

Boundary-Layer Theory of Fluid Flow past a Flat-Plate: Numerical Solution using MATLAB

Ehab Hussein Bani-Hani
Mechanical Engineering Department
School of Engineering
Australian College of Kuwait
Kuwait

Mamdouh El Haj Assad
SREE Department
University of Sharjah
United Arab Emirates

ABSTRACT

Fluid mechanics may have complicated systems where the analytical solution is tedious and time consuming. Changing one or more boundary conditions may add more challenges. Computer software provides easy and flexible solution to the fluid mechanics systems even when the boundary conditions are changing to describe the reality. In this work MATLAB code is used to solve the well-known third order ordinary differential equation that is Blasius equation. The results obtained are compared to other numerical and analytical solution available in the literature.

Results showed that with a simple code written using MATLAB the problem can be simulated and solved easily. A comparison between the solution obtained by MATLAB and the solutions published in literature showed a comparable results and same trends. Computers software allows getting very accurate results depending on the numerical method selected for the solution.

Keywords

MATLAB, Blasius, Fluid mechanics, numerical integration

1. INTRODUCTION

The laminar flow past a flat plate, which will be represented by the boundary-layer equations, can be derived by Navier stocks equation. This derivation leads to the well-known equation Blasius equation [1]. Figure 1 shows a stream of fluid flowing past a flat plate with upstream velocity u_∞ .

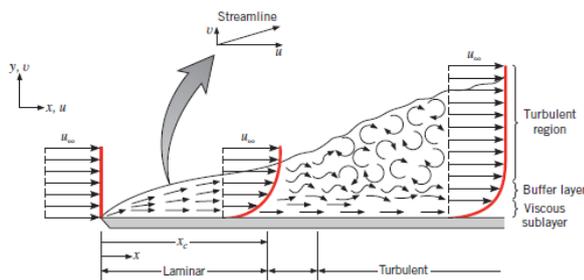


Fig 1: Velocity boundary layer development on a flat plate, [2]

The numerical solution published in the literature [1] is performed for non-dimensional parameters as follows:

$$\frac{u}{u_\infty} = f'(\eta) \quad (1)$$

Where η is defined as:

$$\eta = y \left(\frac{u_\infty}{\nu x} \right)^{\frac{1}{2}} \quad (2)$$

Rewriting Navier-stocks equation with manipulation and assumptions the standard form of Blasius equation can be expressed in Eq.(3) as follows:

$$f''' + \frac{1}{2} f f'' = 0 \quad (3)$$

With the following boundary conditions:

$$\text{At } y = 0: f(0) = f'(0) = 0$$

$$\text{At } y = \infty: f'(\infty) = 1.0$$

The accurate solution of Blasius equation has been obtained by numerical integration [1] and other numerical methods [3, 4]. However, this fluid problem derived by the Blasius equation has been solved by various analytical methods [5, 6].

Numerical solutions using computer software and algorithms are widely spread in engineering applications relevant to fluid mechanics [7-9]. MATLAB is also a useful tool in solving fluid mechanics applications such as piping network design and selection [10]. In this work, Blasius equation is solved numerically using MATLAB codes. Despite the Blasius Equation is solved by MATLAB using Runge Kutta method (11) and bisection method (12) this work shoes the solution using Euler integration method. The results obtained will be compared to the published results in the literature numerical and analytical ones.

2. MATLAB CODE

The MATLAB code used to solve Blasius equation is written using Euler method as follows:

Mat Lab code for Euler to solve Blasius Equation

```
function
E=Euler(ff,fy,fz,a,b,ffa,ya,za,m)
m=80;
a=0.0;
b=8;
h=(b-a)/m;
ya=0.0;
za=0.39; %changed by trial and error
until f(b)=1.0
ffa=0.0;
y(1)=ya;
z(1)=za;
ff(1)=ffa;
for j=1:m;
    eta(j)=a+(h*(j-1));
    ff(j+1)=ff(j)+(h*y(j));
    y(j+1)=y(j)+(h*z(j));
    z(j+1)=z(j)+(h*(-1*ff(j)*z(j)));
end
```

```

j=1:m;
eta=a:h:b;
ff(j)=ff(j)';
y(j)=y(j)';
z(j)=z(j)';
disp('      eta      f      fprim
fdoubprim')
ans=[eta' ff' y' z']
plot(eta,z);
    
```

3. RESULTS AND DISCUSSION

Figure 2 shows f' , f'' , and f''' respectively.

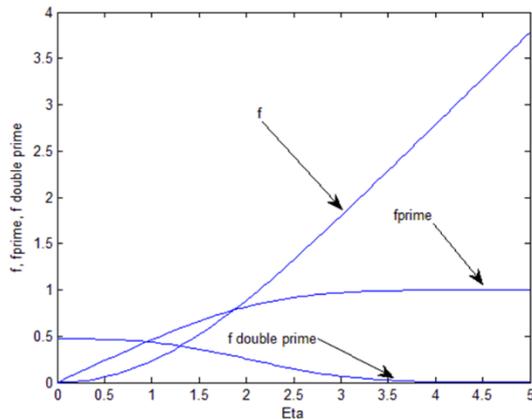


Fig 2: MATLAB solution using Euler method

It can be clearly seen in Fig. 2 that f' which is defined in Eq. (1) increases as the distance over the flat plate increases till it reached a value of 1.0 at the plate end. At this stage the velocity of the fluid is identical to the upstream velocity. This result is constant with all data published in the literature.

The values of f' obtained using MATLAB are comparable to the published data [1]. Figure 3 below shows the results of comparing the data obtained in this work and the data obtained by numerical integration. It is seen that the results are close to each other.

Finally the results of f as the final solution of Blasius equation are compared with a published data [13] in figure 4. Same trend is noticed, and again the results are comparable and close to each other at low values of η . However, there is constant deviation between the values at η values of more than 2. For example the MATLAB solution shows a value of 1.9545 at η value of 3.5, however, the published data shown in figure 4 show a value of 1.85 at the same value of η .

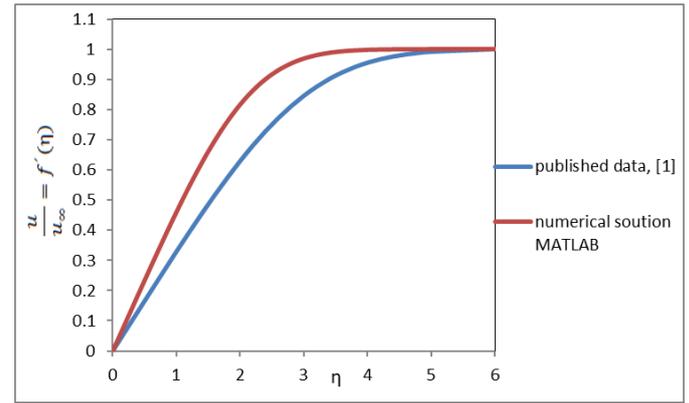


Fig 3: Results comparison between MATLAB work and results in the literature of f'

To check the results with another source of information, literature results [6] is chosen. The authors used analytical techniques to solve Blasius equation, their solution provides a value of 2.00 at same value of η . considering this value as the real one the published data [11] leads to about 7% error; however, MATLAB solution leads to only 2.00% error which provides an evidence of closer solution to analytical one. Using another numerical solution technique to solve third order differential equation and program it using MATLAB may lead to closer results. Even if there is a small deviations in the results, the ease of program using MATLAB is an advantage among other numerical and analytical methods of solution.

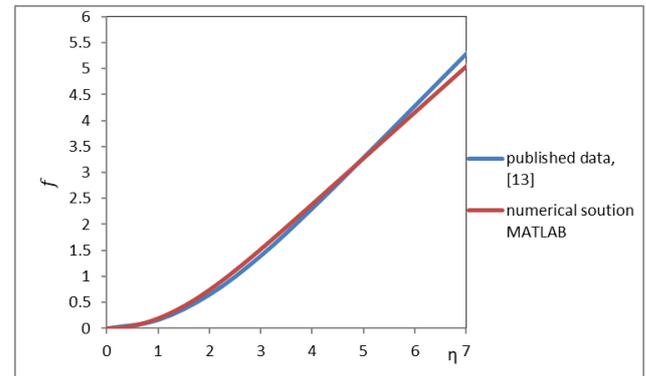


Fig 4: Results comparison between MATLAB work and results in the literature of f

4. CONCLUSION

MATLAB provides an effective tool to solve numerical integration. Some complicated systems in mechanical engineering such as fluid mechanics needs high techniques in mathematics to solve it accurately. These techniques are time consuming and tedious. When the boundary conditions are changed slightly to make the mathematical model close to reality the analytical and numerical solutions need to be repeated. This situation is solved easily by using soft wares such as MATLAB where these changes can be made by just changing numbers.

5. REFERENCES

- [1] Frank M. White., Fluid Mechanics, McGraw-Hill, Fourth Edition, pp. 435-437. 1999.
- [2] Theodore L. Bergman., Adrienne S. Lavine., Frank P. Incropera., David P. Dewit., Fundamentals of heat and mass transfer, John Wiley & Sons, seventh Edition, pp. 389-391, 2011.
- [3] S. Abbasbandy A numerical solution of Blasius equation by Adomian's decomposition method and comparison with homotopy perturbation method, Chaos, Solitons and Fractals, 31, 2007, pp. 257-260.
- [4] Yucheng Liu , Sree Navya Kurra. Solving Blasius Equation using HVIM. World Academy of Science, Engineering and Technology, 65, 2012.
- [5] L. Wang. A new algorithm for solving classical Blasius equation, Applied Mathematics and Computation, 157(1), 2004, pp. 1-9.
- [6] Jun Zheng et al., A globally convergent and closed analytical solution of the Blasius equation with beneficial applications, AIP ADVANCES 7, 065311 2017.
- [7] Tsan-Hsing Shih et al., A new $k-\epsilon$ eddy viscosity model for high reynolds number turbulent flows, Computers & Fluids, Volume 24, Issue 3, March 1995, pp. 227-238.
- [8] E.Robertson, V.Choudhury, S.Bhushan, D.K.Walters, Validation of Open FOAM numerical methods and turbulence models for incompressible bluff body flows, Computers & Fluids Volume 123, 21 December 2015, pp. 122-145.
- [9] Wahba, E. M., An Improved Computational Algorithm for Teaching Hydraulics of Branching Pipes in Engineering Curricula, Computer Applications in Engineering Education. 10, 2015.
- [10] Bani-Hani E. MATLAB application for the selection of the best pipe series/parallel arrangement in piping network. International journal of computer applications. 2017; 165:0975-8887.
- [11] Math works, Blasius equation solution, 1994-2018, accessed 3 January 2018, www.mathworks.com.
- [12] Jason Monschke, solving the Blasius boundary layer equation using Matlab 2018, accessed 3 January 2018, www.jasonmonschke.com.
- [13] Online courses, 2014-2018, accessed 24 December, 2017, www.nptel.ac.in/courses.