Xinyuan Wu
Xiong You
Bin Wang

Structure-Preserving Algorithms for Oscillatory Differential Equations

Science Press
Beijing

Springer

Structure-Preserving Algorithms for Oscillatory Differential Equations

Xinyuan Wu • Xiong You • Bin Wang

Structure-Preserving Algorithms for Oscillatory Differential Equations

Science Press
Beijing

Springer

Xinyuan Wu
Nanjing University
Nanjing, China

Xiong You
Nanjing Agricultural University
Nanjing, China

Bin Wang
Nanjing University
Nanjing, China

ISBN 978-3-642-35337-6
ISBN 978-3-642-35338-3 (eBook)
DOI 10.1007/978-3-642-35338-3
Springer Heidelberg New York Dordrecht London

Jointly published with Science Press Beijing
ISBN: 978-7-03-035520-1 Science Press Beijing

Library of Congress Control Number: 2013931848

© Science Press Beijing and Springer-Verlag Berlin Heidelberg 2013
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Effective numerical solution of differential equations, although as old as differential equations themselves, has been a great challenge to numerical analysts, scientists and engineers for centuries. In recent decades, it has been universally acknowledged that differential equations arising in science and engineering often have certain structures that require preservation by the numerical integrators. Beginning with the symplectic integration of R. de Vogelaere (1956), R.D. Ruth (1983), Feng Kang (1985), J.M. Sanz-Serna (1988), E. Hairer (1994) and others, structure-preserving computation, or geometric numerical integration, has become one of the central fields of numerical differential equations. Geometric numerical integration aims at the preservation of the physical or geometric features of the exact flow of the system in long-term computation, such as the symplectic structure of Hamiltonian systems, energy and momentum of dynamical systems, time-reversibility of conservative mechanical systems, oscillatory and high oscillatory systems.

The objective of this monograph is to study structure-preserving algorithms for oscillatory problems that arise in a wide range of fields such as astronomy, molecular dynamics, classical mechanics, quantum mechanics, chemistry, biology and engineering. Such problems can often be modeled by initial value problems of second-order differential equations with a linear term characterizing the oscillatory structure of the systems. Since general-purpose high order Runge–Kutta (RK) methods, Runge–Kutta–Nyström (RKN) methods, and linear multistep methods (LMM) cannot respect the special structures of oscillatory problems in long-term integration, innovative integrators have to be designed. This monograph systematically develops theories and methods for solving second-order differential equations with oscillatory solutions.

As the basis of the whole monograph, Chap. 1 reviews the general notions and ideas related to the numerical integration of oscillatory differential equations. Chapter 2 presents multidimensional RKN methods adapted to second-order oscillatory systems.

Chapter 3 proposes extended Runge–Kutta–Nyström (ERKN) methods for initial value problems of second-order oscillatory systems with a constant frequency matrix or with a variable frequency matrix. The scheme of ERKN methods incorporates the particular structure of the differential equations into both the internal stages and the updates. A tri-colored tree theory, namely, the special extended Nyström tree (SEN-tree) theory and the related B-series theory are established, based on which the order conditions for ERKN methods are derived. The relation between ERKN methods and exponentially fitted methods is investigated. Multidimensional ERKN methods and multidimensional exponentially fitted methods are constructed.

Chapter 4 focuses on ERKN methods for oscillatory Hamiltonian systems. The symplecticity and symmetry conditions for ERKN methods are presented. Symplectic and symmetric ERKN (SSERKN) methods are applied to the Fermi–Pasta–Ulam problem and some nonlinear wave equations such as the sine-Gordon equation.

The idea of ERKN methods is extended to two-step hybrid methods in Chap. 5, to Falkner-type methods in Chap. 6, to energy-preserving methods in Chap. 7, to asymptotic methods for highly oscillatory problems in Chap. 8, and to multisymplectic methods for Hamiltonian partial differential equations in Chap. 9.

All the numerical integrators presented in this monograph have been tested for oscillatory problems from a variety of applications. They are shown to be more efficient than some existing high quality methods in the scientific literature.

Chapters 1 and 2 and Sect. 3.1 of Chap. 3 are more theoretical. Scientists and engineers who are mainly interested in numerical integrators may skip them, and this will not affect the comprehension of the rest of the monograph.

We are grateful to all the friends and colleagues for their selfless help during the preparation of this monograph. Special thanks are due to John Butcher of The University of Auckland, Christian Lubich of Universität Tübingen, Arieh Iserles of University of Cambridge, Jeff Cash of Imperial College London, Maarten de Hoop of Purdue University, Qin Sheng of Baylor University, Tobias Jahnke of Karlsruher Institut für Technologie (KIT), Achim Schädle of Heinrich Heine University Düsseldorf, Reinout Quispel and David McLaren of La Trobe University, Jesus Vigo-Aguiar of Universidad de Salamanca, and Richard Terrill of Minnesota State University for their encouragement.

We are also grateful to many friends and colleagues for reading the manuscript and for their valuable suggestions and discussions. In particular, we are grateful to Robert Peng Kong Chan of The University of Auckland, Weixing Zheng, Zuhe Shen of Nanjing University, Jianlin Xia of Purdue University, Adrian Turton Hill of Bath University, Jichun Li of University of Nevada, Las Vegas, and Xiaowen Chang of McGill University.

Thanks also go to the following people for their various help and support: Cheng Fang, Peiheng Wu, Rong Zhang, Cong Cong, Manchun Li, Shengwang Wang, Jipu Ma, Qiguang Wu, Xianglin Fei, Lin Liu, Yucheng Su, Xuesong Bao, Chengsen Lin, Wenting Tong, Chunhong Xie, Dongping Jiang, Zixiang Ouyang, Liangsheng Luo, Jinxi Zhao, Xinbao Ning, Weixue Shi, Chengkui Zhong, Jiangong You, Hourong Qin, Huicheng Yin, Xiaosheng Zhu, Zhiwei Sun, Qiang Zhang, Gaofei Zhang, Chun Li and Zhi Qian of Nanjing University, Yaolin Jiang of Xi'an Jiao Tong University, Yongzhong Song and Yushun Wang of Nanjing Normal University, Jialin Hong and Zaijiu Shang of Chinese Academy of Sciences, Jijun Liu and Zhizhong Sun of Southeast University, Shoufo Li and Aiguo Xiao of Xiang Tan University, Chuanmiao Chen of Hunan Normal University, Siqing Gan of Central South University, Chengjian Zhang and Chengming Huang of Huazhong University of Science & Technology, Shuanghu Wang of the Institute of Applied Physics and Computational Mathematics, Beijing, Hongjiong Tian of Shanghai Normal University, Yongkui Zou of Jilin University, Jingjun Zhao of Harbin Institute of Technology, Qinghong Li of Chuzhou University, Yonglei Fang of Zaozhuang University, Fan Yang and Hongli Yang of Nanjing Institute of Technology, Jiyong Li of Hebei Normal University, Wei Shi, Kai Liu, Qihua Huang, Jun Wu, Jinsong Yu and Guozhong Hu.

We would like to thank Ji Luo for her help with the editing, the editorial and production group of the Science Press, Beijing, and Springer-Verlag, Heidelberg.

We also thank our families for their love and support throughout all these years.

The work on this monograph was supported in part by the Specialized Research Foundation for the Doctoral Program of Higher Education under Grant 20100091110033, by the 985 Project at Nanjing University under Grant 9112020301, by the Natural Science Foundation of China under Grant 10771099, Grant 11271186 and Grant 11171155, and by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

Nanjing

Xinyuan Wu
Xiong You
Bin Wang

Contents

Chapter 1
Runge–Kutta (–Nyström) Methods for Oscillatory Differential Equations

In this chapter we first survey Runge–Kutta (RK) methods for initial value problems of first-order ordinary differential equations. For the purpose of deriving order conditions, the rooted tree theory is set up. For second-order differential equations, Runge–Kutta–Nyström (RKN) methods are formulated, and their order conditions are obtained based on the Nyström tree theory. For oscillatory differential equations, the dispersion and dissipation of classical numerical methods are examined. We also recall the symplectic RK and RKN methods for Hamiltonian systems. Finally, we make some comments on structure-preserving methods for solving oscillatory problems.

1.1 RK Methods, Rooted Trees, B-Series and Order Conditions

We start with an initial value problem of ordinary differential equations defined on the interval $[x_0, x_{\text{end}}]$:

$$y' = f(x, y), \quad y(x_0) = y_0, \tag{1.1}$$

where $y \in \mathbb{R}^d$ and $f : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$. From the existence theory of ordinary differential equations, the problem (1.1) has a unique solution on $[x_0, x_{\text{end}}]$ if the function $f(x, y)$ is continuous in its first variable and satisfies a Lipschitz condition in its second variable (see Butcher [3]). However, on most occasions, the true solution to the initial value problem (1.1) arising in applications, is not accessible even though it *exists*. Therefore it becomes common practice to solve the initial value problem (1.1) by numerical approaches, among which the classical RK methods are most popular.

RK methods were developed by Runge [17], Heun [12] and Kutta [14]. Although a number of different approaches have been employed in the analysis of RK methods, the one used in this chapter is that established by Butcher [1, 2], following on from the work of Gill [5] and Merson [15].

X. Wu et al., *Structure-Preserving Algorithms for Oscillatory Differential Equations*,
DOI 10.1007/978-3-642-35338-3_1,

Definition 1.1 An s-stage *Runge–Kutta (RK) method* with stepsize h for the initial value problem (1.1) reads

$$\begin{cases} Y_i = y_n + h\sum_{j=1}^{s} a_{ij} f(x_n + c_j h, Y_j), \quad i = 1, \ldots, s, \\ y_{n+1} = y_n + h\sum_{i=1}^{s} b_i f(x_n + c_i h, Y_i), \end{cases} \tag{1.2}$$

or equivalently,

$$\begin{cases} k_i = f\left(x_n + c_i h, y_n + h\sum_{j=1}^{s} a_{ij} k_j\right), \quad i = 1, \ldots, s, \\ y_{n+1} = y_n + h\sum_{i=1}^{s} b_i k_i, \end{cases} \tag{1.3}$$

where $a_{ij}, b_i, c_i, i, j = 1, \ldots, s$ are real constants, $x_n = x_0 + nh$, h is the stepsize and $y_n \approx y(x_n)$, $n = 0, 1, \ldots$.

The RK method (1.2) can be expressed briefly by the following Butcher tableau:

$$\begin{array}{c|c} c & A \\ \hline & b^{\mathrm{T}} \end{array} \quad = \quad \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array}$$

where $b = (b_1, \ldots, b_s)^{\mathrm{T}}$, $c = (c_1, \ldots, c_s)^{\mathrm{T}}$ are s-dimensional vectors and $A = (a_{ij})$ is an $s \times s$ matrix. When the matrix A is strictly lower triangular, i.e., $a_{ij} = 0$ for all $1 \le i \le j \le s$, the method is *explicit*, otherwise it is *implicit*.

Conventionally, it is assumed that $Ae = c$ with $e = (1, \ldots, 1)^{\mathrm{T}}$, the $s \times 1$ vector of units, i.e.,

$$c_i = \sum_{j=1}^{s} a_{ij}, \quad i = 1, \ldots, s.$$

Sometimes, it is convenient to express the RK method (1.2) in block-matrix notation

$$\begin{cases} Y = e \otimes y_n + h(A \otimes I) f(Y), \\ y_{n+1} = y_n + h\big(b^{\mathrm{T}} \otimes I\big) f(Y), \end{cases} \tag{1.4}$$

where I is the $d \times d$ identity matrix, $\otimes$ is the Kronecker product, $Y = (Y_1^{\mathrm{T}}, \ldots, Y_s^{\mathrm{T}})^{\mathrm{T}}$ and $f(Y) = (f(x_n + c_1 h, Y_1)^{\mathrm{T}}, \ldots, f(x_n + c_s h, Y_s)^{\mathrm{T}})^{\mathrm{T}}$.

Definition 1.2 An RK method (1.2) has *order* p if for a sufficiently smooth problem (1.1), under the assumption that $y_n = y(x_n)$, the local error $y(x_n + h) - y_{n+1}$ satisfies

$$y(x_n + h) - y_{n+1} = \mathscr{O}\big(h^{p+1}\big). \tag{1.5}$$

In principle, order conditions can be obtained by comparing the Taylor series of the numerical solution y_{n+1} with that of the true solution $y(x_n + h)$. By definition, an RK method has order p if and only if these two series coincide up to the term h^p. However, for high orders, the derivation of order conditions becomes very complicated due to the large number of terms. To find a way out, Butcher provides a graphical representation for Taylor expansions of the true solution and the numerical solution in terms of the so-called rooted tree theory (see [3, 9]).

As is known, the non-autonomous problem (1.1) can be converted to an autonomous form by appending the equation $x' = 1$. The application of the RK method (1.2) to the non-autonomous problem (1.1) and to its autonomous form yields the same numerical solution. Therefore we need only develop the order condition theory for the following autonomous problem:

$$y' = f(y), \quad y(x_0) = y_0. \tag{1.6}$$

The RK method (1.3) then becomes

$$\begin{cases} k_i = f\left(y_n + h\sum_{j=1}^{s} a_{ij}k_j\right), \quad i = 1, \dots, s, \\ y_{n+1} = y_n + h\sum_{i=1}^{s} b_i k_i. \end{cases} \tag{1.7}$$

To explain how the rooted trees are formulated, we consider the first four derivatives of the true solution of the problem (1.6) at $x = x_n$:

$$\begin{aligned} y' &= f, \\ y'' &= f'f, \\ y''' &= f''(f, f) + f'f'f, \\ y^{(4)} &= f'''(f, f, f) + 3f''\big(f'f, f\big) + f'f''(f, f) + f'f'f'f, \end{aligned} \tag{1.8}$$

where the arguments (x_n) on the left-hand side and $(y(x_n))$ on the right-hand side are suppressed. We use a vertex to represent each f, a vertex with one branch pointing upwards to represent each f', and a vertex with k branches pointing upwards to represent each $f^{(k)}$. The following is the formal definition of the set of rooted trees.

Definition 1.3 The set of (rooted) *trees* T is recursively defined as follows:

(i) the graph • with only one vertex (the root) is in T;

(ii) if $t_1, t_2, \dots, t_m \in T$, then the graph obtained by grafting the roots of $t_1, \dots, t_m$ to a new vertex, is also in T. This tree is denoted by

$$t = [t_1, \dots, t_m],$$

and the new vertex is the root of the tree t.

Definition 1.4 The *order*, an integer-valued function $\rho : T \to \mathbb{N}$, is recursively defined as follows:

(i) $\rho(\bullet) = 1$;
(ii) for $t = [t_1, \dots, t_m] \in T$,

$$\rho(t) = 1 + \sum_{i=1}^{m} \rho(t_i).$$

For each $t \in T$, the order $\rho(t)$ is the number of vertices of t. The set of all trees of order q is denoted by T_q.

Definition 1.5 The integer-valued function $\alpha : T \to \mathbb{N}$ is recursively defined as follows:

(i) $\alpha(\bullet) = 1$;
(ii) for $t = [t_1^{\mu_1}, \dots, t_m^{\mu_m}] \in T$ with t_i, $i = 1, \dots, m$ distinct,

$$\alpha(t) = \big(\rho(t) - 1\big)! \prod_{i=1}^{m} \frac{1}{\mu_i!} \left(\frac{\alpha(t_i)}{\rho(t_i)!} \right)^{\mu_i},$$

where μ_i is the multiplicity of t_i, $i = 1, \dots, m$.

For each $t \in T$, $\alpha(t)$ is the integer coefficient of each term in the formula (1.8), and is the number of different monotonic labellings of t.

Definition 1.6 For each $t \in T$, the *elementary differential*, a vector-valued function $\mathscr{F}(t) : \mathbb{R}^d \to \mathbb{R}^d$, is recursively defined as follows:

(i) $\mathscr{F}(\bullet)(y) = f(y)$;
(ii) for $t = [t_1, \dots, t_m] \in T$,

$$\mathscr{F}(t)(y) = f^{(m)}(y)\big(\mathscr{F}(t_1)(y), \dots, \mathscr{F}(t_m)(y)\big).$$

With the above definitions, we have the following result.

Theorem 1.1 *The qth derivative of the true solution of the problem* (1.6) *can be expressed by*

$$y^{(q)}(x_n) = \sum_{t \in T_q} \alpha(t)\mathscr{F}(t)\big(y(x_n)\big), \tag{1.9}$$

and the true solution $y(x_n+h)$ *can be expressed by the following series*:

$$\begin{aligned} y(x_n+h) &= y(x_n) + \sum_{q=1}^{\infty} \frac{h^q}{q!} \sum_{t \in T_q} \alpha(t)\mathscr{F}(t)\big(y(x_n)\big) \\ &= y(x_n) + \sum_{t \in T} \frac{h^{\rho(t)}}{\rho(t)!} \alpha(t)\mathscr{F}(t)\big(y(x_n)\big). \end{aligned} \tag{1.10}$$

On the other hand, if the numerical solution y_{n+1} is regarded as a function of h, we observe that the first four derivatives of $y_{n+1}(h)$ at $h=0$ have the following expressions:

$$\begin{aligned} y'_{n+1} &= (1\cdot 1)\sum_{i=1}^{s} b_i f, \\ y''_{n+1} &= (1\cdot 2)\sum_{i,j=1}^{s} b_i a_{ij} f' f, \\ y'''_{n+1} &= (1\cdot 3)\sum_{i,j,k=1}^{s} b_i a_{ij} a_{ik} f''(f,f) + (1\cdot 6)\sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} f' f' f, \\ y^{(4)}_{n+1} &= (1\cdot 4)\sum_{i,j,k,l=1}^{s} b_i a_{ij} a_{ik} a_{il} f'''(f,f,f) \\ &\quad + (3\cdot 8)\sum_{i,j,k,l=1}^{s} b_i a_{ij} a_{ik} a_{kl} f''\big(f, f' f\big) \\ &\quad + (1\cdot 12)\sum_{i,j,k,l=1}^{s} b_i a_{ij} a_{jk} a_{jl} f' f''(f,f) \\ &\quad + (1\cdot 24)\sum_{i,j,k,l=1}^{s} b_i a_{ij} a_{jk} a_{kl} f' f' f' f, \end{aligned} \tag{1.11}$$

where the arguments (0) on the left-hand sides and (y_n) on the right-hand sides are suppressed, the first factor in the coefficient in the bracket of each term is $\alpha(t)$ and the second is $\gamma(t)$ as defined below.

Definition 1.7 The *density*, an integer-valued function $\gamma : T \to \mathbb{N}$, is recursively defined by

(i) $\gamma(\bullet) = 1$;
(ii) for $t = [t_1, \ldots, t_m] \in T$,

$$\gamma(t) = \rho(t) \prod_{i=1}^{m} \gamma(t_i).$$

Definition 1.8 The function Φ_i on T is recursively defined by

(i) $\Phi_i(\bullet)=1$;
(ii) for $t=[t_1^{\mu_1},\dots,t_m^{\mu_m}]\in T$, with $t_1,\dots,t_m$ distinct,

$$\Phi_i(t)=\left(\sum_{j=1}^{s}a_{ij}\Phi_j(t_1)\right)^{\mu_1}\cdots\left(\sum_{k=1}^{s}a_{ik}\Phi_k(t_m)\right)^{\mu_m},$$

where μ_i is the multiplicity of t_i, $i=1,\dots,m$. We write the vector

$$\Phi(t)=\big(\Phi_1(t),\dots,\Phi_s(t)\big)^{\mathrm{T}}.$$

The rooted trees of order up to five, with values of the related functions are listed in Table 1.1. A dot "·" between two vectors indicates the componentwise product and a power of a vector is also understood as componentwise.

Based on the above Definitions 1.3–1.8, we have the following theorem.

Theorem 1.2 *The qth derivative at $h=0$ of the numerical solution y_{n+1} of the problem* (1.6) *produced by the RK method* (1.7) *can be expressed by*

$$y_{n+1}^{(q)}\big|_{h=0}=\sum_{t\in T_q}\alpha(t)\gamma(t)b^{\mathrm{T}}\Phi(t)\mathscr{F}(t)(y_n), \tag{1.12}$$

and the numerical solution y_{n+1} can be expressed by the following series:

$$\begin{aligned}y_{n+1}&=y_n+\sum_{q=1}^{\infty}\frac{h^q}{q!}\sum_{t\in T_q}\alpha(t)\gamma(t)b^{\mathrm{T}}\Phi(t)\mathscr{F}(t)(y_n)\\&=y_n+\sum_{t\in T}\frac{h^{\rho(t)}}{\rho(t)!}\alpha(t)\gamma(t)b^{\mathrm{T}}\Phi(t)\mathscr{F}(t)(y_n).\end{aligned} \tag{1.13}$$

From Theorems 1.1 and 1.2, we arrive at the following order conditions.

Theorem 1.3 *The RK method* (1.7) *has order p if and only if the condition*

$$b^{\mathrm{T}}\Phi(t)=\frac{1}{\gamma(t)} \tag{1.14}$$

holds for every tree $t\in\bigcup_{q=1}^{p}T_q$.

For a standard and systematic presentation with original insights of rooted tree theory, see Butcher's book [3].

Table 1.1 Rooted trees of order up to five, elementary differentials and coefficients

$\rho(t)$	t	$\alpha(t)$	$\gamma(t)$	$\Phi(t) = (\Phi_1(t), \ldots, \Phi_s(t))^{\mathrm{T}}$	$\mathscr{F}(t)(y)$
1		1	1	e	f
2		1	2	c	$f'f$
3		1	3	c^2	$f''(f,f)$
		1	6	Ac	$f'f'f$
4		1	4	c^3	$f'''(f,f,f)$
		3	8	$c \cdot Ac$	$f''(f,f'f)$
		1	12	Ac^2	$f'f''(f,f)$
		1	24	A^2c	$f'f'f'f$
5		1	5	c^4	$f^{(4)}(f,f,f,f)$
		6	10	$c^2 \cdot Ac$	$f'''(f,f,f'f)$
		4	15	$c \cdot Ac^2$	$f''(f,f''(f,f))$
		4	30	$c \cdot A^2c$	$f''(f,f'f'f)$
		3	20	$(Ac)^2$	$f''(f'f,f'f)$
		1	20	Ac^3	$f'f'''(f,f,f)$
		3	40	$A(c \cdot Ac)$	$f'f''(f,f'f)$
		1	60	A^2c^2	$f'f'f''(f,f)$
		1	120	A^3c	$f'f'f'f'f$

As typical examples, two classical fourth order explicit RK methods are given by the following Butcher tableaux:

$$
\begin{array}{c|cccc}
0 & & & & \\
1/2 & 1/2 & & & \\
1/2 & 0 & 1/2 & & \\
1 & 0 & 0 & 1 & \\
\hline
 & 1/6 & 2/6 & 2/6 & 1/6
\end{array}
\qquad
\begin{array}{c|cccc}
0 & & & & \\
1/3 & 1/3 & & & \\
2/3 & -1/3 & 1 & & \\
1 & 1 & -1 & 1 & \\
\hline
 & 1/8 & 3/8 & 3/8 & 1/8
\end{array}
\tag{1.15}
$$

1.2 RKN Methods, Nyström Trees and Order Conditions

1.2.1 Formulation of the Scheme

We turn to the initial value problem of second-order non-autonomous ordinary differential equations defined on the interval $[x_0, x_{\mathrm{end}}]$:

$$\begin{cases} y'' = f\big(x, y, y'\big), \\ y(x_0) = y_0, \quad y'(x_0) = y_0', \end{cases} \tag{1.16}$$

where $y \in \mathbb{R}^d$ and $f : \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$. This system can be transformed into a first-order system:

$$\begin{pmatrix} y \\ y' \end{pmatrix}' = \begin{pmatrix} y' \\ f(x, y, y') \end{pmatrix}, \qquad \begin{pmatrix} y(x_0) \\ y'(x_0) \end{pmatrix} = \begin{pmatrix} y_0 \\ y_0' \end{pmatrix}, \tag{1.17}$$

and the RK method (1.2) is applicable.

Nyström [16] suggests a direct approach. To this end, we apply the variation-of-constants formula in Wu et al. [19] to (1.17) and get the following formula of integral equations:

$$\begin{cases} y(x_n + \mu h) = y(x_n) + \mu h y'(x_n) + h^2 \displaystyle\int_0^{\mu} (\mu - z)\varphi(x_n + hz)\,\mathrm{d}z, \\ y'(x_n + \mu h) = y'(x_n) + h \displaystyle\int_0^{\mu} \varphi(x_n + hz)\,\mathrm{d}z, \end{cases} \tag{1.18}$$

where $\varphi(\nu) := f(\nu, y(\nu), y'(\nu))$. Approximating the integrals in the formula (1.18) with suitable quadrature formulae yields the well-known RKN scheme (see [16]) given by the following definition.

Definition 1.9 An s-stage *Runge–Kutta–Nyström (RKN) method* for the initial value problem (1.16) is defined by

$$\begin{cases} Y_i = y_n + c_i h y_n' + h^2 \displaystyle\sum_{j=1}^{s} \bar{a}_{ij} f\big(x_n + c_j h, Y_j, Y_j'\big), \quad i = 1, \ldots, s, \\ Y_i' = y_n' + h \displaystyle\sum_{j=1}^{s} a_{ij} f\big(x_n + c_j h, Y_j, Y_j'\big), \quad i = 1, \ldots, s, \\ y_{n+1} = y_n + h y_n' + h^2 \displaystyle\sum_{i=1}^{s} \bar{b}_i f\big(x_n + c_i h, Y_i, Y_i'\big), \\ y_{n+1}' = y_n' + h \displaystyle\sum_{i=1}^{s} b_i f\big(x_n + c_i h, Y_i, Y_i'\big), \end{cases} \tag{1.19}$$

or equivalently,

$$\begin{cases} k_i = f\left(x_n + c_i h,\, y_n + c_i h y_n' + h^2 \sum\limits_{j=1}^{s} \bar{a}_{ij} k_j,\, y_n' + h \sum\limits_{j=1}^{s} a_{ij} k_j\right), \quad i = 1, \ldots, s, \\ y_{n+1} = y_n + h y_n' + h^2 \sum\limits_{i=1}^{s} \bar{b}_i k_i, \\ y_{n+1}' = y_n' + h \sum\limits_{i=1}^{s} b_i k_i, \end{cases} \tag{1.20}$$

where $\bar{a}_{ij}, a_{ij}, \bar{b}_i, b_i, c_i, i, j = 1, \ldots, s$ are real constants.

The RKN method (1.19) can be expressed by the following Butcher tableau:

$$\begin{array}{c|c|c} c & \bar{A} & A \\ \hline & \bar{b}^{\mathrm{T}} & b^{\mathrm{T}} \end{array} \quad = \quad \begin{array}{c|ccc|ccc} c_1 & \bar{a}_{11} & \cdots & \bar{a}_{1s} & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_s & \bar{a}_{s1} & \cdots & \bar{a}_{ss} & a_{s1} & \cdots & a_{ss} \\ \hline & \bar{b}_1 & \cdots & \bar{b}_s & b_1 & \cdots & b_s \end{array}$$

where $\bar{b} = (\bar{b}_1, \ldots, \bar{b}_s)^{\mathrm{T}}$, $b = (b_1, \ldots, b_s)^{\mathrm{T}}$ and $c = (c_1, \ldots, c_s)^{\mathrm{T}}$ are s-dimensional vectors, $\bar{A} = (\bar{a}_{ij})$ and $A = (a_{ij})$ are $s \times s$ matrices. In block-matrix notation, we can write the scheme (1.19) as

$$\begin{cases} Y = e \otimes y_n + hc \otimes y_n' + h^2 (\bar{A} \otimes I) f\big(Y, Y'\big), \\ Y' = e \otimes y_n' + h (A \otimes I) f\big(Y, Y'\big), \\ y_{n+1} = y_n + h y_n' + h^2 \big(\bar{b}^{\mathrm{T}} \otimes I\big) f\big(Y, Y'\big), \\ y_{n+1}' = y_n' + h \big(b^{\mathrm{T}} \otimes I\big) f\big(Y, Y'\big), \end{cases} \tag{1.21}$$

where

$$Y = \big(Y_1^{\mathrm{T}}, \ldots, Y_s^{\mathrm{T}}\big)^{\mathrm{T}}, \quad Y' = \big(Y_1'^{\mathrm{T}}, \ldots, Y_s'^{\mathrm{T}}\big)^{\mathrm{T}},$$

$$f\big(Y, Y'\big) = \Big(f\big(x_n + c_1 h, Y_1, Y_1'\big)^{\mathrm{T}}, \ldots, f\big(x_n + c_s h, Y_s, Y_s'\big)^{\mathrm{T}}\Big)^{\mathrm{T}}.$$

1.2.2 Nyström Trees and Order Conditions

Definition 1.10 An RKN method (1.19) has *order p* if for the sufficiently smooth problem (1.16), under the assumptions $y_n = y(x_n)$ and $y_n' = y'(x_n)$, the local errors of the solution and its derivative satisfy

$$y(x_n + h) - y_{n+1} = \mathscr{O}\big(h^{p+1}\big), \qquad y'(x_n + h) - y_{n+1}' = \mathscr{O}\big(h^{p+1}\big). \tag{1.22}$$

As in the case of RK methods, when working on the order conditions for RKN methods, we need only consider the autonomous problem

$$\begin{cases} y'' = f\big(y, y'\big), \\ y(x_0) = y_0, \quad y'(x_0) = y_0'. \end{cases} \tag{1.23}$$

Accordingly, the scheme (1.20) takes the form

$$\begin{cases} k_i = f\left(y_n + c_i h y_n' + h^2 \sum\limits_{j=1}^{s} \bar{a}_{ij} k_j, y_n' + h \sum\limits_{j=1}^{s} a_{ij} k_j\right), \quad i = 1, \ldots, s, \\ y_{n+1} = y_n + h y_n' + h^2 \sum\limits_{i=1}^{s} \bar{b}_i k_i, \\ y_{n+1}' = y_n' + h \sum\limits_{i=1}^{s} b_i k_i. \end{cases} \tag{1.24}$$

By definition, an RKN method has order p if and only if the Taylor series of the true solution $y(x_n + h)$ in powers of h and that of the numerical solution y_{n+1} coincide up to the term h^p, and the Taylor series of the derivative of the true solution $y'(x_n + h)$ and that of the approximate derivative y_{n+1}' coincide up to the term h^p. By analogy with the case of RK methods, the Nyström tree theory is developed to determine the order conditions for RKN methods. We refer the reader to [4, 6, 7, 10, 11] for details.

We first consider the first to the fifth order derivatives of the true solution of the problem (1.23) at $x = x_n$:

$$\begin{aligned} y' &= y', \\ y'' &= f, \\ y''' &= f_y' y' + f_{y'}' f, \\ y^{(4)} &= f_{yy}''\big(y', y'\big) + 2 f_{yy'}''\big(y', f\big) + f_{y'y'}''(f, f) + f_y' f + f_{y'}' f_y' y' + f_{y'}' f_{y'}' f, \\ y^{(5)} &= f_{yyy}'''\big(y', y', y'\big) + 3 f_{yyy'}'''\big(y', y', f\big) + 3 f_{yy'y'}'''\big(y', f, f\big) + f_{y'y'y'}'''(f, f, f) \\ &\quad + 3 f_{yy}''\big(y', f\big) + 3 f_{yy'}''\big(y', f_y' y'\big) + 3 f_{yy'}''\big(y', f_{y'}' f\big) + 3 f_{y'y}''(f, f) \\ &\quad + 3 f_{y'y'}''\big(f, f_y' y'\big) + 3 f_{y'y'}''\big(f, f_{y'}' f\big) + f_{y'}' f_{yy}''\big(y', y'\big) + 2 f_{y'}' f_{yy'}''\big(y', f\big) \\ &\quad + f_{y'}' f_{y'y'}''(f, f) + f_y' f_y' y' + f_y' f_{y'}' f + f_{y'}' f_y' f + f_{y'}' f_{y'}' f_y' y' + f_{y'}' f_{y'}' f_{y'}' f, \end{aligned} \tag{1.25}$$

where the arguments (x_n) and $(y(x_n), y'(x_n))$ are suppressed. In order to express geometrically each term in the above formula, we use

(i) a terminal white vertex to represent each f;
(ii) a terminal black vertex to represent each y';
(iii) a white vertex, with k branches pointing upwards to black vertices and with l branches pointing upwards to white vertices, to represent each $f_{y\cdots yy'\cdots y'}^{(k+l)}$, the kth partial derivative with respect to y, lth partial derivative with respect to y';

(iv) a non-terminal black vertex, with one branch pointing upwards to a white vertex, to represent 1.

This leads to the following definition of the set of Nyström trees.

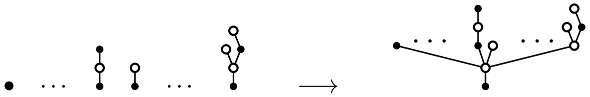
Definition 1.11 The set of Nyström trees (N-trees) NT is recursively defined as follows:

(i) the graph • with only one black vertex (the root), denoted by τ, belongs to NT; the graph , denoted by τ_2, belongs to NT;

(ii) if $t_1, \dots, t_r, t_{r+1}, \dots, t_m \in NT$, $t_{r+1}, \dots, t_m$ being different from τ, then the graph obtained by connecting the roots of $t_1, \dots, t_r$ downward to a new white vertex, merging the roots of $t_{r+1}, \dots, t_m$ into this white vertex, and then connecting this white vertex downward to a new black vertex, belongs to NT. It is denoted by

$$t = \big[t_1, \dots, t_r, \langle t_{r+1}, \dots, t_m\rangle\big]_2,$$

and the new black vertex is the root of the tree t.



Definition 1.12 The *order*, an integer-valued function $\rho : NT \to \mathbb{N}$, is recursively defined as follows:

(i) $\rho(\tau) = 1$, $\rho(\tau_2) = 2$;

(ii) for $t = [t_1, \dots, t_r, \langle t_{r+1}, \dots, t_m\rangle]_2 \in NT$,

$$\rho(t) = 2 + \sum_{i=1}^{r} \rho(t_i) + \sum_{i=r+1}^{m} \big(\rho(t_i) - 1\big).$$

For each $t \in NT$, the order $\rho(t)$ is the number of vertices of t. The set of all N-trees of order q is denoted by NT_q.

Definition 1.13 The integer-valued function $\alpha : NT \to \mathbb{N}$ is recursively defined as follows:

(i) $\alpha(\tau)= 1$, $\alpha(\tau_2)= 1$;

(ii) for $t = [t_1^{\mu_1}, \dots, t_r^{\mu_r}, \langle t_{r+1}^{\mu_{r+1}}, \dots, t_m^{\mu_m}\rangle]_2 \in NT$ with $t_1, \dots, t_r$ distinct, and $t_{r+1}, \dots, t_m$ distinct,

$$\alpha(t) = \big(\rho(t) - 2\big)! \prod_{i=1}^{r} \frac{1}{\mu_i!}\left(\frac{\alpha(t_i)}{\rho(t_i)!}\right)^{\mu_i} \prod_{i=r+1}^{m} \frac{1}{\mu_i!}\left(\frac{\alpha(t_i)}{(\rho(t_i) - 1)!}\right)^{\mu_i},$$

where μ_i is the multiplicity of t_i, $i = 1, \dots, m$.

For each $t \in NT$, $\alpha(t)$ is the integer coefficient of a term in the formula (1.25), which is the number of different monotonic labellings of t.

Definition 1.14 For each tree $t \in NT$, the *elementary differential* is a vector-valued function $\mathscr{F}(t): \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ recursively defined as follows:

(i) $\mathscr{F}(\tau)(y, y') = y'$, $\mathscr{F}(\tau_2)(y, y') = f(y, y')$;
(ii) for $t = [t_1, t_2, \ldots, t_r, \langle t_{r+1}, \ldots, t_m\rangle]_2 \in NT$,

$$\mathscr{F}(t)\big(y, y'\big) = \frac{\partial^m f}{\partial y^r \partial y'^{m-r}}\big(y, y'\big)\big(\mathscr{F}(t_1)\big(y, y'\big), \ldots, \mathscr{F}(t_m)\big(y, y'\big)\big).$$

Thus we have the following result.

Theorem 1.4 *The qth derivative of the true solution of the problem* (1.23) *can be expressed by*

$$y^{(q)}(x_n) = \sum_{t \in NT_q} \alpha(t)\mathscr{F}(t)\big(y(x_n), y'(x_n)\big). \tag{1.26}$$

The true solution $y(x_n + h)$ and its derivative can be expressed by the following series:

$$\begin{aligned} y(x_n + h) &= y(x_n) + \sum_{q=1}^{\infty} \frac{h^q}{q!} \sum_{t \in NT_q} \alpha(t)\mathscr{F}(t)\big(y(x_n), y'(x_n)\big) \\ &= y(x_n) + \sum_{t \in NT} \frac{h^{\rho(t)}}{\rho(t)!}\alpha(t)\mathscr{F}(t)\big(y(x_n), y'(x_n)\big), \end{aligned} \tag{1.27}$$

$$\begin{aligned} y'(x_n + h) &= y'(x_n) + \sum_{q=2}^{\infty} \frac{h^{q-1}}{(q-1)!} \sum_{t \in NT_q} \alpha(t)\mathscr{F}(t)\big(y(x_n), y'(x_n)\big) \\ &= \sum_{t \in NT} \frac{h^{\rho(t)-1}}{(\rho(t)-1)!}\alpha(t)\mathscr{F}(t)\big(y(x_n), y'(x_n)\big). \end{aligned} \tag{1.28}$$

On the other hand, if the numerical solution y_{n+1} and the approximate derivative y'_{n+1} are regarded as functions of h, we observe that the first four derivatives of

$y_{n+1}(h)$ and $y'_{n+1}(h)$ at $h = 0$ have the following expressions:

$$
\begin{aligned}
y'_{n+1} &= (1\cdot 1)\,y'_n,\\
y''_{n+1} &= (1\cdot 2)\sum_{i=1}^{s}\bar{b}_i f,\\
y'''_{n+1} &= (1\cdot 6)\sum_{i=1}^{s}\bar{b}_i c_i f'_y y'_n + (1\cdot 6)\sum_{i,j=1}^{s}\bar{b}_i a_{ij} f'_{y'} f,\\
y^{(4)}_{n+1} &= (1\cdot 12)\sum_{i=1}^{s}\bar{b}_i c_i^2 f''_{yy}\big(y'_n, y'_n\big) + (2\cdot 12)\sum_{i=1}^{s}\bar{b}_i c_i a_{ij} f''_{yy'}\big(y'_n, f\big)\\
&\quad + (1\cdot 12)\sum_{i=1}^{s}\bar{b}_i\left(\sum_{j=1}^{s}a_{ij}\right)^2 f''_{y'y'}(f, f) + (1\cdot 24)\sum_{i,j=1}^{s}\bar{b}_i\bar{a}_{ij} f'_y f\\
&\quad + (1\cdot 24)\sum_{i,j=1}^{s}\bar{b}_i a_{ij} c_j f'_{y'} f'_y y'_n + (1\cdot 24)\sum_{i,j,k=1}^{s}\bar{b}_i a_{ij} a_{jk} f'_{y'} f'_{y'} f,
\end{aligned}
\tag{1.29}
$$

and

$$
\begin{aligned}
\big(y'_{n+1}\big)' &= (1\cdot 1)\sum_{i=1}^{s} b_i f,\\
\big(y'_{n+1}\big)'' &= (1\cdot 2)\sum_{i=1}^{s} b_i c_i f'_y y'_n + (1\cdot 2)\sum_{i=1}^{s} b_i a_{ij} f'_{y'} f,\\
\big(y'_{n+1}\big)''' &= (1\cdot 3)\sum_{i=1}^{s} b_i c_i^2 f''_{yy}\big(y'_n, y'_n\big) + (2\cdot 3)\sum_{i=1}^{s} b_i c_i a_{ij} f''_{yy'}\big(y'_n, f\big)\\
&\quad + (1\cdot 3)\sum_{i=1}^{s} b_i\left(\sum_{j=1}^{s}a_{ij}\right)^2 f''_{y'y'}(f, f) + (1\cdot 6)\sum_{i,k=1}^{s} b_i\bar{a}_{ik} f'_y f\\
&\quad + (1\cdot 6)\sum_{i,j=1}^{s} b_i a_{ij} c_j f'_{y'} f'_y y'_n + (1\cdot 6)\sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} f'_{y'} f'_{y'} f,\\
\big(y'_{n+1}\big)^{(4)} &= (1\cdot 4)\sum_{i=1}^{s} b_i c_i^3 f'''_{yyy}\big(y'_n, y'_n, y'_n\big) + (3\cdot 4)\sum_{i,j=1}^{s} b_i c_i^2 a_{ij} f'''_{yyy'}\big(y'_n, y'_n, f\big)\\
&\quad + (3\cdot 4)\sum_{i=1}^{s} b_i c_i\left(\sum_{j=1}^{s}a_{ij}\right)^2 f'''_{yy'y'}\big(y'_n, f, f\big)
\end{aligned}
$$

$$
\begin{aligned}
&+ (1\cdot 4)\sum_{i=1}^{s} b_i\left(\sum_{j=1}^{s} a_{ij}\right)^{3} f'''_{y'y'y'}(f, f, f)\\
&+ (3\cdot 8)\sum_{i,k=1}^{s} b_i c_i \bar{a}_{ik} f''_{yy}\big(y'_n, f\big) + (3\cdot 8)\sum_{i,j,k=1}^{s} b_i a_{ij}\bar{a}_{ik} f''_{y'y}(f, f)\\
&+ (3\cdot 8)\sum_{i,j=1}^{s} b_i c_i a_{ij} c_j f''_{yy'}\big(y'_n, f'_y y'_n\big)\\
&+ (3\cdot 8)\sum_{i,j,k=1}^{s} b_i c_i a_{ij} a_{jk} f''_{yy'}\big(y'_n, f'_{y'} f\big)\\
&+ (3\cdot 8)\sum_{i,j,k=1}^{s} b_i a_{ij} a_{ik} c_k f''_{y'y'}\big(f, f'_y y'_n\big)\\
&+ (3\cdot 8)\sum_{i,j,k,l=1}^{s} b_i a_{ij} a_{ik} a_{kl} f''_{y'y'}\big(f, f'_{y'} f\big)\\
&+ (1\cdot 12)\sum_{i,j=1}^{s} b_i a_{ij} c_j^2 f'_{y'} f''_{yy}\big(y'_n, y'_n\big)\\
&+ (2\cdot 12)\sum_{i,j,k=1}^{s} b_i a_{ij} c_j a_{jk} f'_{y'} f''_{yy'}\big(y'_n, f\big)\\
&+ (1\cdot 12)\sum_{i,j=1}^{s} b_i a_{ij}\left(\sum_{k=1}^{s} a_{jk}\right)^{2} f'_{y'} f''_{y'y'}(f, f)\\
&+ (1\cdot 24)\sum_{i,k=1}^{s} b_i \bar{a}_{ik} c_k f'_y f'_y y'_n\\
&+ (1\cdot 24)\sum_{i,k,l=1}^{s} b_i \bar{a}_{ik} a_{kl} f'_y f'_{y'} f + (1\cdot 24)\sum_{i,j,l=1}^{s} b_i a_{ij}\bar{a}_{jl} f'_{y'} f'_y f\\
&+ (1\cdot 24)\sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} c_k f'_{y'} f'_{y'} f'_y y'_n\\
&+ (1\cdot 24)\sum_{i,j,k,l=1}^{s} b_i a_{ij} a_{jk} a_{kl} f'_{y'} f'_{y'} f'_{y'} f,
\end{aligned}
\tag{1.30}
$$

where the arguments (0) on the left-hand side and (y_n, y'_n) on the right-hand side are suppressed.