A comparison of Multi-Step and Multi-stage Methods

Yonghyeon Jeon
Department of Mathematics,
Kyungpook National University,
Daegu, 41566, Republic of Korea
Email: dydgus1020@knu.ac.kr

Sunyoung Bu
Department of Liberal arts,
Hongik University,
Sejong, 30016, Republic of Korea
Email: syboo@hongik.ac.kr

Soyoon Bak
Department of Mathematics,
Kyungpook National University,
Daegu, 41566, Republic of Korea
Email: jiya525@knu.ac.kr

***Abstract*—We study a multi-stage method compared with a multi-step method for solving a stiff initial value problem. Due to expensive computational costs of the multi-stage methods for solving a massive linear system induced from the linearization of a highly stiff system, stiff problems are usually solved by the multi-step method, rather than the multi-stage method. In this work, we investigate properties of both the multi-step and the multi-stage methods and discuss the difference between the two methods through numerical tests in several examples. Furthermore, the advantages of multi-stage methods can be heuristically proved even for stiff systems by the comparison of two methods with several numerical tests.**

I. Introduction

It is well known that every time dependant problem can be solved either a multi-stage method or multi-step method. In general, there is no significant difference of the structure between them when multi-stage method is applied to initiate the multi-step method [14]. The comparison of the methods is quite interested in the practical computations and efficiency of methods for each problem. Multi-stage methods such as Runge-Kutta type method do not require additional memory for function values at previous steps, since it does not use any previously computed values. Compared with multi-stage methods, multi-step methods require additional memory in the sense that they use previous computed function values, so that they have insufficient function values for an initial data. Another big difference comes from the size of the system to compute each time step. The size of system to solve for the multi-stage method is $d \times s$ for each time step, while multi-step methods need to solve only a system of size d, where d and s represent the dimension and number of stages of the system, respectively. This difference of size gives a critical issue of the choice of the methods for the problems we concern.

In particular, the question for more efficient method is quite susceptible to the problems which are stiff and non-linear. For non-linear stiff problems, multi-step methods are traditionally recommended to apply, since function evaluations are computationally expensive, which is required to evaluate only once at each time step, but several times for multi-stage methods [1], [5]. A lot of research has been to develop a numerical solver for big size of system such as eigenvalue decomposition and LU decomposition, etc [4], [5], [10], [13], with the development of computer, so that multi-stage methods are comparable with multi-step methods for non-linear stiff problems. In addition, multi-stage methods have no restriction to express an initial data contrast to the other. Nowadays there is not such a clear a-priori distinction between multi-stage and multi-step methods. In this study, we compare the two methods, multi-step and multi-stage methods, by investigating properties of both method and examine advantages of the each method suitable for solving stiff systems.

II. Background

We describe a formula of multi-step and multi-stage method. As before, let $\phi'(t) = f(t, \phi(t))$ and $h > 0$ and define the nodes by $t_n = t_0 + nh$, $n \geq 0$. The general form of a multi-step methods is given by

$$\phi_{n+1} = \sum_{j=0}^{p} a_j \phi_{n-j} + h \sum_{j=-1}^{p} b_j f(t_{n-j}, \phi_{n-j}) \quad n \geq p \quad (1)$$

The coefficients $a_0, \cdots, a_p, b_{-1}, b_0, \cdots, b_p$ are constants, and $p \geq 0$. If either $a_p \neq 0$ or $b_p \neq 0$, the method is called a $p+1$ step method. The general form of a multi-stage method(RK) is given by

$$\phi_{n+1} = \phi_n + h \sum_{i=1}^{s} b_i k_i, \quad (2)$$

where

$$k_i = f\left(t + c_i h, \phi_n + h \sum_{j=1}^{s} a_{ij} k_i\right). \quad (3)$$

To comparison two methods for stiff problems, we take one of implicit Rung-Kutta method(IRK) of order 3 and backward difference formula(BDF) of order 3 as the multi-stage and the multi-step method, respectively. There is a formula of BDF3.

$$y_{n+3} - \frac{18}{11} y_{n+2} + \frac{9}{11} y_{n+1} - \frac{2}{11} y_n = \frac{6}{11} h f(t_{n+3}, y_{n+3}) \quad (4)$$

And there is a butcher tableau for IRK3 instead of form (2)

TABLE I
THE ORDER CHECK FOR RK3 AND BDF3 WITH $\nu = -1$

	RK3		BDF3	
h	Err(h)	rate	Err(h)	rate
2^{-1}	9.8926e-5	-	0.0219	-
2^{-2}	1.6149e-5	2.6149	0.0029	2.9189
2^{-3}	2.2800e-6	2.8243	3.6735e-4	2.9808
2^{-4}	3.0244e-7	2.9143	4.6110e-5	2.9940
2^{-5}	3.8928e-8	2.9577	5.7679e-6	2.9990

$$\begin{array}{c|ccc} \frac{1}{4} & \frac{53}{144} & -\frac{11}{48} & \frac{1}{9} \\ \frac{3}{4} & \frac{9}{16} & \frac{3}{16} & 0 \\ 1 & \frac{5}{9} & \frac{1}{3} & \frac{1}{9} \\ \hline & \frac{5}{9} & \frac{1}{3} & \frac{1}{9} \end{array}$$

Additionally, considering stability, RK3 is L-stable and BDF3 is $A(\alpha)$-stable.

III. ADVANTAGE OF MULTI-STAGE METHOD

For comparison of numerical results, we introduce the following notations:

$$Err(h) = \max_{1 \le i \le n} ||y(t_i) - y_i||_\infty, \tag{5}$$

$$rate = \frac{log(Err(h_1)/Err(h_2))}{log(h_1/h_2)}, \tag{6}$$

where $\|\cdot\|_\infty$ is the maximum norm. We test Prothero-Robinson Equation [1], which is a well known problem for testing for ODE solvers,

$$\phi'(t) = \nu(\phi(t) - g(t)) + g'(t), \quad t \in (0, 10]; \quad \phi(0) = 0, \tag{7}$$

where the eigenvalue ν is a parameter to present stiffness of (7) and $g(t) = \sin(t)$. We examine both non-stiff and stiff cases. The parameter ν is setted to -1 and -10^6 for non-stiff and stiff, respectively. The exact solution of this problem is $\phi(t) = \sin(t)$. Note that the exact solution does not have stiffness, but the magnitude of the eigenvalue ν causes the stiffness of the problem. The first aim of numerical test is to check a convergence order. There are two tables. One is a result of non-stiff case as shown in Table.I. The other is a result of stiff case, Table.II. In Table.I and Table.II, we display $Err(h)$ and $rate$ for the step size $h = 2^{-n}, n = 1, 2, 3, 4, 5$. It can be seen that two methods have numerically convergence order 3.

The second aim is to observe the accuracy. For the test, we use a fixed step size $h = 0.5$. Also, the accuracy test is proceed for non-stiff and stiff cases. For comparison of the accuracy, we measure the absolute error at each integration step. As seen in the Fig.1 and Fig.2, the accuracy of multi-stage method(RK3) is much better than multi-step method(BDF3). In non-stiff case, absolute error of BDF3 has a magnitude $1.0e{-}2$ on average. But absolute error of RK3 has a magnitude

TABLE II
THE ORDER CHECK FOR RK3 AND BDF3 WITH $\nu = -10^6$

	RK3		BDF3	
h	Err(h)	rate	Err(h)	rate
2^{-1}	9.7977e-10	-	3.0343e-8	-
2^{-2}	1.2183e-10	2.9957	3.8784e-9	2.9679
2^{-3}	1.5254e-11	2.9976	4.8741e-10	2.9922
2^{-4}	1.9069e-12	2.9999	6.1006e-11	2.9981
2^{-5}	2.3836e-13	3.0000	7.6286e-12	2.9995

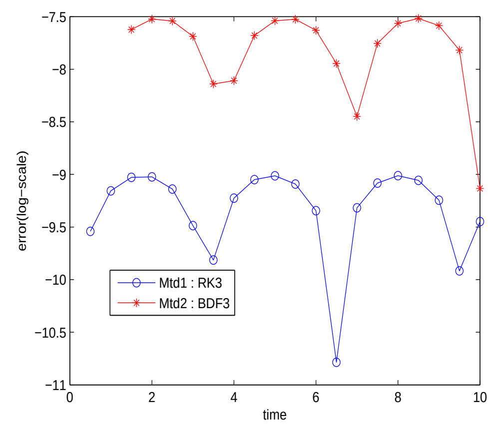

Fig. 1. Absolute error with stiffness $\lambda = -10^6$ and with having the same time step size

$1.0e-4$ on average. In stiff case, absolute error of BDF3 has a magnitude $1.0e-7$ on average. On the other hand absolute error of RK3 has a magnitude $1.0e-9$ on average.

Finally, we test more complex Prothero-Robinson Equation given by

$$\begin{bmatrix} \phi_1'(t) \\ \phi_2'(t) \end{bmatrix} = \begin{bmatrix} -\lambda_1\big(\phi_1(t) - \sin(t)\big) + \cos(t) \\ -\lambda_2\big(\phi_2(t) - \cos(t)\big) - \sin(t) \end{bmatrix} \qquad t \in [0, 10] \tag{8}$$

where $\lambda_1 = 1.0e + 6$, $\lambda_2 = 1$, exact solution is $[\phi_1(t),\ \phi_2(t)]^T = [\sin(t),\ \cos(t)]^T$.
We try comparison two methods on different condition. The step size $\tilde{h}$ for multi-stage method is changed to $\tilde{h} = h/3$. The reason why the condition taken is shown in Fig.2. One can say that one step in multi-step method and one stage in multi-stage method have similar meaning. For the comparison, we measure the absolute error at each integration step. From the combined result of Fig.3 and Fig.4, we can know that multi-stage method is superior to multi-step method. Furthermore, we can see that multi-stage method is better from the result of relation between absolute error and costs(the number of function evaluations)

IV. CONCLUSION

Throughout several preliminary tests, we find a multi-stage method is much better than multi-step method in terms of

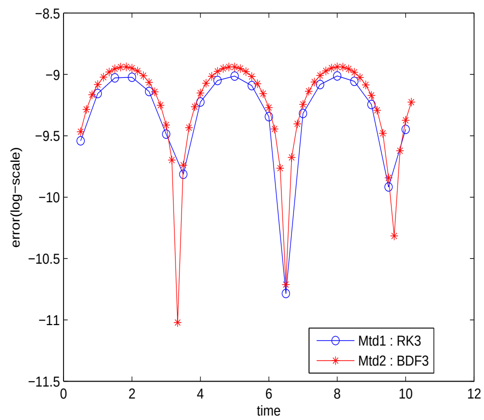

Fig. 2. Absolute error with stiffness $\lambda = -10^6$ for (7). The step size of RK3 is 3 times larger than that of BDF3

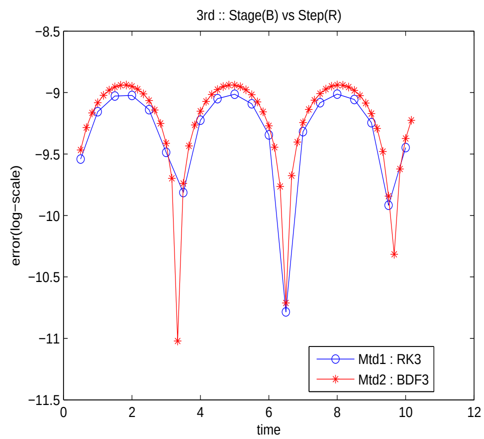

Fig. 3. Absolute error of 1st component with stiffness $\lambda = -10^6$ for (8). The step size of RK3 is 3 times larger than that of BDF3

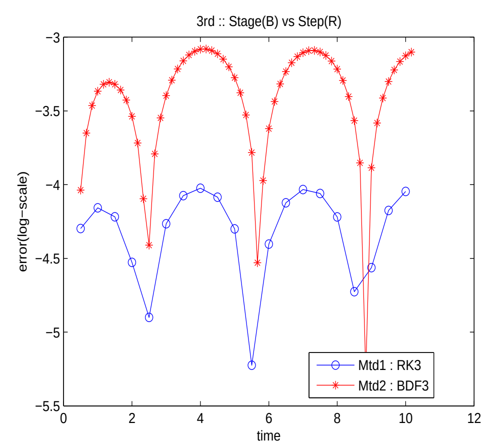

Fig. 4. Absolute error of 2nd component with stiffness $\lambda = -10^6$ for (8). The step size of RK3 is 3 times larger than that of BDF3

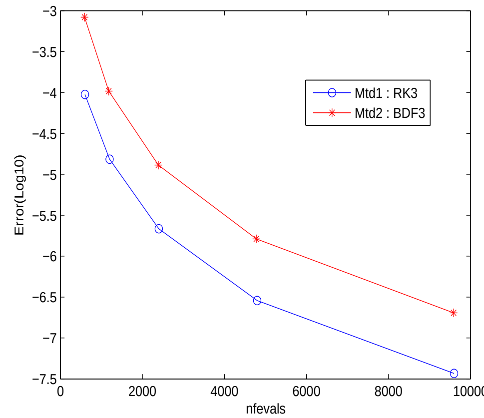

Fig. 5. Absolute error versus the number of function evaluation with stiffness $\lambda = -10^6$ for 8. The step size of RK3 is 3 times larger than that of BDF3

the error behavior. In the further research, we would like to analyze these phenomena in terms of a concrete mathematical tool.

Acknowledgements

This research was supported by the basic science research program through the National Research Foundation of Korea (NRF), and funded by the Ministry of Education, Science and Technology (grant number 2016R1A2B2011326 and 2016R1D1A1B03930734).

References

[1] A. Prothero, A. Robinson, *On the stability and accuracy of one step methods for solving stiff systems of ordinary differential equations* , Math. Comput. 28 (125) (1974) 145-162

[2] C. Fredebeul, *A-BDF: a generalization of the backward differentiation formulae*, SIAM J. Numer. Anal. 35 (1998) 1917–1938.

[3] C.W. Gear, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice–Hall, 1971.

[4] E. Hairer, S.P. Nørsett, G. Wanner, *Solving Ordinary Differential Equations. I Nontstiff Problems*, Springer Ser. Comput. Math., Springer, 1993.

[5] E. Hairer, G. Wanner, *Solving Ordinary Differential Equations. II Stiff and Differential-Algebraic Problems*, Springer Ser. Comput. Math., Sprigner, 1996

[6] G.J. Cooper, R. Vignesvaran, *Some schemes for the implementation of implicit Runge-Kutta methods*, J. Comput. Appl. Math. 45 (1993) 213-225.

[7] J.C. Butcher, *Implicit Runge-Kutta processes*, Math. Comput. 18 (1964) 50-64

[8] J.C. Butcher, G. Wanner, *Runge-Kutta methods:some historical notes*, Appl. Numer. Math. 22 (1996) 133-151.

[9] J.R. Cash, S. Considine, *An MEBDF code for stiff initial value problems*, ACM Trans. Math. Softw. 18 (1992) 142-155.

[10] K.E. Atkinson, *An Intoroduction to Numerical Analysis*, John Wiley and Sons, Inc., 1989.

[11] K. Wright, *Chebyshev collocation methods for ordinary differential equations*, Comput. J. 6 (1963) 358–363.

[12] S.D. Kim, J. Kwon, X. Piao, P. Kim, *A Chebyshev collocation method for stiff initial value problems and its stability*, Kyungpook Math. J. 51(4) (2011) 435–456.

[13] S. Gilbert, *Introduction to Linear Algebra. 5th ed.*, Wellesley-Cambridge Press

[14] U. Kirchgraber, *Multi-step methods are essentially one-step methods*, Numer. Math. 48 (1986) 85-90.

[15] Z. Wang, B. Guo, *Legendre–Gauss–Radau collocation method for solving initial value problems of first order ordinary differential equations*, J. Sci. Comput. 52(1) (2012) 226–255.