CAS 741: Problem Statement
Radius of Convergence

John Ernsthausen (macid: ernsthjm)

Table 1: Revision History

Date	**Developer(s)**	**Change**
September 18, 2020	John Ernsthausen	Initial draft

After generating a real-valued Taylor series (TS) approximate solution of order p to the ordinary differential equation (ODE) initial-value problem (IVP)

$$y'(t) = f\big(t, y(t)\big), \quad y(t_0) = y_0 \in \mathcal{D} \subset \mathbb{R}^d, \quad t \in [t_0, t_{\text{end}}] \subset \mathbb{R}, \tag{1}$$

the TS methods defined in Jorba and Zou [1] (JZ), Bergsma and Mooij [2] (BM), and Chang and Corliss [3] (CC) explicitly require an estimate for the TS radius of convergence (RC).

At each (t_n, x_n), $n \geq 0$, the TS method for the numerical solution of (1) computes Taylor coefficients (TCs) $(x_n)_i$ at t_n to construct the TS approximate solution

$$T(t) = x_n + \sum_{i=1}^{p} (x_n)_i (t - t_n)^i \quad \text{on} \quad [t_n, t_{n+1}]. \tag{2}$$

TCs can be readily computed through automatic differentiation (AD) with packages such as FADBAD++ [4] and ADOL-C [5]. In choosing the order p and the stepsize $h = t_{n+1} - t_n$, the goal is to minimize the amount of computational work required during the integration process while maintaining a user specified accuracy tolerance.

JZ compute the RC from the TS as the minimum of the $p-1$ and p terms in the usual ratio test. While the JZ process is straightforward, it can be inaccurate. CC fit the tail of the computed TS to the TS of a model problem, which has proven to be satisfactory on standard test problems [6].

I propose to translate the decision logic of CC's method DRDCV from FORTRAN 77 to C/C++ and broaden its applicability to TS methods in the differential-algebraic equation(DAE) solver DAETS.

References

[1] À. Jorba, M. Zou, A software package for the numerical integration of ODEs by means of high-order Taylor methods, Experimental Mathematics 14 (1) (2005) 99–117.

[2] M. Bergsma, E. Mooij, Application of taylor series integration to reentry problems, in: AIAA Atmospheric Flight Mechanics Conference, 2016, p. 0024.

[3] Y. Chang, G. Corliss, Solving ordinary differential equations using Taylor series, ACM TOMS 8 (2) (1982) 114–144.

[4] C. Bendsten, O. Stauning, FADBAD, a flexible C++ package for automatic differentiation using the forward and backward methods, Tech. Rep. 1996-x5-94, Department of Mathematical Modelling, Technical University of Denmark, DK-2800, Lyngby, Denmark (August 1996).

[5] A. Griewank, A. Walther, Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation, 2nd Edition, SIAM, Philadelphia, PA, USA, 2008.

[6] W. H. Enright, J. D. Pryce, Two FORTRAN packages for assessing initial value methods, ACM Transactions on Mathematical Software 13 (1) (1987) 1–27.