Open Phys. 2018; 16:554–562 DE GRUYTER

Research Article Open Access

Hillary Muzara, Stanford Shateyi*, and Gerald Tendayi Marewo

On the bivariate spectral quasi-linearization method for solving the two-dimensional Bratu problem

https://doi.org/10.1515/phys-2018-0072
Received April 9, 2018; accepted May 30, 2018

Abstract: In this paper, a bivariate spectral quasi-linearization method is used to solve the highly non-linear two dimensional Bratu problem. The two dimensional Bratu problem is also solved using the Chebyshev spectral collocation method which uses Kronecker tensor products. The bivariate spectral quasi-linearization method and Chebyshev spectral collocation method solutions converge to the lower branch solution. The results obtained using the bivariate spectral quasi-linearization method were compared with results from finite differences method, the weighted residual method and the homotopy analysis method in literature. Tables and graphs generated to present the results obtained show a close agreement with known results from literature.

Keywords: Bratu problem, quasi-linearization, Chebyshev-Gauss-Lobatto points, bivariate interpolation, spectral collocation

PACS: 02.30.Hq, 02.60.Cb, 02..60.Lj, 02.70.Hm

1 Introduction

The most of real life phenomena are modeled by partial differential equations (PDEs). In science, engineering, biological sciences and fluid mechanics, most of these phenomena are described by PDEs, which are usually non-linear [1, 2]. Due to the complexity of the domains in which they are defined, it is usually very difficult or even impossible, except for a few special cases, to find exact solutions to the defining PDEs. This motivated researchers to develop numerical and analytical methods to approximate solutions to these non-linear PDEs. Some of the well known analytical methods that have been used to solve non-linear PDEs include homotopy pertubation method [3], Adomain decomposition method [4] , power series expansions [5], the artificial small parameter method [6] and the δ-pertubation expansion method [7]. Although these methods help us to understand many non-linear phenomena, they have their own disadvantage in that the convergence of the solution series is not guaranteed due to their dependence on small or large physical parameters. Some examples of numerical methods that have been used to solve non-linear phenomena include finite element methods [8], finite difference methods [9], quasi-linearization technique [10], iterative finite difference method [11], the B-spline method [12]. It is worth mentioning that the fractional-wavelet approach is an important technique which can be used to solve nonlinear PDEs. For further reading on the fractional-wavelet approach and fractional calculus, we refer interested readers to [13, 14].

An example of a highly non-linear differential equation is the so-called Bratu problem, first set up by Bratu and named after him [15]. The simplest form of the Bratu problem in one dimension is:

$$\frac{d^2u}{dx^2} + \lambda e^{u(x)} = 0, \; x \in [0, 1] \tag{1}$$

subject to boundary conditions $u(0) = u(1) = 0$. The exact solution of Eq. (1) is given by

$$u(x) = -2 \ln \left[\frac{\cosh((x - \frac{1}{2})\frac{\omega}{2})}{\cosh(\frac{\omega}{4})} \right] \tag{2}$$

where ω is the solution of the equation $\omega = 2\sqrt{2\lambda} \cosh\left(\frac{\omega}{4}\right)$ [16]. The Bratu problem has no, unique, or two solutions if $\lambda > \lambda_c$, $\lambda = \lambda_c$, or $\lambda < \lambda_c$ respectively, where the critical value $\lambda_c = 3.51382$ [17]. The

Hillary Muzara: Department of Mathematics and Applied Mathematics, University of Venda, Thohoyandou 0950, South Africa, E-mail: stanford.shateyi@univen.ac.za
***Corresponding Author: Stanford Shateyi:** Department of Mathematics and Applied Mathematics, University of Venda, Thohoyandou 0950, South Africa, E-mail: muzarahillary@gmail.com
Gerald Tendayi Marewo: Department of Mathematics and Applied Mathematics, University of Limpopo, P Bag X1106, Sovenga 0727, South Africa, E-mail: Gtmarewo@gmail.com

generalization of the Bratu problem is the Liouville-Bratu-Gelfand problem [16] which in the n-dimensional coordinate system takes the form

$$\Delta u(\mathbf{x}) + \lambda e^{u(\mathbf{x})} = 0, \ \mathbf{x} \in \Omega \tag{3}$$

where the square domain Ω is bounded in $\mathbb{R}^n$ together with homogeneous Dirichlet boundary conditions $u(\mathbf{x}) = 0$, $\mathbf{x} \in \partial\Omega$, where $\partial\Omega$ is the boundary of Ω.

In this study, we consider the two dimensional Bratu problem, which has the form

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \lambda e^u = 0, \ (x, y) \in [0, 1] \times [0, 1] \tag{4}$$

subject to boundary conditions

$$u(0, y) = u(1, y) = u(x, 0) = u(x, 1) = 0 \tag{5}$$

where λ is a positive number. Similar to the 1D case, depending on the value of the parameter λ, Eq. (4) has no, one, or two solutions if $\lambda > \lambda_c$, $\lambda = \lambda_c$ or $\lambda < \lambda$, respectively, where the critical value $\lambda_c = 6.808124423$ [18]. The two dimensional Bratu problem has no exact solution as is the case with the one dimensional Bratu problem. However, Odejide and Aregbesola [19] presented a near exact solution that satisfies Eq. (4) at just one collocation point $(x_c, y_c) = (\frac{1}{2}, \frac{1}{2})$ as well as the boundary conditions. The analytical solution is given as

$$u(x, y) = 2 \ln \left[\frac{\cosh(\frac{\omega}{4}) \cosh[(x - \frac{1}{2})(y - \frac{1}{2})\omega]}{\cosh((x - \frac{1}{2})\frac{\omega}{2}) \cosh(y - \frac{1}{2})\frac{\omega}{2}} \right] \tag{6}$$

where ω is a solution of the equation

$$\omega^2 = \lambda \cosh^2 \left(\frac{\omega}{4} \right). \tag{7}$$

The two dimensional Bratu problem has attracted the attention of many researchers because of its wide range of physical, chemical and engineering applications. It is reported in [16] that the Bratu problem is used to model the thermal reaction process in a combustible non-deformable material. The Bratu problem also appears in the Chandrasekhar model of the expansion of the universe, chemical reactor theory and nanotechnology [20]. Recently, the Bratu problem has found applications in engineering such as electro-spinning process for the fabrication of nano-fibers [21]. Apart from the physical applications, the Bratu problem is also used as a benchmark for newly developed numerical and analytical methods [22].

Some of the numerical methods that have been used to treat the Bratu problem in two dimensions include: a wavelet homotopy analysis method (wHAM) by Zhaochen and Shijun [22], the finite difference (FD) and the weighted residual method (WRM) by Odejide and Aregbesola [19] and the Chebyshev pseudospectral method using Gegenbauer polynomials [23] by Boyd.

The main objective of this work is to solve (for the first time), the so-called Bratu problem in two dimensions using the bivariate spectral quasi-linearization method (BSQLM). The BSQLM introduced by Motsa et al [24] is a modification of the spectral quasi-linearization method [25] to solve non-linear PDEs in two dimensions. Some of the problems successfully solved using the BSQLM include the modified Kdv equation, Burger equation, the Cahn-Hillard equation and the Fitzhugh-Nagumo equations [24]. It is in this work that researchers concluded that the method is accurate, reliable and applicable to nonlinear evolution equations. The obtained results also showed that the method achieves high accuracy with relatively fewer spatial grid points and converges fast to the exact solution. Also, Motsa and Ansari [26] solved non-dimensionalized PDEs describing a time dependent boundary layer flow and heat transfer of an incompressible Oldroyd-B nanofluid past an impulsively stretching sheet using the BSQLM. The results obtained converged rapidly.

2 Method of solution

2.1 Bivariate spectral quasi-linearization method

In this section, we briefly describe the Bivariate spectral quasi-linearization method (BSQLM). The BSQLM uses the quasi-linearization method, Chebyshev collocation method and bivariate Lagrange interpolation.

2.1.1 *Quasi-linearization method*

The quasi-linearization method (QLM) which is based on Newton-Raphson method was introduced by Bellman and Kalaba [27]. It is a technique for simplifying non-linear PDEs using the linear terms of the Taylor series expansion about an initial approximation. There are some techniques, which can be used to transform nonlinear PDEs to linear equations. As an example, the Hopf-Cole transformation discussed in [28, 29], is used to turn a strongly nonlinear Burgers equation into a linear equation. In this work, let us consider a general second order non-linear differential equation

$$F(\mathbf{u}) = R(\mathbf{x}),\ \ \mathbf{x} = (x, y) \in [a, b] \times [c, d] \quad (8)$$

subject to given boundary conditions. The unknown function $\mathbf{u} = (u, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial x \partial y}, \frac{\partial u}{\partial y}, \frac{\partial^2 u}{\partial y^2})$ and F is a non-linear operator. Expanding F using linear Taylor series expansion about $\mathbf{v}$ we get

$$F(\mathbf{v}) + \nabla F(\mathbf{v}) \cdot (\mathbf{u} - \mathbf{v}) \approx R(\mathbf{x}) \quad (9)$$

Assuming that $\mathbf{v}$ is an approximate solution sufficiently close to $\mathbf{u}$ and adopting the notation $\mathbf{u}_r$ and $\mathbf{u}_{r+1}$ for $\mathbf{v}$ and $\mathbf{u}$ respectively, we have

$$\nabla F(\mathbf{u}_r) \cdot \mathbf{u}_{r+1} = \nabla F(\mathbf{u}_r) \cdot \mathbf{u}_r - F(\mathbf{u}_r) + R(\mathbf{x}) \quad (10)$$

where $r = 0, 1, 2, \cdots$. Solving Eq. (10) generates a sequence $\{\mathbf{u}_r\}$ and hence u_r such that $u_r \to u$ as $r \to \infty$.

2.1.2 *Bivariate Lagrange interpolation and Chebyshev differentiation*

We seek the solution u of Eq. (10) of the form

$$u(x, y) \approx \sum_{j=0}^{N} \sum_{i=0}^{M} u(x_j, y_i) L_{ji}(x, y) \quad (11)$$

where the functions $L_{ji}(x, y) = L_j(x) L_i(y)$ are bivariate Lagrange polynomials defined as

$$L_j(x) = \prod_{\substack{k=0 \\ k \neq j}}^{N} \frac{x - x_k}{x_j - x_k},\ \ L_i(y) = \prod_{\substack{k=0 \\ k \neq i}}^{M} \frac{y - y_k}{y_i - x_k} \quad (12)$$

The functions $L_j(x)$ and $L_i(y)$ both obey the Kronecker delta equation, that is,

$$L_{ji}(x_n, y_m) = \begin{cases} 1, & \text{if } j = n,\ i = m \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

Before applying the spectral method, it is convenient to transform the physical domain $[a, b] \times [c, d]$ in the x-y axis to the computational domain $[-1, 1] \times [-1, 1]$ in the ξ-η axis using linear transformations $x(\xi) = \frac{a+b}{2} + \frac{b-a}{2}\xi$ and $y(\eta) = \frac{c+d}{2} + \frac{d-c}{2}\eta$. Approximating the partial derivatives of u at Chebyshev-Gauss-Lobatto collocation points

$$\{x_j\}_{j=0}^{N} = \cos\left(\frac{\pi j}{N}\right),\ \ \{y_i\}_{i=0}^{M} = \cos\left(\frac{\pi i}{M}\right), \quad (14)$$

we have

$$\left.\frac{\partial u}{\partial x}\right|_{\substack{x=x_j \\ y=y_i}} \approx \sum_{p=0}^{N} \sum_{q=0}^{M} u(x_p, y_q) \frac{dL_p(x_j)}{dx} L_q(y_i) \quad (15)$$

$$= \sum_{p=0}^{N} D_{jp} u(x_p, y_i) = \mathbf{D}\mathbf{U}_i. \quad (16)$$

$$\left.\frac{\partial u}{\partial y}\right|_{\substack{x=x_j \\ y=y_i}} \approx \sum_{q=0}^{M} d_{iq} u(x_j, y_q) = \sum_{q=0}^{M} d_{iq} \mathbf{U}_q, \quad (17)$$

and

$$\left.\frac{\partial^2 u}{\partial x \partial y}\right|_{\substack{x=x_j \\ y=y_i}} \approx \sum_{p=0}^{N} \sum_{q=0}^{M} u(x_p, y_q) \frac{dL_p(x_j)}{dx} \frac{dL_q(y_i)}{dy} = \sum_{q=0}^{M} d_{iq} \left(\mathbf{D}\mathbf{U}_q\right), \quad (18)$$

where $\widehat{\mathbf{D}} = \left(\frac{b-a}{2}\right)\mathbf{D}$ and $\widehat{\mathbf{d}}_{iq} = \left(\frac{d-c}{2}\right) d_{iq}$ are the standard Chebyshev differentiation matrices [30] of orders $(N+1) \times (N+1)$ and $(M+1) \times (M+1)$, respectively, and $\mathbf{U}_i = \left(u(x_0, y_i), u(x_1, y_i), \cdots, u(x_N, y_i)\right)^T$, for $i = 0, 1, 2, \cdots, N$. The superscript T denotes matrix transposition. Higher order derivatives of u are defined as follows:

$$\left.\frac{\partial^n u}{\partial x^n}\right|_{\substack{x=x_j \\ y=y_i}} \approx \mathbf{D}^n \mathbf{U}_i,\ \ \left.\frac{\partial^n u}{\partial y^n}\right|_{\substack{x=x_j \\ y=y_i}} \approx \sum_{q=0}^{M} d_{iq}^n \mathbf{U}_q,\ \ n = 2, 3, \cdots.$$

and

$$\frac{\partial^{n+m} u}{\partial x^n \partial y^m} = \sum_{q=0}^{M} d_{iq}^m \left(\mathbf{D}^n \mathbf{U}_q\right),\ \ n, m = 1, 2, \cdots$$

Expanding Eq. (10) we get

$$\alpha_{5,r} \frac{\partial^2 u_{r+1}}{\partial x^2} + \alpha_{4,r} \frac{\partial^2 u_{r+1}}{\partial y^2} + \alpha_{3,r} \frac{\partial^2 u_{r+1}}{\partial x \partial y} + \alpha_{2,r} \frac{\partial u_{r+1}}{\partial x} + \alpha_{1,r} \frac{\partial u_{r+1}}{\partial y} + \alpha_{0,r} u_{r+1} = K_r, \quad (19)$$

where

$$\alpha_{5,r} = \frac{\partial F(u_r)}{\partial u_{xx}},\ \ \alpha_{4,r} = \frac{\partial F(u_r)}{\partial u_{yy}},\ \alpha_{3,r} = \frac{\partial F(u_r)}{\partial u_{xy}},$$
$$\alpha_{2,r} = \frac{\partial F(u_r)}{\partial u_x},\ \ \alpha_{1,r} = \frac{\partial F(u_r)}{\partial u_y},\ \ \alpha_{0,r} = \frac{\partial F(u_r)}{\partial u}$$

and

$$K_r = \alpha_{5,r} \frac{\partial^2 u_r}{\partial x^2} + \alpha_{4,r} \frac{\partial^2 u_r}{\partial y^2} + \alpha_{3,r} \frac{\partial^2 u_r}{\partial x \partial y} + \alpha_{2,r} \frac{\partial u_r}{\partial x} + \alpha_{1,r} \frac{\partial u_r}{\partial y} + \alpha_{0,r} u_r - F(\mathbf{u}_r) + R(\mathbf{x}).$$

Approximating the u and its derivatives in Eq. (19) at collocation points (x_j, y_i) we get

$$\alpha_{5,r}(\mathbf{x}, y_i) \widehat{\mathbf{D}}^2 \mathbf{U}_{r+1,i} + \alpha_{4,r}(\mathbf{x}, y_i) \sum_{q=0}^{M} \widehat{d}_{jq}^2 \mathbf{U}_{r+1,q}$$

$$
\begin{aligned}
&+ \quad \alpha_{3,r}(\mathbf{x}, y_i) \sum_{q=0}^{M} \hat{d}_{iq}^2 \left(\hat{\mathbf{D}}^2 \mathbf{U}_{r+1,q} \right) + \alpha_{2,r}(\mathbf{x}, y_i) \hat{\mathbf{D}} \mathbf{U}_{r+1,i} \\
&+ \quad \alpha_{1,r}(\mathbf{x}, y_i) \sum_{q=0}^{M} \hat{d}_{iq} \left(\hat{\mathbf{D}} \mathbf{U}_{r+1,q} \right) + \alpha_{0,i}(\mathbf{x}, y_i) \mathbf{U}_{r+1,i} \\
&= \quad \mathbf{K}_{r,i} \qquad (20)
\end{aligned}
$$

where
$\alpha_{a,r}(\mathbf{x}, y_i)$
=

$$
\begin{bmatrix} \alpha_{a,r}(x_0, y_i) & & & \\ & \alpha_{a,r}(x_1, y_i) & & \\ & & \ddots & \\ & & & \alpha_{a,r}(x_N, y_i) \end{bmatrix}
$$

and

$$
\begin{aligned}
K_{r,i} \quad = \quad & \alpha_{5,r}(\mathbf{x}, y_i) \hat{\mathbf{D}}^2 \mathbf{U}_{r,i} + \alpha_{4,r}(\mathbf{x}, y_i) \sum_{q=0}^{M} \hat{d}_{jq}^2 \mathbf{U}_{r,q} \\
+ \quad & \alpha_{3,r}(\mathbf{x}, y_i) \sum_{q=0}^{M} \hat{d}_{iq}^2 \left(\hat{\mathbf{D}}^2 \mathbf{U}_{r,q} \right) + \alpha_{2,r}(\mathbf{x}, y_i) \hat{\mathbf{D}} \mathbf{U}_{r,i} \\
+ \quad & \alpha_{1,r}(\mathbf{x}, y_i) \sum_{q=0}^{M} \hat{d}_{iq} \left(\hat{\mathbf{D}} \mathbf{U}_{r,q} \right) + \alpha_{0,i}(\mathbf{x}, y_i) \mathbf{U}_{r,i} \\
- \quad & F(u_r(\mathbf{x}, y_i)) + R(\mathbf{x}, y_i).
\end{aligned}
$$

In compact form, Eq. (20) can be written as:

$$
\begin{bmatrix} A_{0,0} & A_{0,1} & \cdots & A_{0,M} \\ A_{1,0} & A_{1,1} & \cdots & A_{1,M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N-1,0} & A_{N-1,1} & \cdots & A_{N-1,M} \\ A_{N,0} & A_{N,1} & \cdots & A_{N,M} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{r+1,0} \\ \mathbf{U}_{r+1,1} \\ \vdots \\ \mathbf{U}_{r+1,N-1} \\ \mathbf{U}_{r+1,N} \end{bmatrix} = \begin{bmatrix} \mathbf{K}_0 \\ \mathbf{K}_1 \\ \vdots \\ \mathbf{K}_{M-1} \\ \mathbf{K}_M \end{bmatrix}
$$

where

$$
\begin{aligned}
A_{ii} \quad &= \quad \alpha_{5,r}(\mathbf{x}, y_i) \hat{\mathbf{D}}^2 + \alpha_{4,r}(\mathbf{x}, y_i) \hat{d}_{ii} \mathbf{I} \\
&+ \quad \alpha_{3,r}(\mathbf{x}, y_i) \hat{d}_{ii}^2 \hat{\mathbf{D}}^2 + \alpha_{2,r} \mathbf{x}, y_i \hat{\mathbf{D}} \\
&+ \quad \alpha_{1,r}(\mathbf{x}, y_i) \hat{d}_{ii} \hat{\mathbf{D}} + \alpha_{0,r}(\mathbf{x}, y_i) \\
A_{ij} \quad &= \quad \hat{d}_{ij} \mathbf{I}, \quad \text{when} \quad i \neq j
\end{aligned}
$$

2.2 Application to the current problem

The equivalent form of Eq. (4) in the ξ-η axis is

$$
\frac{\partial^2 u}{\partial \xi^2} + \frac{\partial^2 u}{\partial \eta^2} + \frac{\lambda}{4} e^{u(\xi,\eta)} = 0, \ (\xi, \eta) \in [-1, 1] \times [-1, 1] \qquad (21)
$$

subject to boundary conditions

$$
u(-1, \eta) = u(1, \eta) = u(\xi, -1) = u(\xi, 1) = 0. \qquad (22)
$$

Applying formula Eq. (10) on Eq. (21) we get its linear counterpart

$$
\frac{\partial^2 u_{r+1}}{\partial \xi^2} + \frac{\partial^2 u_{r+1}}{\partial \eta^2} + \frac{\lambda}{4} e^{u_r} u_{r+1} = \frac{\lambda}{4} e^{u_r} [u_r - 1], \qquad (23)
$$

$r = 0, 1, 2, \cdots$ together with boundary conditions

$$
u_{r+1}(-1, \eta) = u_{r+1}(1, \eta) = u_{r+1}(\xi, -1) = u_{r+1}(\xi, 1) = 0.
$$

Applying the spectral collocation method we get

$$
\begin{aligned}
&\sum_{p=0}^{N} \hat{D}_{jp}^2 U_{r+1}(\xi_p, \eta_i) + \sum_{q=0}^{M} \hat{d}_{iq}^2 U_{r+1}(\xi_j, \eta_q) \\
&\quad + \frac{\lambda}{4} e^{U_r(\xi_j, \eta_i)} U_{r+1}(\xi_j, \eta_i) = K_i, \qquad (24)
\end{aligned}
$$

subject to boundary conditions

$$
\begin{aligned}
&U_{r+1}(\xi_N, \eta_j) = U_{r+1}(\xi_0, \eta_j) \\
&= U_{r+1}(\xi_i, \eta_M) = U_{r+1}(\xi_i, \eta_0) = 0. \qquad (25)
\end{aligned}
$$

Substituting Eqn. (25) into Eq. (24) we get

$$
\begin{aligned}
&\sum_{p=1}^{N-1} \hat{D}_{jp}^2 U_{r+1}(\xi_p, \eta_i) + \sum_{q=1}^{M-1} \hat{d}_{iq}^2 U_{r+1}(\xi_j, \eta_q) \\
&\quad + \frac{\lambda}{4} e^{U_r(\xi_j, \eta_i)} U_{r+1}(\xi_j, \eta_i) = K_i, \qquad (26)
\end{aligned}
$$

$$
\Rightarrow \mathbf{A}\mathbf{U}_{r+1} = \mathbf{K}_i \qquad (27)
$$

Eq. (27) can be written in a matrix form as

$$
\begin{bmatrix} A_{0,0} & A_{0,1} & \cdots & A_{0,M} \\ A_{1,0} & A_{1,2} & \cdots & A_{1,M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N-1,0} & A_{N-1,1} & \cdots & A_{N-1,M} \\ A_{N,0} & A_{N,1} & \cdots & A_{N,M} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{r+1,0} \\ \mathbf{U}_{r+1,1} \\ \vdots \\ \mathbf{U}_{r+1,N-1} \\ \mathbf{U}_{r+1,N} \end{bmatrix} = \begin{bmatrix} \mathbf{K}_0 \\ \mathbf{K}_1 \\ \vdots \\ \mathbf{K}_{M-1} \\ \mathbf{K}_M \end{bmatrix} \qquad (28)
$$

where

$$
\begin{aligned}
A_{ii} \quad &= \quad \hat{\mathbf{D}}^2 + \hat{d}_{ii} I + \text{diag}\left[\frac{\lambda}{4} e^{\mathbf{U}_{r,i}} \right], \\
A_{ij} \quad &= \quad \hat{d}_{ij}^2 \mathbf{I}, \\
\mathbf{K}_i \quad &= \quad \frac{\lambda}{4} e^{\mathbf{U}_{r,i}} \circ \left[\mathbf{U}_{r,i} - \mathbf{i} \right].
\end{aligned}
$$

The column vector $\mathbf{i} = \left[11 \cdots 1 \right]^T$ and the Hamadard product $A \circ B$ is the element-wise multiplication of matrices A and B of the same order. Boundary conditions are applied to the system (28) as follows:

$$
\begin{bmatrix} \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{A}}_{1,1} & \cdots & \hat{\mathbf{A}}_{1,M-1} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \hat{\mathbf{A}}_{N-1,1} & \cdots & \hat{\mathbf{A}}_{N-1,M-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{U}}_{r+1,0} \\ \hat{\mathbf{U}}_{r+1,1} \\ \vdots \\ \hat{\mathbf{U}}_{r+1,N-1} \\ \hat{\mathbf{U}}_{r+1,N} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{K}}_1 \\ \vdots \\ \hat{\mathbf{K}}_{M-1} \\ \mathbf{0} \end{bmatrix} \qquad (29)
$$

where $\mathbf{I}$ and $\mathbf{0}$ are identity and zero matrices, respectively, of order $N \times M$ and

$$\widehat{\mathbf{A}}_{ij} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ \hline & & A_{ij} & & \\ \hline 0 & 0 & \cdots & 0 & 1 \end{pmatrix} \text{ and } \widehat{\mathbf{K}}_i = \begin{pmatrix} 0 \\ \hline \mathbf{K}_i \\ \hline 0 \end{pmatrix} \quad (30)$$

2.3 Chebyshev spectral collocation method

As mentioned before, the exact solution of the two dimensional Bratu problem is unknown. As a basis of comparison, we will use the Chebyshev spectral collocation method, which uses Kronecker multiplication, abbreviated CSCM-K in this work, to solve the transformed Eq. (21) subject to boundary conditions Eq. (22). We intend to use the CSCM-K because of its known high order accuracy [31]. In multi-dimensional problems, the spectral collocation methods make use of Kronecker products to discretize the differential operators. It is worth noting that though the BSQLM uses spectral collocation, its main difference with the CSCM-K is the manner in which the two methods treat non-linearity.

Definition 2.1. *If A and B are of dimensions $p \times q$ and $r \times s$, respectively, then the Kronecker product $A \otimes B$ is the matrix of dimension $pr \times qs$ with $p \times q$ block form where the i, j block is $a_{i,j}B$, that is*

$$A \otimes B = \begin{bmatrix} a_{1,1}B & a_{1,2}B & \cdots & a_{1,q-1}B & a_{1,q}B \\ a_{2,1}B & a_{2,2}B & \cdots & a_{2,q-1}B & a_{1,q}B \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{p-1,1}B & a_{p-1,2}B & \cdots & a_{p-1,q-1}B & a_{p-1,q}B \\ a_{p,1}B & a_{p,2}B & \cdots & a_{p,q-1}B & a_{p,q}B \end{bmatrix} \quad (31)$$

The matrix $A \otimes B$ is not dense, at the same time not as sparse as matrices from traditional methods, like the finite element or finite difference methods. Since Eq. (22) is highly non-linear, its solution is computed iteratively using the linear system

$$M\mathbf{u}_{s+1} = \mathbf{f}_s, \quad s = 0, 1, 2, \cdots \quad (32)$$

where $M = I \otimes \widehat{D}^2 + \widehat{D}^2 \otimes I$, $\mathbf{f}_s = -\exp(\mathbf{u}_s)$ and $\mathbf{u}_s$ is the current iteration. $I \otimes \widehat{D}^2$ and $\widehat{D}^2 \otimes I$ denote second order spectral differentiation in the x and y directions respectively. $A \otimes B$ can be easily computed using the `MATLAB` command `kron(A,B)`. The homogeneous boundary conditions Eq. (22) are implemented by deleting the first and the last rows and columns of the spectral differentiation matrix.

3 Results and discussion

Solving Eq. (7) for different values of the parameter λ (shown in Table 1), we obtain two values of ω labeled ω_1 and ω_2. Substituting the values of ω_1 and ω_2 into Eq. (6) and take the maximum of $u(x, y)$(denoted u_{max1} and u_{max2}) we get results, which are shown in Table 1. We consider the values of λ as done in [19].

A graphical representation of the results in Table 1 is done in Figure 1

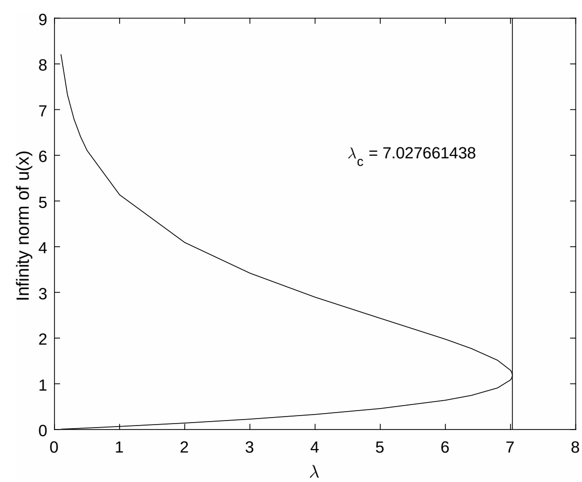


Figure 1: The maximum values of $u(x, y)$ versus λ.

Figure 1 shows that the analytical solution has two branches (lower and upper). It is also clear that the Bratu problem has two solutions for $0 < \lambda < \lambda_c$, one solution for $\lambda = \lambda_c$ and no solution for $\lambda > \lambda_c$. The mesh plots of the lower branch ($\lambda = 5$, $\omega_1 = 2.811554938$) and upper branch ($\lambda = 5$, $\omega_2 = 7.548098106$) are shown in Figure 2.

In Table 2, we compare the values of u_{max} of the three methods, WRM, CSCM-K and BSQLM. We consider the values of the parameter λ as done in [19]. We also consider the solution in the domain subdivided into equal 4×4 sub-regions. Since the exact solution of the two dimensional Bratu problem is not known in literature, we use the results from WRM and CSCM-K as our basis of comparison to the BSQLM results. The results in Table 2 show that the BSQLM solution agree to 5 decimal places with the results from CSCM-K, which is known to be highly accurate method.

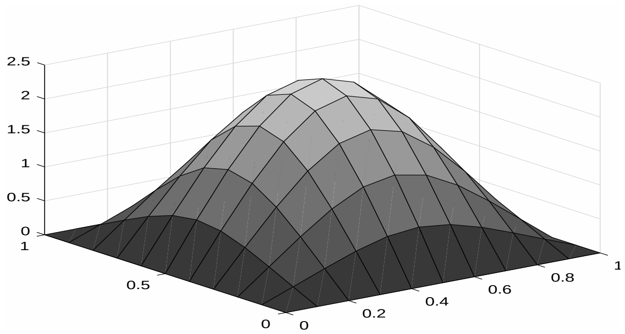
Rounding off the results in Table 2 to the nearest 5 decimal places for $\lambda(= 1, 2, 3, 4, 5, 6)$ we get the results in Table 3. These results compare very well with the results obtained by [22] after solving the two dimensional Bratu problem using the optimal homotopy analysis (oHAM)

Table 1: The maximum values of the solution $u(x, y)$ by the WRM, CSCM-K and BSQLM for different values of λ

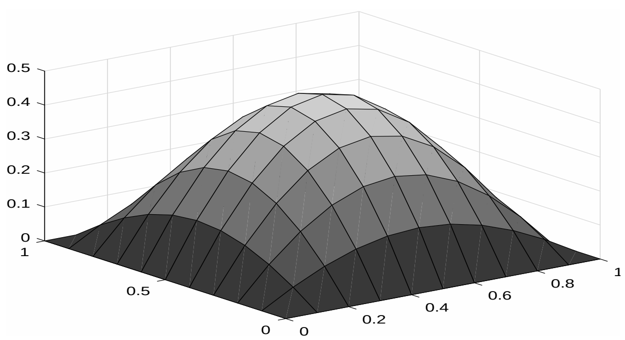
λ	ω		u_{max}	
	ω_1	ω_2	u_{max1}	u_{max2}
0.1	0.317222727	19.19637291	0.006282809236	8.212027794
0.5	0.718546417	14.98479987	0.03209723648	6.107219872
1.0	1.033569462	13.03823930	0.06603661666	5.135773050
2.0	1.517164599	10.93870277	0.1405392141	4.091467246
3.0	1.939726525	9.581699793	0.2264817040	3.421097726
4.0	2.357551054	8.507199571	0.3289524214	2.895531266
5.0	2.811554938	7.548098106	0.4580374660	2.433153338
6.0	3.373507764	6.576569259	0.6401466966	1.975266972
6.4	3.674358094	6.131465409	0.7464589086	1.770569562
6.8	4.108395792	5.56288431	0.9091426554	1.515096612
7.0	4.551853663	5.054342699	1.085158948	1.515096612
7.02	4.667812741	4.932041041	1.132617977	1.294585480
7.027661438	4.798690688	4.798714561	1.186832218	1.242742595

Table 2: The maximum values of the solution $u(x, y)$ by the WRM, CSCM-K and BSQLM for different values of λ

λ	WRM	CSCM-K	BSQLM
0.1	0.00739148232	0.007427053468645	0.007427132966698
0.5	0.03781451306	0.037984412494968	0.037985298386246
1.0	0.0779460505	0.078104419480805	0.078103863139426
2.0	0.1665928797	0.166902012852600	0.166900505362755
3.0	0.2698543078	0.270370407490254	0.270374143745502
4.0	0.3945269608	0.395540911835512	0.395540918829025
5.0	0.554387275	0.556970351040996	0.556971761314595
6.0	0.7871142546	0.796804803376730	0.796803797175867
6.4	0.9292292746	1.003321398467646	1.003326671983020
6.8	1.169825663	0.968377381095159	0.968377828328672
6.939571828	1.453226755	1.099599554422270	1.099599287729232



(a) Upper branch solution $\lambda = 5$ and $\omega_2 = 7.548098106$



(b) Lower branch solution $\lambda = 5$ and $\omega_1 = 2.811554938$

Figure 2: Mesh plots of the upper and lower solutions for selected values of λ and ω

and wavelet homotopy analysis (wHAM) and the iterative wHAM.

We solve the two dimensional Bratu problem using BSQLM for $\lambda = 1$ and 10×10 sub-regions of the problem domain $[0, 1] \times [0, 1]$. A graphical representation of the solution is a mesh plot in Figure 3 which agrees with the solution plot in [22]. With the fuel ignition model as one of the physical applications of the Bratu problem, taking $u(x, y)$ to represent temperature, the results in Figure 3 show that there is a continuous decrease of temperature from the midpoint towards the boundary. Table 4 compares u_{max} from BSQLM, CSCM-K and FDM for different values of the λ_c as done in [19]. For all the cases, we consider equal sub-regions and a tolerance level of 10^{-6}. The results show that the BSQLM produces results in good agreement with CSCM-K. Due to lack of an exact solution, it is impossible to directly compute the accuracy of the BSQLM in solving the two dimensional Bratu problem. We now compare the two methods, BSQLM and CSCM-K in terms of speed of convergence and computational efficiency.

Table 5 shows a comparison between BSQLM and CSCM-K in terms of runtime in seconds and number of iterations for the solutions to converge within a tolerance level of 10^{-15} for different values of equal sub-regions

Table 3: The maximum values of $u(x, y)$ for various values of λ

λ	WRM [19]	oHAM [22]	wHAM [22]	iterative wHAM [22]	CSCM-K	BSQLM
1.0	0.07795	0.07810	0.07810	0.07810	0.07810	0.07810
2.0	0.16659	0.16689	0.16690	0.16690	0.16690	0.16690
3.0	0.26985	0.27036	0.27037	0.27037	0.27037	0.27037
4.0	0.39453	0.39552	0.39554	0.39554	0.39554	0.39554
5.0	0.55439	0.55696	0.55697	0.55697	0.55697	0.55697
6.0	0.78711	0.79711	0.79680	0.79710	0.79680	0.79710

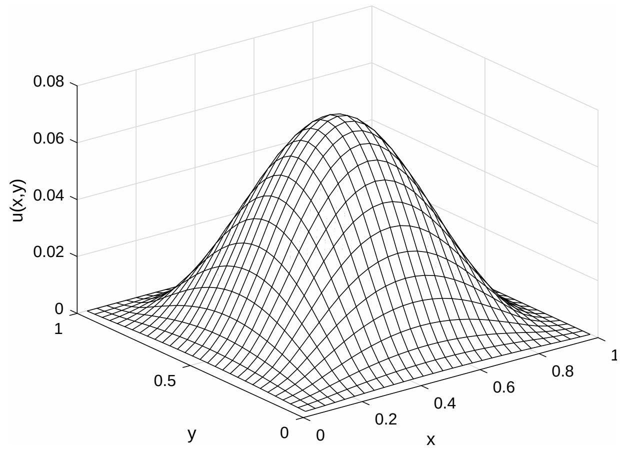


Figure 3: Solution of the Bratu problem using BSQLM for $\lambda = 1$.

Table 4: The maximum values of $u(x, y)$ for various values of λ_c and $N \times N$.

$N \times N$	λ_c	WRM [19]	CSCM-K	BSQLM
5×5	6.739545	1.2295280	0.9396735	0.9396720
10×10	6.792610	1.3835357	1.2979979	1.2979965
20×20	6.80497	1.3911583	1.3488587	1.3488579
40×40	6.81565	1.3881487	1.2560298	1.2560290
100×100	7.12222	1.3565527	1.1507510	1.1507509

Table 5: Runtime and the number of iterations

	BSQLM		CSCM-K	
$N \times N$	Runtime(s)	No. of iter	Runtime(s)	No. of iter
5×5	0.035430	6	0.020919	12
10×10	0.039587	5	0.035721	12
20×20	0.132422	4	0.532621	12
40×40	0.497956	4	4.559524	12
60×60	7.462640	3	21.314275	12
80×80	31.353469	3	144.779811	12
100×100	126.152106	3	577.760840	12

$N \times N$. In all the cases, we consider the constant parameter $\lambda = 1$. It is worth noting that after the 10×10 sub-region, the BSQLM solution takes less time to converge than the CSCM-K solution. This shows that the BSQLM is more computationally efficient than CSCM-K. Moreover, as the subdivisions are made finer, the BSQLM needs less iterations to converge than the CSCM-K, which needs a constant 12 iterations to converge. This proves that the BSQLM is faster than CSCM-K.

4 Conclusion

We solved the two dimensional Bratu problem using the bivariate spectral collocation method (BSQLM) and the Chebyshev spectral collocation method that uses Kronecker multiplication (CSCM-K). Both methods produced solutions which converge to the lower solution. We compared the results with those from finite difference method (FDM) and weighted residual method (WRM) in literature. We observed that results from BSQLM and CSCM-K are in close agreement with the results in Table 2. Rounding-off the results in Table 2 to 5 decimal places for selected values of λ, we get results in Table 3, which are in excellent agreement with the results obtained using the oHAM, wHAM and iterative HAM. This proves that the BSQLM is capable of producing reliable results as it compares well with the wHAM, which is known to have a high computational efficiency. From the results in Table 5, we conclude that BSQLM is faster and more computationally efficient than CSCM-K.

Conflict of Interests
The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] Slunyaev A.V., Shrira V.I., On the highest non-breaking wave in a group: fully nonlinear water wave breathers versus weakly nonlinear theory, J. Fluid Mech., 2013, 735, 203-48.
[2] Sibanda P., Motsa S , Makukula Z . A spectral-homotopy analysis method for heat transfer flow of a third grade fluid between parallel plates, Int. J. Numer. Methods Heat. Fluid Flow, 2012, 22(1), 4-23.
[3] He J.H., Homotopy perturbation method for bifurcation of nonlinear problems, Int. J. Nonlinear Sci. Numer. Simul., 2005, 6, 27-33.
[4] Adomian G., A review of the decomposition method and some recent results for nonlinear equation, Math. Comput. Model., 1992, 13(7), 17-43.
[5] Mohan C., Al-Bayaty A.R., Power-series solutions of the Lane-Emden equation, Astrophysics and Space Science, 1980, 73(1), 227-239.
[6] Lyapunov A.M., The general problem of the stability of motion, Int. J. Control., 1992, 55(3), 531-534.
[7] Ji-Huan H.E., A note on delta-perturbation expansion method, Appl. Math. Mech., 2002, 23(6), 634-638.
[8] Argyris J., Haase M., An engineer's guide to soliton phenomena: application of the finite element method, Comp. Meth. Appl. Mech. Engineering, 1987, 61, 1, 71-122.
[9] Vliegenthart A.C, On inite-difference methods for the Korteweg-de Vries equation, J. Eng. Math., 1971, 5, 137-155.
[10] Motsa S.S., Sibanda P., Some modifications of the quasilinearization method with higher-order con vergence for solving nonlinear BVPs, Numer. Algorithms, 2013, 63(3), 399-417.
[11] Temimi H., Ben-Romdhane M., An iterative finite difference method for solving Bratu's problem, J. Comp. Appl. Math., 2016,

292, 76-82.

[12] Caglar H. et al., B-spline method for solving Bratu's problem, J. Comp. Appl. Math., 2010, 87(8), 1885-1891.

[13] Guariglia E., Silvestrov S., Fractional-Wavelet Analysis of Positive definite Distributions and Wavelets on D'(C), In: Engineering Mathematics II, 2016, 337-353, Springer, Cham.

[14] Guariglia E., Fractional Derivative of the Riemann Zeta Function, Fractional Dynamics, Chapter: 21, De Gruyter, Cattani C., Srivastava H., Yang X.J., 2015, 357-368.

[15] G.Bratu G, Sur les equation integrals non-lineaires, Bull. Math. Soc. France, 1914, 42, 113-142.

[16] Jacobsen J., Schmitt K., The Liouville– Bratu–Gelfand problem for radial operators, J. Diff. Equations, 2002, 184.1, 283-298.

[17] Mohsen A., Sedeek L.F., Mohamed S.A., Newsmoother to enhance multigrid-based methods for Bratu problem, Appl. Math Comp., 2008, 204.1, 325-339.

[18] Chang S.L., Chien C.S., A Multigrid-Lanczos algorithm for the numerical solutions of nonlinear eigenvalue problems, Int. J. Bifurc. Chaos, 2003, 13, 1217-1228.

[19] Odejide S.A., Aregbesola Y.A.S., A note on two dimensional Bratu problem, Kragujevac J. Math., 2006, 29(29), 49-56.

[20] Mohsen A., A simple solution of the Bratu problem, Comp. & Math. Appl., 2014, 67(1), 26-33.

[21] Wan Y.Q., Guo Q. Pan N., Thermo-electro-hydrodynamic model for electrospinning process, Int. J. Nonlinear Sci. Numer. Simulation, 2004, 5, 5-8.

[22] Yang Z., Shijun L., A HAM-based wavelet approach for nonlinear partial differential equations: Two dimensional Bratu problem as an application, Comm. Nonlinear Sci. Numer. Simulation, 2017, 53, 249-262.

[23] Boyd J.P., An analytical and numerical study of the two- dimensional Bratu equation, J. Sci. Comput., 1986, 1(2), 183-206.

[24] Motsa S.S., Magagula V. M., Sibanda P., A bivariate Chebyshev spectral collocation quasilinearization method for nonlinear evolution parabolic equations, Sci. World J., 2014, 2014.

[25] Shateyi S., Mabood F., Marewo G.T., On a new numerical approach on micropolar fluid, heat and mass transfer over an unsteady stretching sheet through porous media in the presence of a heat source/sink and chemical reaction, Numer. Simul., 2016, 1, 291-312.

[26] Motsa S.S., Ansari M.S., Unsteady boundary layer flow and heat transfer of Oldroyd- B nanofluid towards a stretching sheet with variable thermal conductivity, Thermal Sci., 2015, 19(1), 239-248.

[27] Bellman R.E., Kalaba R.E, Quasilinearization and nonlinear boundary-value problems, 1965, Elsevier, New York (1965).

[28] Smaoui N., Belgacem F., Connections between the convective diffusion equation and the forced Burgers equation, J. Appl. Math. Stoch. Anal., 2002, 15(1), 53-69.

[29] Belgacem F., Smaoui N., Interaction of Parabolic Conective Diffusion Equations and Navier-Stokes Equations Connected with Population Dispersal, Comm. Appl. Nonlin. Anal., 2001, 8(3), 47-67.

[30] Trefethen L.N., Spectral methods in MATLAB, Society for Ind. Appl. Math., 2000.

[31] Liu F., Ye X., Wang X., Efficient Chebyshev spectral method for solving linear elliptic PDEs using quasi-inverse technique, Numer. Math. Theor. Meth. Appl., 2011, 4, 197-215.