JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 25, No. 3, May–June 2002

Rocket Nozzle Flow Control Using a Reduced-Order Computational Fluid Dynamics Model

David J. Lucia* and Meir Pachter†
Air Force Institute of Technology, Wright–Patterson Air Force Base, Ohio 45433
and
Philip S. Beran‡
U.S. Air Force Research Laboratory, Wright–Patterson Air Force Base, Ohio 45433

The use of proper orthogonal decomposition-based reduced-order computational fluid dynamics models for multidisciplinary optimization are investigated. An open-loop optimal control problem is solved concerning the regulation of a rocket engine thrust profile. The control synthesis uses a numerical flowfield solver as the plant model. Controllers are synthesized using both the full-order nonlinear fluid dynamics, which serves as a truth model, and a reduced-order model of the flowfield generated via proper orthogonal decomposition. The latter has been successfully applied to reduce the order of both steady and unsteady subsonic flowfield solutions obtained via computational fluid dynamics. Specifically, the application of reduced-order flowfield solvers to a rocket's thrust controller design is explored. A quasi-one-dimensional supersonic convergent-divergent nozzle with varying backpressure is used as a model problem. Treating the nozzle flow as one-dimensional, the reduced-order model is used to synthesize a nonlinear controller that controls thrust by varying nozzle throat area along an ascent trajectory. The performance of the reduced-order controller is quantified.

Nomenclature

A	=	Jacobian of flux terms E in Euler equations
$A(x)$	=	nozzle cross-sectional area, m^2
E	=	vector of X-axis fluxes
E_T	=	total energy, J
$\boldsymbol{G}(t)$	=	system dynamic response function
H	=	numerical flux function from Roe's scheme
$\dot{m}$	=	mass flow rate, kg/s
N, M	=	dimensions of reduced-order mapping matrix
nx	=	total number of grid points in the nozzle
P	=	pressure, Pa
R	=	transformation matrix
R_g	=	ideal gas constant, J/kg · K
$R()$	=	right-hand side or nonlinear portion of Euler equations
S	=	matrix of flowfield data, or snapshots
T	=	temperature, K
$T(t)$	=	thrust, N
t	=	time, s
U	=	vector of conserved flow variables
u	=	fluid velocity at a single spatial location in the flow, m/s
$\boldsymbol{u}(t)$	=	vector of control deflections
V	=	matrix of singular values of S
$\boldsymbol{w}$	=	represents any single fluid variable
$\boldsymbol{X}$	=	vector of X-axis positions x in nozzle, m
x	=	spatial coordinate along axis of nozzle, m
$\boldsymbol{x}(t)$	=	spatial state vector of fluid variables
Z	=	vector of forcing terms for quasi-one-dimensional nozzle Euler equations
γ	=	ratio of specific heats
Λ	=	matrix of eigenvalues
λ	=	ratio of time and spatial step size
ρ	=	density, kg/m^3
Ψ	=	reduced-order mapping matrix

Subscripts

amb	=	ambient
c	=	reference signal for control
i	=	index to increment discrete time steps for controller time scale
j	=	index to increment discrete spatial steps
opt	=	optimal
stag	=	stagnation
t	=	time derivative
x	=	spatial derivative
0	=	initial value

Superscripts

n	=	index to increment discrete time steps for fluid timescale
$*$	=	value at nozzle throat

Introduction

FLUID dynamics are governed by a set of partial differential equations (PDEs) known as the Navier–Stokes (NS) equations.[1] Because current control synthesis methods are almost exclusively tailored for ordinary differential equations (ODEs) in time as plant models, fluids/controls research currently being reported in the literature approximates the NS equations with ODEs. These ODEs are carefully designed to capture the critical dynamics in the flowfield, yet facilitate a tractable control synthesis via conventional methods.[2,3] Although this approach has yielded results for specific flows,[2,3] few control applications have been demonstrated.

Fluid dynamics modeling is problematic because no general solution to the NS equations is known to date. Although there are closed-form solutions to a handful of flowfields, most of these solutions require assumptions that are not valid for problems of interest. In the past, aerodynamicists used these closed-form solutions to educate themselves on the behavior of the NS equations, whereas actual aerodynamic design relied on wind-tunnel testing. More recently, the wind tunnel has been augmented by the use of computational fluid dynamics (CFD). CFD creates a computational wind tunnel

Received 22 February 2001; revision received 25 October 2001; accepted for publication 30 December 2001. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/02 $10.00 in correspondence with the CCC.

*Doctoral Candidate, Department of Aeronautics and Astronautics. Member AIAA.

†Professor of Electrical Engineering, Department of Electrical Engineering. Associate Fellow AIAA.

‡Senior Research Engineer, Air Vehicles Directorate. Senior Member AIAA.

where the complete NS equations are solved via computer-based numerical approximation. This enables the prediction of stability and control derivatives.

This paper pursues the synthesis of controllers for high-speed, compressible fluid flow problems using online CFD-based solvers to model the fluid dynamics and online calculations/optimization algorithms for online control design. Most flow problems require very large (high-order) CFD models to capture the dynamics of the flow. These CFD models are so large that design optimization via iterative search algorithms is infeasible in real time. The use of proper orthogonal decomposition (POD) for order reduction of CFD flowfield solutions was introduced in the mid 1990s. This method has been successfully applied to low-speed, nonlinear aeroelastic problems.[4] POD is a promising tool for use in model-based control, and a few low-speed flow control applications with POD have recently been reported in the literature.[5–8]

This paper develops the controller synthesis strategy for a one-dimensional model problem involving high-speed compressible fluid flow. A CFD code is used as the system dynamics model. The dynamics model is integrated into an optimization algorithm, and an open-loop optimal controller is synthesized for the full-order system. Next, the reduced-order fluid model is developed using POD, and a suboptimal, model-based, open-loop controller is constructed via the POD reduced-order model (ROM) of the plant. The performance of the full-order and POD ROM-based controllers is compared.

Problem Statement

In this paper, a thrust controller for a liquid fueled rocket engine with a variable throat area and adjustable propellant flow rate is designed. The rocket engine dynamics are modeled with a quasi-one-dimensional nozzle flow, which is a compressible, one-dimensional, inviscid flowfield. The one-dimensional nozzle assumption provides accuracy within about 6% of a higher-order analysis.[9] Flow control is accomplished by changing the nozzle geometry, which entails changing the nozzle throat area and thereby the fuel mass flow rate. This novel control concept has been tested on experimental rockets.[9]

This problem is treated with a numerical method applicable to complex problems with no known analytical solution, even though the analytical solution for quasi-one-dimensional nozzle flows is well known and provided in gasdynamics textbooks.[10] This general numerical method is intentionally demonstrated on a problem with a known solution to allow for verification of the solution.

The relevant parameters for the chemical rocket engine were taken from the open literature.[9] The rocket engine was designed for an optimal thrust of 5000 N at 25,000 m of altitude. This yielded the following specifications: $A^* = 13.87$ cm^2, $A_{\text{exit}} = 612.5$ cm^2, $\dot{m} = 1.919$ kg/s, $u_{\text{exit}} = 2605$ m/s.

The chamber conditions are as follows: $P_{\text{stag}} = 2.068$ MPa, $T_{\text{stag}} = 2800$ K, $\gamma = 1.3$, $R_g = 355.4$ J/kg · K. The rocket engine operates on a simulated ascent trajectory between 10,000 and 25,000 m of altitude. The simplified trajectory entails a linear decrease in ambient pressure over a 120-s flight time, as in Eq. (1):

$$P_{\text{amb}} = \text{altitude} \cdot \left\{ \frac{(P_{25,000} - P_{10,000})}{(25{,}0000 - 10{,}000)} \right\} \tag{1}$$

The rocket's thrust is given by Eq. (2) (Ref. 11),

$$\text{thrust} = \dot{m} u_{\text{exit}} + (P_{\text{exit}} - P_{\text{amb}}) A_{\text{exit}} \tag{2}$$

Without any control input, the nominal thrust along this ascent trajectory is shown in Fig. 1.

A schematic of the rocket nozzle is shown in Fig. 2. The control only affected the nozzle geometry near the throat (shaded region in Fig. 2), whereas most of the nozzle geometry (especially the nozzle exit area) remained fixed. No shock was allowed to stand in the nozzle, resulting in isentropic flow throughout the nozzle. Hence, the isentropic flow relations yield the analytical solution for this flowfield.[10] The flow in the nozzle was constant in the uncontrolled rocket, and the increase in thrust throughout the trajectory (Fig. 1) was due only to the drop in ambient pressure relative to the fixed exit exhaust pressure. Certain launch vehicles need to restrict this increase in thrust with altitude. Such would be the case when launching a gravity-sensitive payload into orbit. For example, the space shuttle uses fuel mixture ratio control in the combustion chamber of the main engine to restrict launch dynamics to less than 3 g (Ref. 11). For this model problem, an arbitrary thrust profile was selected to be tracked by the controller. The thrust level directly determines the rockets acceleration. Digital control was envisioned and the discrete time variable is t_i, $i = 1, 2, \ldots$. The control problem is described as follows.

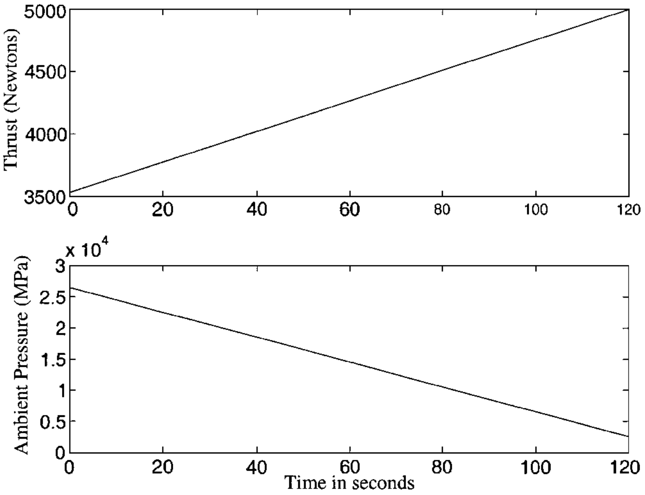

Fig. 1 Thrust from unmodified rocket.

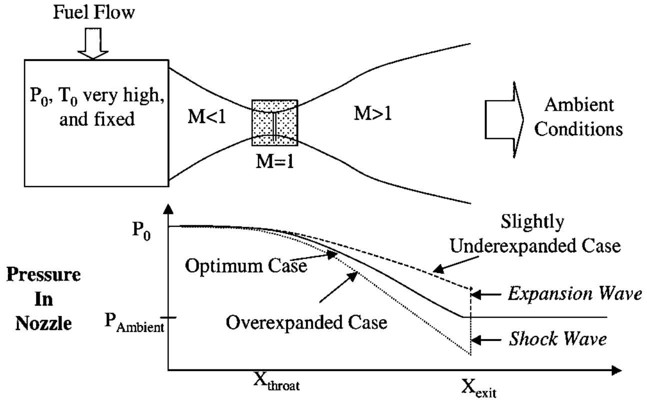

Fig. 2 Rocket engine schematic.

The reference command signal is T_c: thrust desired $= T_c(t_i)$. The control variable is A_c^*: nozzle throat area $= A_c^*(t_i)$. The nozzle throat area and mass flow rate are related by an equality constraint that enforces the choked flow condition at the nozzle throat. This relationship is given by the one-dimensional quasi-stationary isentropic flow relation.[12] Therefore, nozzle throat area is the only control variable:

$$\dot{m}_c(t_i) = A_c^*(t_i)\left\{\left(P_0 \big/ \sqrt{RT_0}\right)\sqrt{\gamma}\,[2/(\gamma+1)]^{(\gamma+1)/2(\gamma-1)}\right\} \tag{3}$$

The states $\boldsymbol{x}(t_i)$ at time t_i are a function of the x-axis position in the nozzle $\boldsymbol{X}$ at nx discrete locations,

$$\boldsymbol{x}(t_i) = \begin{bmatrix} \rho(\boldsymbol{X}, t_i) \\ \rho(\boldsymbol{X}, t_i) \cdot u(\boldsymbol{X}, t_i) \\ E_T(\boldsymbol{X}, t_i) \end{bmatrix}, \qquad \boldsymbol{X} = \{x_1, x_2, \ldots, x_{nx}\}$$

Here, ρ is density, ρu is x-direction momentum, P is pressure, and E_T is total energy per unit mass. The output variable of interest (assuming no shock wave exists in the nozzle) is thrust $= T(t_i)$, and the disturbance signal is ambient pressure $= P_{\text{amb}}(t_i)$. For a prespecified nozzle geometry, the system dynamics are given by

$$\boldsymbol{x}(t_i + \Delta t) = f\{\boldsymbol{x}(t_i);\, A_c^*(t_i), \dot{m}(t_i)\}, \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{4}$$

The state transition function f is obtained from the inviscid flow equations (also known as the Euler equations). Each time step requires a function call to a CFD algorithm to propagate the state

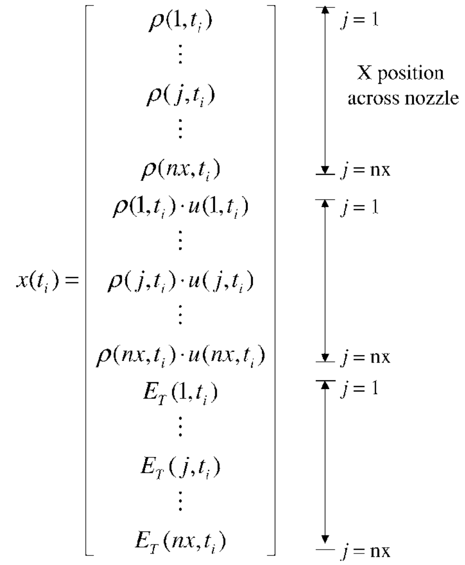

Fig. 3 State vector of flow variables.

from one time to the next. The states for this system are arranged into a single column vector as shown in Fig. 3. Notice that the disturbance did not affect the system dynamics because the flow at the nozzle exit was always supersonic. Changes in ambient pressure outside the nozzle cannot propagate upstream in a supersonic flow. The disturbance affects the overall thrust accordingly. Thus, the output variable is

$$T(t_i + \Delta t) = \dot{m} u_{\mathrm{exit}}[\boldsymbol{x}(t_i + \Delta t)] + [P_{\mathrm{exit}}(\boldsymbol{x}(t_i + \Delta t)) - P_{\mathrm{amb}}(t_i + \Delta t)]A_{\mathrm{exit}} \tag{5}$$

where u_{exit} and P_{exit} are functions of the state vector $\boldsymbol{x}$.

The CFD fluid model used an explicit Roe scheme solver. The CFD flow model spatially discretized the one-dimensional nozzle into a grid along its length. For nx grid points, there were nx values of each flow variable computed in the model. The number of grid points was determined by the accuracy and stability of the CFD algorithm. For this problem, the scheme yielded a very nice solution for the convergence divergent nozzle problem with 250 grid points. Therefore, the state vector for the full-order system had dimension 750.

Controller Synthesis

Perfect a priori knowledge of the disturbance, for example, the ambient pressure profile, is assumed along the trajectory of the rocket, and an open-loop optimal controller is pursued for this problem. The open-loop control system is shown in Fig. 4. Equation (6) shows the open-loop controller signal $\boldsymbol{u}(t_i)$, which is a function of the commanded thrust, the ambient pressure, and time [shown as $\boldsymbol{G}(t_i)$]:

$$\boldsymbol{u}(t_i) = \begin{bmatrix} A_c^*(t_i) \\ \dot{m}(t_i) \end{bmatrix}, \qquad \boldsymbol{G}(t_i) = g\{t_i, T_c(t_i), P_{\mathrm{amb}}(t_i)\} \tag{6}$$

Finally, the flowfield prediction and the myopic control evaluation can be separated in time. The considerable timescale separation between the flowfield and actuator response times mean that the actuator dynamics will not affect the stability of the solution, and so they are left out for simplicity. The flowfield reaches steady state between each time step for the control signal calculation/optimization. Strictly speaking, an otherwise steady flowfield (time-independent flowfield) is disturbed by a step change in geometry (A^* and the portion of the nozzle very near the throat) at each discrete value of t_i in the open-loop simulation. The frequency content of the command signal $T_c(t_i)$ and the disturbance signal $P_{\mathrm{amb}}(t_i)$ is low compared to the speed at which disturbances propagate through the flowfield, l/c, where l is the length of the nozzle and c is the reference speed of sound. The CFD code is used to perform integration in time and a new stationary (steady-state) one-dimensional flow field is quickly established. This new flowfield is used to predict the thrust produced with the new A^* geometry. The polarity is obvious: An increase in A^* produces an increase in thrust, whereas a decrease in A^* produces a decrease in thrust. This physical insight offers an easy interpolation process for matching the predicted (calculated) thrust to the commanded thrust T_c. Nonlinear equation (5) is solved when $\dot{m}(A_c^*)$ is found such that T matches T_c. This procedure encapsulates the myopic optimization algorithm used to synthesize the optimal controller. The algorithm is outlined in Fig. 5.

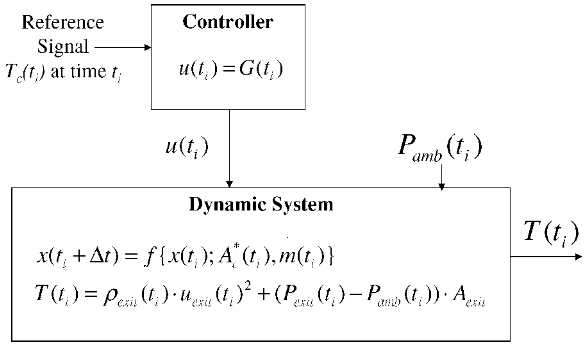
Reference Signal
$T_c(t_i)$ at time t_i
Controller
$u(t_i) = G(t_i)$
$u(t_i)$
$P_{amb}(t_i)$
Dynamic System
$x(t_i + \Delta t) = f\{x(t_i); A_c^*(t_i), m(t_i)\}$
$T(t_i) = \rho_{exit}(t_i)\cdot u_{exit}(t_i)^2 + (P_{exit}(t_i) - P_{amb}(t_i))\cdot A_{exit}$
$T(t_i)$

Fig. 4 Open-loop control diagram.

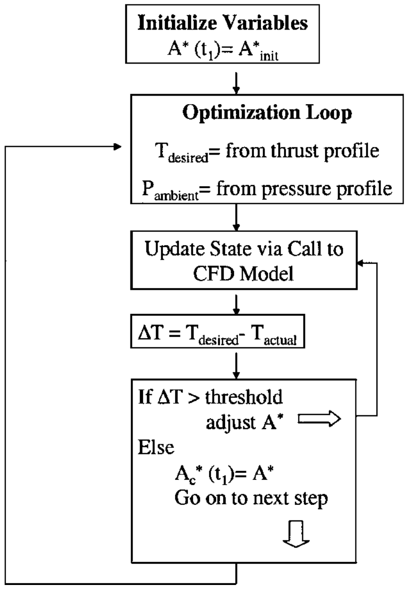

Fig. 5 Solution process.

Fluid Model

The Euler equations are used to approximate inviscid flowfields. For unsteady one-dimensional flow in a duct of variable (but known) cross-sectional area, the Euler equations (in conservation form) reduce to the equation set given here[13]:

$$U_t + E_x = Z, \qquad U = \begin{bmatrix} \rho A \\ \rho u A \\ \rho E_T A \end{bmatrix} \tag{7}$$

$$E = \begin{bmatrix} \rho u A \\ (\rho u^2 + P)A \\ (\rho E_T + P)uA \end{bmatrix} \tag{8}$$

$$Z = \begin{bmatrix} 0 \\ -P\dfrac{\mathrm{d}A}{\mathrm{d}x} \\ 0 \end{bmatrix} \tag{9}$$

The nozzle geometry for the unmodified part of the nozzle was selected to be a quadratic area profile[14] where A^*_{design} was the throat

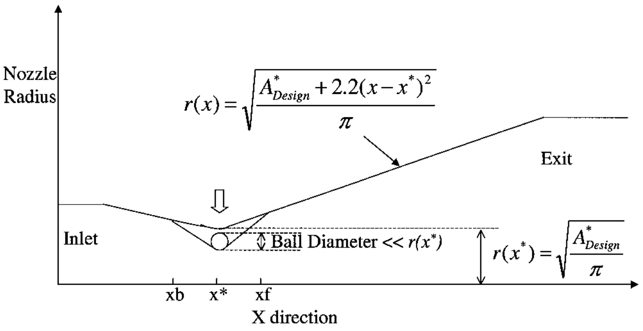
Nozzle Radius
$r(x) = \sqrt{\dfrac{A^*_{Design} + 2.2(x - x^*)^2}{\pi}}$
Exit
Inlet
Ball Diameter << r(x*)
$r(x^*) = \sqrt{\dfrac{A^*_{Design}}{\pi}}$
xb
x*
xf
X direction

Fig. 6 Variable throat nozzle geometry.

area for the original nozzle design (13.87 cm^2), and x^* was the fixed x location of the minimum nozzle area, (5.5456 cm):

$$A(x) = A^*_{\text{design}} + 2.2(x - x^*)^2$$

The quadratic area profile was modified to include a variable throat area as shown in Fig. 6. This was accomplished by inserting a circular ball of radius r, centered on the fixed x^*, into the flowfield. The ball was lowered into the flowfield so that the lower most edge of the ball results in a nozzle throat radius whose area ($A = \pi R^2$) was the throat area commanded by the controller A^*_c. A region near the nozzle throat was created (denoted by xb and xf) where the original nozzle configuration was linked to the ball via a line segment that connects the nozzle wall to the nearest tangent point on the ball. This ensured a smooth surface in the vicinity of the nozzle throat. The nozzle entrance and exit were also extended with a constant area section to ensure that no large flow gradients were passed through the boundary. This aided in convergence from poor initial conditions.

Notice that Eq. (9) requires spatial differentiation of the nozzle area, but the nozzle area function lacks smoothness at xb, xf and at the transition to the inlet and exit constant area sections. This is accommodated by numerically approximating the spatial derivative of the cross-sectional area via a second-order central difference. This smoothed the area function and provided good results.

Roe's scheme was used to solve numerically this system of PDEs. Roe's scheme for the Euler equations is a finite difference scheme that approximates the flux terms at each discrete grid point with an averaging technique using information at the surrounding points in the flow. See Tannehill et al.[1] for a detailed formulation of the Roe's scheme. This finite difference scheme was used to solve explicitly the entire flowfield by marching a known initial condition in time via small time steps. A steady-state flow solution was achieved when the maximum norm of the difference between flow solutions at adjacent time steps was less than a small error threshold (pointwise convergence).

For the quasi-one-dimensional nozzle problem, the boundary conditions had to be set at the nozzle's inlet and exit. When the characteristics of the governing flow equations are analyzed, the appropriate number of terms to specify were determined. The inlet represents propellant flow out of the combustor and was subsonic. It was appropriate to specify two of the three solution variables for this part of the flowfield, which consisted of subsonic flow. Momentum and total energy were specified, letting the density term at the inlet be adjusted by the flow solution. Because a shock was not allowed to form in the nozzle, the exit flow was always supersonic, and no information from the ambient environment affected the flowfield. The exit condition was simply an outflow of the last grid point's flow solution. It was convenient to extrapolate extra grid points beyond the nozzle inlet and exit. The flow values at the extrapolated inlet and exit grid points were computed using a one-sided second derivative approximation for the velocity variable that was set equal to zero to enforce smoothness in the extrapolated boundary points.

Finally, Roe's scheme loses its dissipation when the eigenvalues of the Roe averaged matrix go through zero. As a result, a nonphysical expansion shock can result in the solution, which is indeed the case for the convergent-divergent nozzle solved via Roe's scheme. An expansion shock will form at the nozzle throat unless an entropy fix is introduced. To accomplish this, the eigenvalues in $|\hat{\Lambda}|$ were checked and replaced with a finite value ϵ when they were smaller than a certain threshold.[1] The value of this threshold was sensitive to grid size.

Reduced-Ordered Modeling

Reduced-order modeling of the flowfield was accomplished via POD. A non-Galerkin approach was used.[4] The development of POD ROM for inviscid flow through a quasi-one-dimensional nozzle, along with a performance analysis has been recently completed.[15] Consider the column vector of flow variables shown in Fig. 3. This vector represents the state variable in time and was governed by Eq. (4). For order reduction of the flowfield, the state vector was decomposed into the conserved flow variables, which yielded three vectors (one for density, one for momentum, and one for total energy). Let the vector $\boldsymbol{w}(t)$ represent any one of these three flow variable vectors. Then the Roe's scheme flow solver can be considered as a nonlinear state transition function R acting on $\boldsymbol{w}(t)$. Notice that for $\boldsymbol{w}(t)$, t now represents time at the fluid timescale:

$$\frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}t} = R(\boldsymbol{w}) \tag{10}$$

A linear transformation was sought between the full-order state and a reduced-order state $\hat{\boldsymbol{w}}$ as follows:

$$\boldsymbol{w}(t) = \Psi \cdot \hat{\boldsymbol{w}}(t)$$

Ψ was constructed by collecting observations of the full-order state vector at different time intervals throughout the integration of the full-order Roe scheme solution. These observations, or snapshots, were collected before reaching steady state, because linearly independent snapshot vectors were required. For simplicity, a specific throat area configuration was selected. With this configuration fixed, snapshots were generated that were used to model every flowfield of interest to the optimization algorithm (including those generated via different throat area geometries). Collecting snapshots from a variety of throat area geometries proved to be unnecessary for this model problem. The parameter space of interest was the small throat area changes required to affect control for this problem. The variations in the nozzle flowfield across this parameter space were small enough to be reasonably represented by the flow structures from a single set of snapshots.

M total snapshots [usually $\mathcal{O}(10)$ or less] of the full-order state vector length N were collected. These snapshots were compiled into a $N \times M$ matrix S, known as the snapshot matrix. POD guaranteed that each of the three reduced-order flowfield variables would yield an optimally convergent representation of the full-order variable if the mapping function Ψ was developed as follows:

$$S^T S \cdot V = V \cdot \Lambda, \qquad \Psi = S \cdot V$$

Here, V is the matrix of eigenvectors of $S^T S$ and Λ is the corresponding diagonal matrix of eigenvalues that was used to yield the following reduced-order mapping:

$$\boldsymbol{w}(t) = S \cdot V \cdot \hat{\boldsymbol{w}}(t), \qquad S^T \cdot \boldsymbol{w}(t) = S^T \cdot S \cdot V \cdot \hat{\boldsymbol{w}}(t)$$

$$S^T \cdot \boldsymbol{w}(t) = V \cdot \Lambda \cdot \hat{\boldsymbol{w}}(t), \qquad \hat{\boldsymbol{w}}(t) = \Lambda^{-1} \cdot V^{-1} \cdot S^T \cdot \boldsymbol{w}(t)$$

Thus, when this relationship was inserted into Eq. (10) and a forward difference approximation was applied, the flow model produced the following reduced-order flow solver:

$$\frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}t} = R(\boldsymbol{w}) \Rightarrow \boldsymbol{w}^{n+1} = \boldsymbol{w}^n + \Delta t \cdot R(\boldsymbol{w})$$

which becomes

$$\hat{\boldsymbol{w}}^{n+1} = \hat{\boldsymbol{w}}^n + \Delta t \cdot \Lambda^{-1} \cdot V^{-1} \cdot S^T \cdot R(S \cdot V \cdot \hat{\boldsymbol{w}}^n) \tag{11}$$

Thus, the POD ROM reduced each flow variable from N to M, where M was the number of snapshots. Each reduced-order variable represented a solution mode, whose contribution to the full-order solution was of order with the magnitude of the corresponding eigenvalue. No modal truncation was employed in this implementation. The method of order reduction in Eq. (11) relied on the full-order function evaluation at each flowfield integration step. As such, the order of each integration step was not actually reduced. However, this reduction technique can greatly increase the time-step size allowed for stability. Therefore, the total number of time steps required for the explicit time accurate solver to reach steady state was significantly reduced.

For implicit schemes, where time-step size does not affect stability, the governing equations must be projected into the reduced-order space (for instance, via a Galerkin projection) to realize a computational benefit from POD ROM. For a complete derivation and discussion of POD as it pertains to fluids, see the text by Holmes et al.[16]

Results

For the model problem, a controller was designed to maintain a constant 5000-N thrust as the rocket flies through the simplified trajectory shown in Fig. 1. The unmodified rocket was designed for optimal performance of 5000-N at 25,000 m. The controller was designed to maintain 5000-N thrust throughout the entire trajectory. (The target thrust was actually 5047 N.) The nozzle required a larger mass flow rate and nozzle throat area at the lower altitudes to track the thrust profile. When the optimization algorithm was exercised, errors were allowed to go uncorrected within ±5 N, which was the convergence tolerance.

First consider the controller synthesized from the full-order CFD plant model: the controller synthesized using the full-order plant produced the results shown in Fig. 7. Clearly, the optimal controller had no problem maintaining thrust within the 5 N threshold. As the rocket ascended and the pressure dropped, the tendency was for the thrust to increase. Once the increase broke the 5-N threshold, the controller commanded some increment in throat area that brought the thrust down to well within limits. This process of a naturally creeping increase in thrust, with a periodic controller initiated decrease in thrust, produced the variations shown in Fig. 7. These dynamics resulted in a slightly biased thrust error, which is shown in Fig. 8. The bias and standard deviation were small (less than 1%) and deemed insignificant for the model problem. Of course, the solution algorithm could be tuned to provide smoother results, or could use a smaller threshold to reduce the bias. The controller actuator commands are shown in Fig. 9. These show the commanded throat area A_c^*, the resulting nozzle throat radius, and the commanded propellant flow rate. The low bandwidth of the control signal is apparent. Thus, spill over is avoided.

The controller started with a larger throat area and mass flow rate, and, as the flight progresses to higher altitudes, these were reduced via actuator commands. At the terminal point of the trajectory, the controller commanded the throat area and flow rate for the optimum 5000-N nozzle at 25,000-m altitude (with a fixed divergent section and exit area). Because the disturbance and desired thrust were linear functions in time, the resulting controller commands were also linear. Note that the commanded actuator dynamics only required about a 7% change in throat geometry and a 14% change in propellant flow rate to achieve the desired performance.

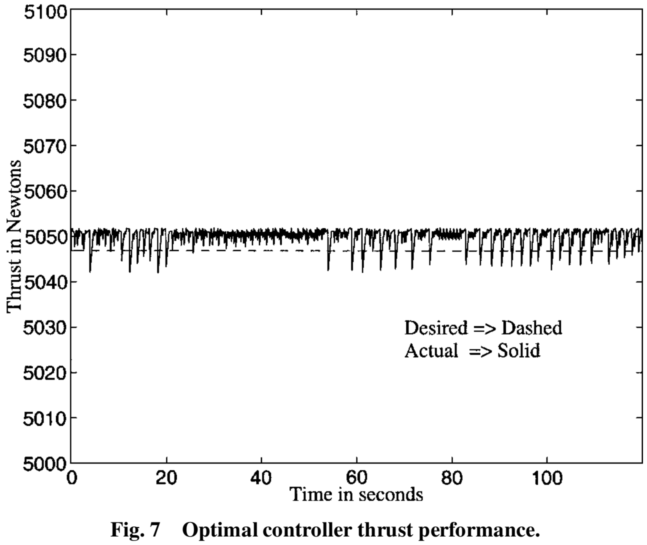

Fig. 7 Optimal controller thrust performance.

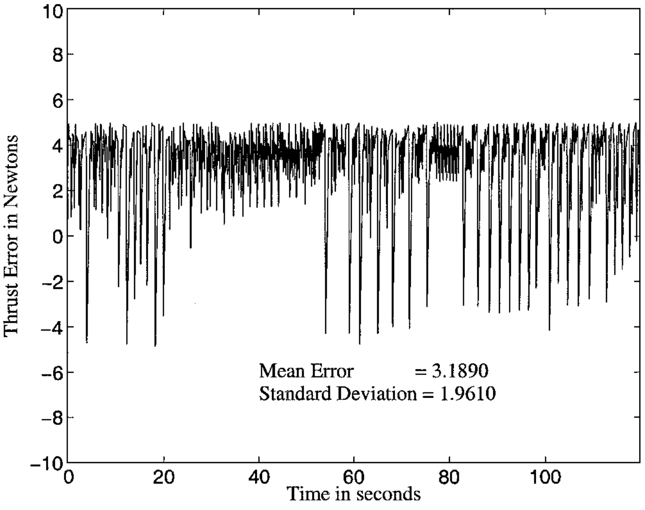

Fig. 8 Optimal controller thrust error.

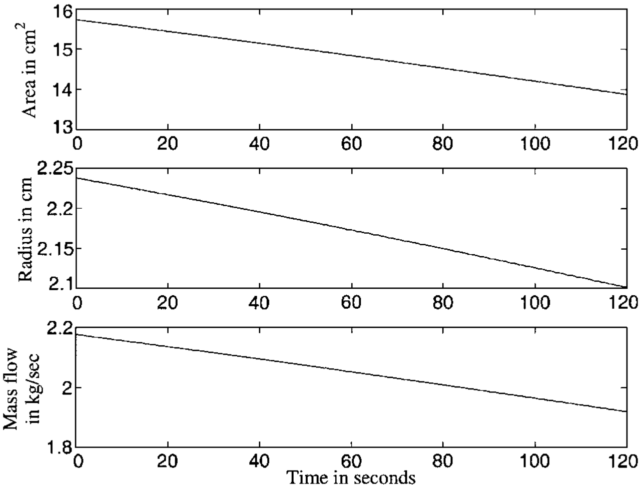

Fig. 9 Optimal controller actuator commands.

For problems where the full-order model is too large to iterate through thousands of function calls in real time, the performance of a controller synthesized from a reduced-order model must be considered. A reduced-order model was generated from four snapshots of the full-order system state vector, taken at even intervals as the flow solver progressed from initial condition to steady state. The snapshots were obtained from the full-order flow solver explicit time integration with the nozzle geometry required at 20 s into the flight trajectory (approximate). These four snapshots produced an ROM with four modes for each of the three conserved flow variables. This resulted in a total state dimension of 12. The 12 modes were applied to the unsteady equations as described earlier.

The reduced-order plant was inserted into the optimization algorithm to replace the full-order function call, and the algorithm was rerun to generate a suboptimal controller. The suboptimal controller thrust performance, based on the full-order plant, is shown in Fig. 10. The optimization algorithm, based on the reduced-order plant, introduced a slowly growing error in thrust. The error is shown in Fig. 11. The mean error for the suboptimal controller was 21.8 N, which is a 0.43% error.

Reduced-order model accuracy can be adjusted to achieve desired performance goals by varying the number of modes and dispersion

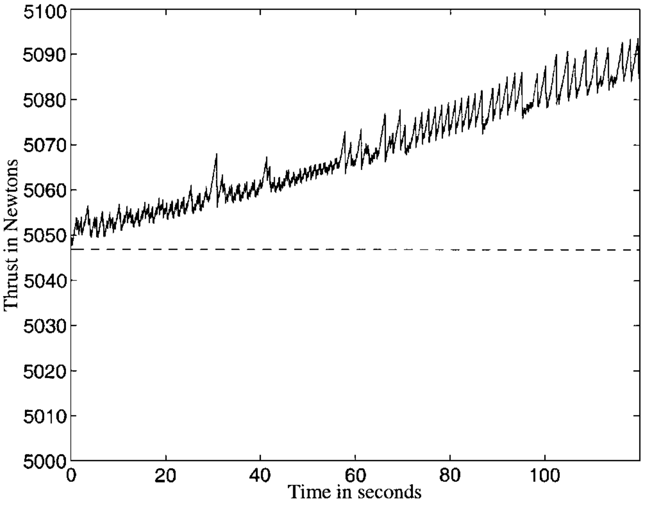

Fig. 10 Suboptimal thrust performance.

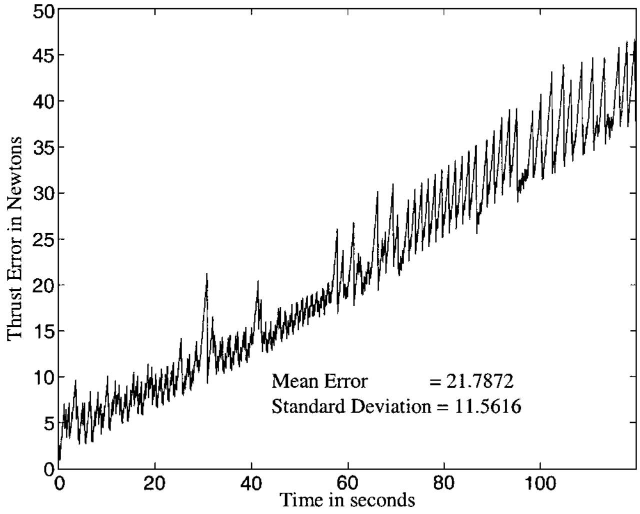

Fig. 11 Suboptimal thrust error.

of snapshots.[4] These improvements were not pursued because the performance shown in Fig. 11 was adequate to demonstrate the controller strategy. For future implementations involving quasi-steady flow dynamics, a variety of steady-state analysis approaches could be used instead of time-accurate integration. It has been shown that steady analysis using POD ROM is both efficient and accurate.[17] For a steady analysis, steady-state flow solutions across the parameter space could be blended[18] into a larger set of snapshots. This approach to data collection would cover a wider parameter space than simply using time integration at one parameter value. In such an implementation, lower energy modes should be truncated to keep the dimensionality small.

Conclusions

A suboptimal controller for a high-speed, compressible fluid control problem, albeit without shocks, has been demonstrated using proper orthogonal decomposition based reduced-order CFD models. Both an optimal and suboptimal controller were generated for a quasi-one-dimensional supersonic convergent-divergent nozzle with varying backpressure. The optimal controller, generated from a full-order plant, tracked the desired thrust profile within an arbitrarily small threshold. The ROM yielded a suboptimal controller that replicated the optimal controller with 0.43% mean error. The reduced-order plant had a state dimensionality of 12, vs the full-order state dimensionality of 750, which is about a 62–1 order reduction. Proper orthogonal decomposition based reduced-order models could realize much greater order reductions for problems with full-order models requiring higher dimensionality. For example, order reductions for two-dimensional inviscid problems are typically 1000–1, and order reductions would be much greater still in three-dimensions.

References

[1] Tannehill, J. C., Anderson, D. A., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, Hemisphere, Washington, DC, 1997, Chap. 6, pp. 388–395.

[2] Joshi, S., Speyer, J., and Kim, J., "Finite Dimensional Optimal Control of Poiseuille Flow," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 2, 1999, pp. 340–348.

[3] Cortelezzi, L., Leonard, A., and Doyle, J., "An Example of Active Circulation Control of the Unsteady Separated Flow Past a Semi-Infinite Plate," *Journal of Fluid Mechanics*, Vol. 260, 1994, pp. 127–154.

[4] Beran, P. S., Huttsell, L. J., Buxton, B. J., Noll, C., and Osswald, G., "Computational Aeroelastic Techniques for Viscous Flow," CEAS/AIAA/ICASE/NASA Langley International Forum on Aeroelasticity and Structural Dynamics, June 1999.

[5] Coller, B. D., Holmes, P., and Lumley, J. L., "Control of Bursting in Boundary Layer Models," *Applied Mechanics Review*, Vol. 6, No. 2, 1994, pp. S139–S143.

[6] Ito, K., and Ravindran, S. S., "A Reduced-Order Method for Simulation and Control of Fluid Flows," *Journal of Computational Physics*, Vol. 143, No. 2, 1998, pp. 403–425.

[7] Delville, J., Cordier, L., and Bennet, J., "Large-Scale-Structure Identification and Control in Turbulent Shear Flows," *Flow Control: Fundamentals and Practices*, Springer, Berlin, 1998, pp. 199–273.

[8] Prabhu, R. D., Collis, S. S., and Chang, Y., "The Influence of Control on Proper Orthogonal Decomposition of Wall-Bounded Turbulent Flows," *Physics of Fluids*, Vol. 13, No. 2, 2001, pp. 520–537.

[9] Sutton, G., *Rocket Propulsion Elements*, Wiley, New York, 1992, Chap. 3, pp. 56–82.

[10] Anderson, J., *Fundamentals of Aerodynamics*, 2nd ed., McGraw–Hill, New York, 1991, pp. 402, 403.

[11] Huzel, D., and Huang, D., *Modern Engineering for Design of Liquid-PropellantRocket Engines*, AIAA, Washington, DC, 1992, Chap. 7, pp. 219–276.

[12] Hill, P., and Peterson, C., *Mechanics and Thermodynamics of Propulsion*, Addison Wesley, Reading, MA, 1965, Chaps. 3 and 10, pp. 43–57, 319–338.

[13] Shubin, G., Stephens, A., and Glaz, H., "Steady Shock Tracking and Newton's Method Applied to One-Dimensional Duct Flow," *Journal of Computational Physics*, Vol. 39, 1981, pp. 364–374.

[14] Anderson, J., *Computational Fluid Dynamics*, McGraw–Hill, New York, 1995, Chap. 7, pp. 307, 308.

[15] Lucia, D. J., King, P. I., Beran, P. S., and Oxley, M. E., "Reduced-Order Modeling for a One-Dimensional Nozzle Flow with Moving Shocks," AIAA Paper 2001-2602, June 2001.

[16] Holmes, P., Lumley, J., and Berkooz, G., *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge Univ. Press, Cambridge, England, U.K., 1996, Chap. 3, pp. 86–113.

[17] Beran, P., and Silva, W., "Reduced-Order Modeling: New Approaches for Computational Physics," AIAA Paper 2001-0853, Jan. 2001.

[18] Antonnen, J. S. R., King, P. I., and Beran, P. S., "Accuracy of POD-Based Reduced-Order Models with Deforming Grids," AIAA Paper 2001-2541, June 2001.