

# 基于Newton迭代动态调整的Runge-Kutta法的构造及其在Blasius方程中的应用

唐树江

湖南科技学院理学院, 湖南 永州

收稿日期: 2024年12月9日; 录用日期: 2025年1月2日; 发布日期: 2025年1月13日

## 摘要

本文提出了一种新型数值求解方法, 该方法将四阶Runge-Kutta法与Newton迭代法相结合, 旨在高效求解流体力学中的Blasius方程边值问题。首先, 我们将Blasius方程转化为一组一阶微分方程组, 并采用四阶Runge-Kutta法(RK4)进行数值求解。随后, 引入Newton迭代法动态调整初始条件, 以确保满足边界条件的要求。实验结果表明, 与传统的打靶法和结合打靶法的四阶Runge-Kutta法(SRK)进行对比实验, 新方法在迭代次数和计算时间上均展现显著优势, 同时求解精度也得到提升。

## 关键词

Blasius方程, Runge-Kutta法, Newton迭代法, 打靶法

# Construction of the Runge-Kutta Method Based on Newton's Iterative Dynamic Adjustment and Its Application to the Blasius Equation

Shujiang Tang

School of Science, Hunan University of Science and Engineering, Yongzhou Hunan

Received: Dec. 9<sup>th</sup>, 2024; accepted: Jan. 2<sup>nd</sup>, 2025; published: Jan. 13<sup>th</sup>, 2025

## Abstract

In this paper, we propose a novel numerical solution method that combines the fourth-order Runge-Kutta method with the Newton iterative method, aiming to efficiently solve the margin

problem of Blasius equation in fluid mechanics. Firstly, we transform the Blasius equation into a set of first-order differential equations and solve it numerically using the fourth-order Runge-Kutta method (RK4). Subsequently, the Newton iteration method is dynamically introduced to adjust the initial conditions to ensure that the boundary conditions are satisfied. The experimental results show that the new method exhibits significant advantages in terms of the number of iterations and computation time, as well as improved solution accuracy, in comparison experiments with the traditional shooting method and the Runge-Kutta method (SRK) combined with the improved shooting method.

## Keywords

Blasius Equation, Runge-Kutta Method, Newton's Iterative Method, Shooting Method

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

在流体力学领域，边界层现象对于深入理解流体在固体表面附近的流动特性具有至关重要的意义。作为描述无粘流体在平板上稳态边界层流动的核心方程，Blasius 方程[1]在该领域占据了不可替代的重要地位。尽管 Blasius 方程的解已广为人知，但在实际工程应用中，复杂的边界条件和多变的流体特性常常使得数值求解方法成为更为实用且必要的选择。

传统的数值求解方法，如打靶法、有限差分法、有限元法和谱方法等，各自具有独特的优势和局限性。打靶法结构简单，但强烈依赖于初始猜测解。有限差分法虽然实现简单且易于理解，但在处理非线性方程和高阶边界条件时，常常面临收敛速率慢和稳定性不足的挑战。有限元法则以其强大的解的逼近能力而著称，但其实现过程相对复杂，且对参数调整的要求较高。谱方法虽然在某些特定情况下能提供高效的精度，但其高度依赖于高频率的基函数，这在一定程度上限制了其适用范围。

为了克服这些传统方法的局限性，研究者们一直在不断探索新的数值求解策略。在打靶法的发展过程中，该方法最初被广泛应用于解决各种边值问题，通过不断调整初始条件来逼近满足边界条件的解[2]。随着计算技术的不断进步，打靶法也逐渐得到了改进和优化。多重打靶法(Multiple Shooting Method)作为打靶法的一种重要变体，在 1984 年由 Bock 和 Plitt [3]首次提出并应用于优化控制问题的求解。该方法通过将整个求解区间划分为多个子区间，并在每个子区间上分别应用打靶法，从而显著提高了计算效率和收敛性。此后，多重打靶法在数值求解领域得到了广泛应用，并不断发展和完善[4]-[6]。

与此同时，Runge-Kutta 法[7]作为一种常用的求解常微分方程的数值方法，也在不断改进和优化中。该方法基于离散化的思想，通过逐步逼近的方式计算方程的数值解，具备收敛速度快、精度高等优点，尤其适用于求解非线性方程和高阶方程[8]。近年来，Sadasivuni 等人[9]提出了一种创新的数值求解方法，将 Runge-Kutta 法与打靶法相结合，利用打靶法确定边值问题的初始条件，并借助 Runge-Kutta 法求解相应的常微分方程，实现了更快的收敛性与更高的求解精度。

尽管这些方法取得了显著成果，但在调整初始条件时仍需进行多次迭代，并且 Runge-Kutta 法在处理某些复杂问题时仍存在一定的局限性。为此，本研究提出了一种结合四阶 Runge-Kutta 法与 Newton 迭代法的新型数值求解方法，旨在高效求解 Blasius 方程的边值问题。这种方法不仅提高了收敛效率，还确保了求解精度，对工程应用和科学计算具有重要的实用价值。

## 2. 问题描述

Blasius 方程是流体力学中描述无粘性流体在平板上流动的边界层方程，其形式如下：

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

其中， $f(\eta)$  表示流函数， $\eta$  是垂直于平板的坐标。该方程的边界条件为：

$$f(0) = 0, f'(0) = 0, f'(\infty) = 1. \quad (2)$$

这些边界条件具有明确的物理意义：

$f(0) = 0$  表示在平板表面流体的流速为零，这是由于流体与固体表面之间的粘性作用导致的流体粘附现象。

$f'(0) = 0$  表示在平板表面流体的流函数为零，这是无滑移条件的直接体现。

$f'(\infty) = 1$  表示在边界层外，流体的流速梯度趋向于自由流速，即远离固体表面的区域，流体的流动不再受固体表面的影响。

尽管 Blasius 方程的近似解已为人所熟知，但在实际工程应用中，由于边界条件的复杂性和流体特性的多变性，数值求解方法成为更为实用且必要的选择。本研究的目标是通过数值方法求解 Blasius 方程，以为流体力学领域的研究和工程应用提供精确的数值解。

## 3. 方法

本节将详细介绍用于求解 Blasius 方程的结合打靶法的四阶 Runge-Kutta 法及结合 Newton 迭代的新方法 Runge-Kutta 法。

### 3.1. 结合打靶法的四阶 Runge-Kutta 法

四阶 Runge-Kutta 法(RK4)是一种广泛用于求解常微分方程初值问题的数值方法。该方法通过在每个步长内计算多个中间斜率来近似下一个点的值，具有较高的精度和稳定性，特别适合处理 Blasius 方程这类非线性问题。

#### 3.1.1. 方法步骤

考虑微分方程的通用形式：

$$y' = f(x, y). \quad (3)$$

四阶 Runge-Kutta 法的步骤如下：

- 初始化：给定初始条件  $x_0$  和  $y_0$ ，设定步长  $h$ 。
- 迭代计算：对于每一步计算，使用以下公式：

$$\begin{aligned} k_1 &= f(x_n, y_n), \\ k_2 &= f\left(x_n + \frac{h}{2}, y_n + \frac{hk_1}{2}\right), \\ k_3 &= f\left(x_n + \frac{h}{2}, y_n + \frac{hk_2}{2}\right), \\ k_4 &= f(x_n + h, y_n + hk_3), \\ y_{n+1} &= y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4). \end{aligned} \quad (4)$$

### 3.1.2. 应用于 Blasius 方程

为了将 RK4 方法应用于 Blasius 方程，我们首先将(1)转化为一组一阶微分方程。设定：

$$\begin{aligned} y_1 &= f, \\ y_2 &= f', \\ y_3 &= f''. \end{aligned} \quad (5)$$

则 Blasius 方程可重写为：

$$\begin{aligned} \frac{dy_1}{d\eta} &= y_2, \\ \frac{dy_2}{d\eta} &= y_3, \\ \frac{dy_3}{d\eta} &= -\frac{1}{2}y_1y_3, \end{aligned} \quad (6)$$

在求解过程中，我们设定初始条件  $y_1(0)=0$ ， $y_2(0)=0$ ，而  $y_3(0)=f''(0)$  为未定参数。为了满足  $y_2'(\infty)=1$ ，Raghu Karthik Sadasivuni 等人通过采用打靶法进行动态调整。通过迭代的方式调整初始条件  $f''(0)$ ，使最终解逐步接近目标值。具体步骤如下：

- 设定初始条件  $f''(0)$  的初始猜测(例如 1.0)。
- 通过 RK4 方法计算  $f'(\eta)$  的值。
- 计算  $f'(\infty)$ (实际上是  $f'(\eta_{end})$  的近似值)。
- 根据计算结果与目标值之间的差异，利用二分法动态调整初始条件，重复此过程，直到结果满足预设精度。

为方便起见，我们将这种使用打靶法动态调整的 Runge-Kutta 简记为 SRK。

## 3.2. Newton 法动态调整的四阶 Runge-Kutta 法

在求解 Blasius 方程的过程中，满足边界条件  $f'(\infty)=1$  是一个关键挑战。为了克服这一难题，我们引入了 Newton 法来动态调整初始条件  $f''(0)$ ，以确保最终解能够满足边界条件。

### 3.2.1. 方法步骤

Newton 法[10]，也称为牛顿-拉弗森方法(Newton-Raphson method)，是一种迭代求解方程根的数值方法。该方法通过线性化非线性方程的根来寻找方程的近似解。对于给定的非线性方程  $f(x)=0$ ，Newton 法的迭代公式为：

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad (7)$$

其中， $f(x_n)$  是目标函数在  $x_n$  处的值， $f'(x_n)$  是目标函数在  $x_n$  处的导数。

### 3.2.2. 目标函数与导数的计算

在本文中，目标函数为：

$$g(f''(0)) = f'(\infty) - 1. \quad (8)$$

我们的目标是通过调整初始条件  $f''(0)$  使得  $g(f''(0))=0$ ，即  $f'(\infty)=1$ 。

为了实施 Newton 迭代，我们需要计算目标函数的导数  $g'(f''(0))$ 。在实际应用中，直接计算导数可能较为复杂，因此我们采用中心差分法来近似导数：

$$g'(f''(0)) \approx \frac{g'(f''(0)+\Delta) - g'(f''(0)-\Delta)}{2\Delta}, \quad (9)$$

其中  $\Delta$  为一个小的增量，以确保差分近似的准确性。本文中，我们取  $\Delta = 10^{-6}$ 。

### 3.2.3. 迭代更新与收敛条件

根据 Newton 法的更新公式，我们迭代更新初始条件：

$$f''(0)_{new} = f''(0)_{old} - \frac{g(f''(0)_{old})}{g'(f''(0)_{old})}. \quad (10)$$

迭代过程持续进行，直到满足预设的收敛条件。收敛条件通常设置为：

$$|g(f''(0))| < \varepsilon,$$

其中  $\varepsilon$  为预设的容忍度，用于控制迭代的精度。

通过将四阶 Runge-Kutta 法与 Newton 法结合，我们可以在运行过程中有效调整初始条件，迅速满足边界条件，从而提高求解的准确性和效率。初始阶段利用 Runge-Kutta 法获取近似解，在后续步骤中通过 Newton 法进行动态微调，以求得更精确的结果。为方便，我们将这种使用 Newton 法结合的四阶 Runge-Kutta 法简记为 NRK。

## 4. 数值实验

在本节中，我们将通过使用改进的 Runge-Kutta 法结合 Newton 迭代法求解 Blasius 方程边值问题，并与传统打靶法及 SRK 方法进行对比。

### 4.1. 数值实验设置

我们设定的 Blasius 方程如下：

$$f''' + \frac{1}{2}ff'' = 0,$$

定义边界条件为：

$$f(0) = 0, f'(0) = 0, f'(\infty) = 1.$$

在数值实验中，设定结束的  $\eta_{end} = 10$ ，步长  $h$  选择为 0.1，初始猜测  $f''(0) = 1$ 。预设的容忍度  $\varepsilon = 10^{-9}$ ，最大迭代次数为  $iter_{max} = 1000$ 。我们分别使用四阶 Runge-Kutta 法结合 Newton 法和传统的打靶法进行求解。

### 4.2. 运行结果

本节将展示采用改进的 Runge-Kutta 法结合 Newton 迭代法求解 Blasius 方程边值问题的实验结果。我们将对比分析新方法与传统打靶法及结合打靶法的 Runge-Kutta 法(SRK)的性能差异。具体来说，我们将关注迭代次数、计算时间以及在边界条件  $f''(0)$  处的计算精度。这些指标将帮助我们评估新方法在收敛速度和求解精度上的优势。

表 1 为三种格式在同一条件的情况下的迭代次数、计算时间和  $f''(0)$  的计算结果。从表中可以看出，打靶法在收敛至预期值时需要大量的迭代，而 SRK 方法仅需要 20 次迭代即可达到高精度的解，相比而言，本文的 NRK 法迭代次数更少，显著减少了计算时间。而从最终的  $f''(0)$  的结果可以看出，两种 Runge-Kutta 都取得了与经典结论 0.33206 一致的结果，而打靶法却没有达到。

**Table 1.** Comparison of iteration times, computation time, and calculation results of  $f''(0)$ **表 1.** 迭代次数、计算时间及  $f''(0)$  的计算结果对比

方法	迭代次数	计算时间(s)	$f''(0)$
打靶法	27,001	51.9142	0.3308222675879946562
SRK	20	0.0359	0.3320573462609329729
NRK	2	0.0309	0.3320573459459646437

考虑到打靶法需要两万多步才能达到要求，我们并没有将结果呈现在文章之中。表 2 为两种 Runge-Kutta 法的迭代误差，对比可以清晰地看到两种改进的 Runge-Kutta 方法在收敛速度和精度上的差异：

**Table 2.** Comparison of iteration step errors**表 2.** 迭代步误差对比

迭代步	SRK 法	NRK 法
1	1.08540902e+00	1.08540902e+00
2	2.77606469e-01	2.22044605e-16
3	8.50898571e-02	
4	2.75948094e-02	
5	9.11493542e-03	
6	3.02912691e-03	
7	1.00869116e-03	
8	3.36117394e-04	
9	1.12026579e-04	
10	3.73407980e-05	
11	1.24467776e-05	
12	4.14890857e-06	
13	1.38296759e-06	
14	4.60988977e-07	
15	1.53662966e-07	
16	5.12209852e-08	
17	1.70736609e-08	
18	5.69122016e-09	
19	1.89707317e-09	
20	6.32358166e-10	

NRK 法的误差在第二次迭代就已经达到了足够的要求, 而 SRK 则需要二十步, 且 NRK 的误差要远小于 SRK。说明 Runge-Kutta 法与 Newton 迭代法的结合在求解 Blasius 方程时提供了显著的收敛速度优势。

## 5. 结论

本文成功地开发并验证了一种结合四阶 Runge-Kutta 法和 Newton 迭代法的新型数值求解方法, 用于求解流体力学中的关键问题——Blasius 方程的边值问题。通过将 Blasius 方程转化为一阶微分方程组, 并应用改进的 Runge-Kutta 法, 结合 Newton 迭代法动态调整初始条件, 本研究显著提高了求解效率和精度。

实验结果表明, 与传统的打靶法和结合打靶法的 Runge-Kutta 法(SRK)相比, 新方法在迭代次数和计算时间上均有显著减少, 同时在求解精度上也更为准确。具体来说, 新方法(NRK)在仅两次迭代后即可达到与经典解析解 0.33206 一致的结果, 而打靶法需要两万多次迭代, SRK 方法也需要二十次迭代。此外, NRK 法的迭代误差在第二次迭代后已降至足够小, 远低于 SRK 法。

本研究的结论强调了结合 Runge-Kutta 法和 Newton 迭代法在处理边值问题时的优越性。该方法不仅提高了收敛速度, 还确保了求解的高精度, 这对于工程应用和科学计算中的流体力学问题具有重要的实用价值。未来的工作可以进一步探索该方法在其他类型的边界层方程和更广泛的流体力学问题中的应用, 以及进一步优化算法以适应更复杂的流动条件。

## 参考文献

- [1] Blasius, H. (1908) Grenzsichten in Flüssigkeiten mit kleiner Reibung. *Zeitschrift für Mathematik und Physik*, **53**, 1-37.
- [2] Keller, H.B. (1968) Numerical Solution of Two-Point Boundary Value Problems. *SIAM Journal on Numerical Analysis*, **5**, 612-625.
- [3] Bock, H.G. and Plitt, K.J. (1984) A Multiple Shooting Algorithm for Direct Solution of Optimal Control Problems. *IFAC Proceedings Volumes*, **17**, 1603-1608. [https://doi.org/10.1016/s1474-6670\(17\)61205-9](https://doi.org/10.1016/s1474-6670(17)61205-9)
- [4] Ascher, U.M. and Petzold, L.R. (1998) Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations. Society for Industrial and Applied Mathematics. <https://doi.org/10.1137/1.9781611971392>
- [5] Lambert, J.D. (1973) Numerical Methods for Ordinary Differential Systems: The Initial Value Problem. John Wiley & Sons.
- [6] Cebeci, T. and Cousteix, J. (2005) Modeling and Computation in Boundary Layer Aerodynamics. Horizons Publishing.
- [7] Butcher, J.C. (2008) Numerical Methods for Ordinary Differential Equations. John Wiley & Sons. <https://doi.org/10.1002/9780470753767>
- [8] Hairer, E., Norsett, S.P. and Wanner, G. (1993) Solving Ordinary Differential Equations I: Non Stiff Problems. Springer Series in Computational Mathematics.
- [9] Sadasivuni, R.K. (2024) Solving Blasius Equation Using RK-4 Numerical Method. <https://www.mathworks.com/matlabcentral/fileexchange/102189-solving-blasius-equation-using-rk-4-numerical-method>
- [10] Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (2007) Numerical Recipes: The Art of Scientific Computing. 3rd Edition, Cambridge University Press.