misalignment effects. Also, a simple method of discretization that is suitable for programming is discussed here. At this stage, it is suggested that the entire problem of the static aeroelastic analysis of guided launch vehicles, including various misalignment effects, may be approached systematically in order to find out which factor has the greatest contribution to the aeroelastic behavior of the vehicle. As a first step, one may analyze this problem by including the flexibility of vehicle and assuming rigid joints. Later, joint rotations, misalignment effects, etc., may be included in succession. This will provide a clear picture to the understanding of the problem. Also, it may be of considerable help to the designer before he takes up the dynamic aeroelastic investigations.

Acknowledgment

The author is grateful to S. Srinivasan, Deputy Project Director, SLV, VSSC, Thumba Trivandrum, for helpful discussions.

References

[1] Humbad, N. G., "Static Aeroelastic Analysis of Guided Slender Launch Vehicles," *Journal of Spacecraft and Rockets*, Vol. 15, Jan. 1978, pp. 12-17.

[2] Alley, V. L. Jr., and Gerringer, A. H., "An Analysis of Aeroelastic Divergence in Unguided Launch Vehicles," NASA TN D-3281, March 1966.

Comparison of Rocket Nozzle Heat Transfer Calculation Methods

Tarit K. Bose*
Indian Institute of Technology, Madras, India

Nomenclature

C = density-viscosity ratio, Eq. (5d)
c_f = friction coefficient $= 2\tau_w/(\rho_e U_e^2)$
D = local diameter of nozzle, m
Ec = Eckert number, Eq. (5j)
F,G,H = functions defined in Eqs. (5g-i)
f_η = $\bar{u}/U_e$
h = specific enthalpy, J/kg
h^0 = specific total enthalpy, J/kg
$\hat{h}^0$ = dimensionless total enthalpy
k = heat conductivity coefficient, W/m-K
M = Mach number
Pr = heat flux, W/m^2
R_θ, R_{δ_H} = Reynolds number based on θ and δ_H, respectively
r = recovery factor, Eq. (2b); local radius from axis of nozzle, m
r_0 = local nozzle wall radius, m
St_r = Stanton number based on recovery temperature
s = transformed coordinate along nozzle wall, Eq. (5b)
T = temperature, K
U_e = freestream gas velocity, m/s
u,v = gas velocity components along and normal to wall, m/s
V = dimensionless normal velocity, Eq. (5a)
x,y = physical coordinates along and normal to wall, m
β = pressure gradient parameter, Eq. (5e)
γ = specific heat ratio
δ^* = displacement thickness $= \int_0^\infty [1-(\rho u/\rho_e U_e)]\,dy$, m
δ_H = enthalpy thickness $= \int_0^\infty (\rho u/\rho_e U_e)(1-\hat{h}^0)\,dy$, m
ϵ = eddy viscosity coefficient, m^2/s
ϵ_h = eddy conductivity coefficient, m^2/s
η = transformed coordinate, Eq. (5c)
θ = momentum thickness, m $= \int_0^\infty (\rho u/\rho_e U_e)[1-(u/U_e)]\,dy$
μ = dynamic viscosity coefficient, kg/m-s
ν = kinematic viscosity coefficient, m^2/s
ψ = stream function, kg/m-s
ρ = mass density, kg/m^3
τ_w = shear stress on wall, N/m^2
ω = viscosity-temperature exponent in relation $\mu \sim T^\omega$

Subscripts and superscripts

$(\bar{\ })$ = time-averaged property
e = inviscid flow property
r = property at recovery temperature or enthalpy
s,η,x,y = derivatives with respect to the coordinate, respectively
w = property at the wall
$(\ )^*$ = property at recovery temperature or enthalpy

Introduction

HEAT flux and boundary-layer thickness for a rocket nozzle with real-gas properties are compared in the present Note for the purpose of evaluating the applicability of two approximate methods: fully developed turbulent pipe-flow applied to rocket nozzles by Bartz,[1] and an integral method by Bartz.[2] A differential method,[3] which has not been applied previously to realistic rocket nozzle heat transfer calculations, is used here for comparison. Inviscid flow properties for one-dimensional equilibrium flow required for all three methods were computed for a typical solid-propellant rocket of about 75 cm exit diameter and an approximate throat to exit area ratio of 0.2 (Fig. 1). All calculations were done for the nozzle wall temperature of 300 K.

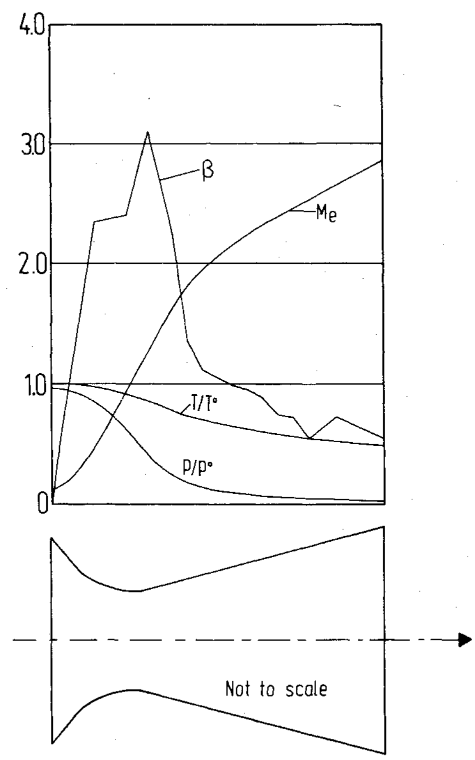

Fig. 1 Inviscid flow properties of a rocket nozzle.

Received March 6, 1978; revision received April 21, 1978.

Index categories: Nozzle and Channel Flow; Boundary Layers and Convective Heat Transfer – Turbulent.

*Professor, Department of Aeronautical Engineering. Associate Fellow AIAA.

Method

The simple fully developed turbulent pipe-flow formulas proposed by Bartz[1] to calculate heat flux in a rocket nozzle are

$$Nu = 0.026(\rho^* U_e D/\mu^*)^{0.8} Pr^{*0.4} \tag{1a}$$

$$q = (k^* Nu/D)(T_r - T_w) \tag{1b}$$

where the properties with superscript asterisk denote those determined for the reference enthalpy

$$h^* = h_e + 0.5(h_w - h_e) + 0.22(h_r - h_e) \tag{2a}$$

In the above equations h_r and T_r are recovery enthalpy and temperature, respectively, which, for turbulent boundary layers, are determined from the following relations:

$$T_r/T_e = 1 + [r(\gamma - 1)M_e^2/2] \qquad (r = \sqrt[3]{Pr_e}) \tag{2b}$$

Equations for the change of momentum and enthalpy thickness, obtained with the integral method of Bartz[2] are

$$\frac{d\theta}{dx} = \frac{c_f}{2} - \theta\left[\frac{1}{D}\frac{dD}{dx} + \frac{2 + (\delta^*/\theta) - M_e^2}{M_e\{1 + [(\gamma - 1)/2]M_e^2\}}\frac{dM_e}{dx}\right] \tag{3a}$$

$$\frac{d\delta_H}{dx} = St_r - \delta_H\left[\frac{1}{D}\frac{dD}{dx} + \frac{(1 - M_e^2)}{M_e\{1 + [(\gamma - 1)/2]M_e^2\}}\frac{dM_e}{dx}\right] \tag{3b}$$

where it has been assumed that the walls are impermeable and maintained at constant temperature. The friction coefficient and the Stanton number based on the recovery temperature follow the empirical relations

$$c_f = \frac{2\tau_w}{\rho_e U_e^2} = \frac{0.0256}{R_\theta^{*0.25}}\left[0.5\left(\frac{T_w}{T_e} + 1\right)\right]^{(\omega - 3)/4} \tag{3c}$$

$$(R_\theta^* = \rho^* U_e \theta/\mu^*)$$

$$St_r = \frac{q}{\rho_e U_e (h_r - h_w)} = \frac{c_f}{2}\left(\frac{R_\theta^*}{R_{\delta_H}^*}\right)^{0.25}\frac{1}{P_r^{*\frac{2}{3}}} \tag{3d}$$

$$(R_{\delta_H} = \rho^* U_e \delta_H/\mu^*)$$

For the differential method, the transformed boundary-layer equations are

$$2sf_{\eta s} + V_\eta + f_\eta = 0 \tag{4a}$$

$$2sf_\eta f_{\eta s} + Vf_{\eta\eta} - (Ff_{\eta\eta})_\eta - \beta[(\rho_e/\rho) - f_\eta^2] = 0 \tag{4b}$$

$$2sf_\eta \hat{h}^0{}_\eta - [G + H\hat{h}^0{}_\eta]_\eta = 0 \tag{4c}$$

where

$$V = -\sqrt{2s}\,\psi_s \tag{5a}$$

$$s = \int_0^x \rho_e \mu_e U_e r_0^2 dx \tag{5b}$$

$$\eta = (\rho_e U_e/\sqrt{2s})\int_0^y r(\rho/\rho_e)dy \tag{5c}$$

$$C = \rho\mu/(\rho_e\mu_e) \tag{5d}$$

$$\beta = 2sU_{e_s}/U_e \tag{5e}$$

$$\hat{h}^0 = (h^0 - h_w)/(h^0{}_e - h_w) \tag{5f}$$

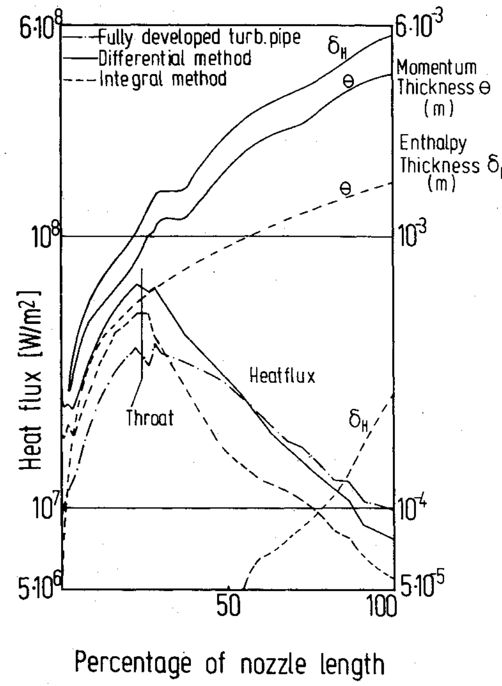

Fig. 2 Comparison of heat flux, momentum thickness, and enthalpy thickness.

$$F = (r/r_0)^2 C[1 + (\epsilon/\nu)] \tag{5g}$$

$$G = (r/r_0)^2 C[1 - Pr^{-1}]Ecf_\eta^2 \tag{5h}$$

$$H = (r/r_0)^2 C[Pr^{-1} + (\epsilon_h/\nu)] \tag{5i}$$

$$Ec = U_e^2/[2(h^0{}_e - h_w)] \tag{5j}$$

Equations (4a-c), together with the usual boundary conditions, are solved by the Runge-Kutta method first for a laminar, locally similar boundary layer, and at subsequent planes downstream by a modified finite-difference iterative scheme of Blottner.[5] The required eddy viscosity coefficient is calculated by the Cebecci-Smith method[3] for the velocity gradient in the inner layer and the local boundary-layer thickness in the outer layer. The eddy conductivity coefficient is obtained[3] over the turbulent Prandtl number.

Numerical Results and Conclusions

The first two methods require a knowledge of the specific heat ratio γ. Local values of γ for the equilibrium composition of the rocket exhaust gas used in these calculations changed from 1.18 in the combustion chamber to 1.29 at the exit plane. The value of ω, needed in the integral method, was 0.75. A few sample calculations with the differential method for both laminar and turbulent boundary layers in high-temperature gas flows showed small, even negative, values of the shape factor which is the ratio of displacement to momentum thickness. Thus, calculations by the integral method were performed with the quite realistic shape factors of 0 and −0.4. No noticeable difference in results with the two shape factors was found.

Finally, numerical results for the heat flux, momentum thickness, and enthalpy thickness, obtained by the three methods are shown in Fig. 2. It is seen that the fully developed turbulent pipe-flow method gives lower heat fluxes in regions where it matters, although it gives an order-of-magnitude correct result. Further, the integral method gives a quite satisfactory result for heat flux and its distribution, although the various boundary-layer thicknesses determined by the method are altogether unsatisfactory.

Acknowledgment

This work was supported by the Alexander von Humboldt Foundation. It was carried out partly at the Institute for Technical Thermodynamics, Technical University, Darmstadt, FRG.

References

[1]Bartz, D. R., "A Simple Equation for Rapid Estimation of Rocket Nozzle Convective Heat Transfer Coefficients," *Jet Propulsion*, Vol. 27, Jan. 1957, pp. 49-51.

[2]Bartz, D. R., "Heat Transfer from Rapidly Accelerating Flows," in *Advances in Heat Transfer*, edited by J. P. Hartnett and T. F. Irvine Jr., Vol. 2, Academic Press, New York, 1965.

[3]Cebecci, T. and Smith, A. M. O., *Analysis of Turbulent Boundary Layers*, Academic Press, New York, 1974.

[4]Eckert, E. R. G., *Analysis of Heat and Mass Transfer*, McGraw Hill, New York, 1972.

[5]Blottner, F. G., "Finite Difference Methods of Solution of the Boundary Layer Equations," *AIAA Journal*, Vol. 8, Feb. 1970, pp. 193-205.

Technical Comments

Comment on "Singularity-Free Extraction of a Quaternion from a Direction-Cosine Matrix"

Richard A. Spurrier*
Honeywell Avionics Division, St. Petersburg, Fla.

KLUMPP[1] describes a direction-cosine matrix to quaternion conversion algorithm which, although valid for all rotations, is unnecessarily sensitive to numerical imprecision. This imprecision can come from the algorithm which produces the direction-cosines or simply from the limited length of the registers that realize the direction-cosines. The sensitivity can be seen by noting that the magnitudes of the quaternion components are computed from the diagonal elements of the direction-cosine matrix. For a rotation of an angle ϕ about the first axis, the diagonal elements are, in order, 1, $\cos\phi$, $\cos\phi$. For small ϕ the off-diagonal elements, $\sin\phi$ and $-\sin\phi$, are more suitable numerically for determining ϕ.

In general, since the magnitudes of the quaternion components are computed as square-roots, the squares of the components are most closely related to the direction-cosine matrix elements. The sensitivity of the magnitude θ of a rotation caused by an error in the square of a quaternion component q_i, to the error, can be expressed

$$\frac{\partial\theta}{\partial q_i^2}=\frac{\sqrt{1-q_i^2}}{|q_i|} \tag{1}$$

Thus precision is reduced whenever any quaternion component is small. This occurs for small rotations, for rotations about an axis nearly perpendicular to a reference axis, and for rotations of nearly 180 deg about any axis. Floating-point arithmetic is of little benefit because the expression for the square of each quaternion component contains a large constant term which limits the scaling reduction.

The following algorithm retains precision for all rotations by computing only the component of largest magnitude as a square-root and by using only this component as a divisor in computing the other components. Since there is always at least one component of magnitude greater than or equal to ½, numerical imprecision will have a limited effect on precision and will not cause a negative square-root argument or division by zero. The expressions follow immediately from the expression for a direction-cosine matrix in terms of quaternion components (given in Klumpp[1]). Klumpp's notation is followed but, in some cases, the (equivalent) negative quaternion is computed.

Choose the largest (algebraically) of $\mathrm{tr}(M)$, $M_{i,i}$ ($i=1$-3). If $\mathrm{tr}(M)$ is largest, compute the quaternion using the following expressions:

$$q_0=\sqrt{1+\mathrm{tr}(M)}/2 \tag{2}$$

$$q_i=(M_{k,j}-M_{j,k})/4q_0 \qquad (i=1\text{-}3) \tag{3}$$

where j and k are chosen so that i, j, k is a cyclic permutation of 1, 2, 3. If $\mathrm{tr}(M)$ is not largest, use the following expressions:

$$q_i=\sqrt{M_{i,i}/2+(1-\mathrm{tr}(M))/4} \tag{4}$$

$$q_0=(M_{k,j}-M_{j,k})/4q_i \tag{5}$$

$$q_l=(M_{l,i}+M_{i,l})/4q_i \qquad (l=j,k) \tag{6}$$

where i, j, k is the cyclic permutation of 1, 2, 3 such that $M_{i,i}$ is the largest above.

The precision of this algorithm was verified with a Fortran program. Quaternions were converted to direction-cosine matrices and then converted back using both algorithms. Precision, with respect to the magnitude of the error, was reduced with Klumpp's algorithm whenever any quaternion component was small, while full precision was always retained with the above algorithm. Precision, with respect to the direction of the rotation (of interest when computing the eigenvector of the direction-cosine matrix), was reduced with Klumpp's algorithm whenever a quaternion vector component was small, while full precision was retained with the above algorithm even (because of the floating point arithmetic) for small rotations.

Received Dec. 7, 1977.
Index categories: Navigation, Control and Guidance Theory; Computer Technology and Computer Simulation Techniques.

*Senior Development Engineer.

References

[1]Klumpp, A. R., "Singularity-Free Extraction of a Quaternion from a Direction-Cosine Matrix," *Journal of Spacecraft and Rockets*, Vol. 13, Dec. 1976, pp. 754-755.