



A High-Fidelity Simulation of a Generic Commercial Aircraft Engine and Controller

Ryan D. May
ASRC Aerospace Corporation, Cleveland, Ohio

Jeffrey Csank
N&R Engineering and Management Services, Cleveland, Ohio

Thomas M. Lavelle, Jonathan S. Litt, and Ten-Huei Guo
Glenn Research Center, Cleveland, Ohio

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI Help Desk at 443-757-5803
- Telephone the NASA STI Help Desk at 443-757-5802
- Write to:
NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320



A High-Fidelity Simulation of a Generic Commercial Aircraft Engine and Controller

Ryan D. May
ASRC Aerospace Corporation, Cleveland, Ohio

Jeffrey Csank
N&R Engineering and Management Services, Cleveland, Ohio

Thomas M. Lavelle, Jonathan S. Litt, and Ten-Huei Guo
Glenn Research Center, Cleveland, Ohio

Prepared for the
46th Joint Propulsion Conference and Exhibit
cosponsored by the AIAA, ASME, SAE, and ASEE
Nashville, Tennessee, July 25–28, 2010

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

Acknowledgments

The authors would like to thank Diana Drury of ASRC Aerospace Corporation, who handled the version control system for C-MAPSS40k, and T. Shane Sowers, of QinetiQ North America, who integrated C-MAPSS40k into the flight simulator at NASA Glenn Research Center. Additionally, Don Simon of NASA Glenn Research Center provided insight into the development of the steady-state solver and linearization routine. The authors also wish to acknowledge the employees of Pratt & Whitney who graciously reviewed C-MAPSS40k and lent us their expertise, specifically: Juan Marcos, James Fuller, Bruce Wood, and Nicole Romeo. Our thanks go out to those who have helped to test, develop, and debug C-MAPSS40k: at Scientific Monitoring, Inc.: Walt Merrill, George Mink, and Hoang Van Tran; at the University of Connecticut: Dr. Chengyu Cao and Andrew Thompson; at Pratt & Whitney: Kevin Hendricks. Finally, our thanks to Sanjay Garg of NASA Glenn Research Center for his support of the development of C-MAPSS40k, and the NASA Aviation Safety Program's Integrated Resilient Aircraft Control project for funding this work.

Trade names and trademarks are used in this report for identification only. Their usage does not constitute an official endorsement, either expressed or implied, by the National Aeronautics and Space Administration.

Level of Review: This material has been technically reviewed by technical management.

Available from

NASA Center for Aerospace Information
7115 Standard Drive
Hanover, MD 21076-1320

National Technical Information Service
5301 Shawnee Road
Alexandria, VA 22312

Available electronically at <http://gltrs.grc.nasa.gov>

A High-Fidelity Simulation of a Generic Commercial Aircraft Engine and Controller

Ryan D. May
ASRC Aerospace Corporation
Cleveland, Ohio 44135

Jeffrey Csank
N&R Engineering and Management Services
Cleveland, Ohio 44130

Thomas M. Lavelle, Jonathan S. Litt, and Ten-Huei Guo
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract

A new high-fidelity simulation of a generic 40,000 lb thrust class commercial turbofan engine with a representative controller, known as C-MAPSS40k, has been developed. Based on dynamic flight test data of a highly instrumented engine and previous engine simulations developed at NASA Glenn Research Center, this non-proprietary simulation was created especially for use in the development of new engine control strategies. C-MAPSS40k is a highly detailed, component-level engine model written in MATLAB/Simulink (The MathWorks, Inc.). Because the model is built in Simulink, users have the ability to use any of the MATLAB tools for analysis and control system design. The engine components are modeled in C-code, which is then compiled to allow faster-than-real-time execution. The engine controller is based on common industry architecture and techniques to produce realistic closed-loop transient responses while ensuring that no safety or operability limits are violated. A significant feature not found in other non-proprietary models is the inclusion of transient stall margin debits. These debits provide an accurate accounting of the compressor surge margin, which is critical in the design of an engine controller. This paper discusses the development, characteristics, and capabilities of the C-MAPSS40k simulation.

Nomenclature

C-MAPSS	Commercial Modular Aero-Propulsion System Simulation
C-MAPSS40k	Commercial Modular Aero-Propulsion System Simulation 40,000 lbf thrust
FMV	Fuel Metering Valve
EPR	Engine Pressure Ratio (P_{50}/P_2)
GUI	Graphical User Interface
HPC	High Pressure Compressor
HPT	High Pressure Turbine
LPC	Low Pressure Compressor
LPT	Low Pressure Turbine
MAPSS	Modular Aero-Propulsion System Simulation
MEX	MATLAB EXecutable file—compiled binary code
N_c	Core shaft speed (rpm)
N_f	Fan speed (rpm)

PI	Proportional-Integral
P_{sx}	Static Pressure at station X (psig)
P_x	Total Pressure at station X (psig)
PLA	Power Lever Angle (degrees)
PR	Pressure ratio
RU	Ratio Unit defined as W_f/P_{s3} (lbm/sec/psig)
SM	Stall/surge Margin (%)
T_x	Temperature at station X (Rankine)
VBV	Variable Bleed Valve (Variable Inter-stage Bleed)
VSV	Variable Stator Vanes (Variable Inlet Guide Vanes)
W_f	Fuel flow rate (lbm/sec)
W_c	Corrected mass flow rate (lbm/sec)

1.0 Introduction

A detailed, physics-based component-level engine model, known as C-MAPSS40k, was developed to match the characteristics of data provided by a flight test of a highly instrumented engine during steady-state and transient maneuvers. These new data included transient engine behavior at specific flight conditions that are of interest in control system design. With these data serving as validation, the C-MAPSS40k open-loop engine model provides realistic responses in flight conditions spread across the flight envelope. A unique feature of C-MAPSS40k is that it models transient stall margin effects such as heat transfer and changes in compressor blade tip clearance in addition to calculating the steady-state stall margin for the low-pressure compressor (LPC) and high-pressure compressor (HPC).

NASA has previously developed generic, nonlinear, dynamic, component-level engine simulations capable of dynamic tests for research purposes, written in the MATLAB/Simulink environment. These include the Modular Aero-Propulsion System Simulation (MAPSS) and the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) (Refs. 1 and 2). These previous simulations defined much of the framework and requirements of C-MAPSS40k. Significant features present in MAPSS and/or C-MAPSS such as modularity, faster than real-time execution, a graphical user interface (GUI), and the ability to linearize the non-linear engine model were defined as requirements of C-MAPSS40k. New requirements and goals were added including:

- Increasing the robustness of the simulation to handle input and errors gracefully
- Improving the ease of operation by allowing the user to utilize either the GUI or the MATLAB command line to run the simulation
- Increasing the simulation accuracy by incorporating a more precise steady-state operating point solver
- Developing an engine controller representative of those found in industry

With these new data and requirements, a new engine system simulation was constructed using the structure of the engine component models from C-MAPSS. The component maps were scaled to match those of a 40,000 lb thrust class engine and the Simulink components were rewritten to match the design requirements. Finally, a new controller was designed and implemented.

The rest of this paper will discuss some of the details of the C-MAPSS40k implementation and the model's capabilities. Section 2.0 will present an overview of the open-loop engine and the baseline controller. In Section 3.0 some of the advanced capabilities of the model will be discussed in more detail, followed by Section 4.0 in which a flight test example is presented. Section 5.0 will summarize the C-MAPSS40k software and its capabilities.

2.0 Overview of the C-MAPSS40k Platform

The C-MAPSS40k platform is comprised of a number of elements: an open-loop engine, an “industry standard” controller, and additional support functionality such as a steady-state solver and a linearization routine. This section will provide high-level insight into the function of the open-loop engine and the baseline controller. The additional support components will be discussed in Section 3.0.

C-MAPSS40k is composed in the MATLAB/Simulink environment and links compiled C-code to Simulink blocks to allow for fast execution and rapid control system development. C-MAPSS40k was designed to be as flexible and modular as possible, thus, there are often numerous methods to achieve a desired result. For example, simulations may be executed and subsequently analyzed directly in the MATLAB workspace or through the provided GUI. The C-MAPSS40k User’s Guide provides the detail necessary for users to employ the full functionality of the simulation system (Ref. 3). The scope of this paper is limited to a description of the simulation components and discussions of the development approach.

2.1 The C-MAPSS40k Open-Loop Engine

The C-MAPSS40k engine simulation represents a generic, high-bypass, twin-spool commercial turbofan engine capable of generating 35,000 lbf of thrust at take-off conditions. Table 1 highlights some of the operational and physical specifications of the engine. The engine is comprised of six main components: fan, low pressure compressor (LPC), high pressure compressor (HPC), combustor or burner, high pressure turbine (HPT), and low pressure turbine (LPT). The HPC and HPT are connected through the core shaft or high-speed shaft; the fan, LPC, and LPT are all connected to the fan shaft or low-speed shaft. In addition to the five turbo-machinery components and the combustor, the engine has an inlet at the front, a nozzle at the rear, a bypass duct, a variable sized inter-stage bleed valve, a set of variable angle stator or guide vanes, and a number of cooling bleeds. A schematic of the engine is shown in Figure 1 along with station numbers that will be used throughout the rest of this paper. Note that for modeling purposes, the fan has been separated into two components: the fan tip and the fan hub. The flow through the fan tip passes directly into the bypass duct, while the flow through the fan hub passes into the core of the engine. As is commonly done, the fan hub has been combined with the LPC as it effectively serves as an additional compressor stage (Ref. 4).

TABLE 1.—SPECIFICATIONS OF THE OPEN-LOOP ENGINE

Bypass ratio.....	6:1
Maximum thrust.....	35,000 lbf
Fan diameter.....	74.5 in.
Overall Pressure Ratio (OPR).....	25
High-speed shaft inertia.....	21.465 lbf-ft ²
Low-speed shaft inertia.....	106.0 lbf-ft ²
Takeoff specific thrust.....	29 lbf/lbm/sec

The engine has three control inputs: fuel flow rate into the combustor (W_f), the position of the variable bleed valve (VBV), and the angle of the variable stator vanes (VSV). By controlling these values appropriately, the baseline controller is capable of operating at any desired power level in the flight envelope shown in Figure 2. The desired power level is based on the pilot’s commanded power lever angle (PLA). The PLA has a range of 40° to 80°. The controller interprets the PLA into a thrust demand that, by design, correlates with the PLA in a linear fashion (Ref. 4).

The next subsections will discuss the implementation of the engine model, then the sensor and actuator models, and then off-nominal engine operation.

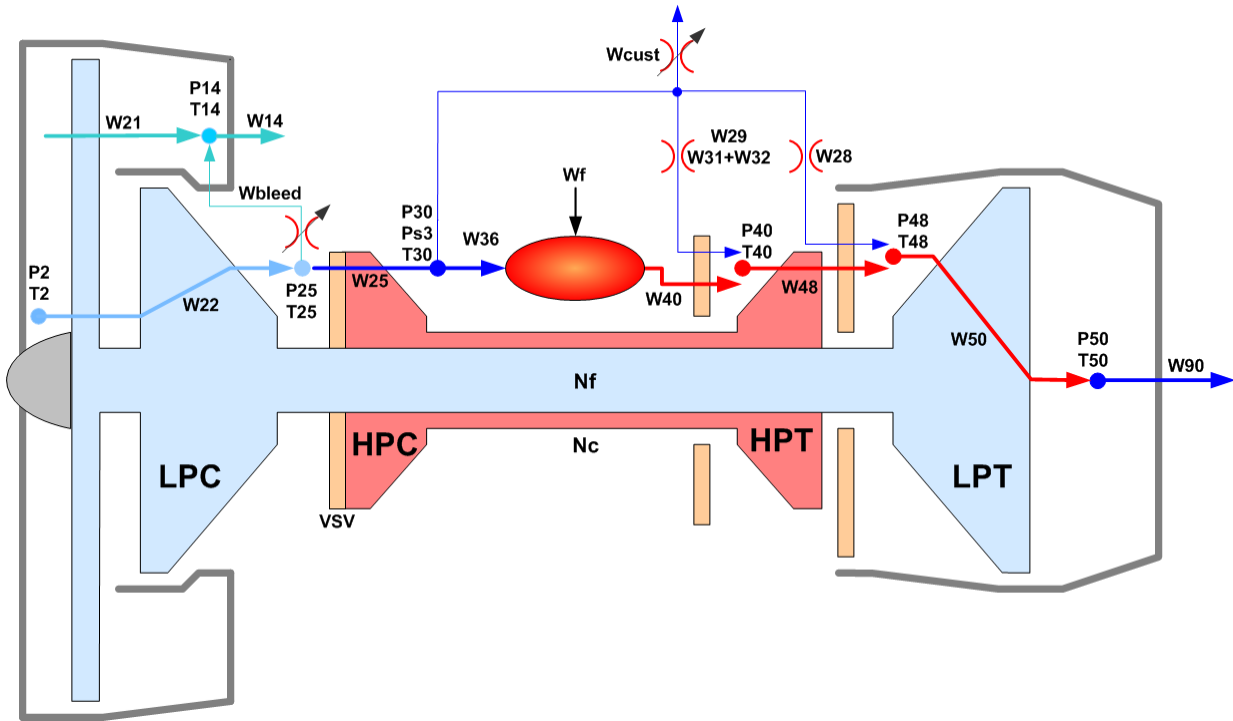


Figure 1.—Schematic of the engine.

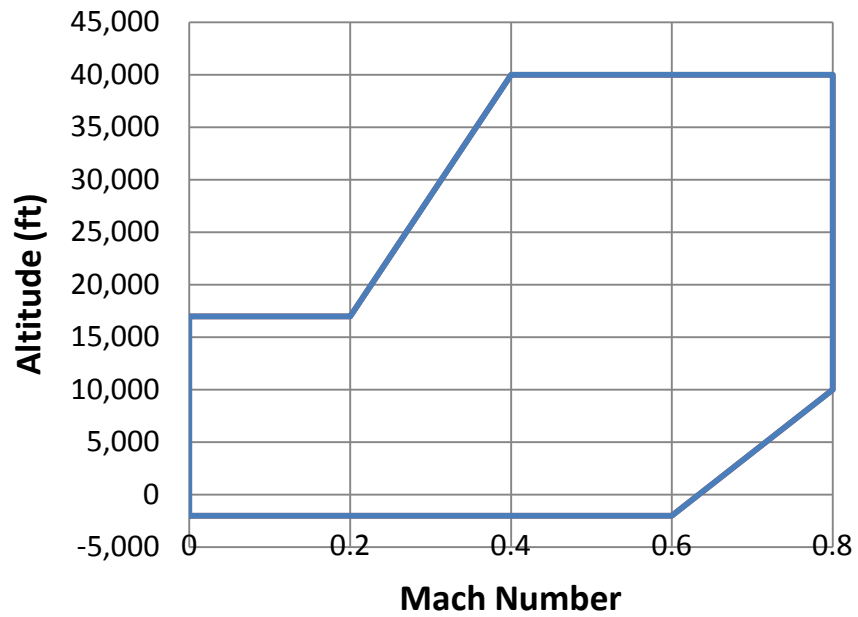


Figure 2.—C-MAPSS40k flight envelope.

2.1.1 The Open-Loop Engine Model

The engine model implemented in C-MAPSS40k is a physics-based, component-level model. Each of the engine components is represented as an infinitesimally small volume with an input pressure and temperature, an output pressure and temperature, and a mass flow rate through the volume. The engine is then balanced by ensuring continuity of mass flow rate through the components, thereby capturing the gross characteristics of each engine component.

The outputs and flows of each component are governed by maps relating pressure ratio, corrected mass flow rate, and adiabatic efficiency, which are parameterized to corrected shaft speed. The process of “correcting” engine parameters is well known in the field and is extremely useful for comparing engine operation at various environmental conditions (Ref. 5). An example compressor map for the fan hub/LPC is shown in Figure 3 with the thick black constant corrected shaft speed lines shown in addition to the red surge line. Note that the pressure ratios are not necessarily unique for a given speed line, most notably at low corrected speeds; thus, an additional independent parameter, known as an EM-line, is added based on pressure ratio and corrected mass flow rate. The EM-lines have a positive slope as shown by the thin black lines in Figure 3. The combination of EM-line and corrected shaft speed (the thick black lines) enables a specific point on the compressor map to be uniquely identified. It is worth noting that EM-lines, as well as the commonly used R-lines, are used for numerical purposes in simulations and are not part of the physics of axial compressors.

Each of the engine component models is programmed in C, which is then compiled into a binary MEX file using the default MATLAB compiler. The Simulink block diagram then links the MEX files together and handles the coordination of their inputs and outputs. Using compiled code in this manner produces significantly faster execution time compared to using interpreted code. For example, a 30 sec

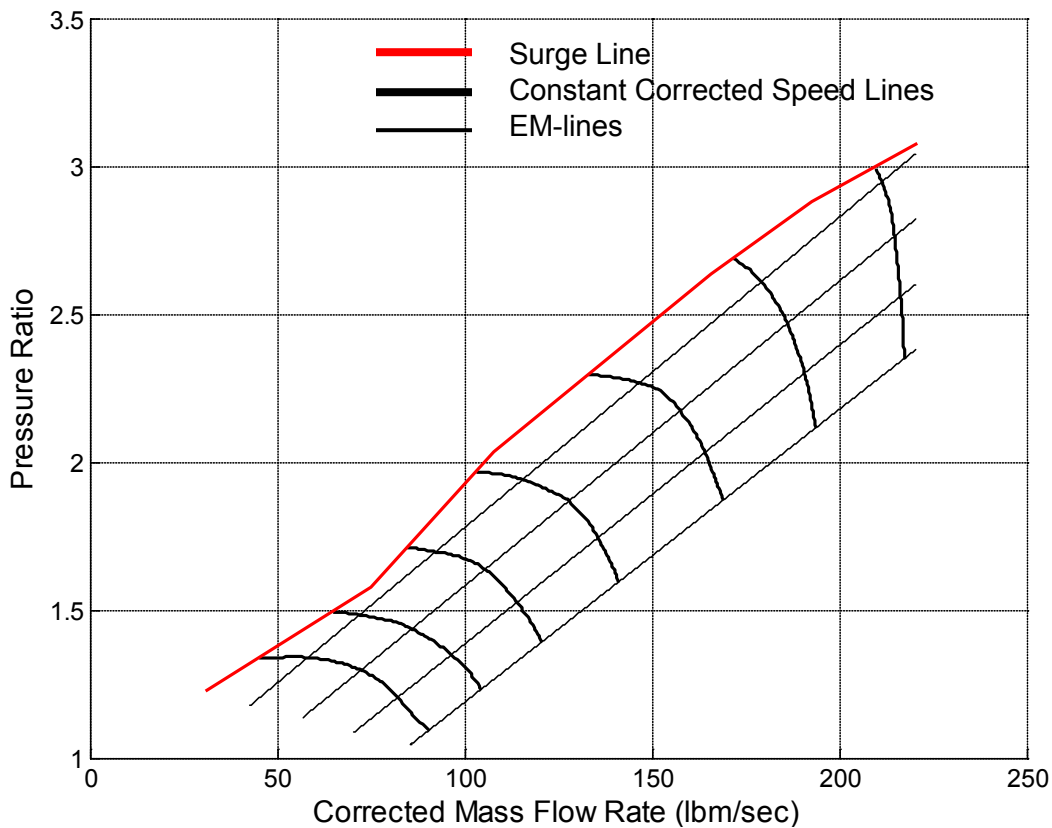


Figure 3.—Compressor map of the fan hub/LPC.

simulation of a large PLA transient with the default configuration takes less than 3 sec on a laptop with a 2.5 GHz processor when executed using the compiled components. Because the raw C-code is distributed with the software it can be compiled to work on any processor or machine that is capable of running MATLAB/Simulink. Further, this code can be modified and updated easily if desired.

As noted earlier, Simulink serves to interconnect the various engine components and enforce the boundary conditions necessary for their operation. A critical part of this task is to ensure that the results returned by the MEX files every time step are physically realizable. In C-MAPSS40k, the primary concern is that energy is conserved at each time step. To enforce this condition, C-MAPSS40k employs a solver that iteratively changes each component's operating point until the mass flow rate through the engine is in balance for that time step while enforcing the constraint that all pressures and temperatures are physically realizable. Under normal operation, the simulation will only proceed once all of the errors in mass flow rates are less than some specified threshold. The iterative solver will be discussed in more detail in Section 3.4.

Since the iterative solver ensures mass flow balance, the only unbalanced term remaining in the engine is the torque on the fan and core shafts. As discussed earlier each shaft connects a compressor and a turbine. The turbine's purpose is to extract energy from the flow in order to drive the associated compressor. Thus, each component model computes the amount of work done on the shaft. When there is imbalance in work, the shaft will accelerate. In this way the engine can vary the shaft speeds and thus the generated thrust. Once there is a power balance between the compressor and turbine the shaft will cease to accelerate and will attain a steady-state speed that results in steady-state behavior from the engine.

2.1.2 Engine Sensors and Actuators

The engine represented in C-MAPSS40k includes dynamic models of the sensors and actuators typical of those found on commercial engines. The installed sensors include fan speed, core speed, T_2 , T_{25} , T_{30} , T_{50} , P_2 , P_{25} , P_{30} , and P_{50} . The engine controller in C-MAPSS40k utilizes all of these sensors with the exception of P_{25} , T_{30} , and T_{50} (these sensors are provided for diagnostic purposes). To improve the realism of C-MAPSS40k, the user has the ability to specify the variance of Gaussian noise that is added to the sensor signal. At the present time sensor failures are not modeled, but can be added if a user desires.

The engine has three control effectors: a fuel-metering valve (FMV), the VBV, and VSV. The dynamics of each of the actuators is modeled to more accurately capture the dynamics of the complete engine control system. For example, the FMV takes the desired fuel flow rate as an input that is then converted to the position of a linear actuator that will produce the desired fuel flow rate. A feedback loop drives the actuator to follow the desired position subject to rate limits and hysteresis. The fuel flow is then subject to a configurable transport delay of 40 msec to simulate the time required to deliver the fuel from the actuator to the combustor. The variable bleed valve actuator has a significantly simpler model comprised entirely of bandwidth-dominated dynamics. The variable stator vane actuator is necessarily more complex as the mechanical system has a significant amount of backlash and delay. The VSV actuator modeled in the C-MAPSS40k engine contains a proportional-integral (PI) controller that drives the vanes to the desired angle. The backlash is nominally specified as 1 percent of the full motion of the vanes.

Because of the modular architecture of C-MAPSS40k any of these components can be modified without necessitating changes to the rest of the engine. Adding more details to a component or simply replacing one altogether with a different model can be easily accomplished.

2.1.3 Engine Degradation and Component Fault Simulation

Every engine is subject to gradual wear during its life and occasionally subject to rapid onset faults such as the ingestion of foreign objects. C-MAPSS40k gives the user the ability to specify the level of normal engine deterioration from the nominal "just broken-in" engine to an engine that is at the end of its service life via a configurable deterioration level parameter. This parameter changes the engine health parameters to produce realistic deterioration.

The engine health parameters are a set of inputs to each of the engine components that modify the nominal behavior of the component. For each compressor the user can modify the flow capacity, efficiency, and the pressure ratio; for the two turbines the flow capacity and efficiency are available to be modified, for a total of 13 health parameters in C-MAPSS40k. The ability to introduce faults or failures into the simulation is very helpful for engine diagnostics or for evaluating control system robustness. These rapid failures are implemented by introducing an abrupt shift in the engine health parameters, which is the standard modeling approach (Ref. 6).

2.2 The C-MAPSS40k Baseline Controller

C-MAPSS40k has a controller that is representative of a common industry controller in both architecture and performance to ensure that transient performance realistically represents that of commercial aircraft engines. Additionally, for users that desire to design a new engine controller, it is critical to be able to compare the response of the new or modified controller to that of a realistic baseline controller. While the development process for the C-MAPSS40k baseline controller is documented in detail in Reference 7 it will be briefly discussed here. The controller has two main components: the power management controller and the protection logic controllers. The purpose of the controller is to convert the pilot's PLA into a thrust in a linear and predictable manner and to maintain that thrust when faced with disturbances while providing safe operation. To accomplish this task the controller can command fuel flow rate, VBV position, and VSV angle. A notional diagram of the baseline controller is shown in Figure 4 for reference. For the sake of clarity, many of the inputs are not shown (e.g., inlet environmental conditions are used in nearly every block). Additionally, it is worth noting that the output, Wf_des , is the desired fuel flow rate command that is then sent to the fuel-metering valve where a separate closed-loop system tracks the value of Wf_des .

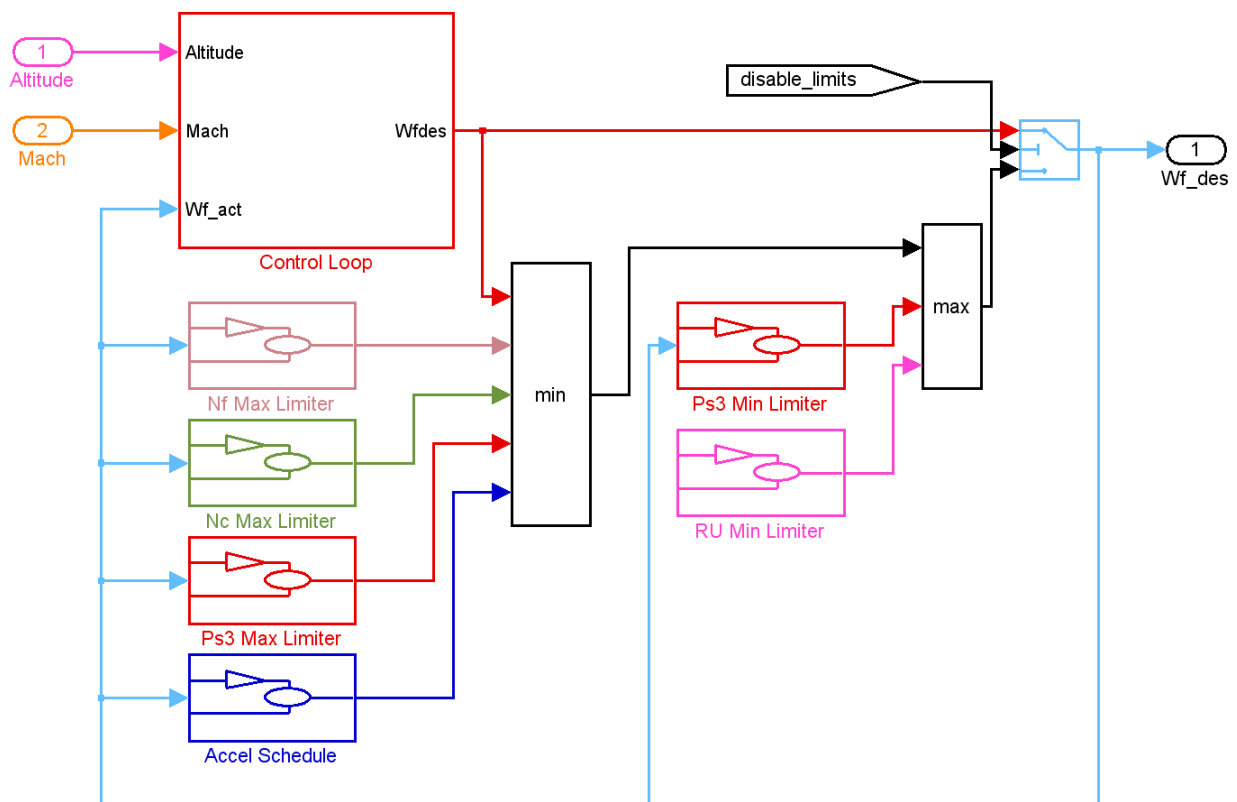


Figure 4.—Notional diagram of C-MAPSS40k controller.

2.2.1 Power Management Controller

Since engine thrust cannot be directly measured, other sensed parameters that correlate well with thrust are used for control purposes. The two most common such parameters are fan speed (N_f) and engine pressure ratio (EPR) (Ref. 4), which is the ratio of turbine exit pressure to inlet pressure (P_{50}/P_2). The C-MAPSS40k baseline controller has both EPR and N_f controllers so that users may select whichever type they desire. The controller determines an EPR or N_f setpoint based on the PLA, altitude, Mach number, and ambient temperature that, when achieved, results in the desired linear thrust profile. A PI-controller then regulates fuel flow rate to maintain the setpoint regardless of external or internal disturbances.

In addition to modulating fuel-flow rate, the controller is responsible for specifying the desired position of the VBV and VSV. In C-MAPSS40k, as in industry practice, these parameters are handled in an open-loop manner, and thus are scheduled based on sensed engine variables. The Variable Bleed Valve position is scheduled on corrected fan speed and free-stream Mach number while the desired Variable Stator Vane angle is based solely on corrected core speed.

2.2.2 Protection Logic Controllers

The major function of the protection logic controllers is to provide safe and smooth thrust transitions between distant set points. Safe operation implies 1) protection of physical components by preventing overstress, and 2) stall avoidance. This protection is achieved through the use of controller limits on physical variables including limits on the maximum fan speed, core speed, and burner pressure (Ref. 8) as shown in Figure 4. Exceeding any of these limits will cause mechanical stress and potentially catastrophic damage to the engine. Operability limits are more difficult to capture and to ensure. The three main operability limits governed by the baseline controller are engine compressor surge (also known as compressor stall), combustor lean-blow out, and flight idle operation. These operability limits are intuitive except for the flight idle case, which preserves the ability of the aircraft to execute a “go around” maneuver while on approach.

High pressure compressor surge occurs on acceleration when enough compressor blades stall aerodynamically that the pressure gradient can no longer be maintained. This is typically associated with a sudden drop in inlet pressure and causes shock, vibration, and loss of thrust (Ref. 9). Since compressor stall margin cannot be measured, it is protected indirectly by limiting the rate of acceleration of the engine. In C-MAPSS40k this is implemented as a limit on the acceleration of the core shaft and is labeled as the “Accel Schedule” in Figure 4. For any given condition, a maximum allowed core shaft acceleration is determined based on the current core shaft speed. For small power transients, this limiter will not become active, but when a large change in thrust is desired, the acceleration limit will restrict the rate at which the engine can spin-up to ensure that the HPC does not stall even if the engine is fully deteriorated.

Lean combustor blow-out can occur during decelerations when a command to cut thrust rapidly is issued. Combustion can only occur in a limited range of fuel/air ratios for a given air mass flow (Ref. 9). If the fuel flow rate is cut to near zero while the mass flow rate is still high, the flame will extinguish resulting in lost power and in aerodynamic and turbine blade vibration, which reduces blade life. In addition, in cases of rapid deceleration, LPC stall is a concern. To avoid both of these operational risks, the baseline controller limits the rate at which fuel can be decreased based on the current combustor pressure (P_{s3}). This limit is implemented by preventing the ratio W_f/P_{s3} from going below a specified threshold and is represented by the “RU Limiter” shown in Figure 4. In addition to limiting the fuel flow rate, the controller may use the variable bleed valve to help avoid LPC stall in some cases. Upon detecting a specified level of deceleration, the engine controller will issue a command to open the VBV further than the nominal scheduled value. This air is then bled to the bypass duct rather than fed into the engine core, thus preserving a stable pressure ratio across the LPC.

Idle control is maintained by preventing the controller from producing an unobtainable EPR or N_f value. These cases occur at low power where the mass flow rate through the engine is too low to drive engine components. As discussed in Reference 4 there are typically two idle levels: flight idle and ground idle. Ground idle will run the engine at the lowest steady-state thrust level possible while maintaining

enough power to drive aircraft accessories during taxiing. Flight idle is scheduled to balance between minimizing thrust during descent and landing, and maintaining the required time to accelerate from idle to take-off power in case of an emergency go-around. The Federal Aviation Administration requires that the engines on commercial aircraft be capable of accelerating from the flight idle limit to 95% of full power in less than five seconds at sea level (Ref. 10). Flight idle will typically produce more thrust than the ground idle schedule and a switch that detects the presence of the aircraft's weight on the landing gear governs the transition between the two states. In C-MAPSS40k, a switch handles changing between ground and flight idle.

In addition to preserving the engine limits, the engine controller is responsible for meeting the transient performance requirements specified in the Federal Aviation Requirement (FAR) Section 33.73 (Ref. 11). These requirements set the maximum time allowed for power transients while ensuring that all safety and operability conditions are satisfied. This requirement forces the control system designer to strike a balance between an always safe but slow engine and a quickly responding engine. More details on the C-MAPSS40k baseline controller and the FAR 33.73 regulation can be found in Reference 7.

3.0 C-MAPSS40k Advanced Capabilities

C-MAPSS40k contains the functionality found in MAPSS and C-MAPSS, as well as a number of advanced capabilities that are not found in those simulations. These topics are highlighted here.

3.1 Surge Margin Models

Surge margin is a measure of distance between the compressor operating point and the surge line. While there are a number of ways to define surge margin, C-MAPSS40k has adopted the Society of Automotive Engineers formulation:

$$SM = 100 * \frac{PR_{\text{surge}} - PR_{\text{operating point}}}{PR_{\text{operating point}}} \quad (\text{Ref. 10}) \quad (1)$$

where the pressure ratios are specified at a constant corrected mass flow rate and the computed surge margin is a percentage. This definition is shown on the HPC compressor map in Figure 5. The marked operating point has a pressure ratio of 8.577 at a corrected mass flow rate of 75.83 lbm/sec. The corresponding surge point is the point on the red surge line with the same corrected mass flow rate, which in this case has a pressure ratio of 12.09. Therefore, using Equation (1), the surge margin at this operating condition is 40.96 percent.

As with most high-fidelity engine simulations, the LPC and HPC compressor surge margins are readily calculable based on the current operating point and the distance to the component's surge line as discussed above. However, the actual surge margin remaining in a compressor is usually less than that specified by the compressor map as can be seen in the HPC surge margin plot of Figure 6. There are a number of factors that contribute to a change (usually a reduction) in surge margin, including: compressor blade heat transfer effects, variations in compressor tip clearance, debits due to normal engine deterioration, and debits due to engine-to-engine variation. Unlike other non-proprietary engine models, C-MAPSS40k includes models that represent each of these surge margin effects except for engine-to-engine variation. These surge margin debits are not based on the compressor map and operating point, rather they are based on the properties of the engine components. For example, the deterioration debit is constant based on engine deterioration while the heat transfer debit is based on the state of the HPC and is heavily influenced by the slow time decaying heat transfer between the compressor blades and casing. The individual debits for an acceleration transient are shown in the bottom plot of Figure 6. A further interesting point is that the tip clearance debit is negative prior to the transient at 14 sec, which highlights the statement that the surge margin remaining is not necessarily less than the surge margin as determined from the compressor map.

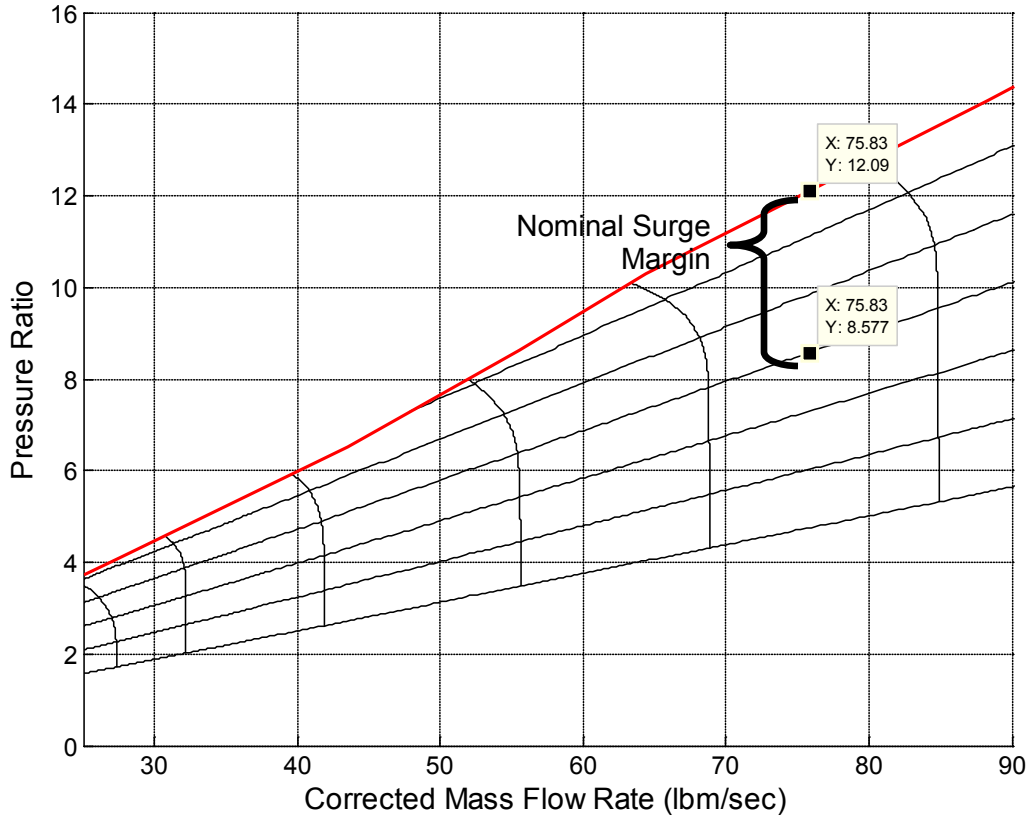


Figure 5.—HPC compressor map with operating and surge points.

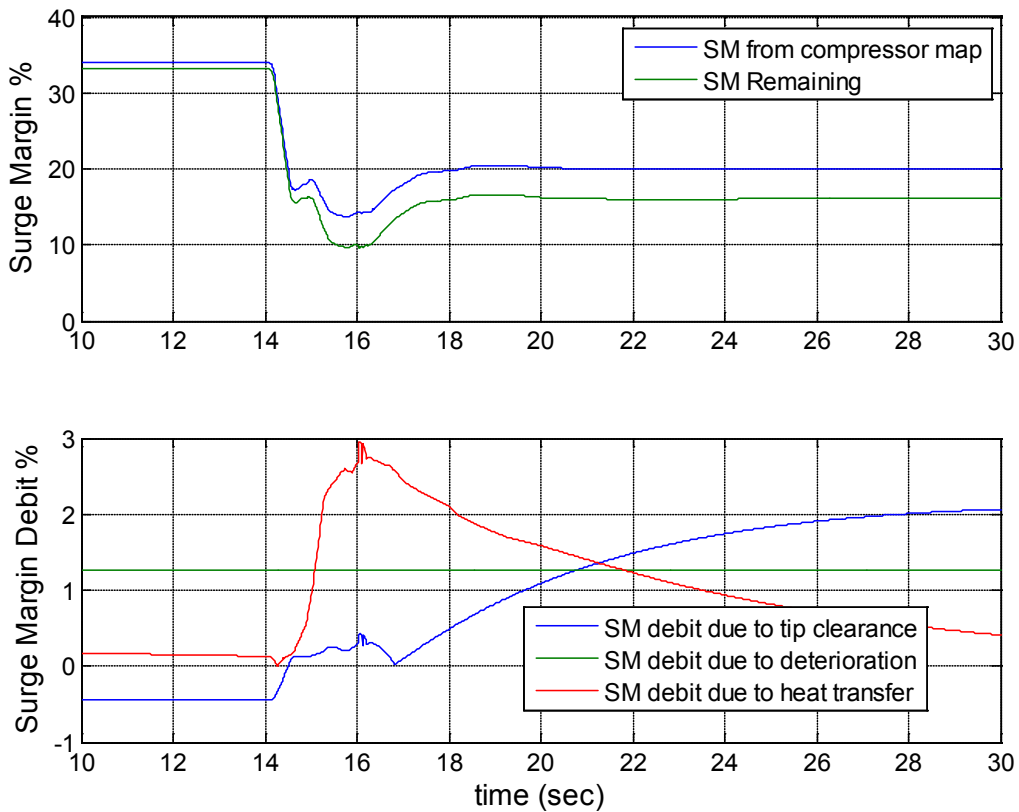


Figure 6.—HPC surge margin debits during acceleration.

Modeling the remaining surge margin with the debits included allows a control system designer to design a more aggressive controller while ensuring stability during acceleration. When designing a controller it is necessary to maintain a surge margin reserve to account for each of the surge margin debits (Ref. 12). This reserve is significant and in Reference 10 it is estimated to be approximately 24.4 percent, which means that the compressor surge margin as calculated in Equation (1) for a median, new engine can never drop below this level in steady state flight. Since many of the debits are explicitly modeled in C-MAPSS40k, the amount of uncertainty in the reserve decreases (to approximately 8 percent). With this smaller reserve, we can state that the closed-loop engine will not enter the surge regime if the worst-case transient surge margin remaining is greater than the reserve.

3.2 Steady-State Solver

C-MAPSS40k employs a steady-state engine solver to trim the engine to the desired operating point at the start of every simulation. The steady-state solver is flow-balanced in the same manner as the engine model; however, the shaft torques are also balanced. Thus, the point returned by the steady-state solver is very close to the actual trim point of the engine at that operating condition. It should be noted that the steady-state solver trims the open-loop engine model, and the point determined may not be feasible. Thus, when running in closed loop from the trimmed point, users should allow a few seconds for any minor start-up transients to subside prior to beginning any simulation study.

3.3 Linearization Routine

Like MAPSS and C-MAPSS before it, C-MAPSS40k contains a routine to determine a linear representation of the open-loop engine at a user-specified operating point. The linear system generated is of the form described in Equation (2):

$$\begin{cases} \dot{x} = Ax + Bu + Lh \\ y = Cx + Du + Mh \end{cases} \quad (2)$$

The state vector x is comprised of the two state variables of the linear system: fan speed and core speed. In C-MAPSS40k, the user has the ability to specify the linearized system's control input vector u . The user can choose to use fuel flow rate as the only explicit control input, leaving the implicit VBV and VSV to operate on their pre-defined schedule; alternately all three of the variables can be used as explicit control inputs. The elements of the h vector are the health parameters as discussed earlier in Section 2.1.3. The output vector y contains most of the sensed engine values as well as the net thrust and stall margin remaining for both compressors. The linearization routine returns values for the six matrices (A , B , L , C , D , and M), which can then be used to generate a piece-wise linear model as well as to design linear point controllers.

3.4 Iterative Solver

The engine mass flow solver in C-MAPSS40k is an advanced version of the solver found in C-MAPSS. The new flow solver has an iterative form that ensures that all five flow errors (shown in Table 2) are below a user-configurable threshold. In order to prevent an endless loop if the flow errors do not converge, an upper limit is placed on the number of iterations allowed. This technique enables the user to adjust the level of acceptable flow errors for the specific situation. During large transient tests, it may be acceptable to sacrifice accuracy for execution speed. However, in cases where very small transient behaviors are being studied a high level of accuracy is required, and the solver can be easily tuned to achieve the requisite performance. Figure 7 shows two cases of simulated engine thrust for the same throttle input; one used the iterative solver and allowed up to 200 iterations per time step to converge, resulting in small error, the other case used a single iteration per time step, resulting in significantly larger error. Note that the peak number of iterations occurs during rapid changes in simulated variables, PLA in this case; however, at no point does the number of required iterations to meet the error tolerance exceed 100 iterations.

TABLE 2.—FLOW CONTINUITY ERRORS

Flow Balance Equations
$e(1) = W_{21} + W_{Bleed} - W_{14}$
$e(2) = W_{22} - W_{Bleed} - W_{25}$
$e(3) = W_{50} - W_{90}$
$e(4) = W_{48} - W_{31} - W_{32} - W_{29} - W_{40}$
$e(5) = W_{50} - W_{48} - W_{28}$

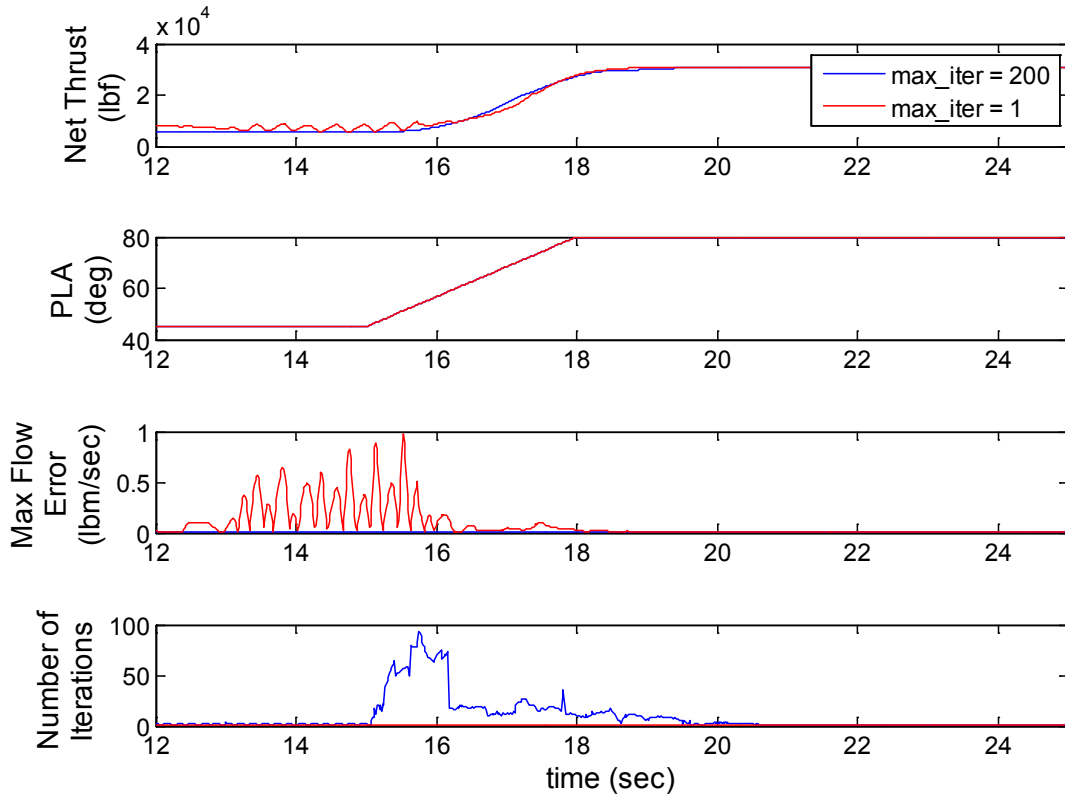


Figure 7.—Comparison between engine flow solvers; one case allows as many as 200 iterations to converge, the other case uses a single iteration.

3.5 Graphical User Interface

C-MAPSS40k includes a GUI, shown in Figure 8, that allows users who are not familiar with MATLAB to easily run, save, and analyze simulation test cases. On the left-hand side of the GUI one can specify a custom flight profile, controller configuration, and adjust the parameters of the iterative solver. The right-hand side provides the user the ability to specify normal engine degradation as well as the engine health parameters. After running a simulation, all of the input and output data can be saved to a MATLAB data file or a comma separated file for further analysis and processing. Additionally, the user can use the “Plot” button to run a user configurable MATLAB script on the output data. Through this function, users can plot the data or run analyses as desired. More experienced MATLAB users or those who want to customize the simulation’s features can bypass the GUI and run C-MAPSS40k directly from the command line.

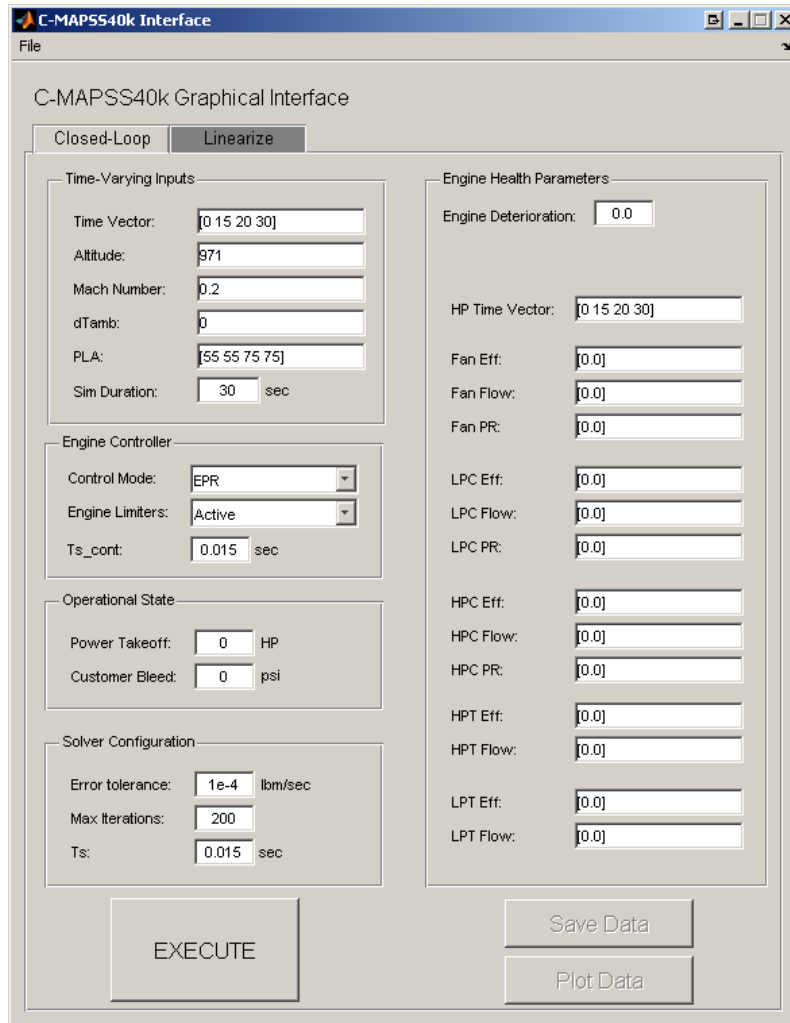


Figure 8.—The C-MAPSS40k graphical user interface.

3.6 Real-Time Capability

C-MAPSS40k has been designed with the capability to run faster than real-time by utilizing C-code compiled into MEX files and efficient Simulink coding practices. The number of iterations required to balance the engine flows primarily governs the execution duration of a simulation; thus, decreasing the flow error tolerance will increase the simulation execution time. The maximum number of iterations per time step is governed by a configurable parameter. In typical tests, the simulation never requires more than the default 200 iterations to converge for a single time step. Figure 9 shows the worst-case processing time on a modern laptop PC.¹ If real-time operation is required, the user can modify the maximum number of iterations to ensure that no time-step will take longer than real-time to calculate (per the results shown in Figure 9 for the computer specified above, this limit is 56 iterations). Note that the accuracy of the simulation this will decrease if the iterative solver cannot reduce the flow errors to within the specified tolerance in the given number of iterations.

¹A 32-bit Windows XP Professional with a 2.5 GHz Intel processor running MATLAB 2010a

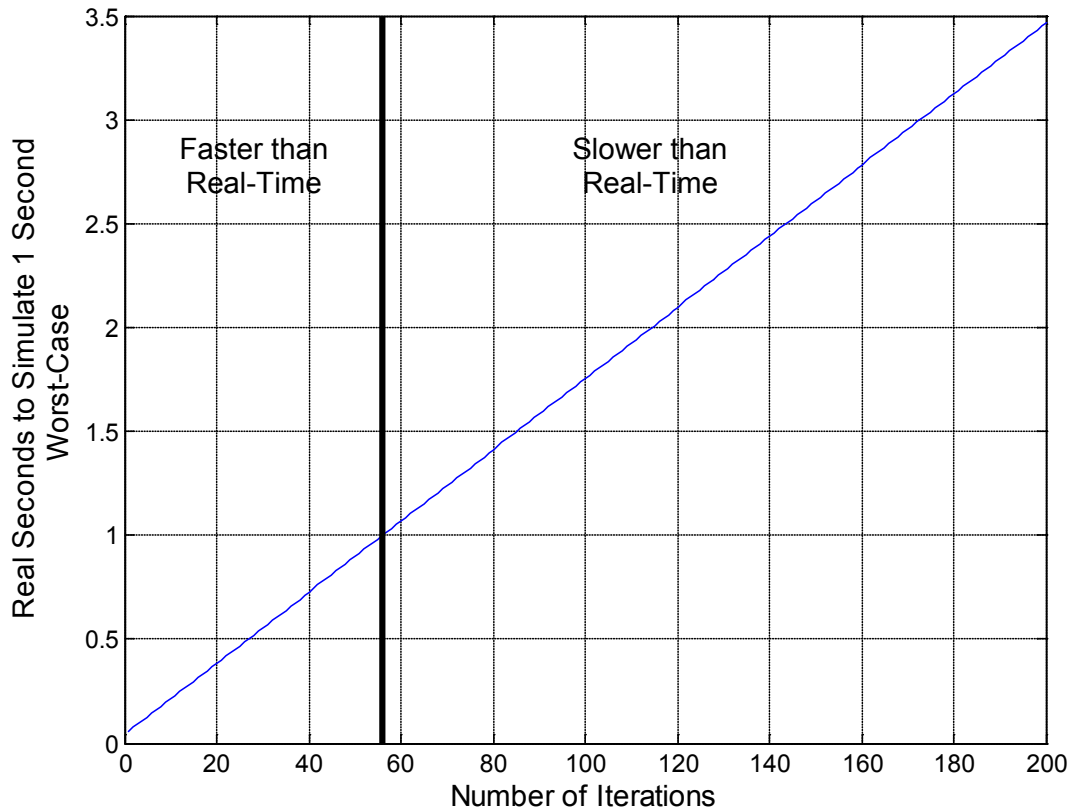


Figure 9.—Worst case execution times for the specified computer.

Another option is to use the MATLAB Real-Time Workshop to produce a compiled executable from the Simulink model (Ref. 13). The techniques necessary to achieve this are beyond the scope of this paper.

4.0 C-MAPSS40k Flight Examples

The following set of plots are taken from a two-second (starting at $t = 14$ sec) idle to full power transient of C-MAPSS40k to demonstrate the open-loop engine and the baseline controller. The situation being modeled is an emergency go-around on approach to landing with the environmental conditions set to an altitude of 2,000 ft at a free stream Mach number of 0.25 on a standard day and the engine is 50 percent deteriorated. Figure 10 shows the net thrust generated by the engine as well as the fan and core shaft speeds. Here we observe the expected smooth thrust profile with no overshoot transitioning from low to high power settings. Other features of note in the plot are the long time delay in the engine response and the fact that the emergency maneuver was completed within the time allowed by the FAR 33 regulation. Figure 11 contains plots of the LPC and HPC operating points on the compressor map during the transient as well as the associated surge margins, including the HPC surge margin with the surge margin debits accounted for. Here it is important to note that neither surge margin fell below zero, thus compressor surge did not occur on the transient. Moreover, since the surge margin remaining (which includes the surge margin debits) is greater than the previously discussed reserve limit of 8 percent, this maneuver will not cause the HPC to surge regardless of factors such as engine-to-engine variation and inlet distortion. The operating lines moved as expected during the transient: the LPC moved away from the surge line and the HPC moved towards it. One final point is that since the engine has used half of its normal life, the stall margin debit due to deterioration can be observed prior to the transient. The larger offset after and during the transient are due to tip clearance and heat transfer debit effects.

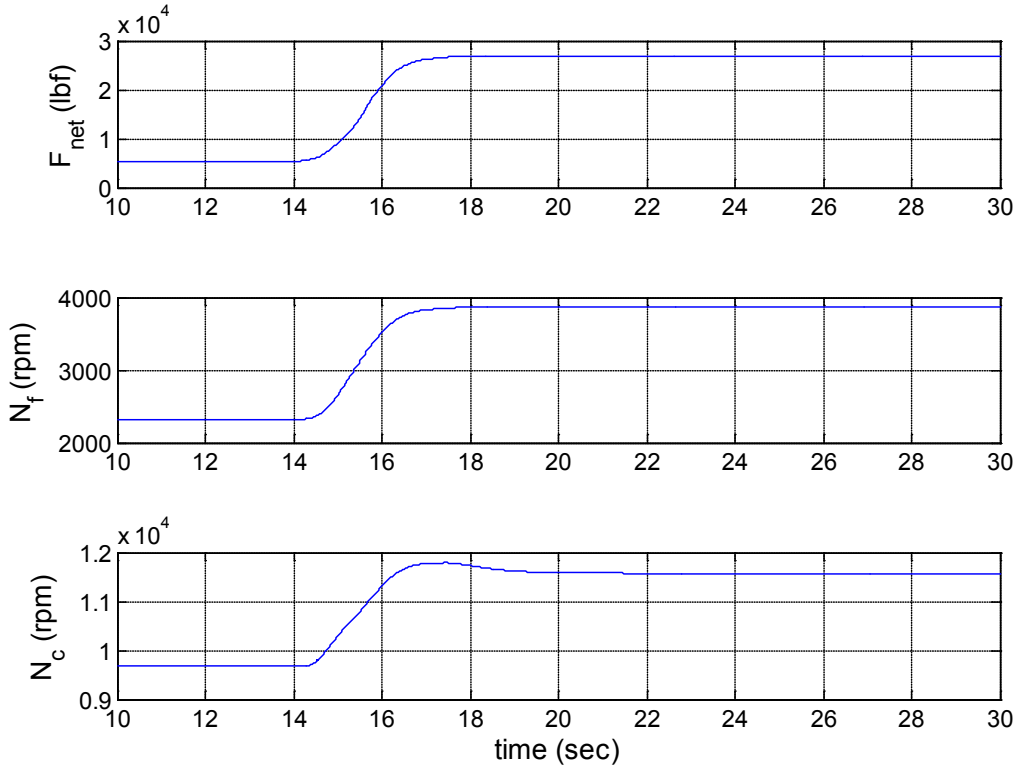


Figure 10.—C-MAPSS40k transient output.

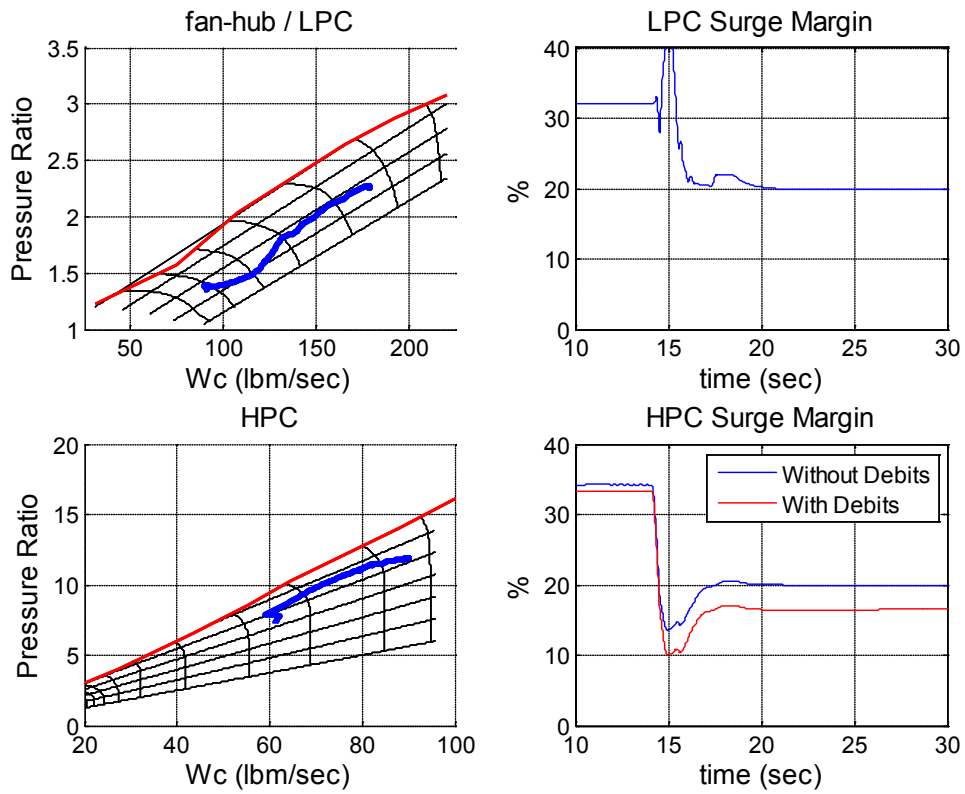


Figure 11.—C-MAPSS40k transient surge margins.

5.0 Summary

C-MAPSS40k is a detailed, non-proprietary, component-level model of a commercial turbofan engine in the 40,000 lb thrust class. In addition to the high-fidelity engine simulation, a realistic engine controller that is representative of the current industry approach is included. This controller delivers realistic closed-loop performance as well as provides a baseline against which to compare controller developments. The closed-loop system meets all relevant government and airframe manufacturer's requirements. Also contained in the simulation are highly detailed steady-state and transient stall margin models not found in other publicly available simulations. The engine model can be degraded due to normal wear and component faults can be simulated by using the engine health parameters. Due to the manner in which the simulation was developed, C-MAPSS40k is capable of being run in real-time. Finally, due to the modular design, engine or controller components can be easily modified or completely redesigned without requiring changes to the rest of the simulation.

References

1. Parker, K.I. and Guo, T.H., "Development of a Turbofan Engine Simulation in a Graphical Simulation Environment," NASA/TM—2003-212543, August 2003.
2. DeCastro, J.D., Litt, J.S., and Fredrick, D.K., "A Modular Aero-Propulsion System Simulation of a Large Commercial Aircraft Engine," NASA/TM—2008-215303, September 2008.
3. May, R.D., Csank J., Litt, J.S., and Guo, T.H., "Commercial Modular Aero-Propulsion System Simulation 40k (C-MAPSS40k) User's Guide," NASA/TM—2010-216831.
4. Jaw, Link C., with Jack D. Mattingly, *Aircraft Engine Controls: Design, System Analysis, and Health Monitoring*, American Institute of Aeronautics and Astronautics, Inc, VA, 2009.
5. Volponi, A.J., "Gas Turbine Parameter Corrections," *Journal of Engineering for Gas Turbines and Power*, Vol. 121, Issue 4, October 1999.
6. Ogaji, S.O.T., Sampath, S., Singh, R., and Probert, S.D., "Parameter Selection for Diagnosing a Gas-Turbine's Performance-Deterioration," *Applied Energy*, 73, 2002, pp. 25–46.
7. Csank, J., May, R.D., Litt, J.S., Guo, T.H., "Control Design for a Generic Commercial Aircraft Engine," AIAA–2010–6629, 2010 AIAA Joint Propulsion Conference, Nashville, TN, July, 2010.
8. Spang III, H.A., and Brown, H., "Control of Jet Engines," *Control Engineering Practice*, Vol. 7, No. 9, 1999, pp. 1043–1059.
9. Cohen, H., Rogers, G.F.C., Saravanamuttoo, H.I.H., *Gas Turbine Theory*, 3rd Edition, Longman Group UK Ltd., 1987.
10. Walsh, P.P., Fletcher, P., *Gas Turbine Performance*, 2nd Edition, ASME Press, Blackwell Publishing, NJ, 2004.
11. Federal Aviation Administration, "Title 14 of the Code of Federal Regulations," http://www.airweb.faa.gov/Regulatory_and_Guidance_Library/rgfar.nsf/MainFrame, accessed on June 2010.
12. DeLaat, J.D., Southwick, R.D., Gallops, G.W., "High Stability Engine Control (HISTEC)," NASA/TM—1996-107272, July 1996.
13. The Mathworks Inc., Simulink 2010 Documentation, <http://www.mathworks.com/access/helpdesk/help/toolbox/simulink/>

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>				
1. REPORT DATE (DD-MM-YYYY) 01-10-2010		2. REPORT TYPE Technical Memorandum		3. DATES COVERED (From - To)
4. TITLE AND SUBTITLE A High-Fidelity Simulation of a Generic Commercial Aircraft Engine and Controller			5a. CONTRACT NUMBER	
			5b. GRANT NUMBER	
			5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) May, Ryan, D.; Csank, Jeffrey; Lavelle, Thomas, M.; Litt, Jonathan, S.; Guo, Ten-Huei			5d. PROJECT NUMBER	
			5e. TASK NUMBER	
			5f. WORK UNIT NUMBER WBS 457280.02.07.03.03.01	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration John H. Glenn Research Center at Lewis Field Cleveland, Ohio 44135-3191			8. PERFORMING ORGANIZATION REPORT NUMBER E-17442	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001			10. SPONSORING/MONITOR'S ACRONYM(S) NASA	
			11. SPONSORING/MONITORING REPORT NUMBER NASA/TM-2010-216810	
12. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified-Unlimited Subject Category: 07 Available electronically at http://gltrs.grc.nasa.gov This publication is available from the NASA Center for AeroSpace Information, 443-757-5802				
13. SUPPLEMENTARY NOTES				
14. ABSTRACT A new high-fidelity simulation of a generic 40,000 lb thrust class commercial turbofan engine with a representative controller, known as C-MAPSS40k, has been developed. Based on dynamic flight test data of a highly instrumented engine and previous engine simulations developed at NASA Glenn Research Center, this non-proprietary simulation was created especially for use in the development of new engine control strategies. C-MAPSS40k is a highly detailed, component-level engine model written in MATLAB/Simulink (The MathWorks, Inc.). Because the model is built in Simulink, users have the ability to use any of the MATLAB tools for analysis and control system design. The engine components are modeled in C-code, which is then compiled to allow faster-than-real-time execution. The engine controller is based on common industry architecture and techniques to produce realistic closed-loop transient responses while ensuring that no safety or operability limits are violated. A significant feature not found in other non-proprietary models is the inclusion of transient stall margin debits. These debits provide an accurate accounting of the compressor surge margin, which is critical in the design of an engine controller. This paper discusses the development, characteristics, and capabilities of the C-MAPSS40k simulation.				
15. SUBJECT TERMS Turbofan engine control				
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 22
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U		
				19b. TELEPHONE NUMBER (include area code) 443-757-5802

