JOURNAL OF AIRCRAFT
Vol. 46, No. 2, March–April 2009

Engineering Notes

ENGINEERING NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Performance, Usage, and Turbofan Transient Simulation Comparisons Between Three Commercial Simulation Tools

Kirk J. Gomes,* Konstanty C. Masiulaniec,† and Abdollah A. Afjeh‡
University of Toledo, Toledo, Ohio 43606

DOI: 10.2514/1.38902

I. Introduction

THE development of aircraft propulsion systems is dependent, to a great extent, on the ability of a simulation tool to predict the performance of the propulsion system and its associated controls before the building and testing of a prototype. Computer simulations provide the means for analyzing the behavior and interactions of these complex systems and can also serve as aids in understanding and solving problems that arise after the propulsion system is developed. In this work, three unique simulation tools, Modelica (Dymola), MATLAB/Simulink, and the Numerical Propulsion System Simulation (NPSS) have been used to model a two-spool, two-stream augmented turbofan engine. The results from these tools are compared to the results from DIGTEM [1] (digital computer program for generating dynamic turbofan engine models) which was written in Fortran in 1983. Additionally, a comparison between the three tools is provided in terms of usage and performance.

Modelica is primarily a modeling language that allows specification of mathematical models of complex natural or man-made systems, for the purpose of computer simulation of dynamic systems where behavior evolves as a function of time. Modelica is also an object-oriented equation-based programming language, oriented toward computational applications with high complexity requiring high performance [2].

Simulink is an environment for multidomain simulation and model-based design for dynamic and embedded systems. It provides an interactive graphical environment and a customizable set of block libraries that facilitate design, simulation, implementation, and testing of a variety of time-varying systems, including communications, controls, signal processing, video processing, and image processing. Add-on products extend Simulink software to multiple modeling domains, as well as provide tools for design, implementation, and verification and validation tasks. Simulink is integrated with MATLAB, providing immediate access to an extensive range of tools that allow the development of algorithms, analysis and visualization of simulations, creation of batch processing scripts, customization of the modeling environment, and the definition of signal, parameter, and test data.

The NPSS project is a technology developed by the NASA Glenn Research Center, in conjunction with the U.S. aeropropulsion industry and the Department of Defense and is capable of supporting detailed aerothermomechanical computer simulations of complete aircraft engines. NPSS can realistically model the physical interactions that take place throughout an engine, accelerating the concept-to-production development time and reducing the need for expensive full-scale tests and experiments. At its foundation, NPSS is a component-based object-oriented engine cycle simulator designed to perform cycle design, steady-state and transient off-design performance prediction, test data matching, and many other traditional tasks of engine cycle simulation codes. As with traditional codes, an NPSS engine model is assembled from a collection of interconnected components and controlled through the implementation of an appropriate solution algorithm [3].

II. Engine Description and Mathematical Models of Components

The model being simulated is a two-spool, two-stream, low bypass ratio (bypass ratio of 0.808 at the design point) augmented turbofan engine. Figure 1 shows a schematic representation of that engine with the engine stations numbered. A single inlet is used to supply airflow to the fan. After leaving the fan, the air is separated into two streams. One stream passes through the engine core and the other stream passes through an annular bypass duct. The fan is driven by a low-pressure turbine. The core airflow passes through a compressor that is driven by a high-pressure turbine. Both the fan and the compressor are assumed to have variable geometry for better stability at low speeds. At the compressor exit, engine airflow bleeds are extracted and used for turbine cooling (flow returns to the cycle). Fuel flow is injected in the main combustor and burned to produce hot gas that drives the turbines. The engine core and bypass streams combine in an augmentor duct, where the flows are assumed to be thoroughly mixed and additional fuel is added to further increase the gas temperature (and thus thrust). Finally, the augmentor flow is discharged through a variable convergent–divergent nozzle. The nozzle throat area and exhaust area are varied to maintain airflow and to minimize drag during augmentor operation. The following sections briefly describe the mathematical models [4] of the engine components.

A. Flight Conditions and Inlet

A steady-state representation [1] of a typical inlet recovery is used to provide the proper fan inlet conditions (station 2) for a specific flight condition (station 0).

B. Gas Properties

The fuel is assumed to be JP-4 and curve fits of data [5] are used to compute variable thermodynamic properties.

Received 2 June 2008; revision received 6 November 2008; accepted for publication 8 November 2008.

*Research Assistant, Mechanical, Industrial and Manufacturing Engineering Department, MS312.

†Associate Professor, Mechanical, Industrial and Manufacturing Engineering Department, MS312. Member AIAA.

‡Professor, Mechanical, Industrial and Manufacturing Engineering Department, MS312. Member AIAA.

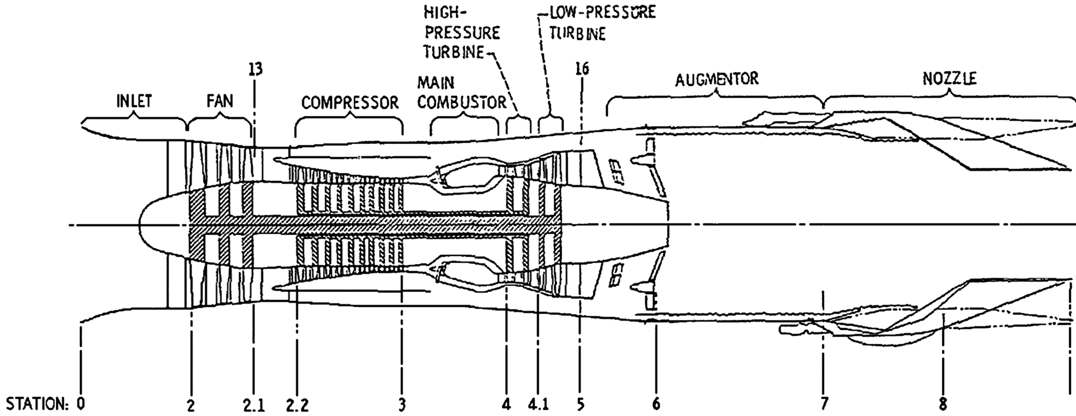

Fig. 1 Schematic of the augmented turbofan engine [1].

C. Fan

Fan performance is represented by a set of overall performance maps and separate maps are used for the tip and hub sections [1]. The maps are assumed to represent fan performance with variable geometry at nominal, scheduled positions. Map-generated corrected fan airflow is adjusted to account for off-schedule geometry effects. A linear interpolation algorithm is used to determine map values.

D. Compressor

A similar procedure is followed for the compressor with overall performance maps (shown in Figs. 2 and 3) being used with a shift in the corrected airflow based on off-schedule values of variable geometry position [1].

E. Bleeds

Flow through the bleed passages is assumed to be choked and both turbine cooling and overboard bleeds are modeled [1].

F. Turbines

Overall performance of the high- and low-pressure turbine is represented by bivariate maps [1]. Cooling bleed for each turbine is assumed to reenter the cycle at the turbine discharge, although a portion of each bleed is assumed to do work.

G. Combustors and Ducts

Total pressure losses are included in the models of the main combustor, bypass duct, mixer entrance, and augmentor. Heat addition associated with the burning of fuel in the main combustor and augmentor is assumed to take place in control volumes V_4 and V_7, respectively [1].

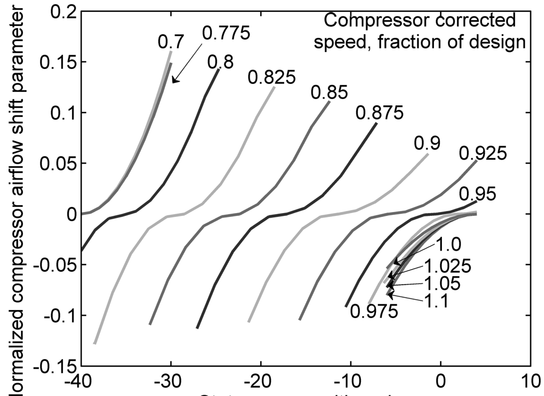

Fig. 2 Effect of variable stator vane position on compressor performance [4].

H. Exhaust Nozzle

A convergent–divergent nozzle configuration is assumed and standard isentropic compressible flow equations are used [6].

I. Intercomponent Volumes

Intercomponent volumes are assumed between the compressor and combustor, between the combustor and high-pressure turbine, between the high-pressure turbine and low-pressure turbine, between the low-pressure turbine and augmentor, between the augmentor and nozzle, and between the fan and bypass duct. These are required to avoid an iterative solution of the equations. Storage of mass and energy occurs in these volumes [1].

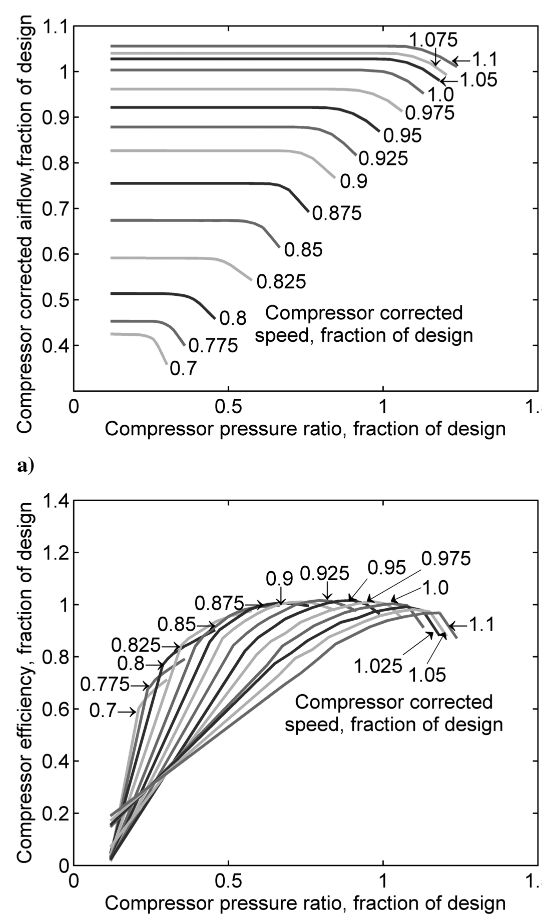

Fig. 3 Compressor performance map with stator vanes at their nominally scheduled positions [4].

J. Fluid Momentum

The effects of fluid momentum are considered in the bypass duct and augmentor duct models [1].

K. Rotor Inertias

Rotor speeds are computed from dynamic forms of the angular momentum equations [1].

L. Correction Coefficients

To balance the engine at the design point, correction coefficients are introduced [1]. If the design point data (which are specified as input) are not exact, incompatibilities between the data and the engine model will result in nonzero derivatives or mismatches between the specified and calculated outputs of the component maps. To compensate for these differences, a "self-trimming" feature is built in through the use of correction coefficients. These coefficients are calculated using the design point values and balance the engine by making the derivative terms become zero and compensating for numerical inconsistencies created when values are interpolated from performance map data. The correction coefficients are then part of the model and are used at both design and off-design points.

The correction coefficients calculated for the dry design point are shown in Table 1.

Table 1 Correction coefficients (CC) for the dry design point

CC	Value	CC	Value
1	1	11	1.00109
2	1	12	1.02671
3	1	13	1.0046
4	1	14	0.99257
5	0.998474	15	1.0017
6	1	16	1.02267
7	1.00419	17	1
8	1	18	1
9	0.988879	19	1.00893
10	1.00243		

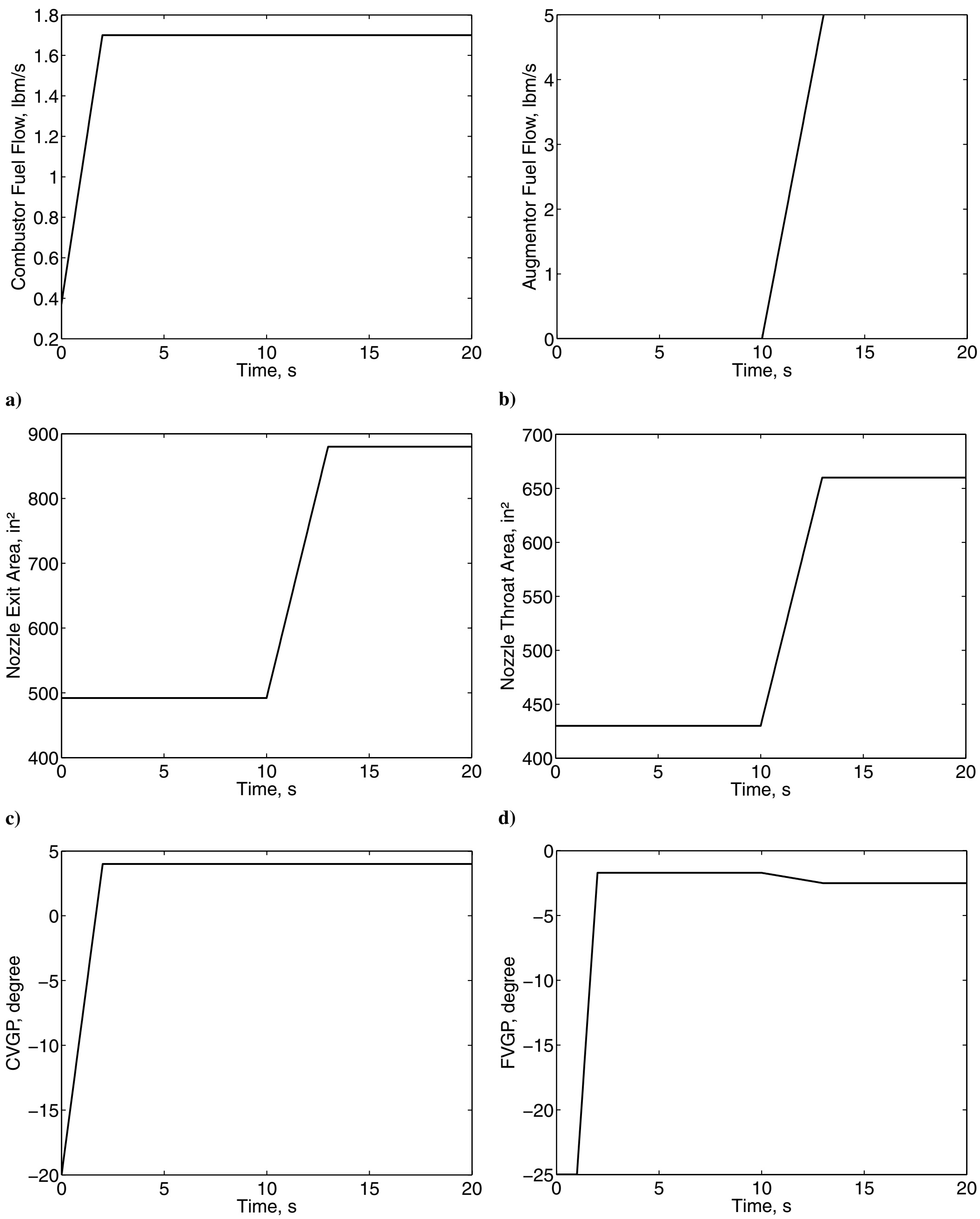

Fig. 4 Transient control inputs.

Table 2 Simulation run time in seconds ($dt = 0.0005$ s)

Solver	Dymola	Simulink	NPSS
Euler	25.6 s	47 s	150.0 s
Default	7.5 s (DASSL)	47 s (Euler)	148.3 s (Gear)

Table 3 Simulation run time in seconds (Euler solver)

Time step in seconds	Dymola	Simulink	NPSS
0.0005 s	25.6 s	47 s	150.0 s
0.00058 s	22.7 s	41.4 s	129.3 s
0.00075 s	——-	——	102.8 s
0.001 s	——	——	79.8 s
0.0015 s	——	——	56.4 s
0.002 s	——	——	43.2 s

M. Transient Control Inputs

During the transient simulation, the values of six inputs are varied over a period of 20 s. These inputs are the combustor and augmentor fuel flow rates, nozzle throat and exit areas, and the compressor and fan variable geometry parameters (CVGP and FVGP). Plots of these inputs are shown in Fig. 4.

III. Simulation Results

The effect of the time step on the simulation run time and the variation of rotational speed, pressure, and temperature in response to the transient control inputs over a period of 20 s are shown in Fig. 5 and Figs. 6–19, respectively. The individual results from Dymola, Simulink, and NPSS are superimposed over the results from DIGTEM. A fixed time step of 0.0005 s and the default solver was

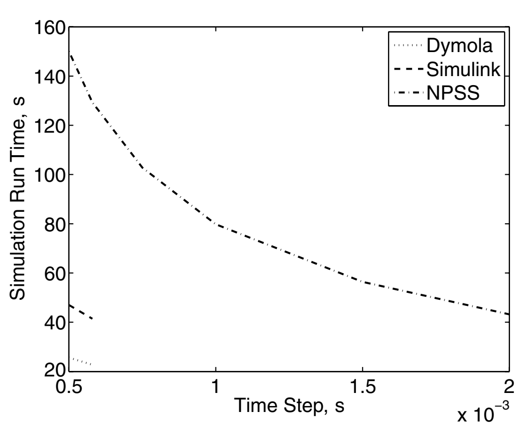

Fig. 5 Time sensitivity results.

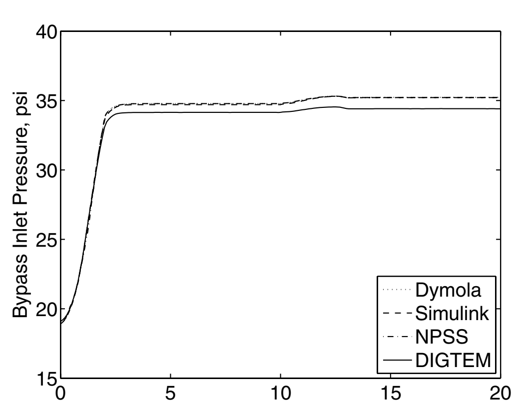

Fig. 6 Variation of bypass duct inlet pressure during the transient.

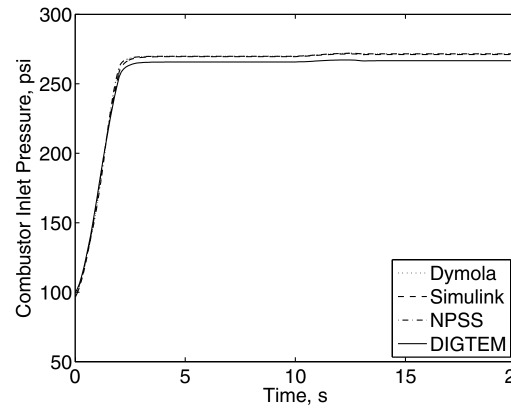

Fig. 7 Variation of combustor inlet pressure during the transient.

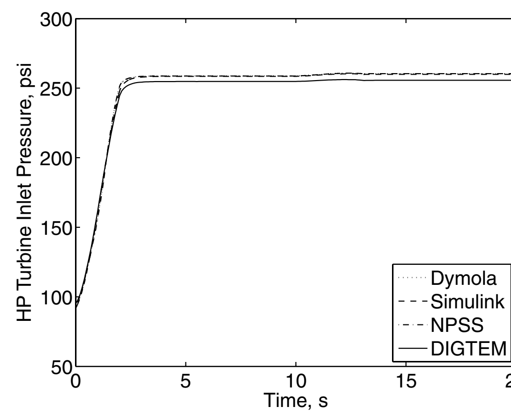

Fig. 8 Variation of high-pressure (HP) turbine inlet pressure during the transient.

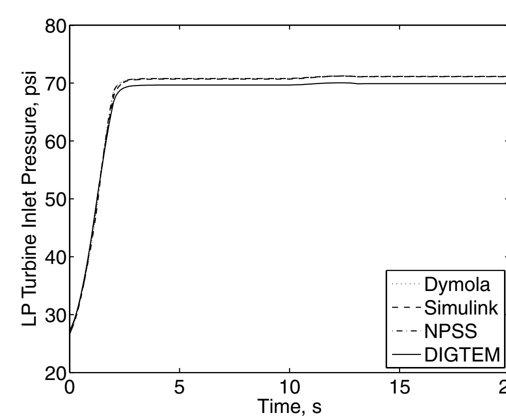

Fig. 9 Variation of low-pressure (LP) turbine inlet pressure during the transient.

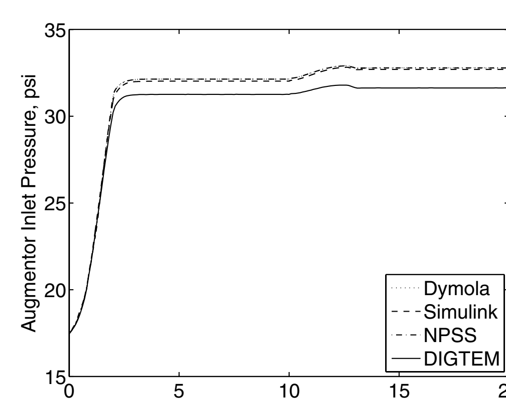

Fig. 10 Variation of augmentor inlet pressure during the transient.

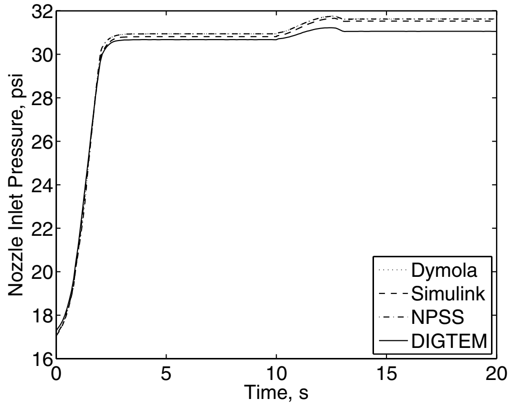

Fig. 11 Variation of nozzle inlet pressure during the transient.

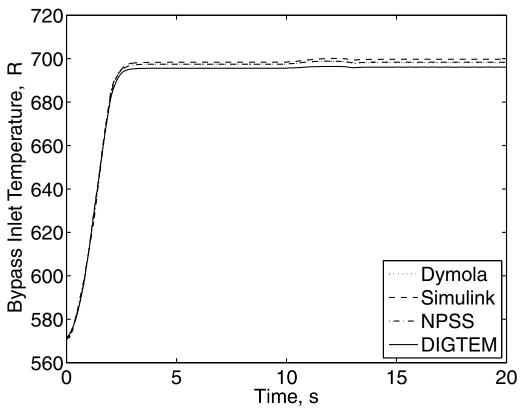

Fig. 12 Variation of bypass duct inlet temperature during the transient.

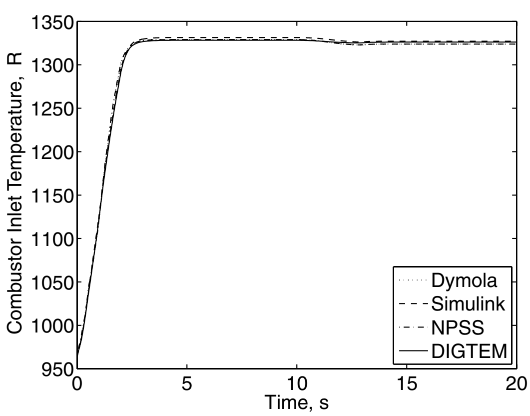

Fig. 13 Variation of combustor inlet temperature during the transient.

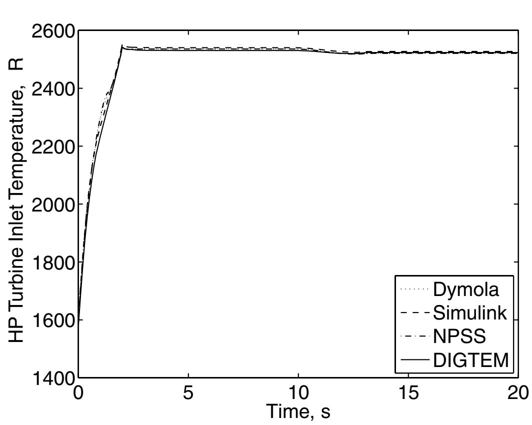

Fig. 14 Variation of high-pressure turbine inlet temperature during the transient.

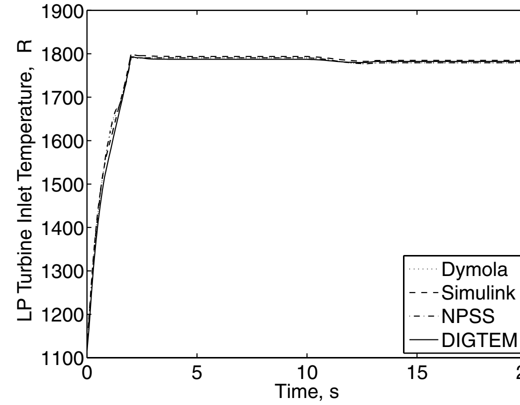

Fig. 15 Variation of low-pressure turbine inlet temperature during the transient.

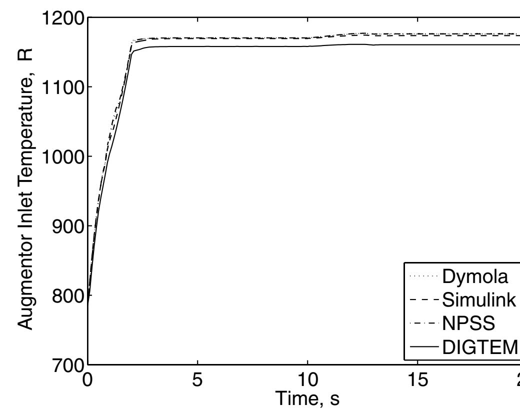

Fig. 16 Variation of augmentor inlet temperature during the transient.

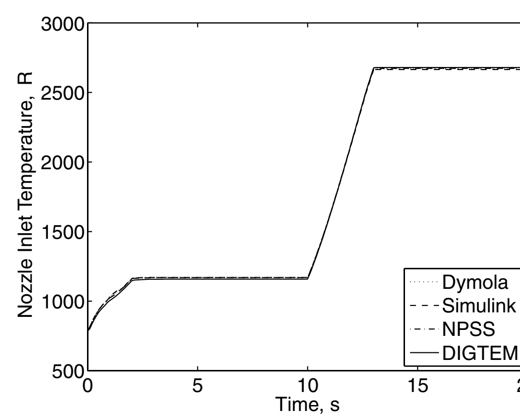

Fig. 17 Variation of nozzle inlet temperature during the transient.

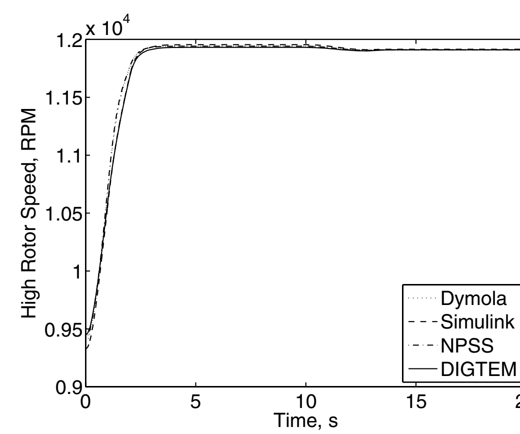

Fig. 18 Variation of high-pressure spool speed during the transient.

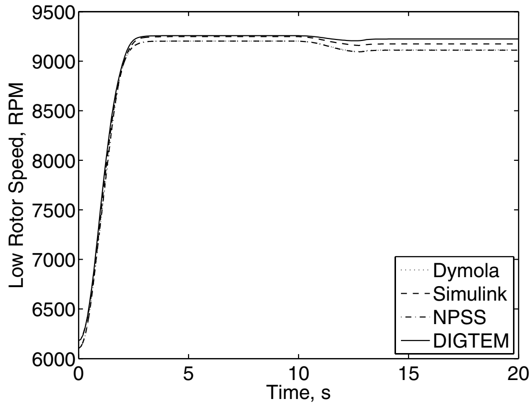

Fig. 19 Variation of low-pressure spool speed during the transient.

used in the three modern tools. Dymola uses the Differential Algebraic System Solver (DASSL) by default, Simulink uses the Euler solver, and NPSS uses the Gear first order implicit solver. From the plots, one can see that the results from the three modern tools are fairly close to those from DIGTEM. A few of the plots show differences such as the low-pressure spool rotational speed (Fig. 19) and the augmentor inlet pressure and temperature (Figs. 10 and 16). However, these differences are fairly small and range from about 1–5%.

For performance comparison purposes, the 20-s transient was repeated using identical solver parameters. Once again, a fixed time step of 0.0005 s was used. The Euler solver was chosen because it was the only one common to all three programs. Simulink and NPSS were run on a 3.8-GHz Pentium 4 PC with 1024 MB RAM while Dymola was run on a 2.4-GHz Pentium 4 PC with 512 MB RAM. Both machines were running the latest updated version of the Windows XP Professional operating system. Table 2 shows the time in seconds taken by each program to simulate the 20-s transient using the Euler solver and the default solver of each individual program.

A time step sensitivity study was also performed using the Euler solver. In the cases of Dymola and Simulink, it was found that the largest time step that could be used was 0.00058 s, whereas with NPSS, it was 0.002 s. Values larger than these resulted in the inputs to the performance maps going out of range and subsequent convergence problems. The time steps used and the simulation run times are summarized in Table 3 and depicted graphically in Fig. 5. In all cases, the differences observed in the results of the transient simulation were smaller than 0.01.

IV. Conclusions

From the results, one can see that Dymola, Simulink, and NPSS produce nearly identical results. One can also take advantage of the object-oriented nature of these tools when writing code and hence they are well suited for use in the simulation of propulsion systems. Among the three programs, Dymola was found to be the fastest and the easiest to use. One of the advantages of Dymola is that the equations being solved do not need rearranging and they do not have to be entered strictly in the order in which they are to be solved. Another time-saving advantage is that plots of every possible variable used in the simulation as a function of time can be viewed as soon as the simulation is completed, eliminating the need for postprocessing software. A disadvantage of Dymola, which uses the Modelica programming language, is a lack of extensive documentation.

On the other hand, Simulink, aside from being extremely well documented by Mathworks, is also the subject of numerous books containing plenty of examples that highlight the program's features. Similar to Dymola, Simulink also includes a postprocessor that enables the viewing of results during and after the simulation. Simulink also permits the embedding of programs and functions written using MATLAB.

NPSS was found to be the slowest in simulating the model in this paper. Despite being more flexible than Dymola and Simulink in selecting the time step, NPSS still took longer to simulate the model. NPSS is available in a Graphical User Interface (GUI) version and a command line version. An attempt to use the GUI version of the program was rendered useless due to the CPU usage spiking to 100% when performing even mundane tasks such as opening a file. It was also felt that the inclusion of postprocessing functions would be beneficial and the documentation could use some improvement.

Acknowledgment

Kirk J. Gomes thanks John Reed for his help and patience during this project.

References

[1] Daniele, C. J., Krosel, S. M., Szuch, J. R., and Westerkamp, E. J., "Digital Computer Program for Generating Dynamic Turbofan Engine Models (DIGTEM)," NASA TM-83446, 1983.

[2] Fritzson, P., *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*, IEEE Press, Piscataway, NJ, 2004, Chap. 2.

[3] NPSS, Numerical Propulsion System Simulation, Software Package, Ver. 1.6.4, NASA John H. Glenn Research Center, Cleveland, OH, 2006.

[4] Szuch, J. R., Krosel, S. M., and Bruton, W. M., "Automated Procedure for Developing Hybrid Computer Simulations of Turbofan Engines. Part 1—General Description," NASA TP-1851, 1982.

[5] Keenan, J. H., and Kaye, J., *Gas Tables*, Wiley, New York, 1948.

[6] Shapiro, A. H., *The Dynamics and Thermodynamics of Compressible Fluid Flow*, Vol. 1, Ronald Press Co., New York, 1953.