NUMERICAL SIMULATION OF GAS INJECTION

Vladimir V. Riabov, Ph.D.[*]

Professor, Department of Mathematics and Computer Science, Rivier University

Keywords: *boundary layer, Prandtl-Blasius equations, gas injection, exponential box scheme, uniform convergence, nonlinear differential equations, numerical methods, MATLAB code, FORTRAN*

Abstract

Students have used computer programs and created their own programming codes based on exponential box-scheme approximations for solving systems of nonlinear differential equations that contain small parameters for the highest derivative terms or singularities in boundary conditions. The uniform second-order accuracy is obtained for functions and derivatives in this approach. The methodology is widely applied to practical studies of boundary layers with gas injection and combustion.

In the Memory of Ludwig Prandtl and Paul Richard Heinrich Blasius
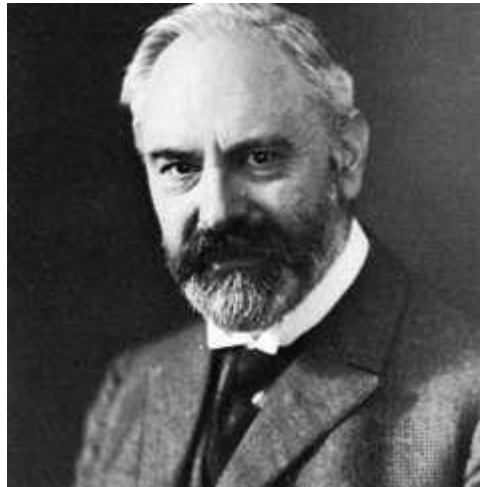
Dr. Ludwig Prandtl. *From [R1]*
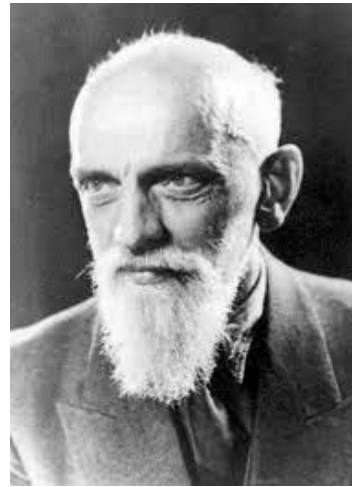
Dr. Heinrich Blasius. *From [R7]*

Ludwig Prandtl (Feb. 4, 1875 — Aug. 15, 1953) was a German physicist who is considered to be the father of aerodynamics [R1]. From 1904 to 1953, he served as professor of applied mechanics at the University of Göttingen, where he established a school of aerodynamics and hydrodynamics that achieved world renown [R2]. In 1925 he became director of the Kaiser Wilhelm (later the Max Planck) Institute for Fluid Mechanics. His discovery (1904) [R3] of the boundary layer, which adjoins the surface of a body moving in air or water, led to an understanding of skin friction drag and of the way in which streamlining reduces the drag of airplane wings and other moving bodies [R1]. On August 8th 1904, he delivered a groundbreaking paper, *Über Flüssigkeitsbewegung bei sehr kleiner Reibung* (*On the Motion of Fluids in Very Little Friction*), at the Third International Mathematics Congress in Heidelberg [R3]. In this paper (republished in [R4] and discussed in detail in [R5, R6]), he described the boundary layer and its importance for drag and streamlining. Several of his students made attempts at closed-form solutions, but failed.

Paul Richard Heinrich Blasius (9 August 1883 – 24 April 1970) was a German fluid dynamics physicist [R7], one of the first students of Dr. Ludwig Prandtl [R8], who first successfully computed (in 1908) the steady two-dimensional laminar flow in the boundary layer that forms on a semi-infinite plate, which is held parallel to a constant unidirectional flow, as an explicit solution of the Prandtl equations [R9].

STATEMENT

Motivation
This case study promotes the development of numerical methods for solving singular (ordinary or partial) differential equations with small coefficients for the highest derivative terms. This singularity leads to the formation of regions with small linear dimensions where gradients of functions are large thus making numerical steps unsteady. (The analytical analyses of these zones do not provide reliable quantitative data estimations). The numerical analysis of such problems by traditional box-schemes is restricted by non-uniform convergence or even divergence of numerical solutions. In this case study, students will find the numerical solutions of the model singular ordinary differential equation evaluated for the linear boundary value problem. The developed numerical method (based on exponential box-schemes) can be used for the analysis of gas flow parameters in boundary layers and viscous shock layers under the conditions of gas injection from the body surface. The practical applications of this methodology include the active heat protection methods for spacecraft and analyses of supersonic hydrogen combustion regimes in engines of modern hypersonic vehicles.

1. Introduction
Experimental, analytical, and numerical studies of fluid flows are challenging, but they are always promising with strong practical applications in sciences and engineering.
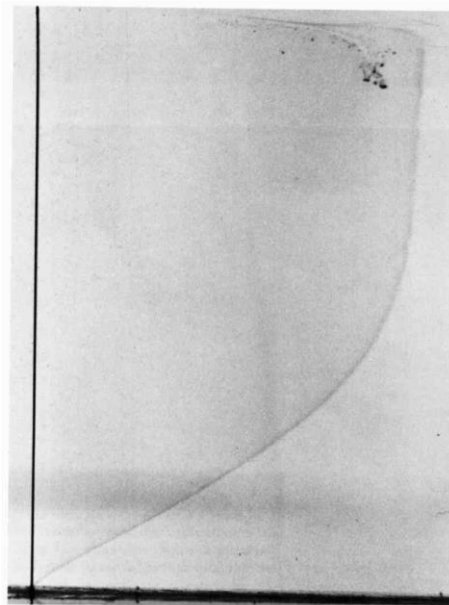

Figure 1. Blasius boundary-layer profile on a flat plate. *Photograph by F. X. Wortmann. From Milton Van Dyke's "An Album of Fluid Motion" [R10, page 22].*

As an example, Milton Van Dyke [R10] describes the tangential velocity profile in the laminar boundary layer on a flat plate (see Figure 1), discovered by Ludwig Prandtl in 1904 [R3] and calculated accurately by his student, Paul Richard Heinrich Blasius in 1908 [R9], which is made visible by tellurium. Water is flowing (from left to right) along the plate at 9 cm/s. The Reynolds number ($Re_d = \rho_\infty u_\infty d/\mu_\infty$) is 500 based on distance d from the leading edge, and the displacement thickness [1] is about 5 mm. (Here index ∞ indicates parameters in the upstream flow, ρ is density, u – steam velocity, and μ – medium viscosity). A fine tellurium wire perpendicular to the plate at the left is subjected to an electrical impulse of a few milliseconds duration. A chemical reaction produces a slender colloidal cloud, which drifts with the steam and is photographed a moment later to define the velocity profile. The photograph was taken by F. X. Wortmann [R10].

In general, the fluid flow of a continuous medium is described by the Navier-Stokes differential equations (see [1, pp. 47-69] and [R11, p. 406]. These equations are nonlinear, and, in the subsonic range, they are of elliptic type [R11]. Only a few explicit solutions are known; also, it is quite difficult to determine approximation solutions [R12, R13]. Therefore, several attempts were made to simplify the Navier-Stokes equations [R14] by omitting nonessential terms. In 1904 [R3], Ludwig Prandtl obtained his famous boundary-layer equations [R11, p. 407] that are also nonlinear, but parabolic ones, and hence much "simpler" from the theoretical and practical points of view than the "classical" Navier-Stokes differential equations. Right after Prandtl's discovery, the boundary-layer theory has become an invaluable modeling approach in various practical applications related to fluid flow studies [R14]. In 1908, Blasius [R9] first successfully computed the flow over a flat plate as an explicit solution of the Prandtl equations. Many other new results [1] were obtained in the XX century with the aid of the Prandtl concept, including the studies of two- and three-dimensional, steady, and unsteady flows of an incompressible or compressible medium consisting of one or multiple components, with or without energy addition, under the influence of electromagnetic or gravitational forces, etc., etc. [R11].

Herbert Keller [R15] made an overview of numerical methods used in boundary-layer theory. Many scholars continue these efforts offering new computing techniques [5-8] and programming codes [R16, R17] with adaptations to new complex practical applications.

Recommended Preliminary Readings

Students are encouraged to read a few articles before starting the simulation of gas injection into a boundary layer:

[R10] Van Dyke, Milton. *An Album of Fluid Motion*. Stanford, CA: Parabolic Press, 1982, 176 p. [Online] http://courses.washington.edu/me431/handouts/Album-Fluid-Motion-Van-Dyke.pdf
  - ○ Read Chapter 2 "Laminar Flow" (pp. 18-23);
  - ○ Review Photograph #30 "Blasius boundary-layer profile on a flat plate" (p. 22).

[1] Schlichting, H. *Boundary-Layer Theory*, 7th edition. New York: McGraw-Hill, 1979. [Online} http://ae.sharif.edu/~viscousflow/Schlichting - Boundary Layer Theory.pdf
  - o Review the properties of Navier-Stokes differential equations (pp. 47-69).
  - o Review the properties of boundary-layer equations for two-dimensional flow (pp. 127-143).

Exploring Numerical Solutions of Prandtl-Blasius Boundary Layer Equation using Maple and MATLAB Codes

Students could use the Maple code [R16] (available in the Supplements folder) that utilizes the Runge-Kutta method to explore numerical solutions of Prandtl-Blasius boundary layer equation:

[R16] Sun, Bo-Hua. "Solving Prandtl-Blasius Boundary Layer Equation Using Maple." Preprint. 5 Sept. 2020. DOI:10.20944/preprints202008.0296.v2. Available from https://pdfs.semanticscholar.org/.

Optionally, the similar study can be conducted using the MATLAB code [R17] (also available in the Supplements folder):

[R17] Bani-Hani, E.H., and Assad, M. E. H. "Boundary-Layer Theory of Fluid Flow past a Flat Plate: Numerical Solution Using MATLAB." *International Journal of Computer Applications*, 2018, 180(18): 6-8. DOI: 10.5120/ijca2018916374. [Online] https://www.researchgate.net/publication/323218834_Boundary-Layer_Theory_of_Fluid_Flow_past_a_Flat-Plate_Numerical_Solution_using_MATLAB.

Modern Challenges

Various modern problems of applied mathematics, thermophysics, and aerodynamics (e.g., stability of boundary layers [1], supersonic hydrogen combustion in hypersonic-vehicle engines [2], and active methods of heat protection of aerospace vehicles [3]) lead to solving differential equations with small coefficients for the highest derivatives and/or at singular boundary conditions. The latter leads to the formation of regions with small linear dimensions where gradients of functions are large. The numerical analysis of such problems by traditional box-schemes [4] is limited by non-uniform convergence or even divergence of numerical solutions. In this study, the numerical solutions of the model singular ordinary differential equation [5] have been evaluated for the linear boundary value problem. The developed numerical method is used for the analysis of gas flow parameters in boundary and viscous shock layers under the conditions of gas injecting from the body surface.

From a mathematical point of view, the increase of the flow rate of gas injection or chemical-reaction rates is equivalent to the existence of a small coefficient for the highest derivative in the boundary-layer (BL) equations [4, 6]. A sublayer with large gradients of functions is created. The gas flow in the boundary layer is studied using a three-point uniform exponential box-scheme [6] and an effective regularization algorithm [7]. The identical problem was considered in [7] by using a two-point exponential box-scheme.

A similar phenomenon is observed in the case of hydrogen combustion at small Reynolds number $Re_0 < 100$ [3, 8]. In this study the models of diffusive combustion of hydrogen, which is injected with different intensity from the surface of a parabolic cylinder into airflow, are considered by using the thin-viscous-shock-layer (TVSL) approach [9] at moderate Reynolds numbers $1500 > Re_0 > 100$.

2. The Model Linear Boundary Value Problem

2.1 Analytical Solution of the Model Second-Order Ordinary Differential Equation with Constant Coefficients

In general, this method is designed for solving the following model equation:

$$\varepsilon u'' + au' - bu = d \tag{1}$$

Here the parameter ε can take on very small magnitudes, and $a \geq 0$, $b \geq 0$. The solution of (1) with constant coefficients is the following [6]:

$$u = A e^{\alpha\eta} + B e^{\gamma\eta} + \psi, \quad \psi = \frac{d}{\varepsilon\alpha\gamma} \tag{2}$$

$$\alpha = -\frac{a}{2\varepsilon} + \sqrt{\frac{a^2}{4\varepsilon^2} + \frac{b}{\varepsilon}} \tag{3}$$

$$\gamma = -\frac{a}{2\varepsilon} - \sqrt{\frac{a^2}{4\varepsilon^2} + \frac{b}{\varepsilon}} \tag{4}$$

where A and B are arbitrary constants, and η is the independent variable.

Finding the Solution of the Model Equation
Students are asked to find the general solution (2) of the model equation (1) with constant parameters $\varepsilon > 0$, $a \geq 0$, and $b \geq 0$.

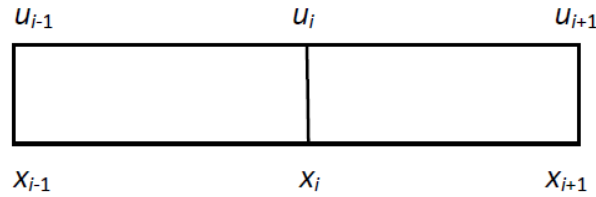### 2.2 Designing the Box Scheme

Figure 2. The box-scheme for the three-point u-function approximation.

The three-point uniform exponential box-scheme was introduced in [6] (see Figure 2 for details). The entire x-argument domain is divided by N intervals (cells) of the same size $h = x_i - x_{i-1}$, i = 1, …, N. We are trying to numerically calculate the values of function u_i in the corresponding points x_i. The solution (2-4) that includes exponents is used to obtain the box-scheme characteristics by considering that the functions as well as the derivatives are continuous in the cells [6, 7]. The identical problem was considered in [7] by using a two-point exponential box-scheme with estimating functions only in centers of two neighboring cells.

2.3 Numerical Solution of the Linear Boundary Value Problem
The linear boundary value problem is studied for the following model singular ordinary differential equation [5] and boundary conditions:

$$\varepsilon u'' - (1 + x^2)u = -(4x^2 - 14x + 4)(1 + x)^2 \tag{5}$$
$$u(0) - u'(0) = 0 \tag{6}$$
$$u(1) + u'(1) = 0 \tag{7}$$

We can expect that the solution of (5)-(7) is sensitive to the value of ε near the x-variable domain boundaries. (A similar effect is observed in the boundary layers [1]). The numerical solutions of (5) with boundary conditions (6) and (7) have been calculated by using the three-point exponential box-scheme [6] described above (the FORTRAN program RPMEB is listed in Appendix-A). The results for the function u and its derivative u' calculated at the cell sizes $h = 1/128$ are shown in Figure 3. At various parameters ε, these demonstrate the formation of regions with small linear dimensions where gradients of functions are large.
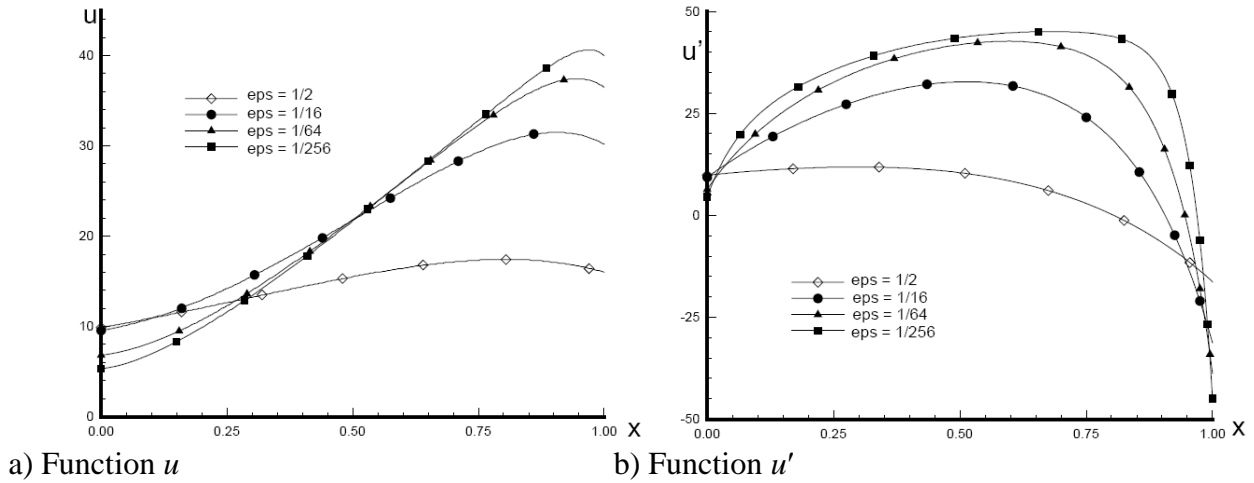
a) Function u b) Function u'

Figure 3. Functions u and u' as solutions of the linear boundary-value problem (5) – (7).

Solving Numerically the Model Singular Ordinary Differential Equation

Students are instructed to use the FORTRAN program RPMEB (listed in Appendix-A) for solving numerically the model singular ordinary differential equation (5) with boundary conditions (6)-(7) at the cell size $h = 1/128$ and various values of parameter $\varepsilon = ½$, ¼, 1/8, 1/16, 1/32, 1/64, 1/128, and 1/256, and observe the behavior of function u and its derivative u' near the interval boundaries at $x \geq 0$ and $x \leq 1$.

3. Uniform Convergence of Box-Schemes

The three-point box-scheme was offered by El-Mistikawy and Werle [6] for the solution of the boundary value problems, which could be described by (5). They assumed that this box-scheme is uniform and of the second order. These properties of the box scheme were proven by Doolan *et al*. [5] under the conditions $b = 0$ (see (1) above). An accurate numerical analysis of the determination of the order of uniform convergence of both the two-point and three-point box-schemes has been conducted in [7] by the method of Doolan *et al*. [5] Two steps (*A* and *B*) of calculations were used.

Step *A*: The parameters $z_{k,\varepsilon}$ are calculated:

$$z_{k,\varepsilon} = \max \left| u_j^{h/m(k)} - u_{2j}^{h/m(k+1)} \right|, \text{ where } m(k) = 2^k, \qquad k = 0, 1, \dots \qquad (8)$$

The maximum is estimated in all grid cells. The parameter h is the size of the largest cell.

Step *B*: The criterion of the uniformity is assessed:

$$z_{k,\varepsilon} \leq C(h/2^k)^P, \ \ k = 0, 1, \dots \qquad (9)$$

Here P is the order of uniform convergence. Assume, that the criterion (9) is equivalent to the criterion:

$$z_{k,\varepsilon} \leq C_\varepsilon(h/2^k)^{P\varepsilon}, \quad k = 0, 1, \dots \qquad (10)$$

where parameters C_ε and P_ε are independent of the parameter k, and therefore, the parameter P_ε could be estimated by formula:

$$P_\varepsilon = \log_2(z_{k,\varepsilon}/z_{k+1,\varepsilon}) \qquad (11)$$

Compared to the study of Doolan *et al*. [5] the calculations of the parameters in (11) have been conducted with double precision. The order of uniform convergence P_ε has been estimated by the considered procedure of calculations at constant ε and at $h \to 0$. The values of the parameter P_ε have been calculated for the linear boundary-value problem (5)-(7) using the FORTRAN program RPMEB (listed in Appendix-A).

Studying the Uniform Convergence of Exponential Box-Schemes

Students are instructed to use the FORTRAN program RPMEB (listed in Appendix-A) for solving numerically the model singular ordinary differential equation (5) with boundary conditions (6)-(7) at various values of parameter ε = ½, ¼, 1/8, 1/16, 1/32, 1/64, 1/128, and 1/256, and various cell sizes h = 1/8, 1/16, 1/32, 1/64, and 1/128. Following algorithm (8)-(11), they calculate values of the parameter P_ε (the estimated order of uniform convergence) for function u and its derivative u' at various values of ε and h. The results are shown in Tables 1 and 2, correspondingly (see the sample of students' report below).

Table 1: Parameters P_ε for function u at various values of ε and h

ε	h=1/8	h=1/16	h=1/32	h=1/64	h=1/128
1/2	1.998	1.999	2.000	2.000	2.000
1/4	1.995	1.999	2.000	2.000	2.000
1/8	1.990	1.997	1.999	2.000	2.000
1/16	1.979	1.995	1.999	2.000	2.000
1/32	1.959	1.989	1.997	1.999	2.000
1/64	1.921	1.978	1.994	1.999	2.000
1/128	1.924	1.958	1.989	1.997	1.999
1/256	1.932	1.983	1.992	1.994	1.999

Table 2: Parameters P_ε for function u' at various values of ε and h

ε	h=1/8	h=1/16	h=1/32	h=1/64	h=1/128
1/2	1.988	2.000	2.000	2.000	2.000
1/4	1.996	1.997	2.000	2.000	2.000
1/8	1.992	1.997	1.999	2.000	2.000
1/16	1.984	1.992	1.999	2.000	2.000
1/32	1.959	1.992	1.998	1.999	2.000
1/64	1.934	1.980	1.996	1.999	2.000
1/128	1.874	1.966	1.991	1.998	1.999
1/256	1.772	1.931	1.982	1.995	1.999

In the case of the three-point box-scheme, the magnitudes P_ε are shown in Tables 1 and 2 for functions u and u', correspondingly, which represent the solutions of (5)-(7). The obtained data leads to the conclusion that this exponential box-scheme is of second order ($P_\varepsilon \approx 2.0$) for the uniform convergence characterized by the criteria (8)-(10).

4. Gas Injection into a Boundary Layer

Consider the perfect-gas flow in the boundary layer (BL) near the flow-stagnation point of a blunt body with uniform gas injection from the body surface [1]. The flow properties can be characterized by the

tangential component of the velocity U, enthalpy S, and their derivatives U', U'', S', and S'' along the normal Y. The system of BL equations acquires the following form [4, 7]:

$$U'' + fU' + \beta(S + 1 - U^2) = 0 \tag{12}$$
$$f' = U' \tag{13}$$
$$S'' + \sigma f S' = 0 \tag{14}$$

where the Faulkner-Scan constant [1] $\beta = (1 + j)^{-1}$ characterizes the pressure gradient in inviscid flow; $j = 0$ or 1 in plane and axisymmetric cases (e.g., near the blunt plate or axisymmetric cylinder) correspondingly; and $\sigma = v/\kappa = 0.72$ is the Prandtl number (here v is kinematic viscosity and κ – coefficient of thermal diffusivity).

Boundary conditions are the following:

At the body surface ($Y = 0$) considering the gas injection:

$$f = f_w = \text{const}, \quad U' = 0, S = S_w \tag{15}$$

At the external boundary of the layer ($Y \to \infty$):
$$U = 1, \quad S = 0. \tag{16}$$

In (15) the parameter f_w characterizes the mass flow rate of the gas injected from the body to the boundary layer. Special box-schemes with uniform convergence [5, 10] or exponential schemes [4, 6, 7] should be used in order to solve the problem at large values of the parameter $|f_w|$. The exponential box-schemes developed [6, 7] have the second order of uniform convergence. The above-considered three-point exponential box-scheme has been used for the numerical solution of (12)-(16) under the conditions of moderate and intensive gas injection ($|f_w| > 0$) from the thermally isolated body surface ($S_w = 0$). (The FORTRAN program RPEBOX3 is listed in Appendix-B).

Studying the Effects of Gas Injection into a Boundary Layer
Students are instructed to use the FORTRAN program RPEBOX3 (listed in Appendix-B) for solving numerically the system of BL equations (12)-(14) with boundary conditions (15)-(16) at various gas-injection parameters ($f_w = 0, -2.5, -10, -25$). We expect that students obtain the profiles of the tangential component of the velocity U and its derivative U' along the normal Y at the stagnation point on the surface of the axisymmetric blunt body ($\beta = 0.5$) that are similar to those shown in Figure 4 for various gas-injection parameters ($f_w = 0, -2.5, -10, -25$). The presence of the injected gas flow significantly changes the flow structure. As the gas-injection rate $|f_w|$ increases, the boundary layer becomes thicker (see the changes of profiles of the tangential component of the velocity U shown in Figure 4a), and the friction on the body surface decreases (see the changes of profiles of derivative of the tangential component of the velocity U' shown in Figure 4b).
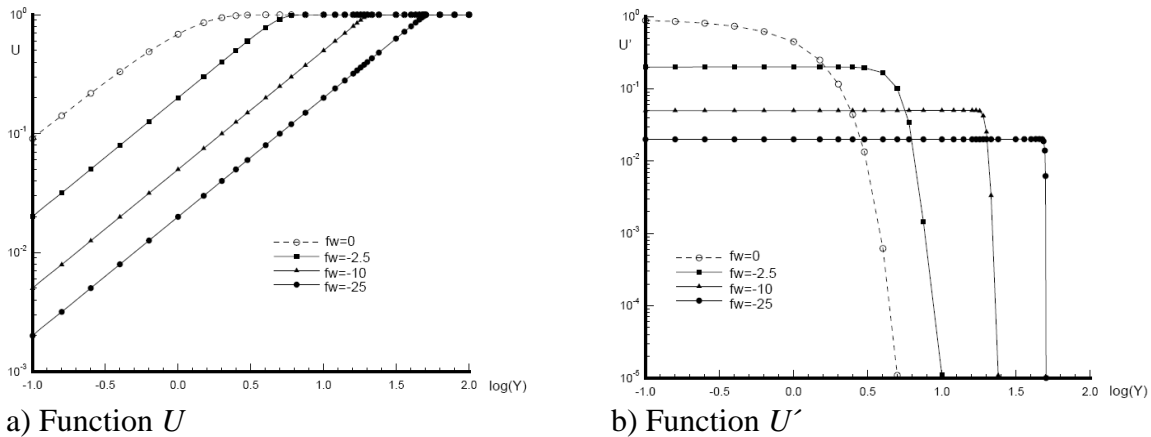
a) Function *U*   b) Function *U´*

Figure 4. Functions *U* and *U´* across the boundary layer for various gas injection factors.

Supplements

The case-study supplementary materials (computer codes and files listed in Appendix C) will be available for students and instructors from the website: https://www.simiode.org/resources/8640.

Downloading Software

If your school does not have a license for running *MATLAB software*, then a student can get MATLAB for a 30-day trial. Details and downloads are available from https://www.mathworks.com/campaigns/products/trials.html.

A free 15-day trial of *Maple software* (*Student's Edition*) can be downloaded from https://www.maplesoft.com/products/maple/free-trial/.

The FORTRAN software free-license kit (as a part of the Intel® oneAPI Base Toolkit) can be downloaded from https://software.intel.com/content/www/us/en/develop/tools/oneapi/base-toolkit/download.html. It can be installed on Windows, macOS or Linux operating systems.

Concluding Remarks on Students' Involvement

The numerical results (obtained by students) demonstrate the effectiveness of using the exponential box-scheme that has a property of uniform second-order convergence in the full range of small parameters such as the inverse gas-injection factors. They could lead to the finding that moderate gas injections lead to decreasing friction on the body surface.

After in-class discussions of the considered case studies, each student can continue working on a selected case analyzing in-depth mathematical models, creating computer codes (in MATLAB, Maple, C/C++, Java, or FORTRAN), running the codes with various parameters, comparing computations with experimental data, and presenting the findings to classmates.

Comments

At the beginning of case-study activities, students are introduced (in historical retrospective [R3-R5, R9, R14]) to the concept of boundary layer [1], mathematical properties of Prandtl-Blasius equations [R11],

and their numerical solutions using Maple [R16] and MATLAB codes [R17]. Students should be instructed on how to run programming codes in Maple and/or MATLAB environment.

The existence of singularities in the fluid-motion nonlinear differential equations that contain small parameters for the highest derivative terms or singularities in boundary conditions requires the utilization of exponential box-scheme approximations for finding reliable numerical solutions of the equations. Students find the analytical solution of the model second-order ordinary differential equation with constant coefficients [7] and apply it in the study of uniform convergence of exponential box-schemes [5, 7] using the FORTRAN program RPMEB (listed in Appendix-A). Students should be instructed on how to run programming codes in FORTRAN environment.

Finally, the unique practical case of uniform gas injection from the body surface to boundary layer [6, 7] is analyzed in detail by students using the FORTRAN program RPBOX3 (listed in Appendix-B). The values of gas-injection rate significantly affect the flow structure, boundary-layer thickness, and the friction on the body surface. A similar approach can be used in the analyses of even more complicated cases of injecting chemically reactive gases (e.g., hydrogen) into multicomponent boundary layer, simulating supersonic combustion in a hypersonic-vehicle engine [8].

Acknowledgments

References

[1] Schlichting, H. *Boundary-layer theory*, 7th edition. New York: McGraw-Hill, 1979. [Online] http://ae.sharif.edu/~viscousflow/Schlichting - Boundary Layer Theory.pdf

[2] Billig, F. S. "Research on Supersonic Combustion." *Journal of Propulsion and Power*, 1993; 9(4): 499-514.

[3] Pappas, C. C. and Lee G. "Heat Transfer and Pressure on a Hypersonic Blunt Cone with Mass Addition." *AIAA Journal*, 1970 8(8): 984-995.

[4] Liu, T. M. and Chiu, H. H. "Fast and Stable Numerical Method for Boundary-Layer Flow with Massive Blowing." *AIAA Journal*, 1976 14(1): 114-116.

[5] Doolan, E. P., Miller, J. J. H., and Schilders, W. H. A. *Uniform numerical methods for problems with initial and boundary layers*. Dublin: Boole Press, 1980.

[6] El-Mistikawy, T. M. and Werle, M. J. "Numerical Method for Boundary Layers with Blowing - the Exponential Box Scheme." *AIAA Journal*, 1978 16(7): 749-751.

[7] Riabov, V. and Provotorov, V. P. "Exponential Box Schemes for Boundary-Layer Flows with Blowing." *Journal of Thermophysics and Heat Transfer*, 1996 10(1): 126-130.

[8] Riabov, V. and Botin, A. V. "Hypersonic Hydrogen Combustion in the Thin Viscous Shock Layer." *Journal of Thermophysics and Heat Transfer*, 1995 9(2): 233-239.

[9] Cheng, H. K. *The blunt-body problem in hypersonic flow at low Reynolds number*. Cornell Aeronautical Lab., AF-1285-A-10. Buffalo, New York, 1963.

[10] Keller, H. B. "Accurate Difference Methods for Nonlinear Two-Point Boundary Value Problems." *SIAM Journal of Numerical Analysis*, 1974 11(2): 305-320.

Further Readings

[R1] Crouch, Tom D. "Ludwig Prandtl." *Encyclopedia Britannica*. [Online] https://www.britannica.com/biography/Ludwig-Prandtl.

[R2] Eckert, Michael. "The Beginnings of Fluid Mechanics in Göttingen, 1904–14". *The Dawn of Fluid Dynamics: A Discipline between Science and Technology*. Weinheim: Wiley-VCH, 2006, pp. 31–56.

[R3] Prandtl, Ludvig. "Über Flüssigkeits-bewegungen bei sehr kleiner Reibung." *Verhandlinger II Int. Math.-Kongr. Heidelberg*. Leipzig: Teubner, 1904, pp. 484-494.

[R4] Tollmien, Walter; Schlichting, Hermann; Görtler, Henry; Riegels, F. W. (eds.), "Über Flüssigkeitsbewegung bei sehr kleiner Reibung", *Ludwig Prandtl Gesammelte Abhandlungen: zur angewandten Mechanik, Hydro- und Aerodynamik* (in German), Springer Berlin Heidelberg, 1961, pp. 575–584, doi:10.1007/978-3-662-11836-8_43

[R5] Anderson, John D. (2005). "Ludwig Prandtl's Boundary Layer". *Physics Today*. 2005, 58(12): 42–48. doi:10.1063/1.2169443

[R6] *IUTAM Symposium on One Hundred Years of Boundary Layer Research: Proceedings of the IUTAM symposium held at DLR-Göttingen, Germany, August 12-14, 2004*. International Union of Theoretical and Applied Mechanics. Dordrecht: Springer. 2006.

[R7] Hager, W.H., "Blasius: A life in research and education," *Experiments in Fluids,* 2003, 34: 566–571. DOI: 10.1007/s00348-002-0582-9. [Online] http://users.df.uba.ar/cobelli/Estructura_1/Blasius-Biography.pdf.

[R8] Paul Richard Heinrich Blasius. [Online] https://en.wikipedia.org/wiki/Paul_Richard_Heinrich_Blasius.

[R9] Blasius, P. R. Heinrich. "Grenzschichten in Flüssigkeiten mit kleiner Reibung." *Zeitschrift für Mattematik und Physik,* 1908, 56(1): 1-37. (English translation is available: H. Blasius, "The Boundary Layers in Fluids with Little Friction," *NACA Technical Memorandum 1256*, Washington D.C., 1950. [Online] https://ntrs.nasa.gov/citations/20050028493).

[R10] Van Dyke, Milton. *An Album of Fluid Motion*. Stanford, CA: Parabolic Press, 1982, 176 p. [Online] http://courses.washington.edu/me431/handouts/Album-Fluid-Motion-Van-Dyke.pdf.

[R11] Nickel, Karl. "Prandtl's Boundary-Layer Theory from the Viewpoint of a Mathematician". *Annual Review of Fluid Mechanics*, 1973, 5: 405-428. https://doi.org/10.1146/annurev.fl.05.010173.002201.

[R12] Van Dyke, Milton. *Perturbation Methods in Fluid Mechanics*. Stanford, CA: Parabolic Press, 1975. [Online] https://cdn.preterhuman.net/texts/science_and_technology/physics/Fluid_Mechanics/Perturbation%20Methods%20In%20Fluid%20Mechanics%20-%20Van%20Dyke.pdf.

[R13] Van Dyke, Milton. "Higher-Order Boundary-Layer Theory." *Annual Review of Fluid Mechanics*, 1969, 1: 265-292. https://doi.org/10.1146/annurev.fl.01.010169.001405.

[R14] Tani, Itiro. "History of Boundary Layer Theory". Annual Review of Fluid Mechanics, 1977, 9: 87-111. https://doi.org/10.1146/annurev.fl.09.010177.000511.

[R15] Keller, Herbert B. "Numerical Methods in Boundary-Layer Theory". *Annual Review of Fluid Mechanics*, 1978, 10: 417-433. https://doi.org/10.1146/annurev.fl.10.010178.002221.

[R16] Sun, Bo-Hua. "Solving Prandtl-Blasius Boundary Layer Equation Using Maple." Preprint. Posted: 5 September 2020. DOI:10.20944/preprints202008.0296.v2. Available from https://pdfs.semanticscholar.org/.

[R17] Bani-Hani, E.H., and Assad, M. E. H. "Boundary-Layer Theory of Fluid Flow past a Flat Plate: Numerical Solution Using MATLAB." *International Journal of Computer Applications*, 2018, 180(18): 6-8. DOI: 10.5120/ijca2018916374. [Online] https://www.researchgate.net/publication/323218834_Boundary-Layer_Theory_of_Fluid_Flow_past_a_Flat-Plate_Numerical_Solution_using_MATLAB.

Appendix-A: FORTRAN Program RPMEB for Solving the Model Singular Ordinary Differential Equation

```
      PROGRAM RPMEB
C*********************************************************************
C****** PROGRAM 'RPMEB' IS FOR SOLVING MODEL SINGULAR EQUATION ***
C****** BY USING A THREE-POINT BOX-SCHEME                      ******
C*********************************************************************
   DOUBLE PRECISION A(201),D(201),C(201),B(201), DY1, DYL1, DYL5,
  *C0,C1,C5, SIM, BETA, PR, FW, TW, SW, YN, Y0, DYL, XMAX,
  *Y(201),U(201),U0(201),
  *A1(201),B1(201),FI1(201),DU(201),
  *DY(201),Y5(201), EPS, EPS1, AA, C4, C14,
  *F15, U15, S15, T1, R1, S1, T2, R2, S2, EA, EB, AA1, BB1, SU
C------- MAJOR CONSTANTS ---------------------------------------
   N = 201
   N1 = N - 1
   C0 = 0.0D0
   C1 = 1.0D0
   C5 = 0.5D0
   C4 = 4.0D0
   C14 = 14.0D0
   AA = C0
   EPS = C1/2.0D0
   EPS1 = C1/EPS
   YN = 1.0D0
   Y0 = 0.0D0
   DYL = (YN - Y0)/N1
   DYL1 = C1/DYL
   DYL5 = C5*DYL
   ITER = 0
   ITSTOP = 500
   Y(1) = Y0
   Y5(1) = DYL5
   DO 14 I = 2, N1
   Y(I) = Y(1) + DYL*(I-1)
   Y5(I) = Y(I) + DYL5
 14 CONTINUE
   Y(N) = YN
C------- INITIAL FIELD OF PARAMETERS/FUNCTIONS --------------------
C    OPEN (UNIT = 8, FILE = 'RMES0')
C    REWIND 8
C    READ (8,905) ITER, EPS, DYL
C    DO 810 I = 1, N
C    READ (8,905) IP, Y(I), U(I), DU(I)
C 810 CONTINUE
```

```
C     REWIND 8
      DO 10 I = 1, N
      U(I) = C0
 10 CONTINUE
C----------------------------------
      OPEN (UNIT = 9, FILE = 'RMES05')
      REWIND 9
C+++++++++++++++++ GLOBAL ITERATIONS ++++++++++++++++++++++++
 1 CONTINUE
      ITER = ITER + 1
      DO 11 I = 1, N
      U0(I) = U(I)
 11 CONTINUE
C------- U-SOLUTION ------------------------------------------
      DO 20 I = 2, N
      IM = I - 1
      DY(IM) = Y(I) - Y(IM)
      S15 = -(Y5(IM)*(C14+C4*Y5(IM))+C4)*(C1+Y5(IM))*(C1+Y5(IM))
      F15 = -C5*AA*EPS1
      U15 = EPS1*(C1+Y5(IM)*Y5(IM))
      A1(IM) = F15 + DSQRT(F15*F15+U15)
      B1(IM) = F15 - DSQRT(F15*F15+U15)
      FI1(IM) = EPS1*S15/(A1(IM)*B1(IM))
 20 CONTINUE
      DO 21 I = 2, N1
      IM = I - 1
      T1 = C1 - C1/DEXP((A1(IM)-B1(IM))*DY(IM))
      R1 = (A1(IM)-B1(IM))*DEXP(B1(IM)*DY(IM))/T1
      S1 = (A1(IM) - (C1-T1)*B1(IM))/T1
      T2 = C1 - C1/DEXP((A1(I)-B1(I))*DY(I))
      R2 = (A1(I)-B1(I))/DEXP(A1(I)*DY(I))/T2
      S2 = (A1(I)*(C1-T2)-B1(I))/T2
      A(IM) = R1
      D(I) = -(S1+S2)
      C(I) = R2
      B(I) = (R1-S1)*FI1(IM) + (R2-S2)*FI1(I)
 21 CONTINUE
C------- U-BOUNDARY CONDITIONS ------------------------------
      D(1) = C1 + DYL1
      C(1) = -DYL1
      B(1) = C0
      A(N1) = -DYL1
      D(N) = C1 + DYL1
      B(N) = C0
      CALL TRI(N, A, D, C, B, U)
```

```
C------------------ END OF THE U-SOLUTION -----------------------
C----------- DU-SOLUTION ----------------------------------------
      DO 25 I = 1, N1
      I1 = I+1
      EA = DEXP(DY(I)*A1(I))
      EB = DEXP(DY(I)*B1(I))
      AA1 = ((U(I1)-U(I)*EB)-FI1(I)*(C1-EB))/(EA-EB)
      BB1 = ((U(I1)-U(I)*EA)-FI1(I)*(C1-EA))/(EB-EA)
      DU(I1) = AA1*A1(I)*EA+BB1*B1(I)*EB
      IF (I.EQ.1) DU(I)=AA1*A1(I)+BB1*B1(I)
      CONTINUE
   25 CONTINUE
C------------------ END OF THE DU-SOLUTION -----------------------
C-------- CONVERGENCE ESTIMATION --------------------------------
      XMAX = C0
      DO 12 I = 1, N
      XMAX=XMAX + DABS(U(I)-U0(I))
   12 CONTINUE
      IF (ITER. EQ. ITSTOP) GO TO 2
      CONTINUE
      IF (XMAX. GE. 1.0D-4) GO TO 1
    2 CONTINUE
      WRITE (9, 905) ITER, EPS, DYL
      DO 800 I = 1, N
      I1 = I
      WRITE (9, 905) I, Y(I), U(I), DU(I)
  800 CONTINUE
      REWIND 9
  905 FORMAT (I3, 3(D11.4))
      STOP
      END
C*************** END OF THE PROGRAM RPMEB ***********************
      SUBROUTINE TRI(N,A,D,C,B,X)
C+++++++++ FOR SOLVING THE TRIDIAGONAL SYSTEMS +++++++++++++++++
      DOUBLE PRECISION A(N), D(N), C(N), B(N), X(N), XMULT
      DO 2 I = 2, N
      I1 = I-1
      XMULT = A(I1)/D(I1)
      D(I) = D(I) - XMULT*C(I1)
      B(I) = B(I) - XMULT*B(I1)
    2 CONTINUE
      X(N) = B(N)/D(N)
      DO 3 I = N-1,1,-1
      X(I) = (B(I) - C(I)*X(I+1))/D(I)
    3 CONTINUE
```

```
      RETURN
      END
C++++++++++++++++++++ END OF THE SUBROUTINE TRI ++++++++++++++++
```

Appendix-B: FORTRAN Program RPEBOX3 for Solving the Boundary Layer Problem with Gas Injection

```
      PROGRAM RPEBOX3
C***********************************************************************
C****** PROGRAM 'RPEBOX3' IS FOR SOLVING BOUNDARY-LAYER PROBLEMS ***
C****** WITH GAS INJECTION USING A THREE-POINT BOX-SCHEME        ***
C***********************************************************************
      DOUBLE PRECISION A(229),D(229),C(229),B(229),YPR(32), DY1,
     *C0,C1,C5, SIM, BETA, PR, FW, TW, SW, YN, Y0, DYL, DY12,XMAX,
     *Y(229),U(229),F(229),S(229),U0(229),F0(229),S0(229),
     *A1(229),B1(229),FI1(229),DU(229),DS(229),
     *DY(229),YLOG(229),
     *F15, U15, S15, T1, R1, S1, T2, R2, S2, EA, EB, AA1, BB1, SU
C------- MAJOR CONSTANTS ----------------------------------------
      N = 229
      N1 = N - 1
      NL = 11
      C0 = 0.0D0
      C1 = 1.0D0
      C5 = 0.5D0
      SIM = C1
      BETA = C1/(C1 + SIM)
      YN = 2.0D0
      Y0 = - 1.0D0
      DYL = (C0 - Y0)/20
      DY12 = C1/12.0D0
C-------    DY12 = C0 ------ AS AN OPTION
      ITER = 0
      ITSTOP = 500
      Y(1) = C0
      YLOG(1) = C0
      DY1 = 10**Y0/(NL-1)
      DO 14 I = 2, NL-1
      Y(I) = Y(1) + DY1*(I-1)
      YLOG(I) = DLOG10(Y(I))
   14 CONTINUE
      DO 15 I = NL, 31
      YLOG(I) = Y0 + DYL*(I-NL)
      Y(I) = Y(1) + (10.0D0)**YLOG(I)
   15 CONTINUE
      DO 16 I = 32, N
```

```
      Y(I) = Y(31) + 0.5D0*(I-31)
      YLOG(I) = DLOG10(Y(I))
   16 CONTINUE
      DY1 = (YN-Y0)/30
      DO 17 J = 1, 31
      YPR(J) = 10**(DY1*(J-1))
   17 CONTINUE
      YPR(32) = 200.0D0
C------- INITIAL FIELD OF PARAMETERS/FUNCTIONS ------------
      OPEN (UNIT = 8, FILE = 'REXPON10')
      REWIND 8
      READ (8,905) ITER, BETA, PR, FW, TW
      PR = 0.72D0
      FW = -10.0D0
      TW = 0.1D0
      SW = TW - C1
      DO 810 I = 1, N
      READ (8,905) IP, Y(I), YLOG(I), U(I), DU(I), F(I), S(I), DS(I)
  810 CONTINUE
      REWIND 8
      DO 10 I = 1, N
      S(I) = C1 + S(I)
      DS(I) = C0
   10 CONTINUE
C---------------- BOUNDARY CONDITIONS AT THE WALL ---------
      U(1) = C0
      F(1) = FW
      S(1) = TW
      OPEN (UNIT = 9, FILE = 'REXP1001')
      REWIND 9
C+++++++++++++++++ GLOBAL ITERATIONS ++++++++++++++++++++
    1 CONTINUE
      ITER = ITER + 1
      DO 11 I = 1, N
      U0(I) = U(I)
      F0(I) = F(I)
      S0(I) = S(I)
   11 CONTINUE
C---------------- U-SOLUTION ------------------------------------
      DO 20 I = 2, N
      IM = I - 1
      DY(IM) = Y(I) - Y(IM)
      S15 = C5*(S(IM)+S(I))
      F15 = -C5*C5*(F(IM)+F(I))
      U15 = C5*(U(IM)+U(I))*BETA
```

```
      A1(IM) = F15 + DSQRT(F15*F15+U15)
      B1(IM) = F15 - DSQRT(F15*F15+U15)
      FI1(IM) = - BETA*S15/(A1(IM)*B1(IM))
  20  CONTINUE
      DO 21 I = 2, N1
      IM = I - 1
      T1 = C1 - C1/DEXP((A1(IM)-B1(IM))*DY(IM))
      R1 = (A1(IM)-B1(IM))*DEXP(B1(IM)*DY(IM))/T1
      S1 = (A1(IM) - (C1-T1)*B1(IM))/T1
      T2 = C1 - C1/DEXP((A1(I)-B1(I))*DY(I))
      R2 = (A1(I)-B1(I))/DEXP(A1(I)*DY(I))/T2
      S2 = (A1(I)*(C1-T2)-B1(I))/T2
      A(IM) = R1
      D(I) = -(S1+S2)
      C(I) = R2
      B(I) = (R1-S1)*FI1(IM) + (R2-S2)*FI1(I)
  21  CONTINUE
C------- U-BOUNDARY CONDITIONS ------------------------------
      D(1) = C1
      C(1) = C0
      B(1) = C0
      A(N1) = C0
      D(N) = C1
      B(N) = C1
      CALL TRI(N, A, D, C, B, U)
C------------------- END OF THE U-SOLUTION ----------------------
C---------------------- F-SOLUTION -------------------------------------
      DO 25 I = 1, N1
      I1 = I+1
      EA = DEXP(DY(I)*A1(I))
      EB = DEXP(DY(I)*B1(I))
      AA1 = ((U(I1)-U(I)*EB)-FI1(I)*(C1-EB))/(EA-EB)
      BB1 = ((U(I1)-U(I)*EA)-FI1(I)*(C1-EA))/(EB-EA)
      DU(I1) = AA1*A1(I)*EA+BB1*B1(I)*EB
      IF (I.EQ.1) DU(I)=AA1*A1(I)+BB1*B1(I)
      CONTINUE
  25  CONTINUE
C--- THE EULER-MACLAURIN SUMMATION FORMULA FOR INTEGRATION -----
      F(1) = FW
      SU = C0
      DO 30 I = 2, N
      I1 = I - 1
      SU = SU + DY(I1)*(C5*(U(I)+U(I1))-DY12*DY(I1)*(DU(I)-DU(I1)))
      F(I) = F(1) + SU
  30  CONTINUE
```

```
C------------------ END OF THE F-SOLUTION -----------------------
C------------------------- S-SOLUTION ---------------------------------
      DO 40 I = 2, N
      IM = I - 1
      S15 = C5*(S(IM)+S(I))
      F15 = -C5*C5*(F(IM)+F(I))*PR
      A1(IM) = F15 + DSQRT(F15*F15+S15)
      B1(IM) = F15 - DSQRT(F15*F15+S15)
      FI1(IM) = - S15*S15/(A1(IM)*B1(IM))
   40 CONTINUE
      DO 41 I = 2, N1
      IM = I - 1
      T1 = C1 - C1/DEXP((A1(IM)-B1(IM))*DY(IM))
      R1 = (A1(IM)-B1(IM))*DEXP(B1(IM)*DY(IM))/T1
      S1 = (A1(IM) - (C1-T1)*B1(IM))/T1
      T2 = C1 - C1/DEXP((A1(I)-B1(I))*DY(I))
      R2 = (A1(I)-B1(I))/DEXP(A1(I)*DY(I))/T2
      S2 = (A1(I)*(C1-T2)-B1(I))/T2
      A(IM) = R1
      D(I) = -(S1+S2)
      C(I) = R2
      B(I) = (R1-S1)*FI1(IM) + (R2-S2)*FI1(I)
   41 CONTINUE
C------- S-BOUNDARY CONDITIONS --------------------------------
      D(1) = C1
      C(1) = C0
      B(1) = TW
      A(N1) = C0
      D(N) = C1
      B(N) = C1
      CALL TRI(N, A, D, C, B, S)
C------------------ END OF THE S-SOLUTION -----------------------
C------------------------ DS-SOLUTION ---------------------------------
      DO 45 I = 1, N1
      I1 = I+1
      EA = DEXP(DY(I)*A1(I))
      EB = DEXP(DY(I)*B1(I))
      AA1 = ((S(I1)-S(I)*EB)-FI1(I)*(C1-EB))/(EA-EB)
      BB1 = ((S(I1)-S(I)*EA)-FI1(I)*(C1-EA))/(EB-EA)
      DS(I1) = AA1*A1(I)*EA+BB1*B1(I)*EB
      IF (I.EQ.1) DS(I)=AA1*A1(I)+BB1*B1(I)
      CONTINUE
   45 CONTINUE
C-------- CONVERGENCE ESTIMATION ----------------------------------
      XMAX = C0
```

```
        DO 12 I = 1, N
        XMAX=XMAX + DABS(U(I)-U0(I))+DABS(F(I)-F0(I))+DABS(S(I)-S0(I))
     12 CONTINUE
        IF (ITER. EQ. ITSTOP) GO TO 2
        CONTINUE
        IF (XMAX. GE. 1.0D-4) GO TO 1
      2 CONTINUE
        WRITE (9, 905) ITER, BETA, PR, FW, TW
C       K = 2
        DO 800 I = 1, N
        S(I) = (S(I)-C1)/SW
        DS(I) = DS(I)/SW
        WRITE (9,905) I, Y(I), YLOG(I), U(I), DU(I), F(I), S(I), DS(I)
    800 CONTINUE
C       DO 801 I = NL + 1, N
C       DYL = YPR(K)
C       IF (Y(I).GE.DYL) GO TO 805
C       GO TO 801
C 805 WRITE (9,905) I, Y(I), YLOG(I), U(I), DU(I), F(I), S(I), DS(I)
C       K = K + 1
    801 CONTINUE
        REWIND 9
    905 FORMAT (I3, 7(D11.4))
        STOP
        END
C*************** END OF THE PROGRAM 'RPEBOX3' ********************
        SUBROUTINE TRI(N,A,D,C,B,X)
C++++++++++ FOR SOLVING THE TRIDIAGONAL SYSTEMS ++++++++++
        DOUBLE PRECISION A(N), D(N), C(N), B(N), X(N), XMULT
        DO 2 I = 2, N
        I1 = I-1
        XMULT = A(I1)/D(I1)
        D(I) = D(I) - XMULT*C(I1)
        B(I) = B(I) - XMULT*B(I1)
      2 CONTINUE
        X(N) = B(N)/D(N)
        DO 3 I = N-1,1,-1
        X(I) = (B(I) - C(I)*X(I+1))/D(I)
      3 CONTINUE
        RETURN
        END
C+++++++++++++++ END OF THE SUBROUTINE TRI +++++++++++++++
```

Appendix C: Supplements

The following case-study supplementary materials (computer codes and PDF files) are available for students and instructors from the website: https://www.simiode.org/resources/8640:

- FORTRAN code files:
    - RPMEB.FOR for Solving the Model Singular Ordinary Differential Equation.
    - RPEBOX3.FOR for Solving the Boundary Layer Problem with Gas Injection.
- Maple code with annotated text:
    - Bo-Hua Sun_2020.pdf – The copy of [R16] Sun, Bo-Hua. "Solving Prandtl-Blasius Boundary Layer Equation Using Maple." Preprint. Posted: 5 September 2020. DOI:10.20944/preprints202008.0296.v2. Available from https://pdfs.semanticscholar.org/.
- MATLAB code with annotated text:
    - banihani-2018-ijca-916374.pdf – The copy of [R17] Bani-Hani, E.H., and Assad, M. E. H. "Boundary-Layer Theory of Fluid Flow past a Flat Plate: Numerical Solution Using MATLAB." *International Journal of Computer Applications*, 2018, 180(18): 6-8. DOI: 10.5120/ijca2018916374. [Online] https://www.researchgate.net/publication/323218834_Boundary-Layer_Theory_of_Fluid_Flow_past_a_Flat-Plate_Numerical_Solution_using_MATLAB.
- PDF files with texts from books and articles recommended for reading:
    - Blasius_1908.pdf – The copy of [R9] Blasius, P. R. Heinrich. "Grenzschichten in Flüssigkeiten mit kleiner Reibung." *Zeitschrift für Mattematik und Physik*, 1908, 56(1): 1-37. (English translation is available: H. Blasius, "The Boundary Layers in Fluids with Little Friction," *NACA Technical Memorandum* 1256, Washington D.C., 1950. [Online] https://ntrs.nasa.gov/citations/20050028493).
    - Hager_2003.pdf – The copy of [R7] Hager, W.H., "Blasius: A life in research and education," *Experiments in Fluids*, 2003, 34: 566–571. DOI: 10.1007/s00348-002-0582-9. [Online] http://users.df.uba.ar/cobelli/Estructura_1/Blasius-Biography.pdf.
    - Schlichting - Boundary Layer Theory_7ed_1979.pdf – The copy of [1] Schlichting, H. *Boundary-layer theory*, 7th edition. New York: McGraw-Hill, 1979. [Online} http://ae.sharif.edu/~viscousflow/Schlichting - Boundary Layer Theory.pdf.
    - Van-Dyke_1982_Album-Fluid-Motion.pdf – The copy of [R10] Van Dyke, Milton. *An Album of Fluid Motion*. Stanford, CA: Parabolic Press, 1982, 176 p. [Online] http://courses.washington.edu/me431/handouts/Album-Fluid-Motion-Van-Dyke.pdf.

---

* **Dr. Vladimir V. Riabov**, Professor of Computer Science in the Department of Mathematics and Computer Science at Rivier University, teaches algorithms, networking technologies, computer security, software engineering, software quality assurance, object-oriented system design, system simulation and modeling, numerical methods, introduction to computing, and professional seminar in computer science. In 2010-2022, Vladimir served as the Department Coordinator and Director of Computer Science programs. He received a Ph.D. in Applied Mathematics and Physics from Moscow Institute of Physics and Technology and M.S. in Computer Information Systems from Southern New Hampshire University. Vladimir published about 130 articles in encyclopedias, handbooks, journals, and international and national conference proceedings, including *The Internet Encyclopedia*, *The Handbook of Information Security*, *The Handbook of Computer Networks*, *International Journal of Computers and Structures, Journal of Spacecraft and Rockets*, *Journal of Aircraft*, *Journal of Thermophysics and Heat Transfer*, *Congress Proceedings of International Council of the Aeronautical Sciences*, *International Symposia on Rarefied Gas Dynamics and Shock Waves*, *International Conferences on Computer Science and Information Systems*, *International Conferences on Technology in Collegiate Mathematics*, *Conferences of American Institute of Aeronautics and Astronautics*, and others. He is a senior member of ACM and AIAA, and a member of IEEE and MAA.