

**ADVANCED PROPULSION SYSTEM SIMULATION MODEL FOR A MODERN FIGHTER AIRCRAFT
TRAINING AID**

M. Bolívar , A. Jiménez, C. Pérez,
ITP, Industria de Turbo Propulsores S.A.
Av. de Castilla 2, Parque Empresarial San Fernando, 28830 San Fernando de Henares, Madrid, SPAIN

T. Breuer, C. Riegler, K. Salchow
MTU Aero Engines GmbH
Dachauer Str. 665, 80995 München, GERMANY

ABSTRACT

In times of shrinking budgets for nations' air forces there is an increasing demand for affordable, yet highly realistic training devices to perform initial pilot training as well as to provide the capability to allow for routine training in both emergency situations and air combat tactics. In order to fulfill these requirements, simulator devices need to provide the required fidelity as well as affordable operating costs.

This paper describes the development of a simulation model of the EJ200 engine which will be used in the simulator for the Eurofighter/Typhoon fighter aircraft. Following a short overview of the different concepts which are in use for this purpose, the approach adopted for the EJ200 engine simulation software is outlined.

The simulator model consists of modules representing physical entities of the real engine, such as the control unit, the actuators and the engine itself. Each of these software modules aims to mimic the physical/functional properties of its parent module as closely as possible, always taking into account requirements specific to the simulation environment, such as real-time capability, quick turn-around times between training elements by supporting special simulator-specific functions and simulation of malfunctions.

The result is a software model which offers a highly realistic simulation of engine behaviour during both normal and emergency conditions with the advantage of allowing easy upgrades in the future due to its modular approach.

Copyright © 2003 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved

INTRODUCTION

Aircrew training is an important but costly and risky task during the entry into service activities of nations' air forces when moving to a new type of fighter aircraft. In order to meet shrinking budget requirements and to lower flight training operational risks the use of flight simulators for aircrew training purposes has become more and more common. Simulators allow cost reduction and a flexible, affordable and risk free training against unpredictable mission foes and potential areas of conflict in normal and emergency aircraft operation.

For these reasons the Eurofighter/Typhoon aircraft contracting authority NETMA, founded by the four European nations Germany, Great Britain, Italy and Spain, asked the Eurofighter companies for the development of an Aircrew Synthetic Training Aid (ASTA).

Being the aircraft propulsion system responsible partner Alenia in turn asked the Eurojet companies (Avio, ITP, MTU and Rolls-Royce) for supporting ASTA by providing a Eurofighter engine (EJ200) simulation sub-model called PSSM (Propulsion System Simulation Model). Eurojet companies Avio, ITP and MTU accepted the task and decided to build the PSSM.

ADVANCED PSSM CONCEPT

PSSM APPLICATION AND MODELING CONCEPT OPTIONS

Propulsion system simulation models are used in a variety of applications as for example engine control system development and validation, air-

craft/engine control system interaction studies, embedded models in engine control or in flight systems and last but not least for flight simulators. A very profound overview of PSSM applications can be found in [1].

On the other hand a variety of modeling concepts exists to design and implement a real-time PSSM or its sub-models respectively. The most common model types comprise

- transfer function models,
- piece-wise linear models usually implemented as state-space models and
- full physical or hardware related models as for example full thermodynamic models for the engine itself.

The question of which model type is best suited to fulfill the requirements of a given application is not easy to answer. A lot of trades have to be made concerning calculation time, cost (of course for the hardware but also for development, implementation and maintenance effort) and accuracy to design a PSSM which meets the customer requirements. Typical criteria to assess model application requirements on the one hand and model type characteristics on the other hand are

- consistency (accuracy compared to reference data),
 - versatility (adaptability to new data, maintainability, application range),
 - bandwidth (frequency range),
 - execution rate (required update rate in terms of numerical stability),
 - complexity (of the propulsion system for example in terms of number of variable geometry control parameters) and
 - operation regime (flight conditions and power lever settings the model has to cope with),
- see [1] for reference.

STATE-OF-THE-ART APPROACH IN AIRCREW TRAINING FLIGHT SIMULATORS

From the modeling concept options available a state-of-the art approach has emerged as the best compromise between the competing requirements a flight simulator application imposes on the PSSM. Most flight simulators having been developed during the last couple of years are based on a modular approach from sub-models for the engine thermodynamics, the electronic engine control (EEC, DECU), the sensor system and the actuator system. The sub-models themselves are usually based on medium accurate modeling techniques which allow for fulfilling the real-time capability requirements. For example for engine thermodynamics often piece-wise linear state-space sub-models are used, which represent the thermodynamic processes in the engine with a set of matrix equations, the matrices themselves often

being referred to as so-called ABCD matrices, see [2]. As an example the engine model used for control system development purposes in the EJ200 project follows this kind of modeling approach.

Despite the obvious advantages of being real-time capable and providing at least medium consistency level there are major drawbacks of this approach.

As far as the customer of the PSSM is concerned the only medium consistency might be tolerable in the first place as long as overall engine performance data as e.g. thrust or cockpit indications such as spool speeds are of major interest. On the other hand any inconsistency of the PSSM compared to engine manufacturer (OEM) provided propulsion system data based on non-real-time simulation models, e.g. steady-state and transient performance customer decks, will give reason for distrust of the customer towards the PSSM. When it comes to modeling sub-idle engine operation (start, shutdown, windmilling) or instructor induced malfunctions, both being important issues during aircrew training, the consistency and versatility of piece-wise linear models is no longer appropriate to meet the requirements. This holds especially true for complex propulsion system configurations with various variable geometry control parameters and/or afterburner configurations as for example modern fighter aircraft propulsion systems as the EJ200.

As far as the supplier is concerned usually big effort is necessary to establish the tracking parameters the ABCD matrices have to be dependent on in order to allow for somewhat sufficient accuracy for all flight conditions, installations, power lever settings and malfunctions. For the validation of the model a comparably big effort is necessary, too, because the artificial propulsion system model being mainly a set of matrices has to be validated for its own for all possible boundary conditions including malfunctions. Last but not least maintaining the PSSM throughout its life cycle imposes a major effort because already slight updates in the engine model status require a fully new set of ABCD matrices to be released to the customer.

ADVANCED CONCEPT

The main driver of the advanced PSSM concept discussed in this paper is to reduce the shortcomings of the state-of-the-art approach for both the customer and the supplier.

The modular approach based on subsystems is thought to be appropriate and therefore does not require any changes. In contrary, for modeling the subsystems the software applied by the engine manufacturer during the propulsion system devel-

opment should be used to the highest extent possible in order to minimize the effort for establishing, validating and maintaining the PSSM. The concept even includes the idea to build subsystem models as close as possible along the real hardware of the engine, the actuators and the sensors (physical models) in order to gain the malfunctions from the model with the lowest additional effort possible. Of course automatic code generation tools should be used as far as possible to simplify and speed up the process of providing new deliverables, updates, bug fixes, etc., and to significantly reduce the error rate.

The key benefit of the advanced PSSM concept is the fact that it is inherently consistent when compared to other OEM provided propulsion system data and even a lot of malfunctions are inherently imbedded in the model. This is true not only for overall performance parameters such as thrust but also for any internal engine or other subsystem parameter the customer might be interested in, as for example pressures and temperatures at various stations in the oil system. Furthermore the versatility is much higher and in turn the effort to establish, to validate and to maintain (including upgrades) the PSSM is much lower because it is built directly from software code used at the OEM anyway. Most of this code has been validated already during the development program.

Of course the advanced PSSM concept yielding major improvements in consistency and versatility has to meet the requirement of real-time capability, too. Thus a major prerequisite is that the sub-models are capable to deliver the calculation results with the given update rate in a numerically stable manner. This holds true for the whole flight envelope, installation and operating range from ground start up to maximum power including all steady-state and transient conditions the real propulsion system might envisage (windmill e-light, auto relight, shut down, various malfunctions, etc.). Both the calculation time and the numerical stability especially in the sub-idle operating range impose some challenges to the subsystem models which have to be addressed carefully.

OVERALL PSSM LAYOUT

ASTA REQUIREMENTS

The PSSM is part of the overall aircraft flight simulator (ASTA) and thus has to cope with the requirements set by the latter. Besides a rather long list of detailed demands the top level ones are listed in the following to provide a profound understanding for the chosen layout:

- The PSSM has to be built from two engine models acting independently from each other.

- The PSSM has to communicate with all the other ASTA subsystems being relevant for propulsion system functionality.
- The interface between PSSM and ASTA is provided by ASTA in terms of a so-called wrapper which is ADA coded.
- The PSSM has to cope with the so-called Simulator Specific Functions (SSF) which provide special functionality for training purposes. Those functions include malfunctioning of the propulsion system as well as trimming, quick start, flight and total freeze, etc.
- In transient mode the update frequency set by the ASTA requirements list is 70 Hz.

OPERATING MODES

As for the operation of the PSSM two basically different modes have to be provided:

- A transient mode which is used during flight training requiring real-time behavior of the PSSM with an update frequency of 70 Hz.
- A steady-state mode which is used during initialization, quick start, trimming and flight or total freeze. In this mode less demanding update rates have to be fulfilled.

For the transient mode the PSSM is built from several detailed sub-models used for the simulation of

- the EJ200 engine based on a full thermodynamic engine model, including the intake and afterbody effects,
- the Digital Engine Control Unit (DECU),
- the sensors and
- the actuators consisting of the Main Fuel Metering Unit (MFMU), the Afterburner Fuel Control Unit (ABFCU), the Variable Inlet Guide Vanes Control Unit (VACU) and the Nozzle Area Control Unit (NACU).

Figure 1 gives a schematic of the PSSM layout used in transient mode including all the relevant sub-model blocks as well as the input and output interface blocks.

The steady-state mode is required during all simulator operation when a steady-state trimmed flight condition (which may also be SLS engines out condition) is initialized. During (in-flight) trimming the aircraft system simulation is looking for the engine power lever position providing the trimmed thrust for the given flight condition. To this end the PSSM has to deliver the steady-state thrust for a given flight condition and power lever position without simulating the transient history to arrive there.

The PSSM layout for the steady-state mode consists of the EJ200 thermodynamic engine model including the intake and afterbody effects and

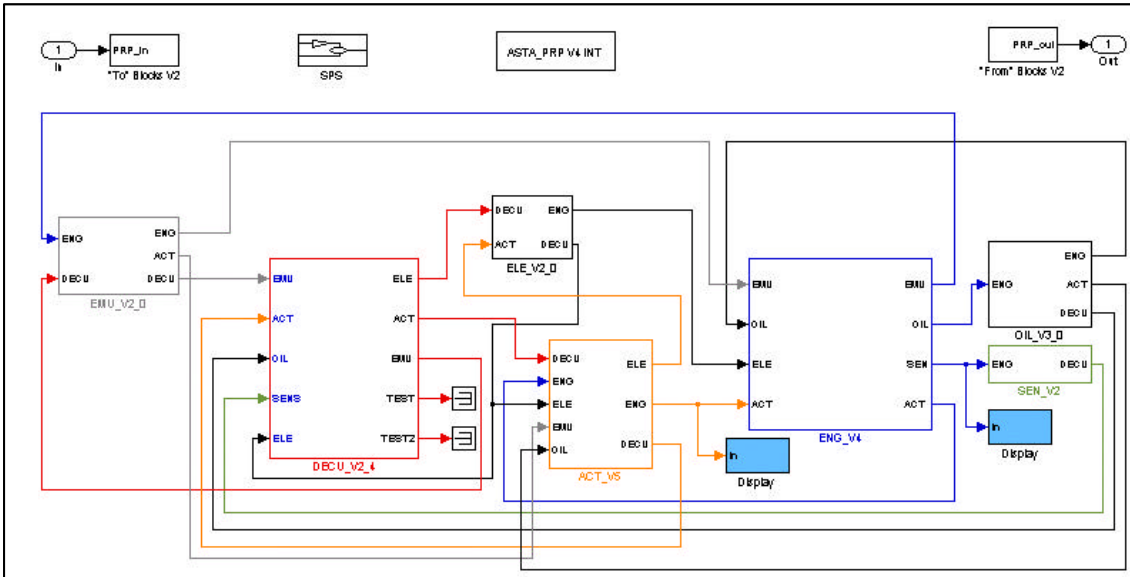


Figure 1: Propulsion system

including the steady-state functionality of the DECU, the fuel, oil and hydraulic system.

INTERFACE TO ASTA

The interaction between flight simulation and propulsion system simulation is assured via a special real-time environment provided by ASTA. The relevant ASTA sub-models the PSSM has to communicate with are

- the aircraft system (AFCSS),
- the fuel system,
- the synthetic environment (flight conditions),
- the secondary power system (SPS),
- the cockpit system,
- the electrical system and
- the instructor panel.

The PSSM interface to the ASTA real-time environment is built by a PSSM wrapper (ADA language code) which communicates with the ASTA environment via shared memory with an update rate of 70 Hz. The wrapper itself defines the interface to the C-coded PSSM which at its top level provides a so-called SSF handler which in turn addresses the PSSM transient or steady-state operating mode depending on the task scheduled by the wrapper. The ASTA – PSSM interface is illustrated in Figure 2.

As mentioned above the update rates in steady-state mode are less demanding than 70 Hz because the simulator is not in the actual training mode for this task. Nevertheless the PSSM wrapper is still calling the PSSM with a frequency of 70 Hz. In order not to over-engineer the PSSM in steady-state mode a special threads and semaphore functionality has been included in the SSF handler and in the PSSM wrapper, allowing for lower update rates in steady-state mode.

MODEL GENERATION PROCESS

With the exception of the thermodynamic engine sub-model, all the PSSM sub-models used for the transient operating mode are based on the Matlab/Simulink® software tool family. The Simulink® toolbox has been used to generate the sub-models and to design the data flow. Based on this graphical model a real-time capable sequence of C code has been created using the Real Time Workshop® (RTW) environment. As for the thermodynamic engine sub-model existing FORTRAN code has been included into the overall system layout via

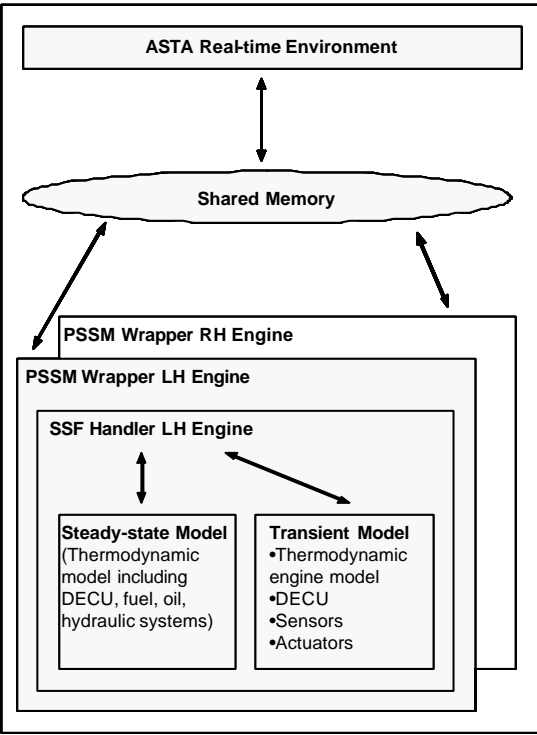


Figure 2: ASTA – PSSM interface

the S-Function interface provided by Simulink®. For the steady-state operating mode a pure FORTRAN-coded model is used. The FORTRAN code is identical to the EJ200 customer performance deck.

The SSF handler as the PSSM top level element and interface to the PSSM wrapper is C-coded by hand.

PSSM SUB-MODELS

DECU MODEL

The DECU is modeled as a separate sub-model rather than being included within the engine sub-model as it is the approach in other engine simulators.

The DECU sub-model is derived directly from the EJ200 Software Requirements Specification (SRS) document and therefore no hardware parts are implemented in this sub-model (see actuator model). The relevant DECU SRS requirements are copied one-to-one into Simulink® block diagrams. The main functionality included in the DECU sub-model covers

- the control loops for the variable geometry (fuel valves, nozzle and VGV actuators),
- the model check for actuator position,
- the generation of cockpit indications,
- the state logic for main engine, afterburner, start/shutdown torque motors, ignition unit, etc.,
- the detection of special events during engine operation, e.g. flame-out, surge, stall, buzz, screech,
- the generation of recovery actions during abnormal operation and
- the generation of milbus signals to the flight control system.

With this functionality all the possible operating modes including normal operation, ground and flight start, shutdown and failures are covered.

The constants and tables used in the sub-model have been transferred directly from the original DECU ADA code into Simulink® using import tools to minimize conversion errors.

The testing process involves two stages. In level one a code walk-through analyzing the implementation of the software requirements is performed. Level two consists of a validation against test case results coming from the DEAR (DECU-EMU Acceptance Rig) model. This rig model is a linear model used in control software development and testing. It includes engine and actuators/sensors table reading models. The validation is based on the comparison between results from the real DECU + DEAR model and the Simulink® DECU model + DEAR model.

The state logic included in the DECU has been implemented using the Stateflow® toolbox from Matlab/Simulink®. This tool allows to simulate the transitions between the different engine states that occur while the engine is running.

ACTUATORS AND SENSORS MODEL

Sub-model generation: For the development of the actuator sub-models, the starting point was a set of full physical models coded in FORTRAN, where all the piping, pressure losses and devices were included. The model generation process consists in the proper translation from FORTRAN code into Simulink® environment. Typical elements included in the models are

- torque motors and servo-valves,
- metering valves and emergency valve (for the fuel metering unit),
- pumps,
- start and shutdown valves and
- pressure losses.

Some of the less important elements, e.g. internal flows and pressure drop regulation devices have been removed in the Simulink model, since they have only a minor impact on the accuracy of the sub-models.

DECU hardware/software interface has also been modeled within the actuators sub-system. Signal conditioning and noise suppression filters with characteristic frequencies higher than 70 Hz have been removed to simplify the model and due to the necessity to capture their dynamics. The interface with other sub-systems has been defined mainly as follows:

- Inputs: demand positions from DECU (they will be converted into currents to drive the torque motors) and pressures from other sub-systems.
- Outputs: valve, VGV and nozzle positions, flows and pump power consumption.

For the fuel system models further simplifications to develop the simplified models have been made:

- With the current from the DECU hardware/software interface and the delta pressure the flow is obtained by using the servo valve hydraulic characteristic (linear relation).
- With the flow entering the metering valve and its area the speed of the valve is derived.
- By integrating the speed of the valve its position to feed back the DECU is obtained.
- The fuel flow through the manifold is calculated from table reading dependant on the metering valve position.

Most of the functionality of the Engine Monitoring Unit (EMU) such as incident diagnosis and failure detection is not required for ASTA purposes. Therefore, only a simplified model for en-

gine rating updates and another for handling of vibration malfunctions have been included.

Challenges: In the case of the fuel system sub-models (MFMU and ABFCU) instabilities occurred when running at 70 Hz due to a higher characteristic frequency (around 100 Hz) of the true physical model. The main problem was the constraint imposed by ASTA that forces to run the whole PSSM to a single sample rate of 70 Hz.

The solution was to call the actuator system twice in each time step meaning to run to a double sample rate of 140 Hz. This has been achieved using embedded code included into Matlab/Simulink® environment with the so called “Custom Code Blocks” functionality.

The result of this measure can be seen in Figure 3, where the blue line corresponds to an ABFCU call at 70 Hz and the red line corresponds to a call at 140 Hz. The DECU Hardware/Software interface sub-model generates the driving currents for the actuator models, these currents being proportional to the error between demand position and real position. As long as these currents are not updated at 140 Hz the actuator response in terms of fuel flow will not be satisfying. This is the reason why this model was included as a sub-model within the actuators sub-system so it could be run also to a double sampling rate.

Sub-model validation: The simplified models have been validated against the high accuracy full physical models and real data from whole engine test bed.

As an example, the Figures 4a, 4b and 4c are showing the MFMU main metering valve position for an acceleration maneuver from idle to maximum reheat rating and back to idle. The position given by the model (red line) follows both the test bed demand and real position (blue and green line respectively). The small differences between test bed demand and the real position are due to the exclusion of integral parts within the control loop.

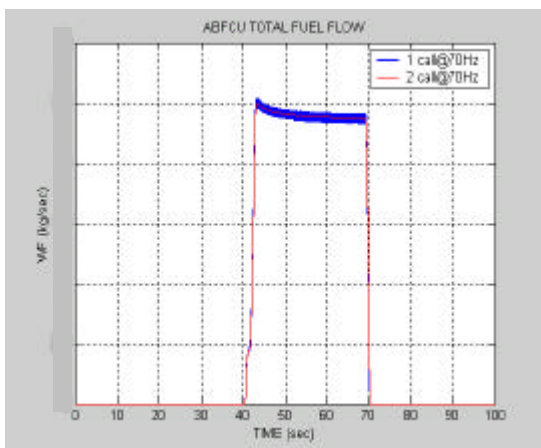


Figure 3: ABFCU instability

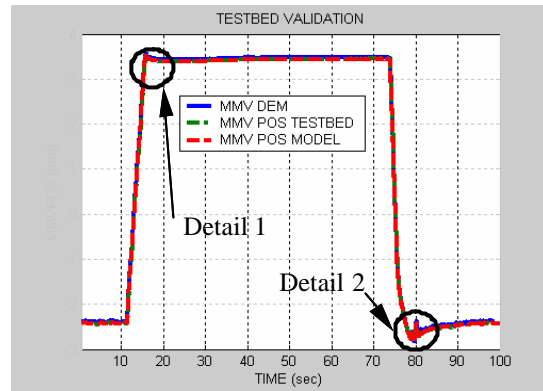


Figure 4a: Main metering valve position

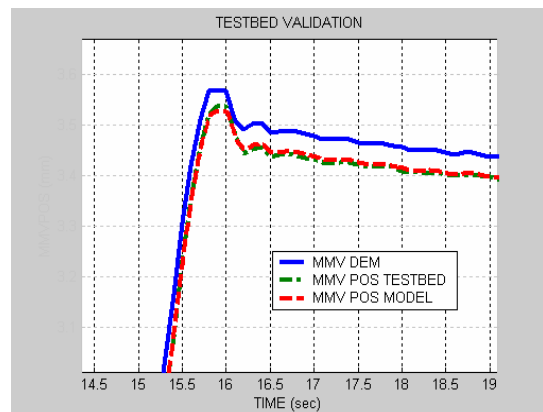


Figure 4b: Detail 1

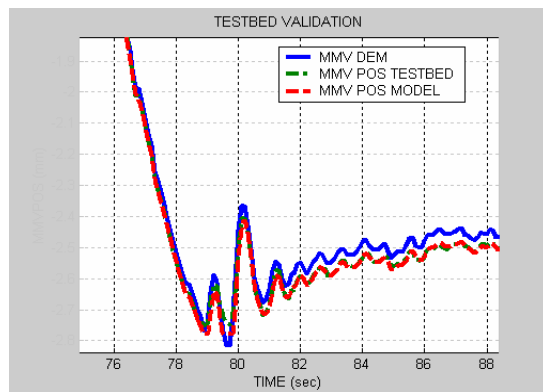


Figure 4c: Detail 2

THERMODYNAMIC ENGINE MODEL

Engine sub-model functionality: The thermodynamic engine sub-model covers the air breathing engine sub-model and therefore all the hardware parts that are directly contributing to the aerothermodynamic cycle and to the thrust production of the engine. The sub-model is embedded in the overall PSSM architecture and communicates exclusively via specified data interfaces with the SSF handler and the other PSSM sub-models. In order to address the PSSM functionalities, the engine sub-model features two operating modes: a

steady-state mode used during trimming of the components. At certain points in the calculation

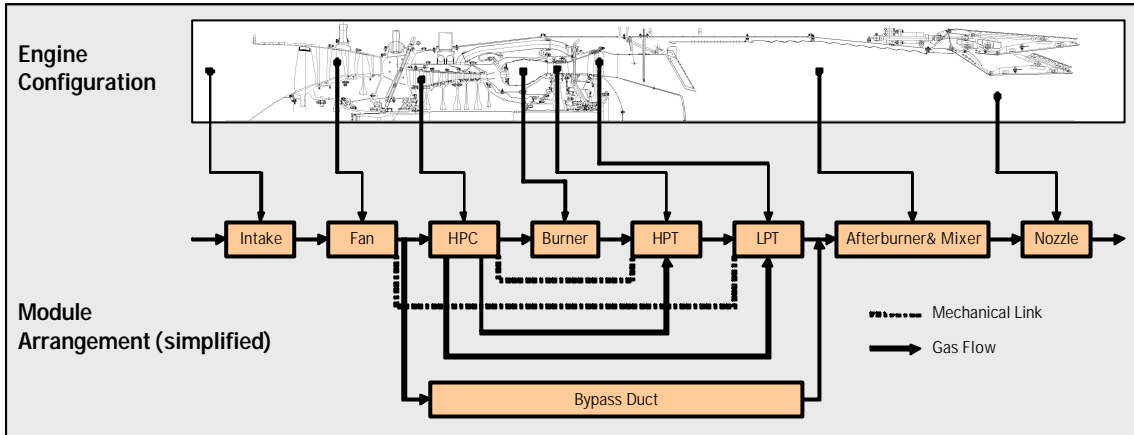


Figure 5: Simplified engine sub-model schematic

aircraft model and quick-start procedure and a transient operating mode used during the true transient simulation.

When running in steady-state mode the engine sub-model is asked to return steady-state engine data for a given power lever command and for a given set of atmospheric conditions. Since the other PSSM sub-models are switched off during trimming and quick-start, the engine sub-model has to incorporate an internal set of steady-state DECU schedules as well as simplified steady-state models of the fuel, oil and hydraulic system. These models are already included in the status model delivered to the customer, for example in steady-state customer performance decks.

When running in transient mode, the PSSM DECU and actuator sub-models are providing the control inputs such as fuel flows, VGV and exit nozzle area to the engine sub-model which is running as a state-space model. In this mode the simplified internal DECU and accessories models are switched off. Together with the atmospheric conditions and the state inputs the engine sub-model is producing the outputs requested by the other PSSM sub-models as well as the state derivatives.

Engine modelling technique: The engine sub-model is based on the MTU performance synthesis program used for engine development purposes (see [6]) and includes all the data reflecting the current EJ200 engine standard. The engine is described on a component level, each component being represented by a specific source code module, as shown in Figure 5. In order to avoid time consuming iterative harmonisation of the available aircraft inlet mass flow and the demanded engine mass flow, the source code parts describing the behaviour of the aircraft inlet have been included in the engine sub-model.

Following the gas path, the calculation starts with the front components and proceeds to the rear

some parameters must be estimated, these parameters being balanced by an equal number of matching conditions. Fitting the estimated parameters to the matching conditions results in solving a system of non-linear equations which is done by using iterative techniques.

To select the engine sub-model operating mode (steady-state or transient) simply switching the iteration scheme is necessary. In transient mode only the effect of rotor inertia is included, this phenomenon being considered as the major contributor to the transient engine behaviour.

Challenges: Using an off-the-shelf full thermodynamic engine sub-model in a simulator environment presents significant advantages, e.g. low model generation effort, high level of model consistency and high versatility. On the other hand, this application also implies special requirements in terms of

- real-time capability (in transient mode only),
- full operating range capability and robustness of calculation as well as
- achievement of the simulator specific functions.

As presented in [3] and [5] several techniques may be applied to generate models that are meeting these requirements. Since one of the ASTA objectives has been to avoid as much as possible supplemental work related to the special simulator application, all the effort to achieve the requirements has been focussed on the improvement of the performance synthesis program itself.

Real-time capability: According to [1], the time consumption of a thermodynamic engine model is mainly due to the detailed calculations within each component, to the computation of the Jacobian matrix related to the system of non-linear equations and to the iteration loops necessary to find the system solution. The efforts to reduce calcula-

tion time for one transient operating point in the ASTA application have therefore focussed on these three items.

Concerning the reduction of the calculation time per loop through the engine thermodynamics, compiler optimisation options, code profiling techniques derived from automated customer deck generation procedures and direct run-time memory access for data exchange instead of time consuming file I/O have been used.

A significant amount of calculation time may be spared if the generation of the Jacobian matrix is handled more efficiently by using proper update techniques (see [4]) and by simply taking over the matrix from one operating point to the next. During transient operation this is reasonable since the changes of engine sub-model control and state inputs are small between time steps, due to the inertia of the actuators and spools and due to the control of engine acceleration/deceleration rates by the engine control unit.

Finally, the number of iteration loops must be limited in order to return data with the required update rate. Experience has shown that 5 iteration loops per operating point are sufficient to produce a result well in line with the ASTA accuracy demands.

As presented in [8] the combination of these techniques, all of them implemented in the basic performance synthesis program, has contributed to successfully run the engine sub-model in real-time with an update rate of 70 Hz on a 833 MHz ALPHA processor with an accuracy as high as in normal performance work in the office.

Full operating range capability: Beside the normal operating range between idle and maximum reheat rating the ASTA engine sub-model must also be able to simulate the engine behaviour under conditions as for example engine start and shut-down or windmilling and windmill relight. When it comes to malfunctions, even hung or hot start events have to be modelled realistically. These special operating conditions are very demanding with regard to the convergence of the calculation and special effort on both, the performance synthesis program as well as on the data fed into the model has shown to be necessary to provide a robust working engine sub-model.

As mentioned in [7] the improvements on the computer program side have focussed on the ability to perform calculations and internal iterations at spool speeds and mass flows approaching zero, e.g. by using torque based compressor and turbine maps. Special care has also to be taken on the burner ignition and extinction boundary as well as on the burner efficiency model during engine start

since these items have a significant impact on the fidelity of the simulation.

The improvements on the source code side have been accompanied by an enlargement of the data base including suitable extrapolation of compressor and turbine maps in the low speed range as well as detailed test data analysis of engine start events.

Simulator specific functions: The engine sub-model is playing an essential role in the PSSM initialisation procedure and a special sequence of calculations is necessary to address the simulator start and reset requirements. Due to time limitations, especially in case of engine quick-start or trimming of the aircraft model, some attention has also to be paid to the duration of the initialisation procedure.

The initialisation starts with a single operating point during which the engine sub-model is running in steady-state mode. The purpose of this calculation is to initialise or refresh the run-time memory data set needed by the engine sub-model and to return the outputs according to the power lever position. This position corresponds to the commanded quick-start regime or to the first trimming point setting.

If trimming is requested, the process continues with a sequence of steady-state operating points being commanded by the aircraft model. During trimming the power lever command may change significantly from one operating point to the next, depending on the thrust demand for aircraft equilibration. Simply taking over the Jacobian matrix and the estimated parameters as in transient mode is no longer an advantage and to ensure rapid convergence of the calculation the pre-calculated guess parameter tables already available in the status model have been used.

The main initialisation process ends with a final operating point aiming at the pre-calculation of the Jacobian matrix used in transient mode. Since the matrix calculation takes more than 5 loops through the engine thermodynamics it is vital to perform this calculation before the update rate limited transient simulation begins.

Once the final initialisation point is processed, most of the data needed for the initialisation of the other PSSM sub-models are available and the transient simulation can be started. However, switching the engine sub-model operating mode also means to use the external DECU, actuator and sensor models instead of the internal models. Any difference between these two model sets may lead to output inconsistencies in the transition phase and some care for the harmonisation of both model sets is necessary.

Model generation and maintenance: The status engine model available at the beginning of the ASTA project was only covering the normal operating range between idle and maximum reheat rating. Therefore, in a first step, this model (basic performance synthesis program source code as well as EJ200 specific data set) has been extended to the sub-idle range. These modifications have then been fed back to the status model (see [7]).

In a second step, the ASTA real-time model has been created with a fully automated procedure similar to that used for the generation of costumer decks. The main purpose of this procedure is to produce a piece of source code that contains all the FORTRAN modules necessary to perform the thermodynamic and numeric calculations and to store the engine specific data in form of built-in data statements. Running this procedure and final scrambling of the source code takes no longer than one hour and is fully automated.

As a final step, the real-time code is included in the PSSM development environment. Standard test cases covering the required operating range and functionality are processed and the results visualized in prepared plots. These tests are focussing on the robustness as well as on the accuracy of the simulation.

Maintenance of the engine sub-model is easy since it requires only an update of the status model, all the further steps being more or less automated. Because the real-time model is using the same source code and data set as the status model, special testing is necessary only for new features.

RESULTS

PSSM VERSATILITY

The versatility is describing the ability of the PSSM model to cover the complete engine operating range. To demonstrate this, Figure 6a is showing the power lever history for a typical test case sequence including

- a stabilised windmilling operation at Mach number 0.08 (t = 0 to 10 seconds),
- an engine start to idle (t = 10 seconds),
- a stabilisation at idle rating (t = 10 to 50 seconds),
- a throttle push to maximum reheat (t = 50 seconds),
- a stabilisation at maximum reheat (t = 50 to 60 seconds),
- a shut down (t = 60 seconds) and finally
- a run down of the engine at Mach number 0.08 (t = 60 to 120 seconds).

In this example the Mach number has been limited to 0.08 in order to keep a minimum spool speed to

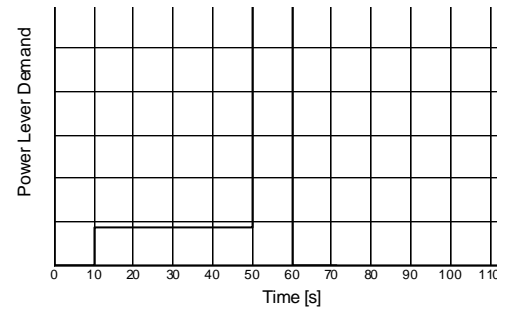


Figure 6a: Power lever demand history

ensure convergence of the thermodynamic engine model.

The PSSM response in terms of high pressure spool speed is shown in Figure 6b. During windmill operation the spool speed is close to zero. As soon as the power lever is moved to idle position the engine start sequence begins. The starter motor is connected thus delivering the necessary torque, the igniters are switched on and fuel is injected. Due to the low combustion efficiency in the sub-idle range the high pressure spool accelerates slowly. After a stabilization at idle and maximum reheat rating the power lever is moved to shut position again, leading to a deceleration of the high pressure spool.

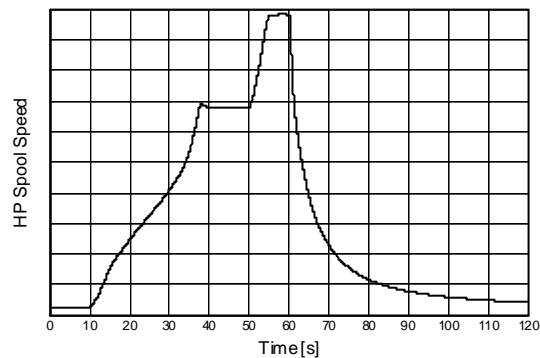


Figure 6b: HP spool speed

MALFUNCTION HANDLING

With the so called malfunctions the instructor can set an abnormal operation to train the pilot how to react in a real flight situation. There are two types of malfunctions:

- Recoverable: The pilot can perform some recovery actions according to the flight crew check list in order to go back to a normal flight operation.
- Non-recoverable: The pilot can not do anything to go back to a normal flight operation.

Each sub-model has (or not) its own malfunctions, some examples are:

- Recoverable: Afterburner extinction simulates an afterburner flame out, as shown in the Figures 7a and 7b. The ABFCU reacts providing zero fuel flow to the engine. The pilot must throttle back to dry operation to go back to a normal engine running operation.

The test case is shown for a 10000 ft Mach 1.0 flight condition. The malfunction is set at 1 second, the pilot reacts at 4 seconds pulling down the throttle to dry condition. At 8 seconds the pilot reselects the afterburner but

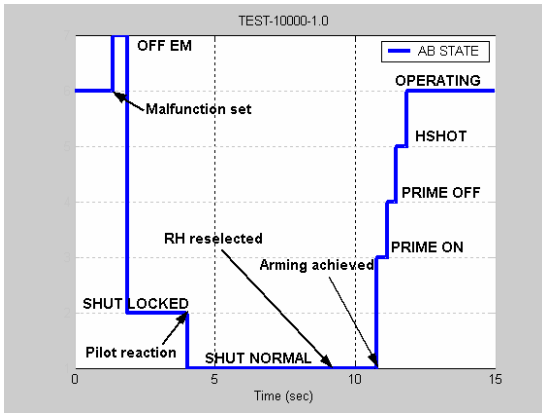


Figure 7a: Afterburner state

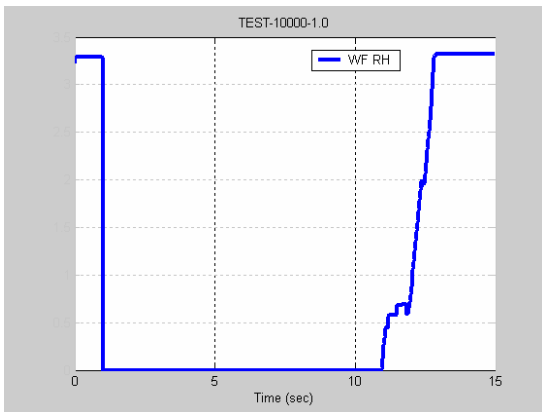


Figure 7b: Reheat fuel flow

it does not light up until the arming condition is achieved.

- Non-recoverable: TBT (Turbine Blade Temperature) dual line failure. This malfunction generates the DECU reversionary control actions when both TBT lines fail. In Figures 8a and 8b a test case is shown for a 10000 ft Mach 1.0 flight condition where the malfunction is set at 1 second. Therefore the AB goes to an emergency off state, at 4 seconds the pilot pulls down the throttle to taxi and at 8 seconds tries to reselect AB again unsuccessfully. The engine behavior is as expected, AB is inhibited, low pressure spool speed limit is reduced to a safety value, as can be seen in Figure 8c, and TBT control loop is deactivated.

PSSM CONSISTENCY

A major requirement for the PSSM consists in returning data with the specified accuracy. The

accuracy is determined by comparing the PSSM model output with the output generated by a model with higher fidelity or with data collected from engine tests.

Since the PSSM model is inherently consistent with the high fidelity models used in engine development, the accuracy can be demonstrated directly with engine test data. Figure 9 shows the PSSM response to a change of the power lever demand for three typical maneuvers between idle and maximum dry rating:

- A slam deceleration (red line),
- a slow deceleration (green line) and
- a slam acceleration (black line).

The plain lines are corresponding to the test data, the dotted lines are giving the simulation result.

Whereas the accuracy is high for the deceleration maneuvers, the acceleration maneuver features a gap at the beginning of the stabilization at $t = 7$ seconds. Since the difference between measure-

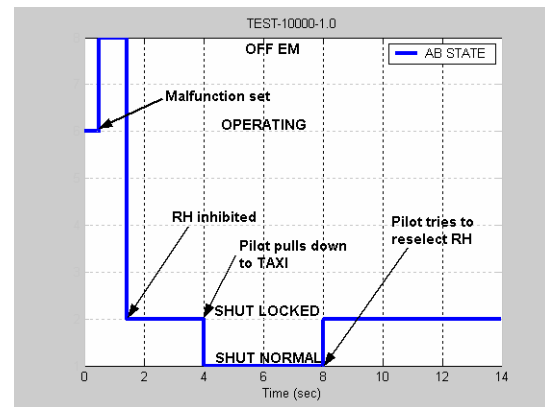


Figure 8a: Afterburner state

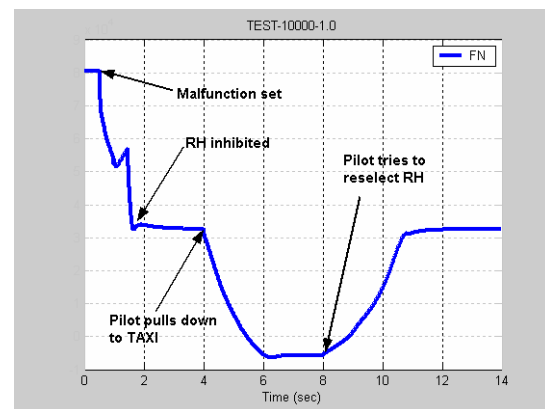
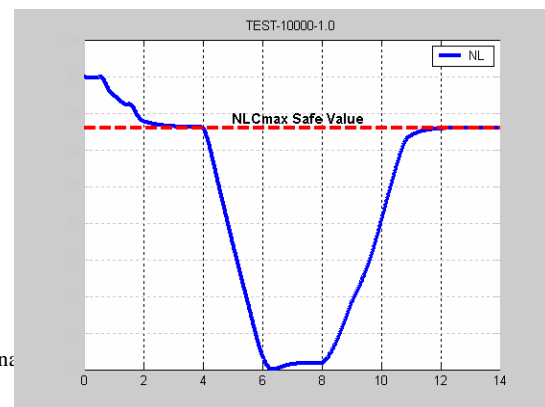


Figure 8b: Net thrust



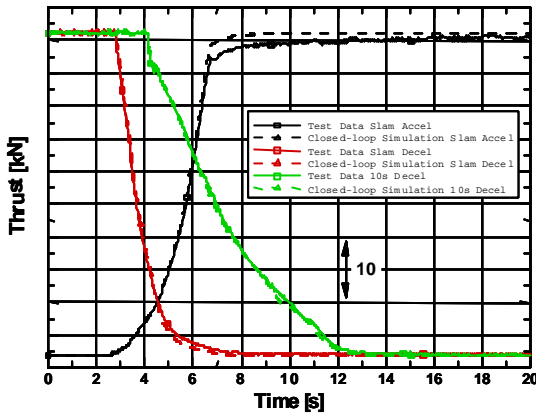


Figure 9: PSSM simulation versus test data

ment and simulation tends to disappear after some time, it is supposed that this gap is due to some heat soakage phenomena that are not included in the actual PSSM model. Although the accuracy displayed in Figure 9 is already in line with the requirements and the gap is hardly to be sensed by the trainee, heat soakage will be taken into account in later model releases in order to further improve the fidelity.

CONCLUSIONS

The use of a modular approach to develop a simulation model of the EJ200 engine for the Eurofighter/Typhoon has led to a software model which provides a high level of realism for the purpose of pilots training.

The model consists of a variety of interconnected software modules representing physical entities of the real engine. This approach allowed an efficient development of the software, by sharing resources between the different companies being involved in the development. Furthermore, the process of developing the modules by mimicking the physical/functional properties of the real engines' components as closely as possible ensures that the resulting simulation software represents engine behaviour with the required level of accuracy. The combination of actuator and engine control models being developed in Matlab/Simulink® with a full thermodynamic model of the engine itself reduced development time by making use of existing models as far as possible. Furthermore, the modular approach allows for easy upgrades of the simulator in the future, as new versions of the engine are introduced into fleet service.

The approach presented in this paper is demanding in terms of computational requirements, therefore special care has been taken to achieve real-time capability while maintaining numerical stability, both for the thermodynamic model and the fuel system model. Matlab/Simulinks® custom code blocks allowed to repeat function calls in order to

double the update rate of stability-critical components of the fuel system. The thermodynamic model achieved real-time capability by employing special techniques to reduce the computational overhead required for the update of its Jacobian matrix.

The result of all these efforts is a simulation model of the EJ200 engine which provides a very realistic simulation of the real engines' behavior within the environment of the ASTA, offering the full range of capabilities required to train both student and experienced pilots for their specific needs.

ACKNOWLEDGEMENTS

The authors wish to thank NETMA for the permission to publish the paper.

REFERENCES

- [1] NN, 2001, "Real-Time Modelling Methods for Gas Turbine Engine Performance", SAE AIR-4548-Rev.A.
- [2] N. Sugiyama, 1992, "Derivation of ABCD System Matrices from Nonlinear Dynamic Simulation of Jet Engines", AIAA-92-3319.
- [3] W. Visser, M. Broomhead, J. van der Vorst, 2001, "TERTS, a Generic Real-Time Gas Turbine Simulation Environment", ASME 2001-GT-0446.
- [4] J. E. Dennis, R. B. Schnabel, 1996, "Numerical Methods for Unconstrained Optimisation and Nonlinear Equations", SIAM Classics in Applied Mathematics.
- [5] C. Crainic, A. Thompson, R. Harvey, 1997, "Real Time Thermodynamic Transient Model for Three Spool Turboprop Engine", ASME 97-GT-223.
- [6] J. Kurzke, 1992, "Calculation of Installation Effects within Performance Computer Programs", AGARD Lecture Series 183.
- [7] M. Bauer, C. Riegler, H. Schulte, 2003, "Validation of a Mixed Flow Turbofan Performance Model in the Sub-Idle Operating Range", ASME 2003-GT-38223.
- [8] M. Klobe, C. Riegler, K. Salchow, 2003, "Real Time Capability for a Full Thermodynamic Engine Performance Computer Program", to be presented at the 9th CEAS European Propulsion Forum, Rome.