

Application of numerical continuation to compute all solutions of semilinear elliptic equations

Eugene Allgower, Stefan-Gicu Cruceanu and Simon Tavener

(Communicated by A. Sommesse)

Abstract. We adapt numerical continuation methods to compute *all* solutions of finite difference discretizations of nonlinear boundary value problems involving the Laplacian in two dimensions. New solutions on finer meshes are obtained from solutions on coarser meshes using a complex homotopy deformation. Two difficulties arise. First, the number of solutions typically grows with the number of mesh points and some form of filtering becomes necessary. Secondly, bifurcations may occur along homotopy paths of solutions and efficient methods to swap branches are developed when the mappings are analytic. For polynomial nonlinearities we generalize an earlier strategy for finding all solutions of two-point boundary value problems in one dimension and then introduce exclusion algorithms to extend the method to general nonlinearities.

1 Introduction

This article addresses the problem of numerically approximating *all* the solutions of a class of nonlinear second order semilinear elliptic boundary value problems with Dirichlet boundary conditions on a rectangular domain. It has been noted, see e.g. [6], that there are very few theoretical results concerning how many solutions such a problem may have, or indeed, if there are any solutions at all. This state of affairs is somewhat better in the corresponding case of second order ordinary differential equations, where a number of existence and multiplicity results are available in several papers and books, see, e.g., the references in [3]. The approach taken here is to perform a standard finite difference approximation of the partial derivatives and then to numerically seek all of the solutions to the resulting nonlinear systems of equations. This approach worked very successfully for ordinary differential equations when the nonlinearities were of polynomial type [3]. The example considered in [3] was

$$u'' = f(x, u, u')$$

with general linear boundary conditions and f assumed to be a polynomial map in the u -variable. For this case, the totality of solutions of the polynomial systems of equations was

found by embedding the systems in a complex setting and applying a numerical homotopy continuation method to compute all of the complex solutions of the polynomial system.

There are some drawbacks however, which need to be addressed. One of them is that the requirement for an accurate discretization can result in a very large polynomial system and therefore an enormous number of complex solutions, among which are only a few real solutions that are actually of interest. To illustrate this point, let us mention the familiar theorem of Bezout [13, 8, 14], which essentially states that the number of (finite) complex solutions, accounting for multiplicities, can be as much as the product of the degrees of each of the equations. So, for example, N quadratic equations would generally have 2^N solutions in \mathbb{C}^N . Furthermore, even among the real solutions, there may be spurious solutions which arise as numerical artifacts and do not converge to solutions of the boundary value problem.

The above drawbacks were treated in [3] for two point boundary value problems by means of two remedial steps. The first step was to discard the obviously irrelevant solutions, including the real solutions which did not exhibit properties which theoretical results showed must hold, for example, symmetry properties. The second remedy was to start with a crude mesh (and hence a low dimensional system) and then to introduce a new mesh point via a continuous deformation of the mesh. Assuming that the solutions had been obtained for a uniform mesh with, say N points, a new point was introduced, for example, at the right boundary and this point was then allowed to be continuously moved leftward until a uniform mesh with $(N + 1)$ points was achieved. This device was suggested in [2] and implemented in [3]. It uses numerical continuation in yet another way since the homotopy parameter now is used to continuously deform the mesh in the difference equations. Starting points for solutions when the new point is introduced are simply the zero points of a single polynomial equation, which are generally easy to find. For example, one may apply an algorithm for finding the eigenvalues of the corresponding companion matrix.

In essence, it would seem straightforward to extend the ideas used in [3] to a corresponding case of partial differential equations in two dimensions. Let us take as our paradigm the equation

$$\begin{aligned}\Delta u &= f(x, y, u, u_x, u_y) \quad \text{on } \Omega \subset \mathbb{R}^2, \\ u &= g \quad \text{on } \partial\Omega,\end{aligned}$$

where f is assumed to be an analytic map. One new issue which needs to be confronted is that of introducing mesh points in a manner which does not require a large number of new points, due to the fact that the number of solutions would grow too rapidly. Such a moving mesh is described in Section 3.4. The approach proceeds as follows. Suppose that the solutions have been found for a uniform square mesh in the square domain. In a fashion as described in the previous paragraph, a new point is introduced at the right boundary of the first row of mesh points and is allowed to move leftward until the points in the first row are again equally spaced. The same process is repeated for each of the rows and then for each of the columns, this time by introducing a point at the top of each column until once again a uniform square mesh is attained. The entire procedure requires a careful handling of the underlying stiffness matrices for the moving meshes. Now however, the

resulting meshes do not generally exhibit symmetries even at termination of introducing the moving mesh-point. Symmetry conditions for discarding spurious solutions need to be correspondingly modified.

Bifurcation can occur along the homotopy paths. For the class of problems being considered here, the homotopy paths are the solutions of a system of equations of the form $H(t, z) = 0$ where $H : [0, 1] \times \mathbb{C}^n \rightarrow \mathbb{C}^n$ is analytic in the z -variables. We have shown in [4] that in the above analytic setting any turning point of a homotopy path is necessarily also a bifurcation point and that the bifurcating directions are orthogonal. Thus the tangent vectors at the bifurcation point are orthogonal and switching solution paths on to the new bifurcating branch at such points is easily accomplished.

McKenna and Plum [6] study a particular boundary value problem involving a quadratic nonlinearity on a square domain and prove that at least four solutions exist for a particular value of a parameter arising in the equation. They conjecture the form of the bifurcation diagram and suggest that more than four solutions may occur as the parameter is increased. We apply our techniques to this problem in Section 4.3, where we confirm the bifurcation diagram and demonstrate that no more than four solutions occur over a large positive range of the parameter values.

In Section 5, a generalization is made which allows non-polynomial nonlinearities in the partial differential equation. In the polynomial case when a new mesh point was introduced, the starting solutions were obtained as the eigenvalues of the corresponding companion matrix. For non-polynomial nonlinearities there may be infinitely many complex solutions and it is necessary to seek solutions in a bounded region which is taken to be a rectangle in \mathbb{C} . The real solutions are then found by means of a cellular exclusion algorithm. Cellular exclusion algorithms have recently been studied by several authors, see e.g. [9] and the references there. With cellular exclusion methods, all real solutions to nonlinear systems of equations within a rectangular n -cell can be found, provided the nonlinearity satisfies some rather mild conditions, such as Lipschitz conditions on the components. On the other hand, if the nonlinearity has a Taylor expansion, the efficiency of the exclusion algorithm can be significantly improved. This approach is applied to the familiar Bratu equation in one dimension.

The present study has been restricted to a special, but familiar class of boundary value problems mainly to illustrate the effectiveness of the techniques presented. It should be emphasized that the methods presented here are not meant to provide fast and accurate methods for solving partial differential equations. Rather, the idea is to glean reliable information concerning the number and qualitative properties of solutions. The approximations which are obtained may however be used as starting values to obtain more accurate solutions on much finer meshes.

2 Homotopy continuation for ODEs

In this section we briefly review the approach used in [3] for numerically finding all solutions to two-point boundary value problems. Consider the second order boundary

value problem on the interval $[a, b] \subset \mathbb{R}$

$$u'' = f(x, u, u'), \quad (1)$$

with the boundary conditions $u(a) = \alpha$ and $u(b) = \beta$. Using a central difference approximation with a uniform mesh for example, we can approximate a solution $u(x)$ of (1) by an N -tuple of numbers $(u_1, u_2, \dots, u_N)^T$ such that $u_i \approx u(x_i)$ for $i = 1, \dots, N$, where we set $h := \frac{b-a}{N+1}$, $x_i := a + ih$ for $i = 0, \dots, N+1$, $u_0 = \alpha$, and $u_{N+1} = \beta$. The discretization of (1) takes the form of the following system \mathcal{D}_N ,

$$\mathcal{D}_N \begin{cases} u_0 - 2u_1 + u_2 = h^2 f(x_1, u_1, \frac{u_2 - u_0}{2h}), \\ \vdots \\ u_{N-1} - 2u_N + u_{N+1} = h^2 f(x_N, u_N, \frac{u_{N+1} - u_{N-1}}{2h}). \end{cases}$$

We are seeking all the real solutions of (1) and we know that depending upon the right hand side f , Equation (1) may have no solution, a unique solution, multiple solutions, or even infinitely many solutions. There are several theorems that state sufficient conditions for the existence of solutions for such equations, but even when the existence is known, the number of solutions is often not. In [3], the authors studied a relatively secure numerical technique for finding all the solutions for such an equation in the case that $f(x, u, u')$ is a polynomial depending on u . The idea is to complexify the problem and find all the solutions (real and complex) of an associated polynomial system $P_N(z)$ using a homotopy function which will track some known starting solutions to all the solutions of $P_N(z)$. A drawback of this approach is that the number of solutions of $P_N(z)$ can be quite large (as is known from Bezout's theorem) and hence sifting out solutions which are irrelevant to (1) becomes a significant issue.

The process of finding the solutions can be sketched in four steps.

1. Find all the solutions of the discretization \mathcal{D}_N for some small N .
2. Discard all unreasonable solutions and denote by \mathcal{S}_N the set of the solutions which are kept.
3. If the mesh size is not sufficiently small or the cardinality of \mathcal{S}_N has not yet stabilized, then add a mesh point to obtain the discretization \mathcal{D}_{N+1} . Use the solutions in \mathcal{S}_N to generate solutions of \mathcal{D}_{N+1} and then return to step 2 to generate \mathcal{S}_{N+1} .
4. Once the mesh size is sufficiently small and the cardinality of \mathcal{S}_N becomes stable, refine the solutions to a more consistent grid with a fast nonlinear solver.

As one can imagine, the key step in this process is step 3. To solve this, we build a homotopy function such as the following, which produces a mesh refinement as a result of a continuous deformation.

Let $H_{N+1}(u_1, u_2, \dots, u_{N+1}, t) :=$

$$\begin{pmatrix} u_0 - 2u_1 + u_2 & - & h(t)^2 f(x_1(t), u_1, \frac{u_2 - u_0}{2h(t)}) \\ & \vdots & \\ u_{N-2} - 2u_{N-1} + u_N & - & h(t)^2 f(x_{N-1}(t), u_{N-1}, \frac{u_N - u_{N-2}}{2h(t)}) \\ u_{N-1} - 2u_N + u_{N+1}(t) & - & h(t)^2 f(x_N(t), u_N, \frac{u_{N+1}(t) - u_{N-1}}{2h(t)}) \\ u_N - 2u_{N+1} + u_{N+2}(t) & - & h(t)^2 f(x_{N+1}(t), u_{N+1}, \frac{u_{N+2}(t) - u_N}{2h(t)}) \end{pmatrix} \quad (2)$$

with

$$\left\{ \begin{array}{l} x_i(t) = a + ih(t), \quad \text{for } i = 1, \dots, N + 1, \\ u_0 = \alpha, \\ h(t) = t\left(\frac{b-a}{N+1}\right) + (1-t)\left(\frac{b-a}{N+2}\right), \\ U_{N+1}(t) = (1-t)u_{N+1} + \beta t, \\ U_{N+2}(t) = \beta(1-t). \end{array} \right. \quad (3)$$

- Remark 2.1.** (a) At $t = 0$, H_{N+1} represents the system \mathcal{D}_{N+1} .
 (b) At $t = 1$, H_{N+1} can be interpreted as the system \mathcal{D}_N with a new mesh point having the value u_{N+1} at $x_{N+1} = b$ and a new right-hand boundary having the value $U_{N+2}(1) = 0$ at $x_{N+2} = b + h(1)$.
 (c) There is an incompatibility between the old boundary condition at $x = b$ and the new one at $x = b + h(1)$, but this is accommodated by the presence of both u_{N+1} and U_{N+1} , which are not necessarily equal. As t goes from 1 to 0, the mesh points are squeezed back inside of $[a, b]$ and right hand boundary condition $u(b) = \beta$ is transferred from U_{N+1} to U_{N+2} as U_{N+1} is enforced to equal u_{N+1} , i.e.,

$$U_{N+2}(0) = \beta \quad \text{and} \quad U_{N+1}(0) = u_{N+1}.$$

To find all the solutions of \mathcal{D}_{N+1} , we will use numerical continuation to track the zeros of H_{N+1} as t goes from 1 to 0. At $t = 1$, we have a list \mathcal{S}_N of solutions $(u_1, u_2, \dots, u_N)^T$ satisfying the first N equations of H_{N+1} , while the final equation is

$$u_N - 2u_{N+1} = h(1)^2 f\left(b, u_{N+1}, \frac{-u_N}{2h(1)}\right), \quad (4)$$

which is the only place where u_{N+1} appears. For each solution $(u_1, u_2, \dots, u_N)^T$ in \mathcal{S}_N , we use this equation to find the corresponding values of u_{N+1} . These are the starting points of continuation paths leading to solutions of \mathcal{D}_{N+1} .

Remark 2.2. This framework will not change in any of its essentials if we prescribe in (3) a different function for U_{N+2} (for example, the constant function $U_{N+2} = \beta$). *The essential feature of U_{N+2} is that it goes to β as t goes from 1 to 0.*

Remark 2.3. In the case of polynomial nonlinearity when $f(x, u, u')$ from (1) is a real polynomial $p(u)$, we can conveniently obtain the starting points for $H_{N+1}(u_1, u_2, \dots, u_{N+1}, 1) = 0$ by solving the polynomial equation

$$u_N - 2u_{N+1} + U_{N+2}(1) - \left(\frac{b-a}{N+1}\right)^2 p(u_{N+1}) = 0, \quad (5)$$

for u_{N+1} given u_N from the solutions in \mathcal{S}_N . All the solutions (real and complex) of (5) can be found using standard available software. For small degree polynomials, the companion matrix may be used to find these zeros.

If we denote $d = \deg p(u)$, then one can see that over the complex numbers we obtain d values of u_{N+1} for every solution in \mathcal{S}_N . If we suppose that at each stage of the algorithm these will continue to finite, nonsingular solutions of \mathcal{D}_{N+1} , then \mathcal{S}_N will have d^N entries. Therefore, the number of the solutions for \mathcal{D}_{N+1} grows exponentially as N increases, and hence the need of some filters becomes very important.

Next, we will extend the concept to two-dimensional partial differential equations and to non-polynomial right hand sides.

3 Homotopy continuation for two-dimensional PDEs

The method of finding all the solutions of second order ordinary differential equations presented in the previous section is now generalized to a class of nonlinear second order semilinear elliptic boundary value problems with Dirichlet boundary conditions on a rectangular domain, but still with polynomial nonlinearities. Most of the difficulties met for the one-dimensional case are also encountered in the new type of problems. Further difficulties that appear due to the generalization to a higher dimension are described in the first part of this section.

We build a homotopy function associated with a central difference discretization of the Laplace operator as a combination of some sparse matrices and vectors. All calculations necessary when introducing a new mesh point into a row are provided. The calculations necessary when introducing a new mesh point into a column are similar and are not presented. Both cases were implemented since we introduced mesh points into rows and columns alternately. The geometry of the moving mesh-grid is presented in Section 3.2.1.

3.1 Difficulties for ODEs and PDEs. We will concentrate on finding *all* the solutions for a problem of the form

$$\begin{aligned} \Delta u &= f(\lambda, x, y, u, u_x, u_y) \quad \text{on } \Omega \subset \mathbb{R}^2, \\ u|_{\partial\Omega} &= g, \end{aligned}$$

where Ω is a rectangular domain. In the next sections we will develop a method to solve this problem numerically. Before doing so, we review some difficulties that arose in the one-dimensional case.

3.1.1 Exponential growth of the number of the solutions. In the case of a polynomial nonlinearity, the number of complex solutions grows exponentially with the number of interior mesh points. We stop the computation with a heuristic criterion, e.g., when a desired number of interior mesh points has been attained, or the number of real solutions has stabilized, or the number of real solutions is growing without bound. In all the one-dimensional problems that we have examined, the number of real solutions stabilized or grew without bound. Once this was determined, we were able to take these solutions with 6–7 interior points and refine them to a finer grid. Supposing that 6–7 interior points for a one-dimensional interval will be satisfactory (providing that the number of real solutions

stabilizes or grows without bound), then for a two-dimensional problem on a rectangular domain we would be satisfied with 36–49 interior points. Clearly so many mesh points will lead to a great number of solutions for the homotopy and this will require stricter tests for filtering out unwanted solutions. For the one-dimensional case, properties of f to conclude symmetries of solutions were used as a filter in [3]. In higher dimensions it becomes more critical to filter out spurious solutions.

3.1.2 Turning points and bifurcations. An issue of concern arising in homotopy methods is the possibility that singular points such as turning points and bifurcation points may occur on the homotopy paths. One remedy is to introduce a random perturbation into the system, as for example, in the “T-trick” as discussed in [3]. While multiple solutions are relatively rare in the discretizations of boundary value problems for ODE’s, they arise regularly for PDE’s in the presence of solutions which are symmetric under groups of rotation and reflection transformations. For these reasons we have used continuation methods which incorporate arclength parametrization and easily round turning points. Arclength parametrization for analytic homotopy maps was examined in [4, 7] where it was shown that simple turning points in the homotopy parameter are also the site of simple bifurcation points, as defined below. We summarize here the results which enable a monotone tracing of homotopy paths.

Definition 3.1.1 (Simple turning point). Let $H : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ be sufficiently smooth. Suppose that $c : J \rightarrow \mathbb{R}^{n+1}$, $c(s) = (t(s), u(s))$ is a smooth curve, defined on an open interval J , and parametrized (for reasons of simplicity) with respect to arclength such that $H(c(s)) = 0$ for $s \in J$. The point $c(\bar{s})$ is said to be a simple turning point of the equation $H = 0$ if $\dot{t}(\bar{s}) = 0$, $\dot{t}(\bar{s}) \neq 0$, and the Jacobian

$$\begin{pmatrix} \dot{t} & \dot{u} \\ H_t & H_u \end{pmatrix} \Big|_{c(\bar{s})}$$

has minimum rank deficiency.

Definition 3.1.2 (Simple bifurcation point). Let $H : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ be a sufficiently smooth map. A point $\bar{u} \in \mathbb{R}^{n+1}$ is called a simple bifurcation point of the equation $H = 0$ if the following conditions hold,

- (1) $H(\bar{u}) = 0$,
- (2) $\dim \ker H'(\bar{u}) = 2$,
- (3) $e^* H''(\bar{u})[\phi, \psi]$ has one positive and one negative eigenvalue, where the vector e spans $\ker H'(\bar{u})^*$ and ϕ and ψ together span $\ker H'(\bar{u})$, where $H''(\bar{u})[\phi, \psi]$ is a bilinear form acting on the vectors ϕ and ψ .

Now, let $H : \mathbb{R} \times \mathbb{C}^n \rightarrow \mathbb{C}^n$ be a smooth homotopy. Assume that $H(t, w)$ is analytic in the variables w . We will also write $w = u + iv$ for $w \in \mathbb{C}^n$, where $u, v \in \mathbb{R}^n$ denote the real and the imaginary parts of w respectively. Note that $\overline{H(t, \bar{w})} = H(t, w)$ since H is analytic. Let us define now the real and imaginary parts $H^r, H^i : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n$

by

$$\begin{aligned} H^r(t, u, v) &= \frac{1}{2}(H(t, w) + H(t, \bar{w})), \\ H^i(t, u, v) &= \frac{-i}{2}(H(t, w) - H(t, \bar{w})), \end{aligned} \quad (6)$$

and the map $\hat{H} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{2n}$ by

$$\hat{H}(t, u, v) = \begin{pmatrix} H^r(t, u, v) \\ -H^i(t, u, v) \end{pmatrix}. \quad (7)$$

Theorem 3.1.3. *Let $\hat{c}(s) = (t(s), u(s), v(s))$ be a solution curve of $\hat{H}^{-1}(0)$. Any simple turning point $\hat{c}(\bar{s})$ of the equation $\hat{H} = 0$ is also a simple bifurcation point of the same equation.*

Theorem 3.1.4. *Under the assumptions of Theorem 3.1.3, let us now denote the two bifurcating solution curves of $\hat{H}^{-1}(0)$ by $\hat{c}_i(s) = (t_i(s), u_i(s), v_i(s))$, $i \in \{1, 2\}$. The curves are defined for s near \bar{s} and $\tilde{c} = \hat{c}_1(\bar{s}) = \hat{c}_2(\bar{s})$ is the bifurcation point. Then*

- (i) $(0, \dot{u}_1(\bar{s}), \dot{v}_1(\bar{s}))$ and $(0, -\dot{v}_1(\bar{s}), \dot{u}_1(\bar{s}))$ are orthogonal unit tangents to $\hat{c}_1(s)$ and $\hat{c}_2(s)$ at $s = \bar{s}$, respectively,
- (ii) $\ddot{t}_1(\bar{s}) = -\ddot{t}_2(\bar{s})$.

Remark 3.1.5. The above theorems generalize Propositions 11.8.10 and 11.8.16 in [5]. The proofs are similar in concept and may be seen in [4] which is available as a preprint at <http://www.math.colostate.edu/~tavener/>.

Remark 3.1.6. Theorem 3.1.3 shows that bifurcation points can be detected by noting turning points along the homotopy path. Turning points are easily detected by, for example, monitoring whether the inner product of successive unit tangent vectors becomes negative. Most numerical implementations of continuation methods construct tangent “predictor” vectors as points along the branch are calculated. Theorem 3.1.4 shows that one can easily change branches because the appropriate “predictor” vector for the bifurcating branch is orthogonal to the tangents of the solution curve near the bifurcation point. In this way, one can trace homotopy paths monotonically as t decreases to 0.

3.2 Implementation details.

3.2.1 Details on the two-dimensional grid for $\Delta u = f(\lambda, x, y, u, u_x, u_y)$. In this section we use numerical continuation to find all the solutions for a problem of the form

$$\Delta u = f(\lambda, x, y, u, u_x, u_y) \quad \text{on } \Omega = [a, b] \times [c, d] \subset \mathbb{R}^2, \quad (8)$$

$$u = g \quad \text{on } \partial\Omega. \quad (9)$$

Using a standard central difference approximation, with a uniform mesh with N interior points, the discretization of (8, 9) takes the form

$$\mathcal{D}_N : A_{xx}\vec{u} + \vec{b}_{xx} + A_{yy}\vec{u} + \vec{b}_{yy} = \vec{f} \quad (10)$$

where $\vec{u} = (u_1, u_2, \dots, u_N)^T$ is the discretized solution vector of the unknowns. Here, the discretization for $\frac{\partial^2 u}{\partial x^2}$ incorporating the left and right boundary conditions is $A_{xx}\vec{u} + \vec{b}_{xx}$. The discretization for $\frac{\partial^2 u}{\partial y^2}$ incorporating the upper and lower boundary conditions is $A_{yy}\vec{u} + \vec{b}_{yy}$.

As in the one-dimensional case, we will continue introducing *one* point at a time since introducing a row or column of p points would require solving a system of p polynomial equations to find the starting solutions for our homotopy. Introducing new mesh points one at a time also allows us to filter more frequently.

3.2.2 The homotopy function for $\Delta u = f(\lambda, x, y, u, u_x, u_y)$. We rewrite the boundary value problem (8, 9) as

$$\Delta u = f(\lambda, x, y, u, u_x, u_y) \quad \text{on } \Omega = [a, b] \times [c, d] \subset \mathbb{R}^2, \tag{11}$$

$$u|_{x=a} = u_a(x), \quad u|_{x=b} = u_b(x), \quad u|_{y=c} = u_c(y), \quad u|_{y=d} = u_d(y). \tag{12}$$

The main obstacle to be overcome when introducing extra points one at a time is the approximation of the Laplacian on non-regular finite difference meshes. This is achieved by appropriate application of linear interpolation. Details regarding the approximation of the derivatives for the general case of introducing a mesh point are given in Appendix A.

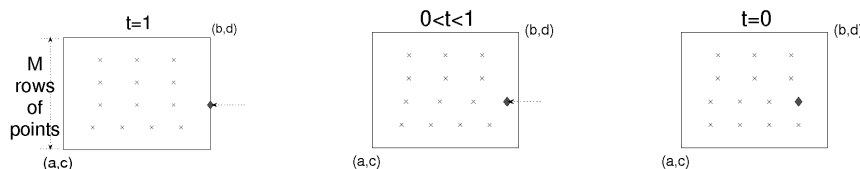


Figure 1. Introducing a new point (♦) in the 2nd row.

As indicated in (10), the boundary value problem (11, 12) can be discretized as

$$A_{xx}\vec{u} + \vec{b}_{xx} + A_{yy}\vec{u} + \vec{b}_{yy} = \vec{f},$$

where $A = A_{xx} + A_{yy}$ is the stiffness matrix and $\vec{b}_{xx}, \vec{b}_{yy}$ are what we will call boundary vectors. The corresponding homotopy (2) for this two-dimensional boundary value problem is

$$H_N(\vec{u}, t) := (A_{xx} + A_{yy})\vec{u} + (\vec{b}_{xx} + \vec{b}_{yy}) - \vec{f}, \tag{13}$$

where the explicit forms for $A_{xx}, A_{yy}, \vec{b}_{xx}, \vec{b}_{yy}$, and \vec{f} are given in Appendix B.

3.3 Stopping strategies. Independently of the details of the method we use to track the solutions of the homotopy H_N from $t = 1$ to $t = 0$, we must land exactly on $t = 0$ because only at $t = 0$ does the incompatibility between the old boundary conditions and the new ones disappear (see Remark 2.1 for more details). We have two stopping strategies.

- i) As we track a solution, the homotopy parameter t will decrease from 1 toward 0. When t is very close to 0, we choose the final steplength such that t is precisely 0. Using this method, t will always stay in the interval $[0, 1]$.
- ii) We track a solution path until t becomes negative for the first time. We then choose a “backward” steplength such that t lands precisely on 0.

The disadvantage of the second strategy is that we require an extra step to land on $t = 0$. This however is insignificant in comparison with the advantage that any turning point near $t = 0$ will be observed and the switch to the proper branch as suggested in Remark 3.1.6 can be performed. Some of the possible paths that can be met in the tracking process are presented in Figure 2.

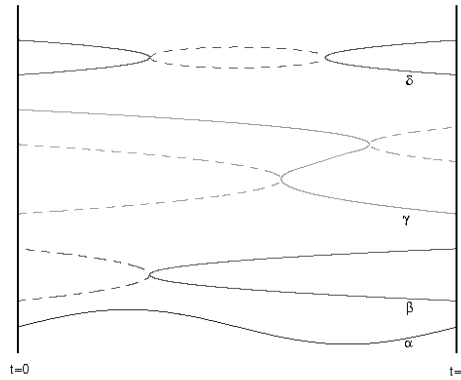


Figure 2. Possible paths in the tracking process.

3.4 Algorithm for the moving mesh. Let us describe the steps for finding all solutions to

$$\begin{aligned} \Delta u &= f(u) \quad \text{on } \Omega = [0, 1] \times [0, 1], \\ u &= 0 \quad \text{on } \partial\Omega. \end{aligned}$$

Let us use the finite difference discretization as described in Appendix B and form the homotopy map as in Equation (13). Let l be the number of the row or column to which a mesh point is introduced in a continuous manner as t goes from $t = 1$ to $t = 0$. Let N be the current number of mesh points. For $t = 1$, the solutions to $H_N(\bar{u}, 1) = 0$ are available from the previous solution. For example, if f is a polynomial, the solutions to $H_N(\bar{u}, 0) = 0$ are found by finding the eigenvalues of the corresponding companion matrix of $H_N(\bar{u}, 0)$. Similarly, at the introduction of a new mesh point on a row or column, the starting points for the homotopy are also found by solving a single polynomial equation in the l^{th} equation.

Algorithm 1 Mesh refinement procedure

Solve $H_1(\bar{u}, 0) = 0$ (companion matrix)
for $M = 1$ to M_{\max} **do**
 {Add mesh points to rows}
 for $l = 1$ to M **do**
 $N = M^2 + l$
 Find all solutions \bar{s} , such that $H_N(\bar{s}, 1) = 0$ (companion matrix)
 For all solutions \bar{s} , trace $H_N(\bar{u}, t) = 0$ with $H_N(\bar{s}, 1) = 0$ to obtain all solutions \bar{u} with $H_N(\bar{u}, 0) = 0$ (homotopy)
 Delete spurious solutions to obtain a set \mathcal{S}_N
 end for
 {Add mesh points to columns}
 for $l = 1$ to $M + 1$ **do**
 $N = M^2 + M + l$
 Find all solutions \bar{s} , such that $H_N(\bar{s}, 1) = 0$ (companion matrix)
 For all solutions \bar{s} , trace $H_N(\bar{u}, t) = 0$ with $H_N(\bar{s}, 1) = 0$ to obtain all solutions \bar{u} with $H_N(\bar{u}, 0) = 0$ (homotopy)
 Delete spurious solutions to obtain a set \mathcal{S}_N
 end for
end for

Remark 3.4.1. Either M_{\max} is chosen at the outset or one can stop when the number of solutions in \mathcal{S}_N has become constant. The homotopy tracking incorporates the tests for turning points and branch switching as outlined above.

4 Numerical results for polynomial right hand side

4.1 Numerical results for $\Delta u = -(1 + u^2)$. We now consider the following partial differential equation in two dimensions,

$$\Delta u = -(1 + u^2) \quad \text{on } \Omega = [0, 1] \times [0, 1], \quad (14)$$

$$u = 0 \quad \text{on } \partial\Omega. \quad (15)$$

Using a tracking method based on arclength continuation and the techniques presented in Section 3 we obtained the results in Table 1. In particular, we switched branches at turning points as discussed in Remark 3.1.6.

In every case, the starting points for the homotopies were the solutions from the previous mesh. A solution is considered to be real if the magnitude of the imaginary part at each mesh point was less than 10^{-8} . For each $N = 1, \dots, 16$ interior points, we have obtained exactly the maximum possible number of solutions indicated by Bezout's theorem. Out of these, the number of real solutions stabilized to 2 as N increased.

Remark 4.1.1. Our implementation of numerical continuation used steplength controls as suggested in [5]. In particular, the residuals of predictors, angles between successive

N	l	r/c	No. solutions	No. real solutions	Solutions with same turning points	The value of t at the turning points
1	1	r	2	2		
2	1	r	4	2		
3	1	c	8	2	(5, 7)	0.1003
4	2	c	16	2	(11, 15)	0.0036
5	1	r	32	6	(18, 16) (2, 10) (23, 31)	0.9245 0.8573 0.3553
6	2	r	64	4	(47, 51) (48, 52) (3, 19) (35, 63)	0.9831 0.9438 0.9242 0.3592
7	1	c	128	4	(7, 39) (8, 72) (71, 127)	0.9739 0.8996 & 0.0175 0.3853
8	2	c	256	4	(79, 143) (80, 144) (15, 255)	0.9970 0.9616 0.4410
9	3	c	512	2	(31, 511)	0.3820
10	1	r	1024	2	(63, 1023)	0.4828
11	2	r	2048	2	(127, 2047)	0.5638
12	3	r	4096	2	(255, 4095)	0.4858
13	1	c	8192	2	(511, 8191)	0.4945
14	2	c	16384	2	(1023, 16383)	0.5907
15	3	c	32768	2	(2047, 32767)	0.5920
16	4	c	65536	2	(4095, 65535)	0.4985

Table 1. The number of solutions for $\Delta u = -(1 + u^2)$ on $\Omega = [0, 1] \times [0, 1]$ with zero Dirichlet boundary conditions.

tangents and empirical monitoring of the contraction rate of corrector steps were incorporated. Thus we were able to resolve turning points which were very close to $t = 1$ and $t = 0$ successfully. Without such care we would perhaps not have noted the turning points at $t = 0.0036$ and $t = 0.9970$ for $N = 4$ and $N = 8$ interior points, respectively.

Remark 4.1.2. In this example, as in fact in all the ones we studied, during our tracking we met all the possible paths presented in Figure 2, except the one denoted γ . A path with exactly two turning points as portrayed by δ occurred at $N = 7$ interior points.

Of those 2^{16} solutions for 16 interior points, there were exactly two real and four complex ones which are invariant under the dihedral group D_4 of symmetries of the square. In Figure 3 we have plotted these six solutions with 4×4 interior points. Using interpolation of these as initial guesses for Newton iterations, we have obtained the refined solutions plotted in Figure 4. S-R-1 and S-R-2 are the two real solutions, r-S-C-1 and i-S-C-1 are the real and the imaginary parts respectively of one of the four complex solutions, r-S-C-2 and i-S-C-2 are the real and the imaginary parts respectively of another of the four complex solutions. The other two complex solutions are the conjugate of S-C-1 and S-C-2. Only these six solutions did not undergo significant modifications as we refined the grid.

4.2 Numerical results for $\Delta u = -\lambda(1 + u^2)$. The bifurcation diagram for $\Delta u = -\lambda(1 + u^2)$ with zero Dirichlet boundary conditions was independently produced using an arclength continuation algorithm on a 39×39 mesh. Here, and elsewhere in this paper, we obtained bifurcation diagrams by plotting the infinity norm of the solutions versus the

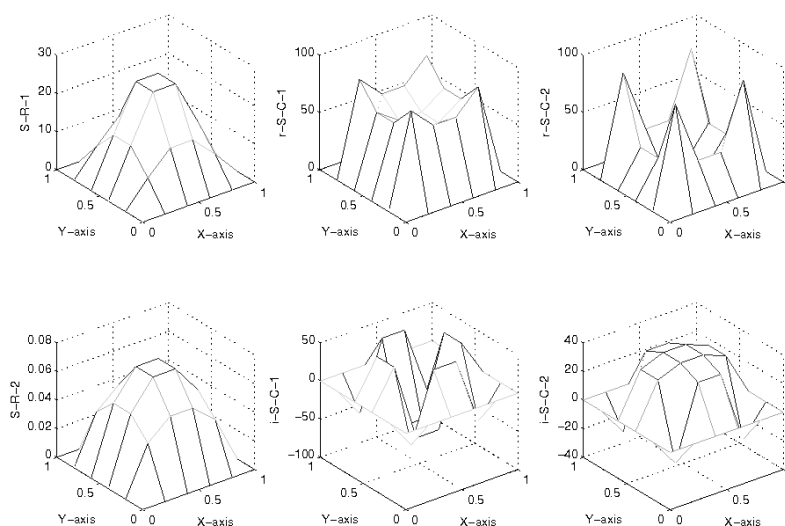


Figure 3. Six solutions for $\Delta u = -(1 + u^2)$ with 4×4 interior points.

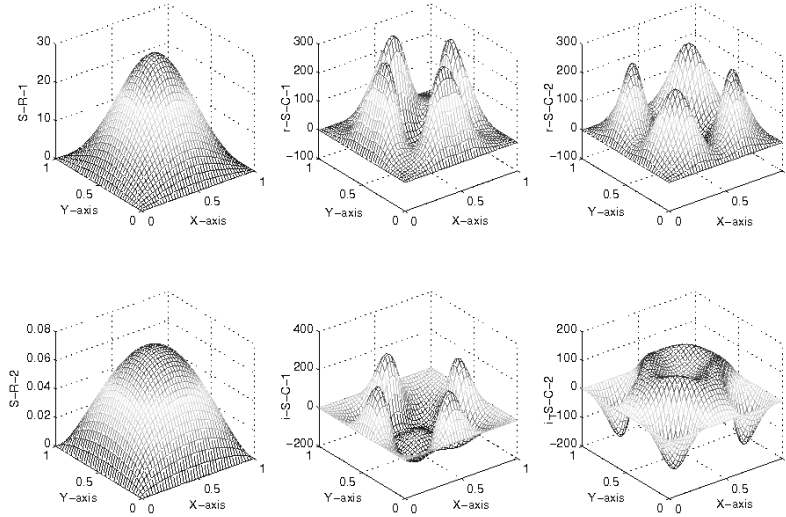


Figure 4. The six solutions for $\Delta u = -(1 + u^2)$ refined to 39×39 interior points.

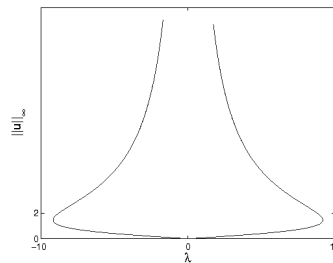


Figure 5. The bifurcation diagram for $\Delta u = -\lambda(1 + u^2)$ with zero Dirichlet boundary conditions.

parameter λ . Two turning points are attained at $\pm\lambda_*$, where $\lambda_* = 9.1890$. The two real solutions shown in Figure 4 correspond to the two intersections of the bifurcation diagram with a vertical line drawn at $\lambda = 1$. For $\lambda = 10$, our mesh deformation method produced no real solutions out of 2^N complex ones. For $\lambda = 9$ (a value close to the turning point $\lambda_* = 9.1890$), we found no real solutions for the first 4 interior points. One real solution appeared after introducing the 5th interior point. The second real solution appeared only after introducing many more interior points. For values of $9 < \lambda < \lambda_*$ the real solutions appeared only after introducing more than 5 interior points. Our results also indicate there are no disjoint branches in the bifurcation diagram.

4.3 The problem of Breuer, McKenna and Plum. We used the methods described in the previous section to examine a conjecture posed in [6] where the following theorem was provided.

Theorem 4.3.1. *The equation*

$$\Delta u + u^2 = 800 \sin \pi x \sin \pi y \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega \tag{16}$$

where $\Omega = (0, 1) \times (0, 1)$, has at least four solutions.

Our approach will give all the solutions of (8) not only for $f(u)$ being a polynomial of u , but also for any function $f(x, y, u)$ that is a polynomial as a function of u , because even in this case, the equation corresponding to (5) for our two-dimensional BVP is still a polynomial equation in one unknown. For problem (16), we found using our homotopy continuation that there are exactly four real solutions which are essentially different. We also got real solutions that are rotations or reflections of these, but after filtering to remove these conjugate solutions, we obtained only the four truly distinct real solutions that appear in Figure 6.

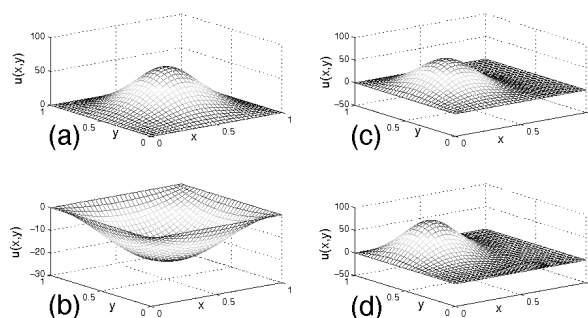


Figure 6. The four real solutions for (16).

The solutions (a) and (b) are fully symmetric (i.e., symmetric with respect to reflections about the axes $x = \frac{1}{2}$, $y = \frac{1}{2}$, $x = y$, and $x = 1 - y$), the solution (c) is symmetric only with respect to reflection about $y = \frac{1}{2}$, and the solution (d) is symmetric only with respect to reflection about $x + y = 0$.

The peaks of the positive and negative fully symmetric solutions (a) and (b) are 61.776 and -21.358 , respectively. The peak of the solution (c) has the value 69.923 and is attained at $(x, y) \approx (\frac{5}{16}, \frac{1}{2})$. The peak of the solution (d) has the value 76.321 and is attained at $(x, y) \approx (\frac{1}{3}, \frac{2}{3})$. All these values were found using a mesh with 63×63 interior points.

A generalization of Equation (16) considered in [6] is

$$\Delta u + u^2 = \lambda \sin \pi x \sin \pi y \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega. \tag{17}$$

Starting with the 4 solutions found for $\lambda = 800$ on a mesh with 31×31 interior points, we used arclength numerical continuation to construct the bifurcation diagram in Figure 7.

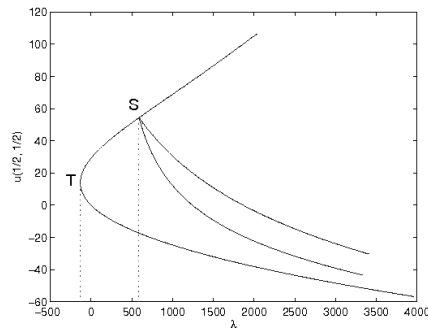


Figure 7. The bifurcation diagram for (16).

We found a turning point T at $\lambda_T \approx -133.3$ at which one eigenvalue of the Jacobian changes sign. The Jacobian is in fact $A + 2 \text{diag}(\vec{u})$, where A is the stiffness matrix (see Section 3.2.2). We also found a symmetry breaking bifurcation S at $\lambda_S \approx 587.7$ at which a pair of other eigenvalues of the Jacobian (a double eigenvalue) changes sign. We performed continuation for values of λ up to 8000 and saw no indication that another eigenvalue will approach zero as λ increases.

In order to test our findings, we performed homotopy continuation for a few different values of λ and confirmed that there are no real solutions for $\lambda < \lambda_T$, two real solutions if $\lambda_T < \lambda < \lambda_S$, and four real solutions if $\lambda > \lambda_S$. Furthermore, we confirmed that there do not seem to be disjoint real branches in this bifurcation diagram.

Breuer, McKenna and Plum review several natural conjectures which may be made for equations such as (17). Two such conjectures are, (i) as $\lambda \rightarrow \infty$ there are at least four solutions (weak version) and, (ii) more solutions are created as bifurcations from the positive curve, considerably further up the positive branch (the stronger version). Using a Mountain Pass Algorithm, Breuer, McKenna and Plum find the weak conjecture is true, but the strong one seems to be false. This coincides with our results.

5 Homotopy continuation for nonpolynomial nonlinearity

We now generalize our method to nonpolynomial nonlinearities in the partial differential problems. The algorithm described in Section 2 works well for the case when the function f is a polynomial in u , because in this case, by using companion matrices we are able to find *all* the solutions of the polynomial Equation (4) that resulted from the introduction of a new mesh point. When the nonlinearity f is not a polynomial, there is no general method to find all solutions.

Indeed, solving Equation (4) now becomes quite different. For example, it may have no solutions at all, even under complexification. In this case, we may conclude that the corresponding boundary value problem also has no solutions. Another possibility is that (4) may have infinitely many solutions, particularly when the complexification is considered. Since it is not feasible to compute infinitely many solutions, it becomes necessary to confine the search for solutions to some compact region (such as a rectangle in \mathbb{C} , or correspondingly in \mathbb{R}^2). The choice of the rectangle may be predicted based upon a priori estimates or on some practical bound on the norm of the solutions.

The Bratu problem considered in the next section illustrates how matters differ from the polynomial case, but also the necessity to use the complexification of (4). Since we now seek *all* the solutions to (4) in a bounded domain and under weakened assumptions on f , we are led to consider cellular exclusion methods. We briefly introduce the concept in Section 5.2.

5.1 The Bratu problem and the exclusion method. The toolbox developed in [3] was used to find all the solutions of a one-dimensional boundary value problem of the form

$$\begin{aligned} u'' &= f(x, u, u'), & x \in (a, b), \\ u(a) &= \alpha, & u(b) = \beta \end{aligned}$$

where f is a polynomial of u . We now consider the case in which f is not a polynomial function of u . As an example, we will look at the one-dimensional Bratu Problem

$$\begin{aligned} u'' + \lambda \exp(u) &= 0 & \text{on } (0, 1), \\ u(0) &= u(1) = 0. \end{aligned} \tag{18}$$

Let us associate the homotopy function from (2) and (3) to this problem. To be able to use our algorithm for finding all the solutions of (18) for a specific value of λ , we need to be able to find all the roots of the Equation (4) which, for this case, can be rewritten as

$$u_N - 2u_{N+1} + \lambda \left(\frac{1}{N+1} \right)^2 \exp(u_{N+1}) = 0. \tag{19}$$

Therefore we now have to find all the real solutions of the equation

$$\exp(u) - au + b = 0, \tag{20}$$

where a and b are positive constants, $a = 2(N+1)^2/\lambda$, $b = u_N(N+1)^2/\lambda$. One can easily see that the number of real zeros for such an equation is either 2, 1 or 0, depending upon whether $a(1 - \ln(a)) + b$ is negative, zero or positive, respectively. One can also easily implement a method to find the real zeros of this equation and integrate that into our homotopy continuation approach (it might help to see that b/a is a lower bound for these zeros, if they exist).

For every $\lambda \in [0, 2.1]$ which we tried, we obtained only two real solutions every time we introduced a new grid point. It is well known that the Bratu problem has two real solutions for $\lambda \in [0, \lambda_*]$, where $\lambda_* \approx 3.5127$ represents the value of the turning point in the bifurcation diagram from Figure 8.

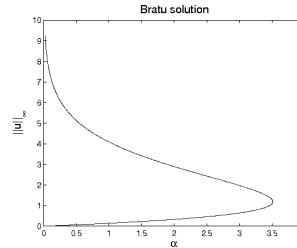


Figure 8. The bifurcation diagram for the Bratu Problem.

In the interval $[2.1, \lambda_*]$ the difficulty we encountered was the fact that the Equation (20) did not have any *real* zeros for small numbers of mesh points, and therefore we could not start using our homotopy. This reminds us of a fact encountered in the polynomial case, that sometimes real solutions will arise from complex ones as we increase the number of interior points. Therefore, it is not sufficient to know how to solve (20) only for real solutions, but also for the complex ones.

Writing $u = u_{\Re} + iu_{\Im}$ and $b = b_{\Re} + ib_{\Im}$, Equation (20) now takes the form

$$\begin{aligned} \exp(u_{\Re}) \cos(u_{\Im}) - au_{\Re} + b_{\Re} &= 0 \\ \exp(u_{\Re}) \sin(u_{\Im}) - au_{\Im} + b_{\Im} &= 0. \end{aligned} \quad (21)$$

The necessity of solving this system for the real variables u_{\Re} and u_{\Im} led us to consider the exclusion algorithms. Using these algorithms, we were able to find all the solutions u_{\Re} and u_{\Im} of this system in a rectangle.

5.2 Exclusion algorithm. In this section we will give some background about the exclusion algorithms that we will use here (see also [9]). Exclusion methods provide a useful tool for finding all the solutions of a nonlinear system of equations over a compact domain. They may also be used for finding the global minima of a function. The concept goes back to Moore [12]. Further research on cellular exclusion algorithms to find all the solutions of a nonlinear system appears in [17, 16, 15, 11]. Georg and collaborators [9, 1, 10] introduced and analyzed several new tests for finding the zeros and the global minima of functions over a compact domain. The results led to significant improvements in the efficiency of the methods.

In \mathbb{R}^n and $\mathbb{R}^{m \times n}$ we use the component-wise “ \leq ” as a partial ordering, “ $|\cdot|$ ” as the component-wise absolute value, and “ $\|\cdot\|_{\infty}$ ” as the max norm. For example, for two matrices $A, B \in \mathbb{R}^{m \times n}$, the symbol $A \leq B$ means that $A(i, j) \leq B(i, j)$ for $i = 1, \dots, m, j = 1, \dots, n$.

Definition 5.2.1. An interval σ in \mathbb{R}^n is a compact cell of the form

$$\sigma = [m_{\sigma} - r_{\sigma}, m_{\sigma} + r_{\sigma}] = \{x \in \mathbb{R}^n : m_{\sigma} - r_{\sigma} \leq x \leq m_{\sigma} + r_{\sigma}\},$$

where $m_{\sigma}, r_{\sigma} \in \mathbb{R}^n$ are called the midpoint and the radius of σ respectively, with $r_{\sigma}(i) > 0$ for $i = 1, \dots, n$. Also, $m_{\sigma} - r_{\sigma}$ and $m_{\sigma} + r_{\sigma}$ are called the lower and upper corners respectively.

Definition 5.2.2 (Exclusion test). Let $\sigma \in \mathbb{R}^n$ be an interval and $F : \sigma \rightarrow \mathbb{R}^n$ be a function defined on σ . A test

$$T_F(\sigma) \in \{0, 1\} \quad \text{where} \quad 0 \equiv \text{no} \text{ and } 1 \equiv \text{yes}$$

is called an exclusion test for F on σ iff $T_F(\sigma) = 0$ implies that F has no zero point in σ .

Therefore, $T_F(\sigma) = 1$ is a necessary condition for F to have a zero point in σ .

Example 5.2.3. Let $L > 0$ be a Lipschitz constant for f on the interval σ . Then

$$f(m_\sigma) \leq L \|r_\sigma\| \tag{22}$$

is an exclusion test for f on σ .

If an exclusion test is given (assuming that $T_F(\sigma)$ is available for any subinterval σ of some initial interval Λ on which F is defined), then we can recursively bisect intervals and discard the ones which yield a negative test. This is the basic idea of an Exclusion Algorithm.

We use a strategy of cyclic bisections of the intervals along successive axes. We say that we have reached a new bisection level whenever one cycle of bisections is accomplished. We also think of an exclusion algorithm as performing a fixed number of bisection levels. We will denote by Γ_l the list of the intervals generated by the algorithm on the l^{th} level, which are in fact the intervals that have not been discarded after l bisection levels.

It is obvious that if $\Gamma_l = \emptyset$ for some level l , then the algorithm has shown that there are no zero points of F in the initial interval Λ .

Algorithm 2 Exclusion Algorithm

```

 $\Gamma \leftarrow \{\Lambda\}$ 
for  $l = 1, \dots, \text{maximal\_level}$  do
  for  $a = 1, \dots, n$  do
     $\tilde{\Gamma}$  is obtained by bisecting each  $\sigma \in \Gamma$  along the axis  $a$ 
    for  $\sigma \in \tilde{\Gamma}$  do
      if  $T_F(\sigma) = 0$  then
        drop  $\sigma$  from  $\tilde{\Gamma}$  ( $\sigma$  is excluded)
      end if
    end for
     $\Gamma \leftarrow \tilde{\Gamma}$ 
  end for
   $\Gamma_l \leftarrow \Gamma$ 
end for

```

The exclusion tests we discuss are applied component-wise on vector-valued function $F : \sigma \rightarrow \mathbb{R}^n$. Therefore, we only need to consider an exclusion test for a scalar-valued function $f : \sigma \rightarrow \mathbb{R}$ and then combine such (possibly different types of) exclusion tests

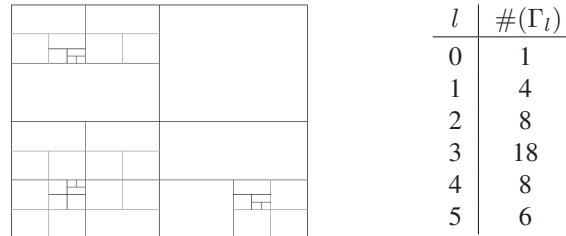


Figure 9. Bisection levels for a cell $\sigma \subset \mathbb{R}^2$. The figure shows the surviving cells at levels $l = 0, \dots, 6$.

to obtain an exclusion test for a vector-valued function $F = \{f_i\}_{i=1, \dots, n} : \sigma \rightarrow \mathbb{R}^n$ by setting $T_F(\sigma) = \prod_{i=1}^n T_{f_i}(\sigma)$. Hence, we may concentrate our attention on scalar functions $f : \sigma \rightarrow \mathbb{R}$ when designing exclusion tests. We also need good exclusion tests which are computationally inexpensive but relatively tight, because otherwise too many intervals remain undiscarded on each bisection level and this will lead to significant numerical inefficiency.

Definition 5.2.4. For two power series $f(x) = \sum_{\alpha} f_{\alpha} x^{\alpha}$ and $g(x) = \sum_{\alpha} g_{\alpha} x^{\alpha}$ we define

$$f \prec\prec g \iff |f_{\alpha}| \leq g_{\alpha} \text{ for all } \alpha.$$

Some simple exclusion tests were given in [16, 15] and the following complexity result was also shown.

Theorem 5.2.5. Let $\Lambda \subset \mathbb{R}^n$ be an interval, $F : \Lambda \rightarrow \mathbb{R}^n$ and zero a regular value of F . Then there is a constant $C > 0$ such that the algorithm started in Λ generates no more than C intervals on each bisection level, i.e. $\#(\Gamma_l) \leq C$ independent of l .

Remark 5.2.6. The constant C can be very big and numerical experiments have shown that some exclusion tests ((22) for example) are not tight enough for non-linear systems in which some solutions may be singular.

5.3 Exclusion tests via dominant functions. Georg [9] gave efficient exclusion tests based on the concept of dominant functions by using Taylor expansions. He gave several results for explicitly constructing dominant functions. We now briefly review some of these ideas from that paper.

Definition 5.3.1. Let $\sigma \subset \mathbb{R}^n$ be an interval. We denote:

$$\begin{aligned} \mathcal{A}_k(\sigma) &:= \{f : \sigma \rightarrow \mathbb{R}^n : \partial^{\alpha} f \text{ is absolutely continuous for } |\alpha| \leq k\}, \\ \mathcal{K}_k(\sigma) &:= \{g \in \mathcal{A}_k(\sigma) : 0 \leq \partial^{\alpha} g(x) \leq \partial^{\alpha} g(y) \text{ for } 0 \leq x \leq y, |\alpha| \leq k\}. \end{aligned}$$

Definition 5.3.2. Let $f \in \mathcal{A}_k(\sigma)$ and $g \in \mathcal{K}_k(\sigma)$. We say that g dominates f with order k on σ and write $f(x) \prec_k g(x)$ for $x \in \sigma$ iff the estimates

$$|\partial^\alpha f(x)| \leq \partial^\alpha g(|x|) \quad \text{for } x \in \sigma$$

hold for all $x \in \sigma$ and $|\alpha| \leq k$.

See Appendix C for some results on building a dominant function g when f is given.

Remark 5.3.3. $f(x) \prec_k g(x)$ for $x \in \sigma$ implies that $f(x) \prec_q g(x)$ for $x \in \tau$ for any $\tau \subset \sigma$ and $q \leq k$. From now on we will try to use the notation $f \prec_k g$ instead of $f(x) \prec_k g(x)$ if there is no ambiguity about the underlying interval.

Theorem 5.3.4. Let $\sigma \subset \mathbb{R}^n$ be an interval, and $q > 0$ be an integer. Let $f(m_\sigma + x) \prec_q g(x)$ for $|x| \leq r_\sigma$. Then

$$|f(m_\sigma)| \leq g(r_\sigma) - g(0) - \sum_{0 < |\alpha| < q} \underbrace{(\partial^\alpha g(0) - |\partial^\alpha f(m_\sigma)|)}_{\geq 0} r_\sigma^\alpha \quad (23)$$

is an exclusion test for f on σ .

Corollary 5.3.5. Let $\sigma \subset \mathbb{R}^n$ be an interval, and $q > 0$ be an integer. Let $f \prec_q g$ on σ . Then

$$|f(m_\sigma)| \leq g(|m_\sigma| + r_\sigma) - g(|m_\sigma|) - \sum_{0 < |\alpha| < q} \underbrace{(\partial^\alpha g(|m_\sigma|) - |\partial^\alpha f(m_\sigma)|)}_{\geq 0} r_\sigma^\alpha \quad (24)$$

is an exclusion test for f on σ .

The terms inside the summation sign in (23) and (24) are nonnegative, and therefore the test tightens as q increases. To increase the efficiency of the implementation, one would successively apply the test for $q = 1, \dots, q_0$ (for some given q_0) and discard the intervals as soon as the test fails.

5.4 The solutions for the Bratu problem. Consider again the Bratu Problem (18). The homotopy function which gives a mesh refinement in continuous deformation for this problem is given in (2), (3) where $f(x, u, u') = \lambda \exp(u)$.

The starting points u_{N+1} for $H_{N+1}(u_1, u_2, \dots, u_{N+1}, \lambda, 1) = 0$ can be obtained by solving (20), but as we saw in Section 5.1, it was not sufficient to look only for the real solutions. Therefore, using the exclusion test (24), we will seek for the solutions of the associated complexified system of Equations (21).

Let

$$F(x, y) = \begin{bmatrix} \exp(x) \cos(y) - ax + b_{\Re} \\ \exp(x) \sin(y) - ay + b_{\Im} \end{bmatrix}. \quad (25)$$

Using the results from Appendix C, for a dominating function we take

$$G(x, y) := \left[\begin{array}{l} \left(1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{\exp(m_\sigma^1 + r_\sigma^1)}{120} x^5\right) \left(1 + \frac{y^2}{2} + \frac{y^4}{24} + \frac{y^5}{120}\right) + |a|x + |b_{\Re}| \\ \left(1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{\exp(m_\sigma^1 + r_\sigma^1)}{120} x^5\right) \left(y + \frac{y^3}{6} + \frac{y^5}{120}\right) + |a|y + |b_{\Im}| \end{array} \right].$$

One can easily check that $F \prec_5 G$ on any interval $\sigma = [m_\sigma - r_\sigma, m_\sigma + r_\sigma] \subset \mathbb{R}^2$. We can now apply the numerical homotopy method to solve (2), (3) with $f(x, u, u') = \lambda \exp(u)$ for $\lambda = 3$, a value close to the turning point λ_* . We will use the exclusion test (24) with $q = 5$ for the functions above until both components of the radii of the generated cells in Γ_l are less than $\epsilon = 0.1$. The results we obtained are shown in Tables 2 and 3.

N	No. solutions	No. real solutions
1	6	0
2	35	1
3	210	2
4	1259	3
5	7547	3

Table 2. $m_\sigma = (10, 0)$ and $r_\sigma = (10, 15)$.

N	No. solutions	No. real solutions
1	2	0
2	3	1
3	6	2
4	11	3
5	19	3
6	35	3
7	68	3
8	133	3
9	265	5
10	527	5

Table 3. $m_\sigma = (5, 0)$ and $r_\sigma = (5, 5)$.

Table 2 lists the number of solutions found using continuation in t and the exclusion algorithm (24) with the starting cell given by $m_\sigma = (10, 0)$ and $r_\sigma = (10, 15)$. Table 3 lists the number of solutions found using continuation in t and the exclusion algorithm (24) with the starting cell given by $m_\sigma = (5, 0)$ and $r_\sigma = (5, 5)$.

Remark 5.4.1. The exclusion algorithm process can be very costly and it is applied very often (every time we add a new interior point and for every solution on the previous grid).

For example, in Table 2, for adding one more point to the already existing four points implies using the exclusion algorithm 1259 times to solve equations of the form (21). Also, to reach the goal of $\|r_\sigma\| \leq \epsilon$, for any $\sigma \in \Gamma_l$, we needed $l = 8$ bisection levels each time we used the exclusion algorithm. Table 4 shows how the number of intervals $\sigma \in \Gamma_l$ changes as l increases when using the exclusion algorithm to solve (21) as a fifth point is added to the already existing four interior points.

The bisection level l	0	1	2	3	4	5	6	7	8
Card(Γ_l)	1	4	16	64	247	870	1534	14	9

Table 4. Card(Γ_l) at each bisection level l .

The nine small cells at level 8 actually approximated six different solutions for (21). This arises due to adjacency of surviving cells. Tables 2 and 3 illustrate the advisability of choosing an initial cell which is not too large, unless it is necessary.

Remark 5.4.2. For each $N = 1, \dots, 5$, all the solutions found in Table 3 have also been present in Table 2; this was expected since the cell chosen for the exclusion algorithm in Table 2 was bigger and included the one used in Table 3. Using interpolation of these real solutions (three for Table 2 for $N = 5$ and five for Table 3 for $N = 10$) as initial guesses for Newton iterations, only two of them could be refined to more interior points. For the other ones, the Newton process did not converge. The value of u at the peaks for these two symmetric solutions at $\lambda = 3$ with 1407 interior points are 0.6401 and 1.975 respectively.

6 Conclusions

While there is a dearth of theoretical results for existence, number and qualitative properties of solutions for nonlinear PDE problems, we have seen that reliable information may be obtained through solving discretizations numerically by homotopy methods. First the system of equations is embedded into a complex setting. Next, when introducing a new mesh point, we use a method for finding all solutions to a single complex equation. For polynomial equations we use the Matlab program which finds the eigenvalues of the corresponding companion matrix. For general nonlinearities, we have written a cellular exclusion program for finding the zeros of the real and imaginary parts. These steps furnish the starting points for tracing solution paths defined by a homotopy deformation of the mesh. We cannot throw away complex solutions found at the end of the homotopy, since as we have seen in Section 5.4 the real solutions sought may only arise after the discretization is sufficiently fine.

Techniques for finding *all* solutions of systems have an important role to play. The methods presented in this paper are not meant to produce fast and accurate methods for solving PDE's but rather the idea is to provide reliable information about the number of solutions as well as their qualitative properties. Furthermore, the approximations we obtain can be used as starting values to get more accurate solutions on finer meshes.

Appendix A Approximating the Laplacian

In these appendices we use the notation,

- t is the homotopy parameter,
- l is the row in which the new mesh point is to be introduced,
- m is the number of rows of mesh points,
- n is the number of mesh points on each of the rows $l + 1, \dots, m$; hence, rows $1, \dots, l$ each has $n + 1$ points,
- N is the total number of mesh points; observe that $N = l(n + 1) + (m - l)n$, for $t \in [0, 1)$.

When we introduce a new point on the l^{th} column (\blacklozenge) from the right, all the other points on the l^{th} row shift leftward as in the figure below. Remark that rows $1, 2, \dots, l$ have $n + 1$ points each, and rows $l + 1, l + 2, \dots, m$ have n points each.

- $\Delta x_{\text{old}} = \frac{b-a}{n+1}$
- $\Delta x_{\text{new}} = \frac{b-a}{n+2}$
- $\Delta y_{\text{old}} = \frac{d-c}{m+1}$
- $h(t) = t\left(\frac{b-a}{n+1}\right) + (1-t)\left(\frac{b-a}{n+2}\right)$
- $d(t) = (b-a) - (n+1)h(t) = \dots = (1-t)\frac{b-a}{n+2}$

For example, in Figure 10, the discretized solution \vec{u} of the system (10) will be written as $\vec{u} = [u_{1,1}, u_{2,1}, u_{3,1}, u_{4,1}, u_{1,2}, u_{2,2}, u_{3,2}, u_{4,2}, \dots, u_{1,5}, u_{2,5}, u_{3,5}]_{18 \times 1}^T$.

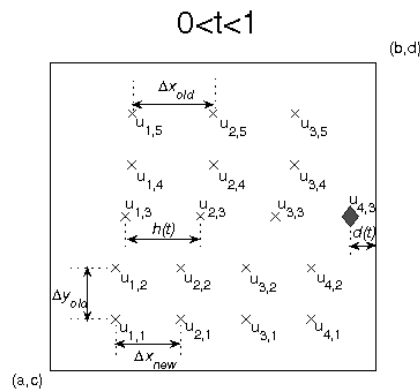


Figure 10. Introducing a new point (\blacklozenge) on the l^{th} row.

Remark. Notice the way we order the unknowns $u_{i,j}$ inside of the vector \vec{u} . When a new point will be introduced on a column (from above for example) instead of a row, the unknowns $u_{i,j}$ inside of the vector \vec{u} will be ordered differently (to keep the sparsity

structure of the stiffness matrix A). In this case, the vector \vec{u} will have the form

$$\vec{u} = [u_{1,1}, u_{1,2}, u_{1,3}, u_{1,4}, u_{1,5}, u_{2,1}, u_{2,2}, \dots]^T.$$

Appendix A.1 All rows except $l - 1, l$, and $l + 1$. We can easily approximate the Laplacian at all the points which are not on the rows $l - 1, l$, and $l + 1$ by

- Formula for $(u_{xx})_{j,k}$

$$(u_{xx})_{j,k} \approx \frac{u_{j-1,k} - 2u_{j,k} + u_{j+1,k}}{\Delta x^2},$$

for $j = 1, \dots, n$ (or $n + 1$), $k = 1, \dots, m$, $k \neq l$, where

$$\Delta x = \begin{cases} \Delta x_{\text{new}} & \text{if } 1 \leq k \leq l - 1 \\ \Delta x_{\text{old}} & \text{if } l + 1 \leq k \leq m. \end{cases}$$

- Formula for $(u_{yy})_{j,k}$

$$(u_{yy})_{j,k} \approx \frac{u_{j,k-1} - 2u_{j,k} + u_{j,k+1}}{\Delta y^2},$$

for $j = 1, \dots, n$ (or $n + 1$), $k = 1, \dots, m$, $k \notin \{l - 1, l, l + 1\}$, where $\Delta y = \Delta y_{\text{old}}$.

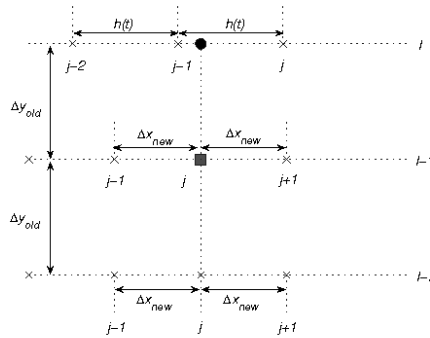


Figure 11. Introducing a new point (\blacklozenge) on the l^{th} row. Approximating the Laplacian at the points on the $(l - 1)^{\text{th}}$ row.

Appendix A.2 Row $l - 1$. On the $(l - 1)^{\text{th}}$ row we have the following.

- Formula for $(u_{xx})_{j,l-1}$

$$(u_{xx})_{j,l-1} \approx \frac{u_{j-1,l-1} - 2u_{j,l-1} + u_{j+1,l-1}}{\Delta x^2},$$

for $j = 1, \dots, N + 1$, where $\Delta x = \Delta x_{\text{new}}$.

- Formula for $(u_{yy})_{j,l-1}$

$$(u_{yy})_{j,l-1} \approx \frac{u_{j,l-2} - 2u_{j,l-1} + \tilde{u}_j}{\Delta y^2},$$

for $j = 1, \dots, n+1$, where $\Delta y = \Delta y_{\text{old}}$; $\tilde{u}_j := u_{j-1,l} + r_j \cdot (u_{j,l} - u_{j+1,l})$, where

$$r_j = \frac{j\Delta x_{\text{new}} - (j-1)h(t)}{h(t)}.$$

Hence,

$$(u_{yy})_{j,l-1} \approx \frac{(1-r_j)u_{j-1,l} + u_{j,l-2} - 2u_{j,l-1} + r_j u_{j,l}}{\Delta y_{\text{old}}^2},$$

for $j = 1, \dots, n+1$.

Appendix A.3 Rows l and $l+1$. In a similar manner, one needs to take care of approximating the Laplacian at the interior points on the l^{th} and on the $(l+1)^{\text{th}}$ rows.

Appendix B The components of the homotopy H_n

A_{xx} is a block diagonal matrix of the form¹

$$A_{xx} = \text{diag}(T_1, \dots, T_1, T_2, T_3, \dots, T_3)$$

where T_1, T_2, T_3 are tridiagonal square matrices of sizes $N+1, N+1$, and N , respectively $T_1 = \frac{1}{\Delta x_{\text{new}}^2} S$, $T_3 = \frac{1}{\Delta x_{\text{old}}^2} S$, $T_2 = \frac{1}{h(t)^2} S$, where S is a tridiagonal matrix having -2 on the main diagonal and 1 on the upper and lower diagonals.

\vec{b}_{xx} is a *block* vector of the form

$$\vec{b}_{xx} = (\vec{b}_1(y_1), \dots, \vec{b}_1(y_{l-1}), \vec{b}_2(y_l), \vec{b}_3(y_{l+1}), \dots, \vec{b}_3(y_m))^T$$

where $\vec{b}_1, \vec{b}_2, \vec{b}_3$ are sparse vectors of sizes $1 \times (n+1)$, $1 \times (n+1)$, and $1 \times n$, respectively:

$$\begin{aligned} \vec{b}_1(y_j) &= \frac{1}{\Delta x_{\text{new}}^2} (u_a(y_j), 0, \dots, 0, u_b(y_j)), \\ \vec{b}_2(y_j) &= \frac{1}{h(t)^2} (u_a(y_j), 0, \dots, 0, u_b(y_j)), \\ \vec{b}_3(y_j) &= \frac{1}{\Delta x_{\text{old}}^2} (u_a(y_j), 0, \dots, 0, u_b(y_j)). \end{aligned}$$

¹In the structure of A_{xx} , there are $l-1$ blocks T_1 and $m-l$ blocks T_3 .

\vec{b}_{yy} is a *block* vector of the form²

$$\vec{b}_{yy} = (\vec{\alpha}, \vec{0}, \dots, \vec{0}, \vec{\varphi}, \vec{\phi}, \vec{0}, \dots, \vec{0}, \vec{\beta})^T,$$

where $\vec{\alpha}$, $\vec{\varphi}$, $\vec{\phi}$ are vectors of sizes $1 \times (n+1)$, and $\vec{\beta}$ is a vector of size $1 \times n$.

- $\vec{\alpha} = \frac{1}{\Delta y_{\text{old}}^2} (u_c(x_{1,\text{new}}), u_c(x_{2,\text{new}}), \dots, u_c(x_{n+1,\text{new}}))$.
- $\vec{\beta} = \frac{1}{\Delta y_{\text{old}}^2} (u_d(x_{1,\text{old}}), u_d(x_{2,\text{old}}), \dots, u_d(x_{n,\text{old}}))$.
- $\vec{\varphi}^T = \frac{1}{\Delta y_{\text{old}}^2} \begin{pmatrix} (1 - \psi_1)u_a(y_L) \\ 0 \\ \vdots \\ 0 \end{pmatrix}$,

where $\psi_1 = 1 - \frac{t}{t+n+1}$.

- $\vec{\phi}^T = \frac{1}{\Delta y_{\text{old}}^2} \begin{pmatrix} (1 - \phi_1)u_a(y_{l+1}) \\ 0 \\ \vdots \\ 0 \\ \phi_2 u_b(y_{l+1}) + \phi_3 u_a(y_{l+1}) \end{pmatrix}$,

where $\phi_1 = 1 - \frac{1-t}{n+2}$, $\phi_2 = 1 - \frac{(n+1)(1-t)}{n+2}$, $\phi_3 = t$.

Remark. A similar calculation was also performed to write the homotopy function when the new point is introduced on a column instead of a row.

Appendix C Building dominant functions

First, let us mention a connection between dominant functions and the relation defined in the Definition 5.2.4.

Theorem C.1. Let $f(x) = \sum_{\alpha} f_{\alpha} x^{\alpha}$ and $g(x) = \sum_{\alpha} g_{\alpha} x^{\alpha}$ be two power series convergent on an interval $\sigma \subset \mathbb{R}^n$ which contains the origin. Then

$$f \prec_{\infty} g \iff f \prec\prec g.$$

The following examples point out the differences between the various estimates.

Example C.2.

- If $g \in \mathcal{K}_k$ then $g \prec_k g$. This includes examples such as $\exp(m+x) \prec\prec \exp(m+x)$, and $\tan(x) \prec\prec \tan(x)$ for $|x| < \frac{\pi}{2}$.
- $\sin(x) \prec\prec \sinh(x)$, but $\sin(x) \prec_1 x$, $\sin(x) \prec_2 x + \frac{1}{2}x^2$, $\sin(x) \prec_3 x + \frac{1}{6}x^3$.

²In the structure of \vec{b}_{yy} from above, $\vec{\alpha}^T$ is on the first position, $\vec{\varphi}^T$ is on the $(l-1)$ st position, $\vec{\phi}^T$ is on l^{th} position, and $\vec{\beta}^T$ is on the m^{th} position.

- $\cos(x) \prec \prec \cosh(x)$, but $\cos(x) \prec_1 1 + x$, $\cos(x) \prec_2 1 + \frac{1}{2}x^2$, $\cos(x) \prec_3 1 + \frac{1}{2}x^2 + \frac{1}{6}x^3$.
- $\log(1 + x) \prec \prec -\log(1 - x)$, but $\log(1 + x) \prec_3 x + \frac{1}{2}x^2 + \frac{1}{3}x^3$ for $|x| < 1$.
- $\sin(m + x) \prec \prec \sinh(|m| + x)$, but $\sin(m + x) \prec_2 |\sin(m)| + |\cos(m)|x + \frac{1}{2}x^2$.

The following theorem consists of a list of rules that can be used as a tool to generate dominant functions, in much the same way as rules about differentiation are used as a tool to generate derivatives (see the examples following the theorem).

Theorem C.3.

1. If $f \prec_k g$, then $f(m + x) \prec_k g(|m| + x)$.
2. If $f \prec_1 g$, then $|f| \prec_1 g$.
3. If $f \prec_k g$, then $\lambda f \prec_k |\lambda|g$, for any $\lambda \in \mathbb{R}$.
4. If $f_i \prec_k g_i$, $i = 1, \dots, q$, then $\sum_i f_i \prec_k \sum_i g_i$.
5. If $f_i \prec_k g_i$, $i = 1, \dots, q$, then $\prod_i f_i \prec_k \prod_i g_i$.
6. Let $f \prec_k g$ and $f_i \prec_k g_i$, $i = 1, \dots, n$. Set $F = f(f_1, \dots, f_n)$ and $G = g(g_1, \dots, g_n)$. Then $F \prec_k G$.

Example C.4.

- $e^{|\sin(m+x)|} \prec_1 e^{|\sin(m)|+x}$ since $e^t \prec \prec e^t$ and $\sin(m + x) \prec_1 |\sin(m)| + x$.
- $\sin(x^2) \cos(y - 2z) \prec_3 (x^2 + \frac{1}{6}(x^2)^3)(1 + \frac{1}{2}(y + 2z)^2 + \frac{1}{6}(y + 2z)^3)$.
- $\frac{1}{1 + \frac{1}{5}\cos(x)} \prec_2 \frac{1}{1 - \frac{1}{5}(1 + \frac{1}{2}x^2)}$ for $x \in [-2\sqrt{5}, 4]$ since $\frac{1}{1+t} \prec \prec \frac{1}{1-t}$ for $|t| < 1$ and $\cos(x) \prec_2 1 + \frac{1}{2}x^2$.

References

- [1] E. Allgower, M. Erdmann, K. Georg, On the complexity of exclusion algorithms for optimization. *J. Complexity* **18** (2002), 573–588. [MR1919450 \(2003f:65080\)](#) [Zbl 1005.68082](#)
- [2] E. L. Allgower, A survey of homotopy methods for smooth mappings. In: *Numerical solution of nonlinear equations (Bremen, 1980)*, volume 878 of *Lecture Notes in Math.*, 1–29, Springer 1981. [MR644324 \(83j:65057\)](#) [Zbl 0461.65037](#)
- [3] E. L. Allgower, D. J. Bates, A. J. Sommese, C. W. Wampler, Solution of polynomial systems derived from differential equations. *Computing* **76** (2006), 1–10. [MR2174348 \(2006j:65191\)](#) [Zbl 1086.65075](#)
- [4] E. L. Allgower, S.-G. Cruceanu, S. Tavener, Turning points and bifurcations for homotopies of analytic maps. Submitted, 2007.
- [5] E. L. Allgower, K. Georg, *Numerical continuation methods*, volume 13 of *Springer Series in Computational Mathematics*. Springer 1990. [MR1059455 \(92a:65165\)](#) [Zbl 0717.65030](#)
- [6] B. Breuer, P. J. McKenna, M. Plum, Multiple solutions for a semilinear boundary value problem: a computational multiplicity proof. *J. Differential Equations* **195** (2003), 243–269. [MR2019251 \(2005a:35083\)](#) [Zbl pre02017250](#)
- [7] S.-G. Cruceanu, Numerical Solutions of Nonlinear Systems Derived From Semilinear Elliptic Equations. PhD Dissertation, Colorado State University 2007.

- [8] C. B. García, T. Y. Li, On the number of solutions to polynomial systems of equations. *SIAM J. Numer. Anal.* **17** (1980), 540–546. [MR584729 \(81m:65074\)](#) [Zbl 0445.65037](#)
- [9] K. Georg, Improving the efficiency of exclusion algorithms. *Adv. Geom.* **1** (2001), 193–210. [MR1840221 \(2002d:65043\)](#) [Zbl 0982.65069](#)
- [10] K. Georg, A new exclusion test. In: *Proceedings of the International Conference on Recent Advances in Computational Mathematics (ICRACM 2001) (Matsuyama)*, volume 152, 147–160, 2003. [MR1991287 \(2004f:65067\)](#) [Zbl 1034.65033](#)
- [11] B. L. Meng, Z. D. Xing, A cell exclusion algorithm for finding all solutions to systems of nonlinear equations. *Pure Appl. Math. (Xi'an)* **15** (1999), 72–76. [MR1716264](#) [Zbl 0953.65035](#)
- [12] R. E. Moore, A test for existence of solutions to nonlinear systems. *SIAM J. Numer. Anal.* **14** (1977), 611–615. [MR0657002 \(58 #31801\)](#) [Zbl 0365.65034](#)
- [13] B. L. van der Waerden, *Algebra. Vol. I*. Springer 1991. [MR1080172 \(91h:00009a\)](#) [Zbl 1032.00001](#)
- [14] B. L. van der Waerden, *Algebra. Vol. II*. Springer 1991. [MR1080173 \(91h:00009b\)](#) [Zbl 1032.00002](#)
- [15] Z.-B. Xu, J.-S. Zhang, Y.-W. Leung, A general CDC formulation for specializing the cell exclusion algorithms of finding all zeros of vector functions. *Appl. Math. Comput.* **86** (1997), 235–259. [MR1465554 \(98g:65051\)](#) [Zbl 0910.65033](#)
- [16] Z.-B. Xu, J.-S. Zhang, W. Wang, A cell exclusion algorithm for determining all the solutions of a nonlinear system of equations. *Appl. Math. Comput.* **80** (1996), 181–208. [MR1417322 \(98b:65053\)](#) [Zbl 0883.65042](#)
- [17] J. S. Zhang, W. Wang, B. L. Chen, A cell exclusion algorithm for finding all global minimizers of functions of several variables. *Math. Numer. Sinica* **17** (1995), 443–455. [MR1477379](#) [Zbl 0860.65049](#)

Received 24 July, 2007; revised 13 November, 2007

E. Allgower, Department of Mathematics, Colorado State University, 101 Weber Building,
Ft. Collins, CO 80523, USA
Email: allgower@math.colostate.edu

S.-G. Cruceanu, Department of Mathematics, Colorado State University, 101 Weber Building,
Ft. Collins, CO 80523, USA
Email: cruceanu@math.colostate.edu

S. Tavener, Department of Mathematics, Colorado State University, 101 Weber Building,
Ft. Collins, CO 80523, USA
Email: tavener@math.colostate.edu