AN EFFICIENT SPECTRAL TRUST-REGION DEFLATION METHOD FOR MULTIPLE SOLUTIONS

LIN LI[†], LI-LIAN WANG[‡] AND HUIYUAN LI[§]

Abstract. It is quite common that a nonlinear partial differential equation (PDE) admits multiple distinct solutions and each solution may carry a unique physical meaning. One typical approach for finding multiple solutions is to use the Newton method with different initial guesses that ideally fall into the basins of attraction confining the solutions. In this paper, we propose a fast and accurate numerical method for multiple solutions comprised of three ingredients: (i) a well-designed spectral-Galerkin discretization of the underlying PDE leading to a nonlinear algebraic system (NLAS) with multiple solutions; (ii) an effective deflation technique to eliminate a known (founded) solution from the other unknown solutions leading to deflated NLAS; and (iii) a viable nonlinear least-squares and trust-region (LSTR) method for solving the NLAS and the deflated NLAS to find the multiple solutions sequentially one by one. We demonstrate through ample examples of differential equations and comparison with relevant existing approaches that the spectral LSTR-Deflation method has the merits: (i) it is quite flexible in choosing initial values, even starting from the same initial guess for finding all multiple solutions; (ii) it guarantees high-order accuracy; and (iii) it is quite fast to locate multiple distinct solutions and explore new solutions which are not reported in literature.

1. Introduction

Many nonlinear differential equations arisen from mathematical modeling in physics, mechanics, biology, energy and engineering, admit multiple solutions and each may carry its own physics and unique dynamics (see, e.g., [14,21,27]). Most problems, if not all, do not have explicit solutions, so the development of efficient and accurate numerical methods for finding multiple solutions becomes a topic of longstanding interest.

In this paper, we intend to identify multiple solutions of the nonlinear ordinary differential equations

$$F(x, u, u', \cdots, u^{(n)}) = 0, \quad x \in \Omega, \tag{1.1}$$

and the second-order nonlinear elliptic problems

$$\Delta u + F(x, y, u) = 0, \quad (x, y) \in \Omega, \tag{1.2}$$

2000 *Mathematics Subject Classification.* 65N35, 65N22, 65F05, 65L10.

Key words and phrases. Multiple solutions, trust-region method, deflation, Legendre-Galerkin method.

[†]School of Mathematics and Physics and Hunan Key Laboratory for Mathematical Modeling and Scientific Computing, University of South China, Hengyang, 421001, China. This work of this author is partially supported by the Science Foundations of Hunan Province (Nos: 2020JJ5464, 20C1595). Email: lilinmath@usc.edu.cn (L. Li).

[‡]Corresponding author. Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, 637371, Singapore. The research of this author is partially supported by Singapore MOE AcRF Tier 1 Grant: RG15/21. Email: lilian@ntu.edu.sg (L. Wang).

[§]State Key Laboratory of Computer Science/Laboratory of Parallel Computing, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China. Email: huiyuan@iscas.ac.cn (H. Li).

The first author would like to thank the hospitality of Nanyang Technological University for the visit to finalize this work.

supplemented with some boundary conditions, where Ω is a bounded domain in $\mathbb{R}^d, d = 1, 2$, and the nonlinear functional F has a regularity to be specified later. We refer to [3, 20, 22, 24] for the study of the existence and multiplicity of the solutions to such problems.

Before we introduce our approach, we feel compelled to elaborate on the relevant background and motivations. Choi and Mckenna [8] proposed the mountain pass algorithm (MPA) for (1.2). As commented in Xie, Chen and Yu [31], the MPA is feasible for finding two solutions of mountain pass type with Morse index 1 or 0. However, according to Ding, Costa and Chen [11], the MPA may fail to locate the sign-changing solutions, so they proposed a high linking algorithm (HLA) to compute such solutions with some inspirations from [29]. Later, Li and Zhou [19] developed the minimax algorithm (MNA) for multiple solutions for which we refer to [7, 35–37, 40] for more recent advancements along this line. It is noteworthy that these methods are more or less based on the variational structures of the underlying nonlinear differential equations. The second category of the existing methods is to discretize the differential equation by a numerical method (e.g., finite difference, finite element or spectral method) and then search for multiple solutions of the resulting nonlinear algebraic system (NLAS). In this regard, a search-extension method (SEM) was proposed in [5] for the semilinear PDEs with some improvements in [32, 33]. Roughly speaking, the SEM searches for the starter from solving the companion linear problem (e.g., by the eigenfunction expansion method) and then extend to solve the NLAS, e.g., by Newton iteration with the starter as initial guess. Noticeably, a homotopy continuation method was introduced by Allgower et al. [1, 2] for finding multiple solutions of the NLAS obtained from finite difference discretization of type (1.2). In general, in order to solve the NLAS: $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$, we consider $\boldsymbol{H}_\tau(\boldsymbol{x}) = (1-\tau)\boldsymbol{F}(\boldsymbol{x}) + \gamma\tau\boldsymbol{G}(\boldsymbol{x}) = \boldsymbol{0}$, where $\boldsymbol{H}_\tau(\boldsymbol{x})$ is known as the homotopy function, $\tau \in [0, 1]$ is the homotopy tracking number, and γ is a randomly chosen complex number. Here, the starting system $\boldsymbol{H}_1(\boldsymbol{x}) = \boldsymbol{G}(\boldsymbol{x}) = \boldsymbol{0}$ should be constructed which is expected to be easier to solve and closely related to the original NLAS. Then the homotopy systems $\boldsymbol{H}_{\tau_j}(\boldsymbol{x}) = \boldsymbol{0}$ should be solved for various τ_j from 1 to 0 using a suitable iterative method, where τ can be viewed as an extra dimension. This technique has inspired some interesting recent works, see, e.g., [18, 28, 39].

This work is mostly motivated by Farrell et al. [13] where a deflation technique and Newton iteration were integrated to solve the NLAS resulted from finite element discretization of the underlying PDE with multiple distinct solutions. Given a solution $\boldsymbol{x}_1$ of $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$, the essence of deflation is to eliminate this known solution $\boldsymbol{x}_1$ and then continue to search for a new solution by modifying the NLAS as $\boldsymbol{G}(\boldsymbol{x}; \boldsymbol{x}_1) = \boldsymbol{0}$, where $\boldsymbol{G}$ is properly constructed so that $\boldsymbol{G}(\boldsymbol{x}_1; \boldsymbol{x}_1) \neq \boldsymbol{0}$. In a nutshell, the algorithm in [13] repeated this process and used the Newton iteration for finding $\boldsymbol{x}_1$ and solving the deflated systems. In fact, the deflation technique has its root in polynomial root-finding problems [16] and been extended by Brown and Gearhart [4] for solutions of nonlinear algebraic systems. As we know, the Newton iteration is sensitive to the initial guess and costly to compute the inverse of the Jacobian matrix that become even severer when the NLAS are deflated. Indeed, the choice of good initial inputs becomes ad hoc and problematic, and at times, this integrated algorithm fails to converge.

We aim to overcome the sensitivity of initial guesses and substantially improve the efficiency and accuracy of the deflation technique. Our proposal consists of the following ingredients.

(i) We employ well-designed Legendre spectral-Galerkin methods to discretise (1.1) and (1.2) that can guarantee a good approximability of the resulted NLAS to the original differential equations with a relatively low computational cost.

(ii) We make use of the deflation technique to sequentially search for multiple solutions of the NLAS.

(iii) We introduce the nonlinear least-squares trust-region method to solve the NLAS and the deflated systems.

Accordingly, the proposed algorithm is dubbed as the spectral LSTR-Deflation method. Compared with the existing methods, the main differences and advantages of our algorithm reside in the following aspects.

- Finding multiple solutions based on the Newton method typically requires the attempt of many different initial guesses that lie in different basins of attraction [13]. However, the use of the LSTR method has a good convergence where the choices of initial inputs are much more relaxed and fairy flexible at times. As the deflated systems turn out to be more and more complicated, the LSTR method becomes vitally important to ensure a good performance of the whole algorithm. This allows us to start even with the same initial values for multiple solutions.
- Compare with several existing methods, the proposed approach is capable of finding new solutions and much faster (see the examples in Section 3).

The remainder of this paper is organized as follows. In Section 2, we describe the spectral LSTR-Deflation method for the model problem (1.2). In Section 3, we provide ample numerical experiments to demonstrate the efficiency and accuracy of the algorithm. We then conclude the paper with some remarks.

2. The Legendre spectral-Galerkin LSTR-Deflation method

In this section, we present the spectral LSTR-Deflation method for the model problem (1.2). We start with its Legendre spectral-Galerkin discretisation that leads to the nonlinear linear algebraic system to be solved. We then introduce the nonlinear least-squares trust-region method for the NLAS, followed by the exposition of the deflation technique. Finally, we summarise the whole algorithm.

2.1. Legendre spectral-Galerkin method for a model problem.

As an illustrative example, we consider the nonlinear elliptic problem:

$$-\Delta u + F(u) = f(x,y), \quad (x,y) \in \Omega := (-1,1)^2; \quad u|_{\partial\Omega} = 0, \tag{2.1}$$

where $F(z)$ is a smooth nonlinear function (e.g., a polynomial as in [28]), and f is continuous.

As the first step, we discretize (2.1) by using the Legendre spectral-Galerkin method [25]. Let $\mathbb{P}_N$ be the set of the polynomials of degree at most N, and $\mathbb{P}_N^0 = \{\phi \in \mathbb{P}_N : \phi(\pm 1) = 0\}$. The spectral-Galerkin approximation is to find $u_N \in X_N^0 = (\mathbb{P}_N^0)^2$ such that

$$(\nabla u_N, \nabla v_N) + (I_N F(u_N), v_N) = (I_N f, v_N), \quad \forall v_N \in X_N^0, \tag{2.2}$$

where I_N the Legendre-Gauss-Lobatto tensorial interpolation operator with $N+1$ points in each coordinate direction. As in [25], we introduce the basis of $\mathbb{P}_N^0$ as follows

$$\phi_k(x) = L_{k+2}(x) - L_k(x), \quad 0 \le k \le N-2,$$

and write

$$\begin{aligned}
u_N &= \sum_{k,j=0}^{N-2} \hat{u}_{kj}\phi_k(x)\phi_j(y), \quad U = (\hat{u}_{kj})_{k,j=0,1,\cdots,N-2};\\
a_{kj} &= \int_I \phi_j'(x)\phi_k'(x)dx, \qquad A = (a_{kj})_{k,j=0,1,\cdots,N-2};\\
b_{kj} &= \int_I \phi_j(x)\phi_k(x)dx, \qquad B = (b_{kj})_{k,j=0,1,\cdots,N-2};\\
f_{kj} &= (I_N f, \phi_k(x)\phi_j(y)), \quad g_{kj} = (I_N F(u_N), \phi_k(x)\phi_j(y))_{k,j=0,1,\cdots,N-2}.
\end{aligned} \tag{2.3}$$

We obtain the linear system

$$(A\otimes B + B\otimes A^\top)\boldsymbol{u} = \boldsymbol{f} - \hat{\boldsymbol{g}}(\boldsymbol{u}), \tag{2.4}$$

where we arrange

$$\begin{aligned}
\boldsymbol{u} &= (\hat{u}_{00}, \hat{u}_{10}, \cdots, \hat{u}_{q0}; \hat{u}_{01}, \cdots, \hat{u}_{q1}; \cdots; \hat{u}_{0q}, \cdots, \hat{u}_{qq})^\top,\\
\boldsymbol{f} &= (f_{00}, f_{10}, \cdots, f_{q0}; f_{01}, \cdots, f_{q1}; \cdots; f_{0q}, \cdots, f_{qq})^\top,\\
\hat{\boldsymbol{g}} &= (g_{00}, g_{10}, \cdots, g_{q0}; g_{01}, \cdots, g_{q1}; \cdots; g_{0q}, \cdots, g_{qq})^\top.
\end{aligned}$$

Here $\otimes$ denotes the Kronecker product, i.e. $A\otimes B = (Ab_{ij})_{i,j=0,1,\cdots,q}$ with $q = N-2$.

Note that (2.4) is a nonlinear system with $\hat{\boldsymbol{g}} = \hat{\boldsymbol{g}}(\boldsymbol{u})$, which will be solved by the trust-region method. Essentially, we only need to evaluate $\hat{\boldsymbol{g}}(\boldsymbol{u})$ for given $\boldsymbol{u}$ that can be implemented efficiently by the pseduo-spectral technique described in [25, Ch. 4].

Remark 2.1. *For the problem* (1.1) *with high-order derivatives, it is advantageous to use spectral methods, which we shall describe in Section 3.* □

2.2. **Nonlinear least-squares and trust-region method.** Formally, we write (2.4) as the following system of nonlinear equations:

$$\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}, \quad \boldsymbol{F} = \big(F_1,\ F_2,\ \cdots,\ F_n\big)^\top, \tag{2.5}$$

with the unknown vector $\boldsymbol{x} = (x_1, \cdots, x_n)^\top$. We reformulate the zero-finding problem as the optimisation problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} Q(\boldsymbol{x}), \quad Q(\boldsymbol{x}) := \frac{1}{2}\big\|\boldsymbol{F}(\boldsymbol{x})\big\|^2 = \frac{1}{2}\sum_{i=1}^n F_i^2(\boldsymbol{x}), \tag{2.6}$$

where $\|\cdot\|$ is the vector 2-norm. Denote the gradient and Hessian matrices by

$$\begin{aligned}
\boldsymbol{g}(\boldsymbol{x}) &:= \nabla Q(\boldsymbol{x}) = \boldsymbol{J}^\top(\boldsymbol{x})\boldsymbol{F}(\boldsymbol{x}),\\
\boldsymbol{G}(\boldsymbol{x}) &:= \nabla^2 Q(\boldsymbol{x}) = \boldsymbol{J}^\top(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x}) + \boldsymbol{S}(\boldsymbol{x}),
\end{aligned}$$

where

$$\boldsymbol{J}(\boldsymbol{x}) = \boldsymbol{F}'(\boldsymbol{x}) = (\nabla F_1(\boldsymbol{x}), \nabla F_2(\boldsymbol{x}), \cdots, \nabla F_n(\boldsymbol{x}))^\top, \quad \boldsymbol{S}(\boldsymbol{x}) = \sum_{i=1}^n F_i(\boldsymbol{x})\nabla^2 F_i(\boldsymbol{x}).$$

We resort to the trust-region method to solve the nonlinear least-squares minimization problem (2.6). For this purpose, we introduce a region around the current best solution, called the trust region (say of radius h_k), and approximate the objective function by a quadratic form which boils down to solving a sequence of trust-region subproblems:

$$\min_{\boldsymbol{s}\in\mathbb{B}_{h_k}} q^{(k)}(\boldsymbol{s}) := Q(\boldsymbol{x}^{(k)}) + \boldsymbol{g}(\boldsymbol{x}^{(k)})^\top \boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^\top \boldsymbol{G}(\boldsymbol{x}^{(k)})\boldsymbol{s}, \tag{2.7}$$

where the trust region $\mathbb{B}_{h_k} := \{\boldsymbol{s} \in \mathbb{R}^n : \|\boldsymbol{s}\| \le h_k\}$, and the matrices $\boldsymbol{g}(\boldsymbol{x}^{(k)})$ and $\boldsymbol{G}(\boldsymbol{x}^{(k)})$ are the gradient and Hessian matrices at current point $\boldsymbol{x}^{(k)}$, respectively. Let $\boldsymbol{s}_k$ be the minimizer of $q^{(k)}(\boldsymbol{s})$ in the trust-region of radius h_k. Then we update $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \boldsymbol{s}_k$. It is critical to choose a proper radius h_k. In general, when there is good agreement between $q^{(k)}(\boldsymbol{s}_k)$ and the objective function value $Q(\boldsymbol{x}^{(k+1)})$, one should choose h_k as large as possible. More precisely, assuming that $q^{(k)}(0) \neq q^{(k)}(\boldsymbol{s}_k)$ (otherwise, $\boldsymbol{s}_k$ is a minimizer), we define

$$r_k = \frac{Q(\boldsymbol{x}^{(k)}) - Q(\boldsymbol{x}^{(k+1)})}{q^{(k)}(\boldsymbol{0}) - q^{(k)}(\boldsymbol{s}_k)}. \tag{2.8}$$

The ratio r_k is an indicator for the expansion and contraction the trust regions. If r_k is close to 1, it means there is good agreement, so we can expand the trust-region for the next iteration; if r_k is close to zero or negative, we should shrink the trust-region radius; otherwise, we do not alter the trust-region.

We remark that as an efficient numerical optimization method for solving nonlinear programming (NLP) problems, the TRM enjoys the desirable global convergence with a local superlinear rate of convergence as follows.

Theorem 2.1 (see [26])**.** *Assume that*

(i) *the function $Q(\boldsymbol{x})$ is bounded below on the level set*

$$S := \{\boldsymbol{x} \in R^n \mid Q(\boldsymbol{x}) \le Q(\boldsymbol{x}^{(0)})\}, \quad \forall\, \boldsymbol{x}^{(0)} \in \mathbb{R}^n, \tag{2.9}$$

and is Lipschitz continuously differentiable in S;

(ii) *the Hessian matrixes $G(\boldsymbol{x}^{(k)})$ are uniformly bounded in 2-norm, i.e., $\|G(\boldsymbol{x}^{(k)})\| \le \beta$ for any k and some $\beta > 0$.*

If $\boldsymbol{g}(\boldsymbol{x}^{(k)}) \neq \boldsymbol{0}$, then

$$\lim_{k\to\infty} \inf \|\boldsymbol{g}(\boldsymbol{x}^{(k)})\| = 0. \tag{2.10}$$

Moreover, if $\boldsymbol{g}(\boldsymbol{x}^*) = \boldsymbol{0}$, and $\boldsymbol{G}(\boldsymbol{x}^*)$ is positive definite, then the convergence rate of the TRM is quadratic.

Remark 2.2. *When k is large enough, the TRM becomes the Newton iteration. As a result, it has the same convergence rate as the Newtonian method.* □

Remark 2.3. *In practice, the gradient and Hessian matrices might be appropriately approximated by some numerical means. We refer to Zhang et al. [38] for such derivative-free methods for* (2.6) *with $\boldsymbol{F}$ being twice continuously differentiable, but none of their first-order or second-order derivatives being explicitly available.* □

2.3. **Deflation technique for multiple solutions.** The critical issue in finding multiple solutions is to get away from one identified solution and search for a new solution based on ideally the same initial input. Motivated by Farrell et al [13], we introduce an effective technique rooted in the notion of deflation. As illustrated in Figure 2.1, suppose that the first (approximate) solution $\boldsymbol{x}_1^*$ of $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$ in (2.5) via the LSTR method, and we intend to find a second (approximate) root $\boldsymbol{x}_2^*$ of $\boldsymbol{F}$. For this purpose, we consider the modified zero-finding problem $\widehat{\boldsymbol{F}}(\boldsymbol{x}) = \boldsymbol{0}$ where $\widehat{\boldsymbol{F}}$ is constructed in a way to deflate the first solution $\boldsymbol{x}_1^*$, and allow for finding $\boldsymbol{x}_2^*$ by the LSTR method even with the same initial data. We then repeat this deflation process to find more solutions.

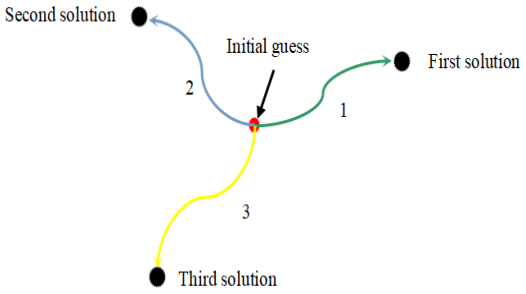


FIGURE 2.1. A schematic illustration of the deflation process integrated with the LSTR method described in Subsection 2.2. Starting from the initial guess (marked by **red dot**), we obtain the first solution (along **Path 1**) via LSTR method. We then deflate this solution away by the deflation technique and update the objective function accordingly. As a result, we can find the second solution (along **Path 2**) via LSTR method with the same initial condition. Repeating this process, we search for other new solutions along different paths (e.g. **Path 3)**.

We next introduce the deflation operator, and highlight the essence of the deflation technique [4, 13, 30].

Definition of deflation operator. *Let $\mathbb{X}, \mathbb{Y}$, and $\mathbb{Z}$ be Banach spaces, and $\widehat{\mathbb{X}}$ be an open subset of $\mathbb{X}$. Let $\boldsymbol{F} : \mathbb{X} \to \mathbb{Y}$ be a Fréchet differentiable operator with the Fréchet derivative $\boldsymbol{F}'$. Suppose that $\boldsymbol{r} \in \widehat{\mathbb{X}}$ is a root of $\boldsymbol{F}$, i.e., $\boldsymbol{F}(\boldsymbol{r}) = \boldsymbol{0}$, such that $\boldsymbol{F}'(\boldsymbol{r})$ is nonsingular. For each $\boldsymbol{x} \in \widehat{\mathbb{X}} \setminus \{\boldsymbol{r}\}$, let $\Psi(\boldsymbol{x}; \boldsymbol{r}) : \mathbb{Y} \to \mathbb{Z}$ be an invertible linear operator, and define*

$$\widehat{\boldsymbol{F}}(\boldsymbol{x}) = \Psi(\boldsymbol{x}; \boldsymbol{r})\boldsymbol{F}(\boldsymbol{x}). \tag{2.11}$$

We call Ψ a deflation operator, if for any sequence $\{\boldsymbol{x}_i\} \subseteq \widehat{\mathbb{X}} \setminus \{\boldsymbol{r}\}$ converging to $\boldsymbol{r}$, we have

$$\liminf_{i \to \infty} \|\widehat{\boldsymbol{F}}(\boldsymbol{x}_i)\|_{\mathbb{Z}} > 0. \tag{2.12}$$

It is seen from (2.11)-(2.12) that the role of the deflation operator is to exclude and deflate the root $\boldsymbol{r}$, so $\widehat{\boldsymbol{F}}(\boldsymbol{x})$ has common roots as $\boldsymbol{F}(\boldsymbol{x})$ except for $\boldsymbol{r}$.

We next present some sufficient conditions for constructing suitable deflation operators.

Theorem 2.2 (see [13])**.** *Let $\boldsymbol{F} : \widehat{\mathbb{X}} \subset \mathbb{X} \to \mathbb{Y}$ be a Fréchet differentiable operator. Suppose that the linear operator $\Psi(\boldsymbol{x}; \boldsymbol{r}) : \mathbb{Y} \to \mathbb{Z}$ has the property that for each $\boldsymbol{r} \in \widehat{\mathbb{X}}$, and any sequence $\boldsymbol{x}_i \to \boldsymbol{r}$, $\boldsymbol{x}_i \in \widehat{\mathbb{X}} \setminus \{\boldsymbol{r}\}$,*

$$\text{if} \quad \|\boldsymbol{x}_i - \boldsymbol{r}\|\Psi(\boldsymbol{x}_i; \boldsymbol{r})\boldsymbol{F}(\boldsymbol{x}_i) \to 0 \quad \text{implies} \quad \boldsymbol{F}(\boldsymbol{x}_i) \to 0, \quad i \to \infty, \tag{2.13}$$

then Ψ defines a deflation operator.

Based on this rule, several deflation operators have been suggested in [13]. In what follows, we adopt the shifted deflation operator:

$$\Psi(\boldsymbol{x};\boldsymbol{r}) = \frac{\mathbb{I}}{\|\boldsymbol{x}-\boldsymbol{r}\|^2} + \mathbb{I}, \tag{2.14}$$

where $\mathbb{I}$ is the identity operator on $\mathbb{Y}$.

2.4. **Summary of the spectral LSTR-Deflation algorithm.** In summary, we have the following algorithm for finding multiple solutions.

Spectral LSTR-Deflation Algorithm for Computing Multiple Solutions

Input: Given initial guess $\boldsymbol{x}^{(0)}$, $\epsilon \geq 0, 0 < \delta_1 < \delta_2 < 1, 0 < \tau_1 < 1 < \tau_2$, and $h_0 = \|\boldsymbol{g}_0\|$
Input: Initial solution set $S \leftarrow$ the empty set Φ
Output: S
1 while the desired multiple solutions are not found do
2 while first-order optimality threshold or failure criterion does not meet do
3 Compute $\boldsymbol{g}_k$ and $\boldsymbol{G}_k$;
4 If $\|\boldsymbol{g}_k\| \leq \epsilon$ and $|Q(\boldsymbol{x}^{(k)})| \leq \epsilon$, stop;
5 Approximately solve the subproblem (2.7) for $\boldsymbol{s}_k$;
6 Compute $Q(\boldsymbol{x}^{(k)} + \boldsymbol{s}_k)$ and r_k. If $r_k \geq \delta_1$, then $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \boldsymbol{s}_k$; Otherwise, set $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)}$;
7 If $r_k < \delta_1$, then $h_{k+1} = \tau_1 h_k$;
  If $r_k > \delta_2$ and $\|\boldsymbol{s}_k\| = h_k$, then $h_{k+1} = \tau_2 h_k$;
  Otherwise, set $h_{k+1} = h_k$;
8 Generate $\boldsymbol{G}_k$, update $q^{(k)}$, set $k := k + 1$, go to step 3;
9 end
10 **if** Convergence threshold met **then**
11 Add converged solution $\tilde{\boldsymbol{x}}$ to S
12 $\hat{\boldsymbol{F}}(\boldsymbol{x}) \leftarrow \Psi(\boldsymbol{x};\tilde{\boldsymbol{x}})\boldsymbol{F}(\boldsymbol{x})$ (i.e. **Deflation**), and it goes back to step 2.
13 **end**
14 end
15 return S

In the above, we have $\boldsymbol{g}_k = \boldsymbol{g}(\boldsymbol{x}^{(k)})$ and $\boldsymbol{G}_k = \boldsymbol{G}(\boldsymbol{x}^{(k)})$. For the trust-region subproblem (see Line 5), an efficient implementation for its solution is the so-called dogleg method (see [15,26]) with the process:

$$q^{(k)}(\boldsymbol{x}^{(k)} - l_k\boldsymbol{g}_k) = Q(\boldsymbol{x}^{(k)}) - l_k\|\boldsymbol{g}_k\|^2 + \frac{1}{2}l_k^2\boldsymbol{g}_k^\top\boldsymbol{G}_k\boldsymbol{g}_k,$$

where based on the exact line search, we have

$$l_k = \frac{\|\boldsymbol{g}_k\|^2}{\boldsymbol{g}_k^\top\boldsymbol{G}_k\boldsymbol{g}_k}.$$

The corresponding step along the steepest descent direction is

$$\hat{\boldsymbol{s}}_k = -l_k\boldsymbol{g}_k = -\frac{\boldsymbol{g}_k^\top\boldsymbol{g}_k}{\boldsymbol{g}_k^\top\boldsymbol{G}_k\boldsymbol{g}_k}\boldsymbol{g}_k,$$

and the Newton step is

$$\tilde{\boldsymbol{s}}_k = -\boldsymbol{G}_k^{-1}\boldsymbol{g}_k.$$

If $\|\hat{\boldsymbol{s}}_k\|_2 = \|l_k\boldsymbol{g}_k\|_2 \geq h_k$, we take

$$\boldsymbol{s}_k = -\frac{h_k}{\|\boldsymbol{g}_k\|}\boldsymbol{g}_k \quad \text{and} \quad \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \boldsymbol{s}_k. \tag{2.15}$$

If $\|\hat{\boldsymbol{s}}_k\| < h_k$ and $\|\tilde{\boldsymbol{s}}_k\| > h_k$, we take

$$\boldsymbol{s}_k(\lambda) = \hat{\boldsymbol{s}}_k + \lambda(\tilde{\boldsymbol{s}}_k - \hat{\boldsymbol{s}}_k), \quad 0 \le \lambda \le 1. \tag{2.16}$$

As a result, we have

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \boldsymbol{s}_k(\lambda) = \boldsymbol{x}^{(k)} + \hat{\boldsymbol{s}}_k + \lambda(\tilde{\boldsymbol{s}}_k - \hat{\boldsymbol{s}}_k),$$

where λ is obtained by solving the equation

$$\|\hat{\boldsymbol{s}}_k + \lambda(\tilde{\boldsymbol{s}}_k - \hat{\boldsymbol{s}}_k)\| = h_k.$$

Otherwise, we set

$$\boldsymbol{s}_k = \tilde{\boldsymbol{s}}_k = -\boldsymbol{G}_k^{-1}\boldsymbol{g}_k. \tag{2.17}$$

In summary, combining (2.15), (2.16) and (2.17) yields

$$\boldsymbol{s}_k = \begin{cases} -\dfrac{\boldsymbol{g}_k}{\|\boldsymbol{g}_k\|}h_k, & \text{if } \|\hat{\boldsymbol{s}}_k\|_2 \ge h_k, \\ \hat{\boldsymbol{s}}_k + \lambda(\tilde{\boldsymbol{s}}_k - \hat{\boldsymbol{s}}_k), & \text{if } \|\hat{\boldsymbol{s}}_k\|_2 < h_k \text{ and } \|\tilde{\boldsymbol{s}}_k\|_2 > h_k, \\ -\boldsymbol{G}_k^{-1}\boldsymbol{g}_k, & \text{if } \|\hat{\boldsymbol{s}}_k\|_2 < h_k \text{ and } \|\tilde{\boldsymbol{s}}_k\|_2 \le h_k. \end{cases} \tag{2.18}$$

Some additional remarks about the implementation of the Spectral LSTR-Deflation Algorithm are in order.

- As highlighted previously, the choice of the trust-region radius h_k in the LSTR method is important. The ratio r_k in (2.8) is used as the criterion for expansion and contraction of the region.
- For the original nonlinear algebraic system $\boldsymbol{F}$ (see Line 12 of the Algorithm), the solution $\tilde{\boldsymbol{x}}$ is found from initial guess $\boldsymbol{x}^{(0)}$ with the Fréchet derivative $\boldsymbol{F}'(\tilde{\boldsymbol{x}})$ being nonsingular. While here the deflation operator Ψ plays a key role in finding multiple solutions other than $\tilde{\boldsymbol{x}}$. Let $\hat{\boldsymbol{F}}(\boldsymbol{x}) = \Psi(\boldsymbol{x}; \tilde{\boldsymbol{x}})\boldsymbol{F}(\boldsymbol{x})$, then this deflated problem $\hat{\boldsymbol{F}}(\boldsymbol{x})$ satisfies two properties: (i) The preservation of solutions of $\hat{\boldsymbol{F}}$ should be hold, i.e. for $\boldsymbol{x} \neq \tilde{\boldsymbol{x}}$, $\hat{\boldsymbol{F}}(\boldsymbol{x}) = 0$ iff $\boldsymbol{F}(\boldsymbol{x}) = 0$; (ii) In view of Theorem 2.2, the iterative algorithm applied to $\hat{\boldsymbol{F}}(\boldsymbol{x})$ will not find $\tilde{\boldsymbol{x}}$ again, when

  $$\lim_{\boldsymbol{x} \to \tilde{\boldsymbol{x}}} \inf \|\hat{\boldsymbol{F}}(\boldsymbol{x})\| > 0,$$

  i.e., the deflated problem $\hat{\boldsymbol{F}}(\boldsymbol{x})$ does not converge to zero for any sequence converging to the known solution $\tilde{\boldsymbol{x}}$.
- In general, based on the deflation technique, the same initial guess allows us to find new solutions in most cases. However, it is noteworthy that the deflation procedure may diverge with the same initial guess in some cases. We find the choice of a random deviation from the obtained solution as the initial guess turns out sufficient to find all other solutions (see Section 3).
- Although the original and deflated nonlinear systems can be solved by other methods, e.g., Newton iteration, the trust-region method is deemed as the method of choice for its flexibility in initial inputs and efficiency in dealing with the increasingly complicated systems in the course of repeating the deflation technique. We shall testify this in the forthcoming section.

3. Applications to differential equations with multiple solutions

In this section, we apply the spectral LSTR-Deflation algorithm to various differential equations with multiple solutions and compare its performance in efficiency and accuracy with some related methods. In the numerical experiments, we choose $\delta_1 = 0.25, \delta_2 = 0.75, \tau_1 = 0.5$, $\tau_2 = 2$ and $\epsilon = 10^{-13}$ in the algorithm described in Subsection 2.4. The codes are carried out on a server: Intel(R) Core(TM) i7-7500U (2.90 GHz) and 120GB RAM via MATLAB (version R2015b).

3.1. **ODE Examples.**

3.1.1. *Viscous flow in a porous channel with expanding or contracting walls.* When concerning systemic circulation in blood circulation, the blood in the left ventricle is being forced into the aorta by systole and the mitral valve between left ventricle and left atrium is closed. At this juncture the left ventricle forms a vessel with one end closed. Meanwhile, the mass transfer of the vessel between inside and outside can be achieved by the seepage across permeable wall of the vessel. The mathematical model established by Majdalani [9] is as follows

$$u^{(4)}(y) + \alpha(yu'''(y) + 3u''(y)) + Re\big(u(y)u'''(y) - u'(y)u''(y)\big) = 0, \quad y \in (0,1), \tag{3.1}$$

supplemented with the boundary conditions

$$u(0) = 0, \quad u''(0) = 0, \quad u(1) = 1, \quad u'(1) = 0, \tag{3.2}$$

where Re is called as the cross-flow Reynolds number ($Re > 0$ for injection and $Re < 0$ for suction), α is the wall expansion ratio and the prime denotes differentiation with respect to x. Here $u(y)$ and $u'(y)$ represent the normal and streamwise velocities in a porous channel, respectively. this problem admits multiple solutions for $Re < -12.165$ and any α, but so far there are only three solutions identified in literature [23, 34].

(i) Legendre spectral Petrov-Galerkin discretisation. As shown in [25], the spectral method is advantageous for high-order problems. In view of the non-symmetric boundary conditions (3.2), we employ the Petrov-Galerkin method using compact combinations of Legendre polynomials that meet the boundary conditions and the dual counterparts. For convenience, we make a change of variable: $y = (1+x)/2$, and convert (3.1)-(3.2) into

$$\begin{cases} \breve{u}^{(4)} + \delta(1+x)\breve{u}''' + \beta\breve{u}'' + \gamma F(\breve{u}) = 0, \quad x \in (-1,1), \\ \breve{u}(-1) = \breve{u}''(-1) = 0, \quad \breve{u}(1) = 1, \quad \breve{u}'(1) = 0, \end{cases} \tag{3.3}$$

where $\breve{u}(x) = u(y)$, $\delta = \frac{\alpha}{4}$, $\beta = \frac{3}{2}\alpha$, $\gamma = \frac{Re}{2}$, and the nonlinear term:

$$F(\breve{u}) := \breve{u}\breve{u}''' - \breve{u}'\breve{u}'' = \frac{1}{2}(\breve{u}^2)''' - 2((\breve{u}')^2)'. \tag{3.4}$$

To deal with the non-homogeneous boundary conditions, we introduce

$$p(x) = \frac{3}{4}(1+x) - \frac{1}{16}(1+x)^3, \quad \text{s.t.,} \quad p(-1) = p''(-1) = p'(1) = 0, \ \ p(1) = 1.$$

Set $\breve{u} = v + p$, and note

$$F(v+p) = F(v) + pv''' - p'v'' - p''v' + p'''v + pp''' - p'p''.$$

Then direct calculation leads to

$$\begin{cases} v^{(4)} + A(x)v''' + B(x)v'' + C(x)v' - \dfrac{3}{8}\gamma v + \gamma F(v) = g(x), \quad x \in (-1,1), \\ v(\pm 1) = v'(1) = v''(-1) = 0, \end{cases} \tag{3.5}$$

where

$$\begin{aligned} A(x) &= \delta(1+x) + \gamma p(x) = \Big(\frac{3\gamma}{4} + \delta\Big)(1+x) - \frac{\gamma}{16}(1+x)^3, \\ B(x) &= \beta - \gamma p'(x) = \beta - \frac{3\gamma}{4} + \frac{3\gamma}{16}(1+x)^2, \quad C(x) = \frac{3\gamma}{8}(1+x), \\ g(x) &= \frac{3}{8}(\delta+\beta)(1+x) - \gamma p p''' + \gamma p' p'' = \frac{3}{8}(\delta+\beta)(1+x) + \frac{3\gamma}{64}(1+x)^3. \end{aligned} \tag{3.6}$$

Let $\mathbb{P}_N$ be the set of polynomials of degree $\le N$, and define

$$\begin{aligned} V_N &:= \big\{\phi \in \mathbb{P}_N : \phi(\pm 1) = \phi'(1) = \phi''(-1) = 0\big\}, \\ W_N &:= \big\{\psi \in \mathbb{P}_N : \psi(\pm 1) = \psi'(\pm 1) = 0\big\}. \end{aligned} \tag{3.7}$$

The Legendre spectral Petrov-Galerkin scheme for (3.5)-(3.6) is to find $v_N \in V_N$ such that

$$\mathcal{B}(v_N, \psi) + \mathcal{N}(v_N, \psi) = (g, \psi), \quad \forall \psi \in W_N, \tag{3.8}$$

where the bilinear form and the approximation to the nonlinear term are

$$\begin{aligned} \mathcal{B}(v_N, \psi) &:= (v_N'', \psi'') + (v_N', (A\psi)'') - (v_N', (B\psi)') + (Cv_N', \psi) - \frac{3\gamma}{8}(v_N, \psi), \\ \mathcal{N}(v_N, \psi) &:= \frac{\gamma}{2}\big([I_N v_N^2]', \psi''\big) + 2\gamma\big(I_N[v_N']^2, \psi'\big), \end{aligned} \tag{3.9}$$

with I_N being the Legendre-Gauss-Lobatto interpolation operator. Here, we use (3.4) to formulate $\mathcal{N}(v_N, \psi)$.

To construct basis functions, we now recall some properties of Legendre polynomials (cf. [25]):

$$\begin{aligned} &(2n+1)L_n(x) = L_{n+1}'(x) - L_{n-1}'(x) = J_{n-1}'(x), \\ &(1-x^2)L_n'(x) = \frac{n(n+1)}{2n+1}\big(L_{n-1}(x) - L_{n+1}(x)\big) = -\frac{n(n+1)}{2n+1}J_{n-1}(x), \end{aligned} \tag{3.10}$$

and

$$L_n(\pm 1) = (\pm 1)^n, \quad L_n'(\pm 1) = \frac{1}{2}(\pm 1)^{n-1} n(n+1), \tag{3.11}$$

where we denoted

$$J_n(x) = L_{n+2}(x) - L_n(x), \quad n \ge 0. \tag{3.12}$$

Direct calculation yields

$$\begin{aligned} &J_n'(1) = (2n+3)L_{n+1}(1) = 2n+3, \\ &J_n''(-1) = (2n+3)L_{n+1}'(-1) = \frac{1}{2}(-1)^n(n+1)(n+2)(2n+3). \end{aligned} \tag{3.13}$$

Now, we construct the basis functions for V_N using the compact combination:

$$\varphi_k(x) = J_k(x) + a_k J_{k+1}(x) + b_k J_{k+2}(x), \quad 0 \le k \le N-4, \tag{3.14}$$

where $\{a_k, b_k\}$ are determined by the homogenous boundary conditions in V_N. In view of the second formula in (3.10), we have $\varphi_k(\pm 1) = 0$. From (3.10)-(3.11) and $\varphi_k'(1) = \varphi_k''(-1) = 0$, we can obtain

$$J_{k+1}'(1)\, a_k + J_{k+2}'(1)\, b_k = -J_k'(1), \quad J_{k+1}''(-1)\, a_k + J_{k+2}''(-1)\, b_k = -J_k''(-1),$$

which, together with (3.13), implies

$$\begin{aligned} a_k &= \frac{J'_{k+2}(1)J''_k(-1) - J'_k(1)J''_{k+2}(-1)}{J'_{k+1}(1)J''_{k+2}(-1) - J'_{k+2}(1)J''_{k+1}(-1)} = -\frac{2k+3}{(k+3)^2}, \\ b_k &= \frac{J'_k(1)J''_{k+1}(-1) - J'_{k+1}(1)J''_k(-1)}{J'_{k+1}(1)J''_{k+2}(-1) - J'_{k+2}(1)J''_{k+1}(-1)} = -\frac{(k+2)^2(2k+3)}{(k+3)^2(2k+7)}. \end{aligned} \tag{3.15}$$

Similarly, we can also construct the basis functions for W_N :

$$\psi_k(x) = J_k(x) + c_k J_{k+2}(x), \quad c_k = -\frac{2k+3}{2k+7}. \tag{3.16}$$

We expand the numerical solution as

$$u_N(x) := u_N(x;\alpha,Re) = p(x) + v_N(x) = p(x) + \sum_{k=0}^{N-4} \tilde{v}_k \varphi_k(x). \tag{3.17}$$

Under these basis functions, we can easily show that the matrix of the linear part $\mathcal{B}(\varphi_k,\psi_j)$ in (3.8) is sparse with a finite bandwidth and its entries can be evaluated explicitly. It is evident that the righthand sided vector $\{(g,\psi_j)\}_{j=0}^{N-4}$ can be computed exactly as well. Then we solve the nonlinear system by the LSTR-Deflation technique summarized in Subsection 2.4, where we choose the initial guess

$$\tilde{v}_k^{(0)} = 1, \quad 0 \le k \le N-4, \tag{3.18}$$

denoted by IG = ones$(N-3)$. Note that in the iterations, the nonlinear part $\{\mathcal{N}(v_N,\psi_j)\}_{j=0}^{N-4}$ can be treated efficiently by using the pseudospectral technique as in [25, Ch. 4].

(ii) Numerical results. In Table 1, we tabulate the outcomes of our algorithm for four pairs of parameters (α, Re) and the above initial guess with $N = 18$, where we list the number of solutions labeled by I,II,$\cdots$, the computational time and iteration number of the LSTR method. Some observations and highlights are as follows.

i) It is noteworthy that in [23,34], the plots of the solutions for (α, Re)=(0, -20), (2,-40) using different methods were reported. Here, we obtain the solutions in Figure 3.1 efficiently in seconds from the same initial guess IG.

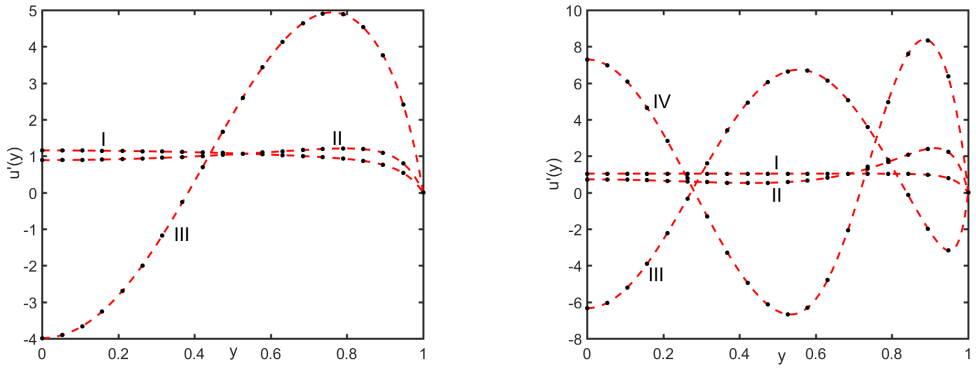
ii) We find a new solution for (α, Re)=(8, -40), see Solution-IV in Figure 3.1.

iii) The algorithm is quite accurate with a fast convergence as shown in Table 2, where we compare the errors in maximum norm with the numerical solution obtained with a relatively large N.

TABLE 1. Performance of our spectral LSTR-Deflation method.

(α, Re)	No.s	Types	Time(s)	n_{it}	(α, Re)	No.s	Types	Time(s)	n_{it}
(0, -20)	3	I	3.4567	24	(2, -40)	3	I	3.2326	24
		II	2.7378	17			II	5.2305	43
		III	2.2060	12			III	4.4526	34
(-2, -40)	3	I	4.0644	31	(8, -40)	4	I	3.0578	22
		II	4.1458	31			II	4.1466	35
		III	4.4088	33			III	4.1352	34
							IV	5.1466	42

As a comparison, we replace the LSTR algorithm for solving the nonlinear systems by the Newton iteration (as in [13] using Newton-Deflation technique). We measure the magnitude of



(a) $\alpha = 0, Re = -20$. (b) $\alpha = 8, Re = -40$.

FIGURE 3.1. Multiple solutions of (3.1)-(3.2) with different (α, Re).

TABLE 2. Accuracy of our spectral LSTR-Deflation method.

(α, Re)	N	I	II	II	(α, Re)	N	I	II	III	IV
	8	7.71e-2	1.06e-1	2.36e-3		8	3.15e-2	1.23e-2	4.25e-2	-
(0, -20)	16	6.32e-5	1.02e-5	3.16e-7	(2, -40)	16	3.67e-5	7.21e-5	3.81e-5	-
	24	5.21e-9	1.78e-8	2.08e-10		24	4.31e-8	9.23e-7	3.71e-9	-
	8	2.81e-2	3.61e-1	4.27e-1		8	1.49e-1	3.07e-1	2.89e-1	9.31e-1
(-2, -40)	16	5.31e-5	2.39e-5	3.86e-7	(8, -40)	16	1.83e-5	4.31e-5	1.09e-5	4.81e-6
	24	3.89e-8	1.84e-8	4.61e-9		24	3.91e-8	3.92e-9	9.18e-9	1.78e-8

$\|\boldsymbol{F}(\boldsymbol{x})\|$ in (2.6) for three choices of initial guesses IG, $0.1 * \text{IG}$ and cos(IG). As shown in Table 3, the Newton iterations fail to converge for these initial inputs, while $\|\boldsymbol{F}(\boldsymbol{x})\|$ of the LSTR method descends very fast. However, it is noteworthy that if we choose the initial inputs to be a small perturbation of the numerical solution from our algorithm, the Newton-Deflation method indeed converges.

TABLE 3. A comparison between Newton iteration in [13] and the LSTR method.

	Newton iteration in [13]				LSTR method			
(α, Re)	n_{it}	IG	$0.1 * \text{IG}$	cos(IG)	n_{it}	IG	$0.1 * \text{IG}$	cos(IG)
	1	4.97e5	2.02e3	3.13e6	10	2.57e-3	3.72e-13	8.87e1
	2	3.10e9	8.59e3	9.91e6	15	6.50e-5	-	2.45e-4
(8, -40)	3	3.39e10	5.07e4	2.17e9	20	4.41e-13	-	4.98e-6
	4	2.64e13	2.16e5	3.03e11	25	-	-	6.91e-9
	5	1.99e15	1.47e5	5.66e13	30	-	-	1.86e-13

3.1.2. *Models with polynomial and exponential nonlinearities.* We consider the model problem:

$$u'' + \lambda F(u) = 0, \quad x \in (0, 1), \quad \lambda > 0, \tag{3.19}$$

with boundary conditions to be specified later. Here, we focus on $F(u) = e^u, 1 + u^p$ (where p is a positive integer).

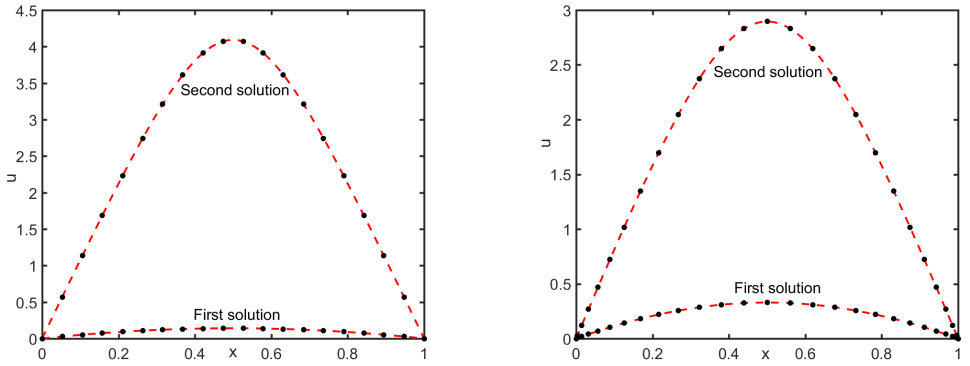
Case 1: The equation (3.19) with $F(u) = e^u$ is known as the Bratu-Gelfand model [10], which together with the boundary conditions $u(0) = u(1) = 0$, has two solutions when $0 < \lambda < \lambda^*(\lambda^* =$

3.51383), but no solution for $\lambda \geq \lambda^*$. Here we assume that $0 < \lambda < \lambda^*$. The spectral-Galerkin discretisation follows [25, Ch. 4].

TABLE 4. Performance of the spectral LSTR-Deflation methods.

		First solution			Second solution		
λ	N	n_{it}	Time(s)	Error	n_{it}	Time(s)	Error
1	8	3	0.0367	3.1159e-5	20	0.1623	6.1052e-5
	16	4	0.0492	6.2209e-8	26	0.1878	4.9821e-8
	24	6	0.0512	4.5019e-10	28	0.1886	6.2309e-10
2	8	4	0.0501	5.6106e-5	26	0.1526	5.2130e-5
	16	5	0.0557	6.4205e-8	28	0.1643	7.0831e-8
	24	6	0.0581	6.4092e-10	30	0.1665	9.2105e-10

In Table 4, we list the LSTR iteration number, computational time and numerical errors compared with the reference solution (obtained from refined grids) with $\lambda = 1, 2$ and the initial guess $-\cos(\text{IG})$. It indicates clearly that this method finds these two solutions, illustrated in Figure 3.2, very fast and accurately.



(a) Multiple solutions with $\lambda = 1$. (b) Multiple solutions with $\lambda = 2$.

FIGURE 3.2. Multiple solutions of the Bratu-Gelfand model for u.

In Figure 3.3, we make a similar comparison and record the magnitude of $\|\boldsymbol{F}(\boldsymbol{x}^{(i)})\|$ for the first several iterative steps in the Newton-Deflation method. Again we observe the divergence and sensitivity of the Newton iteration to the initial guess.

Remark 3.1. *Observe from* Figure 3.2 *that the two solutions satisfy* $u'(0) > 0$ *and* $u'(1) < 0$. *In fact, it is true for all* $\lambda \in (0, \lambda^*)$. *It is evident that by* (3.19) *with* $F(u) = e^u$, *we have* $u''(x) < 0$, *so* $u'(1) < u'(x) < u'(0)$ *for all* $x \in (0, 1)$. *Note that if* $u'(0) \leq 0$, *then* $u(x)$ *is strictly decreasing in* $(0, 1)$, *so* $u(1) < u(x) < u(0)$. *This leads to the contradiction, which implies* $u'(0) > 0$. *Similarly, we can show* $u'(1) < 0$. □

Case 2: We consider (3.19) with $F(u) = 1 + u^p$ and the boundary conditions: $u'(0) = u(1) = 0$. In the spectral-Galerkin discretization, we use the following basis functions to meet the boundary conditions

$$\psi_k(x) = L_k(x) - \frac{k^2 + 6k + 7}{2(k+2)^2} L_{k+1}(x) - \frac{(k+1)^2}{2(k+2)^2} L_{k+2}(x),$$

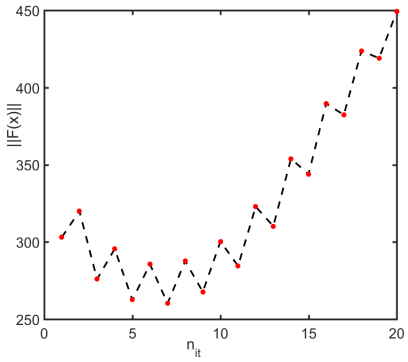


FIGURE 3.3. The magnitude of $\|\boldsymbol{F}(\boldsymbol{x})\|$ at various iterative steps of the Newton-Deflation method.

which satisfies $\psi_k'(-1) = \psi_k(1) = 0$ for any k.

We compare the performance of our proposed method with that of the bootstrapping algorithm composed of the homotopy and two-grid techniques in [18]. We quote [18, Table 1] in the left half of Table 5, where n_1 (resp. n_2) is the number of grid points of the whole interval (resp. number of grid points in each of the n_1-1 subintervals), together with the numerical errors and computational time for two solutions for $\lambda = 1.2$ and $p = 4$ in $F(u)$. On the right half of Table 5, we list the accuracy and computational time using our method, which shows the distinctive advantages of the latter in terms of time-saving and faster convergence.

TABLE 5. A comparison between the bootstrapping method and the LSTR-Deflation method with $p = 4$ and $\lambda = 1.2$.

	Bootstrapping in [18, Table 1]				The LSTR-Deflation method			
λ	(n_1, n_2)	I soln err.	II soln err.	Time(s)	N	I soln err.	II soln err.	Time(s)
1.2	(7, 3)	1.1053e-4	3.7146e-4	4m5s	8	5.6106e-5	2.2891e-5	0.9896s
	(21, 2)	3.2364e-5	1.1336e-4	31m16s	16	4.4109e-8	2.4166e-8	0.3212s
	(42, 2)	8.3840e-6	3.0290e-5	5h39m58s	24	6.6527e-11	5.2984e-9	0.6717s

Notably, the proposed approach allows us to locate more solutions if $p = 3$ but $\lambda = 1.2$ remains (see Figure 3.4 (right)) that seem not been reported in literature.

3.2. **PDE Examples.**

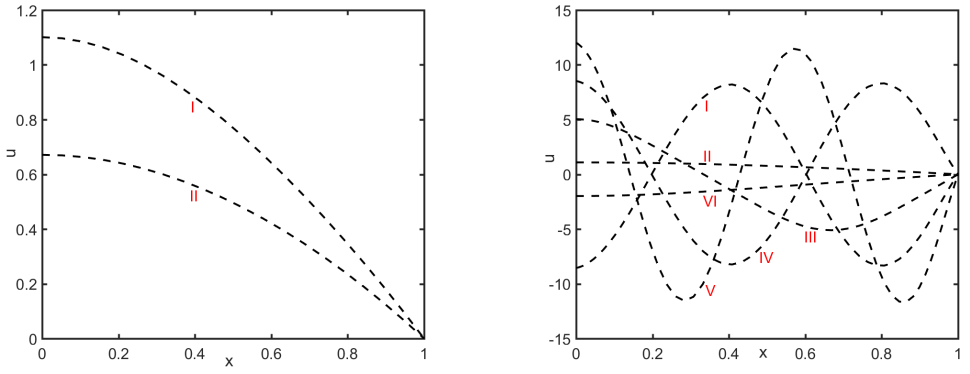
3.2.1. *Phase separation: Allen-Cahn equation.* As in Farrell et al. [13] and [12], we consider the steady-state Allen-Cahn equation:

$$-\epsilon\Delta u + \epsilon^{-1}(u^3 - u) = 0 \quad \text{in} \quad \Omega = (0,1)\times(0,1), \tag{3.20}$$

where the boundary conditions are as follows

$$\begin{cases} u = 1 & \text{on } \partial\Omega_1 := \{x = 0, 1;\ 0 < y < 1\}, \\ u = -1 & \text{on } \partial\Omega_2 := \{y = 0, 1;\ 0 < x < 1\}. \end{cases} \tag{3.21}$$

Here, ϵ is a parameter relating the strength of the free surface tension to the potential term in the free energy. It is evident that $u = \pm 1$ are solutions to (3.20), which correspond two different materials. We can verify that the multiple solutions of (3.20)-(3.21) are invariant under a rotation and reflection. More precisely, if $u(x, y)$ is a solution of (3.20)-(3.21), so is $-u(1 - y, x)$.

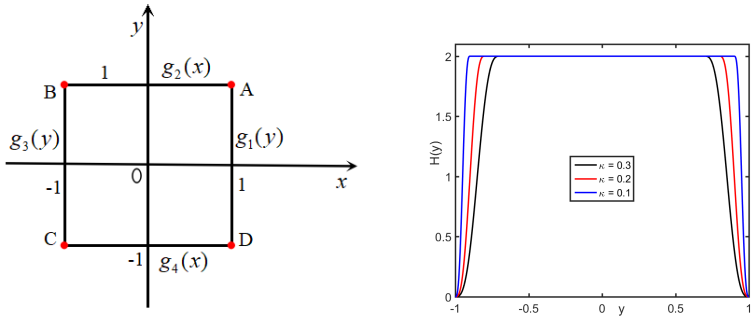


(a) Multiple solutions with $p = 4$. (b) Multiple solutions with $p = 3$.

FIGURE 3.4. Multiple solutions for $\lambda = 1.2$.

As shown in (3.21), the boundary conditions are not compatible at four corners of $\bar{\Omega}$. To facilitate the spectral-Galerkin discretisation, we properly smooth out the jumps around the corners and render the boundary conditions compatible. For this purpose, we make a simple linear mapping: $x \to 2x - 1$, $y \to 2y - 1$ and convert the problem (3.20)-(3.21) into $\Omega = (-1, 1)^2$. With a little abuse of notation, we still use u to denote the unknown solution. Setting $v = u + 1$ leads to

$$\begin{cases} -4\epsilon\Delta v + \epsilon^{-1}((v-1)^3 - v + 1) = 0 \quad \text{in } \Omega = (-1,1) \times (-1,1), \\ v = 2 \quad \text{on } \partial\Omega_1 = \{x = \pm 1,\ -1 < y < 1\}, \\ v = 0 \quad \text{on } \partial\Omega_2 = \{y = \pm 1,\ -1 < x < 1\}. \end{cases} \tag{3.22}$$



(a) Boundary conditions. (b) $H(y)$ vs κ.

FIGURE 3.5. Boundary conditions and $H(y)$ for (3.22).

We smooth out $v = 2$ by $H(y) \in C^2[-1, 1]$ defined by

$$H(y) = \begin{cases} 2\Big[\dfrac{1}{2} + \dfrac{1}{2\kappa}(2y + 2 - \kappa) + \dfrac{1}{2\pi}\sin\Big(\dfrac{\pi}{\kappa}(2y + 2 - \kappa)\Big)\Big], & y \in [-1, -1 + \kappa], \\ 2, & y \in (-1 + \kappa, 1 - \kappa), \\ 2 - 2\Big[\dfrac{1}{2} + \dfrac{1}{2\kappa}(2y - 2 + \kappa) + \dfrac{1}{2\pi}\sin\Big(\dfrac{\pi}{\kappa}(2y - 2 + \kappa)\Big)\Big], & y \in [1 - \kappa, 1], \end{cases} \tag{3.23}$$

where $\kappa > 0$ is a small parameter, see Figure 3.5 (b). As a result, we derive the following boundary conditions illustrated in Figure 3.5 (a):

$$g_1(y) = g_3(y) = H(y), \qquad g_2(x) = g_4(x) = 0. \tag{3.24}$$

In order to transform the non-homogeneous boundary conditions into homogeneous conditions, we introduce (see [17] for general non-homogeneous boundary conditions)

$$\begin{aligned} G(x,y) = \frac{1}{4}\Big\{ & 2(1-y)g_4(x) + 2(1+y)g_2(x) + 2(1-x)g_3(y) + 2(1+x)g_1(y) \\ & - (1-x)(1-y)g_3(-1) - (1-x)(1+y)g_2(-1) \\ & - (1+x)(1-y)g_4(1) - (1+x)(1+y)g_1(1)\Big\}, \end{aligned} \tag{3.25}$$

which satisfies

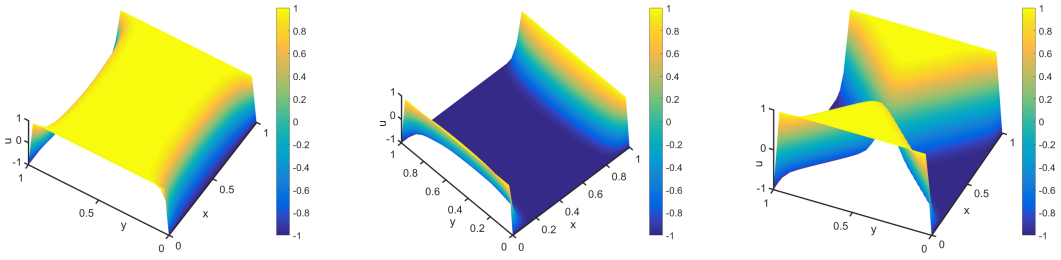
$$G(1,y) = g_1(y), \quad G(x,1) = g_2(x), \quad G(-1,y) = g_3(y), \quad G(x,-1) = g_4(x).$$

Setting $v = w + G$, we obtain from (3.22) that

$$\begin{cases} -4\epsilon\Delta w + \epsilon^{-1}[(w+G-1)^3 - (w+G-1)] = 4\epsilon\Delta G \quad \text{in } \Omega, \\ w = 0 \quad \text{on } \partial\Omega. \end{cases} \tag{3.26}$$

Then we can follow the Legendre spectral-Galerkin method in Subsection 2.1 to discretize (3.26). We omit the details.

In the following tests, we take $\epsilon = 0.04$ and $\kappa = 0.1$. We plot in Figure 3.6 the three solutions obtained by our algorithm, whose shapes and patterns are similar to those in [13, 28] (however in [28], only types I and II solutions were reported) without smoothing out the boundary conditions. In Table 6, we list the number of iterations and computational time of the LSTR method for each solution which shows the high efficiency of the algorithm. It is seen that the first two solutions are identified very fast in a few iterations, but the third solution is relatively expensive and harder to locate. In fact, some care is needed for the initial guess for this solution where we randomly perturb the obtained solution II. We also see that the obtained solutions verify the rotational and reflectional property of the solution, where Type-I and Type-II are a pair, and Type-III is invariant under the transforms.



(a) I (b) II (c) III

FIGURE 3.6. Three solutions of the steady Allen-Cahn equation.

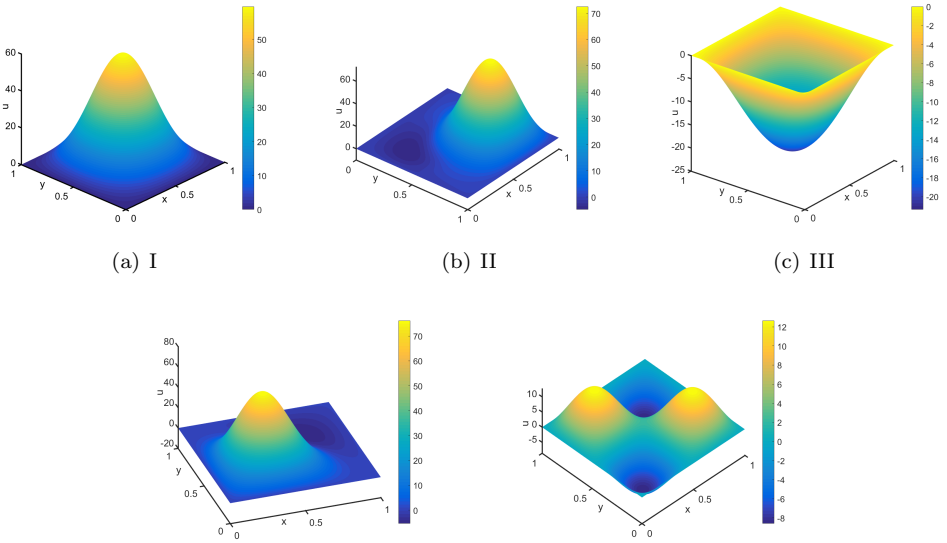
TABLE 6. Performance of the spectral LSTR-Deflation method.

Type	n_{it}	Time(s)	Type	n_{it}	Time(s)	Type	n_{it}	Time(s)
I	39	5.2710	II	23	4.7952	III	160	15.2973

3.2.2. *A model problem analyzed in Breuer, McKenna and Plum [3].* We consider the following problem studied in [3]:

$$\begin{cases} \Delta u + u^2 = 800 \sin(\pi x) \sin(\pi y) & \text{in } \Omega = (0,1) \times (0,1), \\ u = 0 & \text{on } \partial\Omega. \end{cases} \tag{3.27}$$

Allgower et al. [1] claimed that (3.27) has at least four solutions, and these solutions are symmetric



(a) I (b) II (c) III

(d) IV (e) V

FIGURE 3.7. Multiple solutions of (3.27).

(i.e. symmetric with respect to reflections about the axes $x = \frac{1}{2}$, $y = \frac{1}{2}$, $x = y$ and $x = 1 - y$). Indeed, four solutions (i.e. I, II, III and IV) obtained from our algorithm in Figure 3.7 satisfy such property, which were also presented in [1, 39]. More importantly, we find the fifth solution (i.e., V in Figure 3.7 (e)), which verifies the claim in [1]. Again as shown in Table 7, the proposed algorithm has an excellent performance.

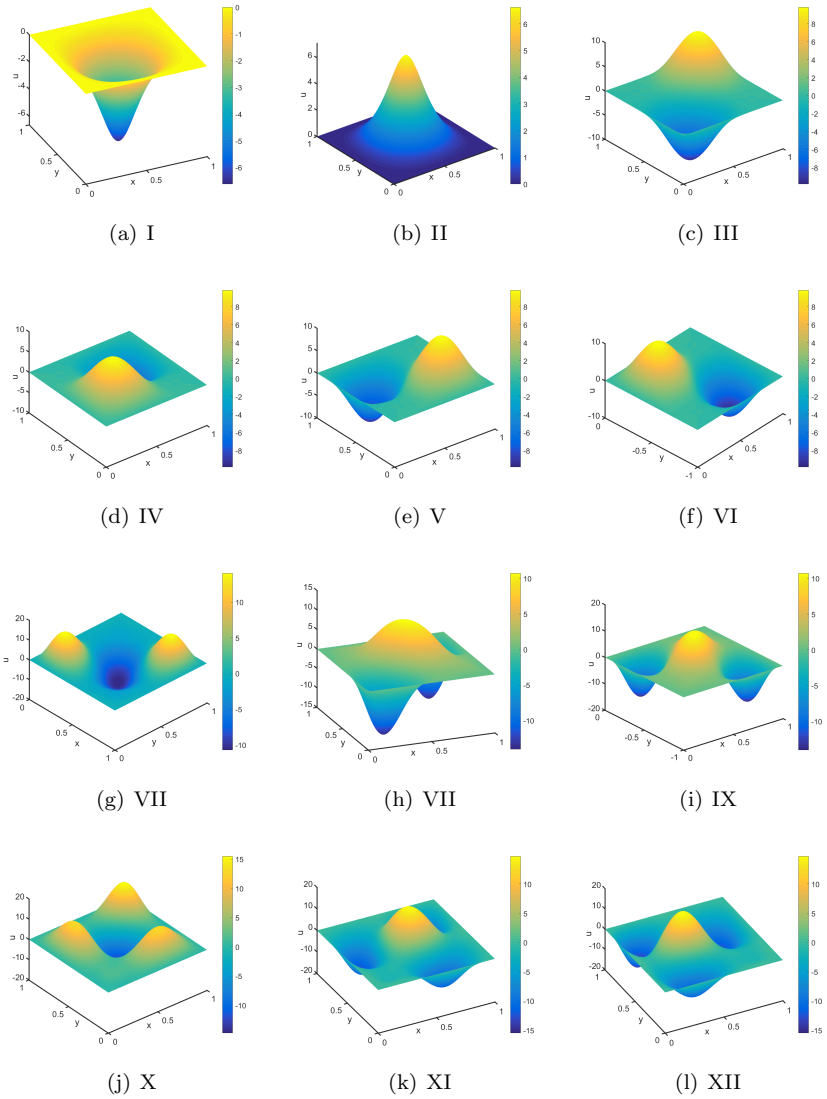
TABLE 7. Numerical results for (3.27).

Types	n_{it}	Time(s)	Symmetry	Types	n_{it}	Time(s)	Symmetry
I	25	3.5421	$x = y$	IV	30	4.2418	$x = \frac{1}{2}$
II	39	5.2191	$y = \frac{1}{2}$	V	37	4.6829	$x = y, x = 1 - y$
III	28	3.5935	$x = \frac{1}{2}, y = \frac{1}{2}$				

3.2.3. *The Henon equation modeling spherical stellar systems.* Finally we consider the Henon equation:

$$\begin{cases} \Delta u + u^3 = 0 & \text{in } \Omega = (0,1)^2, \\ u = 0 & \text{on } \partial\Omega, \end{cases} \tag{3.28}$$

as in [6]. It is known that the Henon equation (3.28) has infinite many solutions. Meanwhile, we further describe the property of solutions to (3.28): If $u(x, y)$ is a solution of (3.28), another solution can be obtained by reflecting the $xy-$plane or rotating it an integer multiple of 90 degrees around the point $(1/2, 1/2)$. Therefore, in the following tests, we intend to verify it and locate some distinct solutions.



(a) I (b) II (c) III

(d) IV (e) V (f) VI

(g) VII (h) VII (i) IX

(j) X (k) XI (l) XII

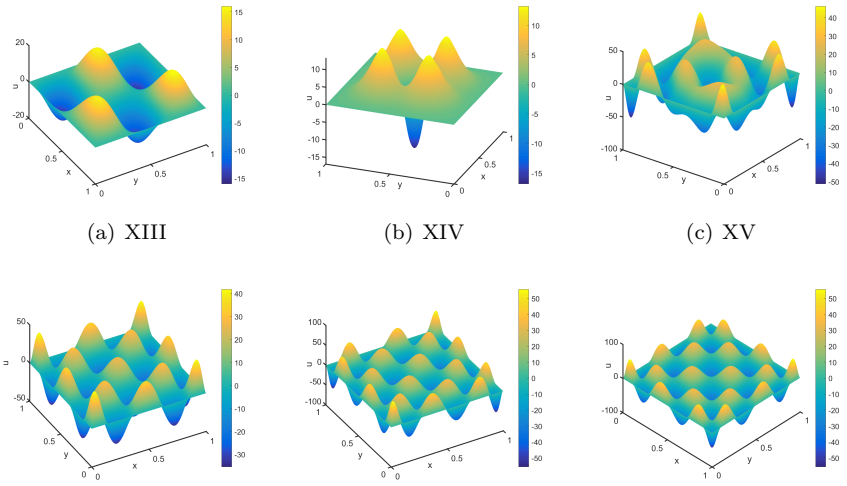
FIGURE 3.8. Twelve solutions of (3.28).

Based on our algorithm, as shown in Figure 3.8, twelve solutions are found, and the connections among them are as follows:

$$\text{I} \xrightarrow{\text{Reflected}} \text{II}, \qquad \text{III} \xrightarrow{\text{Reflected}} \text{IV} \xrightarrow[90 \text{ degree}]{\text{rotated by}} \text{V} \xrightarrow[90 \text{ degree}]{\text{rotated by}} \text{VI},$$

$$\text{VII} \xrightarrow{\text{Reflected}} \text{VIII} \xrightarrow[90 \text{ degree}]{\text{rotated by}} \text{IX}, \qquad \text{X} \xrightarrow{\text{Reflected}} \text{XI} \xrightarrow[90 \text{ degree}]{\text{rotated by}} \text{XII},$$

which agrees well with the property of solutions. On the other hand, we also locate other some distinct solutions depicted in Figure 3.9. Similarly, more solutions can be also obtained by the property of solutions, which we do not plot them.



(a) XIII (b) XIV (c) XV

(d) XVI (e) XVII (f) XVIII

FIGURE 3.9. Some disctinct solutions of (3.28).

4. CONCLUDING REMARKS

In this paper, we proposed to an efficient and accurate Legendre spectral-Galerkin method to discretise some differential equations with multiple solutions that lead to nonlinear algebraic systems, and then we advocated the nonlinear least-squares and trust-region method to solve the nonlinear systems and employ the deflation technique to search for multiple solutions. We demonstrated that this integrated algorithm has significant advantages over the existing approaches in accuracy, efficiency and capability of finding new solutions.

Declarations

- **Availability of data and materials:** The data that support the findings of this study are available from the corresponding author upon reasonable request.
- **Authors' contributions:** All authors contributed to the study conception and design. The manuscript was written by Tongke Wang. All authors read and approved the final manuscript.

References

[1] E. L. Allgower, S. G. Cruceanu, and S. Tavener. Application of numerical continuation to compute all solutions of semilinear elliptic equations. *Adv. Geom*, 76(2009):pp. 1–10.

[2] E. L. Allgower, A. J. Sommese D. J. Bates, and C. W. Wampler. Solution of polynomial systems derived from differential equations. *Computing*, 76(2006):pp. 1–10.

[3] B. Breuer, P. J. Mckenna, and M. Plum. Multiple solutions for a semilinear boundary value problem: A computational multiplicity proof. *J. Differ. Equations*, 195(2003):pp. 243–269.

[4] K. M. Brow and W. B. Gearhart. Deflation techniques for the calculation of further solutions of a nonlinear system. *Numer. Math*, 16(1971):pp. 334–342.

[5] C. M. Chen and Z. Q. Xie. Search extension method for multiple solutions of a nonlinear problem. *Comp. Math. Appl.*, 47(2004):pp. 327–343.

[6] C. M. Chen and Z. Q. Xie. Structure of multiple solutions for nonlinear differential equations. *Sci. China. Ser. A.*, 47(2004):pp. 172–180.

[7] X. J. Chen and J. X. Zhou. A local min-max-orthogonal method for finding multiple solutions to noncooperative elliptic systems. *Math. Comp.*, 79(2011):pp. 2213–2236.

[8] Y. S. Choi and P. J. McKenna. A mountain pass method for the numerical solution of semilinear elliptic problems. *Nonlinear. Anal.*, 20(1993):pp. 417–437.

[9] E. C. Dauenhauer and J. Majdalani. Exact self-similarity solution of the navier–stokes equations for a porous channel with orthogonally moving walls. *Phys. Fluids*, 15(2003):pp. 1485–1495.

[10] H. T. Davis. *Introduction to nonlinear differential and integral equations*. US Atomic Energy Commission, 1960.

[11] Z. H. Ding, D. Costa, and G. Chen. A high-linking algorithm for sign-changing solutions of semilinear elliptic equations. *Nonlinear. Anal.*, 38(1999):pp. 151–172.

[12] W. N. E, W. Ren, and E. Vandeneijnden. Minimum action method for the study of rare events. *Comm. Pure Appl. Math.*, 57(2004):pp. 637–656.

[13] P. E. Farrell, A. Birkisson, and S. W. Funke. Deflation techniques for finding distinct solutions of nonlinear partial differential equations. *SIAM J. Sci. Comput.*, 37(2015):pp. A2026–A2045.

[14] U. Frisch, S. Matarrese, R. Mohayaee, and A. Sobolevski. A reconstruction of the initial conditions of the universe by optimal mass transportation. *Nature*, 417(2002):pp. 260–262.

[15] N. Gould, C. Sainvitu, and P. L. Toint. A filter-trust-region method for unconstrained optimization. *SIAM J. Optim*, 16(2006):pp. 341–357.

[16] A. A. Grau. Rounding errors in algebraic processes (J. H. Wilkinson). *SIAM Rev.*, 8(1966):PP. 397–398.

[17] B. Y. Guo and T. J. Wang. Composite Laguerre-Legendre spectral method for exterior problems. *Adv. Comput. Math.*, 32(2010):pp. 393–429.

[18] W. R. Hao, J. D. Hauenstein, B. Hu, and A. J. Sommese. A bootstrapping approach for computing multiple solutions of differential equations. *J. Comput. Appl. Math.*, 258(2014):pp. 181–190.

[19] Y. X. Li and J. X. Zhou. A minimax method for finding multiple critical points and its applications to semilinear pdes. *SIAM J. Sci. Comput.*, 23(2001):pp. 840–865.

[20] P. L. Lions. On the existence of positive solutions of semilinear elliptic equations. *SIAM Rev*, 24(1982):pp. 441–467.

[21] G. Nicolis. Introduction to Non-Linear Science. *Cambridge University Press*, 1995.

[22] T. Ouyang and J. Shi. Exact multiplicity of positive solutions for a class of semilinear problem, II. *J. Differ. Equa.*, 158(1999):pp. 94–151.

[23] W. A. Robinson. The existence of multiple solutions for the laminar flow in a uniformly porous channel with suction at both walls. *J. Eng. Math.*, 10(1976):pp. 23–40.

[24] M. Rudd and C. C. Tisdell. On the solvability of two-point, second-order boundary value problems. *Appl. Math. Lett.*, 20(2007):pp. 824–828.

[25] J. Shen, T. Tang, and L. L. Wang. *Spectral methods: algorithms, analysis and applications*, volume 41. Springer Science & Business Media, 2011.

[26] W. Sun and Y. Yuan. *Optimization theory and methods. Nonlinear programming.* Optimization theory and methods. Nonlinear programming, 2006.

[27] E. Tadmor. A review of numerical methods for nonlinear partial differential equations. *B. Am. Math. Soc.*, 49(2012):pp. 507–554.

[28] Y. W. Wang, W. Hao, and G. Lin. Two-level spectral methods for nonlinear elliptic equations with multiple solutions. *SIAM J. Sci. Comput.*, 40(2018):pp. B1180–B1205.

[29] Z. Q. Wang. On a superlinear elliptic equation. *Ann. Inst. Henri. Poincaé*, 8(1991):43–57.

[30] J. Xia, A. Pef, and B. Sgpc. Nonlinear bifurcation analysis of stiffener profiles via deflation techniques. *Thin. Wall. Struct.*, 149(2020):pp. 1–11.

[31] Z. Q. Xie, C. M. Chen, and Y. Xu. An improved search-extension method for computing multiple solutions of semilinear PDEs. *IMA J. Numer. Anal.*, 25(2005):pp. 549–576.
[32] Z. Q. Xie, C. M. Chen, and Y. Xu. An improved search-extention method for solving semilinear PDEs. *Acta. Math. Sci.*, 26(2006):pp. 757–766.
[33] Z. Q. Xie, W. F. Yi, and J. X. Zhou. An augmented singular transform and its partial newton method for finding new solutions. *J. Comput. Appl. Math.*, 286(2015):pp. 145–157.
[34] H. Xu, Z. L. Lin, S. J. Liao, J. Z. Wu, and J. Majdalani. Homotopy based solutions of the navier–stokes equations for a porous channel with orthogonally moving walls. *Phys. Fluids*, 22(2010):pp. 053601–053618.
[35] X. D. Yao and J. X. Zhou. A minimax method for finding multiple critical points in banach spaces and its application to quasi-linear elliptic PDEs. *SIAM J. Sci. Comput.*, 26(2005):pp. 1796–1809.
[36] X. D. Yao and J. X. Zhou. Numerical methods for computing nonlinear eigenpairs: Part I. Iso-Homogeneous cases. *SIAM J. Sci. Comput.*, 29(2007):pp. 1355–1374.
[37] X. D. Yao and J. X. Zhou. Numerical methods for computing nonlinear eigenpairs: Part II. Non-Iso-Homogeneous cases. *SIAM J. Sci. Comput*, 30(2008):pp. 937–956.
[38] H. C. Zhang, R. C. Andrew, and K. Scheinberg. A derivative-free algorithm for least-squares minimization. *SIAM J. Optim.*, 20(2010):pp. 3555–3576.
[39] X. P. Zhang, J. T Zhang, and B. Yu. Eigenfunction expansion method for multiple solutions of semilinear elliptic equations with polynomial nonlinearity. *SIAM J. Numer. Anal.*, 51(2013):pp. 2680–2699.
[40] J. X. Zhou. Instability analysis of saddle points by a local minimax method. *Math. Comp.*, 74(2004):pp. 1391–1411.