SI tuning procedure to fail, either by producing no results or by producing PID setting recommendations that may yield unsatisfactory performance?

Simulation goal is to find convincing answers to the above questions.

Generally speaking, simulations always allow a much larger parameter space exploration than experimental studies. Simulations also allow experimentation with hypothetical "What-If " questions. For instance, the impact of specific nonlinear effects can be studied in a selective manner. Furthermore, all nonlinearities may be temporarily removed from the model in an attempt to obtain better insight about the basic mechanisms involved. Simulation models may use actual experimentally-found parameters, or employ fictitious (yet reasonable) parameter values.

It may not be obvious how to translate the above generally phrased goals into a set of specific simulation objectives. Let us elaborate on some of the potential difficulties:

Many of Control Theory design tools address “optimality” in some sense. The SI autotuning method aims at “acceptability” rather than optimality. There is no attempt to create “the fastest possible closed-loop response” due to the inherent limitation created by the unidirectional control action. There is no attempt to create “no overshoot” or the “smallest possible overshoot”.

Many control methods assume a known model (possibly with some parameter tolerances) or a model that can first be identified prior to designing the control law. A standing conceptual dilemma in the literature of PID Control is about the physical meaning of the PID tuning test. Some researchers argue that the Ziegler-Nichols test [4] or related relay-control tests [3, 5] essentially identify a single point on the process’ frequency response curve. Voda and Landau [5] argued that the information obtained by the oscillation parameters created by a relay in closed-loop essentially identify the impact of all unmodeled process dynamic effects. With that in mind, one would be attempted to ask “what kind of information about the unknown parameters of the cryogenic process is revealed by the SI tuning experiment (i.e. Steps 1-6)?” If we assume a process model as in equations (1.2) or (1.3), is the collected information (all measured peak values and their time of occurrence and the gain P at each trial) sufficient to uniquely identify the parameters of the model (K , τ_1 , τ_f (and N) or L)? Can we really argue that “the process (neglecting all nonlinear effects) is effectively fully identified”, and therefore we only need to assess the optimality of the proposed PID control values? These are all open questions.

In [1] the authors presented experimental curves taken from the SI M9700 temperature control process showing how the gain, slow time constant and time-to-10% (interpreted as a crude approximation to the process’ pure time delay L) all behave as monotonically increasing functions of the temperature T . These curves clearly deserve a better physics-based theoretical investigation before jumping to a conclusion that this particular parametric dependence of the transfer function may distinguish a cryogenic temperature control system from other industrial control processes. The SI PID tuning method is basically intended for small-signal applications so the real issue may be the applicability of the method to other industrial small-signal applications. In such cases the parametric characterizations are of no concern. Some industrial processes may exhibit multiple slow time constants, greater than two fast time constants and even open-loop instability and non-minimum phase behavior. Will the SI method produce good PID values for these sets of linear models?

Questions 3 and 4 can be assessed together. To start with, the SI PID tuning method in its most basic form (Section III above) assumes an arbitrary choice of initial P control value P_1 , assumed to produce a stable yet under-damped step response. What if the application of $P=P_1$ fails to produce an overshoot? What if P_1 is too large and creates closed-loop instability? In either case the SI method would become stuck and may not be able to recommend any PID values. The first few steps of SI tuning are also implemented over relatively short time duration

(from $t=0$ to a little past the first peak time) meaning that the method may fail to recognize closed-loop instability. Waiting for the second peak may reveal diverging behavior, but in the SI basic form this is not done. A potential “improved SI method” may involve, for instance, a preliminary P search prior to starting “Trial 1”. Another tradeoff version of the SI tuning method may involve data collection up to the second peak time enabling instability detection at the cost of a slightly slower tuning process.

For the SI M9700 process a typical ratio (at all temperatures) of slow to fast time constants is about 10:1. This is probably typical of many other cryogenic temperature control systems. Will the method work and produce satisfactory results for much larger or much smaller τ_1/τ_f ratios? It is important to explore the boundaries of the time constants ratio range for both the basic SI process as well as possible alternative versions. Will the method work for the number of fast time constants $N>2$? Will it work for a pure time delay (representing $N=\infty$)? Will it work for $N=1$? If so, then the SI PID tuning method may be very relevant to other industrial sectors.

An even more ambitious task, if it turns out that the basic SI method is good only for $N=2$ and for a very limited range of time constant ratios, is to find suitable modifications to the SI basic algorithm. One of the possible improvements to the method may include an improved “PID Control Tweaking Strategy” beyond the qualitative method of reference [1]. Understanding how tweaking works for cryogenic applications may pave the way for understanding how PID values may be set differently to accommodate a broader range of systems.

5. THE SIMULATOR

The linear model simulator is implemented using Matlab® Version 7.0.1 which includes the Control System Toolbox.

The simulator uses a Matlab Control System Toolbox LTI linear model with one slow time constant and N fast time constants (equation (1.3)). The program flow mimics the actual SI autotuning process – the first three P control step responses followed by a closed-loop PID control step response. Figure 5.1 is a representative view of the program’s user-interface (front panel). The example shown in Figure 5.1 is the simulator’s default example and it features a fictitious process that has a gain of $K=3$, a slow time constant of $\tau_1=100$ and two fast time constants of $\tau_f=10$, as shown on the top left corner of the user interface panel.

The initial value of $P=2$ is shown in the upper left corner of the User Interface and is arbitrary. The linear interpolation for P is aimed at finding P value that yields an approximate target overshoot of 60% (or a comparable absolute peak temperature). This entry is shown in the upper left corner of the panel and is user-defined. The upper three graphs on the left side of Figure 5.1 show the first three P control experiments and the lowest left graph provides the PID control implementation result.

In the second step response, the program is set to detect the first two high peaks to be able to detect any unstable process behavior. As a contrast, the upper graph on the right of Figure

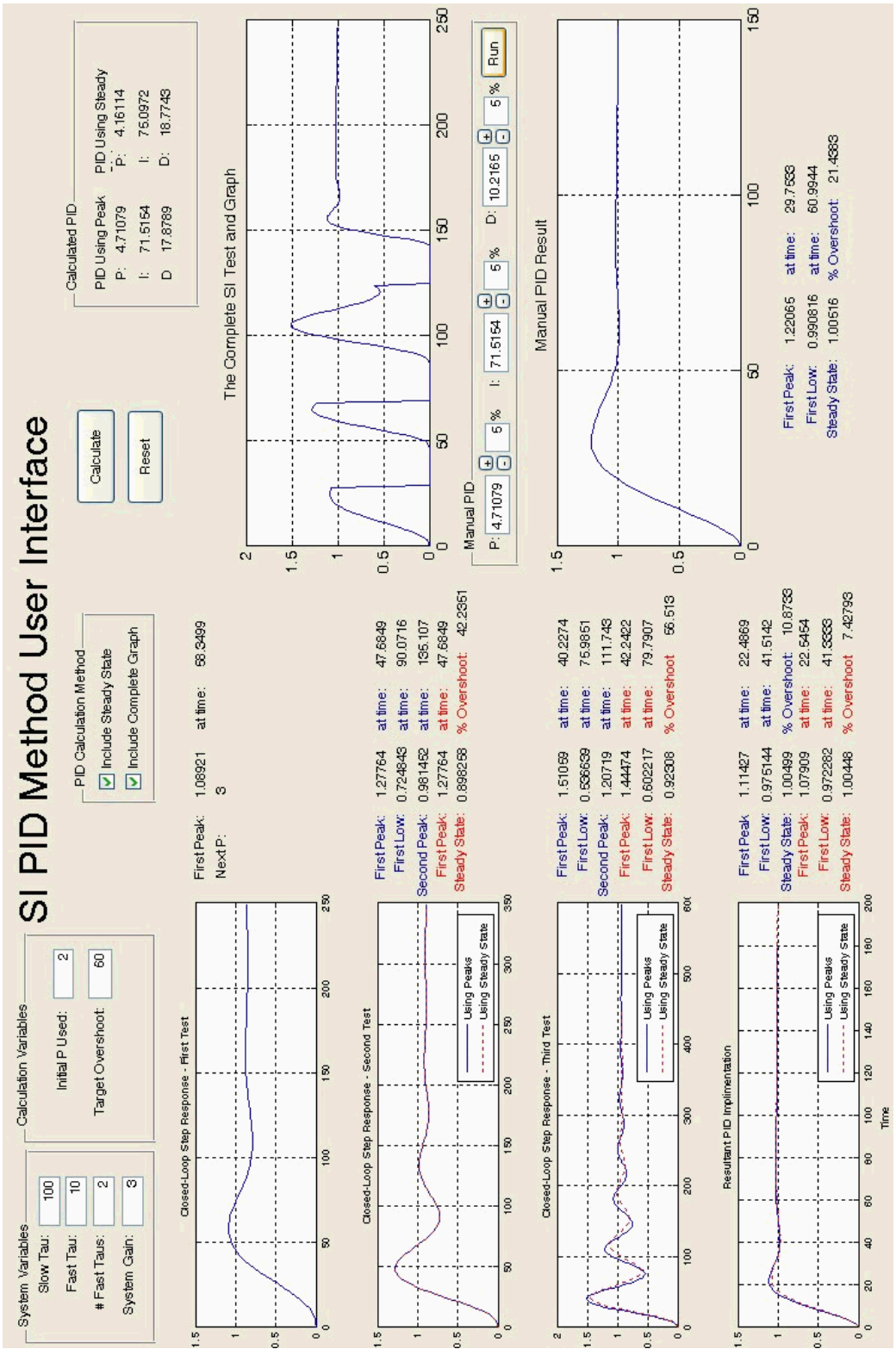


Figure 5.1: User Interface after running the default example. In all figures X axis title is Time (same units as for the time constants) and all Y axes titles are Temperature Change (in K)

5.1, entitled “The Complete SI Test and Graph”, depicts the entire tuning experiment as done in practice at Scientific Instruments, Inc. As can be observed in the actual SI tuning experiment, the output is reset after the first peak is detected.

Another noteworthy feature of the simulation program is the display of two parallel tuning algorithms. The one denoted as “Using steady-state” measures steady-state and overshoot values while the other termed “Using peaks” measures absolute peak values. The latter is the one that is used in practice. In the above example, the two methods yielded almost the same second step response. Only in the third experiment do we begin to notice slight differences. The two methods produce slightly different sets of recommended PID control values as shown on the upper right side of Figure 5.1. The combined graph on the upper right side of the panel portrays only the method of “Using peaks”. The graphs are shown with a set-point change of 1. In a linear model the size of the step input does not matter. In practice, the SI method always uses a 2K set-point change. The simplest way to integrate that into the simulator is to add a Y-axis scale factor of 2 to all time response graphs (which has not been done).

The “Complete SI Test and Graph” as well as the “Using Steady State” response curve are provided only if the user checks the respective buttons at the upper center part of the panel. Leaving both out can considerably shorten the simulation time.

The lowest right quarter of the panel is reserved for the user to manually experiment with the resulting PID values. For instance, in Figure 5.1 we show the effect of changing the D control from $D=1/4$ to $D=1/7$. Similarly the P and I values may be modified manually while comparing such tweaking result to the original SI tuning result. The user interface also allows the P, I and D parameters to be tweaked up or down by a fixed percentage. For instance, if we want to repeatedly increase I by 10% (always with respect to the last revised value), all we have to do is to change the tweaking percentage from its 5% default value and click on the (+) for each intended change. The program computes the new I value, and then waits for the user to click “Run”.

When running the simulator m file for the first time, a user interface panel with blank windows is displayed. Clicking on the top “Calculate” button starts the SI tuning experiment run. At the end of this run, the lower right window continues to be blank. After manually setting the Panel’s P, I and D parameters and clicking on the “Run” button, the panel fills up completely (as shown in Figure 5.1). The above example provides the simulator’s default parameters when the “Reset” button is depressed.

When conducting manual PID setting experiments, the “Calculate” button allows the resetting of the PID values to the ones recommended by the SI tuning experiment.

6. TIME CONSTANT RATIO EFFECT

The first application of the simulator described in Section V is the exploration of the effect of τ_1/τ_f variations on the linear model of (1.3) with $N=2$. In the SI M9700 process that ratio was shown experimentally to be around 10 at all temperatures 15-100K. Let us demonstrate the effects of varying the time constants ratio on the default simulator’s case.

Without changing the initial $P=2$ we observe that as τ_1/τ_f is increased, the overshoot of the first trial step response becomes smaller. Figure 6.1 depicts the SI tuning results for $\tau_1/\tau_f = 33.34$:

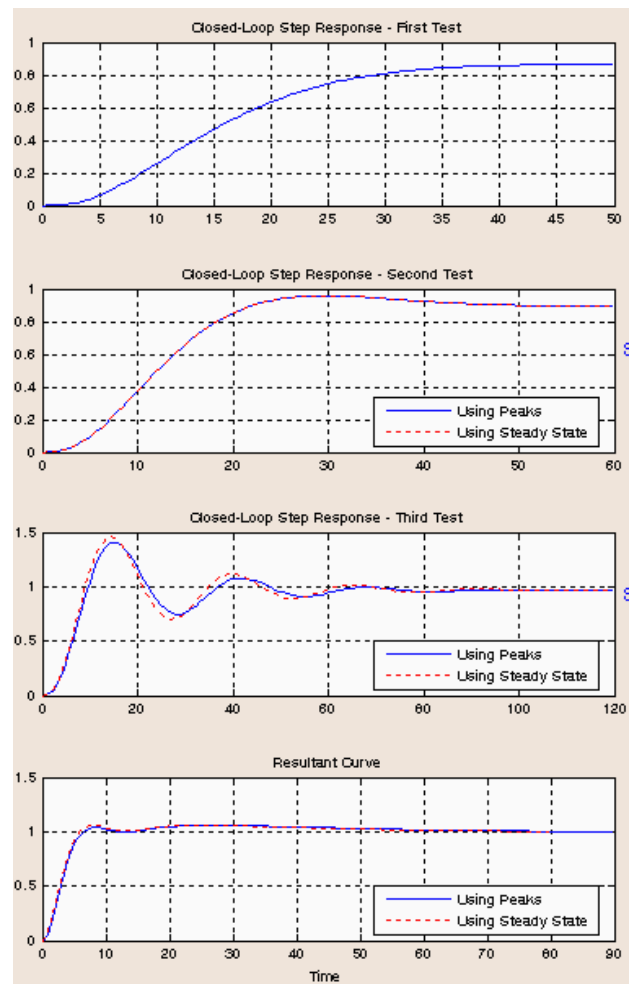


Figure 6.1: SI tuning results for time constants ratio of 33.34. In all figures X axis title is Time (same units as for the time constants) and all Y axes titles are Temperature Change (in K).

An attempt to increase the ratio further resulted in a program error message, due to “no overshoot” in Trial 1. A manual increase of the initial P value from $P=2$ to $P=3$ enabled an increase in the time constant ratio from 33.34 to about 50. For processes of the same type as the default simulator example there always exists a “large enough” initial P value that assures some initial overshoot amount and subsequently a successful tuning experiment. In practical applications without knowing what the process gain K could be there is no way to ensure that an arbitrarily selected initial P value is “large enough”.

Theoretically, there should be no upper limit on the ratio τ_1/τ_f but it necessitates a slight modification to the SI tuning method basic form. The change involves a search for a first overshoot. Starting with an arbitrary value of P, if the first step response fails to produce a peak (up to $t \sim 2\tau_1$), P may be successively

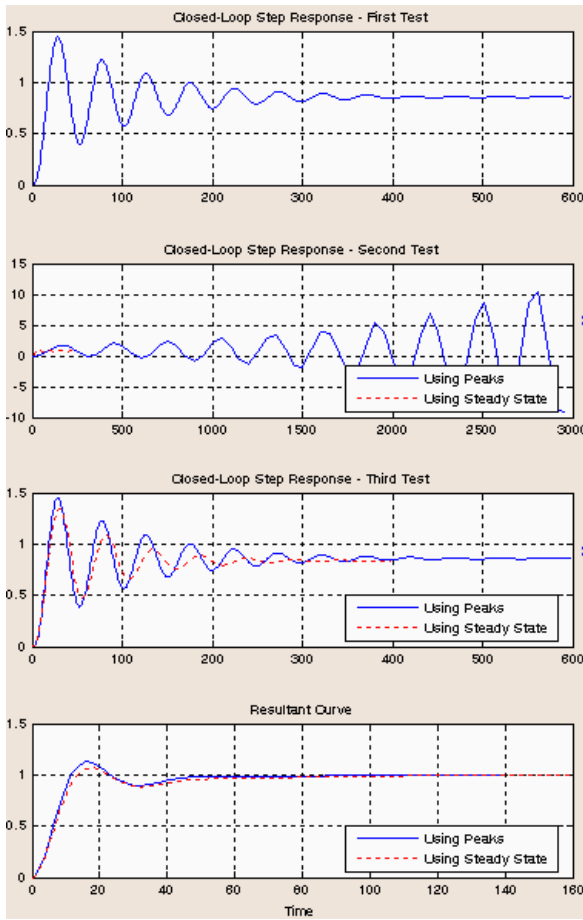


Figure 6.2: SI tuning results for time constants ratio of 1.8. In all figures X axis title is Time (same units as for the time constants) and all Y axes titles are Temperature Change (in K).

raised by 50% with the experiment repeated until a peak overshoot is detected. At that point that last run is declared as the “official” Trial 1.

Lowering the time constants ratio from 10 towards 1 presents an opposite difficulty – Trial 1 or 2 may feature an unstable response. If detected the SI tuning process must be aborted. If the instability goes undetected (because the tuning control program resets the system after the first peak and could not know that the second high peak is larger than the first) the SI tuning procedure may produce erroneous results.

Starting from the program’s default example settings ($K=3$, $\tau_1=100$, $\tau_f=10$, $N=2$ and initial $P=P_1=2$), when multiple runs are conducted, successively lowering τ_1 all the way to 3, the SI algorithm steps produced stable step response with both methods. For $1.8 < \tau_1/\tau_f < 3$, the one method based on absolute peaks failed to detect an unstable step response in Trial 2. Interestingly, despite the instability the erroneous PID settings were not as bad as one might expect (yielding a closed-loop overshoot of less than 20%). The method based on overshoot values (computed knowing steady-state values) continued to produce good PID control performance all the way down to τ_1/τ_f

= 1.8 (shown in Figure 6.2). When the method based on overshoots started to produce unstable Trial 2 response, the simulator program aborted.

The above threshold ratios appear to carry no significance at all. When the same example was run with a lower initial P guess ($P_1=1$) both methods produced good PID results with no instabilities all the way down to $\tau_1/\tau_f = 1$, and amazingly enough even beyond. A ratio $\tau_1/\tau_f < 1$ actually means two slow time constants and one fast time constant (an application which is entirely alien from cryogenic temperature control environments). The SI algorithm (in both methods) provided good PID control results all the way down to $\tau_1/\tau_f = 0.1$. For lower ratios the methods started to produce poor PID results that led to very slow closed loop step response. For low (but greater than 1) τ_1/τ_f the method based on overshoots consistently outperformed the method based on absolute peaks.

7. MULTIPLE FAST TIME CONSTANTS

The simulator allows for selection of arbitrary integer values of N (the number of fast time constants). As a representative example, let us demonstrate the effect of $N>2$ on the performance of the SI PID autotuning on the default example featured in Figure 5.1 (for $N=2$).

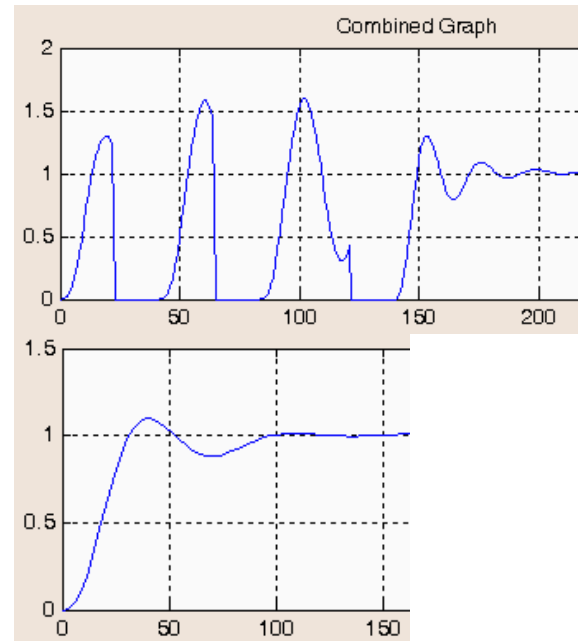


Figure 7.1: SI tuning results for: (a) (up) process with three fast time constants, (b) (down) after tweaking PID values – Increasing I and D and decreasing P. In all figures X axis title is Time (same units as for the time constants) and all Y axes titles are Temperature Change (in K).

Data collection proceeded normally yielding $P=3$, $I=105$ and $D=26$. The resulting closed loop response had an overshoot of around 30%. Slightly increasing I to $I=110$ (with $D=I/4$) and more drastically reducing P (from $P=3$ to $P=2$) yielded an

overshoot of less than 10% albeit with a slightly slower response.

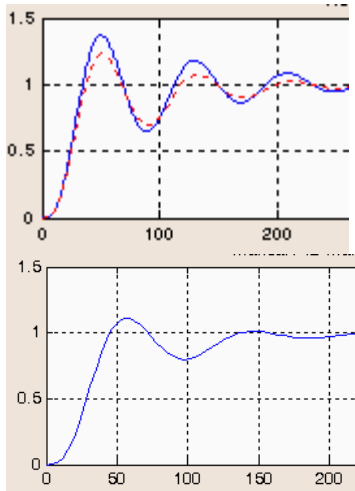


Figure 7.2: SI tuning results, for: (a) (up) process with four fast time constants, (b) (down) after tweaking PID values – Increasing I and D and decreasing P. In all figures X axis title is Time (same units as for the time constants) and all Y axes titles are Temperature Change (in K).

Figure 7.2 shows the result for the same example as N is increased to N=4.

In the N=4 example the SI PID data collection method based on actual peaks produced an unstable Trial 2 and subsequently a closed-loop overshoot of 37%. Application of the same tweaking method (as in the N=3 case) – increasing I and D by 5% and decreasing P by 33% yielded a closed-loop overshoot of about 11%.

After playing with numerous other examples and varying time constant ratios the conclusion is that the SI PID auto tuning method (in its basic form) is valid only for processes that have one slow and two fast time constants. Whenever N increases, the SI PID tuning tests may proceed (as for N=2), however the recommended PID parameters should be tweaked by slightly increasing I (and D) and more drastically decreasing P.

8. PID TWEAKING

Once the basic SI PID control autotuning is completed (or any other method of PID control tuning) a “tweaking” process may commence. Tweaking involves PID parameter modification in an attempt to improve performance, or to study performance tradeoffs. Quantitative closed-loop step response performance measures include:

- Peak overshoot (OS).
- Peak time (t_p).
- First dip size (US_1), this is the amount of first undershoot (i.e. deviation with respect to the steady-state level.)
- Second overshoot (OS_2) – the amount of the second peak overshoot.

For all the above measures – the smaller the measure the better the step response is. It is more customary to replace measures (c) and (d) with the system’s 2% or 5% settling time (t_s). What is considered an acceptable closed-loop step response is one for which US_1 and OS_2 are both smaller than the $\pm 2\%$ “sleeve” defining the 2% settling time. The only reason for including US_1 and OS_2 in the tweaking rules table (below) is the fact that the simulator’s user interface (Figure 5.1) allows explicit computation of peak values and times, but no settling time calculations were included in the code.

The “manual PID” tuning provisions of the simulator (see the lower right corner of the simulator panel) allow for very convenient study of various PID tweaking strategies applicable to models of equation (1.3) type. In this section we look only at the case of N=2, however the study covers a large range of τ_1/τ_2 ratios.

Tweaking typically involves a repeated incrementing of each of the control gains P, I and D by a fixed small percentage amount and in a fixed direction – either upwards or downwards. Table 1 summarizes partial tweaking results for $\pm 5\%$ increments, each computed with respect to the previous value of each term.

Glossary of short-hand notation used in Table 8.1:

- (+) : “Incrementing upwards” (for PID) or “Increasing” (for a response measure).
- (-) : “Incrementing downwards” (for PID), “Decreasing” (for a response measure) or “Not Recommended” (in Comments).
- (=) : “Keeping constant” (for PID) or “Unchanged” (or Unaffected) (for a response measure)
- (\approx) : “More or less stays unchanged” (i.e there could be slight ups and downs) (for a response measure)
- (\rightarrow neg): Response approaches one with one peak and one dip, and slow monotone convergence to steady-state (from below).

P	I	D	OS	t_p	US_1	OS_2	Comments
+	=	=	+	-	+	+	
-	=	=	-	+	-	=	
=	+	+	-	-	+	\rightarrow neg	
=	-	-	+	+	\rightarrow pos		-
+	+	+	+	-	+	\rightarrow neg	
+	-	-	+	=	=	+	-
-	+	+	-	=	+	\rightarrow neg	
-	-	-	=	+	\rightarrow pos		-
=	=	+	\approx	\approx	+	\rightarrow neg	
=	=	-	+	+	\rightarrow pos		-
=	+	=	\approx	\approx	+	\rightarrow neg	-
=	-	=	\approx	\approx	\rightarrow pos		-
=	+	-	+	+	+	\rightarrow neg	-
=	-	+	\approx	-	+	+	

Table 8.1: 5% Tweaking Results for model (1.3) with N=2.

(\rightarrow pos): Response approaches one with one peak and slow monotone convergence to steady-state (from above).

The dependence of response features on the direction of parameter incrementing is generally nonlinear. Therefore, it is incorrect to assume that when a parameter is incremented down, the response feature will go in the opposite direction to the one for positive incrementing. That is why so many combinations had to be tested. Not all PID tweaking combinations have been explored. For instance, incrementing P and I up while keeping D constant, or while incrementing D down.

Some PID tweaking recommendations made in [1] appear to be invalid, now that a simulator allows a more detailed study of the issue.

There are tweaking directions that are absolutely not recommended: a) Decreasing I and D, or even just one of them seem to never improve the performance, b) Increasing I alone seem to diminish the response quality, c) Increasing I while decreasing D is not a good idea either.

A symptom of taking too much I and D is a response denoted by " \rightarrow neg" (see glossary above the table). Similarly, one recognizes a response shape in which I and D are too low (denoted as " \rightarrow pos").

Here is a summary of a few "good" tweaking options:

- 1) Overshoot may be reduced by decreasing P while keeping I and D constant. It comes at a price of slower and more under-damped response.
- 2) Reduction of both the overshoot and peak time may be obtained by increasing I and D while holding P. The price paid is a larger settling time (as convergence to steady-state creeps from underneath the steady-state level).
- 3) Reducing overshoot without changing the peak time can be done by decreasing of P while increasing I and D. As in (2) the settling time increases.

The simulator can also be used to view larger increments in the PID parameters. For instance, the question of "PI vs. PID control – which one is better?" may be studied by manual adjustment of P and I gains (when setting $D=0$) to see whether or not any equivalence of responses may be obtained. Preliminary observations indicate clear superiority of PID over PI in all scenarios that were studied.

9. NONLINEAR SIMULATION

Open loop step response tests of the SI M9700 system are reported in [1]. The paper includes measured steady-state graphs of the gain, slow time constant, holding power and time-to-10% response, all as functions of the temperature T. The paper also includes polynomial regression curve formulas with high goodness-of-fit figures (obtained using Microsoft Excel's Chart Wizard).

The short history of the SI PID tuning simulation studies started with the fundamentally flawed Simulink[®] model shown in Figure 9.1. The model in Figure 9.1 uses the curve-fit formulas for the gain $K(T)$, slow time constant $\tau_1(T)$, pure-time delay $L(T)$ and the temperature-dependent holding power, in an attempt to let these parameters be automatically modified in real

time as the output temperature T varies. The model is fundamentally flawed and numerically unstable. Physically these parameters cannot possibly change instantly. There is no obvious experimental measuring technique that may reveal how fast is the "mixing time" (that is, how long does it take for parameters of a linearized model to reach parametric steady-state).

The second flawed attempt involved the creation of temperature-dependent small-signal model shown in Figure 9.2. The idea was to use the curves for $K(T)$, $\tau_1(T)$ and $L(T)$ to provide constant parameters set at $t=0$. This is done via a zero input integrator. For instance, if we want to study temperature control behavior near $T=30K$, an initial condition of $T=30K$ is assigned to the integrator whose output is T, and subsequently the system's dynamic parameters are computed once. Such a model may be valid only if the set-point commands are small enough (say no more than 5K).

To the authors' surprise the PID parameters that were obtained experimentally (in the model of Figure 9.2) failed to produce similar or acceptable results in the simulated model. The crucial mistake, as was found out, was the inclusion of a pure-time delay L. It was originally assumed (as in most standard PID control references) that a pure-time delay can capture the effect of all fast time constants. In that respect the curves of "Time-to-10% vs. T" were crudely interpreted as curves of $L(T)$. The failure of the simulator (shown in Figure 9.2) eventually led to the important realization that the SI PID autotuning method is very sensitive to the exact number of fast time constants.

The correct way of using the simulator of Figure 9.2 is to replace L with two fast time constants. There is a need to map the time-to-10% temperature curves to $\tau_f(T)$ curves. The graphs shown in Figure 9.3 were developed in a Matlab[®] m. file that repeatedly ran the step response of a third order linear system with one slow time constant and a fast time constant of multiplicity 2. The time to 10% of the step response steady-state value was recorded as function of τ_f . Graphs were parameterized by the value of the slow time constant τ_1 . For the SI M9700 system that has a time constant ratio of about 10 the graphs of Figure 9.3 indicate an approximate 3:1 ratio between the time-to-10% and the fast time constant value. One may now map the temperature dependent time-to-10% curves to temperature dependent fast time constant curve and modify accordingly the model of Figure 9.2. Further testing of nonlinear effects on the SI PID auto tuning process is deferred to a future paper.

10. FUTURE WORK

Simulation studies of the SI PID auto tuning method are far from complete. The linear simulator can be further used to extend the basic PID tuning rules developed for third order processes with two fast time constants, and to extend the basic SI data collection method assuring that no-overshoot or unstable step response results can be contained.

The nonlinear simulation study has practically not yet started at all. This may hold the real key to a full understanding of what it is that distinguishes cryogenic temperature control problems from other industrial control problems.

The insight obtained from the simulation studies should bring us

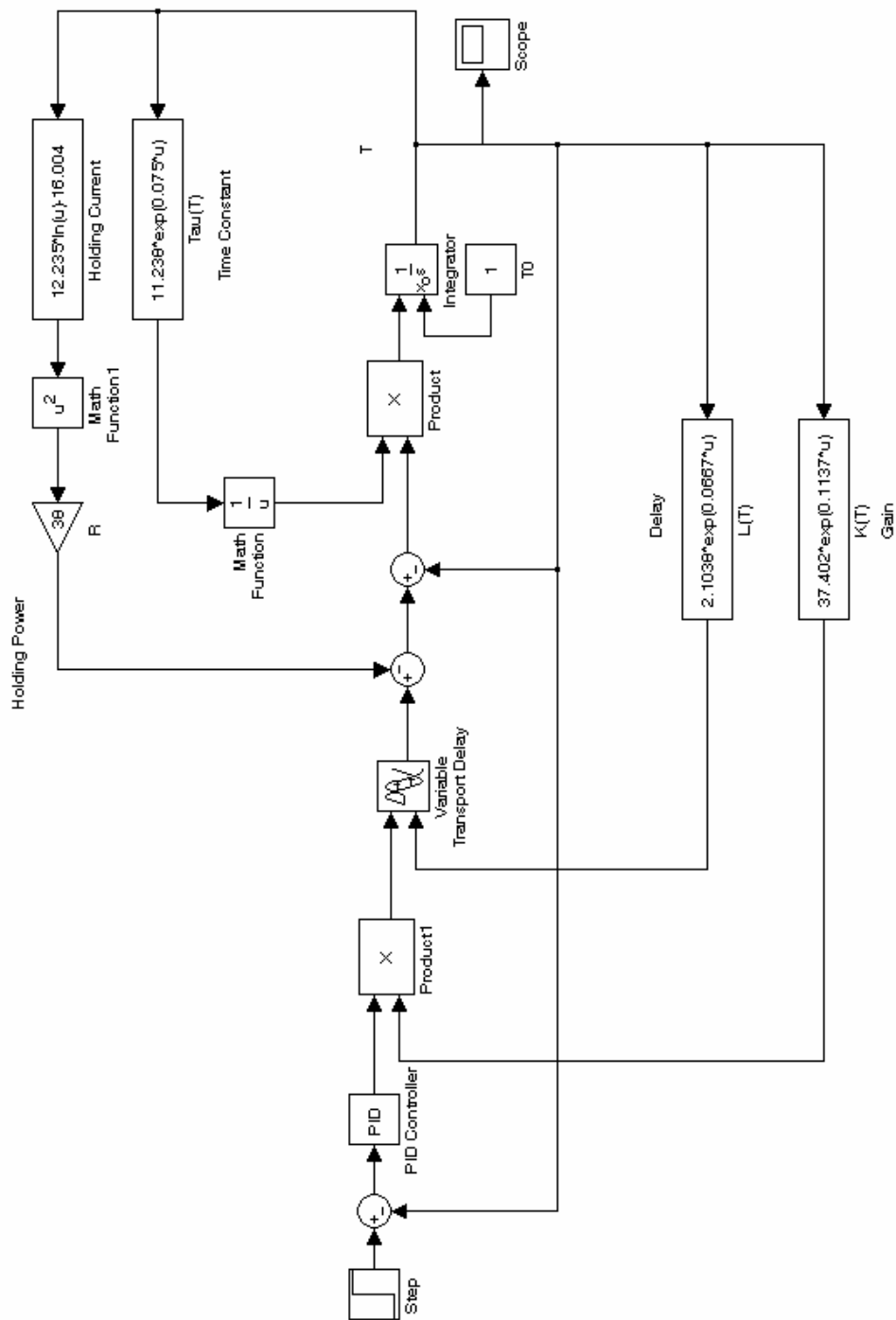


Figure 9.1: Attempted temperature-dependent Simulink[®] dynamic model for the M9700 system.

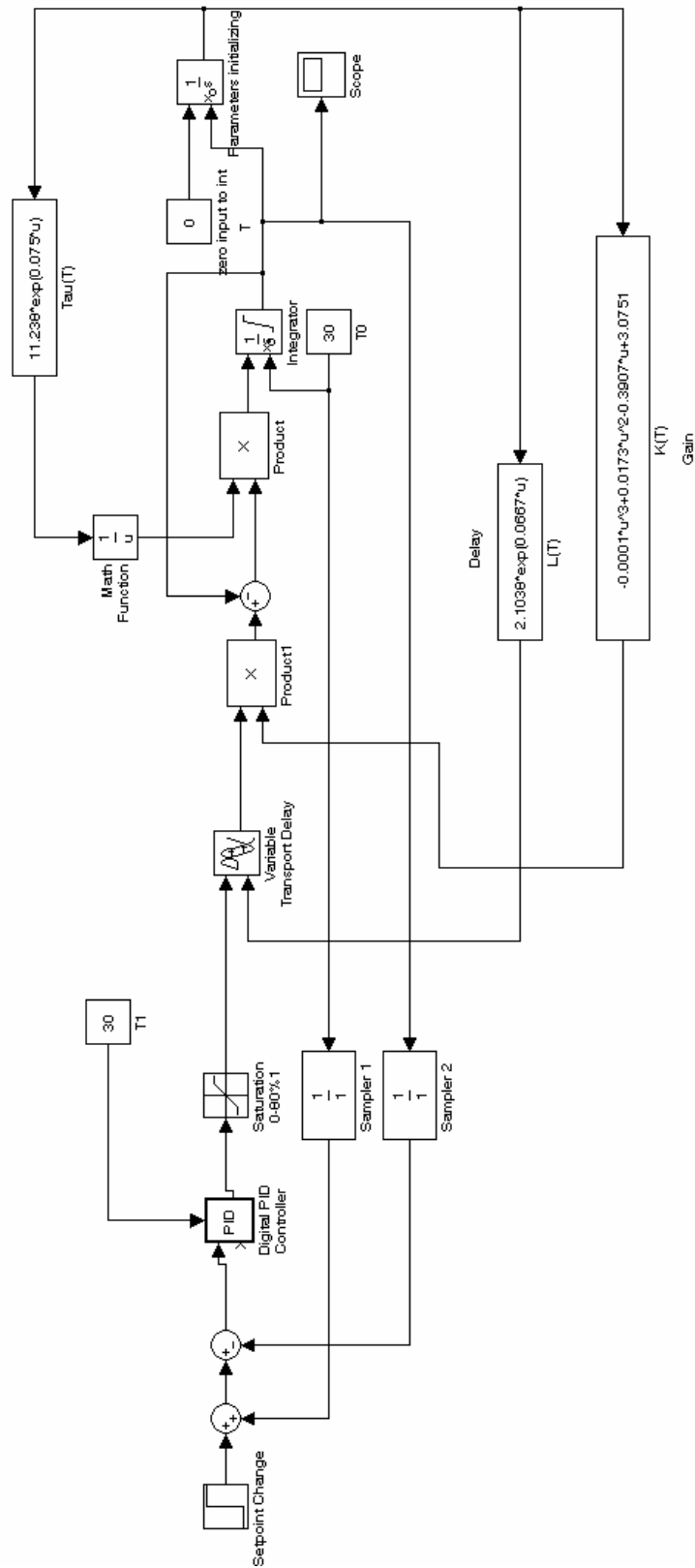


Figure 9.2: Attempted small-signal M9700 Simulink® simulator.

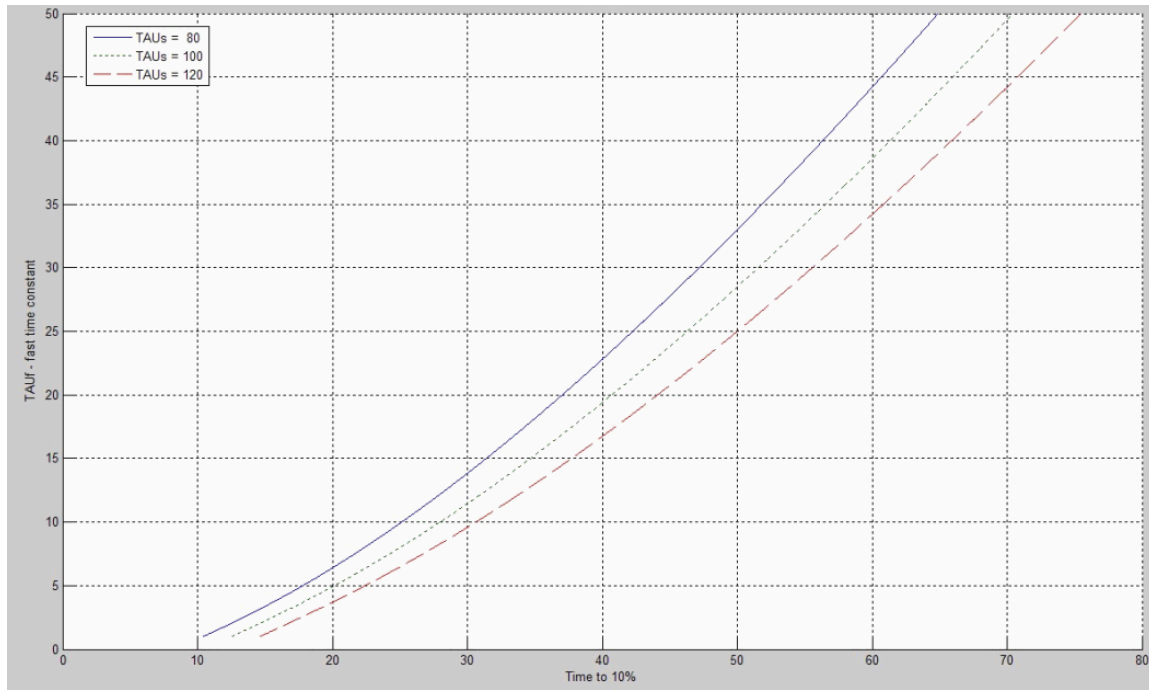


Figure 9.3: X axis is the time that it takes the open-loop step response to reach 10% of the steady-state value. Y axis is the fast time constant τ_f (assuming two identical fast time constants). Graphs are parameterized by the slow time constant τ_1 . The larger τ_1 the lower the graph.

closer to an analytical justification of this important class of PID control methods.

11. REFERENCES

- [1] David Sheats, Zvi S. Roth, Mark Sheats, and Elizabeth Strehlow, "Autotune of PID Cryogenic Temperature Control Based on Closed-Loop Step Response Tests", Proceedings of the 17th Conference on Recent Advances in Robotics (FCRAR 2004), Orlando, Florida, May 6-7 2004.
- [2] E.M. Forgan, "On the use of temperature controllers in cryogenics", Cryogenics, Vol. 14, No. 4, April 1974.
- [3] K.J. Astrom and T. Hagglund, "Automatic Tuning of Simple Regulators with Specifications on Phase and Amplitude Margins", Automatica, Vol. 20, No. 5, pp. 645-651, 1984.
- [4] J.G. Ziegler and N.B. Nichols, "Optimum Settings for Automatic Controllers", Transactions of the ASME 1943
- [5] A. Voda and I.D. Landau, "A Method for the Auto-Calibration of PID Controllers", Automatica, Vol. 31, No.2, February 1995.

APPENDIX

Readers who have access to Matlab[®] version 7 may download the m. and fig. files of the simulator by clicking on the icons.



The paperclip is the Matlab .m file and the push-pin is the Matlab .fig file. Click on each icon to save the files into the Matlab version 7 Work directory. From Matlab run the userInterface.m file.