NASA Technical Memorandum 82600

(NASA-TM-82600) A NONLINEAR PROPULSION SYSTEM SIMULATION TEHNIQUE FOR PILOTED SIMULATORS (NASA) 14 p HC A02/MF A01 CSCL 21E
N81-23085
Unclas
G3/07 42270

A Nonlinear Propulsion System Simulation Technique for Piloted Simulators

James R. Mihaloew
Lewis Research Center
Cleveland, Ohio

Prepared for the
Twelfth Annual Pittsburgh Conference on Modelling and Simulation cosponsored by the IEEE, ISA, SCS, SMCS
Pittsburgh, Pennsylvania, April 30-May 1, 1981

NASA

A NONLINEAR PROPULSION SYSTEM

SIMULATION TECHNIQUE FOR PILOTED SIMULATORS

James R. Mihaloew
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio

ABSTRACT

In the past, propulsion system simulations used in flight simulators have been extremely simple. This resulted in a loss of simulation realism since significant engine and aircraft interactions were neglected and important internal engine parameters were not computed. More detailed propulsion system simulations are needed to permit evaluations of modern aircraft propulsion systems in a simulated flight environment.

A real time digital simulation technique has been developed which provides the capabilities needed to evaluate propulsion system performance and aircraft system interaction on manned flight simulators. A parameter correlation technique is used with real and pseudo dynamics in a stable integration convergence loop. The technique has been applied to a multivariable propulsion system for use in a piloted NASA flight simulator program. Cycle time is 2.0 ms on a Univac 1110 computer and 5.7 ms on the simulator computer, a Xerox Sigma 8. The model is stable and accurate with time steps up to 50 ms. The program evaluated the simulation technique and the propulsion system digital control. The simulation technique and model used in that program are described and results from the simulation are presented.

INTRODUCTION

Cost is a major factor in planning experimental ground and flight test programs. As a result, simulations, with their inherent flexibility, are being used to a greater degree to design and analyze aircraft and propulsion systems controls before hardware is built and tested.

Manned flight simulators have been used to evaluate aircraft stability and engine-out performance for various aircraft (1). The propulsion system models used in these simulation studies were extremely simple, however, providing only a thrust signal. This resulted in a loss of simulation realism to the extent that significant engine and aircraft interactions were not possible and important internal engine parameters were not available for analysis.

To overcome these deficiencies, reasonably detailed real time propulsion simulations are needed. Such simulations will provide the capability to evaluate propulsion systems and their interaction with aircraft controls on manned flight simulators. The goal of the simulation development was to derive such a model and evaluate it in a piloted simulation program (2). This paper describes the development of a real time propulsion system simulation that provides this capability. The modeling technique was developed using the Under-the-Wing (UTW) version of the Quiet Clean Short-Haul Experimental Engine (QCSEE) and evaluated in the STOL aircraft described in (1). The simulation technique and propulsion system model are described and limited results from an evaluation of the simulation are presented and compared with experimental data.

A non-linear propulsion system simulation such as (3) produces a model of high frequency fidelity running slower than real time in an analog format. Extension of this procedural development to hybrid models such as (4) produces a model with reduced high frequency fidelity but capable of just real time. In a digital format as required for piloted simulators these models would require high sampling rates (small time steps) to maintain calculational stability. Real time would be virtually impossible. The general approach taken in the real time digital simulator model development presented here was similar. For the level of steady state and dynamic complexity required to meet this program objective, steady state accuracy does not have to be compromised over detailed models; but, high frequency content must be reduced significantly. The initial improvement in execution time comes from efficient programming to attain minimum calculation time. More rapid improvement occurs by maximizing stability to permit long time steps.

REAL TIME PILOTED SIMULATION REQUIREMENTS

To satisfy the requirements of real time piloted simulation, innovative mathematical modeling is required. One must attain the desired level of fidelity yet have the computations accomplished in a limited amount of time. Also, since the propulsion system is only a part of a larger simulation, only a fraction of the total computation time is available for the propulsion system calculations. Real time, then, in the context of overall simulation requirements, implies that the propulsion simulation must be faster than real time to be effective.

Modeling

The basic technique used in flight simulation is to derive mathematical models that define aircraft system characteristics to the degree necessary to accomplish the objectives. The state of the aircraft system is described by the equations of motion in terms of the aircraft accelerations, velocities and positions. Usually six degrees of motion are considered. The resulting differential equations are then integrated with respect to time to obtain the aircraft states. Resulting kinematic information is then used as input to other simulator sub-systems such as visual motion, force-feel and instrumentation. The simulation equations are usually very complex and are not amenable to analytical solution. Therefore the models must be programmed for numerical solution. Since the aircraft states can change rapidly with time, the computations are made often with small time increments. A fast, accurate and stable integration algorithm is essential.

Real Time Monitoring

The computer cannot meet real time requirements simply by repeating the calculation of the equations as rapidly as possible. This is because the calculation time may be different for each time step. The calculations must be scheduled by a real time monitor to occur within a fixed time interval that is longer than the maximum time needed to solve the equations. Each interval, or frame time, is measured by a real time clock in the computer. The computer is interrupted at the end of each interval and made to recycle through the equations. If the calculations cannot be completed in the specified interval a missed interval will occur and real time operation may not be possible. At the end of each interval, time is incremented by the frame time. This process, as shown at the top of figure 1, satisfies the real time calculation requirement.

The real time requirement introduces a conflict in objectives when selecting the interval length. On the one hand the interval must be minimized to retain higher frequency information in the model. On the other hand, the model's detail and function are improved by adding more equations which in turn increases the calculation time. The conflict can be resolved by using multi-rate scheduling.

Multirate Scheduling

In multirate scheduling, equations are separated into loops according to their frequency content as shown in figure 1. Each loop is run at a multiple of the basic frame time. In the case of three loops, the fast loop (high frequency) might be calculated four times as often as the slow loop (low frequency) and the intermediate loop (middle frequency) twice as often as the slow loop. The ratios can be variable.

While multirate scheduling permits higher frequency content in simulations, innovative modeling techniques are still necessary to reduce computation times to a minimum and maintain simulation stability. The following paragraphs describe these modeling techniques and their application to a state-of-the-art turbofan propulsion system.

PROPULSION SYSTEM APPLICATION

The modeling technique was developed using the Under-the-Wing (UTW) version of the Quiet Clean Short-Haul Experimental Engine (QCSEE) developed for NASA-Lewis by the General Electric Company (5) and shown in figure 2.

The engine is basically an F101 core gas generator with a high bypass fan. It includes a high Mach inlet, a variable pitch fan, a variable geometry fan duct exhaust nozzle and a digital electronic control system combined with a hydromechanical fuel control. The fan is a low pressure ratio, low tip speed configuration with variable pitch blades and is driven by a low pressure turbine through reduction gears.

The fan is capable of blade pitch changes from forward to reverse thrust. The fan pitch actuation and control are designed to move the blades from forward to reverse position in less than one second.

The fan exhaust nozzle is a hydraulically actuated variable area design. It is capable of area change from takeoff to cruise as well as opening to a flare position to form an inlet in the reverse thrust mode.

The control system manipulates four variables to achieve rapid thrust response and noise suppression. Control of engine pressure ratio, fan speed and inlet mach number is accomplished by manipulating fuel flow, fan blade pitch and exhaust nozzle area. Variable stator vanes are scheduled by core speed to attain optimum stall margin.

The digital computer controls the output variables in response to commands from the aircraft. It generates all control laws and logic and most of the limiting functions as well as power management, condition monitoring and failure indication and corrective action. The hydromechanical control provides an electrohydraulic servo fuel valve which is used by the digital control for primary fuel control. It also provides backup fuel control through a core speed controller, acceleration and deceleration limits and primary control of the core compressor stators.

MODELING TECHNIQUE

The analytical model is derived from the real propulsion system. It represents mathematically the fundamental steady state and dynamic relations that exist between the engine components and controls. Engine dynamics are based on the dynamic form of the conservation equations and engine transient experience. Steady state performance is based on component representations derived from engine data. The control model is based on the control system specification.

The form of the engine model and its information flow are shown schematically in figure 3. All major engine components are represented. The level of component detail is approximately the same as for state-of-the-art models.

Since control evaluation is a prime consideration in propulsion system evaluations, a detailed control representation is essential especially in the control law and switching logic areas. The control model is shown in figure 4. Accuracy of the control model was assured by deriving it directly from the control specification diagrams used in the real control.

The dynamic engine simulation is derived by applying the basic conservation equations of continuity, energy and momentum to each component using the steady state component characteristics to define the boundary energy and mass across each boundary. Low frequency dynamics such as rotor speeds and component heat soaks are retained. High frequency dynamics such as volume dynamics are omitted. Algebraic loops that occur from omitted high frequency dynamics are converged using high gain integrators. The resulting dynamic effect is similar to inductive lags. Component maps are generated by correlating input-output parameters to reduce complexity while retaining accuracy. These correlations are then curve-fit using segmented polynomial and geometric functions. The resulting analytical model is a twelfth order model which includes four engine states, four control sensor states and four control states. The resulting set of differential equations are solved as a general initial and boundary value problem using a two-step integration algorithm. The function generation and convergence techniques are innovative methods to accomplish rapid computation and stability.

Function Generation

Engine component performance maps are typically two or three variable functions. A fast, accurate function generation technique is necessary to ensure real-time simulation. In this simulation technique, two approaches are used to achieve fast function generation. The first is to simplify the basic functional relations through parameter correlation and the second is to approximate the functions as polynomial or geometric equations.

This process will be described for the QCSEE variable-pitch fan map. The fan map, as determined from model data, is a function of blade pitch angle, corrected speed and pressure ratio. This function was represented as shown in figure 5. The three-function map was approximated as a two variable basic fan map with multipliers on pressure ratio and corrected flow. The multipliers were functions of the blade pitch angle. The one and two variable functions were represented as multi-segment geometrical and polynomial functions. The functions were segmented to provide better accuracy over a wide range. The criteria used in developing each representation was to derive the relation with the least execution time consistent with a specified accuracy. The calculation time for the fan map function was 85 μs compared to 750μs for a generalized function generator (6). Accuracy relative to the original fan map was about one percent. Other multi-segment polynomial and analytical functions were calculated in about 15 to 30 μs. In this way, all generalized function generation and subroutine calls were eliminated with substantial savings in calculation time.

Convergence Technique

Some form of convergence technique was required to solve the algebraic loops in the simulation. Pure iteration was not used because of its transient dependence. That is, cycle or calculation time per time step varies with the rapidity of the transient. This is to be expected since non-steady mismatches occur during transients and more iterations are required to converge the solution in order to avoid missed intervals. The frame time in piloted simulations which use pure iteration must be based on the maximum calculation time even though the average calculation time may be considerably lower. Calculational efficiency of iteration is therefore poor for real time simulation.

In this simulation the iteration differences associated with algebraic loops were integrated with high integrator gains. A sketch of the process is shown in figure 6. The procedure is similar to using high gain integrators in analog computation to prevent algebraic loops. The previous "guessed" value, PGS, is used to calculate the new required steady state value. The difference between these two values is integrated to generate an updated "guessed" value. The process shares the same integration algorithm used to determine the model states.

Refering to figure 6, the integration convergence method states that convergence takes place through a simple lag. In terms of a transfer function this is::

$$PGS/P = 1/(s/k + 1)$$

where $1/K = \tau$. τ is the system time constant. The smaller the time constant (higher K) the faster the system response and the model approaches steady state convergence more rapidly. From a model convergence viewpoint then, high integrator gains are desireable commensurate with stability requirements.

MODEL EVALUATION

The model was exercised over a range of inputs to evaluate model stability and accuracy. A number of engine transients that might be expected to occur in a simulated flight evaluation were run.

Initially, the simulation was run to determine suitable integrator gains and the effects of frame time on the results. A standard transient was used consisting of: (1) steady state operation at 62.5 percent, (2) a 62.5 to 100 percent power increase, (3) an in-to-reverse transient at 100 percent power and (4) an out-of-reverse transient at 100 percent power. Traces of net thrust and other variables were made to detect transient differences obtained with various combinations of gain and frame time. The traces were compared to those made at a gain of 100 and a frame time of 5 ms. This was done to insure that the data was compared to an accurate dynamic model. Selected results are shown in figure 7.

Transient anomalies were expected with frame times larger than 50 ms since time constants in the engine and control models are higher than 50 ms. Time constants below 50 ms were purposely avoided in the model development. Deterioration beyond 50 ms was probably due to the single pass calculation of time constants in the control. Based on these observations the model was judged to have a useful dynamic capability with frame times less than 50 ms.

Every effort was made to eliminate unecessary calculations in the program. Divide computations were minimized and exponentiation eliminated. The model was programmed on a Univac 1100-40 computer using Fortran V. On that computer the model consumed 2.0 ms per time step. A somewhat simpler reverse model decreased that time to 1.8 ms. Cycle time was not transient sensitive. On the flight simulator computer, a Xerox Sigma 8, cycle time was 5.7 ms or about three times slower.

Transient Performance

A number of transients were run which could be expected to be encountered in a manned flight simulation program to demonstrate the model's capability. A number of simulated engine and control component failures were also programmed into the model to analyze failure modes and effects in the flight program (2). A limited number of these transients are presented here along with corresponding experimental engine data acquired from recent testing at NASA-Lewis.

Figure 8 shows a typical approach transient from approach power at 62.5 percent to go-around power at 100 percent. Shown are net thrust, fan speed and fan pitch angle. These are only a few of the variables available from the model. Superimposed on the model traces are the experimental engine transient data. Discrepancies between these data are attributed mainly to the differences in the fan representation as derived from steady state cycle data and the real experimental fan performance as opposed to model inaccuracies due to model simplification. An updated fan map representation derived from the experimental data would eliminate most of the differences.

The model accurately predicts the fan overspeed on acceleration. The difference in fan pitch angle and thrust are due to differences in the fan pitch-fan flow relationship. The control is manipulating fan pitch to maintain a scheduled fan speed. At the fan power required to maintain this speed, a higher fan pitch angle is required which in turn results in a lower fan flow rate and thrust. The effect is really caused from lower actual fan performance than was predicted in the cycle deck model.

Figure 9 shows the transient response of a simulated aborted takeoff-to-reverse sequence. Again, differences are attributed mainly to the difference in actual fan reverse performance and the reverse map generated from the cycle deck. Of significance here are the predicted thrust peaking on reverse initiation and the predicted stall transition period. The model, however, does not accurately predict the actual fan speed droop during the stall transition to reverse. Again, this is not due to inaccuracies from model simplification, but due to the concept used in modeling stalled fan power consumption.

CONCLUDING REMARKS

The real time digital propulsion system simulation developed under this program has generated a valuable simulation technology and flight simulator experience. It has yielded a feasible real time digital simulation approach. It has provided a fast, accurate and stable model for piloted flight simulation and application to other analytical controls problems. The techniques developed so far have been effective in providing an adequate level of detail to evaluate propulsion systems in a simulated flight environment.

REFERENCES

1. Nieuwenhuijse, A.W., and Franklin, J.A., "A Simulator Investigation of Engine Failure Compensation for Powered-Lift STOL Aircraft", NASA TM X-62363, 1974.

2. Mihaloew, J.R. and Hart, C.E., "Real Time Digital Propulsion System Simulation for Manned Flight Simulators", NASA TM-78958, 1978.

3. Seldner, K., Mihaloew, J.R. and Blaha, R.J., "Generalized Simulation Technique for Turbojet Engine System Analysis", NASA TN D-6610, 1972.

4. Szuch, J.R., Seldner, K., "Real-Time Simulation of F100-PW-100 Turbofan Engine Using the Hybrid Computer", NASA TM X-3261, 1975.

5. General Electric Company, "Quiet Clean Short-Haul Experimental Engine (QCSEE) Under-the-Wing (UTW) Digital Control System Design Report", NASA CR-134920, 1978.

6. Hart, C.E., "Function Generation Subprograms for Use in Digital Simulations", NASA TM X-71526, 1974.

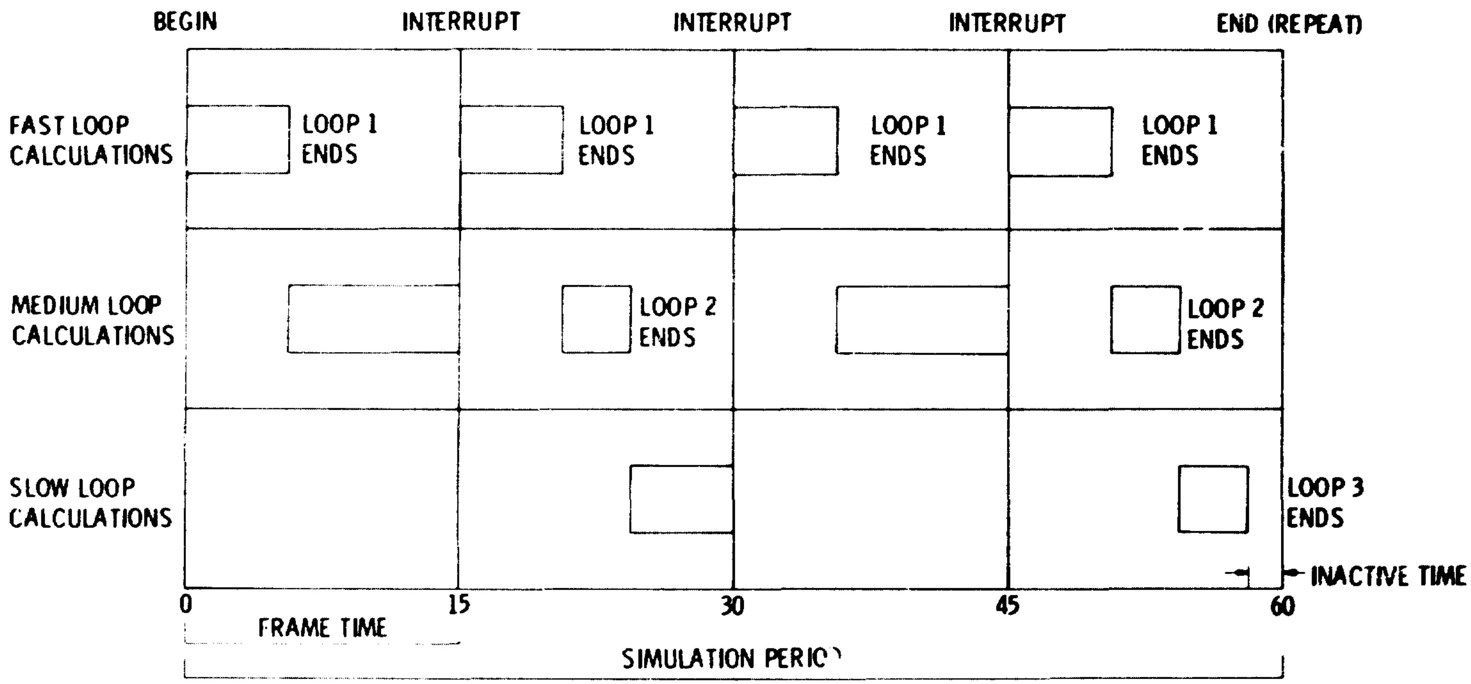

Figure 1. - Multi-rate scheduling of equation loops in 1:2:4 ratio.

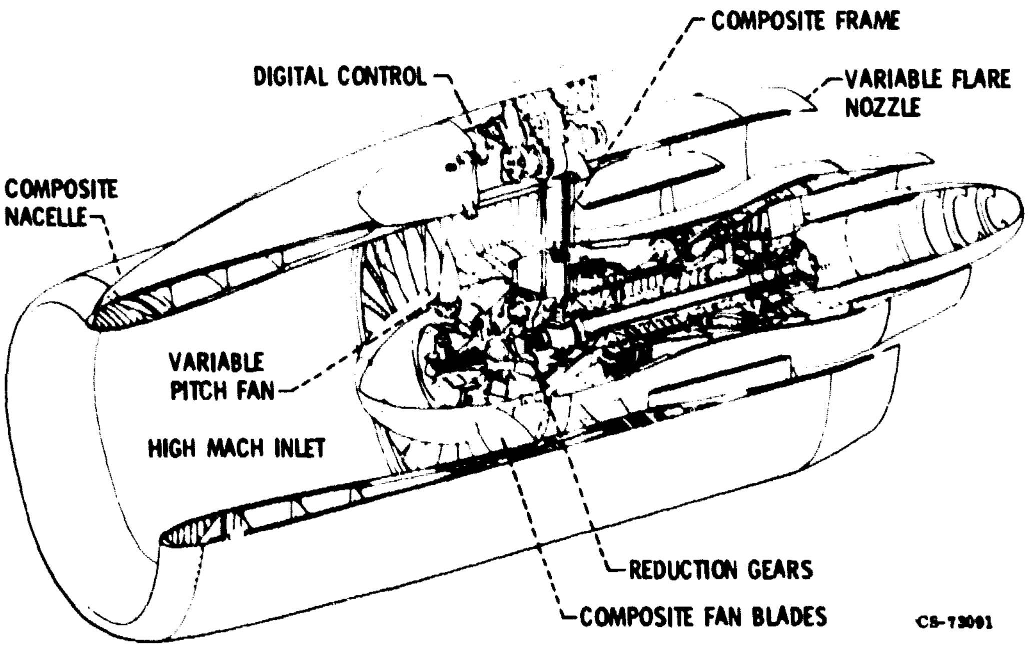

Figure 2. - QCSEE UTW experimental propulsion system.

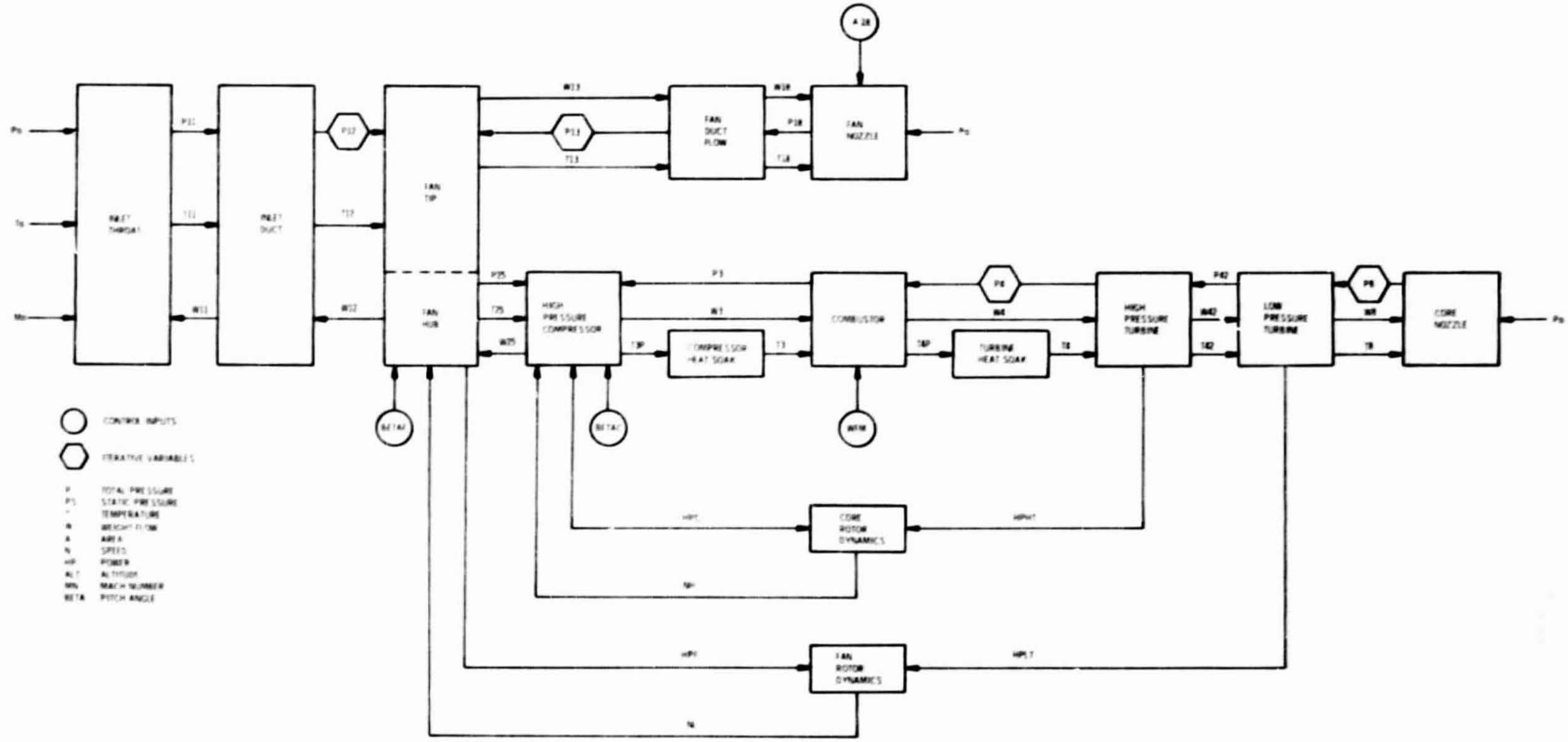

Figure 3. - QCSEE engine model schematic and information flow diagram.

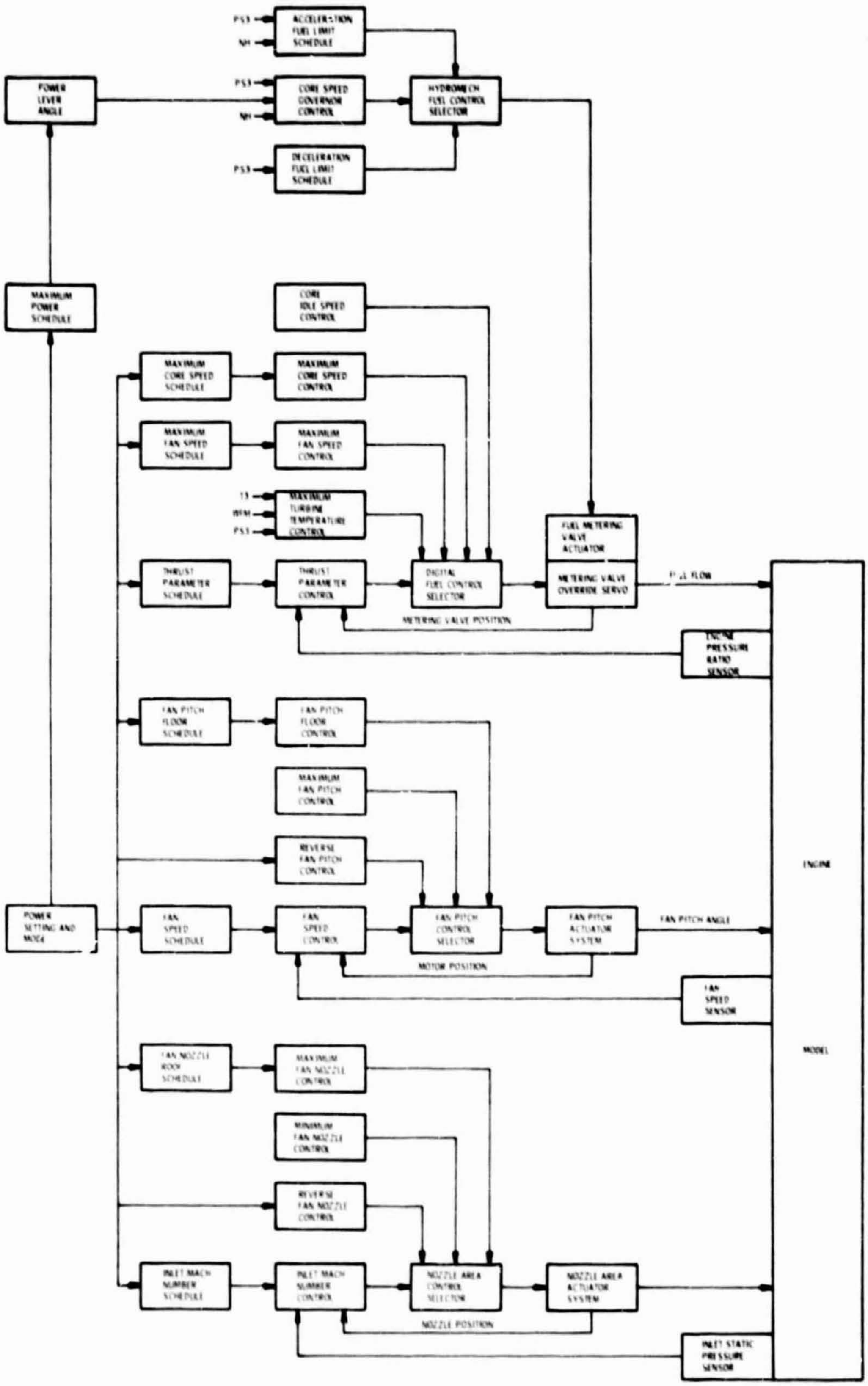

Figure 4. - QCSEE control model schematic.

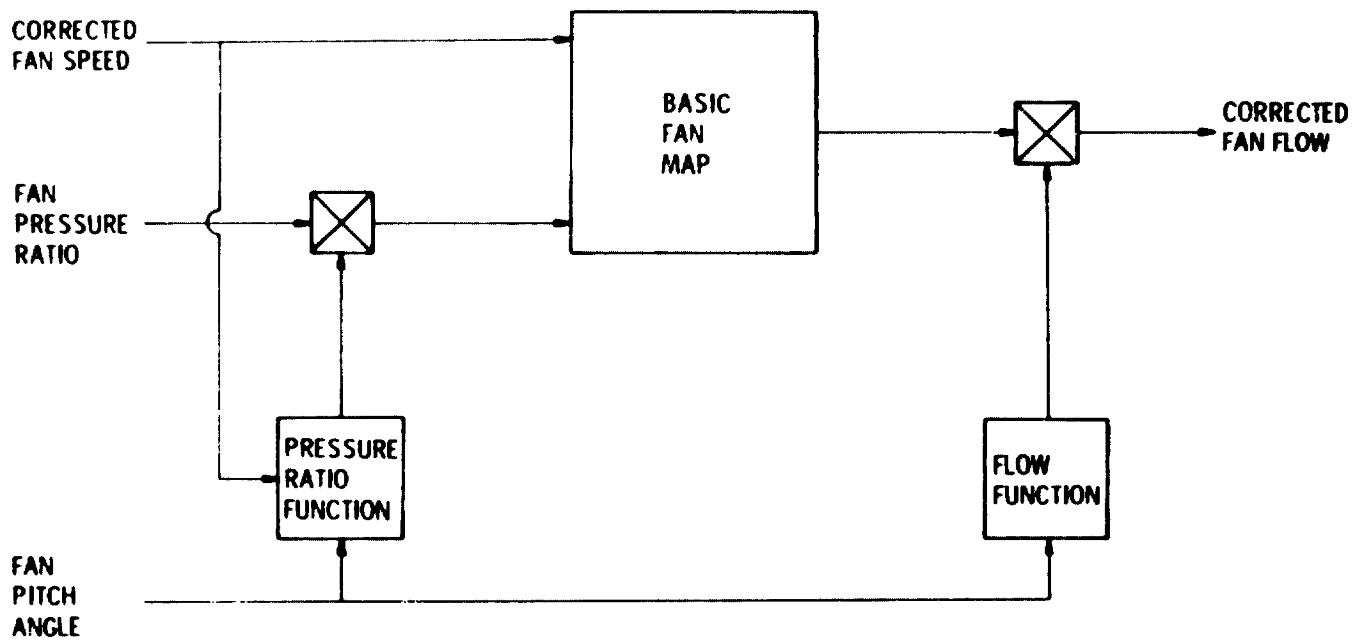

Figure 5. - Variable pitch fan representation and flow diagram.

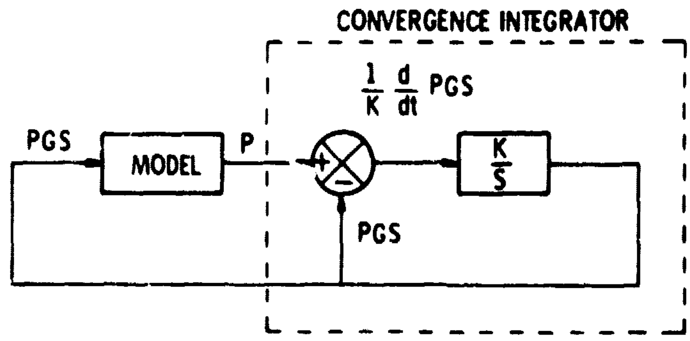

Figure 6. - Integration convergence method.

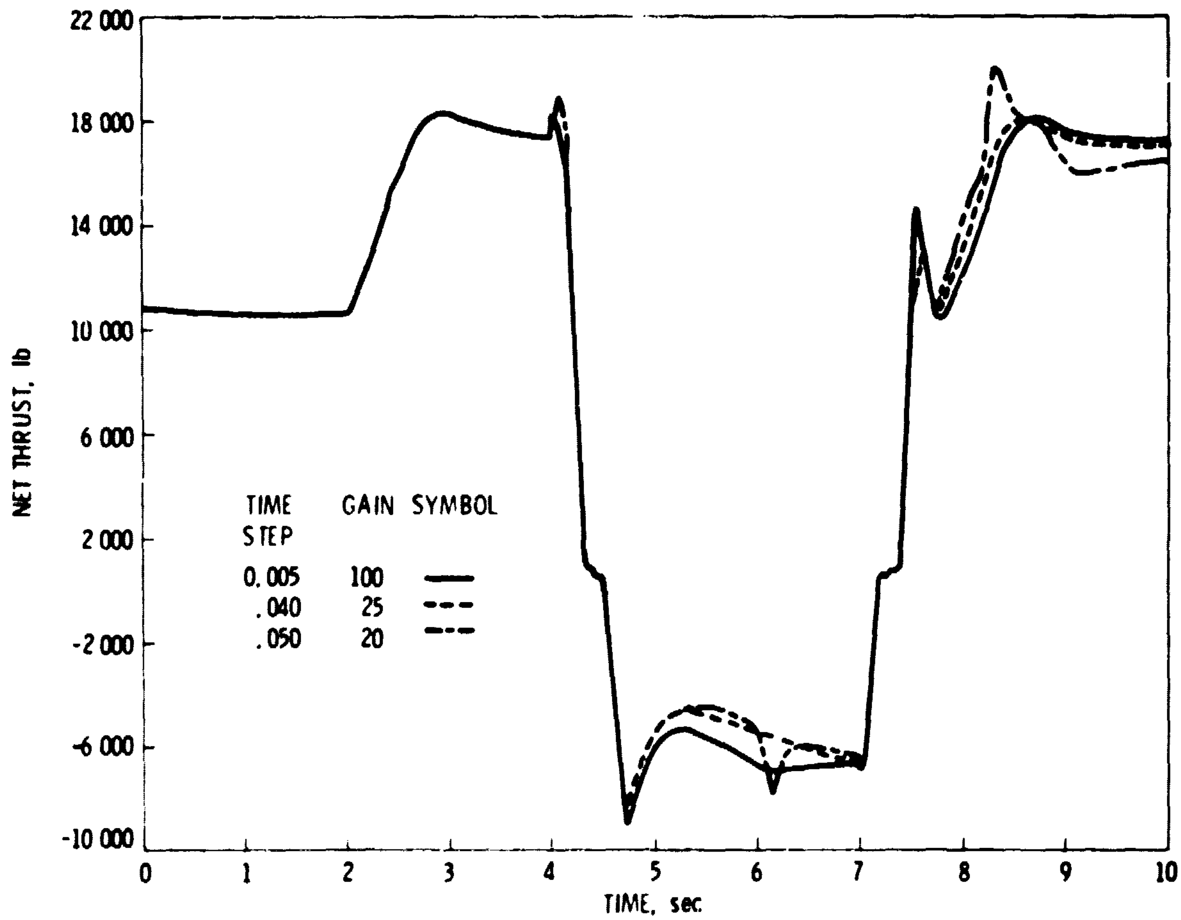

Figure 7. - Comparison of model transients at various time step-gains.

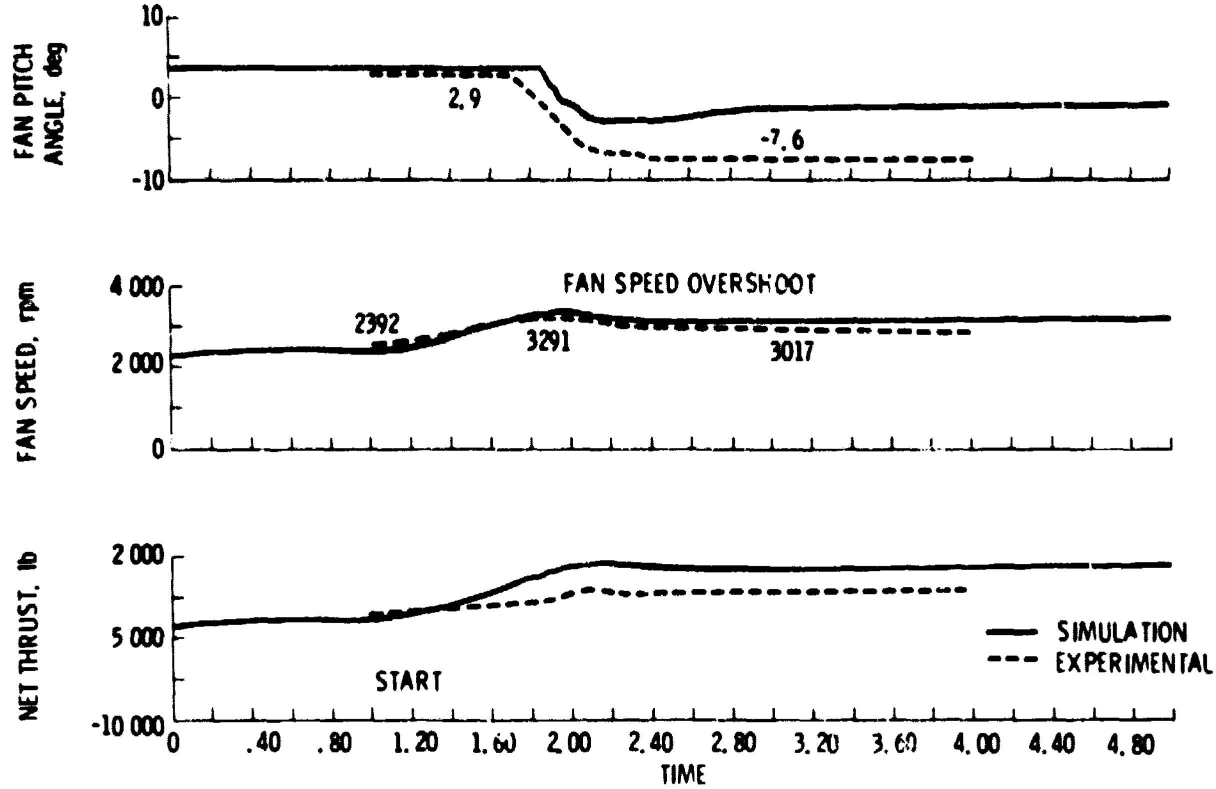

Figure 8. - Forward go-around transient.

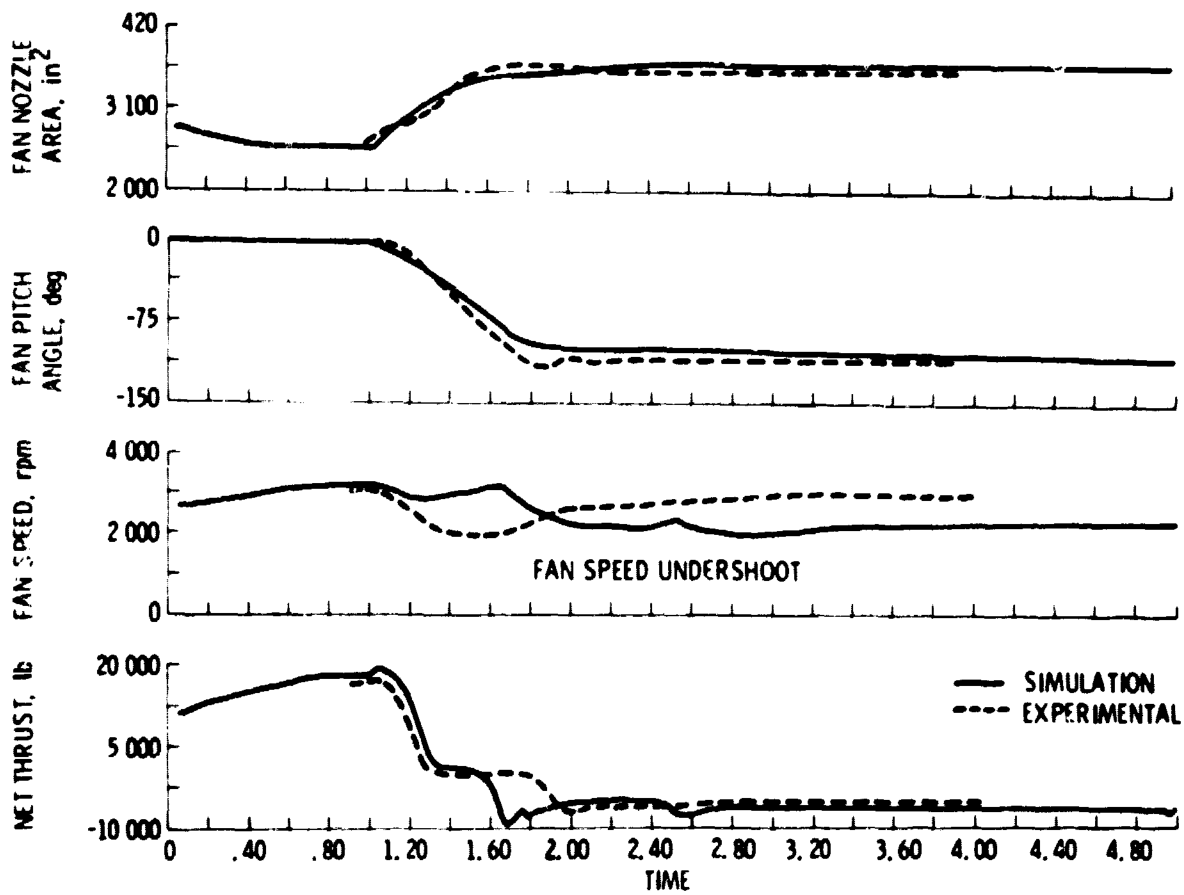

Figure 9. - Aborted takeoff-to-reverse transient.