NUMERICAL ANALYSIS AND METHODS FOR ORDINARY DIFFERENTIAL EQUATIONS

N.N. Kalitkin
Institute for Mathematical Modeling, Russian Academy of Sciences, Moscow, Russia

S.S. Filippov
Keldysh Institute of Applied Mathematics, Moscow, Russia

Keywords: Numerical analysis, linear algebra, spectrum of a matrix, nonlinear equations, quadrature formulae, grid refinement, collocation, interpolation, mean square approximation, spline, interval analysis, differential equations, stiff problems, stability of a numerical method.

Contents

Summary

For many problems of calculus we can not obtain an exact solution in terms of elementary functions. Because of this, we have to use numerical methods.

Mathematicians have developed a great number of methods for various classes of problems. On the basis of these methods packages of applied program have been created. The power of computers including personal ones is increasing fast. This enables a user who is not a highly skilled specialist to perform complicated calculations with the help of standard packages.

However, there does not exist an absolutely reliable package. When solving a complicated problem, these packages may give an unpleasant surprise producing a wrong result. In order to evaluate the correctness of a result, it is necessary to know in which case an algorithm being used in a program can be applied. In this chapter the simplest and most reliable algorithms for solving typical problems of numerical analysis often met with are presented and accuracy and reliability of these algorithms are discussed. Special attention is given to the *a posteriori* asymptotic error estimate that allows one to perform calculations within a guaranteed limit of accuracy.

1. Introduction

1.1. Problems of Numerical Analysis

It is rare for a mathematical problem to admit a solution which is expressed in terms of elementary or, in the last resort, well-studied special functions. For example, the roots of a polynomial of degree higher than four are not expressed in terms of radicals, the function $\sin(x^2)$ can not be integrated exactly, and a simple differential equation $du/dx = u^2 + x^2$ has no solution expressed in terms of elementary functions. This is especially true for applied problems. For example, assume that we need to integrate or to differentiate a function whose values are obtained by experimental measurements, i.e., a function defined on a grid (in the classical sense such a function has neither an integral nor a derivative).

If an exact solution can not be obtained, then numerical methods are applied; calculations are usually performed with the help of a computer program. An “ideal” numerical method consists of three parts:

- a basic algorithm reducing calculation to a sequence of operations that can be performed by a computer (logical and arithmetical operations as well as calculation of values of elementary and special functions included in software);
- the proof of convergence of a numerical solution to an exact one when passing to a limit (for example, as a mesh size tends to zero in grid methods or as a number of iteration steps tends to infinity in iteration methods);
- the estimation of an error of concrete calculations or auxiliary algorithms that choose parameters of a basic algorithm in such a way as to perform calculation within a given accuracy.

It seems that an ideal numerical method enables one to solve a problem successfully which is true in most cases. However, severe challenges may arise. Consider them in detail.

1.1.1. Well-posed Problems

Any original mathematical problem is as follows: find unknown data u from given data $w.$ Formally this can be written as

$$u = A(w), \quad u \in U, \quad w \in W; \tag{1}$$

where *U*, *W* are ranges of the data, an operator *A* defines an original problem. Let us formulate three conditions for an operator to be well-posed in the sense of Hadamard:

- for any admissible *w* there exists a solution *u* of the problem (1);
- in some finite neighborhood this solution is unique;
- a solution continuously depends on given data, to this end variations in some norm must satisfy the inequality

$$\| \delta u \| \le c[w] \cdot \| \delta w \|, \quad 0 \le c[w] < \infty; \tag{2}$$

where *c* depends on *w* (for linear problems *c* = const, i.e., *c* does not depend on *w*).

When solving the problem (1) numerically, all three conditions must be fulfilled. In fact, a) it would make no sense to calculate a solution if it does not exist; b) since an algorithm is a uniquely determined sequence of operations, it can converge to a single solution and not to several ones; c) calculations are always performed with some errors that is equivalent to some small variations $\delta w;$ the violation of the third condition may result in large variations $\delta u,$ i.e., in this case convergence does not take place.

Thus, we require for the problem (1) to be well-posed, and we have to test the validity of the corresponding conditions.

Nevertheless, ill-posed problems exist and have important practical applications. To solve them, additional considerations are used. Besides, an operator *A* and sets *U*, *W* are rearranged in such a way that a problem becomes well-posed.

We point out an important borderline case. Assume that (2) is fulfilled but $c[w] \gg 1$ (i.e., $c[w]$ is many orders greater than 1). Then a problem is formally well-posed but in fact very small variations δw may lead to large variations $\delta u.$ As a result, it becomes difficult to achieve a high accuracy. Such problems are said to be ill-conditioned.

1.1.2. Rigor of Studies

Algorithms do not always have an exhaustive mathematical justification. Usually in a proof some properties of an operator and sets are used (for example, *W* is assumed to be a set of functions that are continuous together with derivatives up to order *p*) or a numerical solution is assumed to belong to some neighborhood of an exact one. For concrete calculation these assumptions may be not satisfied. Then we have one of the following alternatives:

- a basic algorithm fails because of an invalid operation (division by zero, calculation of square root or logarithm of a negative number etc.);

- a basic algorithm works but a numerical solution does not converge to any limit;
- a numerical solution converges to an exact one but the rate of convergence is less than might be expected from theoretical results, moreover, auxiliary algorithms work improperly;
- a numerical solution converges to a limit being something other than an exact solution (this case may take place only for special problems with so-called generalized solutions).

Besides, most of the convergent results are valid only when all operations are performed exactly. In fact, calculations are performed with a restricted number of digits (~15 decimal digits for a 64-bit computer), moreover, the number of valid digits of input data is rather small. In addition, in a number of computer programs along with “pure” algorithms some special tricks are used (for example, a small constant is added to a denominator to avoid division by zero). As a rule, such tricks improve reliability of an algorithm but make its theoretical investigation more complicated.

It is risky to ignore these details as has been illustrated with the following example. It is known that Taylor expansion of the function $\sin x$ is absolutely convergent for any x. For $x > 0$ the series is alternating, thus, the error of a sum of a finite number of terms does not exceed the first truncated term. The summation of the series was performed for $x = 2550^\circ$ (without reduction to the first quadrant) with the use of a 64-bit computer. Calculation was terminated when the last term became equal to 10^{-8}. The computer output was an absurd result: $\sin 2550^\circ = 29.5!$ A close examination shows that this is due to round-off errors and for the given accuracy with this algorithm calculation must be performed with twice as many digits.

All the above is not a cause for despondency. This only reminds that serious numerical calculation requires as much attention as driving in a street with heavy traffic. One should not absolutely rely upon any method or any program. It is necessary to know their possibilities as well as weak points and evaluate reasonableness of obtained results. This is an art rather than a science; one can train in it on the basis of experience in practical calculation.

1.2. Sources of Error

1.2.1. Types of Data and Unknowns and Their Norms

Given data as well as a solution may be of various types: numbers u, w; vectors $\mathbf{u} = \left\{ u_p, 1 \le p \le P \right\}$, $\mathbf{w} = \left\{ w_q, 1 \le q \le Q \right\}$ of different dimension; matrices; functions $u(x), w(y)$ of one variable or of many variables; vector-functions etc. In addition, an argument (arguments) of a function may be continuous $a \le x \le b$ or discrete $x \in \Omega$, where $\Omega = \left\{ x_n, 1 \le n \le N \right\}$ is a grid.

We illustrate this with several examples.

- An equation in one unknown is solved; u, w are real or complex numbers.

- A system of N linear or nonlinear equations relative to N unknowns is solved; u, w are vectors of the same dimension N.
- A definite integral of $w(x)$ is calculated; $w(x)$ is a function of a continuous argument, u is a number.
- Spline-approximation of a function given in a tabulated form on a grid Ω is constructed; given data can be considered as a function $w(x_n)$ of a discrete argument or as a vector $\{w_n\}$; a solution $u(x)$ is a function of a continuous argument.
- A differential equation $du/dx = w(u,x)$ is solved; given data represent a continuous function $w(u,x)$ of two arguments; a solution is a continuous function $u(x)$ of one argument. However, for numerical integration a grid $\{x_n\}$ is introduced and a numerical solution appears to be a function $u(x_n)$ of a discrete argument.

A norm of an error is a quantitative measure of accuracy. We present some popular norms. For a number u there exists the single norm

$$\| u \| = | u | \ . \tag{3}$$

For a bounded function $u(x)$, $x \in [a, b]$ we can use the Chebyshev norm

$$\| u \|_C = \max_{x\in[a,b]} | u(x) | \ , \tag{4}$$

and for a square integrable function with a weight $\rho(x)$ – the Hilbert norm

$$\| u \|_{L_2} = \left[\int_a^b u^2(x)\,\rho(x)\,dx \right]^{1/2} , \qquad \rho(x) > 0. \tag{5}$$

For a function $u(x_n)$ of a discrete argument or for a vector $\{u_n\}$ discrete analogues of the norms (4)–(5) are as follows:

$$\| u \|_c = \max_{1\le n\le N} |u_n| , \qquad \| u \|_{l_2} = \left(\sum_{n=1}^{N} \rho_n\, u_n^2 \right)^{1/2} , \qquad \rho_n > 0. \tag{6}$$

For matrices several norms are used.

We see that for one object various norms may be used. These norms are related in a certain way. For a function of a continuous argument we have single inequalities, for example,

$$\| u \|_C \cdot \left[\int_a^b \rho(x)\, dx \right]^{1/2} \geq \| u \|_{L_2}. \qquad (7)$$

If a norm in the left-hand side is small, the norm in the right-hand side is small as well, but the opposite is not true; the similar statement is valid for convergence of methods in these norms. The first norm is said to be *stronger* than the second one. A clear distinction between the norms (4) and (5) is as follows: if the *C*-norm is small, then $u(x)$ is small at all points of [*a*, *b*]; if the L_2-norm is small, then $u(x)$ is small at almost all points except an insignificant part of [*a*, *b*] where $u(x)$ is not necessarily small.

For a function of a discrete (finite-dimensional) argument norms are related by double inequalities. For example, for (6) we have

$$\| u \|_c \cdot \left(\sum_{n=1}^{N} \rho_n \right)^{1/2} \geq \| u \|_{l_2} \geq \| u \|_c \cdot \sqrt{\min_{1 \leq n \leq N} \rho_n}. \qquad (8)$$

Hence, convergence in one norm implies convergence in another one. Such norms are said to be *equivalent*. In a finite-dimensional space all norms are equivalent, but for an infinite-dimensional space this is not the case.

Three sources of an error of a numerical solution are distinguished: an error of given data, an error of a method, and a round-off error. Consider them in detail.

-
-
-

Bibliography

De Boor C. (1978). A Practical Guide to Splines. Applied Mathematical Sciences, v.27. Springer-Verlag, Berlin-Heidelberg-NewYork. [The theory and algorithms for the construction of polynomial splines are presented in a simple form accessible to a practical specialist in computation. Many FORTRAN programs are cited.]

Golub G.H. and van Loan Ch.F. (1989). Matrix Computation. John Hopkins Univ. Press, Baltimore and London, 2nd ed. [Methods of matrix computation, direct and iterative methods for solving linear systems, algorithms for an eigenvalue problem, applications to the method of least squares are considered. FORTRAN programs are presented.]

Hairer E., Lubuch Ch., and Wanner G. (2002). Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations, 483 pp. Springer-Verlag, Berlin-Heidelberg-New York.

[Up-to-date methods for numerical integration of ODE's developed during the last 10-15 years are presented.]

Hairer E., Norsett S.P., and Wanner G. (2001). Solving Ordinary Differential Equations I. Nonstiff Problems, 2nd ed., 512 pp. Springer-Verlag. Berlin-Heidelberg-New York. [In this book the basic theory of modern numerical methods for ODE's is presented, many examples of practical applications of these methods and FORTRAN programs are cited.]

Hairer E. and Wanner G. (2002). Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, 2nd ed., 632 pp. Springer-Verlag, Berlin-Heidelberg-New York. [The modern theory of numerical methods for ODE's which are applied when solving stiff and differential-algebraic problems is outlined. Many examples and descriptions of computer programs are presented.]

Kalitkin N.N. (1978). Numerical Methods, 512 pp. Nauka, Moscow (in Russian). [In this book numerical methods are considered in detail, from the simplest problems to the solution of partial differential equations. The book is intended for specialists in engineering sciences and physics.]

Kamke E. (1959). Differentialgleichungen: Lösungen. I. Gewöhnliche Differentialgleichungen, 6th ed., 642 pp. Becker & Erler, Leipzig [An indispensable reference book of differential equations whose exact solution are known.]

Marchuk G.I. (1989). Methods of Computational Mathematics, 3d ed. Nauka, Moscow (in Russian). [In this book on numerical methods a wide range of problems, from the simplest ones to partial differential equations, is considered. The book is intended for mathematicians who wish to improve their qualification.]

Marchuk G.I. and Shaidurov V.V. (1970). Difference Schemes of Higher-order Accuracy. Nauka, Moscow (in Russian) [In this book the method of grid refinement is given in detail. Improvement of an accuracy of a solution and the construction of *a posteriori* asymptotic error estimates with the help of this method are considered.]

Nikolskii S.M. (1974). Quadrature Formulae, Nauka, Moscow (in Russian) [The principles of the construction of quadrature formulae are presented and the most important formulae are cited. Error estimates for functions of different smoothness are obtained.]

Ortega J.M. and Rheinboldt W.C. (1970). Iterative Solution of Nonlinear Equations in Several Variables. Academic Press, New York.

Wilkinson J.H. (1965). The Algebraic Eigenvalue Problem. 662 pp. Claredon Press, Oxford. [This book is an encyclopaedia on numerical methods for problems of linear algebra.]

Wilkinson J.H. and Reinsch C. (1970). Linear Algebra. 439 pp. Grundlehren band 186, Springer-Verlag. [A reference book of algorithms for problems of linear algebra with ALGOL programs.]

Biographical Sketches

Nikolai N. Kalitkin is a corresponding member of the Russian Academy of Sciences, the Head of a department at the Institute for Mathematical Modeling of the Russian Academy of Sciences, Professor of the Physics Department at the Moscow State University, the Head of the Chair of Mathematical Modeling at the Moscow Institute of Electronic Engineering. His scientific interests are focussed on the construction of mathematical models for various problems in physics and engineering and in the development of efficient numerical methods for their solution. N.N. Kalitkin proposed the stiff method of lines for problems involving many processes which are reduced to systems of partial differential equations (for example, a flow of gases which enter into chemical reactions); developed the method of quasiuniform grids for various problems, among them problems in an unbounded domain; constructed the method of complemented vector for eccentially nonlinear eigenvalue problems; proposed new methods for approximation of functions which admit extrapolation.

Sergei S. Filippov is Candidate in Physics and Mathematics, a Senior Researcher at the Keldysh Institute of Applied Mathematics of the Russian Academy of Sciences, Professor of the Chair of Applied Mathematics at the Moscow Institute of Physics and Technology. His scientific interests are focussed on the construction of mathematical models for applied problems in physics and engineering and the development of numerical methods for their solution. S.S. Filippov developed a number of numerical methods for ordinary and partial differential equations.