JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

Journal of Computational and Applied Mathematics 138 (2002) 325–333

www.elsevier.com/locate/cam

Locating all the zeros of an analytic function in one complex variable

Michael Dellnitz[a,*], Oliver Schütze[a], Qinghua Zheng[b]

[a]Department of Mathematics and Computer Science, University of Paderborn, D-33095, Paderborn, Germany

[b]Corporate Technology, Siemens AG München, D-81730 München, Germany

Received 17 October 2000

Abstract

Based on the *argument principle*, we propose an adaptive multilevel subdivision algorithm for the computation of all the zeros of an analytic function $f:\mathbb{C}\to\mathbb{C}$ within a bounded domain. We illustrate the reliability of this method by several numerical examples.

MSC: 30C15; 65E05

Keywords: Global zero finding; Argument principle

1. Introduction

Dynamical systems are frequently used as models for the temporal evolution of technical processes. In this case, the stability analysis for an equilibrium state of the system naturally leads to the problem of finding all the zeros—or at least bounds on the set of zeros—of a certain analytic function $f:\mathbb{C}\to\mathbb{C}$. For instance, if there is no temporal delay to be taken into account in the underlying system then one has to consider the spectrum of the Jacobian of a vector field evaluated at the equilibrium. This can e.g., be done by using efficient eigenvalue solvers. However, if the technical process has to be modeled by a delay differential equation then the zero set of a non-polynomial but holomorphic function has to be approximated.

Motivated by this application, we propose in this paper a robust numerical method for the computation of all the zeros of a holomorphic function $f:\mathbb{C}\to\mathbb{C}$ within a certain bounded domain. The underlying idea is to use numerically the *argument principle* and to combine this with an adaptive multilevel subdivision strategy. This way, we construct tight box coverings of the set of zeros of f.

* Corresponding author. http://math-www.uni-paderborn.de/ ~agdellnitz

0377-0427/02/$ - see front matter
PII: S0377-0427(01)00371-5

The reliability of the method is due to the fact that by the argument principle one has to check whether or not a certain number is a non-vanishing integer, and this test is typically quite robust.

The underlying idea of this approach, namely to use the argument principle in combination with a subdivision procedure, already occurs in [4,5]. However, we go one step further and propose an adaptive subdivision technique. Moreover, in contrast to [4,5] we also describe in detail a specific numerical realization. This approach is similar in spirit to the topological methods developed in [3] where degree theory is the underlying concept for the approximation of the set of zeros of a given function. In fact, the argument principle as used here has to be viewed as the specific holomorphic situation for the computation of the degree of a mapping (cf. [1]).

A detailed outline of the article is as follows. In Section 2, we briefly restate the argument principle. Then, in Section 3, we develop our adaptive multilevel algorithm, prove its convergence and discuss a particular numerical realization. Finally, we present three examples indicating the robustness and efficiency of this numerical approach (Section 4).

2. Theoretical background

Our numerical approach is based on an elementary fact from complex analysis namely the so-called "argument principle". For a given meromorphic function f, this result states that the number of zeros minus the number of singularities of f (counting multiplicities) within a specified region in $\mathbb{C}$ can in principle be computed by a certain integration:

Theorem 2.1 (Argument principle). *Let $f: U \to \mathbb{C}$ be a meromorphic non-constant function on the open subset $U \subset \mathbb{C}$ and let γ be a closed curve on the boundary of a compact set K inside U. Finally denote by q_j, $j = 1, \dots, n$, and by p_ℓ, $\ell = 1, \dots, m$, the zeros resp. the singular points of f inside K. Then*

$$\frac{1}{2\pi \mathrm{i}} \int_\gamma \frac{f'(z)}{f(z)}\,\mathrm{d}z = \sum_{j=1}^{n} n(\gamma, q_j)\mu(q_j) - \sum_{\ell=1}^{m} n(\gamma, p_\ell)\mu(p_\ell),$$

where $n(\gamma, q_j)$, $n(\gamma, p_\ell)$ denote the winding numbers of the curve γ with respect to q_j resp. p_ℓ and $\mu(q_j)$, $\mu(p_\ell)$ are the multiplicities of q_j resp. p_ℓ.

In our applications, we are particularly interested in the situation where f is a holomorphic function and where the winding number of γ is one. For this specific case, we immediately obtain

Corollary 2.2. *Let $f: U \to \mathbb{C}$ be a holomorphic non-constant function on the open subset $U \subset \mathbb{C}$ and let γ be a closed curve on the boundary of a compact set K inside U with winding number one for all the points surrounded by γ. Finally, denote by q_j, $j = 1, \dots, n$, the zeros of f inside K. Then*

$$\int_\gamma \frac{f'(z)}{f(z)}\,\mathrm{d}z = 2\pi \mathrm{i} \sum_{j=1}^{n} \mu(q_j), \tag{2.1}$$

where $\mu(q_j)$ are the multiplicities of the zeros q_j.

In the following, we will denote by $\mu(f, \gamma)$ the right hand side in (2.1).

3. Algorithmic description

Our aim is to find all the zeros of an analytic function f in one complex variable inside a given compact domain. The underlying idea of the algorithm is as follows. We start with a (big) rectangle B in $\mathbb{C}$ inside which we would like to find all the zeros of f. Then we compute the integral in the left hand side in (2.1), where γ is a parametrization of the boundary of the rectangle B with winding number one. If this integral is zero then there are no zeros inside B and we are done. Otherwise we subdivide B into smaller rectangles and compute the integrals in (2.1) over their boundaries. Proceeding this way and disregarding rectangles for which the integral is zero we obtain a close covering of all the zeros of f inside B.

3.1. Description of a basic subdivision scheme

We now formalize the idea stated above and present the abstract multilevel subdivision procedure in detail. The general structure of the following abstract algorithmic scheme is essentially known and can be found in the literature (see [4,5]).

3.1.1. Basic subdivision scheme

Let $\mathscr{B}_0$ be an initial collection of finitely many rectangles in $\mathbb{C}$. Then $\mathscr{B}_k$ is inductively obtained from $\mathscr{B}_{k-1}$ in two steps:

(i) *Subdivision step*: Construct from $\mathscr{B}_{k-1}$ a new system $\hat{\mathscr{B}}_k$ of rectangles such that

$$\bigcup_{B\in\hat{\mathscr{B}}_k} B = \bigcup_{B\in\mathscr{B}_{k-1}} B$$

and

$$\operatorname{diam}(\hat{\mathscr{B}}_k) = \theta_k \operatorname{diam}(\mathscr{B}_{k-1}), \tag{3.1}$$

where $0 < \theta_{\min} \leqslant \theta_k \leqslant \theta_{\max} < 1$. (Here $\operatorname{diam}(\mathscr{B}_k)$ denotes the diameter of the largest rectangle inside $\mathscr{B}_k$.)

(ii) *Selection step*: Define the new collection $\mathscr{B}_k$ by

$$\mathscr{B}_k = \{B \in \hat{\mathscr{B}}_k\colon \mu(f,\gamma_B) \neq 0\}.$$

Remark 3.1. (a) If at some stage in the subdivision process a zero of f is lying on the boundary of one of the rectangles B then $\mu(f,\gamma_B) = \infty$. We explicitly allow this possibility in the selection step and keep the corresponding rectangle in this case.

(b) Observe that we do not require that the rectangles inside the collection $\mathscr{B}_k$ are disjoint. However, in our modification and realization of the basic subdivision scheme we are always working with collections having this property in order to avoid unnecessary numerical computations.

Denote by $z(f,B)$ the set of zeros of the holomorphic function f inside the rectangle B and let

$$Z_k = \bigcup_{B\in\mathscr{B}_k} B.$$

Observe that $\lim_{k\to\infty} Z_k$ does exist since the Z_k form a nested sequence of compact sets. Moreover, it is immediate from the construction of the general algorithm that the following result holds.

Proposition 3.2. *An application of the basic subdivision scheme to the rectangle $B=\mathscr{B}_0$ yields a sequence of collections $\mathscr{B}_k$ such that*

$$\lim_{k\to\infty} h(Z_k, z(f,B)) = 0, \tag{3.2}$$

where $h(\cdot,\cdot)$ denotes the standard Hausdorff distance.

3.2. Adaptive version of the basic subdivision scheme: the QZ-40 algorithm

We now present a modification of the basic subdivision scheme which will lead to a robust and efficient numerical realization. The crucial difference consists essentially of two ingredients: first an adaptive subdivision strategy is used and second we introduce an additional search step. The structure of the following algorithm is significantly different to that of the basic subdivision scheme. Therefore—in order to avoid any potential confusion—we will from now on use the letter R instead of B in the notation for rectangles or their collections, respectively.

3.2.1. The QZ-40 algorithm

Let $\mathscr{R}_0$ be an initial collection of finitely many rectangles in $\mathbb{C}$. Then $\mathscr{R}_k$ is inductively obtained from $\mathscr{R}_{k-1}$ in three steps:

(i) *Selection step*: For every rectangle $R\in\mathscr{R}_{k-1}$ denote by γ_R a parametrization of the boundary of the rectangle and compute the winding number $\mu(f,\gamma_R)$. Remove all rectangles R from $\mathscr{R}_{k-1}$ for which $\mu(f,\gamma_R)=0$.

(ii) *Search step*: Search for a zero inside each rectangle $R\in\mathscr{R}_{k-1}$ with $\mu(f,\gamma_R)=2\pi\mathrm{i}$ using Newton's method with a starting point inside R. If a zero is found then store this point and remove the rectangle R from the collection $\mathscr{R}_{k-1}$.

(iii) *Adaptive subdivision step*: Construct from $\mathscr{R}_{k-1}$ a new system $\hat{\mathscr{R}}_k$ of rectangles according to a certain specified subdivision strategy satisfying (3.1). Then additionally subdivide each rectangle inside $\hat{\mathscr{R}}_k$ which is a subset of a rectangle $R\in\mathscr{R}_{k-1}$ with the property that $\mu(f,\gamma_R)/2\pi\mathrm{i}>2$. Let $\mathscr{R}_k$ be the resulting collection of boxes.

As in the case of the basic subdivision scheme the QZ-40 algorithm produces a nested sequence of compact sets Z_k covering the (remaining) zeros of f. Since the diameter of the boxes is shrinking to zero, we have the following result:

Proposition 3.3. *Suppose that all the zeros of f inside the rectangle R are simple. Then the QZ-40 algorithm applied to the rectangle $R=\mathscr{R}_0$ terminates after finitely many steps returning a complete list of all the zeros of f inside R.*

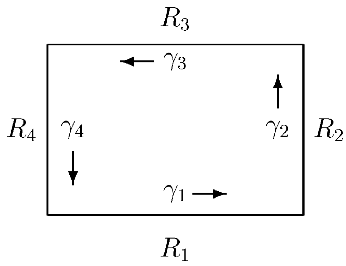

Fig. 1. Boundary of the rectangle R and its parametrization $\gamma_R = \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4$.

Remark 3.4. (a) It is an easy task to modify the algorithm in such a way that also zeros of higher multiplicity can be found.

(b) One has to expect that in general in the first subdivision steps the integers $\mu(f,\gamma_R)/2\pi\mathrm{i}$ will be greater than two for almost every rectangle within the selected box collections. From this point of view, it would seem natural to modify the subdivision step (iii) in such a way that additional subdivisions are just considered from a certain subdivision level $k > 1$ on. We have performed several numerical experiments along these lines, and these indicated that—somewhat counter-intuitively—the most efficient strategy is indeed to consider additional subdivisions right from the beginning.

3.3. Numerical realization

A crucial part in the numerical realization of the QZ-40 algorithm is the computation of integral (2.1) for the different rectangles. In our current implementation, the computation of $\mu(f,\gamma_R)$ is done via an adaptive Romberg quadrature as described in [2] along the four different edges of the rectangle. More precisely, we perform the three steps of the QZ-40 algorithm as follows:

- In a first step, we parametrize each of the edges R_j of a rectangle by γ_j $(j=1,2,3,4)$ and compute the value of

$$I_j \approx \Re\left(\frac{1}{2\pi\mathrm{i}}\int_{\gamma_j}\frac{f'(z)}{f(z)}\,\mathrm{d}z\right) \qquad (j=1,2,3,4)$$

  via an adaptive Romberg quadrature, see Fig. 1. Then

$$\mu(f,\gamma_R) \approx 2\pi\mathrm{i}\sum_{j=1}^{4} I_j.$$

  Observe that the value of the tolerance in the adaptive Romberg scheme can be chosen quite large since we only have to decide whether or not the approximation of $\sum_{j=1}^{4} I_j$ is zero or a non-vanishing integer.
- In the search step, we perform a search for a zero of f using the classical Newton method. This method either terminates with a zero inside the rectangle under consideration or the iteration is stopped if an iterate is more than a specified distance away from the center of the rectangle. In the examples which we considered it turned out to be quite efficient to choose at random a few (five, say) starting points for Newton's method inside the rectangle.

Table 1
Performance of the different zero finding procedures

Method	# initial points	# function calls	# derivative calls	# zeros
Classical Newton	1.500	140.130	140.130	153
	5.000	470.038	470.038	322
	7.000	655.879	655.879	361
	10.000	938.136	938.136	403
	15.000	1405.903	1405.903	419
NAG `c05pbc`	1.500	454.856	6.966	177
	5.000	735.214	23.672	292
	7.000	1033.345	32.209	336
	10.000	1481.284	47.653	378
	15.000	2238.868	71.276	406
QZ-40	—	89.619	89.619	424

- The subdivision rule used in the adaptive subdivision step is simply given by bisection alternating between the two different coordinate directions. That is, if a rectangle is created via bisection with respect to the z_1-direction then, if necessary, it will be bisected in the next step with respect to the z_2-direction and vice versa. Obviously, by this rule condition (3.1) is satisfied.

4. Numerical examples

The following examples indicate both the efficiency and in particular the robustness of the QZ-40 algorithm.

4.1. Academic example

We consider the function

$$f_1 : \mathbb{C} \to \mathbb{C}, \quad f_1(z) = z^{50} + z^{12} - 5\sin(20z)\cos(12z) - 1,$$

and compute its zeros inside the rectangle $[-20.3, 20.7] \times [-20.3, 20.7]$. A couple of coverings of the zeros of f_1 produced by the QZ-40 algorithm are shown in Fig. 2. These computations indicate that there are precisely 424 zeros of f inside the initial rectangle.

We have compared the numerical effort of the QZ-40 algorithm with the one obtained by two alternative approaches namely

- the classical Newton method performing at most 100 iterations with a certain number of initial points chosen at random. The numbers in Table 1 are averages over 20 different computations.
- the zero finding procedure `c05pbc` of the NAG library with a certain number of initial points chosen at random. In these computations, we have specified a tolerance of 1e-15.

It can be observed, see Table 1, that the QZ-40 algorithm is much more efficient than these two approaches.

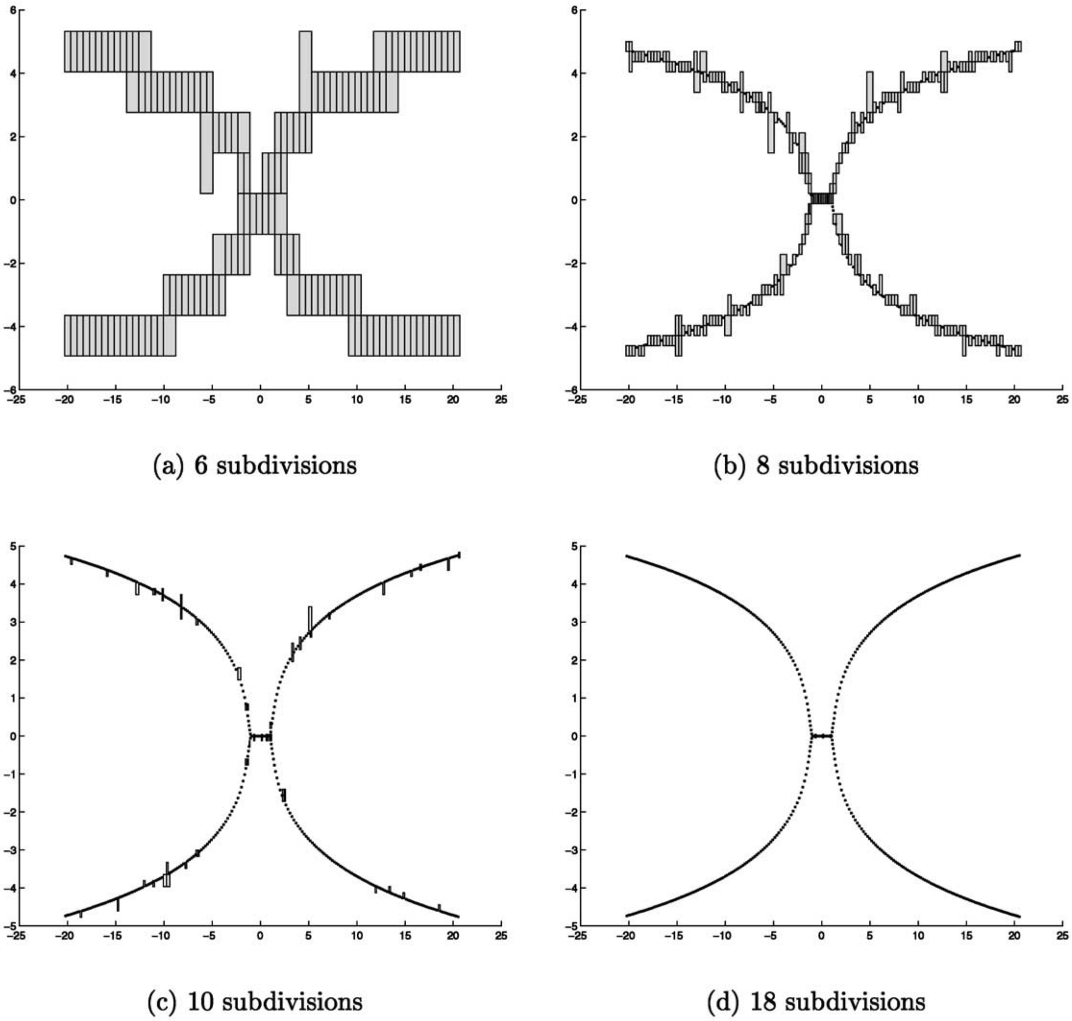

Fig. 2. Box coverings of the set of zeros of f_1 obtained by the QZ-40 algorithm.

4.2. Stability of an annular combustion chamber

Within a project at the Corporate Technology Department of Siemens (Munich), the stability of a flow inside an annular combustion chamber had to be analyzed. In a simplified model this lead to the problem of finding all the zeros of the following holomorphic function:

$$f_2 : \mathbb{C} \to \mathbb{C}, \quad f_2(z) = z^2 + Az + Be^{-Tz} + C,$$

where A, B, C and T are real parameters. Motivated by the actual underlying model for the combustion chamber, we choose the values

$$A = -0.19435, \quad B = 1000.41, \quad C = 522463 \quad \text{and} \quad T = 0.005.$$

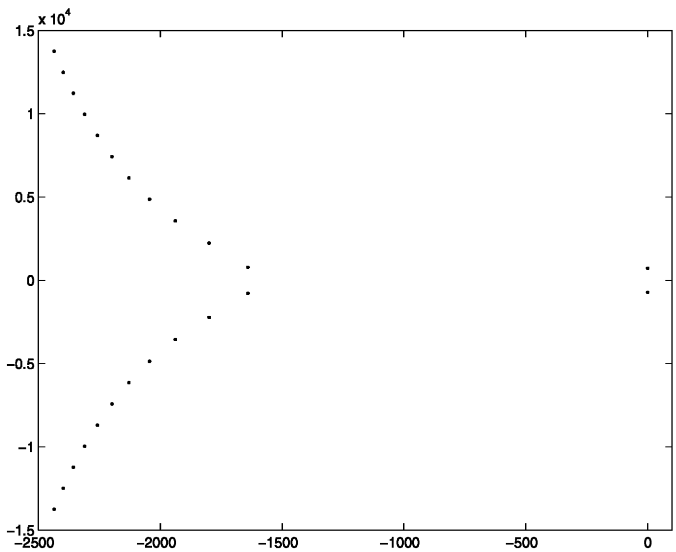

Fig. 3. All the roots of f_2 inside $R = [-15000, 5000] \times [-15000, 15000]$.

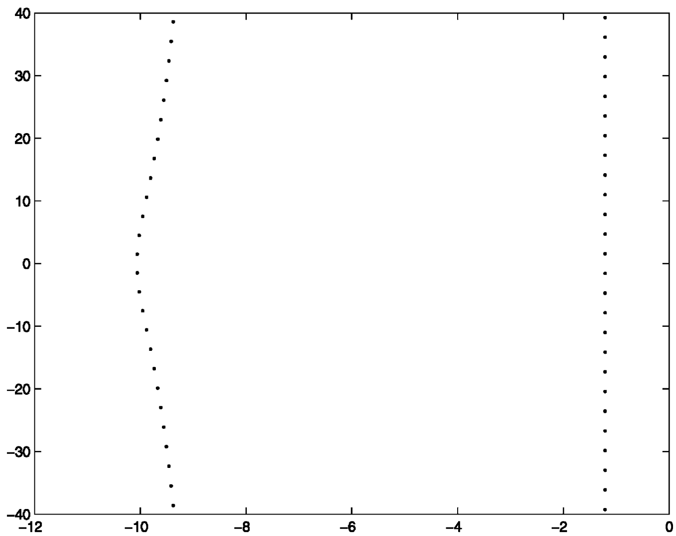

Fig. 4. All the roots of f_3 within the rectangle $R = [-12, 0] \times [-40, 40]$.

In Fig. 3, all the roots of f_2 found by the QZ-40 algorithm inside the rectangle $R=[-15000, 5000]\times[-15000, 15000]$ are shown.

4.3. Stability of a ring oscillator

In [6], the detection of Hopf bifurcations in a ring consisting of coupled oscillators has been investigated. Taking the symmetry of the problem into account—i.e., by restricting to the bifurcation of the so-called *discrete rotating waves*—one has to find all the zeros of the following function:

$$f_3 : \mathbb{C} \to \mathbb{C},$$

$$f_3(\lambda) = \det \begin{pmatrix} -0.0166689 - 2.12 \times 10^{-14}\lambda & \frac{1}{60} + 6.0 \times 10^{-16}\mathrm{e}^{-\tau\lambda}\lambda \\ 0.0166659 + \mathrm{e}^{-\tau\lambda}(-0.000037485 + 6.0 \times 10^{-16}\lambda) & -0.0166667 - 6.0 \times 10^{-16}\lambda \end{pmatrix},$$

where we have chosen the delay to be $\tau = 2.0$. The result is shown in Fig. 4. For more detailed information concerning this problem we refer to [6].

References

[1] K. Deimling, Nichtlineare Gleichungen und Abbildungsgrad, Springer, Berlin, 1974.

[2] P. Deuflhard, A. Hohmann, Numerische Mathematik, De Gruyter, Berlin, 1991.

[3] H.-O. Peitgen, M. Prüfer, The Leray–Schauder continuation method is a constructive element in the numerical study of nonlinear eigenvalue and bifurcation problems, in: H.-O. Peitgen, H. O. Walther (Eds.), Functional Differential Equations and Approximation of Fixed Points, Vol. 730 of Lecture Notes Math., 1979.

[4] X. Ying, N. Katz, A reliable argument principle algorithm to find the number of zeros of an analytic function in a bounded domain, Numer. Math. 53 (1988) 143–163.

[5] X. Ying, N. Katz, A simple reliable solver for all the roots of a nonlinear function in a given domain, Computing 41 (1989) 317–333.

[6] Q. Zheng, M. Dellnitz, Schwingungen eines Ringoszillators—eine numerische Behandlung unter Berücksichtigung der Symmetrie, ZAMM 70 (1990) T135–T138.