THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
Three Park Avenue, New York, N.Y. 10016-5990

99-GT-61

DESIGN AND EVALUATION OF AN AUTO-TUNING CONTROL SYSTEM FOR AN ALTITUDE TEST FACILITY

R.H. Luppold
Luppold & Associates, Inc.
West Newton, Pennsylvania

R. Meisner
Meisner Engineering
Glastonbury, Connecticut

J.M. Norton
Pratt & Whitney
East Hartford, Connecticut

Abstract

Simulated altitude testing of large aircraft engines is a very expensive, but essential step in the development and certification of gas turbines used by commercial airlines. A significant contributor to the cost of this process is the time-intensive task of manually tuning the facility control system that regulates the simulated flight condition. Moreover, control system tuning must be performed each time the test conductor changes the flight condition. An adaptive control system that automatically performs this task can significantly reduce the costs associated with this type of engine testing.

This paper examines the features of an auto-tuning controller architecture that contains both disturbance feedforward and PID feedback components in a two-input, two-output multivariable configuration. The paper reviews the underlying concepts of an auto-tuning system and contrasts its advantages/disadvantages with respect to other adaptive control techniques. The algorithm used to automatically tune the controller does not require a facility model. However, a nonlinear facility model was developed and used to substantiate a decoupled-loop design approach, to validate the controller design concept, and to evaluate the resulting adaptive control system design performance. This analysis and other practical design issues that impact the auto-tuning control system performance are addressed in the paper. The paper also presents results that illustrate the automatic tuning sequence and the disturbance rejection performance exhibited by this system during large engine transients at several key points in the flight envelope. The auto-tuning controller described in the paper was implemented at a Pratt & Whitney flight test facility used in the development of large, high bypass ratio gas turbines.

Introduction

The development and certification of aircraft engines require extensive testing to verify and validate the propulsion system design. One facet of this testing examines engine performance and stability at various points in the operational flight envelope. Acquiring test data at altitude presents the engine manufacturer with two alternatives; either flight test the engine on an aircraft, or ground test the engine in a simulated altitude facility. The latter is obviously preferable to the former from a cost and schedule standpoint, especially in the early development phases of a new engine.

Simulated altitude testing of large aircraft engines requires a facility with significant air handling capabilities. The physical size and time scales of the facility dynamics are directly analogous to many process control problems documented in the open literature. The control problem addressed in this application focuses on regulating the Mach number and altitude at a specific test point. The control problem formulation does not directly regulate these performance parameters. A more natural formulation is the regulation of the plenum and chamber pressures to setpoints defined by the inlet and ambient pressures at the target flight condition, by modulating the inlet and exhaust valves, respectively. Minimizing deviations in altitude error requires good control of chamber pressure. Alternatively, deviations in Mach result from both chamber and plenum pressure perturbations. Large, rapid engine transients will severely test the capability of the facility control system to maintain the test point. Hence, the control system design challenge in this application is to formulate a controller architecture that rejects the disturbances induced by these engine transients.

Presented at the International Gas Turbine & Aeroengine Congress & Exhibition
Indianapolis, Indiana — June 7–June 10, 1999

Given the mammoth facility and the requirements for tightly regulating the flight condition, it is not surprising to find that simulated altitude testing of large aircraft engines is very expensive. A significant contributor to the cost of this process is the time-intensive task of manually tuning the facility control system. The cost impact is further compounded when one realizes that the control system must be retuned each time the flight condition changes. A typical manual tuning scenario requires one to two hours of facility test time to reconfigure the control system. Hence, automating the tuning process will significantly impact the operational overhead costs of the facility. An automation objective was to reduce tuning time by one to two orders of magnitude.

In addition to the economic impact, control system performance was also an issue driving the desire to automate the tuning process. In operability tests, the validity of the test data depends upon the stability of the test point. Excessive perturbations in altitude and Mach number are intolerable. Hence, the design goal in this application was to regulate altitude/Mach number to ±200 feet and ±0.01, respectively. For large engine transient tests in the target facility, this represents a challenging design problem. Pure time delays (transport lags) in both the command and feedback paths as well as valve backlash nonlinearities contribute to the difficulty of achieving these design goals. Historically, the manually tuned control system employed a proportional-plus-integral (PI) controller. The dominant dynamics governing both the plenum (inlet) and chamber (exhaust) pressures are second order. To achieve good control performance by tightly regulating these pressures requires a more complex controller design, e.g., a proportional-plus integral-plus-derivative (PID) controller. Increasing the controller complexity from PI to PID incurs the cost of tuning an additional parameter at each flight condition. Hence, the time allotment for manually tuning the PID control system would increase. Another element of a manual tuning task that consumed facility time was the selection of an input signal that excited the dominant facility dynamics. The valve/actuation nonlinearities, e.g., rate limiting and valve backlash, added complexity to the input selection problem and had the potential for adversely affecting the performance of the tuned controller.

Assuming that an automatic tuning system has been mandated by cost-competitive and performance factors, the control system designers must choose an adaptive control approach that meets the objectives of this application. The candidates included:

- Model Reference Adaptive Control (MRAC),
- Self-Tuning Regulator (STR),
- Gain Scheduling,
- Auto-Tuning.

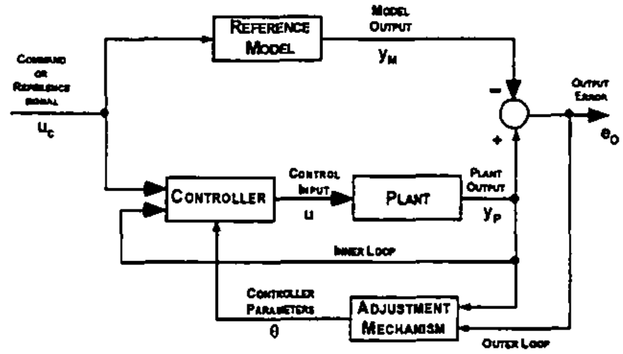

Figure 1a. Model Reference Adaptive Control.

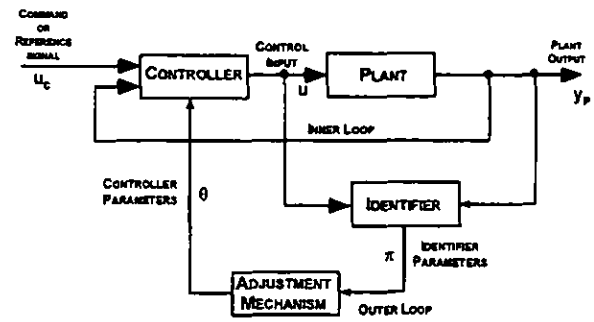

Figure 1b. Self Tuning Regulator

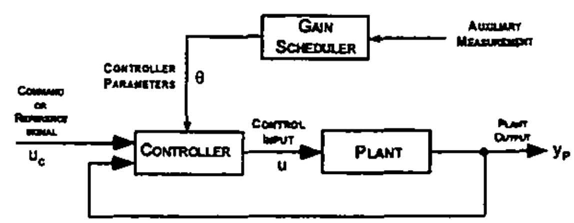

Figure 1c. Gain Scheduling

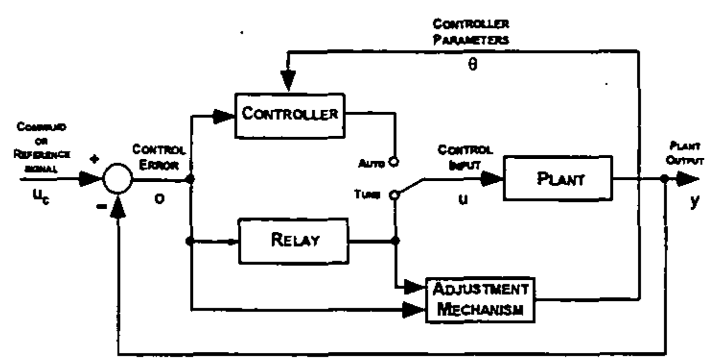

Figure 1d. Auto-Tuning

The resulting system architectures associated with each of these adaptive control formulations are shown in Figure 1. Note that both the MRAC and STR design approaches produce the familiar adaptive control system architectures that frequently appear in the open literature (Sastry & Bodson - 1989, Astrom & Wittenmark – 1989). They differ primarily in the approach taken to adjust the controller parameters as changes in plant dynamics occur. The MRAC concept is a so-called *direct*-adaptive control approach. The MRAC algorithm directly adjusts the controller parameters to force the output error, e_0 in Figure 1a, to zero. In MRAC algorithms, the output error is equal to the difference between measured system outputs and those produced by a reference model. The parallel implementation of MRAC shown in Figure 1a provides the designer with the flexibility of continuously updating the controller parameters or updating the controller parameters at a prescribed periodic rate. The engineering challenge in applying MRAC to real applications focuses on developing an adaptation technique that forces the output error to zero quickly while simultaneously maintaining system stability.

Alternatively, the STR concept takes an *indirect* approach to the adaptation problem. STR partitions the problem into a system identification phase and a controller update phase as shown in Figure 1b. While under closed-loop control, STR performs system identification experiments to formulate a control system design model. When the design model is updated, a new controller is produced and the associated controller parameters are inserted into the closed-loop system. This process can be visualized as an inner and outer loop configuration (see Figure 1b). The inner loop performs the conventional closed-loop control function, while the outer loop contains the system identification and controller update logic that incorporates plant dynamic changes. Though less susceptible to stability problems than the MRAC approach, the development of a real-time system identification algorithm inherent to the STR approach presents a challenging real-time implementation problem.

The most widely used adaptive control concept in engineering practice is the gain scheduling approach. Though this concept is intuitively simple and can very quickly adjust controller parameters to plant changes, it is an open-loop adaptation scheme as shown in Figure 1c. As such, validating and verifying the concept for a full range of expected conditions can be extremely time consuming. In the adaptive control algorithm present in this paper, an interesting variation of gain scheduling is employed in the feedforward control algorithm. Fan rotor speed can be directly correlated with the operating line of high bypass ratio, gas turbines used in commercial applications. Moreover, a change in fan rotor speed indicates a change in the engine airflow requirement. Since engine transient disturbance rejection is the primary performance goal, the designer can exploit the availability of measured fan rotor speed to schedule facility inlet and exhaust valve positions. The details of the feedforward design appear in the paper.

The last adaptive control approach considered, the so-called auto-tuning approach, proved to be the most feasible scheme for implementing an adaptive feedback control in the flight test facility application. The high-level, functional block diagram of this approach appears in Figure 1d. In addition to the stability and system identification problems associated with both the MRAC and STR concepts, successful implementations of these adaptive control systems require a priori information of the of the target application plant dynamics. During the initial phases of the design, a high level of uncertainty existed in the plant description that made the technical risk associated with MRAC and STR unacceptable. A literature search subsequently uncovered the auto-tuning concept (see Astrom – 1996 and references therein). After evaluating the features associated with this adaptive control scheme, the design team decided that this approach provided the necessary adaptation capabilities with acceptable technical risks. The remaining sections of the paper discuss the technical details associated with the auto-tune concept, and the performance achieved with the control system implemented for the altitude test facility application.

Facility Model Description

While an auto-tuned controller is inherently less reliant on plant models than the other adaptive system concepts, a model is still an extremely valuable tool during the design and development phases of the control algorithm. Analysis of system coupling, selection of a controller architecture, control algorithm functional testing, and preliminary performance/robustness testing are all heavily dependent upon the availability of a system model with adequate static and transient fidelity.

The model used to support this work was based on a FORTRAN simulation of the test facility, which had evolved over the past twenty years at Pratt & Whitney. To enable rapid prototyping and test of control algorithms, an equivalent model was developed for the MATLAB®/SIMULINK® environment. The SIMULINK facility model was built from a block diagram library that was constructed specifically for this application. The facility library consisted of blocks modelling volume dynamics, valve characteristics, actuators, venturi tubes, etc.

From a control system design perspective, there are three major plant components: (1) the test-chamber, including the engine under test; (2) the valve actuators driven by the control system, and (3) the sensors that measure the facility's response to the control system inputs. The subsequent sections discuss each model component in detail.

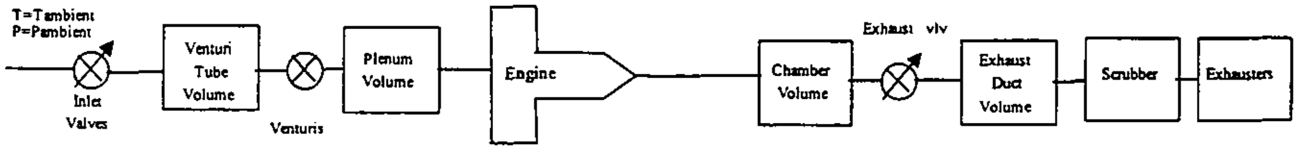

Figure 2. Test Chamber (Facility) Model

Test Chamber

Ground-based simulation of altitude operating conditions requires regulation of the engine inlet temperature, inlet pressure, and exhaust pressure to those levels corresponding to the flight condition of interest. The inlet and exhaust pressures are controlled via throttling valves upstream and downstream of the engine. Figure 2 indicates that the test chamber has two inputs, the inlet and exhaust valve positions that can be used to control the plenum and chamber pressures, respectively. The test chamber also responds to several exogenous (disturbance) inputs, the most important of which is the engine's airflow characteristic. For the purposes of this model, the engine airflow is taken as a function (unique for each engine type) of corrected fan speed, engine inlet temperature and pressure, and exhaust pressure. The modeling approach assumes that the engine fan speed is decoupled from the test chamber dynamics, and is therefore truly a disturbance input to the system.

One approach to capturing the test-chamber dynamics is to consider a lumped parameter four-state model. These states are the pressures in each of four lumped volumes, i.e., the venturi tubes, inlet plenum, exhaust chamber, and exhaust duct, as shown in Figure 2. The inlet plenum and exhaust chamber pressures represent the key process variables regulated by the control system to maintain simulated flight conditions. These variables along with the exhaust duct pressure are instrumented to provide input to the control algorithms.

Valve-Actuation Models

The inlet flow control for this facility consists of an array of inlet valves, each feeding a separate venturi tube. An electric motor drive with a local position feedback and loop closure actuates each valve independently. The inlet valves can be asymmetrically trimmed by the control system to equalize the venturi tube pressure ratio, or in some cases to close individual valves. The control algorithm developed under this program actuates the inlet valves symmetrically, so the model considers all inlet valves to move in unison.

The electrical actuation, gearing, and position feedback give rise to linear and nonlinear dynamics that are reflected in the model. These effects include:

- Transport delay due to discrete time position control
- Hysteresis due to drive backlash
- Slew rate and position limits due to drive motor and valve constraints
- First order lag dynamics due to position control.

The exhaust flow control for this facility consists of a single butterfly valve driven by a hydraulic actuator through a linkage. The actuation model for this valve is very similar to that used for the inlet valve array.

Measurement Models

The inputs to the regulator algorithm consist of the inlet plenum, exhaust chamber, and exhaust duct pressures, as well as the test engine fan speed, which is used by a disturbance feedforward controller in each channel. Since the control algorithm is implemented digitally, a zero-order-hold model on each measurement incorporates the dynamics associated with the sampling process.

The pressure transducers are located remotely, 25 to 100 feet away from the pressure tap. At low frequencies, the dynamics of this transmission line can be approximated by a transport delay with delays of 25 to 100 ms. In addition to the transport delays, an anti-aliasing/noise filter in the transducer signal path results in additional pressure measurement dynamics. Uncorrelated noise at a representative power level is added to each transducer to reflect the random component present in each of these measurements.

Controller Design Formulation

The next several sections of the paper address the underlying principles of the auto-tune control concept and some of the practical considerations that must be incorporated into the design to cope with a realistic environment. Issues covered in this discussion include: 1) the application of feedforward control to improve disturbance rejection capabilities of the system, 2) coping with measurement noise while performing automatic tuning, and 3) dealing with the multivariable control problem in the altitude facility application.

Rationale for the Auto Tune Control Concept

Unlike the MRAC and STR concepts, the Auto-Tune adjustment (adaptation) mechanism does not require any a priori information about system dynamics to compute the PID controller parameters. Moreover, an Auto-Tune system only updates the controller on an operator-demand basis. The MRAC and STR methods do not explicitly interact with the system operator. These two characteristics of the auto-tuning concept were the primary factors in selecting this adaptive concept for the altitude test facility application. This section examines the underlying features of the Auto-Tune concept and motivates the rationale for selecting a PID controller for this application.

The automatic tuning performed with this scheme can be characterized as a crude, but robust method that identifies two key parameters characterizing process dynamics. The Auto-Tune adaptation algorithm approaches the control design in a manner quite familiar to first-generation single input/single output control system designers. The fundamental idea centers on determining the gain and frequency at which the system dynamics become conditionally stable under pure proportional feedback control. These frequency-domain characteristics of the system are designated as the *ultimate gain* and *ultimate frequency*, respectively. Using Ziegler-Nichols relationships, the PID controller parameters can be determined from the ultimate gain and frequency information. It is well known that PID control systems designed with the Ziegler-Nichols method exhibit very good disturbance rejection performance, but tend to have significant overshoot when responding to set-point changes (Astrom & Hagglund - 1995). Degraded set point responses do not present a problem in the altitude test facility application since the control problem focuses completely on disturbance rejection performance. The chamber pressure and plenum pressure set points remain at fixed values throughout an engine transient test scenario.

As in most control system synthesis problems, both time and frequency based methods exist for formulating an experiment that produces the information required to compute the Ziegler-Nichols gains. In most practical control applications, a frequency-based experiment produces superior results and was the method chosen in this application. The central idea in the frequency-based approach relies on the fact that most real systems produce stable limit-cycles under relay feedback. The theoretical basis for this statement was developed in Astrom - 1991. The *method of harmonic balance* or *describing function method* (Gelb and Vander-Velde – 1968) provides the mathematical framework for analyzing relay-induced limit-cycles and extracting the ultimate gain and ultimate frequency from the experimental data.

Figure 1d illustrates the functional blocks required to setup a relay experiment. There are several points to note in examining this very simple figure. Firstly, the plant dynamics are characterized by a linear transfer function, G(s), for small perturbations about a steady-state operating point. Transfer functions are input/output models, i.e., they relate system inputs to observed outputs, and can be generalized to multivariable systems. In the facility model described in Figure 2, G(s) relates inlet and exhaust valve positions to plenum and chamber pressures, respectively. The dominant dynamics captured by the linearized facility model are low-pass in nature, i.e., the valve/actuator and volume dynamics significantly attenuate high frequency inputs.

The second point to note concerns the signals annotated in Figure 1d and their relationship to the ultimate gain and frequency. The objective of the relay experiment is to generate data that will accurately identify the system's ultimate gain and frequency, which will subsequently be used to define the controller parameters. In the limit cycle analysis that follows, we will assume without loss of generality that the command or reference signal (set-point) is zero. Setting the switch to the TUNE (tuning) position, the closed-loop system begins to limit-cycle when one or two pulses are injected into the loop through the system input port. The relay output, **u**, is a periodic square wave signal with a frequency of ω_u; hence, the ultimate frequency is determined from **u** by extracting this signal parameter. To continue the limit-cycle analysis beyond this point, we need to express the periodic control error signal, e, in terms of its Fourier series harmonics. This is the subject covered by describing function theory.

The describing function approximation truncates the infinite Fourier series of a periodic signal after the first harmonic. The validity of this approximation is quite good if the linear system within the control loop is predominately low-pass in nature. Based on the discussion in the Facility Model Description section, this assumption is valid. The describing function for an ideal relay is given by the mathematical expression, $4d/\pi a$, where d is the relay output level and a is the peak amplitude of the relay input. Based on the describing function for an ideal relay, we note that this relay model influences the loop gain of the system in Figure 1d, but does not affect the phase. Nonlinearities with this property are classified as *memoryless*.

With the system set-point at zero, the error signal magnitude, $|e|$, equals the magnitude of the plant output at the ultimate frequency, i.e. $|G(j\omega_u)|$ scaled by the relay describing function, so that

$$|e| = \frac{4d}{\pi}|G(j\omega_u)|. \qquad (1)$$

The closed-loop relationships that must be satisfied to induce an oscillation with proportional feedback are given by

$$\arg(G(j\omega_u)) = -180^\circ \;\; and \;\; 1 + K_u|G(j\omega_u)| = 0 \qquad (2)$$

where the arg(•) function represents the phase angle of the plant transfer function and K_u is the ultimate gain. Using the limit-cycle analysis relationship given in Equation (1) and the magnitude constraint of Equation (2), the ultimate gain, K_u can be expressed in terms of measurable quantities of the relay experiment, i.e.,

$$K_u = \frac{4d}{\pi|e|}. \qquad (3)$$

Since the output level of the relay, **d**, is a design parameter, Equation (3) indicates that we need to estimate the amplitude of the periodic error signal, e, to determine K_u.

The Auto-Tune Adjustment Mechanism shown in Figure 1d can be partitioned into two distinct tasks. These tasks implement:

1) a system identification algorithm that performs the relay experiment, processes the error signal and relay output, and extracts the ultimate gain and ultimate frequency, respectively,
2) Ziegler-Nichols relationships to update the PID controller parameters as functions of the ultimate gain and frequency estimated by the sys-id algorithm.

Controller	K	T_i	T_d
P	0.5*K_u		
PI	0.4*K_u	0.8*T_u	
PID	0.6*K_u	0.5*T_u	0.125*T_u

Table 1. Ziegler-Nichols Controller-Gain Relationships

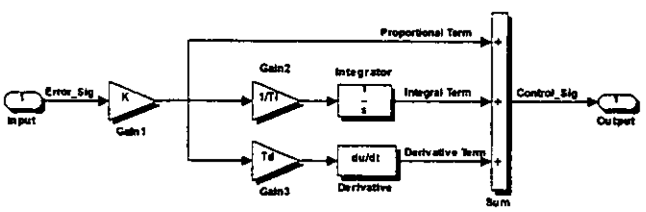

Figure 3. Standard PID Controller Configuration

The Ziegler-Nichols tuning method is an old and established methodology that forms the basis for tuning procedures used by controller manufacturers in the process control industry. Like most frequency-based compensation synthesis strategies, the Ziegler-Nichols relationships can be interpreted as a set of rules for reshaping the open-loop frequency response of the plant. Ziegler-Nichols shifts the ultimate point of the uncompensated system to a more desirable location in the complex G(s)-plane. Table 1 compiles the Ziegler-Nichols formulas for three of the most commonly used feedback compensation techniques. Note that these formulas express the proportional, integral, and derivative gains in terms of the ultimate gain and ultimate period, i.e., $T_u = 2\pi/\omega_u$. Figure 3 shows the correlation of the Ziegler-Nichols parameters in terms of a continuous-time implementation of a PID compensator.

In general, systems using Ziegler-Nichols compensation exhibit the following characteristics:

- Good disturbance rejection performance,
- Poorly damped response to changes in set-point,
- Produce better results when used in PID controllers (as opposed to PI controllers).

The explanation for the first two observations is directly linked to the basic design criterion used by Ziegler-Nichols, i.e., produce a closed-loop system with quarter amplitude decay ratio. Now, decay ratio can be expressed in terms of second order system damping ratio such that

$$d.r. = e^{-2\pi\zeta/\sqrt{1-\zeta^2}} \qquad (4)$$

where ζ is the damping ratio of the dominant second order pole-pair in the closed-loop system. With a decay ratio equal to 0.25, the damping ratio for the closed-loop system is 0.22. Second order systems with this damping ratio respond quickly to disturbances, but exhibit large overshoots in response to step set-point requests. These systems also have poor stability margins, i.e., poor robustness properties.

The rationale for the third observation is based on studies using PI and PID tuning diagrams. The tuning diagrams characterize the PI and PID controller parameters as a function of gain ratio, κ. The formal definition of gain ratio is

$$\kappa = \left|\frac{G(j\omega_u)}{G(0)}\right| \qquad (5)$$

where the numerator term is the open loop system gain at the ultimate frequency and the denominator term is the system static sensitivity, i.e., the d.c. gain. κ indicates the degree of difficulty one will encounter in trying to control the process G(s). Systems possessing small gain ratios are easy to control. As κ increases, the system becomes increasingly difficult to control. Astrom & Hagglund - 1995 showed that the range of controller parameter variations from the Ziegler-Nichols reference for PID controllers are significantly less than those of PI controllers for gain ratios ranging between 0 and 1. This result provides the analytic support for the third observation and the rationale behind the decision to use PID controllers in the altitude flight test application.

To combat the poor damping and stability margins produced by the standard Ziegler-Nichols formulas, a modified set of gains will be used by the Auto-Tune system discussed in this paper. To achieve this objective, the designer selects an alternative position for placing the ultimate point in the closed-loop system. Astrom & Hagglund - 1995 derived a set of equations that perform this function, i.e.,

$$
\begin{aligned}
K &= K_u r_b \cos\phi_b \\
T_i &= \frac{T_u}{\pi}\left(\frac{1+\sin\phi_b}{\cos\phi_b}\right) \\
T_d &= \frac{T_u}{4\pi}\left(\frac{1+\sin\phi_b}{\cos\phi_b}\right)
\end{aligned}
\qquad (6)
$$

Note that r_b and ϕ_b are the Nyquist plot coordinates specifying the desired location of the ultimate point produced with the PID compensation. In this design, the coordinates $r_b = 0.41$ and $\phi_b = 61°$ were chosen. This selection was based on the work published by Pessen - 1954. Implementing this modification of the Ziegler-Nichols relationships produces a compensated system with *potentially* significant improvements in phase margin (52° versus 28.3°). Note that the term "potentially" needs to be taken seriously since there are limitations with a tuning method where only one point on the Nyquist curve is positioned. The properties of the closed-loop system depend heavily on the slope of the Nyquist curve in the neighborhood of the negative real axis, which is not determined by the relay experiment.

From the discussion of STR methods, we can see the similarities between that adaptive concept and auto tuning. Both methods require system identification procedures to extract model parameters needed to synthesize the controller. The adaptation process is distinctly partitioned into system identification and controller update tasks, i.e., both methods are indirect adaptive control schemes. The primary difference between the two approaches is the level of complexity in the implementation of the individual algorithms. STR places no restriction on the type of controller synthesized on-line, but this flexibility incurs a cost. Most controller synthesis algorithms require a process model for determining the controller parameters and dynamic structure. Hence, the system identification algorithms employed by the STR concept are significantly more complex than the one used by Auto-Tune. The more sophisticated system identification and controller synthesis algorithms also require a priori information about time constants of the target process. As the discussion above shows, Auto-Tune requires no a priori information about the process. In the altitude test facility application, the plant dynamics vary as a function of the simulated flight condition; however, individual engine tests are conducted at a fixed flight condition. Therefore, the requirement to update the controller continuously during an engine test does not exist, and would in fact be undesirable since the repeatability of a given engine transient would be compromised. In auto-tune systems, controller tuning takes place only at the request of the operator, e.g., when a new flight condition is desired.

In summary, the Auto-Tune concept provides a framework that enables the design and implementation of an adaptive system within realistic schedule and budget constraints. The altitude test facility application possessed several characteristics that made it an ideal candidate for automating this system. For example, this application does not require a continuously adaptive controller to meet the control system performance objectives. Also, the dominant dynamics in the inlet and exhaust channels are second-order. This system property makes the PID controller a viable approach for achieving good disturbance rejection performance. Hence, the cost-reduction and performance enhancement objectives can be met with an Auto-Tune control system. Interestingly enough, the Auto-Tune system provides the necessary a priori information required by the STR approach and has been used to initialize an STR algorithm in applications reported in the open literature (see Lundh and Astrom – 1994 for an example). This benefit provides a built-in upgrade path for future enhancements of the altitude test facility control system.

Controller Architecture

The stringent set-point regulator requirements and the presence of large engine transient disturbances mandated the need for a two-degree-of-freedom controller. In the context of this discussion, the term *two-degree-of-freedom controller* implies a feedforward/feedback control system architecture. Figure 4 illustrates the top-level view of the proposed controller architecture and the process being controlled. Several points stand out in this figure. First of all, this is a two-input/two-output multivariable control problem. Second, Mach number and altitude are performance variables, but do not directly factor into the control problem. Third, elements of the feedforward controller are scheduled on fan spool shaft speed, but only a single PID feedback controller is used at a given flight condition, i.e., gain scheduling is not used. The system performance depends on the control system's ability to position the inlet and exhaust valves so that the flight condition remains constant throughout a transient engine test. This section examines some of the design issues that need to be addressed to make this candidate architecture work in this application.

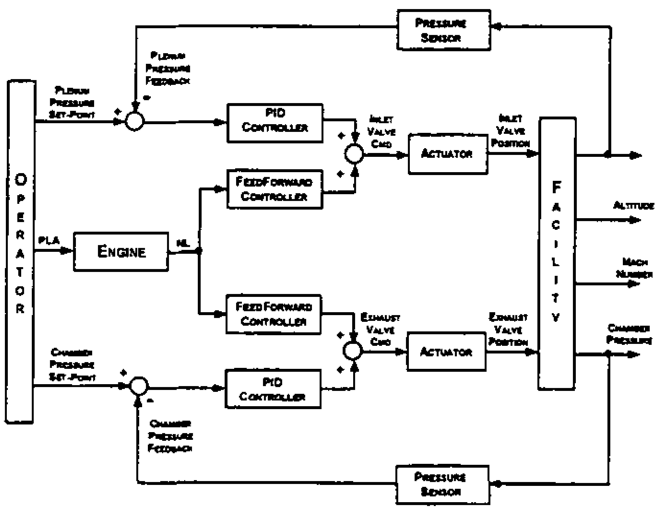

Figure 4. Flight Test Facility Control System Model

Designing multivariable control systems represents a significant leap in design complexity relative to the classical single input/single output (SISO) problem. This complexity becomes magnified when the designer wants to implement the controller synthesis process for use in a real-time application. To avoid this escalation in design complexity and its inherent cost to the program, we used the facility model to evaluate the consequences of treating the control system as two independent loop closures. The most important question to be answered by this analysis was: Can we ignore the cross coupling between the inlet and exhaust channels without adversely affecting closed-loop stability?

By inspection of the plant model (Figure 2), it appears that the plenum volume pressure can be controlled via the inlet valves that affect the inflow to this volume. Similarly, the chamber volume pressure can be controlled via the exhaust valve, which affects the outflow from this volume. This natural pairing of control input and output variables is used by the controller architecture described herein.

These two subsystems are, however, not independent since the engine induces cross-coupling terms in the open loop transfer functions. The nature of this coupling depends upon the sensitivity of the engine inlet and exit airflow to variations in plenum and chamber pressures. Recall that engine fan speed is modeled as a disturbance input, since it is only weakly affected by variations in plenum and chamber pressure. Hence, the coupling analysis focuses on airflow variations that reflect the sensitivity of engine airflow to plenum and chamber pressure at a fixed fan speed.

Since the engine inlet air density is proportional to plenum pressure, the inlet mass flow rate is also proportional to plenum pressure. Furthermore, since the exit mass flow rate equals the inlet mass flow rate (plus a small contribution from the fuel mass flow rate), the mass flow into the chamber volume from the engine is approximately proportional to the plenum pressure. Thus, there is significant cross coupling from the inlet valve to the chamber pressure.

Similarly, the chamber pressure, which is the ambient "back pressure" for the engine, can also affect engine airflow. However, for the class of engines used in the test facility application, this effect is noticeable only at low power levels. At mid to high power, the engine is sufficiently choked that little or no sensitivity to back pressure is observed. As a result, the coupling observed through most of the power range is one-way, i.e., the inlet control system (inlet valve and plenum pressure) affects the exhaust control system (exhaust valve and chamber pressure), but not vice versa.

The effect of cross coupling was evaluated numerically using relative gain array (RGA) analysis (see Skogestad & Postlethwaite - 1996) with linearized models of the plant obtained at trimmed operating conditions. In RGA analysis, we calculate a dimensionless transfer function ratio of the plant. This ratio relates the ith channel transfer function with all control loops open, to the ith channel transfer function with the other channels regulating perfectly. To satisfy the perfect regulation constraint, RGA analysis is most meaningful below loop bandwidth frequencies, and is particularly effective in systems with integral controllers. In our two-input/two output controller configuration, a single array element fully defines the entire RGA, i.e.,

$$\begin{bmatrix} P_{plenum} \\ P_{chamber} \end{bmatrix}(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} X_{inlet\_valve} \\ X_{exhaust\_valve} \end{bmatrix}(s)$$

$$\Lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} = \begin{bmatrix} \lambda_{11} & 1-\lambda_{11} \\ 1-\lambda_{11} & \lambda_{11} \end{bmatrix}$$

$$\lambda_{11} = \frac{G_{OL}(s)}{G_{CL}(s)} = \frac{\left\{ \dfrac{P_{plenum}}{X_{inlet\_valve}}(s) \right\}_{X_{exhaust\_valve}=c}}{\left\{ \dfrac{P_{plenum}}{X_{inlet\_valve}}(s) \right\}_{P_{chamber}=c}} = \frac{G_{11}}{G_{11} - \dfrac{G_{12}G_{21}}{G_{22}}} = \lambda_{22} \tag{7}$$

The magnitude of the RGA-number (defined as $1-\lambda_{11}$ for this system) is a frequency-dependent measure of the cross coupling inherent to the system. An RGA-number of 0 indicates that the system is completely decoupled. The sign of the RGA number is also important. For example, a negative RGA-number indicates that closing one loop will increase the open loop gain (i.e., destabilize) in the other, whereas, a positive RGA-number indicates that closing a loop reduces the gain (i.e., reduces bandwidth).

Figure 5 shows RGA-numbers for the facility model at low, mid, and high power engine operations, for a trimmed flight condition of 38K/0.65. As expected from the above discussion, the coupling is extremely small at mid and high power. Even at low power, however, the coupling is still small, particularly at the target closed loop bandwidths (5 to 15 rad/sec). Accordingly, treating the two control channels as decoupled for the purpose of loop tuning should have a minimal impact on stability robustness.

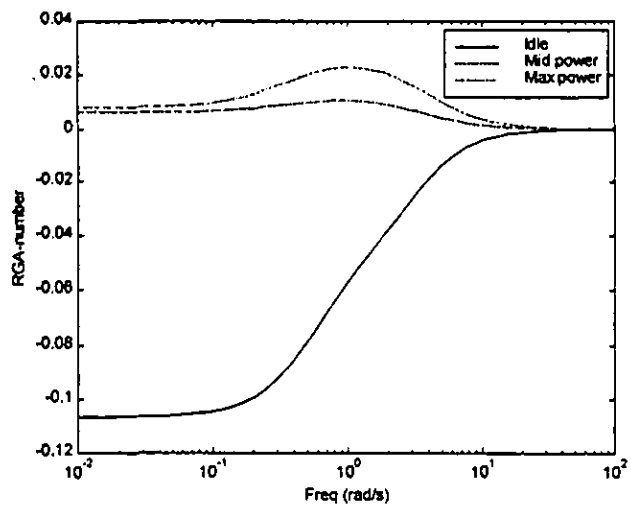

Figure 5. RGA Coupling Analysis

One caveat to note from Equation 7 is that one-way coupling (G_{12} or $G_{21} = 0$) results in a decoupled system from an RGA-number perspective; however, the system is clearly coupled. This apparent contradiction results because the RGA measures the impact of coupling on stability robustness, and one-way coupling does not destabilize the system, since no additional feedback path is formed. However, it does not imply that multivariable control or a decoupling feedforward path between channels would not improve control performance.

It is well known within the control community that controller designs can be significantly simplified by augmenting the feedback structure with a feedforward path (see Astrom and Wittenmark – 1990 or Franklin, et al – 1990). This particular two-degree-of-freedom controller architecture can be exploited if the primary system disturbance is measurable. In many applications, feedforward control is an alternative approach to integral control. This is particularly true in applications where high bandwidth requirements make integral control design very difficult. If the designer chooses to use both feedforward and integral control in tandem, then the demands on the integral controller are significantly alleviated due to a reduction in the steady-state error that the integral controller must eliminate.

In our application, engine transients represent the primary disturbance to the facility control system. Large rate changes in the fan rotor speed indicate that an engine transient is in progress. The rotor shaft speed is denoted by NL and is included in the facility instrumentation package. The Auto-Tune controller uses corrected NL measurements to schedule plenum and chamber valve positions. The performance objective of the feedforward controller is to position these valves so that the altitude/Mach errors are nulled. In reality, an open-loop approach can never completely satisfy this performance objective. Hence, the PID controller reacts to the remaining residual error and achieves the steady-state null objective. This coarse/fine control strategy significantly increases the effective bandwidth of the closed-loop system and reduces the work load of the PID controller.

A simplistic approach to developing a feedforward controller for this application is to employ a feedback controller to position the inlet and exhaust valves at a series of steady state points or during a quasi-steady state transient that covers the engine power range. Throughout this experiment, the facility control system maintains the desired altitude and Mach number. At each point, the valve positions and the corrected fan speed are recorded. These valve positions can then be accessed as a function of corrected fan speed by a table lookup algorithm. The discussion in the sequel illustrates how this approach must be modified to implement a viable feedforward control system in the target application.

It was initially postulated that positioning the inlet and exhaust valves to their steady state positions, solely as a function of fan speed, would provide perfect control of plenum and chamber pressures in the presence of perfect measurements and actuation. Contradictions to this conjecture were discovered during the initial evaluation of the controller. The reason for this contradiction becomes clearer upon inspection of the system model in Figure 2.

The venturis upstream of the plenum are typically choked, i.e., exit mass flow is independent of back pressure (plenum pressure, in this case) and proportional to venturi tube pressure. Maintaining the plenum pressure at its setpoint requires that the plenum volume mass balance be preserved. This implies that the venturi exit flow must equal the engine inlet airflow. The venturi tube pressure is therefore (ideally) proportional to engine inlet flow, and must accordingly vary with engine power. Since the venturi tube volume creates a lag from inlet valve position to venturi tube pressure, a lead term can be used to dynamically compensate the inlet valve position, thereby improving feedforward control performance. Moreover, the venturi tube volume lag varies with inlet valve position; hence, the coefficients of the lead compensator are scheduled as a function of the current inlet valve position.

Similarly, in the exhaust valve channel, the need to accommodate varying airflow rates implies that the exhaust duct pressure must vary with engine power level. However, dynamic lead compensation is not effective in removing the transient lag in the exhaust duct pressure induced by the volume dynamics due to the presence of significant nonlinearities in this subsystem. Instead, a scheduled offset to the exhaust valve position from its steady state value is used. The magnitude and sign of the offset reflect the variation in exhaust duct pressure relative to its corresponding steady-state value. The exhaust channel feedforward control therefore schedules the valve position with a bivariate map. The independent variables in the map are corrected fan speed and exhaust duct pressure.

Auto-Tuning Concept

The fundamental idea underlying the auto-tuning concept is that relay feedback can induce a stable limit-cycle in the inlet and exhaust channels of this system. A stable limit cycle produces periodic signals within each control loop. The error signals of interest are the loop error signal and the relay output signal. By processing these signals, we can reliably extract estimates of the ultimate gain and ultimate frequency required by the Ziegler-Nichols PID gain formulas. This section examines the signal processing requirements and relay model parameters needed to implement the identification algorithm in a realistic environment.

To tune the exhaust channel PID controller, the operator issues the appropriate tuning command. The system responds to this request by disabling the PID controllers in both channels and bringing on-line the relay and system identification software modules. The input to the relay block is the noise corrupted, loop error signal. The random fluctuations in the loop error signal must be addressed in the relay model to avoid inadvertent switches in the output level. One approach for accommodating the noise is to add a hysteresis band to the relay model. A rule-of-thumb used to determine this parameter requires knowledge of the measurement noise level. The noise level present in the measured signal can be extracted directly from the data with a sample variance estimator. Given this estimate, the system sets the relay hysteresis width to an integer multiple of the noise level. The integer value depends upon the magnitude of the variance estimate, since a large hysteresis band can adversely affect the tuned controller's performance. An elaboration of this point is discussed at the conclusion of this section.

The output level of the relay module also affects the accuracy of the ultimate gain estimate. An assumption made in formulating the auto-tune concept is that the relay block represents the only nonlinearity in the control loop. In this application, other nonlinearities exist in the actuator and facility systems. To insure the validity of the linearity assumption, the output of the relay block needs to be large enough to avoid the backlash band in the actuator, but small enough to avoid large-signal nonlinearities in the loop. If the designer has a priori information about the valve backlash width, then the operator can manually adjust the relay output level to avoid this nonlinearity. However, an automatic gain adjustment that uses a priori information as an initial condition would be much more desirable. Note that the relay output level also affects the influence of hysteresis in the ultimate gain calculation. Hence, this factor also needs to be considered by an automatic gain adjustment algorithm.

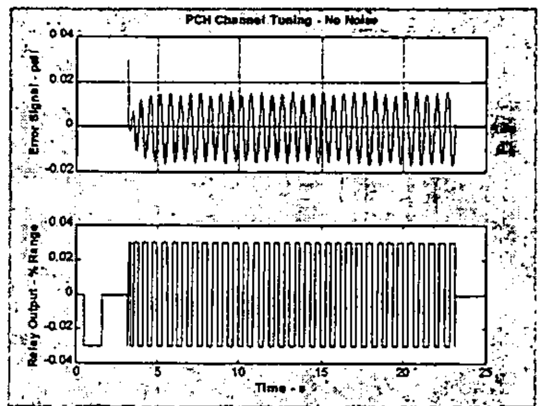

Figure 6a. Auto-Tuning Without Noise

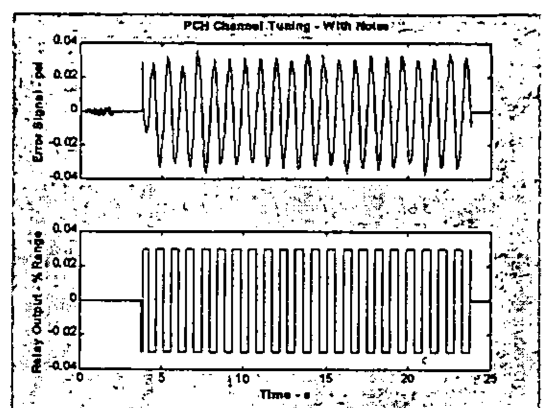

Figure 6b. Auto-Tuning With Noise

Figure 6 shows sample traces of loop error and the relay output for a simulated exhaust-channel tuning run at 38,000 feet and a Mach number of 0.65 (38kft/0.65). Figure 6a presents results for an ideal relay in an absence of measurement noise. The companion figure, Figure 6b, presents the more realistic scenario where measurement noise is present and the relay model has been augmented with a hysteresis band. In both cases, two pulses (0.03 psi in magnitude) that additively perturb the chamber pressure set point induce the limit-cycle. The noise level on the sensed chamber pressure feedback signal was set at 0.2 percent of the set-point value. A relay using a hysteresis band set at the one-sigma level produced the traces that appear in Figure 6b. Note that the limit cycle develops very rapidly and the frequency of the periodic square wave generated by the relay has minimal jitter, i.e., the signal has a stable period. The system identification module processes the loop error and relay output signals to estimate the ultimate gain and ultimate frequency. These signal parameters are then used to compute a new set of gains for the PID controller. Once the system completes the PID tuning in both channels, the system reconfigures into a control mode and commissions the PID controllers.

The objective of comparing these automatic tuning results for relay experiments with and without noise is to illustrate this effect on the estimates of ultimate gain and frequency. From Table I, it is obvious that these estimates directly affect the PID controller parameters, and hence controller performance. The relay hysteresis band that accommodates the measurement noise modifies the describing function analysis discussed previously. The negative inverse of the describing function for a relay with hysteresis is given by

$$-\frac{1}{N(a)} = -\frac{\pi}{4d}\sqrt{a^2 - \varepsilon^2} - i\frac{\pi\varepsilon}{4d} \qquad (8)$$

where a is the relay input signal, d is the relay amplitude, and ε is the hysteresis half-width. Note that the describing function, N(a) is complex when the hysteresis is added to the relay nonlinearity; the *memory* associated with the hysteresis results in a phase shift. The plot of $-1/N(a)$ on the Nyquist diagram no longer resides on the negative real axis as it did in the ideal real relay case. Instead, this plot parallels the real axis but is offset by ($-\pi\varepsilon/4d$). Hence the intersection of the describing function and the Nyquist plot of the plant occurs at a lower frequency, i.e., the ultimate frequency and gain will be reduced. Comparison of Figures 6a and 6b graphically demonstrate the reduction in ultimate frequency. The resulting Ziegler-Nichols tuning relationships produce a controller that reflects this via reduced proportional and derivative gains, and the closed-loop disturbance rejection performance deteriorates.

Note that the describing function shift off the real axis is a function of the hysteresis half-width to relay output amplitude (ε/d) ratio. If the hysteresis band is small relative to the relay output level, then this effect is marginal. An automatic adjustment of relay amplitude (d) to a level that minimizes this ratio is one way of coping with this problem. However, the designer needs to be careful in limiting the magnitude of d, since large values of d will invalidate the linearization assumption used in the relay experiment.

Automatic Tuning Procedure

The automatic tuning procedure implements the concepts discussed in the previous sections. So far, the focal point of the discussion has been the tuning of the feedback, PID controllers. In this section, we address the feedforward tuning requirements and put forth a candidate algorithm to implement these requirements. From an overview perspective, the target application permits the partitioning of the tuning problem into two distinct tasks. In the first task, the tuning algorithm addresses the creation of the feedforward schedules that correlate valve positions with corrected fan speed. This tuning task precedes the relay experiments performed to tune the inlet and exhaust loop PID controllers. The rationale for approaching the problem in this manner is due to the fact that the feedforward controllers can maintain the chamber and plenum pressures near the prescribed setpoints. This feature insures the maintenance of the linearity assumption inherent in the relay tuning experiments. Moreover, the feedforward controllers protect the system against engine-induced disturbances during PID tuning. This protection is needed because both PID controllers are disabled during the relay experiments.

The relay tuning experiments are conducted with the independent, single-relay feedback approach (Wang et al – 1997). With this approach, only one loop at a time is subject to relay feedback while the other is kept open. In the altitude test facility application, the chamber pressure/exhaust valve control loop is done first. This design decision reflects the fact that perturbations to the chamber pressure set point do not disturb the plenum pressure. However, plenum pressure set point perturbations couple into the chamber pressure, but this interaction does not adversely affect system stability as shown in the Controller Architecture section. Hence, a multivariable relay tuning experiment and its attendant complexity are not required in this application.

A Supervisory Control subsystem provides the auto-tune system with an operator interface and the reconfiguration logic that supports the multiple modes inherent to this system. A functional overview of this subsystem and the tasks it performs are discussed in the sequel. The Supervisory Control module plays a critical role in the implementation of the auto-tune control system.

Supervisory Control

The Supervisory Control subsystem supports the multiple operational modes inherent in an auto-tuning controller implementation. This subsystem performs the following functions:

- Operator Interface,
- System Reconfiguration ,
- Event Sequence Generation,
- Feedforward Tuning.

The discussion in the sequel describes how these functions were implemented in the facility control application.

The operator interacts with the auto-tune system through the Supervisory Control. The operator has the option of selecting one of five system modes. These modes provide the resources for performing PID tuning, feedforward tuning, control system engagement, and forcing the auto-tune control system to an idle state. The operator has the option of manually sequencing the system through a procedure or can issue a command that performs a sequence of tasks in a scripted manner. In either case, an Event Sequencer initiates and controls the auto-tune moding commands issued by the operator. The Event Sequencer is implemented as a finite state machine with state transitions governed by operator commands or elapsed time. This event-driven implementation accommodates the insertion of operator override commands that interrupt the normal flow of a task and force the transition of the system to a *safe* state.

The operator interface also includes the capability for archiving PID and feedforward tuning gains for future tests at the same flight condition. If archival gains are available for a particular flight condition, the operator loads the gains and engages the system. This feature eliminates the cost of retuning the system and improves the repeatability of a test scenario. The operator can also provide key parameters for the tuning process. These parameters include relay amplitude, hysteresis width and a set of *soft* PID gains for use by the system prior to the completion of PID tuning. The *soft* PID gains are particularly useful for controlling the facility while feedforward tuning is in progress. These gains are not intended for use during engine operability tests, but will maintain system stability at all points in the flight envelope under benign conditions.

The Reconfiguration Logic responds to operator and Event Sequence Generator requests to reconfigure the system so that the desired mode of operation can be supported. For example, if the operator wants to perform feedforward tuning, the Reconfiguration Logic issues the appropriate moding flags (feedforward acquisition and PID engagement), directs the PID controllers to use the *soft* PID gains, and routes the operator-specified data to the tuning algorithm. The details of the feedforward tuning algorithm will be discussed in the next section. The Reconfiguration Logic also performs the PID-gain archiving function. At the completion of a relay experiment, the PID gains for the inlet or exhaust channel are transferred to the operator console. Saving the gain data for future use is then left to the operator's discretion.

Feedforward Tuning

The objective of the feedforward-tuning task is to generate a set of tables (four in total) that enable the system to schedule inlet and exhaust valve positions as a function of corrected fan speed. In order to generate this data, the engine must be slowly accelerated from an idle condition to maximum power, and then decelerated back to idle. During the transient, a data acquisition module collects fan speed, inlet base valve position, exhaust valve position, and exhaust pressure duct measurements. A data processing algorithm divides the engine's operational speed line into discrete increments, i.e., bins, and maintains a cumulative sum for each sensor. At the completion of the engine transient, the operator puts the system into a wait state and the collected feedforward tuning data is processed. The four feedforward data tables are computed by averaging each set of measurements over three or more bins. The averaging operation mitigates the effects of both sensor noise and actuator hysteresis incurred during the engine acceleration/deceleration transient.

While the feedforward tuning operation is in progress, the Reconfiguration Logic engages the PID controllers in both the inlet and exhaust channels. Using the *soft* PID gain set, these controllers maintain stable operation of the facility during the tuning operation. The operator terminates the feedforward tuning procedure by selecting the Hold mode that transitions the system into a wait-state. As the system transitions to the wait-state, the feedforward tables are updated and a copy is sent to the operator console for archiving. The auto-tune feedforward controllers begin using the updated tables when the operator selects the engagement mode. Upon re-engagement, the PID controllers continue to use the generic *soft* PID gains. However, the system is now ready to begin the relay experiments that will the tune the PID controllers in both the inlet and exhaust channels.

The feedforward tuning experiment requires approximately four minutes of facility time to complete. Though the cost of extracting this information is nontrivial, it is important to note that the tuning experiment can be incorporated into a standard checkout procedure. Each time the simulated flight condition changes, the operator must demonstrate that the facility is capable of supporting the airflow requirements over the entire power range of the engine. To validate this capability, a slow acceleration/deceleration transient is executed. During this validation run, the autotune control system collects the feedforward tuning data.

Hence, there is no additional facility overhead cost associated with this procedure.

PID Tuning

At the completion of the feedforward tuning procedure, the engine is at an idle power setting. This power setting is not conducive for performing the relay experiments because many of the engine bleed valves will be fully open in this power range. Moreover, the RGA analysis indicates adverse channel coupling at this power setting. Hence, the engine must transition to a more desirable tuning point, which in this case is in the neighborhood of forty percent of max power. The operator conducts this transition by engaging the control system, and then slowly throttling the engine to the prescribed power condition.

At this point in the procedure the operator has a choice of manually tuning each control channel or selecting a scripted procedure that performs the relay experiments automatically. In either case the steps in the procedure are exactly the same and the operator can interrupt the process at his/her discretion. Since the primary subject of this paper is adaptive control, we will focus our attention on the automatic tuning procedure. In this case the operator selects the auto mode and the Reconfiguration Logic responds by disengaging both PID controllers, and initializing the relay experiment in the exhaust control channel. The initialization process inserts a transient perturbation on the chamber pressure set point. The magnitude of the offset is equal to the relay output level. The set point perturbation initiates the relay limit-cycle oscillation, which continues for twenty seconds. At the end of the relay experiment, the system automatically transitions into a wait state. This system mode transition triggers the calculation of the PID gains for the exhaust channel using the ultimate gain and ultimate frequency estimates produced during the experiment. The control system PID gains are updated and a copy is sent to the operator console for archiving. While the system is in a wait-state, the exhaust valve is automatically restored to its pre-tuning position. At the completion of the valve restoration procedure (a 10-second interval) the inlet tuning experiment begins. Procedurally, the inlet tuning experiment mirrors its predecessor. After the inlet valve has been restored, the PID controllers in both channels are actively engaged and the system is ready for conducting engine tests at the selected flight condition.

The entire PID tuning process requires one minute of facility time to complete. This compares very favorably with the one to two hours required for manually tuning the system for a PI controller. Part of the time penalty incurred in the manual tuning approach was attributed to multiple tuning experiments conducted at various points through the engine power range. In examining this problem with the facility simulation, we found that the optimal tuning location was in the forty to fifty percent power range. Given these results, the auto-tune system only performs the relay experiment at one power condition. The results presented in the final section of the paper support this approach to the tuning problem.

Performance Evaluation

The design specifications imposed on the automatic tuning control system required that the simulated flight condition be held to within ±200 feet and ±0.01 Mach number during an engine test. The degree of difficulty in meeting these specifications becomes apparent when considering the operability tests used to validate/verify an aircraft gas turbine. Operability tests imply rapid, large power excursion transients designed to stress all of the gas turbine components and push the engine towards its stability limits. Moreover, these operability tests need to be performed at points in the flight envelope that will challenge the facility control system's capability to maintain the test point.

To demonstrate the disturbance rejection performance of the proposed auto-tune system, we shall present results that illustrate the capability of the system to deal with large transients at three key flight conditions. Figure 7 presents the time history of the engine transient used throughout this study. This transient is characterized by a rapid deceleration from maximum power to idle, a short dwell time at idle, and then a rapid acceleration back to max power. This type of transient is often generically referred to as a *Bodie*. The first forty seconds of the transient shown in Figure 7 simulate the completion of the PID tuning process and the slow transition of the engine from the tuning power condition to max power. The total transient time history occurs over a 100-second interval.

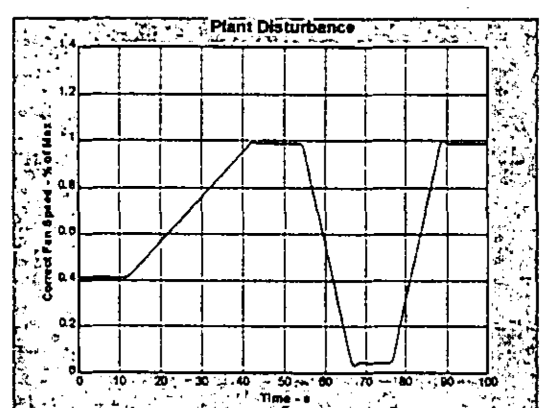

Figure 7. Operability Test Transient

Three flight conditions were chosen as candidate evaluation points. These flight conditions include points at 38kft/0.65, 43kft/0.7, and 38kft/0.35, representing a mixture of flight envelope corners and conditions that present difficult operability hurdles. From a control algorithm verification/validation perspective, demonstration of good performance at these points was essential for proving the capability of the auto-tune concept in this application.

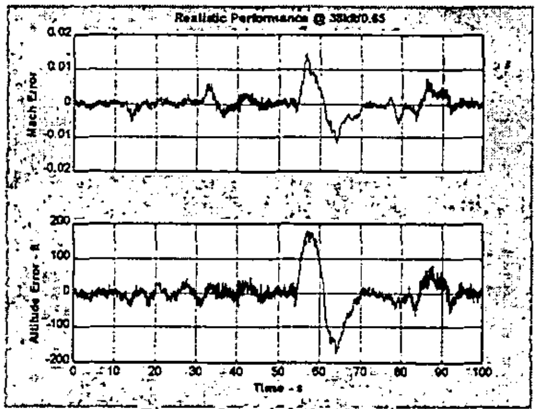

Figure 8. 38kft/0.65 Point

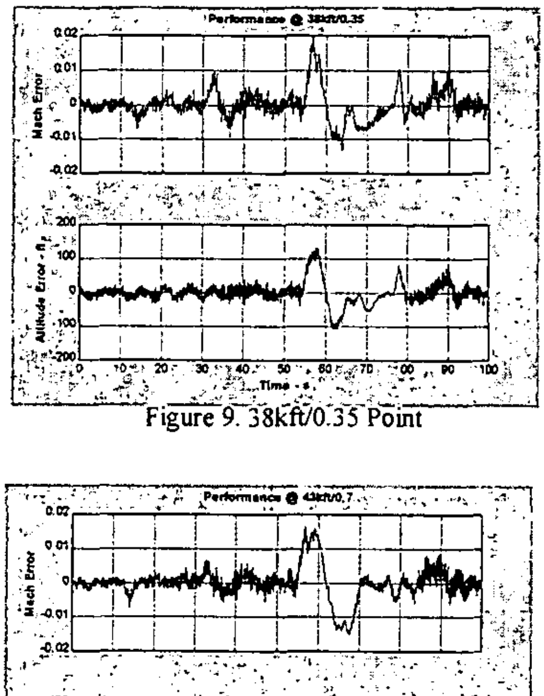

Figure 9. 38kft/0.35 Point

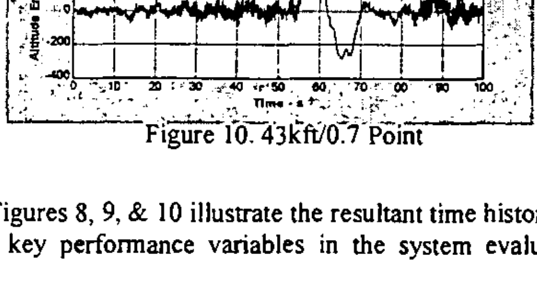

Figure 10. 43kft/0.7 Point

Figures 8, 9, & 10 illustrate the resultant time histories of the key performance variables in the system evaluation, i.e., Mach error, and altitude error. The definition of error in both cases is the reference value minus the actual value. In two of the three cases (see Figure 8 & 9) the auto-tune system meets the altitude performance spec. In the 43kft/0.7 case, the controller could not maintain the desired set-point regulation requirement during the initial deceleration transient (see the 60 to 70 second interval in Figure 10). A first order error sensitivity analysis shows that the altitude error is directly proportional to the relative chamber pressure error, $\Delta PCH/PCH_{ref}$, where ΔPCH represents the deviation of the chamber pressure from the set point PCH_{ref}. Note that plenum pressure error does not affect the altitude error. The chamber pressure error plot in Figure 12 shows the magnitude of ΔPCH in this test case. Comparison of this plot with the chamber pressure error for the 38kft/0.35 case where the maximum altitude error was less than 150 feet (see Figure 11) supports the first-order altitude sensitivity analysis.

The Mach number regulation spec proved to be a much more elusive performance figure. One of the reasons for the higher degree of difficulty is that the facility Mach number depends on both chamber and plenum pressures. In other words, the Mach number control problem is inherently multivariable in nature. From the three test cases presented in the paper, the worst case Mach error occurs at the 38kft/0.35 flight condition. A first order error sensitivity analysis shows that Mach error is approximately equal to

$$\Delta Mach \cong \frac{0.7143}{Mach_{ref}}\left[\left(\frac{\Delta PCH}{PCH_{ref}}\right)-\left(\frac{\Delta PPL}{PPL_{ref}}\right)\right] \quad (9)$$

Obviously, Mach error depends directly upon the relative error in chamber and plenum pressures, but it also depends upon the Mach condition being simulated, i.e., $Mach_{ref}$. At low $Mach_{ref}$, errors in chamber an plenum pressures induce larger errors in Mach. In the test cases presented, the auto-tune control system very effectively regulates plenum pressure, and therefore, plenum pressure error does not significantly affect the Mach performance results. Hence, the error plots presented in Figures 11 and 12 are dominated by the chamber pressure error.

Now let's consider the conjecture that Mach error sensitivity to perturbations in chamber pressure is more pronounced at low Mach conditions. This fact is supported by the data displayed in Figures 11 and 12. The chamber pressure error in the 43kft/0.7 case is 1.5 times larger than that in the 38kft/0.35 case, but the Mach error is greater at the 38kft/0.35 simulated flight condition.

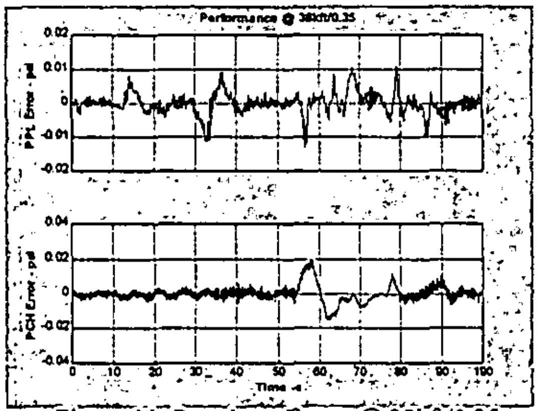

Figure 11. Regulator Errors @ 38kft/0.35

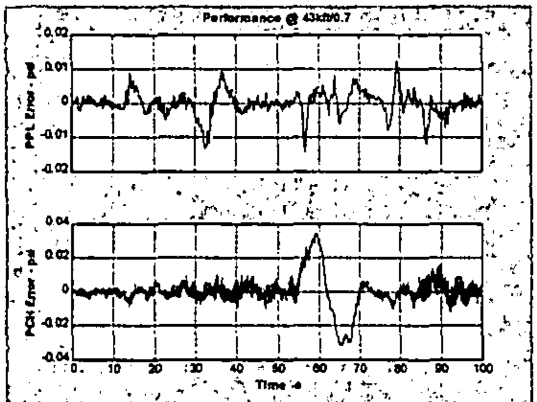

Figure 12. Regulator Errors @ 43kft/0.7

Conclusions

This paper reviews the auto-tuning concept and applies it to a simulated altitude facility control problem. The application of an adaptive control concept to this particular problem was motivated by economic and performance considerations. Manual tuning of the facility control system was an expensive, time consuming task, and the resulting control system performance was significantly less than desired. The implementation of the auto-tune concept described in the paper produced an adaptive, two-degree-of-freedom control system with very good disturbance rejection capabilities. Practical considerations for implementing the auto-tune relay experiments and modifications to the Ziegler-Nichols PID relationships were discussed in detail. The man-machine interface used in the implementation gives the operator the ability to interact with the control system. This interactive capability not only enhances the operational features of the system, but can also be used for operator training and system maintenance. Disturbance rejection capabilities of the system were demonstrated at three flight conditions for a *Bodie* transient. In all three cases, the control system displayed good performance figures under transient conditions. In two of the three cases the control system regulated altitude error to less than 200-feet. Meeting the 0.01 Mach error bogie proved to be more elusive; however, the worst case error performance did not exceed the 0.02 figure in these test cases. Improvement in Mach regulation performance requires either a more complex control system or renovations to the test facility that upgrade the actuation and sensing capabilities.

References

Astrom, K.J., (1996): "Tuning and Adaptation", *Proceedings of the 13th Triennial World Congress*, San Francisco, USA, pp. 1-18.

Astrom, K.J. and Wittenmark B., (1989): *Adaptive Control*, K.J. Astrom & B. Wittenmark, Addison-Wesley, pp.328-341.

Astrom, K.J. and Wittenmark, B. (1990): *Computer-Controlled Systems: Theory and Design 2nd Edition*, Prentice-Hall, pp.199-200.

Astrom, K.J. (1991): "Oscillations in Systems with Relay Feedback", *Adaptive Control, Filtering, and Signal Processing*, K.J. Astrom, G.C. Goodwin, P.R. Kumar – Editors, Springer-Verlag, pp. 1-26.

Astrom, K.J. and Hagglund, T. (1995): *PID Controllers: Theory, Design, and Tuning - 2nd Edition*, Instrument Society of America, pp. 214-217.

Franklin, G.F., Powell, J.D., and Workman, M.L.(1990): *Digital Control of Dynamic Systems 2nd Edition*, Addison-Wesley, pp. 471-472.

Gelb, A. and VanderVelde, W.E. (1968): *Multiple-Input Describing Functions and Nonlinear System Design*, McGraw-Hill, pp. 297-364.

Lundh, M. and Astrom, K.J. (1994): "Automatic Initialization of a Robust Self-Tuning Controller", *Automatica*, Vol. 30, No. 11, pp. 1649-1662.

Pessen, B.W. (1954): "How to 'tune in' a three mode controller", *Instrumentation*, Second Quarter, pp. 29-32.

Sastry, S. and Bodson, M. (1989): *Adaptive Control: Stability, Convergence, and Robustness*, Prentice-Hall, pp. 1-16.

Skogestad, S. and Postlethwaite, I. (1996): *Multivariable Feedback Control: Analysis and Design*, John Wiley and Sons, pp.88-91.

Wang, Q-G, Zou, B., Lee, T-H, Bi, Q. (1997): "Autotuning of Multivariable PID Controllers from Decentralized Relay Feedback", *Automatica*, Vol. 33, No. 3, pp. 319-330