

Numerical solution of the Falkner Skan Equation by using shooting techniques

Dr. Summiya Parveen

Abstract: The aim of the paper is to examine the boundary value problems characterized by the well-known Falkner-Skan Equation. The Falkner-Skan Equation is governed by the third order non linear ordinary differential equation and then to solve numerically using the RungeKutta 4th order with shooting techniques and solve the Falkner Skan Equation for different parameters β and the numerical results are obtain by using Mat lab Software and compare the results of the literature [1],[2],[3].

Key words: Boundary layer, Blasius flow, Falkner Skan flow, RungeKutta method, Shooting Technique.

I. Introduction:

The Falkner-Skan equation arises in the study of laminar boundary layers flow exhibiting similarity solution. Assuming steady, incompressible, laminar flow with constant fluid properties and negligible viscous dissipation, the boundary layer equations can be reduced to [1]

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = U_e \frac{\partial U_e}{\partial x} + \nu \frac{\partial^2 u}{\partial y^2} \quad (1)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (2)$$

$$u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} = a \frac{\partial^2 T}{\partial y^2} \quad (3)$$

where $U_e(x)$ is the free stream velocity, u and v are the velocity components in x and y directions respectively and ν is the kinematic viscosity. Solution of these equations is simplified by the fact that for constant properties, conditions in the velocity boundary layer are independent of temperature and species concentration. Hence we may begin by solving the hydrodynamic problem (1) and (2) to the exclusion of Eq. (3). Once the hydrodynamic problem has been solved, solution to equation (3) can be obtained. In the particular case of the two-dimensional, incompressible boundary-layer flow over a wedge, when the free stream velocity is of the form $U_e(x) = Kx^m$, the governing partial differential equations can be converted to ordinary differential equation by employing the following similarity transformation:

$$u(x, y) = U_e(x) f(\eta) \quad , \quad \eta = \sqrt{\frac{(m+1)kx^{\frac{m-1}{2}}}{2\nu}}$$

This leads Eq.(1) and (2) to the well know Falkner Skan Equation.

$$\frac{\partial^3 f}{\partial \eta^3} + f \frac{\partial^2 f}{\partial \eta^2} + \beta \left[1 - \left(\frac{\partial f}{\partial \eta} \right)^2 \right] = 0$$

With the boundary conditions:

$$\begin{aligned} f(0) &= 0 \\ f'(0) &= 0 \end{aligned}$$

$$f'(\eta = \infty) = 1$$

The Falkner-Skan equation constitutes a third order, nonlinear two point boundary-value problem, no exact analytical solution is known. In the case of $\beta = 0$ the Falkner-Skan equation reduces to the well-known Blasius equation which is perhaps one of the most famous equations of fluid dynamics and represents the problem of an incompressible fluid that passes over a semi-infinite flat plate. In the case of accelerating flows ($\beta > 0$), the velocity profiles have no points of inflection, whereas in the case of decelerated flows [4],[5],[6] ($\beta < 0$). Physically relevant solutions exist only for $-0.19884 < \beta \leq 2$ [2].

In this paper, we write the Falkner Skan Equation as a first order differential system and obtain a numerical solution to the differential using 4th order Runge- kutta method by using a guess α and find out the solution of different values of parameters β .

II. Method of solution

The non-linear differential equations (1) subject to the boundary conditions (2) constitute a two-point boundary value problem. In order to solve these equations numerically, we follow Runge–Kutta 4th order with shooting technique. In this method it is most important to choose the appropriate finite values of $\eta \rightarrow \infty$. The solution process is repeated with another large value of $\eta \rightarrow \infty$ until two successive values of $f''(0)$ differ only after a desired digit signifying the limit of the boundary along η . The last value of $\eta \rightarrow \infty$ is chosen as appropriate value of the limit $\eta \rightarrow \infty$ for that particular set of parameters. The ordinary differential equation (1) was first converted into a set of three first-order simultaneous equations. To solve this system we require three initial conditions but we have only two initial conditions,

$f(0)$ and $f'(0)$ on $f(\eta)$. The initial condition $f''(0)$ is not prescribed. However the values of $f'(\eta)$ is known at $\eta = 0$. Now we employ the numerical shooting technique based where this ending boundary condition is utilized to produce unknown initial conditions at $\eta = 0$ finally, the problem has been solved numerically using Runge-Kutta 4th order.

Description of the Method:

Reduction to a first order system

To solve the falknerskan equation numerically, the equation is reduced to a first order system by introducing the three auxiliary variables.

$$f = u_1 \quad \frac{\partial f}{\partial \eta} = u_2 \quad \text{and} \quad \frac{\partial^2 f}{\partial \eta^2} = u_3,$$

So that we have the following system of three coupled ODE's:

$$f_1(\eta, u_1, u_2, u_3) = u_1' = u_2$$

$$f_2(\eta, u_1, u_2, u_3) = u_2' = u_3$$

$$f_3(\eta, u_1, u_2, u_3) = u_3' = -u_1 u_3 - \beta(1 - u_2^2)$$

The first order system can be written more compactly using vector notation.

$$\frac{\partial f}{\partial \eta} = f_1(\eta, u_1, u_2, u_3)$$

$$\text{i.e.} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} u_2 \\ u_3 \\ -u_1 u_3 - \beta(1 - u_2^2) \end{bmatrix}$$

it is important to note the ODE system is in normal form and then the boundary condition

$$\begin{aligned} u_1(0) &= 0 \\ u_2(0) &= 0 \\ u_2(\eta = \infty) &= 1 \end{aligned}$$

Where $\eta = \infty$ is the unknown free boundary used to truncate the semi-infinite interval to a finite one .which is to be determine as the part of the procedure .in addition ,an initial condition on the second derivatives is introduced to apply the Shooting Method [6],[7].

$$\frac{\partial^2 f}{\partial \eta^2} = \alpha; \quad \text{at} \quad \eta = 0$$

Where α is the shooting angle [8][9].

The shooting algorithm therefore consists of the following procedure:

- (i). Starting from a relatively large value of α as the initial guess $f'(\eta)$ is evaluated by increasing η through steps of h from zero to η_m .
- (ii). If at some η , $f'(\eta) > 1$, then α is decreased and $f'(\eta)$ evaluated until $f'(\eta) < 1$ for some α . At this point, the asymptotic profile is bracketed.

- (iii). A new α is then determined by Newton Method
- (iv). If $f'(\eta)$ does not cross unity from below as η increases from zero to ηm , then α is checked for negativity. If negative, α is below its correct value and Newton Method again determines the next α . Experience indicates that $20\eta m$ is sufficient for $\beta > 0$.
- (v). finally, when the estimate for α is approximately within an order of magnitude of the desired error, Runge- Kutta 4th order method can be used to the initial value problems[11],[12].

Multi-dimensional RungekuttaMethod :

Runge-Kutta Methods

This is the real deal. Runge-Kutta Methods are by far the most commonly use methods in most engineering applications today. They were developed around 100 years ago (relatively new in terms of math history – Newton was 17th century, Euler was early 18th century), and are an extension of the same math Euler developed. If you look at Euler's work, he seemed to love applying Taylor series to all sorts of different problems. In fact, that's how he came up with the method we just discussed – it's a first-order Taylor polynomial expansion, so its accuracy is limited, and if the derivatives of the function don't behave nicely, we can get some pretty serious error.

Runge-Kutta Methods include additional calculation of slope in the middle of each time step, and take a weighted average of the values to evaluate the function. This helps reduce our error as we go from time step to time step, and can result in some very accurate results. These methods are named by how many terms they use: an RK2 (2-term model) is very basic, and equivalent to the Midpoint Method, which we didn't bother to discuss since we're doing it here. An RK4 has four terms, and is generally the most accurate per computation time. FreeMat's (and MATLAB's) ODE45 routine switches between an RK4 and RK5 based on which is providing a better result, hence the name ODE45.

Solution of falknerSkan equation is computed by guessing value for $\alpha_1(0)=\alpha_{11}$ and the using equation to perform a a step by step Runge –Kutta for increasing η .The value of α_1 is systematically adjust until the condition $f' = 1$ for η large is realized . Compare the result the different values of β in the literature view[10],[11],[13],[14].

Comparison of the Values Ofand Different β For Different Authors

Value of η	Present Method	η_∞	Asalthambi[6]	Asalthambi[11]	Zhang&chef[6]
2	1.687218	4.73	1.687222	1.687218	1687218
1	0.332059	8.20	1.232589	1.232588	1232587
0.5	0.927680	5.80	0.927682	0.927680	0.927680
0	0.46900	5.00	0.469601	0.46900	0.46900
-0.1	0.31927	6.75	0.319270	0.319269	0.319270
-0.15	0.21631	6.99	0.216360	0.216358	0.216362
-0.18	0.128637	6.03	0.128637	0.128624	0.128636
-0.1988	0.008416	6.20	0.005218	0.005239	0.00522

III. Conclusion

A new iterative method is presented for the boundary value problems of a class of nonlinear third-order differential equation on semi-infinite intervals. Using the free boundary formulation, the original boundary value problem on an infinite interval is transformed to a free boundary problem on a finite interval, which is then solved by nonlinear shooting. Newton's method is used to find the "shooting angle" and the unknown free boundary. The initial values problems arise during shooting are computed by the Runge-Kutta method. The numerically by employing the fourth order Runge-Kutta method with shooting technique using Matlab software. The numerical result is being compared with the published work by prior researchers and is presented in Table 1-3 and also plotted in Fig. 1-2. Table 1-3.

In the end, we successfully compute several instances of the Falkner-Skan equation. In this paper, we have used an iterative technique to solve the Falkner-Skan problem. The results are in excellent agreement with those mentioned in the literature by using approximation and numerical methods. It will be a useful tool for researchers, who develop numerical techniques for Falkner-Skan equation as does not require any programming. It is a user friendly environment, appears to be a useful tool and does not require much disk space. The authors are working on including more solution methods in the program as well as making the program platform independent, perhaps by making it a web based application.

Our results are in excellent accordance with those already reported in literature.

Table 1: Numerical values of $f(\eta)$, $f'(\eta)$ and $f''(\eta)$ for $\beta = 0$

η	$f(\eta)$	$f'(\eta)$	$f''(\eta)$
0.0	0	0	0.4696
0.2	0.009391	0.093905	0.469306
0.4	0.037549	0.87605	0.467254
0.6	0.084386	0.280575	0.461734
0.8	0.149674	0.371963	0.451190
1.0	0.232990	0.460633	0.434379
1.2	0.333657	0.545246	0.410565
1.4	0.450724	0.624386	0.379692
1.6	0.582956	0.696700	0.342487
1.8	0.728872	0.761057	0.300445
2.0	0.886797	0.816695	0.255669
2.2	1.054947	0.863304	0.210580
2.4	1.231528	0.901065	0.167560
2.6	1.414824	0.930601	0.128613
2.8	1.603284	0.952875	0.095113
3.0	1.795568	0.969055	0.067710
3.2	1.990581	0.980365	0.046370
3.4	2.187467	0.987970	0.030535
3.6	2.385590	0.992888	0.019329
3.8	2.584499	0.995944	0.011759
4.0	2.783886	0.997770	0.006874
4.2	2.983555	0.998818	0.003861
4.4	3.183383	0.999397	0.002084
4.6	3.383296	0.999703	0.001081
4.8	3.583254	0.999859	0.000538
5.0	3.783235	0.999936	0.000258

Table 2: Numerical values of $f(\eta)$, $f'(\eta)$ and $f''(\eta)$ for $\beta = 1$

η	$f(\eta)$	$f'(\eta)$	$f''(\eta)$
0.0	0	0	0.332059
0.2	0.00664105	0.0664081	0.331985
0.4	0.02656	0.132765	0.331468
0.6	0.059735	0.198938	0.330081
0.8	0.106109	0.264711	0.327391
1.0	0.165573	0.329782	0.323009
1.2	0.23795	0.393778	0.31659
1.4	0.322983	0.456264	0.307867
1.6	0.420323	0.516759	0.296665
1.8	0.529521	0.574761	0.282932
2.0	0.650028	0.629769	0.266753
2.2	0.781197	0.681314	0.248352
2.4	0.922295	0.728985	0.228092
2.6	1.07251	0.772459	0.206455
2.8	1.23098	0.811513	0.184007
3.0	1.39682	0.846048	0.161359
3.2	1.5691	0.876085	0.139129
3.4	1.74696	0.901765	0.117876
3.6	1.92953	0.923334	0.0980867
3.8	2.11604	0.941122	0.0801258
4.0	2.30576	0.955522	0.0642341
4.2	2.49805	0.966961	0.0505193
4.4	2.69237	0.975875	0.0389731
4.6	2.88826	0.982687	0.0294829
4.8	3.08533	0.987793	0.0218711
5.0	3.28329	0.991546	0.0159068
5.2	3.48188	0.994249	0.0113414
5.4	3.68094	0.996159	0.00792786
5.6	3.88031	0.997481	0.00543169
5.8	4.0799	0.998379	0.00364835
6.0	4.27964	0.998976	0.00240198
6.2	4.47948	0.999366	0.00155022
6.4	4.67938	0.999615	0.000980419
6.6	4.87932	0.999711	0.000608191
6.8	5.07928	0.999867	0.000369599
7.0	5.27926	0.999925	0.000220207

7.2	5.47925	0.999959	0.000128431
7.4	5.67924	0.999979	0.0000737176
7.6	5.87924	0.999999	0.0000413441
7.8	6.07924	0.999996	0.0000227099
8.0	6.27924	1	0.0000122503
8.2	6.47924	1	6.4618×10^{-6}
8.4	6.67924	1	3.28575×10^{-6}

Table3: Numerical values of $f(\eta)$, $f'(\eta)$ and $f''(\eta)$ for $\beta = -0.1988$

η	f	f'	f''
0.0	0.000000	0.000000	0.008416
0.2	0.000433	0.005659	0.048175
0.4	0.002794	0.019268	0.087908
0.6	0.008670	0.040814	0.127512
0.8	0.019646	0.070249	0.166740
1.0	0.037288	0.107456	0.205139
1.2	0.063131	0.152203	0.241999
1.4	0.098646	0.204087	0.276324
1.6	0.145201	0.262479	0.306832
1.8	0.204011	0.326465	0.332012
2.0	0.276079	0.394820	0.350241
2.2	0.362128	0.465995	0.359983
2.4	0.462544	0.538165	0.360032
2.6	0.577326	0.609319	0.349788
2.8	0.706066	0.677407	0.329487
3.0	0.847956	0.740520	0.300318
3.2	1.001840	0.797079	0.264364
3.4	1.166280	0.845994	0.224361
3.6	1.339969	0.886753	0.183301
3.8	1.520440	0.919431	0.143987
4.0	1.706960	0.944613	0.108648
4.2	1.089785	0.963249	0.078699
4.4	2.091900	0.976488	0.054699
4.6	2.288816	0.985511	0.036472
4.8	2.485900	0.991411	0.023327
5.0	2.684590	0.995113	0.014311
5.2	2.883850	0.997340	0.008421
5.4	3.083460	0.998625	0.004753
5.6	3.283260	0.999337	0.002573
5.8	3.483170	0.999715	0.001335
6.0	3.683130	0.999907	0.000663
6.2	3.883120	1.000000	0.000314

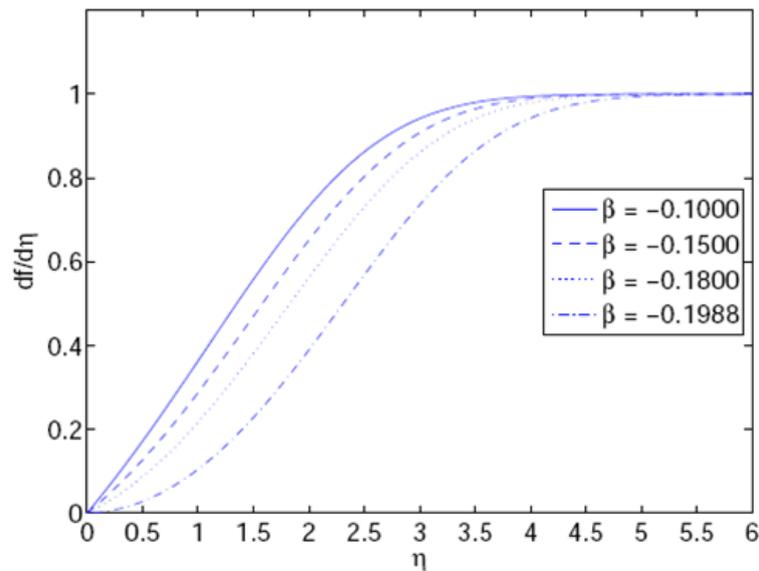


Figure 1: The velocity profile corresponding to different $\beta \in [-0.1988, 0)$.

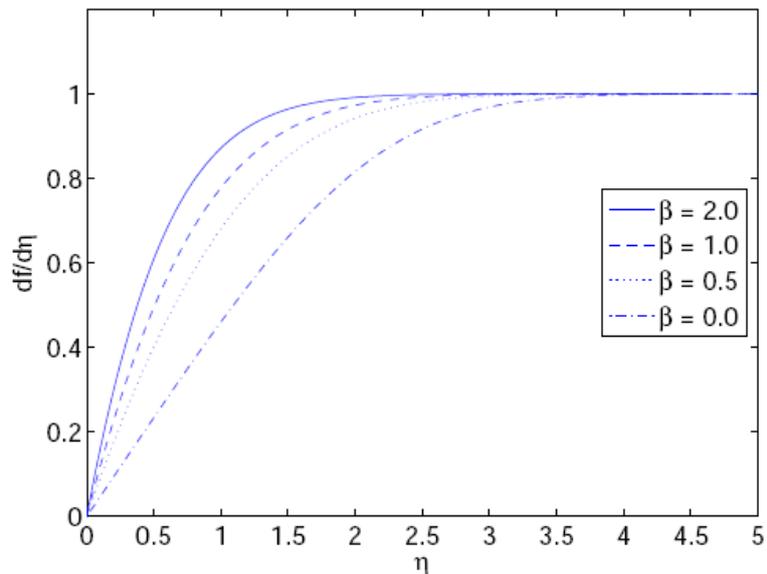


Figure 2: The velocity profile corresponding to different $\beta \in [0, 2]$

References:

- [1]. Incropera, F.P. and D.P. Dewitt, 2002. Fundamentals of Heat and Mass Transfer. Fifth Ed. John Wiley & Sons Inc., NY.
- [2]. Rosales-Vera, M. and A. Valencia, 2010. Solutions of Falkner-Skan equation with heat transfer by Fourier series. International Communications in Heat and Mass Transfer, 37: 761-765.
- [3]. Meksyn, D., 1961. New methods in laminar boundary-layer theory. Pergamon Press.
- [4]. Asaithambi, N.S., 1997. A numerical method for the solution of the Falkner-Skan equation. Appl. Math. Comput., 81: 259-264.
- [5]. Asaithambi, A., 1998. A finite-difference method for the solution of the Falkner-Skan equation. Appl. Math. Comput., 92: 135-141.
- [6]. Shi-Jun Liao, 1999. A uniformly valid analytic solution of two-dimensional viscous flow over a semi –infinite flat plate. J. Fluid Mech., 385: 101-128.
- [7]. Khabibrakhmanov, I.K. and D. Summers, 1998. The use of generalized Laguerre polynomials in spectral methods for nonlinear differential equations. Comput. Math. Appl., 36 (2): 65-70.
- [8]. Rosales, M. and A. Valencia, 2009. A note on solution of blasius equation by fourier series. Adv. Appl. Fluid Mech., 6 (1): 33-38.
- [9]. Lanczos, C., 1938. Trigonometric interpolatin of empirical and analytical functions. J. Math. Phys.,17: 123-129.
- [10]. A. Asaithambi, Numerical solution of the Falkner-Skan equation using piecewise linear function, Appl. Math. Comput. 159:267–273 (2004).
- [11]. N. S. Elgazery, Numerical solution for the Falkner-Skan equation, Chaos Solit. Fract. 35:738–746 (2008).
- [12]. R. Fazio, A novel approach to the numerical solution of boundary value problems on infinite intervals, SIAM J. Numer. Anal. 33:1473–1483 (1996).
- [13]. A. A. Salama, Higher-order method for solving free boundary-value problems, Numer. Heat Transfer, Part B: Fundamentals 45:385–394 (2004).
- [14]. D. F. Rogers, Laminar Flow Analysis, Cambridge University Press, Cambridge,1992.