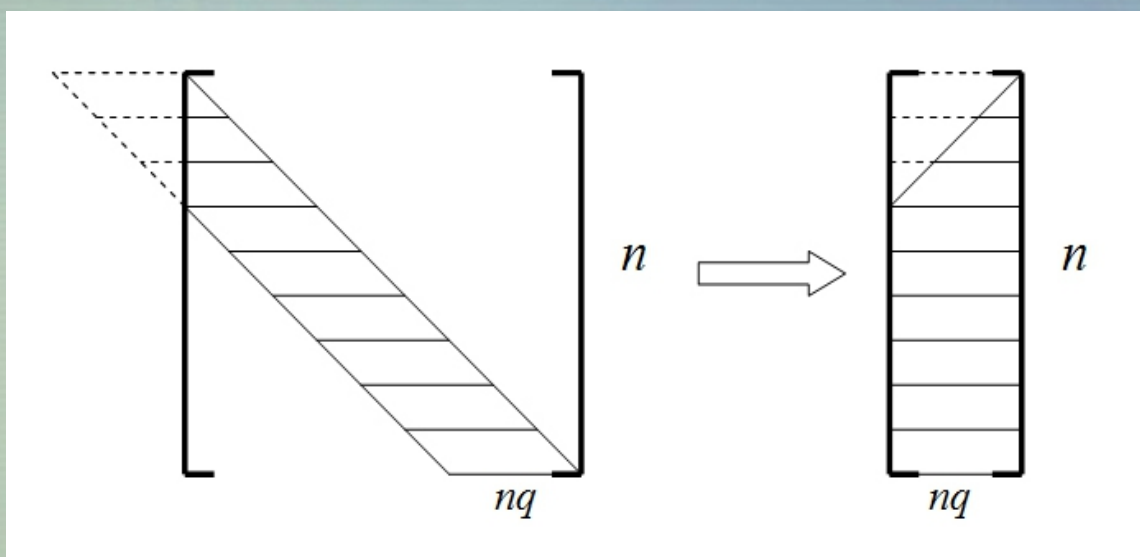




American Journal of Computational Mathematics



ISSN: 2161-1203



Journal Editorial Board

ISSN: 2161-1203(Print) 2161-1211(Online)

<http://www.scirp.org/journal/ajcm>

Editor-in-Chief

Prof. Jose C. Valverde

Department of Mathematics, University of Castilla-La Mancha, Spain

Editorial Board

Prof. Huseyin Bor

Erciyes University, Turkey

Dr. Ibrahim Buyukyazici

Ankara University, Turkey

Prof. Singa Wang Chiu

Chaoyang University of Technology, Chinese Taipei

Prof. Yuan-Shyi Peter Chiu

Chaoyang University of Technology, Chinese Taipei

Dr. Dragana S. Cvetković

University of Nis, Serbia

Dr. Mohamed K. El-Daou

College of Technological Studies, PAAET, Kuwait

Prof. Yuri Anatolyevich Farkov

Russian State Geological Prospecting University, Russia

Prof. Ahmed Fouad Ghaleb

Cairo University, Egypt

Prof. Stanislaw Migorski

Jagiellonian University, Poland

Prof. Gangrong Qu

Beijing Jiaotong University, China

Dr. Fathalla A. Rihan

Helwan University, Egypt

Dr. Victor Rukavishnikov

Russian Academy of Sciences, Russia

Prof. Nickolas S. Sapidis

University of Western Macedonia, Greece

Dr. Nicolae Adrian Secolean

Lucian Blaga University, Romania

Prof. Yi-Fa Tang

Chinese Academy of Sciences, China

Prof. Michael Gr. Voskoglou

Graduate Technological Educational Institute of Patras, Greece

Prof. Jinyun Yuan

Federal University of Parana, Brazil

Prof. Kewen Zhao

Qiongzhou University, Hainan, China

Dr. Su Zheng

Genentech, Inc., USA

Table of Contents

Volume 4 Number 2

March 2014

Numerical Solutions of Second Order Initial Value Problems of Bratu-Type via Optimal Homotopy Asymptotic Method

M. A. Darwish, B. S. Kashkari.....47

An Actual Survey of Dimensionality Reduction

A. Sarveniazi.....55

High Accurate Fourth-order Finite Difference Solutions of the Three Dimensional Poisson's Equation in Cylindrical Coordinate

A. Shiferaw, R. C. Mittal.....73

L-Stable Block Hybrid Second Derivative Algorithm for Parabolic Partial Differential Equations

F. F. Ngwane, S. N. Jator.....87

Derivative of a Determinant with Respect to an Eigenvalue in the Modified Cholesky Decomposition of a Symmetric Matrix, with Applications to Nonlinear Analysis

M. Kashiwagi.....93

Bifurcations of Travelling Wave Solutions for the B(m,n) Equation

M. Z. Wei, Y. J. Gan, S. Q. Tang.....104

American Journal of Computational Mathematics (AJCM)

Journal Information

SUBSCRIPTIONS

The *American Journal of Computational Mathematics* (Online at Scientific Research Publishing, www.SciRP.org) is published quarterly by Scientific Research Publishing, Inc., USA.

Subscription rates:

Print: \$79 per issue.

To subscribe, please contact Journals Subscriptions Department, E-mail: sub@scirp.org

SERVICES

Advertisements

Advertisement Sales Department, E-mail: service@scirp.org

Reprints (minimum quantity 100 copies)

Reprints Co-ordinator, Scientific Research Publishing, Inc., USA.

E-mail: sub@scirp.org

COPYRIGHT

COPYRIGHT AND REUSE RIGHTS FOR THE FRONT MATTER OF THE JOURNAL:

Copyright © 2014 by Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>

COPYRIGHT FOR INDIVIDUAL PAPERS OF THE JOURNAL:

Copyright © 2014 by author(s) and Scientific Research Publishing Inc.

REUSE RIGHTS FOR INDIVIDUAL PAPERS:

Note: At SCIRP authors can choose between CC BY and CC BY-NC. Please consult each paper for its reuse rights.

DISCLAIMER OF LIABILITY

Statements and opinions expressed in the articles and communications are those of the individual contributors and not the statements and opinion of Scientific Research Publishing, Inc. We assume no responsibility or liability for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained herein. We expressly disclaim any implied warranties of merchantability or fitness for a particular purpose. If expert assistance is required, the services of a competent professional person should be sought.

PRODUCTION INFORMATION

For manuscripts that have been accepted for publication, please contact:

E-mail: ajcm@scirp.org

Journal Editorial Board

ISSN: 2161-1203(Print) 2161-1211(Online)

<http://www.scirp.org/journal/ajcm>

Editor-in-Chief

Prof. Jose C. Valverde

Department of Mathematics, University of Castilla-La Mancha, Spain

Editorial Board

Prof. Huseyin Bor

Erciyes University, Turkey

Dr. Ibrahim Buyukyazici

Ankara University, Turkey

Prof. Singa Wang Chiu

Chaoyang University of Technology, Chinese Taipei

Prof. Yuan-Shyi Peter Chiu

Chaoyang University of Technology, Chinese Taipei

Dr. Dragana S. Cvetković

University of Nis, Serbia

Dr. Mohamed K. El-Daou

College of Technological Studies, PAAET, Kuwait

Prof. Yuri Anatolyevich Farkov

Russian State Geological Prospecting University, Russia

Prof. Ahmed Fouad Ghaleb

Cairo University, Egypt

Prof. Stanislaw Migorski

Jagiellonian University, Poland

Prof. Gangrong Qu

Beijing Jiaotong University, China

Dr. Fathalla A. Rihan

Helwan University, Egypt

Dr. Victor Rukavishnikov

Russian Academy of Sciences, Russia

Prof. Nickolas S. Sapidis

University of Western Macedonia, Greece

Dr. Nicolae Adrian Secolean

Lucian Blaga University, Romania

Prof. Yi-Fa Tang

Chinese Academy of Sciences, China

Prof. Michael Gr. Voskoglou

Graduate Technological Educational Institute of Patras, Greece

Prof. Jinyun Yuan

Federal University of Parana, Brazil

Prof. Kewen Zhao

Qiongzhou University, Hainan, China

Dr. Su Zheng

Genentech, Inc., USA

Table of Contents

Volume 4 Number 2

March 2014

Numerical Solutions of Second Order Initial Value Problems of Bratu-Type via Optimal Homotopy Asymptotic Method

M. A. Darwish, B. S. Kashkari.....47

An Actual Survey of Dimensionality Reduction

A. Sarveniazi.....55

High Accurate Fourth-order Finite Difference Solutions of the Three Dimensional Poisson's Equation in Cylindrical Coordinate

A. Shiferaw, R. C. Mittal.....73

L-Stable Block Hybrid Second Derivative Algorithm for Parabolic Partial Differential Equations

F. F. Ngwane, S. N. Jator.....87

Derivative of a Determinant with Respect to an Eigenvalue in the Modified Cholesky Decomposition of a Symmetric Matrix, with Applications to Nonlinear Analysis

M. Kashiwagi.....93

Bifurcations of Travelling Wave Solutions for the B(m,n) Equation

M. Z. Wei, Y. J. Gan, S. Q. Tang.....104

American Journal of Computational Mathematics (AJCM)

Journal Information

SUBSCRIPTIONS

The *American Journal of Computational Mathematics* (Online at Scientific Research Publishing, www.SciRP.org) is published quarterly by Scientific Research Publishing, Inc., USA.

Subscription rates:

Print: \$79 per issue.

To subscribe, please contact Journals Subscriptions Department, E-mail: sub@scirp.org

SERVICES

Advertisements

Advertisement Sales Department, E-mail: service@scirp.org

Reprints (minimum quantity 100 copies)

Reprints Co-ordinator, Scientific Research Publishing, Inc., USA.

E-mail: sub@scirp.org

COPYRIGHT

COPYRIGHT AND REUSE RIGHTS FOR THE FRONT MATTER OF THE JOURNAL:

Copyright © 2014 by Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>

COPYRIGHT FOR INDIVIDUAL PAPERS OF THE JOURNAL:

Copyright © 2014 by author(s) and Scientific Research Publishing Inc.

REUSE RIGHTS FOR INDIVIDUAL PAPERS:

Note: At SCIRP authors can choose between CC BY and CC BY-NC. Please consult each paper for its reuse rights.

DISCLAIMER OF LIABILITY

Statements and opinions expressed in the articles and communications are those of the individual contributors and not the statements and opinion of Scientific Research Publishing, Inc. We assume no responsibility or liability for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained herein. We expressly disclaim any implied warranties of merchantability or fitness for a particular purpose. If expert assistance is required, the services of a competent professional person should be sought.

PRODUCTION INFORMATION

For manuscripts that have been accepted for publication, please contact:

E-mail: ajcm@scirp.org

Numerical Solutions of Second Order Initial Value Problems of Bratu-Type via Optimal Homotopy Asymptotic Method

Mohamed Abdalla Darwish, Bothayna S. Kashkari

Department of Mathematics, Sciences Faculty for Girls, King Abdulaziz University, Jeddah, KSA
Email: dr.madarwish@gmail.com, bkashkari@kau.edu.sa

Received 26 December 2013; revised 26 January 2014; accepted 1 February 2014

Copyright © 2014 by authors and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

We present the optimal homotopy asymptotic method (OHAM) to find the numerical solution of the second order initial value problems of Bratu-type. We solve some examples to illustrate the validity and efficiency of the method.

Keywords

Initial-Value Problem; Bratu; Numerical Solution; Optimal Homotopy Asymptotic Method

1. Introduction

Herişanu *et al.* [1] proposed a new technique called the optimal homotopy asymptotic method (OHAM). The main advantage of OHAM is that it is reliable and straight forward. Also, the OHAM does not need to worry about h curves as homotopy asymptotic method (HAM). Moreover, the OHAM provides controls the convergence of the series solution and its solution agrees with the exact one at large domains, for more information see [2]-[6].

On the other hand, the standard Bratu problem is used in a large variety of applications, such as the fuel ignition model of the theory of thermal combustion, the thermal reaction process model, the Chandrasekhar model of the expansion of the universe, radiative heat transfer, nanotechnology and theory of chemical reaction, for more information see [7] [8] and references therein.

The Bratu initial value problems have been studied extensively because of its mathematical and physical properties. In [9], Batiha studied a numerical solution of Bratu-type equations by the variational iteration method; Feng *et al.* [10] considered Bratu's problems by means of modified homotopy perturbation method; Rashidinia *et al.* [11] applied Sinc-Galerkin method for numerical solution of the Bratu's problems; Syam and Hamdan [12]

How to cite this paper: Darwish, M.A. and Kashkari, B.S. (2014) Numerical Solutions of Second Order Initial Value Problems of Bratu-Type via Optimal Homotopy Asymptotic Method. *American Journal of Computational Mathematics*, 4, 47-54.
<http://dx.doi.org/10.4236/ajcm.2014.42005>

used variational iteration method for numerical solutions of the Bratu-type problems; Wazwaz [13] applied Adomian decomposition method to study the Bratu-type equations.

The main goal of this paper is to extend OHAM method to solve the initial value problems of second order differential equations of Bratu-type. The OHAM is very useful to get an approximate solution of the initial value problems of second order differential equations of Bratu-type. Our numerical examples of OHAM are compared with exact ones.

2. Analysis of OHAM

In this section we start by describing the basic formulation of OHAM, see for example [1] [3]-[5]. Consider the boundary value problem

$$\begin{cases} L(u(x)) + g(x) + N(u(x)) = 0, \\ B\left(u, \frac{du}{dx}\right) = 0, \end{cases} \quad (2.1)$$

where $g = g(x)$ is a given function and $u = u(x)$ is an unknown function. Here, L , N and B represent a linear operator, a nonlinear operator and a boundary operator, respectively.

By means of OHAM one constructs a homotopy $h(x, p) : \mathbb{R} \times [0, 1] \rightarrow \mathbb{R}$, which satisfies the following family of equations

$$\begin{cases} (1-p)[L(h(x, p)) + g(x)] = H(p)[L(h(x, p)) + g(x)N(h(x, p))] \\ B\left(h(x, p), \frac{\partial h(x, p)}{\partial x}\right) = 0, \end{cases} \quad (2.2)$$

where $p \in [0, 1]$ is an embedding parameter, $H(p)$ is a non-zero auxiliary function for $p \neq 0$ and $H(0) = 0$. It is easy to see that when $p = 0$ and $p = 1$ we have $h(x, 0) = u_0(x)$ and $h(x, 1) = u(x)$, respectively, where $u_0(x)$ is obtained from (2.2) for $p = 0$

$$\begin{cases} L(u_0(x)) + g(x) = 0, \\ B(u_0(x), 0) = 0. \end{cases} \quad (2.3)$$

Therefore, the unknown function $h(x, p)$ goes from $u_0(x)$ to $u(x)$ as p changes from 0 to 1.

In the sequel, we choose auxiliary function $H(p)$ in the form

$$H(p) = c_1 p + c_2 p^2 + c_3 p^3 + \dots, \quad (2.4)$$

where c_i , $i = 1, 2, 3, \dots$, are constants to be determined.

In order to obtain an approximate solution, we expand $h(x, p, c_i)$, $i = 1, 2, 3, \dots$, in the form of Taylor's series about p as

$$h(x, p, c_i) = u_0(x) + \sum_{j=1}^{\infty} u_j(x, c_i) p^j, \quad i = 1, 2, 3, \dots \quad (2.5)$$

Now, substituting by Equation (2.5) into Equation (2.2) and equating the coefficients of like powers of p in the resulting equation, we obtain the governing problem of $u_0(x)$, given by Equation (2.3). In addition, the governing problems of $u_1(x)$ and $u_2(x)$ are given in the forms

$$\begin{cases} L(u_1(x)) + g(x) = c_1 N_0(u_0(x)), \\ B\left(u_1(x), \frac{d}{dx} u_1(x)\right) = 0 \end{cases} \quad (2.6)$$

and

$$\begin{cases} L(u_2(x)) = L(u_1(x)) + c_2 N_0(u_0(x)) + c_1 [L(u_1(x)) + N_1(u_0(x), u_1(x))], \\ B\left(u_2(x), \frac{d}{dx} u_2(x)\right) = 0, \end{cases} \quad (2.7)$$

respectively. Also, the general governing problems of $u_j(x)$ are given by

$$\begin{cases} L(u_j(x)) = L(u_{j-1}(x)) + c_j N_0(u_0(x)) \\ \quad + \sum_{i=1}^{j-1} c_i [L(u_{j-i}(x)) + N_{j-i}(u_0(x), u_1(x), \dots, u_{j-1}(x))] \\ B\left(u_j(x), \frac{d}{dx}u_j(x)\right) = 0, j = 2, 3, 4, \dots, \end{cases} \quad (2.8)$$

where $N_m(u_0(x), u_1(x), \dots, u_{j-1}(x))$ is the coefficient of p^m in the expansion of $N(h(x, p))$ about the embedding parameter p :

$$N(h(x, p, c_i)) = N_0(u_0(x)) + \sum_{m=1}^{\infty} N_m(u_0, u_1, \dots, u_m) p^m, \quad (2.9)$$

where $h(x, p, c_i)$, $i = 1, 2, 3, \dots$, is given by Equation (2.5).

Observe that the convergence of the series (2.5) depends upon the auxiliary constants c_i , $i = 1, 2, 3, \dots$. If the series (2.5) converges when $p = 1$, one has

$$h(x, 1, c_i) = u_0(x) + \sum_{j=1}^{\infty} N_j(x, u_0, u_1, \dots, u_j). \quad (2.10)$$

The m -th order approximations are given by

$$\tilde{u}(x, c_1, c_2, \dots, c_m) = u_0(x) + \sum_{i=1}^m u_i(x, c_1, c_2, \dots, c_i). \quad (2.11)$$

By substituting Equation (2.11) into Equation (2.1), we get the following expression for residual

$$R(x, c_1, c_2, \dots, c_m) = L(\tilde{u}(x, c_1, c_2, \dots, c_m)) + g(x) + N(\tilde{u}(x, c_1, c_2, \dots, c_m)). \quad (2.12)$$

If $R = 0$, then \tilde{u} will be the exact solution and this, in general, does not happen especially in nonlinear problems. In order to find the optimal values of c_i , $i = 1, 2, 3, \dots$, we apply the method of least squares as under

$$J(c_1, c_2, \dots, c_m) = \int_a^b R^2(x, c_1, c_2, \dots, c_m) dx, \quad (2.13)$$

where a and b are numbers properly chosen in the domain of the problem. Next, minimizing J with

$$\frac{\partial J}{\partial c_1} = \frac{\partial J}{\partial c_2} = \dots = \frac{\partial J}{\partial c_m} = 0.$$

After knowing those constants, the approximate solution of order m is well determined.

3. Numerical Examples

Example 1 Consider the second order initial value problem of Bratu type

$$\begin{cases} \frac{d^2}{dx^2} u(x) = 2e^{u(x)}, \\ u(0) = 0, u'(0) = 0. \end{cases} \quad (3.1)$$

The initial value problem (3.1) has $u(x) = -2 \ln \cos x$ as the exact solution.

Next, we apply the OHAM method to the initial value problem (3.1). We have

$g(x) = 0$, $L(h(x, p)) = h_{xx}(x, p)$ and $N(h(x, p)) = -2e^{h(x, p)}$. Therefore, according to the OHAM method, we have

Problem of zero order:

$$\begin{cases} \frac{d^2}{dx^2} u_0(x) = 0, \\ u_0(0) = 0, u'_0(0) = 0, \end{cases} \quad (3.2)$$

which has a solution $u_0(x) = 0$.

Problem of first order:

$$\begin{cases} \frac{d^2}{dx^2} u_1(x, c_1) = -2c_1, \\ u_1(0) = 0, u'_1(0) = 0. \end{cases} \quad (3.3)$$

Problem (3.3) has a solution

$$u_1(x, c_1) = -c_1 x^2. \quad (3.4)$$

The problem of second order

$$\begin{cases} \frac{d^2}{dx^2} u_2(x, c_1, c_2) = -2(c_1 + c_2) - 2c_1^2 + 2c_1^2 x^2, \\ u_2(0) = 0, u'_2(0) = 0. \end{cases} \quad (3.5)$$

The solution of Problem (3.5) is given by

$$u_2(x, c_1, c_2) = -\frac{1}{6}(6c_1 x^2 + 6c_2 x^2 + 6c_1^2 x^2 - c_1^2 x^4). \quad (3.6)$$

Third order problem is

$$\begin{cases} \frac{d^2}{dx^2} u_3(x, c_1, c_2, c_3) = -2(c_1 + c_2 + c_3) - 4(1 - x^2)(c_1 c_2 + c_1^2) - \frac{2}{3}(3 - 6x^2 + 2x^4)c_1^3, \\ u_3(0) = 0, u'_3(0) = 0 \end{cases} \quad (3.7)$$

and its solution is given in the form

$$u_3(x, c_1, c_2, c_3) = -x^2(c_1 + c_2 + c_3) - \frac{1}{3}(6x^2 - x^4)(c_1 c_2 + c_1^2) - \frac{1}{45}(45x^2 - 15x^4 + 2x^6)c_1^3. \quad (3.8)$$

Finally, fourth order problem is

$$\begin{cases} \frac{d^2}{dx^2} u_4(x, c_1, c_2, c_3, c_4) = -2(c_1 + c_2 + c_3 + c_4) - 2(1 - x^2)(4c_1 c_2 + 2c_1 c_3 + 3c_1^2 + c_2^2) \\ \quad - (6 - 12x^2 + 4x^4)(c_1^2 c_2 + c_1^3) - \frac{1}{45}(90 - 270x^2 + 180x^4 - 19x^6)c_1^4, \\ u_4(0) = 0, u'_4(0) = 0, \end{cases} \quad (3.9)$$

which has a solution in the form

$$\begin{aligned} u_4(x, c_1, c_2, c_3, c_4) = & -x^2(c_1 + c_2 + c_3 + c_4) - \frac{1}{6}(6x^2 - x^4)(4c_1 c_2 + 2c_1 c_3 + 3c_1^2 + c_2^2) \\ & - \frac{1}{15}(45x^3 - 15x^4 + 2x^6)(c_1^2 c_2 + c_1^3) - \frac{1}{2520}(2520x^2 - 1260x^4 + 336x^6 - 19x^8)c_1^4. \end{aligned} \quad (3.10)$$

Now, by using equations (3.4), (3.6), (3.8) and (3.10), the fourth order approximate solution, using OHAM with $p = 1$, is given by

$$\tilde{u}(x, c_1, c_2, c_3, c_4) = u_0(x) + u_1(x, c_1) + u_2(x, c_1, c_2) + u_3(x, c_1, c_2, c_3) + u_4(x, c_1, c_2, c_3, c_4). \quad (3.11)$$

Next, we follow the procedure presented in Section 2, we obtain the following values of c_i 's:

$$c_1 = -0.9556156427, \quad c_2 = 0.0942570476, \quad c_3 = 0.0374885311 \quad \text{and} \quad c_4 = -0.0116590295 \quad (\text{Table 1}).$$

Table 1. Absolute error between the exact solution and approximation solution.

x	Exact sol.	OHAM sol.	Error
0.0	0.00000000	0.00000000	0.00000000
0.1	0.01001671	0.01001607	$6.41021065 \times 10^{-7}$
0.2	0.04026955	0.04025980	$9.74693876 \times 10^{-6}$
0.3	0.09138331	0.09133801	$4.52998213 \times 10^{-5}$
0.4	0.16445804	0.16433092	$1.27118347 \times 10^{-4}$
0.5	0.26116848	0.26089981	$2.68671650 \times 10^{-4}$
0.6	0.38393034	0.38344668	$4.83656903 \times 10^{-4}$
0.7	0.53617152	0.53533472	$8.36799541 \times 10^{-4}$
0.8	0.72278149	0.72118096	$1.60053795 \times 10^{-3}$
0.9	0.95088489	0.94723518	$3.64970628 \times 10^{-3}$
1.0	1.23125294	1.22186142	$9.39151960 \times 10^{-3}$

Example 2 In this example, let us consider the Bratu initial value problem

$$\frac{d^2}{dx^2} u(x) = \pi^2 e^{u(x)}, \quad u(0) = 0, u'(0) = \pi, \quad (3.12)$$

which has

$$u(x) = -\ln \left(1 + \cos \frac{(2x+1)\pi}{2} \right) \text{ exact solution.}$$

Now, we apply the OHAM method presented in previous section. In this example, we have

$$g(x) = 0, \quad L(h(x, p)) = h_{xx}(x, p) \quad \text{and} \quad N(h(x, p)) = -\pi^2 e^{h(x, p)}.$$
 Now,

Problem of zero order:

$$\begin{cases} \frac{d^2}{dx^2} u_0(x) = 0, \\ u_0(0) = 0, u'_0(0) = \pi. \end{cases} \quad (3.13)$$

Problem (3.13) has a solution $u_0(x) = \pi x$.

Problem of first order:

$$\begin{cases} \frac{d^2}{dx^2} u_1(x) = -\pi^2 c_1 \left(1 + \pi x + \frac{\pi^2}{2} x^2 \right), \\ u_1(0) = 0, u'_1(0) = 0. \end{cases} \quad (3.14)$$

The solution of Problem (3.14) is given by

$$u_1(x) = -\frac{1}{24} \pi^2 x^2 c_1 (12 + 4\pi x + \pi^2 x^2). \quad (3.15)$$

The problem of second order

$$\begin{cases} \frac{d^2}{dx^2} u_2(x, c_1, c_2) \\ = -\frac{\pi^2}{2} (2 + 2\pi x + \pi^2 x^2) (c_1 + c_2) \\ - \frac{\pi^2}{24} (24 + 24\pi x - 16\pi^3 x^3 - 5\pi^4 x^4 - \pi^5 x^5) c_1^2, \\ u_2(0) = 0, u'_2(0) = 0, \end{cases} \quad (3.16)$$

and its solution is given by

$$\begin{aligned} u_2(x) = & -\frac{\pi^2 x^2}{24} (12 + 4\pi x + \pi^2 x^2) (c_1 + c_2) \\ & - \frac{\pi^2 x^2}{5040} (2520 + 840\pi x - 168\pi^3 x^3 - 35\pi^4 x^4 - 5\pi^5 x^5) c_1^2. \end{aligned} \quad (3.17)$$

Third order problem is

$$\begin{cases} \frac{d^2}{dx^2} u_3(x, c_1, c_2, c_3) \\ = -\frac{\pi^2}{2} (2 + 2\pi x + \pi^2 x^2) (c_1 + c_2 + c_3) \\ - \frac{\pi^2}{12} (24 + 24\pi x - 16\pi^3 x^3 - 5\pi^4 x^4 - \pi^5 x^5) (c_1 c_2 + c_1^2) \\ - \pi^2 \left(1 + \pi x - \frac{1}{2} \pi^2 x^2 - \frac{4}{3} \pi^3 x^3 - \frac{1}{4} \pi^4 x^4 + \frac{3}{40} \pi^5 x^5 + \frac{3}{40} \pi^6 x^6 + \frac{5}{336} \pi^7 x^7 + \frac{5}{2688} \pi^8 x^8 \right) c_1^3, \\ u_3(0) = 0, u'_3(0) = 0. \end{cases} \quad (3.18)$$

The solution of Problem (3.18) is given by

$$\begin{aligned} u_3(x, c_1, c_2, c_3) \\ = & -\frac{\pi^2 x^2}{24} (12 + 4\pi x + \pi^2 x^2) (c_1 + c_2 + c_3) \\ & - \frac{\pi^2}{2520} (2520x^2 + 840\pi x^3 - 168\pi^3 x^5 - 35\pi^4 x^6 - 5\pi^5 x^7) (c_1 c_2 + c_1^2) \\ & - \pi^2 \left(\frac{x^2}{2} + \frac{\pi x^3}{6} - \frac{\pi^2 x^4}{24} - \frac{\pi^3 x^5}{15} - \frac{\pi^4 x^6}{120} + \frac{\pi^5 x^7}{560} + \frac{3\pi^6 x^8}{2240} + \frac{5\pi^7 x^9}{24192} + \frac{\pi^8 x^{10}}{48384} \right) c_1^3. \end{aligned} \quad (3.19)$$

In the end, the fourth order problem is given by

$$\begin{cases} \frac{d^2}{dx^2} u_4(x, c_1, c_2, c_3, c_4) \\ = -\frac{\pi^2}{2} (2 + 2\pi x + \pi^2 x^2) (c_1 + c_2 + c_3 + c_4) \\ - \frac{\pi^2}{24} (24 + 24\pi x - 16\pi^3 x^3 - 5\pi^4 x^4 - \pi^5 x^5) (4c_1 c_2 + 2c_1 c_3 + 3c_1^2 + c_2^2) \\ - \frac{\pi^2}{8960} (26880 + 26880\pi x - 13440\pi^2 x^2 - 35840\pi^3 x^3 - 6720\pi^4 x^4 \\ + 2016\pi^5 x^5 + 2016\pi^6 x^6 + 400\pi^7 x^7 + 50\pi^8 x^8) (c_1^2 c_2 + c_1^3) \\ - \frac{\pi^2}{161280} (161280 + 161280\pi x - 161280\pi^2 x^2 - 322560\pi^3 x^3 - 20160\pi^4 x^4 + 56448\pi^5 x^5 \\ + 32032\pi^6 x^6 + 1888\pi^7 x^7 - 1660\pi^8 x^8 - 740\pi^9 x^9 - 110\pi^{10} x^{10} - 10\pi^{11} x^{11}) c_1^4, \\ u_4(0) = 0, u'_4(0) = 0. \end{cases} \quad (3.20)$$

which has a solution in the form

Table 2. Absolute error between the exact solution and approximation solution.

x	Exact sol.	OHAM sol.	Error
-0.3	-0.59278360	-0.59050589	$2.27771434 \times 10^{-3}$
-0.2	-0.46234012	-0.46195092	$3.89206013 \times 10^{-4}$
-0.1	-0.26927647	-0.26926585	$1.06182822 \times 10^{-5}$
0.0	0.00000000	0.00000000	0.00000000
0.1	0.36964005	0.36959323	$4.68162342 \times 10^{-5}$
0.2	0.88621083	0.88427218	$1.93865546 \times 10^{-3}$
0.3	1.65557083	1.63022895	$2.53418803 \times 10^{-2}$

$$\begin{aligned}
& u_4(x, c_1, c_2, c_3, c_4) \\
&= -\frac{\pi^2 x^2}{24} (12 + 4\pi x + \pi^2 x^2) (c_1 + c_2 + c_3 + c_4) - \frac{\pi^2 x^2}{2520} (2520 + 840\pi x - 168\pi^3 x^3 - 35\pi^4 x^4 - 5\pi^5 x^5) \\
&\quad \cdot (2c_1 c_2 + c_1 c_3) - \frac{\pi^2 x^2}{5040} (2520 + 840\pi x - 168\pi^3 x^3 - 35\pi^4 x^4 - 5\pi^5 x^5) (3c_1^2 + c_2^2) \\
&\quad - \frac{\pi^2 x^2}{8960} (13440 + 4480\pi x - 1120\pi^2 x^2 - 1792\pi^3 x^3 - 224\pi^4 x^4 + 48\pi^5 x^5 + 36\pi^6 x^6 + \frac{50}{9}\pi^7 x^7 + \frac{5}{9}\pi^8 x^8) \\
&\quad \cdot (c_1^2 c_2 + c_1^3) - \frac{\pi^2 x^2}{161280} (80640 + 26880\pi x - 13440\pi^2 x^2 - 16128\pi^3 x^3 - 672\pi^4 x^4 + 1344\pi^5 x^5 + 572\pi^6 x^6 \\
&\quad + \frac{236}{9}\pi^7 x^7 - \frac{166}{9}\pi^8 x^8 - \frac{74}{11}\pi^9 x^9 - \frac{5}{6}\pi^{10} x^{10} - \frac{5}{78}\pi^{11} x^{11}) c_1^4. \quad (3.21)
\end{aligned}$$

Now, by using Equations (3.4), (3.6), (3.8) and (3.10), the fourth order approximate solution, using OHAM with $p = 1$, is given by

$$\tilde{u}(x, c_1, c_2, c_3, c_4) = u_0(x) + u_1(x, c_1) + u_2(x, c_1, c_2) + u_3(x, c_1, c_2, c_3) + u_4(x, c_1, c_2, c_3, c_4). \quad (3.22)$$

Next, we follow the procedure presented in Section 0.2, we obtain the following values of c_i 's:
 $c_1 = -1.0391835661$, $c_2 = -0.0042471858$, $c_3 = 0.0000013808$ and $c_4 = 0.0001595594$ (Table 2).

4. Final Remarks

Throughout this paper, an technique for obtaining a numerical solution for second order initial value problems of Bratu-type, is optimal homotopy asymptotic method (OHAM). The main advantage of the used technique is achieving high accurate approximate solutions. In the numerical tables and graphics, our numerical results are compared with the exact ones.

References

- [1] Herişanu, N., Marinca, V., Dordea T. and Madescu, G. (2008) A New Analytical Approach to Nonlinear Vibration of an Electrical Machine. *Proceedings of the Romanian Academy, Series A*, **9**, 229-236.
- [2] Ali, J., Islam, S., Islam, S. and Zamand, G. (2010) The Solution of Multipoint Boundary Value Problems by the Optimal Homotopy Asymptotic Method. *Computers & Mathematics with Applications*, **59**, 2000-2006.
<http://dx.doi.org/10.1016/j.camwa.2009.12.002>
- [3] Ene, R.D., Marinca, V., Negrea, R. and Caruntu, B. (2012) Optimal Homotopy Asymptotic Method for Solving a Nonlinear Problem in Elasticity. *Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 14th International Symposium*, Timisoara, 26-29 September 2012, 98-102.
- [4] Esmaeilpour, M. and Ganji, D.D. (2010) Solution of the Jeffery-Hamel Flow Problem by Optimal Homotopy Asymptotic Method. *Computers & Mathematics with Applications*, **59**, 3405-3411.
<http://dx.doi.org/10.1016/j.camwa.2010.03.024>
- [5] Hashmi, M.S., Khan, N. and Iqbal, S. (2012) Optimal Homotopy Asymptotic Method for Solving Nonlinear Fredholm Integral Equations of Second Kind. *Applied Mathematics and Computation*, **218**, 10982-10989.

- <http://dx.doi.org/10.1016/j.amc.2012.04.059>
- [6] Marinca, V., Herişanu, N., Bota, C. and Marinca, B. (2009) An Optimal Homotopy Asymptotic Method Applied to the Steady Flow of a Fourth-Grade Fluid Past a Porous Plate. *Applied Mathematics Letters*, **22**, 245-251. <http://dx.doi.org/10.1016/j.aml.2008.03.019>
- [7] Abukhaled, M., Khuri, S. and Sayfy, A. (2012) Spline-Based Numerical Treatments of Bratu-Type Equations. *Palestine Journal of Mathematics*, **1**, 63-70.
- [8] Wazwaz, A. (2012) A Reliable Study for Extensions of the Bratu Problem with Boundary Conditions. *Mathematical Methods in the Applied Sciences*, **35**, 845-856. <http://dx.doi.org/10.1002/mma.1616>
- [9] Batiha, B. (2010) Numerical Solution of Bratu-Type Equations by the Variational Iteration Method. *Hacettepe Journal of Mathematics and Statistics*, **39**, 23-29.
- [10] Feng, X., He, Y. and Meng, J. (2008) Application of Homotopy Perturbation Method to the Bratu-Type Equations. *Topological Methods in Nonlinear Analysis*, **31**, 243-252.
- [11] Rashidinia, J., Maleknejad, K. and Taheri, N. (2013) Sinc-Galerkin Method for Numerical Solution of the Bratu's Problems. *Numerical Algorithms*, **62**, 1-11. <http://dx.doi.org/10.1007/s11075-012-9560-3>
- [12] Syam, M.I. and Hamdan, A. (2006) An Efficient Method for Solving Bratu Equations. *Applied Mathematics and Computation*, **176**, 704-713. <http://dx.doi.org/10.1016/j.amc.2005.10.021>
- [13] Wazwaz, A. (2005) Adomian Decomposition Method for a Reliable Treatment of the Bratu-Type Equations. *Applied Mathematics and Computation*, **166**, 652-663. <http://dx.doi.org/10.1016/j.amc.2004.06.059>

An Actual Survey of Dimensionality Reduction

Alireza Sarveniazi

Institut fuer Angewandte Forschung (IAF), Karlsruhe, Germany
Email: alireza.sarveniazi@hs-karlsruhe.de

Received 6 November 2013; revised 6 December 2013; accepted 15 December 2013

Copyright © 2014 by author and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Dimension reduction is defined as the processes of projecting high-dimensional data to a much lower-dimensional space. Dimension reduction methods variously applied in regression, classification, feature analysis and visualization. In this paper, we review in details the last and most new version of methods that extensively developed in the past decade.

Keywords

Dimensionality Reduction Methods

1. Introduction

Any progresses in efficiently using data processing and storage capacities need control on the number of useful variables. Researchers working in domains as diverse as computer science, astronomy, bio-informatics, remote sensing, economics, face recognition are always challenged with the reduction of the number of data-variables. The original dimensionality of the data is the number of variables that are measured on each observation. Especially when signals, processes, images or physical fields are sampled, high-dimensional representations are generated. High-dimensional data-sets present many mathematical challenges as well as some opportunities, and are bound to give rise to new theoretical developments [1].

In many cases, these representations are redundant and the variables are correlated, which means that eventually only a small sub-space of the original representation space is populated by the sample and by the underlying process. This is most probably the case, when very narrow process classes are considered. For the purpose of enabling low-dimensional representations with minimal information loss according dimension reduction methods are needed.

Hence, we are reviewing in this paper the most important dimensional reduction methods, including most traditional methods, such as principal component analysis (PCA) and non-linear PCA up to current state-of-art

methods published in various areas, such as signal processing and statistical machine learning literature. This actual survey is organized as follows: Section 2 reviews the linear nature of Principal component analysis and its relation with multidimensional scaling (classical scaling) in a comparable way. Section 3 introduces non-linear or Kernel PCA (KPCA) using the kernel-trick. Section 4 is about linear discriminant analysis (LDA), and we give an optimization model of LDA which is a measuring of a power of this method. In Section 5 we summarize another higher-order linear method, namely canonical correlation analysis (CCA)), which finds a low dimensional representation maximizing the correlation and of course its optimization-formulation. Section 6 reviews the relatively new version of PCA, the so-called oriented PCA (OPCA) which is introduced by Kung and Diamantaras [2] as a generalization of PCA. It corresponds to the generalized eigenvalue decomposition of a pair of covariance matrices, but PCA corresponds to the eigenvalue decomposition of only a single covariance matrix. Section 7 introduces principal curves and includes a characterization of these curves with an optimization problem which tell us when a given curve can be a principal curve. Section 8 gives a very compact summary about non-linear dimensional-reduction methods using neural networks which include the simplest neural network which has only three layers:

- 1) Input Layer
- 2) Hidden Layer (bottleneck)
- 3) Output Layer

and an auto-associative neural network with five layers:

- 1) Input Layer
- 2) Hidden Layer
- 3) Bottleneck
- 4) Hidden Layer
- 5) Output Layer

A very nice optimizing formulation is also given. In Section 9, we review the Nystroem method which is a very useful and well known method using the numerical solution of an integral equation. In Section 10, we look the multidimensional scaling (MDS) from a modern and more exact consideration view of point, specially a defined objective stress function arises in this method. Section 11 summarizes locally linear embedding (LLE) method which address the problem of nonlinear dimensionality reduction by computing low-dimensional neighborhood preserving embedding of high-dimensional data. Section 12 is about one of the most important dimensional-reduction method namely Graph-based method. Here we will see how the adjacency matrix good works as a powerful tool to obtain a small space which is in fact the eigen-space of this matrix. Section 13 gives a summary on Isomap and the most important references about Dijkstra algorithm and Floyd's algorithm are given. Section 14 is a review of Hessian eigenmaps method, a most important method in the so called manifold embedding. This section needs more mathematical backgrounds. Section 15 reviews most new developed methods such as

- vector quantization
- genetic and evolutionary algorithms
- regression

We have to emphasize here the all of given references in the body of survey are used and they are the most important references or original references for the related subject. To obtain more mathematical outline and sensation, we give an appendix about the most important backgrounds on the fractal and topological dimension definitions which are also important to understand the notion of intrinsic dimension.

2. Principal Component Analysis (PCA)

Principal component Analysis (PCA) [3] [4] [5]-[8] is a linear method that it performs dimensionality reduction by embedding the data into a linear subspace of lower dimensional. PCA is the most popular unsupervised linear method. The result of PCA is a lower dimensional representation from the original data that describes as much of the variance in the data as possible. This can be reached by finding a linear basis (possibly orthogonal) of reduced dimensionality for the data, in which the amount of variance in the data is maximal.

In the mathematical language, PCA attempts to find a linear mapping P that maximizes the cost function $tr(P^T A P)$, where A is the sample covariance matrix of the zero-mean data. Another words PCA maximizes $P^T A P$ with respect to P under the constraint the norm of each column v of P is 1, i.e., $\|v\|^2 = 1$. In fact

PCA solves the eigenvalue problem:

$$AP = \lambda P \text{ or } Av = \lambda v \quad (1.1)$$

Why the above optimization Problem is equivalent to the eigenvalue problem (1.1)? consider the convex form $v^T Av + \lambda(1 - v^T v)$, it is a straightforward calculation that the maximum happens when $Av = \lambda v$.

It is interesting to see that in fact PCA is identical to the multidimensional scaling (classical scaling) [9].

For the given data $\{x_i\}_{i=1}^N$ let $D = [d_{ij}]$ be the pairwise Euclidean matrix whose entries d_{ij} represent the Euclidean distance between the high-dimensional data points x_i and x_j . multidimensional scaling finds the linear mapping P such that maximizes the cost function:

$$\psi(Y) := \sum_{i,j} \left(d_{ij}^2 - \|y_i - y_j\|^2 \right), \quad (1.2)$$

in which $\|y_i - y_j\|$ is the Euclidean distance between the low-dimensional data points y_i and y_j , y_i is restricted to be $x_i A$, with $\|v_j\|^2 = 1$ for all column vector v_j of P . It can be shown [10] [11] that the minimum of the cost function $\psi(Y)$ is given by the eigen-decomposition of the Gram matrix $G = XX^T$ where $X = [x_i]$. Actually we can obtain the Gram matrix by double-centering the pairwise squared Euclidean distance matrix, *i.e.*, by computing:

$$g_{ij} = -\frac{1}{2} \left(d_{ij}^2 - \frac{1}{n} \sum_l d_{il}^2 - \frac{1}{n} \sum_l d_{jl}^2 + \frac{1}{n^2} \sum_{l,m} d_{lm}^2 \right). \quad (1.3)$$

Now consider the multiplication of principal eigenvectors of the double-centered squared Euclidean distance matrix (*i.e.*, the principal eigenvectors of the Gram matrix) with the square-root of their corresponding eigenvalues, this gives us exactly the minimum of the cost function in Equation (1.2).

It is well known that the eigenvectors u_i and v_i of the matrices $X^T X$ and XX^T are related through $\sqrt{\lambda_i} v_i = Xu_i$ [12], it turns out that the similarity of classical scaling to PCA. The connection between PCA and classical scaling is described in more detail in, *e.g.*, [11] [13]. PCA may also be viewed upon as a latent variable model called probabilistic PCA [14]. This model uses a Gaussian prior over the latent space, and a linear-Gaussian noise model.

The probabilistic formulation of PCA leads to an EM-algorithm that may be computationally more efficient for very high-dimensional data. By using Gaussian processes, probabilistic PCA may also be extended to learn nonlinear mappings between the high-dimensional and the low-dimensional space [15]. Another extension of PCA also includes minor components (*i.e.*, the eigenvectors corresponding to the smallest eigenvalues) in the linear mapping, as minor components may be of relevance in classification settings [16]. PCA and classical scaling have been successfully applied in a large number of domains such as face recognition [17], coin classification [18], and seismic series analysis [19].

PCA and classical scaling suffer from two main drawbacks. First, in PCA, the size of the covariance matrix is proportional to the dimensionality of the data-points. As a result, the computation of the eigenvectors might be infeasible for very high-dimensional data. In data-sets in which $n < D$, this drawback may be overcome by performing classical scaling instead of PCA, because the classical scaling scales with the number of data-points instead of with the number of dimensions in the data. Alternatively, iterative techniques such as Simple PCA [20] or probabilistic PCA [14] may be employed. Second, the cost function in Equation (1.2) reveals that PCA and classical scaling focus mainly on retaining large pairwise distances d_{ij}^2 , instead of focusing on retaining the small pairwise distances, which is much more important.

3. Non-Linear PCA

Non-linear or Kernel PCA (KPCA) is in fact the reconstruction from linear PCA in a high-dimensional space that is constructed using a given kernel function [21]. Recently, such reconstruction from linear techniques using the kernel-trick has led to the proposal of successful techniques such as kernel ridge regression and Support Vector Machines [22]. Kernel PCA computes the principal eigenvectors of the kernel matrix, rather than those of the covariance matrix. The reconstruction from PCA in kernel space is straightforward, since a

kernel matrix is similar to the inner product of the data-points in the high-dimensional space that is constructed using the kernel function. The application of PCA in the kernel space provides Kernel PCA the property of constructing nonlinear mappings.

Kernel PCA computes the kernel matrix $K = [k_{ij}]$ of the data-points x_i . The entries in the kernel matrix are defined by

$$k_{ij} := \kappa(x_i, x_j) \quad (1.4)$$

where κ is a kernel function [22], which may be any function that gives rise to a positive-semi-definite kernel K . Subsequently, the kernel matrix \bar{K} is double-centered using the following modification of the entries

$$\bar{k}_{ij} = -\frac{1}{2} \left(k_{ij} - \frac{1}{n} \sum_l k_{il} - \frac{1}{n} \sum_l k_{jl} + \frac{1}{n^2} \sum_{l,m} k_{lm} \right). \quad (1.5)$$

The centering operation corresponds to subtracting the mean of the features in traditional PCA: it subtracts the mean of the data in the feature space defined by the kernel function κ . Hence, the data in the features space defined by the kernel function is zero-mean. Subsequently, the principal d eigenvectors v_i of the centered kernel matrix are computed. The eigenvectors of the covariance matrix a_i (in the feature space constructed by κ) can now be computed, since they are related to the eigenvectors of the kernel matrix v_i (see, e.g., [12]) through

$$a_i = \frac{1}{\sqrt{\lambda_i}} v_i \quad (1.6)$$

In order to obtain the low-dimensional data representation, the data is projected onto the eigenvectors of the covariance matrix (a_i) . The result of the projection (i.e., the low-dimensional data representation $Y = (y_i)$) is given by:

$$y_i = \left(\sum_{j=1}^n a_1^{(j)} \kappa(x_j, x_i), \dots, \sum_{j=1}^n a_d^{(j)} \kappa(x_j, x_i) \right)$$

where $a_i^{(j)}$ indicates the j^{th} value in the vector a_i and κ is the kernel function that was also used in the computation of the kernel matrix. Since Kernel PCA is a kernel-based method, the mapping performed by Kernel PCA relies on the choice of the kernel function κ . Possible choices for the kernel function include the linear kernel (making Kernel PCA equal to traditional PCA), the polynomial kernel, and the Gaussian kernel that is given in [12]. Notice that when the linear kernel is employed, the kernel matrix K is equal to the Gram matrix, and the procedure described above is identical to classical scaling (previous section).

An important weakness of Kernel PCA is that the size of the kernel matrix is proportional to the square of the number of instances in the data-set. An approach to resolve this weakness is proposed in [23] [24]. Also, Kernel PCA mainly focuses on retaining large pairwise distances (even though these are now measured in feature space).

Kernel PCA has been successfully applied to, e.g., face recognition [25], speech recognition [26], and novelty detection [25]. Like Kernel PCA, the Gaussian Process Latent Variable Model (GPLVM) also uses kernel functions to construct non-linear variants of (probabilistic) PCA [15]. However, the GPLVM is not simply the probabilistic counterpart of Kernel PCA: in the GPLVM, the kernel function is defined over the low-dimensional latent space, whereas in Kernel PCA, the kernel function is defined over the high-dimensional data space.

4. Linear Discriminant Analysis (LDA)

The main Reference here is [27] see also [28]. The LDA is a method to find a linear transformation that maximizes class separability in the reduced dimensional space. The criterion in LDA is in fact to maximize between class scatter and minimize within-class scatter. The scatters are measured by using scatter matrices. Let we have r class C_i each including n_i points $x_j^i \in \mathbb{R}^l$ and set $X = [\tilde{C}_1, \dots, \tilde{C}_r] \in \mathbb{R}^{l \times n}$, where $\tilde{C}_i = [x_1^i, \dots, x_{n_i}^i]$

and $n = \sum_{i=1}^r n_i$. Let $\bar{x}^i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_j^i$ and $\bar{x} = \frac{1}{n} \sum_{i=1}^r \sum_{j=1}^{n_i} x_j^i$.

Now we define three scatter matrices:

The between-class scatter matrix $S_b := \sum_{i=1}^r n_i (\bar{x}^i - \bar{x})(\bar{x}^i - \bar{x})^T$,

The within-class scatter matrix $S_w := \sum_{i=1}^r \sum_{j=1}^{n_i} (x_j^i - \bar{x}^i)(x_j^i - \bar{x}^i)^T$,

The total scatter matrix $S_t := \sum_{i=1}^r \sum_{j=1}^{n_i} (x_j^i - \bar{x})(x_j^i - \bar{x})^T$. Actually LDA is a method for the following optimization problem:

$$\arg \max_{U \in \mathbb{R}^{l \times m}} \frac{|U^T S_b U|}{|U^T S_w U|}$$

Hence in this way the dimension is reduced from l to m by a linear transformation \bar{U} which is the solution of above optimization problem. Although we know from Fukunaga (1990), (see [27] and [29]) that the eigenvectors corresponding to the $r-1$ largest eigenvalues of

$$S_b u = \lambda S_w u$$

form the columns of U as above for LDA.

5. Canonical Correlation Analysis (CCA)

CCA is an old method back to the works of Hotelling 1936 [30], recently Sun *et al.* [31] used CCA as an unsupervised feature fusion method for two feature sets describing the same data objects. CCA finds projective directions which maximize the correlation between the feature vectors of the two feature sets.

Let $X = x_{i=1}^n$ and $Y = y_{i=1}^n$ be two data set of n points in \mathbb{R}^p and \mathbb{R}^q respectively, associate with them we have two matrices:

$$A_X = [x_1 - \bar{x}, \dots, x_n - \bar{x}] \in \mathbb{R}^{p \times n}, \quad A_Y = [y_1 - \bar{y}, \dots, y_n - \bar{y}] \in \mathbb{R}^{q \times n}$$

where $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ and $\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$ are the means of x_i and y_i s, respectively.

Actually CCA is a method for the following optimization problem:

$$\arg \max_{U_X, U_Y} \frac{U_X^T A_X A_Y^T U_Y}{\sqrt{U_X^T A_X A_X^T U_X} \sqrt{U_Y^T A_Y A_Y^T U_Y}}$$

which can be modified as

$$\arg \max_{U_X, U_Y} U_X^T A_X A_Y^T U_Y, \quad U_X^T A_X A_X^T U_X = 1, \quad U_Y^T A_Y A_Y^T U_Y = 1$$

Assume the pair of projective directions (U_X^*, U_Y^*) be the solution of above optimization problem, we can find another pair of projective directions by solving

$$\arg \max_{U_X, U_Y} U_X^T A_X A_Y^T U_Y, \quad U_X^{*T} A_X A_X^T U_X^* = U_Y^{*T} A_Y A_Y^T U_Y^* = 0, \quad U_X^T A_X A_X^T U_X = U_Y^T A_Y A_Y^T U_Y = 1$$

repeating the above process $m-1$ times we obtain a m -dimensional specs of linear combination of these vector-solutions.

In fact we can obtain this m -dimensional space with solving of the paired eigenvalue problem:

$$A_X A_Y^T (A_Y A_Y^T)^{-1} U_X = \lambda A_X A_X^T U_X, \quad A_Y A_X^T (A_X A_X^T)^{-1} U_Y = \lambda A_Y A_Y^T U_Y$$

and the eigenvectors $(U_X^{(i)}, U_Y^{(i)})$, $i=1, \dots, m$ corresponding to the m largest eigenvalues are the pairs of projective directions for CCA see [31]. Hence

$$\{U_X^{(i)T} A_X, i=1, \dots, m\} \quad \text{and} \quad \{U_Y^{(i)T} A_Y, i=1, \dots, m\}$$

compose the feature sets extracted from A_x and A_y by CCA. It turns out that the number m is determined as the number of nonzero eigenvalue.

6. Oriented PCA (OPCA)

Oriented PCA is introduced by Kung and Diamantaras [2] as a generalization of PCA. It corresponds to the generalized eigenvalue decomposition of a pair of covariance matrices in the same way that PCA corresponds to the eigenvalue decomposition of a single covariance matrix. For the given pair of vectors u and v the objective function maximized by OPCA is given as follows:

$$\arg \max_w \frac{E(w^T u)^2}{E(w^T v)^2} = \frac{w^T A_u w}{w^T A_v w}$$

where $A_u := E(uu^T)$, $A_v := E(vv^T)$. A solution w_1^* of above optimization problem is called Principal oriented component and it is the generalized eigenvector of matrix pair $[A_u, A_v]$ corresponding to maximum generalized eigenvalue λ_1 . Since A_u and A_v are symmetric all the generalized eigenvalues are real and thus they can be arranged in decreasing order, as with ordinary PCA. Hence we will obtain the rest generalized eigenvectors $w_2^*, w_3^*, \dots, w_m^*$, as second, third, ..., m^{th} oriented principal components. All of these solutions are the solutions under the orthogonality constraint:

$$w_i^{*T} A_u w_j = w_i^{*T} A_v w_j = 0, \text{ for } i \neq j$$

7. Principal Curves and Surfaces

By the definition, principal curves are smooth curves that pass through the middle of multidimensional data sets, see [32]-[34] as main references and also [35] and [36].

Given the n -dimensional random vector $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ with probability density function $p(x)$. Let $f: \mathbb{R} \rightarrow \mathbb{R}^n$ be the given smooth curve which can be parametrized by a real value θ (actually we can choose $\theta \in [0, 1]$). Hence we have $f(\theta) = (f_1(\theta), \dots, f_n(\theta))$.

We can associate to the curve f the projection index $\Theta_f: \mathbb{R}^n \rightarrow \mathbb{R}$ geometrically as the value of θ corresponding to the point on the curve f that under Euclidean metric is the closet point to x .

We say f is self-consistent if each point $f(\theta)$ is the mean of all points in the support of density function p that are projected on θ , i.e.,

$$E[x | \Theta_f(x) = \theta] = f(\theta).$$

It is shown in [32] that the set of principal curves do not intersect themselves and they are self-consistent. Most important fact about principal curves which proved in [32] is a characterization of these curves with an optimization Problem:

Theorem 1 A curve f is a principal curve (associate with the data set $\{x^{(i)}\}_{i=1}^N$) iff it solves following optimization problem

$$\min_{f, \Theta_f} \sum_{i=1}^N \left\| x^{(i)} - f(\Theta_f(x^{(i)})) \right\|^2, \quad (1.7)$$

Of course to solve (or even estimate) minimization (0.7) is a complex problem, to estimate f and θ in [32] an iterative algorithm has given. It started with $f(\theta) = E(x) + \theta u_1$, where u_1 is the first eigenvector of covariance matrix of x and $\theta = \Theta_f(x)$. Then it iterates the two steps:

- For a fixed θ , minimize $\|x - f(\theta)\|$ by setting

$$f_j(\theta) = E[x_j | \Theta_f(x) = \theta] \text{ for each } j$$

- Fix f and set $\Theta_f(x) = \theta$ for each x until the change in $\|x - f(\theta)\|$ is less than a threshold.

One can find in [37] another formulation of the principal curves, along with a generalized EM algorithm for its estimation under Gaussian pdf $p(x)$. Unfortunately except for a few special cases, it is an open problem for what type of distributions do principal curves exist, how many principal curves there exist and which properties they have see [36]. In recent years the concept of principal curves has been extended to higher dimensional principal surfaces, but of course the estimation algorithms are not smooth as the curves.

8. Non-Linear Methods Using Neural Networks

Given Input variables $\{x_1, x_2, \dots, x_N\}$, neural networks getting this input and gives output variables $\{y_1, y_2, \dots, y_m\}$ with

$$y_j = y_j(x, w), j = 1, \dots, m$$

where the weights w are determined by training the neural network using a set of given instances and a cost function see [38]. Over the last two decades there are several developments based on a ring architectures and learning algorithms of dimensional reduction techniques could be implemented using neural networks, see [35] [36] [38]-[40]. Consider the simplest neural network which has only three layers:

- 1) Input Layer
- 2) Hidden Layer (bottleneck)
- 3) Output Layer

there are two steps here:

- In order to obtain the data at node k of the hidden layer, we have to consider any inputs x_i in combination with their associated weight's w_{ik} along with a threshold term (or called bias in some references) ρ_k . Now they are ready passing through to the corresponding activation ϕ_k , hence we are building up the expression $\phi_k(\rho_k + \sum_i w_{ik} x_i)$.
- Here we have to repeat step (1) with changing original data x_i with new one namely $\phi_k(\rho_k + \sum_i w_{ik} x_i)$, of course according the threshold ρ_j and possibly new output function ϕ_{out} . Hence we have:

$$y_j = \phi_{out} \left(\rho_j + \sum_k w_{kj} \phi_k \left(\rho_k + \sum_i w_{ik} x_i \right) \right).$$

We observe that the first part of network reduces the input data into the lower-dimensional space just as same as a linear PCA, but the second part decodes the reduced data into the original domain [36] [35]. Note that only by adding two more hidden layers with nonlinear activation functions, one between the input and the bottleneck, the other between the bottleneck and the output layer, the PCA network can be generalized to obtain non-linear PCA. One can extend this idea from the feed-forward neural implementation of PCA extending to include non-linear activation function in the hidden layers [41]. In this framework, the non-linear PCA network can be considered of as an auto-associative neural network with five layers:

- 1) Input Layer
- 2) Hidden Layer
- 3) Bottleneck
- 4) Hidden Layer
- 5) Output Layer

If $\Theta_f : \mathbb{R}^n \rightarrow \mathbb{R}^l$ be the function modeled by layers (1), (2) and (3), and $f : \mathbb{R}^l \rightarrow \mathbb{R}^n$ be the modeled function by layers (3), (4) and (5), in [35] have been shown that weights of the non-linear PCA network are determined such that the following optimization Problem solved:

$$\min_{f, \Theta_f} \sum_{i=1}^N \|x_i - f(\Theta_f(x_i))\|^2,$$

As we have seen in the last section the function f must be Principal curve(surface). In the thesis [42], one

can find comparison between PCA, Vector Quantization and five layer neural networks, for reducing the dimension of images.

9. Nystroem Method

The Nystroem Method is a well known technique for finding numerical approximations of generic integral equation and specially to eigenfunction problems of the following form:

$$\int_a^b W(x, y) f(y) dy = \lambda f(x)$$

We can divide the interval $[a, b]$ into n points $\theta_1, \dots, \theta_n$ where

$$\theta_{i+1} = \theta_i + i\Delta \quad \text{and} \quad \Delta = \frac{b-a}{n}, \quad \theta_1 = a, \quad \theta_n = b.$$

Now consider the simple quadrature rule:

$$\frac{b-a}{n} \sum_{j=1}^n W(x, \theta_j) \tilde{f}(\theta_j) = \lambda \tilde{f}(x) \quad (1.8)$$

which \tilde{f} approximates f , for $x = \theta_i$ we obtain a system of n equations:

$$\frac{b-a}{n} \sum_{j=1}^n W(\theta_i, \theta_j) \tilde{f}(\theta_j) = \lambda_i \tilde{f}(\theta_i), \quad i = 1, \dots, n$$

without loss of generality we can shift interval $[a, b]$ to unit interval $[0, 1]$ and change the above system of equations to the following eigenvalue problem:

$$\tilde{A}(f) = nD\tilde{f} \quad (1.9)$$

where $A = [W(\theta_i, \theta_j)]$, $\tilde{f} = (\tilde{f}_1, \dots, \tilde{f}_n)$ and $D = \text{diag}(\lambda_1, \dots, \lambda_n)$, substituting back into 0.8 yields the Nystroem extension for each \tilde{f}_i :

$$\tilde{f}_i(x) = \frac{1}{n\lambda_i} \sum_{j=1}^n W(x, \theta_j) \tilde{f}_i(\theta_j)$$

We can extend above arguments for $x \in \mathbb{R}^n$ and $n > 1$, see [42].

Motivated from 0.9 our main question is if A be a given $n \times n$ real symmetric matrix with small rank r , i.e., $r \ll n$, can we approximate the eigenvectors and eigenvalues of A using those of a small sub-matrix of A ?

Nystroem method gives a positive answer to this question. Actually we can assume that the r randomly chosen samples come first and the $n-r$ samples come next. Hence the matrix A in 0.9 can have following form:

$$A = \begin{pmatrix} E & B \\ B^T & C \end{pmatrix}$$

Hence E represents the sub-block of weights among the random samples, B contains the weights from the random samples to the rest of samples and C contains the weights between all of remaining samples. Since $r \ll n$, C must be a large matrix. Let \bar{U} denote the approximate eigenvectors of A , the Nystroem extension method gives:

$$\bar{U} = \begin{pmatrix} U \\ B^T U D^{-1} \end{pmatrix}$$

where U and D are eigenvectors and diagonal matrix associate with E , i.e., $E = U^T D U$. Now the associated approximation of A , which we denote it with \tilde{A} , then we have:

$$\tilde{A} = \bar{U} D \bar{U}^T = \begin{pmatrix} U \\ B^T U D^{-1} \end{pmatrix} D \begin{pmatrix} U^T & D^{-1} U^T B \end{pmatrix} = \begin{pmatrix} U D U^T & B \\ B^T & B^T E^{-1} B \end{pmatrix} = \begin{pmatrix} E & B \\ B^T & B^T E^{-1} B \end{pmatrix} = \begin{pmatrix} E \\ B^T \end{pmatrix} E^{-1} \begin{pmatrix} E & B \end{pmatrix}$$

The last equation is called “bottleneck” form. There is a very interesting application of this form in Spectral

Grouping which it was possible to construct the exact eigen-decomposition of A using the eigen-decomposition of smaller matrix rank r . Also Fowlkes et al have given an application of the Nystroem method to NCut Problem, see [43].

10. Multidimensional Scaling (MDS)

Given N point $X = \{x_1, x_2, \dots, x_N\} \subset \mathbb{R}^n$ and build up the distance matrix $\Delta = [d_{ij}]$ where $d_{ij} = \|x_i - x_j\|$, (or in general $d_{ij} = d(x_i, x_j)$ for some metric which defined d) MDS (better to say a m -dimensional MDS) is a technique that produces output points $\{y_1, y_2, \dots, y_N\} \subset \mathbb{R}^m$ such that the distances d_{ij} are as close as possible to a function f of the corresponding proximity's $f(d_{ij})$. From [36], whether this function ϕ is linear or non-linear, MDS is called either metric or non-metric. Define an objective stress function

MDS-PROCEDURE:

- Define an objective stress function and stress factor α , that it depends on $\sum_{i,j} f(d_{ij})^2$ or on $\sum_{i,j} d_{ij}^2$

$$\Phi_S^f(\Delta, X, f) := \sqrt{\frac{\sum_{i,j} (f(d_{ij}) - d_{ij})^2}{\alpha}} \quad (1.10)$$

- Now if for a given X as above, find f^* that minimize 0.10, i.e.,

$$\Phi_S(\Delta, X, f^*) = \min_f \Phi_S^f(\Delta, X, f)$$

- Determine the optimal data set \tilde{X} by

$$\Phi_S(\Delta, \tilde{X}, f^*) = \min_X \Phi_S^f(\Delta, X, f^*)$$

If we use Euclidean distance and take $f = id$ in Equation (1.10) the produced output data set should be coincide to the Principal component of $\text{cov}(X)$ (without re-scaling to correlation), hence in this special case MDS and PCA are coincide (see [44]) There exist an alternative method to MDS, namely Fast Map see [45] [46].

11. Locally Linear Embedding (LLE)

Locally linear embedding is an approach which address the problem of nonlinear dimensionality reduction by computing low-dimensional neighborhood preserving embedding of high-dimensional data. A data set of dimensionality n , which is assumed to lie on or near a smooth nonlinear manifold of dimensionality $m \ll n$, is mapped into a single **global** coordinate system of lower-dimensionality m . The global nonlinear structure is recovered by locally linear fits. As usual given a Data set of N points on a n -dimensional points $\{x_i\}_{i=1}^N$ from some underlying manifold. Without loss of generality we can assume each data point and its neighbors lie on are close to a locally linear sub-manifold. By a linear transform, consisting of a translation, rotation and rescaling, the high-dimensional coordinates of each neighborhood can be mapped to global internal coordinates on the manifold. In order to map the high-dimensional data to the single global coordinate system of the manifold such that the relationships between neighboring points are preserved. This proceeds in three steps:

- Identify neighbors of each data point x_i . this can be done by finding the K nearest neighbors, or choosing all points within some fixed radius ε .
- Compute the weights $[w_{ij}]$ that best linearly reconstruct x_i from its neighbors.
- Find the low-dimensional embedding vector y_i which is the best reconstructed by the weights determined in the previous step.

After finding the nearest neighbors in the first step, the second step must compute a local geometry for each locally linear sub-manifold. This geometry is characterized by linear coefficients that reconstruct each data point from its neighbors.

$$\min_w \sum_{i=1}^n \left\| x_i - \sum_{j=1}^K w_{ij} x_{N_{i(j)}} \right\|^2 \quad (1.11)$$

where $N_i(j)$ is the index of the j^{th} neighbor of the point. It then selects code vectors so as to preserve the

reconstruction weights by solving

$$\min_Y \sum_{i=1}^n \left\| y_i - \sum_{j=1}^K w_{ij} y_{N_{i(j)}} \right\|^2 \quad (1.12)$$

This objective can be restated as

$$\min_Y \text{Tra}(Y^T Y L) \quad (1.13)$$

where $L = (I - W)T(I - W)$.

The solution for Y can have an arbitrary origin and orientation. In order to make the problem well-posed, those two degrees of freedom must be removed. Requiring the coordinates to be centered on origin ($\sum_{i=1}^n y_i = 0$), and constructing the embedding vectors to have unit covariance ($Y^T Y = I$), removes the first and second degrees of freedom respectively. The cost function can be optimized initially by the second of those two constraints. Under this constraint, the cost is minimized when the column of Y^T (rows of Y) are the eigenvectors with the lowest eigenvalues of L . Discarding the eigenvector associated with eigenvalue 0 satisfies the first constraint.

12. Graph-Based Dimensionality Reduction

As before given a data set X include N points in \mathbb{R}^n , i.e., $X = \{x_1, x_2, \dots, x_N\}$, we associate to X a weighted undirected graph with N vertices and use the Laplacian matrix which defined see [47]. In order to define an undirected graph we need define a pair $(V; E)$ of sets, V the set of vertices and E the set of edges. we follows here the method introduced in [48].

we say $v_i \in V$ if $x_i \in X$ and $(v_i, v_j) \in E$ iff x_i and x_j are **close**. But what it means to be **close**? there are two variations define it:

- ε -neighborhoods, which ε is a positive small real number.
 x_i and x_j are **close** iff $\|x_i - x_j\|^2 \leq \varepsilon$, where the norm is as usual the Euclidean norm in \mathbb{R}^n .
 - K nearest neighbors. Here K is a natural number.
 x_i and x_j are **close** iff x_i is among K nearest neighbors of x_j or x_j is among K nearest neighbors of x_i . that means this relation is a symmetric relation.
- To associate the weights to edges, as well, there is two variations:
- Heat kernel, which γ is a real number.

$$w_{ij} = \begin{cases} \exp\left(-\frac{\|x_i - x_j\|^2}{\gamma}\right), & \text{if } x_i \text{ and } x_j \text{ are close} \\ 0, & \text{otherwise} \end{cases}$$

- Simple adjacency with parameter $\gamma = \infty$.

$$w_{ij} = \begin{cases} 1, & \text{if } x_i \text{ and } x_j \text{ are close} \\ 0, & \text{otherwise} \end{cases}$$

We assume our graph, defined as above, is connected, otherwise proceed following for each connected component. Set $d_{ii} = \sum_{j=1}^N w_{ij}$ and $d_{ij} = 0$ if $i \neq j$, $D = [d_{ij}]$, $W = [w_{ij}]$. $L = D - W$ is the Laplacian matrix of the

graph, which is a symmetric, positive semi-definite matrix, so can be thought of as an operator on the space of real functions defined on the vertices set V of Graph.

Compute eigenvalues and eigenvectors for the generalized eigenvector problem:

$$Lf = \lambda Df$$

Let f_0, f_1, \dots, f_{N-1} be the solutions of the above eigenvalue problem, ordered according to their eigenvalues,

$$\begin{aligned}
Lf_0 &= \lambda Df_0 \\
Lf_1 &= \lambda Df_1 \\
&\vdots \\
Lf_{N-1} &= \lambda Df_{N-1} \\
0 &= \lambda_0 \leq \lambda_1 \leq \dots \leq \lambda_{N-1}
\end{aligned}$$

We leave out the eigenvector (trivial eigenfunction) corresponding to eigenvalue 0, which is a vector with all component equal to 1 and use next m eigenvectors for embedding in m -dimensional Euclidean space:

$$x_i \mapsto (f_1^{(i)}, \dots, f_m^{(i)})$$

which $f^{(i)}$ means i^{th} component of the vector f . This called the Laplacian Eigenmap embedding by Belkin and Nioqi, see [48].

13. Isomap

Like LLE the Isomap algorithm proceeds in three steps:

- Find the neighbors of each data point in high-dimensional data space.
- Compute the geodesic pairwise distances between all points.
- Embed the data via MDS so as preserve those distances

Again like LLE, the first, the first step can be performed by identifying the K -nearest neighbors, or by choosing all points within some fixed radius, ε . These neighborhood relations are represented by graph G in which each data point is connected to its nearest neighbors, with edges of weights $d_x(i, j)$ between neighbors.

The geodesic distances $d_x(i, j)$ between all pairs of points on the manifold M are then estimated in the second step. Isomap approximates $d_M(i, j)$ as the shortest path distance $d_G(i, j)$ in the graph G . This can be done in different ways including Dijkstra algorithm [49] and Floyd's algorithm [50]

14. Hessian Eigenmaps Method

High dimensional data sets arise in many real-world applications. These data points may lie approximately on a low dimensional manifold embedded in a high dimensional space. Dimensionality reduction (or as in this case, called manifold learning) is to recover a set of low-dimensional parametric representations for the high-dimensional data points, which may be used for further processing of the data. More precisely consider a d -dimensional parametrized manifold \mathcal{M} embedded in \mathbb{R}^n where $(d < n)$ characterized by a nonlinear map $\psi: \mathcal{X} \subset \mathbb{R}^d \mapsto \mathbb{R}^n$, where \mathcal{X} is a compact and connected subset of \mathbb{R}^d . Here \mathbb{R}^n is the high-dimensional data space with $\mathcal{M} = \psi(\mathcal{X})$ being the manifold containing data points and \mathbb{R}^d is the low-dimensional parameter space. Suppose we have a set of data points x_1, x_2, \dots, x_N sampled from the manifold \mathcal{M} with

$$x_i = \psi(y_i), i = 1, 2, \dots, N,$$

for some $y_i \in \mathcal{X}$. Then the dimensionality reduction problem is to recover the parameter points y_i s and the map ψ from y_i s.

Of course, this problem is not well defined for a general nonlinear map ψ . However, as is shown by Donoho and Grimes in the derivation of the Hessian Eigenmaps method [51], if ψ is a local isometric map, then $y = \psi^{-1}(x)$ is uniquely determined up to a rigid motion and hence captures the geometric structure of the data set.

Given that the map ψ defined as above, is a local isometric embedding, the map $\phi = \psi^{-1}: \mathcal{M} \subset \mathbb{R}^n \mapsto \mathbb{R}^d$ provides a locally isometric coordinate system for \mathcal{M} . Each component of ϕ is a function defined on \mathcal{M} that provides one coordinate. The main idea of the Hessian Eigenmaps is to introduce a Hessian operator and a functional called the \mathcal{H} -functional defined for functions on \mathcal{M} , for which the null space consists of the d coordinate functions and the constant function. Let $f: \mathcal{M} \mapsto \mathbb{R}$ be a function defined on \mathcal{M} and let x_0 be an interior point of manifold \mathcal{M} . We can define a function $g: \mathcal{X} \mapsto \mathbb{R}$ as $g(y) = f(\psi(y))$, where

$\mathcal{X} = \phi(\mathcal{M}) \subset \mathbb{R}^d$ and $y = [y_1, y_2, \dots, y_d]^T \in \mathcal{X}$ is called a pullback of f to \mathcal{X} . Let $y_0 = \phi(x_0)$. We call the Hessian matrix of g at y_0 the Hessian matrix of function f at x_0 in the isometric coordinate and we denote it by $H_f^{iso}(x_0)$. Then $(H_f^{iso} f)_{i,j}(x_0) = \frac{\partial^2 g(y_0)}{\partial y_i \partial y_j}$. From the Hessian matrix, we define a \mathcal{H} -functional of f in isometric coordinates, denoted by \mathcal{H}_f^{iso} , as

$$\mathcal{H}_f^{iso} = \int_{\mathcal{M}} \|H_f^{iso} f(x)\|_F^2 dx, \quad (1.14)$$

where dx is a probability measure on \mathcal{M} which has strictly positive density everywhere on the interior of \mathcal{M} . It is clear that \mathcal{H}_f^{iso} of the d component functions of ϕ are zero as their pullbacks to \mathcal{X} are linear functions. Indeed, $\mathcal{H}_f^{iso}(\cdot)$ has a $(d+1)$ -dimensional null space, consisting of the span of the constant functions and the d component functions of ϕ ; see [51] (Corollary 4). The Hessian matrix and the \mathcal{H} -functional in isometric coordinates introduced above are unfortunately not computable without knowing the isometric coordinate system ϕ first. To obtain a functional with the same property but independent of the isometric coordinate system ϕ , a Hessian matrix and the \mathcal{H} -functional in local tangent coordinate systems are introduced in [51]. Qiang Ye and Weifeng Zhi [52] developed a discrete version of the Hessian Eigenmaps method of Donoho and Grims.

15. Miscellaneous

15.1. Vector Quantization

The main references for vector quantization are [40] and [53]. In [53] it is introduced a hybrid non-linear dimension reduction method based on combining vector quantization for first clustering the data, after constructing the Voronoi cell clusters, applying PCA on them. In [40] both non-linear method *i.e.*, vector quantization and non-linear PCA (using a five layer neural network) on the image data set have been used. It turns out that the vector quantization achieved much better results than non-linear PCA.

15.2. Genetic and Evolutionary Algorithms

These algorithms introduced in [54] are in fact optimization algorithms based on Darwinian theory of evolution which uses natural selection and genetics to find the optimized solution among members of competing population. There are several references for genetic and evolutionary algorithms [55], see [56] for more detail. An evolutionary algorithm for optimization is different from classical optimization methods in several ways:

- Random Versus Deterministic Operation
- Population Versus Single Best Solution
- Creating New Solutions Through Mutation
- Combining Solutions Through Crossover
- Selecting Solutions Via “Survival of the Fittest”
- Drawbacks of Evolutionary Algorithms

In [55] using genetic and evolutionary algorithms combine with a k-nearest neighbor classifier to reduce the dimension of feature set. Here Input is population matrices which are in fact random transformation matrices $\{W_{m \times N}\}^{(i)}$, then algorithms will find output $Y_{m \times N}$ so that the k-nearest neighbor classifier using the new features $B_{m \times r} = Y_{m \times N} X_{N \times r}$ classifies the training data most accurately.

15.3. Regression

We can use Regression methods for dimension reduction when we are looking for a variable function $y = f(x_1, \dots, x_n)$ for a given data set variables $\{x_i\}$. Under assumption that the x_i s are uncorrelated and relevant to expanding the variation in y . Of course in modern data mining applications however such assumptions rarely hold. Hence we need a dimension reduction for such a case. We can list well-known dimension

reduction methods as follows:

- The Wrapper method in machine learning community [57]
- Projection pursuit regression [36] [58]
- Generalized linear models [59] [60]
- Adaptive models [61]
- Neural network models and sliced regression and Principal hessian direction [62]
- Dimension reduction for conditional mean in regression [63]
- Principal manifolds and non-linear dimension reduction [64]
- Sliced regression for dimension reduction [65]
- Canonical correlation [66]

Acknowledgements

Our research has received funding from the (European Union) Seventh Framework Programme ([FP7/2007-2013]) under grant agreement n [314329]. we would like to thank Eu-Commission for the support.

References

- [1] Donoho, D.L. (2000) High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality. Lecture Delivered at the “Mathematical Challenges of the 21st Century” Conference of the American Math. Society, Los Angeles. <http://www-stat.stanford.edu/donoho/Lectures/AMS2000/AMS2000.html>
- [2] Diamantaras, K.I. and Kung, S.Y. (1996) Principal Component Neural Networks: Theory and Applications. John Wiley, NY.
- [3] Person, K. (1901) On Lines and Planes of Closest Fit to System of Points in Space. *Philosophical Magazine*, **2**, 559-572. <http://dx.doi.org/10.1080/14786440109462720>
- [4] Jenkins, O.C. and Mataric, M.J. (2002) Deriving Action and Behavior Primitives from Human Motion Data. *International Conference on Robots and Systems*, **3**, 2551-2556.
- [5] Jain, A.K. and Dubes, R.C. (1962) Algorithms for Clustering Data. Prentice Hall, Upper Saddle River.
- [6] Mardia, K.V., Kent, J.T. and Bibby, J.M. (1995) Multivariate Analysis Probability and Mathematical Statistics. Academic Press, Waltham.
- [7] (2002) Francesco Camastra Data Dimensionality Estimation Methods, a Survey INFM-DISI, University of Genova, Genova.
- [8] Fukunaga, K. (1982) Intrinsic Dimensionality Extraction, in Classification, Pattern Recognition and Reduction of Dimensionality, Vol. 2 of Handbook of Statistics, North Holland, 347-362.
- [9] Torgerson, W.S. (1952) Multidimensional Scaling I: Theory and Methode. *Psychometrika*, **17**, 401-419. <http://dx.doi.org/10.1007/BF02288916>
- [10] Teng, L., Li, H., Fu, X., Chen, W. and Shen, I-F. (2005) Dimension Reduction of Microarray Data Based on Local Tangent Space Alignment. *Proceedings of the 4th IEEE international Conference on Cognitive Informatics*, 154-159.
- [11] Williams, C.K.I. (2002) On a Connection between Kernel PCA and Metric Multidimensional Scaling. *Machine Learning*, **46**, 11-19. <http://dx.doi.org/10.1023/A:1012485807823>
- [12] Chatfield, C. and Collins, A.J. (1980) Introduction to Multivariate Analysis. Chapman and Hill. <http://dx.doi.org/10.1007/978-1-4899-3184-9>
- [13] Platt, J.C. (2005) FastMap, MetricMap, and Landmark MDS are all Nyström algorithms. *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*, **15**, 261-268.
- [14] Roweis, S.T. (1997) EM Algorithms for PCA and SPCA. *Advances in Neural Information Processing Systems*, **10**, 626-632.
- [15] Lawrence, N.D. (2005) Probabilistic Non-Linear Principal Component Analysis with Gaussian Process Latent Variable Models. *Journal of Machine Learning Research*, **6**, 1783-1816.
- [16] Welling, M., Rosen-Zvi, M. and Hinton, G. (2004) Exponential Family Harmoniums with an Application to Information Retrieval. *Advances in Neural Information Processing Systems*, **17**, 1481-1488.
- [17] Turk, M.A. and Pentland, A.P. (1991) Face Recognition Using Eigenfaces. *Proceedings of the Computer Vision and Pattern Recognition 1991*, Maui, 586-591. <http://dx.doi.org/10.1109/CVPR.1991.139758>
- [18] Huber, R., Ramoser, H., Mayer, K., Penz, H. and Rubik, M. (2005) Classification of Coins Using an Eigenspace Ap-

- proach. *Pattern Recognition Letters*, **26**, 61-75. <http://dx.doi.org/10.1016/j.patrec.2004.09.006>
- [19] Posadas, A.M., Vidal, F., de Miguel, F., Alguacil, G., Pena, J., Ibanez, J.M. and Morales, J. (1993) Spatialtemporal Analysis of a Seismic Series Using the Principal Components Method. *Journal of Geophysical Research*, **98**, 1923-1932. <http://dx.doi.org/10.1029/92JB02297>
- [20] Partridge, M. and Calvo, R. (1997) Fast Dimensionality Reduction and Simple PCA. *Intelligent Data Analysis*, **2**, 292-298.
- [21] Schölkopf, B., Smola, A. and Müller, K.R. (1998) Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, **10**, 1299-1319.
- [22] Shawe-Taylor, J. and Christianini, N. (2004) Kernel Methods for Pattern Analysis. Cambridge University Press, Cambridge.
- [23] Tipping, M.E. (2000) Sparse Kernel Principal Component Analysis. *Advances in Neural Information Processing Systems*, **13**, 633-639.
- [24] Kim, K.I., Jung, K. and Kim, H.J. (2002) Face Recognition Using Kernel Principal Component Analysis. *IEEE Signal Processing Letters*, **9**, 40-42. <http://dx.doi.org/10.1109/97.991133>
- [25] Hoffmann, H. (2007) Kernel PCA for Novelty Detection. *Pattern Recognition*, **40**, 863-874. <http://dx.doi.org/10.1016/j.patcog.2006.07.009>
- [26] Lima, A., Zen, H., Nankaku, Y., Miyajima, C., Tokuda, K. and Kitamura, T. (2004) On the Use of Kernel PCA for Feature Extraction in Speech Recognition. *IEICE Transactions on Information Systems*, **E87-D**, 2802-2811.
- [27] Duda, R.O., Hart, P.E. and Stork, D.G. (2001) Pattern Classification, Wiley Interscience, New York.
- [28] Shin, Y.J. and Park, C.H. (2011) Analysis of Correlation Based Dimension Reduction Methods. *International Journal of Applied Mathematics and Computer Science*, **21**, 549-558.
- [29] Fukunaga, K. (1990) Introduction to Statistical Pattern Recognition. 2nd Edition, Academic Press, San Diego.
- [30] Hotelling, H. (1936) Relations between Two Sets of Vertices. *Biometrika*, **28**, 321-377.
- [31] Sun, Q., Zeng, S., Liu, Y., Heng, P. and Xia, D. (2005) A New Method of Feature Fusion and Its Application in Image Recognition. *Pattern Recognition*, **38**, 2437-2448. <http://dx.doi.org/10.1016/j.patcog.2004.12.013>
- [32] Hastie, T. and Stuetzle, W. (1989) Principal Curves. *Journal of the American Statistical Association*, **84**, 502-516.
- [33] Kegl, B. and Linder, T. (2000) Learning and Design of Principal Curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**, 281-297.
- [34] Ozertem, U. and Erdogmus, D. (2011) Locally Defined Principal Curves and Surfaces. *Journal of Machine Learning Research*, **12**, 1249-1286.
- [35] Malthouse, E. (1996) Some Theoretical Results on Nonlinear Principal Component Analysis. citeseer.nj.net.com/malthouse96some.html
- [36] Carreira-Perpinan, M.A. (1997) A Review of Dimension Reduction Techniques. Technical Report CS-96-09. Department of Computer Science, University of Sheffield, Sheffield.
- [37] Tibshirani, R. (1992) Principal Curves Revisited. *Statistics and Computing*, **2**, 183-190. <http://dx.doi.org/10.1007/BF01889678>
- [38] Bishop, C.M. (1995) Neural Networks for Pattern Recognition. Oxford University Press, New York.
- [39] Ripley, B.D. (1996) Pattern Recognition and Neural Networks. Cambridge University Press, Cambridge.
- [40] Spierenburg, J.A. (1997) Dimension Reduction of Images Using Neural Networks. Master's Thesis, Leiden University, Leiden.
- [41] Kramer, M.A. (1991) Non-Linear Principal Component Analysis Using Associative Neural Networks. *AIChE Journal*, **37**, 233-243. <http://dx.doi.org/10.1002/aic.690370209>
- [42] Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T. (1992) Numerical Recipes in C: The Art of Scientific Computing. 2nd Edition, Cambridge University Press, Cambridge.
- [43] Fowlkes, C., Belongie, S., Chung, F. and Malik, J. (2004) Spectral Grouping Using the Nystrom Method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**, 214-225.
- [44] Marida, K.V., Kent, J.T. and Bibby, J.M. (1995) Multivariate Analysis. Probability and Mathematical Statistics. Academic Press, Waltham.
- [45] Faloutsos, C. and Lin, K.I. (1995) FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets. In: Carey, M.J. and Schneider, D.A., Eds., *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, San Jose, 163-174. <http://dx.doi.org/10.1145/223784.223812>

- [46] Fodor, I.K. (2002) A Survey of Dimension Reduction Techniques. Center for Applied Scientific Computing, Livermore National Laboratory, Livermore.
- [47] Chung, F.R.K. (1997) Spectral Graph Theory. American Mathematical Society. *CBMS Regional Conference Series in Mathematics in American Mathematical Society*, **212**, 92.
- [48] Belkin, M. and Niyogi, P. (2003) Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, **15**, 1373-1396. <http://dx.doi.org/10.1162/089976603321780317>
- [49] Rivest, R., Cormen, T., Leiserson, C. and Stein, C. (2001) Introduction to Algorithms. MIT Press, Cambridge.
- [50] Kumar, V., Grama, A., Gupta, A. and Karypis, G. (1994) Introduction to Parallel Computing. Benjamin-Cummings, Redwood City.
- [51] Donoho, D. and Grimes, C. Hessian Eigenmaps: Locally Linear Embedding Techniques for High-Dimensional Data. *Proceedings of National Academy of Sciences*, **100**.
- [52] Ye, Q. and Zhi, W.F. (2003) Discrete Hessian Eigenmaps Method for Dimensionality Reduction.
- [53] Kamhaltla, N. and Leen, T.K. (1994) Fast Non-Linear Dimension Reduction. In: *Advances in Neural Information Processing Systems*, Morgan Kaufmann Publishers, Inc., Burlington, 152-159.
- [54] Goldberg, D.E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading.
- [55] Raymer, M.L., Goodman, E.D., Kuhn, L.A. and Jain, A.K. (2000) Dimensionality Reduction Using Genetic Algorithms. *IEEE Transactions on Evolutionary Computation*, **4**, 164-171. <http://dx.doi.org/10.1109/4235.850656>
- [56] Jones, G. (2002) Published Online: 15 APR. University of Sheffield, Sheffield.
- [57] Kohavi, R. and John, G. (1998) The Wrapper Approach. In: Liu, H. and Motoda, H., Eds., *Feature Extraction, Construction and Selection: A Data Mining Perspective*, Springer Verlag, Berlin, 33-50. http://dx.doi.org/10.1007/978-1-4615-5725-8_3
- [58] Huber, P.J. (1985) Projection Pursuit. *Annals of Statistics*, **13**, 435-475. <http://dx.doi.org/10.1214/aos/1176349519>
- [59] McCullagh, P. and Nelder, J.A. (1989) Generalized Linear Models. Chapman and Hall, Boca Raton. <http://dx.doi.org/10.1007/978-1-4899-3242-6>
- [60] Dobson, A.J. (1990) An Introduction to Generalized Linear Models. Chapman and Hall, London. <http://dx.doi.org/10.1007/978-1-4899-7252-1>
- [61] Leathwick, J.R., Elith, J. and Hastie, T. (2006) Comparative Performance of Generalized Additive Models and Multivariate Adaptive Regression Splines for Statistical Modelling of Species Distributions. *Ecological Modelling*, 188-196. http://www.stanford.edu/~hastie/Papers/Ecology/leathwick_etal_2006_mars_ecolmod.pdf
- [62] Li, K.C. (2000) High Dimensional Data Analysis via SIR/PHD Approach. Lecture Note in Progress. <http://www.stat.ucla.edu/kcli/>
- [63] Dennis Cook, R. and Li, B. (2002) Dimension Reduction for Conditional Mean in Regression. *Annals of Statistics*, **30**, 455-474. <http://dx.doi.org/10.1214/aos/1021379861>
- [64] Zhang, Z. and Zha, H. (2002) Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment. <http://arxiv.org/pdf/cs.LG/0212008.pdf>
- [65] Wang, H. and Xia, Y. (2008) Sliced Regression for Dimension Reduction. Peking University & National University of Singapore, *Journal of the American Statistical Association*, **103**, 811-821.
- [66] Feng, W.K., He, X. and Shi, P. (2002) Dimension Reduction Based on Canonical Correlation. *Statistica Sinica*, **12**, 1093-1113.
- [67] Lectures on Fractals and Dimension Theory. <http://homepages.warwick.ac.uk/masdbl/dimensiontotal.pdf>

Appendix. Fractal and Topological Dimension

The main Reference for this appendix is [67]. Local (or topological) Methods (1): The definition of topological dimension was given by Brouwer in 1913: **A. Heyting, H. Freudenthal, Collected Works of L.E.J Brouwer, North Holland Elsevier, 1975.**

To begin at the very beginning: How can we best define the dimension of a closed bounded set $\Omega \subset \mathbb{R}^d$, say?

- When Ω is a manifold then the value of the dimension is an integer which coincides with the usual notion of dimension;
- For more general sets Ω we can have fractional dimensional
- Points, and countable unions of points, have zero dimension.

Local (or topological) Methods (2): The earliest attempt to define the dimension:

Definition 1 We can define the Topological dimension ($\dim_T \Omega$) by induction. We say that Ω has zero dimension if for every point $x \in \Omega$ every sufficiently small ball about x has boundary not intersecting Ω . We say that Ω has dimension d if for every point $x \in \Omega$ every sufficiently small ball about x has boundary intersecting Ω in a set of dimension $d-1$.

Local (or topological) Methods (3):

Definition 2 Given $\epsilon > 0$, let $N(\epsilon)$ be the smallest number of ϵ -balls needed to cover Ω . The Box dimension is

$$\dim_B \Omega := \limsup_{\epsilon \rightarrow 0} \frac{\log N(\epsilon)}{\log \left(\frac{1}{\epsilon} \right)}$$

Example 1 For $\Omega = \left\{ \frac{1}{n} : n \geq 1 \right\} \cup \{0\}$

$$\dim_B \Omega = \frac{1}{2}$$

Local (or topological) Methods (4): The Hausdorff dimension $\dim_H \Omega$ for a closed bounded set $\Omega \subset \mathbb{R}^d$ is defined as follows:

Definition 3 Consider a cover $\mathcal{U} = \{U_i\}$ for Ω by open sets. For $\delta > 0$ we can define

$$H_\epsilon^\delta(\Omega) = \inf_{\mathcal{U}} \left\{ \sum_i \text{diam}(U_i)^\delta \right\}$$

where the infimum is taken over all open covers $\mathcal{U} = \{U_i\}$ such that $\text{diam}(U_i) \leq \epsilon$. Then


$H^\delta(\Omega) = \lim_{\epsilon \rightarrow 0} H_\epsilon^\delta(\Omega)$ and finally,

$$\dim_H \Omega := \inf \left\{ \delta : H^\delta(\Omega) = 0 \right\}$$

- Fact1: For any countable set Ω we have $\dim_H \Omega = 0$
- Fact2: $\dim_H \Omega \leq \dim_B \Omega$

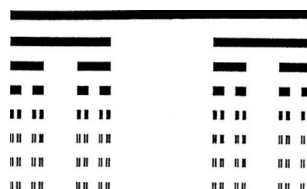
Local (or topological) Methods (4) as shown in **Figure 1**.

Local (or topological) Methods (5):

Example 2 (von Koch curve: ) The von Koch curve is a standard fractal construction. Starting from $\Omega_0 = [0, 1]$, we associate to each piecewise linear curve Ω_n in the plane (which is a union of 4^n segments of length 3^{-n}) a new one Ω_{n+1} . This is done by replacing the middle third of each line segment by the other two sides of an equilateral triangle bases there. Alternatively, one can start from an equilateral triangle and apply this iterative procedure to each of the sides one gets a snowflake curve.

For $\Omega =$ von Koch curve, both the box dimension and Hausdorff dimension are equal in fact, as shown in **Figure 2**:

$$\dim_H \left(\text{von Koch curve} \right) = \dim_B \left(\text{von Koch curve} \right) = \frac{\log 4}{\log 3}$$

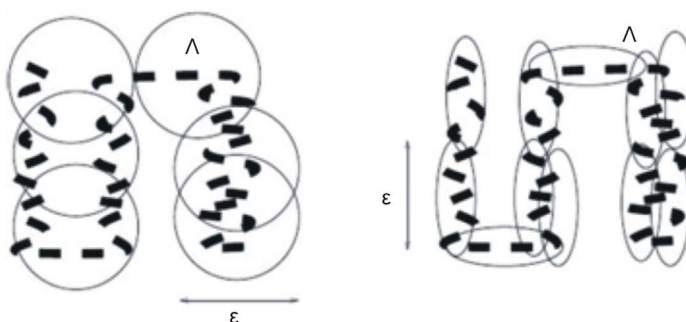


Example 3 (Ω : the Middle third Cantor set E_2 , This is the set of closed set of points in the unit interval whose triadic expansion does not contain any occurrence of the digit 1 :

$$\Omega := \left\{ \sum_{k=1}^{\infty} \frac{i_k}{3^k} : i_k \in \{0, 2\} \right\}$$

For the middle third Cantor set both the Box dimension and the Hausdorff dimension are $\frac{\log 2}{\log 3} = 0.690\dots$

The set E_2 is the set of points whose continued fraction expansion contains only the terms 1 and 2. Unlike the Middle third Cantor set, the dimension of this set is not explicitly known in a closed form and can only be numerically estimated to the desired level of accuracy. as shown in [Figure 3](#), For the Sierpinski carpet both the Box dimension and the Hausdorff dimension are equal to $\frac{\log 8}{\log 3} = 1.892\dots$



- (I) COVER BY BALLS (FOR $\text{DIM}_B(\Lambda)$;
(II) COVER BY OPEN SETS (FOR $\text{DIM}_H(\Lambda)$)

Figure 1. (I) Cocer by balls, (II) Cover by open sets.

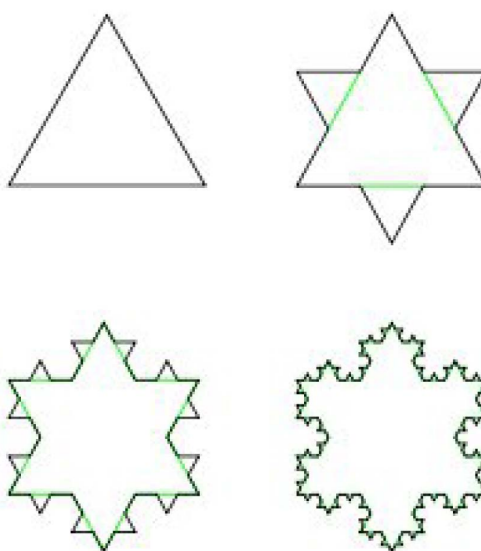


Figure 2. The construction of von Koch-curve.

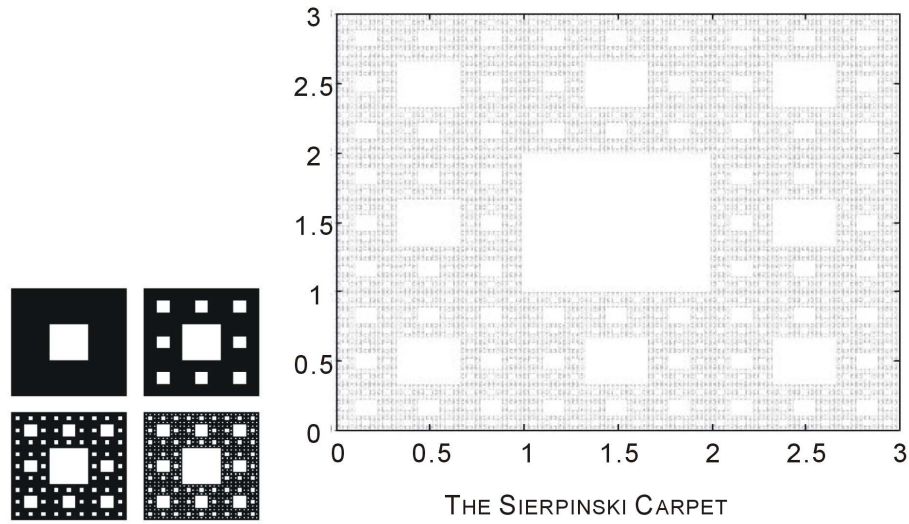


Figure 3. The construction Sierpinski Carpet.

High Accurate Fourth-Order Finite Difference Solutions of the Three Dimensional Poisson's Equation in Cylindrical Coordinate

Alemayehu Shiferaw, Ramesh Chand Mittal

Department of Mathematics, Indian Institute of Technology, Roorkee, India
Email: abelhaim@gmail.com, mittalrc@gmail.com

Received 5 November 2013; revised 5 December 2013; accepted 16 December 2013

Copyright © 2014 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In this work, by extending the method of Hockney into three dimensions, the Poisson's equation in cylindrical coordinates system with the Dirichlet's boundary conditions in a portion of a cylinder for $r \neq 0$ is solved directly. The Poisson equation is approximated by fourth-order finite differences and the resulting large algebraic system of linear equations is treated systematically in order to get a block tri-diagonal system. The accuracy of this method is tested for some Poisson's equations with known analytical solutions and the numerical results obtained show that the method produces accurate results.

Keywords

Poisson's Equation; Tri-Diagonal Matrix; Fourth-Order Finite Difference Approximation; Hockney's Method; Thomas Algorithm

1. Introduction

The three-dimensional Poisson's equation in cylindrical coordinates (r, θ, z) is given by

$$U_{rr} + \frac{1}{r}U_r + \frac{1}{r^2}U_{\theta\theta} + U_{zz} = f(r, \theta, z) \quad (1)$$

has a wide range of application in engineering and science fields (especially in physics).

How to cite this paper: Shiferaw, A. and Mittal, R.C. (2014) High Accurate Fourth-Order Finite Difference Solutions of the Three Dimensional Poisson's Equation in Cylindrical Coordinate. *American Journal of Computational Mathematics*, 4, 73-86.
<http://dx.doi.org/10.4236/ajcm.2014.42007>

In physical problems that involve a cylindrical surface (for example, the problem of evaluating the temperature in a cylindrical rod), it will be convenient to make use of cylindrical coordinates. For the numerical solution of the three dimensional Poisson's equation in cylindrical coordinates system, several attempts have been made in particular for physical problems that are related directly or indirectly to this equation. For instance, *Lai* [1] developed a simple compact fourth-order Poisson solver on polar geometry based on the truncated Fourier series expansion, where the differential equations of the Fourier coefficients are solved by the compact fourth-order finite difference scheme; *Mittal and Gahlaut* [2] have developed high order finite difference schemes of second- and fourth- order in polar coordinates using a direct method similar to Hockney's method; *Mittal and Gahlaut* [3] developed a second- and fourth-order finite difference scheme to solve Poisson's equation in the case of cylindrical symmetry; *Alemayehu and Mittal* [4] have derived a second-order finite difference approximation scheme to solve the three dimensional Poisson's equation in cylindrical coordinates by extending Hockney's method; *Tan* [5] developed a spectrally accurate solution for the three dimensional Poisson's equation and Helmholtz's equation using Chebyshev series and Fourier series for a simple domain in a cylindrical coordinate system; *Iyengar and Manohar* [6] derived fourth-order difference schemes for the solution of the Poisson equation which occurs in problems of heat transfer; *Iyengar and Goyal* [7] developed a multigrid method in cylindrical coordinates system; *Lai and Tseng* [8] have developed a fourth-order compact scheme, and their scheme relies on the truncated Fourier series expansion, where the partial differential equations of Fourier coefficients are solved by a formally fourth-order accurate compact difference discretization. The need to obtain the best solution for the three dimensional Poisson's equation in cylindrical coordinates system is still in progress.

In this paper, we develop a fourth-order finite difference approximation scheme and solve the resulting large algebraic system of linear equations systematically using block tridiagonal system [9] [10] and extend the Hockney's method [9] [11] to solve the three dimensional Poisson's equation on Cylindrical coordinates system.

2. Finite Difference Approximation

Consider the three dimensional Poisson's equation in cylindrical coordinates (r, θ, z) given by

$$\frac{\partial^2 U}{\partial r^2} + \frac{1}{r} \frac{\partial U}{\partial r} + \frac{1}{r^2} \frac{\partial^2 U}{\partial \theta^2} + \frac{\partial^2 U}{\partial z^2} = f(r, \theta, z) \text{ on } D$$

and the boundary condition

$$U(r, \theta, z) = g(r, \theta, z) \text{ on } C \quad (2)$$

where C is the boundary of D and D is

$$D_1 = \{(r, \theta, z) : R_0 < r < R_1, a < z < b, \theta_0 < \theta < \theta_1, \theta_0 < \theta_1 < 2\pi\} \text{ and } D_2 = \{(r, \theta, z) : R_0 < r < R_1, a < z < b, 0 \leq \theta < 2\pi\}$$

Consider **Figure 1** as the geometry of the problem. Let $u(r, \theta, z)$ be discretized at the point (r_i, θ_j, z_k) and for simplicity write a point (r_i, θ_j, z_k) as (i, j, k) and $u(r_i, \theta_j, z_k)$ as $u_{i,j,k}$.

Assume that there are M points in the direction of r , N points in θ and P points in the z directions to form the mesh, and let the step size along the direction of r be Δr , of θ be $\Delta \theta$ and z be Δz .

Here $r_i = R_0 + i\Delta r$, $\theta_j = \theta_0 + j\Delta \theta$ and $z_k = a + k\Delta z$

Where $i = 1, 2, \dots, M$, $j = 1, 2, \dots, N$ and $k = 1, 2, \dots, P$.

When $r = 0$ is an interior or a boundary point of (2), then the Poisson's equation becomes singular and to take care of the singularity a different approach will be taken. Thus in this paper we consider only for the case $r \neq 0$.

Using the approximations that

$$\left(\frac{\partial^2 U}{\partial r^2} \right)_{i,j,k} = \frac{1}{(\Delta r)^2} \left(1 + \frac{1}{12} \delta_r^2 \right)^{-1} \delta_r^2 U_{i,j,k} + O((\Delta r)^4) \quad (3)$$

$$\left(\frac{\partial^2 U}{\partial \theta^2} \right)_{i,j,k} = \frac{1}{(\Delta \theta)^2} \left(1 + \frac{1}{12} \delta_\theta^2 \right)^{-1} \delta_\theta^2 U_{i,j,k} + O((\Delta \theta)^4) \quad (4)$$

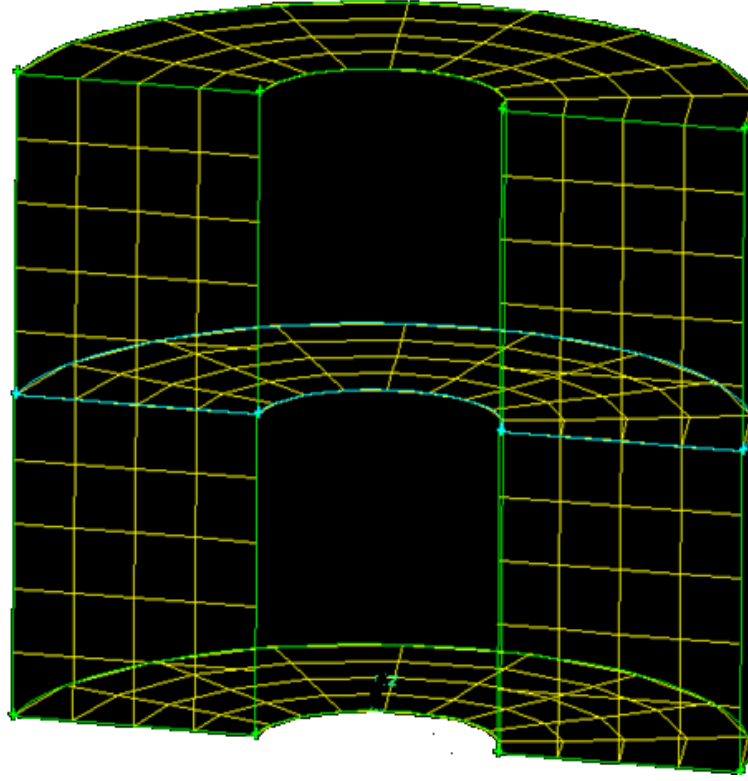


Figure 1. Portion of a cylinder.

$$\left(\frac{\partial^2 U}{\partial z^2} \right)_{i,j,k} = \frac{1}{(\Delta z)^2} \left(1 + \frac{1}{12} \delta_z^2 \right)^{-1} \delta_z^2 U_{i,j,k} + O((\Delta z)^4) \quad (5)$$

Now using (3), (4) and (5), we get (Refer the work of Mittal and Ghalaut in [2])

From (1) consider only the approximation of the sum of the first and the third terms, that is, the sum of $\frac{\partial^2 U}{\partial r^2}$ and $\frac{1}{r_i^2} \frac{\partial^2 U}{\partial \theta^2}$

$$\begin{aligned} & \left(\frac{\partial^2 U}{\partial r^2} + \frac{1}{r_i^2} \frac{\partial^2 U}{\partial \theta^2} \right)_{i,j,k} \\ &= \frac{1}{12(\Delta r)^2} \left[-20 \left(1 + \frac{\omega}{r_i^2} \right) U_{i,j,k} + 2 \left(5 - \frac{\omega}{r_i^2} \right) (U_{i+1,j,k} + U_{i-1,j,k}) + 2 \left(\frac{5\omega}{r_i^2} - 1 \right) (U_{i,j+1,k} + U_{i,j-1,k}) \right. \\ & \quad \left. + \left(1 + \frac{\omega}{r_i^2} \right) (U_{i+1,j+1,k} + U_{i+1,j-1,k} + U_{i-1,j+1,k} + U_{i-1,j-1,k}) \right] - \frac{1}{12} \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} \right) \\ & \quad \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} \right) U_{i,j,k} + O((\Delta r)^4 + (\Delta \theta)^4) \end{aligned} \quad (6)$$

where $\omega = \frac{(\Delta r)^2}{(\Delta \theta)^2}$

Again from (1) consider only the approximation of the sum of the first and the fourth terms, that is, the sum of $\frac{\partial^2 U}{\partial r^2}$ and $\frac{\partial^2 U}{\partial z^2}$, and we get

$$\begin{aligned}
& \left(\frac{\partial^2 U}{\partial r^2} + \frac{\partial^2 U}{\partial z^2} \right)_{i,j,k} \\
&= \frac{1}{12(\Delta r)^2} \left[\left(1 + \frac{(\Delta r)^2}{(\Delta z)^2} \right) (U_{i+1,j,k+1} + U_{i+1,j,k-1} + U_{i-1,j,k+1} + U_{i-1,j,k-1}) \right. \\
&\quad \left. + 2 \left(5 - \frac{(\Delta r)^2}{(\Delta z)^2} \right) (U_{i+1,j,k} + U_{i-1,j,k}) + 2 \left(5 \frac{(\Delta r)^2}{(\Delta z)^2} - 1 \right) (U_{i,j+1,k} + U_{i,j-1,k}) - 20 \left(1 + \frac{(\Delta r)^2}{(\Delta z)^2} \right) U_{i,j,k} \right] \\
&\quad - \frac{1}{12} \left(\frac{\partial^2}{\partial r^2} + \frac{\partial^2}{\partial z^2} \right) \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} + O((\Delta r)^4 + (\Delta z)^4)
\end{aligned} \tag{7}$$

Once again from (1) consider only the approximation of the sum of the second and the fourth terms, that is, the sum of $\frac{1}{r_i^2} \frac{\partial^2 U}{\partial \theta^2}$ and $\frac{\partial^2 U}{\partial z^2}$; to get

$$\begin{aligned}
& \left(\frac{1}{r_i^2} \frac{\partial^2 U}{\partial \theta^2} + \frac{\partial^2 U}{\partial z^2} \right)_{i,j,k} \\
&= \frac{1}{12} \left[\left(\frac{1}{(r_i \Delta \theta)^2} + \frac{1}{(\Delta z)^2} \right) (U_{i,j+1,k+1} + U_{i,j+1,k-1} + U_{i,j-1,k+1} + U_{i,j-1,k-1}) + 2 \left(\frac{5}{(r_i \Delta \theta)^2} - \frac{1}{(\Delta z)^2} \right) (U_{i,j+1,k} + U_{i,j-1,k}) \right. \\
&\quad \left. + 2 \left(\frac{5}{(\Delta z)^2} - \frac{1}{(r_i \Delta \theta)^2} \right) (U_{i,j,k+1} + U_{i,j,k-1}) - 20 \left(\frac{1}{(r_i \Delta \theta)^2} + \frac{1}{(\Delta z)^2} \right) U_{i,j,k} \right] \\
&\quad - \frac{1}{12} \left(\frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial z^2} \right) \left((\Delta \theta)^2 \frac{\partial^2}{\partial r^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} + O((\Delta \theta)^4 + (\Delta z)^4)
\end{aligned} \tag{8}$$

Again taking the approximation of the term $\frac{\partial U}{\partial r}$ by

$$\begin{aligned}
\left(\frac{\partial U}{\partial r} \right)_{i,j,k} &= \frac{\phi \delta_{2r} (U_{i,j+1,k} + U_{i,j-1,k} + U_{i,j,k+1} + U_{i,j,k-1}) + (1-4\phi) \delta_{2r} U_{i,j,k}}{2\Delta r} \\
&\quad - \frac{1}{3} (\Delta r)^2 \frac{\partial^3 U_{i,j,k}}{\partial r^3} - \phi (\Delta \theta)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial \theta^2} - \phi (\Delta z)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial z^2} \\
&\quad + O((\Delta r)^4 + (\Delta \theta)^4 + (\Delta z)^4), 0 \leq \phi \leq 1
\end{aligned} \tag{9}$$

Equation (9) implying that

$$\begin{aligned}
\frac{1}{r_i} \left(\frac{\partial U}{\partial r} \right)_{i,j,k} &= \frac{\phi \delta_{2r} (U_{i,j+1,k} + U_{i,j-1,k} + U_{i,j,k+1} + U_{i,j,k-1}) + (1-4\phi) \delta_{2r} U_{i,j,k}}{2r_i \Delta r} \\
&\quad - \frac{1}{3r_i} (\Delta r)^2 \frac{\partial^3 U_{i,j,k}}{\partial r^3} - \phi (\Delta \theta)^2 \frac{1}{r_i} \frac{\partial^3 U_{i,j,k}}{\partial r \partial \theta^2} \\
&\quad - \phi (\Delta z)^2 \frac{1}{r_i} \frac{\partial^3 U_{i,j,k}}{\partial r \partial z^2} + O((\Delta r)^4 + (\Delta \theta)^4 + (\Delta z)^4)
\end{aligned} \tag{10}$$

Now letting $\alpha = \frac{(\Delta r)^2}{(\Delta z)^2}$ and adding (6), (7), (8) and twice of (10), we get

$$\begin{aligned}
& 2 \left(\frac{\partial^2 U}{\partial r^2} + \frac{1}{r_i} \frac{\partial U}{\partial r} + \frac{1}{r_i^2} \frac{\partial^2 U}{\partial \theta^2} + \frac{\partial^2 U}{\partial z^2} \right)_{i,j,k} \\
&= \frac{1}{12(\Delta r)^2} \left[\left(1 + \frac{\omega}{r_i^2} \right) (U_{i+1,j+1,k} + U_{i+1,j-1,k} + U_{i-1,j+1,k} + U_{i-1,j-1,k}) + 2 \left(5 - \frac{\omega}{r_i^2} \right) (U_{i+1,j,k} + U_{i-1,j,k}) \right. \\
&\quad + 2 \left(\frac{5\omega}{r_i^2} - 1 \right) (U_{i,j+1,k} + U_{i,j-1,k}) + (1+\alpha) (U_{i+1,j,k+1} + U_{i+1,j,k-1} + U_{i-1,j,k+1} + U_{i-1,j,k-1}) \\
&\quad \left. + 2(5-\alpha) (U_{i+1,j,k} + U_{i-1,j,k}) + 2(5\alpha-1) (U_{i,j+1,k} + U_{i,j-1,k}) - 20 \left(2 + \alpha + \frac{\omega}{r_i^2} \right) U_{i,j,k} \right] \\
&\quad + \frac{1}{12} \left[\left(\frac{1}{(r_i \Delta \theta)^2} + \frac{1}{(\Delta z)^2} \right) (U_{i,j+1,k+1} + U_{i,j+1,k-1} + U_{i,j-1,k+1} + U_{i,j-1,k-1}) - 20 \left(\frac{1}{(r_i \Delta \theta)^2} + \frac{1}{(\Delta z)^2} \right) U_{i,j,k} \right] \quad (11) \\
&\quad + 2 \left(\frac{5}{(r_i \Delta \theta)^2} - \frac{1}{(\Delta z)^2} \right) (U_{i,j+1,k} + U_{i,j-1,k}) + 2 \left(\frac{5}{(\Delta z)^2} - \frac{1}{(r_i \Delta \theta)^2} \right) (U_{i,j,k+1} + U_{i,j,k-1}) \\
&\quad - \frac{1}{12} \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} \right) \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} \right) U_{i,j,k} - \frac{1}{12} \left(\frac{\partial^2}{\partial r^2} + \frac{\partial^2}{\partial z^2} \right) \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} \\
&\quad - \frac{1}{12} \left(\frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial z^2} \right) \left((\Delta \theta)^2 \frac{\partial^2}{\partial r^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} + \frac{\phi \delta_{2r} (U_{i,j+1,k} + U_{i,j-1,k} + U_{i,j,k+1} + U_{i,j,k-1})}{\Delta r} \\
&\quad + \frac{(1-4\phi) \delta_{2r} U_{i,j,k}}{\Delta r} - \frac{1}{3} (\Delta r)^2 \frac{\partial^3 U_{i,j,k}}{\partial r^3} - \phi (\Delta \theta)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial \theta^2} - \phi (\Delta z)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial z^2} + O((\Delta r)^4 + (\Delta \theta)^4 + (\Delta z)^4)
\end{aligned}$$

Now choose $\phi = \frac{1}{12}$ and consider the following terms in (11)

$$\begin{aligned}
& -\frac{1}{12} \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} \right) \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} \right) U_{i,j,k} \\
& -\frac{1}{12} \left(\frac{\partial^2}{\partial r^2} + \frac{\partial^2}{\partial z^2} \right) \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} - \frac{1}{3r_i} (\Delta r)^2 \frac{\partial^3 U_{i,j,k}}{\partial r^3} \\
& -\frac{1}{12} \left(\frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial z^2} \right) \left((\Delta \theta)^2 \frac{\partial^2}{\partial r^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} - \frac{1}{12r_i} \left((\Delta \theta)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial \theta^2} + (\Delta z)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial z^2} \right) \\
& = -\frac{1}{3r_i} (\Delta r)^2 \frac{\partial^3 U}{\partial r^3} - \frac{1}{12} \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} \\
& \quad - \frac{1}{12} \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} \left(\frac{\partial^2}{\partial r^2} \right) + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} \left(\frac{\partial^2}{\partial \theta^2} \right) + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \left(\frac{\partial^2}{\partial z^2} \right) \right) U_{i,j,k} \quad (12) \\
& \quad - \frac{1}{12r_i} \left((\Delta \theta)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial \theta^2} + (\Delta z)^2 \frac{\partial^3 U_{i,j,k}}{\partial r \partial z^2} \right) \\
& = -\frac{1}{12} \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} \\
& \quad - \frac{1}{12} \left(\frac{1}{r_i} \frac{\partial}{\partial r} \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} \right) - \frac{1}{4r_i} (\Delta r)^2 \frac{\partial^3 U_{i,j,k}}{\partial r^3} \\
& = -\frac{1}{12} \left((\Delta r)^2 \frac{\partial^2}{\partial r^2} + (\Delta \theta)^2 \frac{\partial^2}{\partial \theta^2} + (\Delta z)^2 \frac{\partial^2}{\partial z^2} \right) \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r_i} \frac{\partial}{\partial r} + \frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} - \frac{1}{4r_i} (\Delta r)^2 \frac{\partial^3 U_{i,j,k}}{\partial r^3}
\end{aligned}$$

Again we can write the term $-\frac{1}{4r_i}(\Delta r)^2 \frac{\partial^3 U_{i,j,k}}{\partial r^3}$ in (12) as

$$\begin{aligned}
 & -\frac{(\Delta r)^2}{4r_i} \frac{\partial^3 U_{i,j,k}}{\partial r^3} \\
 & = -\frac{(\Delta r)^2}{4r_i} \frac{\partial}{\partial r} \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r_i} \frac{\partial}{\partial r} + \frac{1}{r_i^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial z^2} \right) U_{i,j,k} \\
 & \quad + \frac{(\Delta r)^2}{4r_i} \frac{\partial}{\partial r} \left(\frac{1}{r_i} \frac{\partial U_{i,j,k}}{\partial r} \right) + \frac{(\Delta r)^2}{4r_i} \frac{\partial}{\partial r} \left(\frac{1}{r_i^2} \frac{\partial^2 U_{i,j,k}}{\partial \theta^2} \right) + \frac{(\Delta r)^2}{4r_i} \frac{\partial}{\partial r} \left(\frac{\partial^2 U_{i,j,k}}{\partial z^2} \right) \\
 & = -\frac{(\Delta r)^2}{4r_i} \frac{\partial f}{\partial r} - \frac{(\Delta r)^2}{4r_i^3} \frac{\partial U_{i,j,k}}{\partial r} + \frac{(\Delta r)^2}{4r_i^2} \frac{\partial^2 U_{i,j,k}}{\partial r^2} - \frac{1}{2} \frac{(\Delta r)^2}{r_i^4} \frac{\partial^2 U_{i,j,k}}{\partial \theta^2} + \frac{(\Delta r)^2}{4r_i^3} \frac{\partial}{\partial r} \left(\frac{\partial^2 U_{i,j,k}}{\partial \theta^2} \right) \\
 & \quad + \frac{(\Delta r)^2}{4r_i} \frac{\partial}{\partial r} \left(\frac{\partial^2 U_{i,j,k}}{\partial z^2} \right)
 \end{aligned} \tag{13}$$

Using (12), (13), and multiplying both sides of (11) by $12(\Delta r)^2$ and rearranging and simplifying further, we get

$$\begin{aligned}
 & (\Delta r)^2 \left(24 + \delta_r^2 + \delta_\theta^2 + \delta_z^2 + \frac{3\Delta r}{2r_i} \delta_{2r} \right) f_{i,j,k} \\
 & = a_0(i) U_{i,j,k} + a_1(i) U_{i+1,j,k} + a_2(i) U_{i-1,j,k} \\
 & \quad + a_3(i) (U_{i,j+1,k} + U_{i,j-1,k}) + a_4(i) (U_{i,j,k+1} + U_{i,j,k-1}) + a_5(i) (U_{i+1,j+1,k} + U_{i+1,j-1,k}) \\
 & \quad + a_6(i) (U_{i-1,j+1,k} + U_{i-1,j-1,k}) + a_7(i) (U_{i+1,j,k+1} + U_{i+1,j,k-1}) + a_8(i) (U_{i-1,j,k+1} + U_{i-1,j,k-1}) \\
 & \quad + a_9(i) (U_{i,j+1,k+1} + U_{i,j-1,k+1} + U_{i,j+1,k-1} + U_{i,j-1,k-1})
 \end{aligned} \tag{14}$$

where

$$\begin{aligned}
 a_0(i) &= -40 \left(1 + \alpha + \frac{\omega}{r_i^2} \right) - 6 \frac{(\Delta r)^2}{r_i^2} + 12 \frac{\omega}{r_i^2} \frac{(\Delta r)^2}{r_i^2} \\
 a_1(i) &= 20 - 2\alpha - \frac{2\omega}{r_i^2} + 8 \frac{\Delta r}{r_i} - \frac{3}{2} \left(\frac{\Delta r}{r_i} \right)^3 + 3 \left(\frac{\Delta r}{r_i} \right)^2 - 3 \frac{\omega}{r_i^2} \frac{\Delta r}{r_i} - 3\alpha \frac{\Delta r}{r_i} \\
 a_2(i) &= 20 - 2\alpha - \frac{2\omega}{r_i^2} - 8 \frac{\Delta r}{r_i} + \frac{3}{2} \left(\frac{\Delta r}{r_i} \right)^3 + 3 \left(\frac{\Delta r}{r_i} \right)^2 + 3 \frac{\omega}{r_i^2} \frac{\Delta r}{r_i} + 3\alpha \frac{\Delta r}{r_i} \\
 a_3(i) &= -2\alpha + 12 \frac{\omega}{r_i^2} - 2 \quad a_4(i) = 20\alpha - \frac{2\omega}{r_i^2} - 2 \\
 a_5(i) &= 1 + \frac{\omega}{r_i^2} + \frac{\Delta r}{r_i} + \frac{3}{2} \frac{\omega}{r_i^2} \frac{\Delta r}{r_i} \quad a_6(i) = 1 + \frac{\omega}{r_i^2} - \frac{\Delta r}{r_i} - \frac{3}{2} \frac{\omega}{r_i^2} \frac{\Delta r}{r_i} \\
 a_7(i) &= 1 + \alpha + \frac{\Delta r}{r_i} + \frac{3}{2} \alpha \frac{\Delta r}{r_i} \quad a_8(i) = 1 + \alpha - \frac{\Delta r}{r_i} - \frac{3}{2} \alpha \frac{\Delta r}{r_i} \quad a_9(i) = \alpha + \frac{\omega}{r_i^2}
 \end{aligned}$$

The system of equations in (14) is a linear sparse system, and thereby when solving we save both work and storage compared with a general system of equations. Such savings are basically true of finite difference methods: they yield sparse systems because each equation involves only few variables.

To solve equation (14), consider first in the θ direction, next in the z direction and lastly in the r direction, and thus (14) can be written in matrix form as

$$AU = \mathcal{B} \quad (15)$$

where

$$A = \begin{pmatrix} R_1 & S_1 & & & \\ T_2 & R_2 & S_2 & & \\ & T_3 & R_3 & S_3 & \\ & & & \ddots & \\ & & & & T_{M-1} & R_{M-1} & S_{M-1} \\ & & & & & T_M & R_M \end{pmatrix}$$

and it has M blocks and each is of order NP .

$$R_i = \begin{pmatrix} R'_i & R''_i & & & \\ R''_i & R'_i & R''_i & & \\ & R''_i & R'_i & R''_i & \\ & & \ddots & & \\ & & & R''_i & R'_i & R''_i \\ & & & & R''_i & R'_i \end{pmatrix}, \quad S_i = \begin{pmatrix} S'_i & S''_i & & & \\ S''_i & S'_i & S''_i & & \\ & S''_i & S'_i & S''_i & \\ & & \ddots & & \\ & & & S''_i & S'_i & S''_i \\ & & & & S''_i & S'_i \end{pmatrix}$$

$$T_i = \begin{pmatrix} T'_i & T''_i & & & \\ T''_i & T'_i & T''_i & & \\ & T''_i & T'_i & T''_i & \\ & & \ddots & & \\ & & & T''_i & T'_i & T''_i \\ & & & & T''_i & T'_i \end{pmatrix}$$

R_i, S_i , and T_i are of order NP .

For the domain D_1

$$R'_i = \begin{pmatrix} a_0(i) & a_3(i) & & & \\ a_3(i) & a_0(i) & a_3(i) & & \\ & a_3(i) & a_0(i) & a_3(i) & \\ & & \ddots & & \\ & & & a_3(i) & a_0(i) & a_3(i) \\ & & & & a_3(i) & a_0(i) \end{pmatrix}$$

$$R''_i = \begin{pmatrix} a_4(i) & a_9(i) & & & \\ a_9(i) & a_4(i) & a_9(i) & & \\ & a_9(i) & a_4(i) & a_9(i) & \\ & & \ddots & & \\ & & & a_9(i) & a_4(i) & a_9(i) \\ & & & & a_9(i) & a_4(i) \end{pmatrix}$$

$$S'_i = \begin{pmatrix} a_1(i) & a_5(i) & & & \\ a_5(i) & a_1(i) & a_5(i) & & \\ & a_5(i) & a_1(i) & a_5(i) & \\ & & \ddots & & \\ & & & a_5(i) & a_1(i) & a_5(i) \\ & & & & a_5(i) & a_1(i) \end{pmatrix}$$

$$S_i'' = \begin{pmatrix} a_7(i) & & & & \\ & a_7(i) & & & \\ & & a_7(i) & & \\ & & & \ddots & \\ & & & & a_7(i) \end{pmatrix}, \quad T_i' = \begin{pmatrix} a_2(i) & a_6(i) & & & \\ a_6(i) & a_2(i) & a_6(i) & & \\ & a_6(i) & a_2(i) & a_6(i) & \\ & & & \ddots & \\ & & & & a_6(i) & a_2(i) & a_6(i) \\ & & & & & a_6(i) & a_2(i) \end{pmatrix}$$

$$T_i'' = \begin{pmatrix} a_8(i) & & & & \\ & a_8(i) & & & \\ & & a_8(i) & & \\ & & & \ddots & \\ & & & & a_8(i) \end{pmatrix}$$

For the domain D_2 ,

$$R_i' = \begin{pmatrix} a_0(i) & a_3(i) & & & & a_3(i) \\ a_3(i) & a_0(i) & a_3(i) & & & \\ & a_3(i) & a_0(i) & a_3(i) & & \\ & & & \ddots & & \\ & & & & a_3(i) & a_0(i) & a_3(i) \\ a_3(i) & & & & & a_3(i) & a_0(i) \end{pmatrix}$$

$$R_i'' = \begin{pmatrix} a_4(i) & a_9(i) & & & & a_9(i) \\ a_9(i) & a_4(i) & a_9(i) & & & \\ & a_9(i) & a_4(i) & a_9(i) & & \\ & & & \ddots & & \\ & & & & a_9(i) & a_4(i) & a_9(i) \\ a_9(i) & & & & & a_9(i) & a_4(i) \end{pmatrix}$$

$$S_i' = \begin{pmatrix} a_1(i) & a_5(i) & & & & a_5(i) \\ a_5(i) & a_1(i) & a_5(i) & & & \\ & a_5(i) & a_1(i) & a_5(i) & & \\ & & & \ddots & & \\ & & & & a_5(i) & a_1(i) & a_5(i) \\ a_5(i) & & & & & a_5(i) & a_1(i) \end{pmatrix}$$

$$T_i' = \begin{pmatrix} a_2(i) & a_6(i) & & & & a_6(i) \\ a_6(i) & a_2(i) & a_6(i) & & & \\ & a_6(i) & a_2(i) & a_6(i) & & \\ & & & \ddots & & \\ & & & & a_6(i) & a_2(i) & a_6(i) \\ a_6(i) & & & & & a_6(i) & a_2(i) \end{pmatrix}$$

S_i'' and T_i'' are the same as in the domain D_1 .

Here in D_2 , the matrices $R_i', R_i'', S_i', S_i'', T_i',$ and T_i'' are circulant matrices of order N ; and

$$\mathcal{B} = [\mathcal{B}_0 \quad \mathcal{B}_1 \quad \mathcal{B}_2 \quad \cdots \quad \mathcal{B}_M]^T, \quad \mathcal{B}_i = [d_{i1} \quad d_{i2} \quad d_{i3} \quad \cdots \quad d_{iP}]^T \quad \text{and} \quad d_{ik} = [d_{ij1} \quad d_{ij2} \quad \cdots \quad d_{ijP}]^T$$

such that each d_{ijk} represents a known boundary values of U and values of f , and

$$\mathbf{U} = [\mathbf{U}_1 \quad \mathbf{U}_2 \quad \mathbf{U}_3 \quad \cdots \quad \mathbf{U}_M]^T, \quad \mathbf{U}_i = (U_{i1} \quad U_{i2} \quad U_{i3} \quad \cdots \quad U_{iP})^T \quad \text{and} \quad U_{ij} = (U_{ij1} \quad U_{ij2} \quad U_{ij3} \quad \cdots \quad U_{ijP})^T$$

Thus, we write (15) as

$$\begin{aligned}
 R_1 U_1 + S_1 U_2 &= B_1 \\
 T_2 U_1 + R_2 U_2 + S_2 U_3 &= B_2 \\
 T_3 U_2 + R_3 U_3 + S_3 U_4 &= B_3 \\
 &\dots \\
 T_M U_{M-1} + R_M U_M &= B_M
 \end{aligned} \tag{16}$$

3. Extended Hockney's Method

Observe that matrices R'_i, R''_i, S'_i and T'_i are real symmetric matrices and hence their eigenvalues and eigenvectors can easily be obtained as

For D_1

$$\begin{aligned}
 \lambda_{ij} &= a_0(i) + 2a_3(i) \cos\left(\frac{j\pi}{N+1}\right), \quad \beta_{ij} = a_4(i) + 2a_9(i) \cos\left(\frac{j\pi}{N+1}\right), \quad \eta_{ij} = a_1(i) + 2a_5(i) \cos\left(\frac{j\pi}{N+1}\right) \\
 \zeta_{ij} &= a_2(i) + 2a_6(i) \cos\left(\frac{j\pi}{N+1}\right), \quad i = 1(1)M \text{ and } j = 1(1)N
 \end{aligned}$$

and for D_2

$$\begin{aligned}
 \lambda_{ij} &= a_0(i) + 2a_3(i) \cos\left(\frac{2\pi j}{N}\right), \quad \beta_{ij} = a_4(i) + 2a_9(i) \cos\left(\frac{2\pi j}{N}\right), \quad \eta_{ij} = a_1(i) + 2a_5(i) \cos\left(\frac{2\pi j}{N}\right) \\
 \zeta_{ij} &= a_2(i) + 2a_6(i) \cos\left(\frac{2\pi j}{N}\right), \quad i = 1(1)M \text{ and } j = 1(1)N
 \end{aligned}$$

Let q_j be an eigenvector of R'_i, R''_i, S'_i and T'_i corresponding to the eigenvalue $\lambda_{ij}, \beta_{ij}, \eta_{ij}$ and ζ_{ij} ; and matrix $Q = [q_1 \ q_2 \ q_3 \ \dots \ q_N]^T$ be a modal matrix of R'_i, R''_i, S'_i and T'_i , $\forall i$ such that $Q^T Q = I$

The $N \times N$ modal matrix Q is defined by

$$q_{ij} = \sqrt{\frac{2}{N+1}} \sin\left(\frac{ij\pi}{N+1}\right), \quad i, j = 1(1)N \text{ for } D_1; \quad q_{ij} = \left(\frac{\cos\theta + \sin\theta}{\sqrt{N}}\right) \text{ where } \theta = \frac{2\pi}{N}(i-1)(j-1)$$

$i, j = 1(1)N$ for D_2

Let $Q = \text{diag}(Q, Q, Q, \dots, Q)$ be a matrix of order NP ; thus Q satisfy $Q^T Q = I$ since $Q^T Q = I$.

Since R_i, S_i and T_i are symmetric matrices, we have

$$\begin{aligned}
 Q^T R_i Q &= \text{diag}(\mu_{j1}^i, \mu_{j2}^i, \dots, \mu_{jP}^i) = Y_i \text{ where } \mu_{jP}^i = \lambda_{ij} + 2\beta_{ij} \cos\left(\frac{k\pi}{P+1}\right) \\
 Q^T S_i Q &= \text{diag}(\xi_{j1}^i, \xi_{j2}^i, \dots, \xi_{jP}^i) = \Phi_i \text{ where } \mu_{jP}^i = \eta_{ij} + 2a_7(i) \cos\left(\frac{k\pi}{P+1}\right) \\
 Q^T T_i Q &= \text{diag}(\tau_{j1}^i, \tau_{j2}^i, \dots, \tau_{jP}^i) = \Psi_i \text{ where } \tau_{jP}^i = \zeta_{ij} + 2a_8(i) \cos\left(\frac{k\pi}{P+1}\right)
 \end{aligned}$$

Let

$$Q^T U_i = V_i \Rightarrow U_i = Q V_i, \quad Q^T B_i = \bar{b}_i \Rightarrow B_i = Q \bar{b}_i \tag{17}$$

where

$$\begin{aligned}
 V_i &= [V_{i1} \ V_{i2} \ V_{i3} \ \dots \ V_{iP}]^T, \quad V_{ik} = [v_{i1k} \ v_{i2k} \ v_{i3k} \ \dots \ v_{iNk}]^T; \\
 \bar{b}_i &= [b_{i1} \ b_{i2} \ \dots \ b_{ik}]^T \text{ and } b_{ik} = [b_{i1k} \ b_{i2k} \ \dots \ b_{iNk}]^T
 \end{aligned}$$

Pre-multiplying Equation (16) by \mathbb{Q}^T and applying (17), we get

$$\begin{aligned}\Upsilon_1 \mathbf{V}_1 + \Phi_1 \mathbf{V}_2 &= \bar{\mathbf{b}}_1 \\ \Psi_2 \mathbf{V}_1 + \Upsilon_2 \mathbf{V}_2 + \Phi_2 \mathbf{V}_3 &= \bar{\mathbf{b}}_2 \\ \Psi_3 \mathbf{V}_2 + \Upsilon_3 \mathbf{V}_3 + \Phi_3 \mathbf{V}_4 &= \bar{\mathbf{b}}_3 \\ &\dots \\ \Psi_M \mathbf{V}_{M-1} + \Upsilon_M \mathbf{V}_M &= \bar{\mathbf{b}}_M\end{aligned}\quad (18)$$

Now from each Equation of (18) we collect the first equations and put them as one group of equation

$$\begin{aligned}\mu_{jk}^1 V_k^1 + \xi_{jk}^1 V_k^2 &= \bar{\mathbf{b}}_1 \\ \tau_{jk}^2 V_k^1 + \mu_{jk}^2 V_k^2 + \xi_{jk}^2 V_k^3 &= \bar{\mathbf{b}}_2 \\ \tau_{jk}^3 V_k^2 + \mu_{jk}^3 V_k^3 + \xi_{jk}^3 V_k^4 &= \bar{\mathbf{b}}_3 \\ \tau_{jk}^{M-1} V_k^{M-2} + \mu_{jk}^{M-1} V_k^{M-1} + \xi_{jk}^{M-1} V_k^M &= \bar{\mathbf{b}}_{M-1} \\ \tau_{jk}^M V_k^{M-1} + \mu_{jk}^M V_k^M &= \bar{\mathbf{b}}_M\end{aligned}\quad (19)$$

Now put $k=1$ in Equation (19) and collect the entire first set of equations, for $i=1,2,3,\dots,M$ and $j=1,2,3,\dots,N$ to get

$$\tau_{j1}^i v_{j1}^{i-1} + \mu_{j1}^i v_{j1}^i + \xi_{j1}^i v_{j1}^{i+1} = \bar{\mathbf{b}}_i \quad \text{and} \quad v_{j1}^0 = 0 = v_{j1}^M \quad (20a)$$

Again consider the second equations by putting $k=2$, and get

$$\tau_{j2}^i v_{j2}^{i-1} + \mu_{j2}^i v_{j2}^i + \xi_{j2}^i v_{j2}^{i+1} = \bar{\mathbf{b}}_i \quad \text{and} \quad v_{j2}^0 = 0 = v_{j2}^M \quad (20b)$$

Continuing in this manner and finally considering the last equations for $k=P$, we obtain

$$\tau_{jP}^i v_{jP}^{i-1} + \mu_{jP}^i v_{jP}^i + \xi_{jP}^i v_{jP}^{i+1} = \bar{\mathbf{b}}_i \quad \text{and} \quad v_{jP}^0 = 0 = v_{jP}^M \quad (20c)$$

All these set of Equations (20a)-(20c) are tri-diagonal ones and hence we solve for v_{jk}^i by using Thomas algorithm. With the help of (17) again we get all u_{jk}^i and this solves (14) as desired. By doing this we generally reduce the number of computations and computational time.

4. Numerical Results

In order to test the efficiency and adaptability of the proposed method, computational experiments are done on some selected problems that may arise in practice, for which the analytical solutions of U are known to us. The computed solutions are found for all grid points for any values of M, N and P . Here results are reported at some randomly taken mesh points in terms of the absolute maximum error from [Table 1](#) to [7](#).

Example 1. Consider $\nabla^2 U = 0$ with the boundary conditions $U(0, \theta, z) = 0$, $U(1, \theta, z) = z \sin \theta$

$$U(r, 0, z) = 0 = U(r, \pi, z), \quad \text{and} \quad U(r, \theta, 0) = 0, U(r, \theta, 1) = r \sin \theta$$

The analytical solution is $U(r, \theta, z) = rz \sin \theta$ and the computed results of this example are shown in [Table 1](#).

Example 2. Consider $\nabla^2 U = -\pi^2 r \cos \theta \sin \pi z$ with the boundary conditions

$$U(1, \theta, z) = \cos \theta \sin \pi z, \quad U(2, \theta, z) = 2 \cos \theta \sin \pi z$$

$$U(r, 0, z) = r \sin \pi z, \quad U\left(r, \frac{\pi}{2}, z\right) = 0, \quad \text{and} \quad U(r, \theta, 0) = 0 = U(r, \theta, 1)$$

The analytical solution is $U(r, \theta, z) = r \cos \theta \sin \pi z$ and the computed results of this example are shown in [Table 2](#).

Example 3. Consider $\nabla^2 U = -3 \cos \theta$ with the boundary conditions

$$U(0, \theta, z) = U(1, \theta, z) = -2z, \quad U(r, 0, z) = r(1-r) - 2z, \quad U\left(r, \frac{\pi}{2}, z\right) = -2z$$

Table 1. Maximum absolute error of example 1.

(N, P, M)	Max. absolute error	(N, P, M)	Max. absolute error
(9,9,9)	3.51670e-005	(29,9,39)	1.37257e-006
(9,9,29)	1.46565e-005	(29,19,9)	4.15180e-006
(9,19,9)	3.53325e-005	(29,29,19)	2.45633e-006
(9,19,19)	2.06578e-005	(29,29,29)	1.74383e-006
(9,29,39)	1.13280e-005	(29,39,19)	2.45924e-006
(9,39,29)	1.46438e-005	(29,39,29)	1.74829e-006
(19,9,9)	9.21838e-006	(39,9,19)	1.35171e-006
(19,9,19)	5.32850e-006	(39,9,39)	7.75143e-007
(19,19,19)	5.46733e-006	(39,19,29)	9.82456e-007
(19,29,39)	3.02425e-006	(39,29,19)	1.38647e-006
(19,39,9)	9.27536e-006	(39,39,9)	2.34568e-006
(19,39,39)	3.02636e-006	(39,39,39)	7.68613e-007

Table 2. Maximum absolute error of example 2.

(N, P, M)	Max. absolute error	(N, P, M)	Max. absolute error
(9,9,9)	2.93159e-003	(29,9,39)	2.98714e-003
(9,9,29)	2.95649e-003	(29,19,9)	7.39877e-004
(9,19,9)	7.32025e-004	(29,29,19)	3.31950e-004
(9,19,19)	7.38648e-004	(29,29,29)	3.31771e-004
(9,29,39)	3.27574e-004	(29,39,19)	1.86718e-004
(9,39,29)	1.83450e-004	(29,39,29)	1.86618e-004
(19,9,9)	2.95328e-003	(39,9,19)	2.98618e-003
(19,9,19)	2.97861e-003	(39,9,39)	2.98710e-003
(19,19,19)	7.44907e-004	(39,19,29)	7.46353e-004
(19,29,39)	3.31145e-004	(39,29,19)	3.31953e-004
(19,39,9)	1.84585e-004	(39,39,9)	1.84916e-004
(19,39,39)	1.86232e-004	(39,39,39)	1.86784e-004

$$U(r, \theta, 0) = r(1-r)\cos\theta, \quad U(r, \theta, 1) = r(1-r)\cos\theta - 2$$

The analytical solution is $U(r, \theta, z) = r(1-r)\cos\theta - 2z$ and the computed results of this example are shown in **Table 3**.

Example 4. Consider $\nabla^2 U = -\pi^2 \left(r^2 - \frac{1}{r^2} \right) \sin(2\theta) \sin(\pi z)$ with the boundary conditions

$$U(1, \theta, z) = 0, \quad U(2, \theta, z) = \frac{15}{4} \sin(2\theta) \sin(\pi z), \quad U(r, 0, z) = 0 = U\left(r, \frac{\pi}{2}, z\right) \quad \text{and} \quad U(r, \theta, 0) = 0 = U(r, \theta, 1)$$

The analytical solution is $U(r, \theta, z) = \left(r^2 - \frac{1}{r^2} \right) \sin(2\theta) \sin(\pi z)$ and the computed results of this example are shown in **Table 4**.

Example 5 Consider $\nabla^2 U = (8rz(1-z) - 2r^3)(\sin\theta + \cos\theta)$, where $0 \leq \theta < 2\pi$ with the boundary conditions

$$U(0, \theta, z) = 0, \quad U(1, \theta, z) = z(1-z)(\sin\theta + \cos\theta) \quad U(r, \theta, 0) = 0 = U(r, \theta, 1)$$

The analytical solution is $U(r, \theta, z) = r^3 z(1-z)(\sin\theta + \cos\theta)$ and the computed results of this example are shown in **Table 5**.

Table 3. Maximum absolute error of example 3.

(N, P, M)	Max. absolute error	(N, P, M)	Max. absolute error
(9,9,9)	1.81124e-004	(29,9,39)	1.16544e-005
(9,9,29)	4.45263e-005	(29,19,9)	1.82484e-004
(9,19,9)	1.81185e-004	(29,29,19)	4.61297e-005
(9,19,19)	6.02480e-005	(29,29,29)	2.04978e-005
(9,29,39)	3.97430e-005	(29,39,19)	4.61300e-005
(9,39,29)	4.46327e-005	(29,39,29)	2.04979e-005
(19,9,9)	1.81939e-004	(39,9,19)	4.61828e-005
(19,9,19)	4.59426e-005	(39,9,39)	1.17058e-005
(19,19,19)	4.59583e-005	(39,19,29)	2.05467e-005
(19,29,39)	1.50833e-005	(39,29,19)	4.61879e-005
(19,39,9)	1.82013e-004	(39,39,9)	1.82652e-004
(19,39,39)	1.50852e-005	(39,39,39)	1.15493e-005

Table 4. Maximum absolute error of example 4.

(N, P, M)	Max. absolute error	(N, P, M)	Max. absolute error
(9,9,9)	3.68396e-003	(29,9,39)	3.98135e-003
(9,9,29)	4.07400e-003	(29,19,9)	6.33780e-004
(9,19,9)	7.68229e-004	(29,29,19)	3.64070e-004
(9,19,19)	1.04366e-003	(29,29,29)	4.17368e-004
(9,29,39)	5.73867e-004	(29,39,19)	1.75928e-004
(9,39,29)	3.62888e-004	(29,39,29)	2.24720e-004
(19,9,9)	3.58663e-003	(39,9,19)	3.89251e-003
(19,9,19)	3.92179e-003	(39,9,39)	3.97355e-003
(19,19,19)	9.34774e-004	(39,19,29)	9.60868e-004
(19,29,39)	4.61633e-004	(39,29,19)	3.55183e-004
(19,39,9)	7.29565e-004	(39,39,9)	7.23913e-004
(19,39,39)	2.68695e-004	(39,39,39)	2.34933e-004

Table 5. Maximum absolute error of example 5.

(N, P, M)	Max. absolute error	(N, P, M)	Max. absolute error
(9,9,9)	5.97062e-004	(29,9,39)	1.65910e-004
(9,9,29)	4.42157e-004	(29,19,9)	4.11093e-004
(9,19,9)	5.09956e-004	(29,29,19)	1.01380e-004
(9,19,19)	3.72361e-004	(29,29,29)	6.92680e-005
(9,29,39)	3.26827e-004	(29,39,19)	1.03392e-004
(9,39,29)	3.27891e-004	(29,39,29)	6.38312e-005
(19,9,9)	3.72181e-004	(39,9,19)	1.80739e-004
(19,9,19)	2.39220e-004	(39,9,39)	1.49613e-004
(19,19,19)	1.52973e-004	(39,19,29)	6.95506e-005
(19,29,39)	1.04227e-004	(39,29,19)	1.06985e-004
(19,39,9)	3.96923e-004	(39,39,9)	4.28673e-004
(19,39,39)	9.84850e-005	(39,39,39)	3.91182e-005

This example was considered by M.C. Lai [1] as a test problem and our results are better than their results in terms of accuracy. For instance, for (8,16,16) the maximum absolute error in their result is 9.1438e-004 and while ours is 3.28689e-004.

Example 6 Consider $\nabla^2 U = 6rz \cos \theta$, where $0 \leq \theta < 2\pi$ with the boundary conditions

$$U(0, \theta, z) = 0, \quad U(1, \theta, z) = z \cos^3 \theta; \quad U(r, \theta, 0) = 0 \quad \text{and} \quad U(r, \theta, 1) = r^3 \cos^3 \theta$$

The analytical solution is $U(r, \theta, z) = r^3 z \cos^3 \theta$ and the computed results are shown in **Table 6**.

Example 5.7 Consider $\nabla^2 U = -\pi^2 \left(r^2 - \frac{1}{r^2} \right) \sin(2\theta) \sin(\pi z)$ where $0 \leq \theta < 2\pi$ with the boundary conditions

$$U(1, \theta, z) = 0, \quad U(2, \theta, z) = \frac{15}{4} \sin(2\theta) \sin(\pi z); \quad U(r, \theta, 0) = 0 = U(r, \theta, 1)$$

The analytical solution is $U(r, \theta, z) = \left(r^2 - \frac{1}{r^2} \right) \sin(2\theta) \sin(\pi z)$ and the computed results of this example are shown in **Table 7**.

Table 6. Maximum absolute error of example 6.

(N, P, M)	Max. absolute error	(N, P, M)	Max. absolute error
(9,9,9)	3.04648e-003	(29,9,39)	3.06543e-004
(9,9,29)	3.18297e-003	(29,19,9)	2.00777e-004
(9,19,9)	3.05549e-003	(29,29,19)	2.85659e-004
(9,19,19)	3.16606e-003	(29,29,29)	3.02059e-004
(9,29,39)	3.19893e-003	(29,39,19)	2.85718e-004
(9,39,29)	3.19459e-003	(29,39,29)	3.02122e-004
(19,9,9)	6.03143e-004	(39,9,19)	1.42033e-004
(19,9,19)	6.87721e-004	(39,9,39)	1.62951e-004
(19,19,19)	6.89766e-004	(39,19,29)	1.58004e-004
(19,29,39)	7.13568e-004	(39,29,19)	1.42553e-004
(19,39,9)	6.05428e-004	(39,39,9)	1.58596e-004
(19,39,39)	7.13712e-004	(39,39,39)	1.63583e-004

Table 7. Maximum absolute error of example 7.

(N, P, M)	Max. absolute error	(N, P, M)	Max. absolute error
(9,9,9)	3.00418e-003	(29,9,39)	4.13706e-003
(9,9,29)	2.36262e-003	(29,19,9)	8.41886e-004
(9,19,9)	4.54924e-003	(29,29,19)	5.78646e-004
(9,19,19)	4.13088e-003	(29,29,29)	6.30456e-004
(9,29,39)	4.49099e-003	(29,39,19)	3.91287e-004
(9,39,29)	4.67590e-003	(29,39,29)	4.41870e-004
(19,9,9)	3.54731e-003	(39,9,19)	4.01710e-003
(19,9,19)	3.84938e-003	(39,9,39)	4.09806e-003
(19,19,19)	1.08325e-003	(39,19,29)	1.10354e-003
(19,29,39)	6.43373e-004	(39,29,19)	5.01257e-004
(19,39,9)	9.83307e-004	(39,39,9)	7.61254e-004
(19,39,39)	4.66590e-004	(39,39,39)	3.82100e-004

5. Conclusions

In this work, we have transformed the three dimensional Poisson's equation in cylindrical coordinates system into a system of algebraic linear equations using its equivalent fourth-order finite difference approximation scheme. The resulting large number of algebraic equation is, then, systematically arranged in order to get a block matrix. By extending Hockney's method to three dimensions, we reduced the obtained matrix into a block tridiagonal matrix, and each block is solved by the help of Thomas algorithm. We have successfully implemented this method to find the solution of the three dimensional Poisson's equation in cylindrical coordinates system and it is found that the method can easily be applied and adapted to find a solution of some related applied problems. The method produced accurate results considering double precision. This method is direct and allows considerable savings in computer storage as well as execution speed.

Therefore, the method is suitable to apply to some three dimensional Poisson's equations.

References

- [1] Lai, M.C. (2002) A Simple Compact Fourth-Order Poisson Solver on Polar Geometry. *Journal of Computational Physics*, **182**, 337-345. <http://dx.doi.org/10.1006/jcph.2002.7172>
- [2] Mittal, R.C and Gahlaut, S. (1987) High Order Finite Difference Schemes to Solve Poisson's Equation in Cylindrical Symmetry. *Communications in Applied Numerical Methods*, **3**, 457-461.
- [3] Mittal, R.C. and Gahlaut, S. (1991) High-Order Finite Differences Schemes to Solve Poisson's Equation in Polar Coordinates. *IMA Journal of Numerical Analysis*, **11**, 261-270. <http://dx.doi.org/10.1093/imanum/11.2.261>
- [4] Alemayehu, S. and Mittal, R.C. (2013) Fast Finite Difference Solutions of the Three Dimensional Poisson's Equation in Cylindrical Coordinates. *American Journal of Computational Mathematics*, **3**, 356-361.
- [5] Tan, C.S. (1985) Accurate Solution of Three Dimensional Poisson's Equation in Cylindrical Coordinate by Expansion in Chebyshev Polynomials. *Journal of Computational Physics*, **59**, 81-95. [http://dx.doi.org/10.1016/0021-9991\(85\)90108-1](http://dx.doi.org/10.1016/0021-9991(85)90108-1)
- [6] Iyengar, S.R.K. and Manohar, R. (1988) High Order Difference Methods for Heat Equation in Polar Cylindrical Polar Cylindrical Coordinates. *Journal of Computational Physics*, **77**, 425-438. [http://dx.doi.org/10.1016/0021-9991\(88\)90176-3](http://dx.doi.org/10.1016/0021-9991(88)90176-3)
- [7] Iyengar, S.R.K. and Goyal, A. (1990) A Note on Multigrid for the Three-Dimensional Poisson Equation in Cylindrical Coordinates. *Journal of Computational and Applied Mathematics*, **33**, 163-169. [http://dx.doi.org/10.1016/0377-0427\(90\)90366-8](http://dx.doi.org/10.1016/0377-0427(90)90366-8)
- [8] Lai, M.C. and Tseng, J.M. (2007) A formally Fourth-Order Accurate Compact Scheme for 3D Poisson Equation in Cylindrical and Spherical Coordinates. *Journal of Computational and Applied Mathematics*, **201**, 175-181. <http://dx.doi.org/10.1016/j.cam.2006.02.011>
- [9] Smith, G.D. (1985) Numerical Solutions of Partial Differential Equations: Finite Difference Methods. Third Edition. Oxford University Press, New York.
- [10] Malcolm, M.A. and Palmer, J. (1974) A Fast Method for Solving a Class of Tri-Diagonal Linear Systems. *Communications of Association for Computing Machinery*, **17**, 14-17. <http://dx.doi.org/10.1145/360767.360777>
- [11] Hockney, R.W. (1965) A Fast Direct Solution of Poisson Equation Using Fourier Analysis. *Journal of Alternative and Complementary Medicine*, **12**, 95-113. <http://dx.doi.org/10.1145/321250.321259>

L-Stable Block Hybrid Second Derivative Algorithm for Parabolic Partial Differential Equations

Fidele Fouogang Ngwane^{1*}, Samuel Nemsefor Jator²

¹Department of Mathematics, USC Salkehatchie, Allendale, USA

²Department of Mathematics and Statistics, Austin Peay State University, Clarksville, USA

Email: *fifonge@yahoo.com

Received 28 January 2014; revised 28 February 2014; accepted 5 March 2014

Copyright © 2014 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

An *L*-stable block method based on hybrid second derivative algorithm (BHSDA) is provided by a continuous second derivative method that is defined for all values of the independent variable and applied to parabolic partial differential equations (PDEs). The use of the BHSDA to solve PDEs is facilitated by the method of lines which involves making an approximation to the space derivatives, and hence reducing the problem to that of solving a time-dependent system of first order initial value ordinary differential equations. The stability properties of the method is examined and some numerical results presented.

Keywords

Hybrid Second Derivative Method; Off-Step Point; Parabolic; Partial Differential Equations

1. Introduction

We adopt the method of lines approach which is commonly used for solving time-dependent partial differential equations (PDE), whereby the spatial derivatives are replaced by finite difference approximations (see Lambert [1], Ramos and Vigo-Aguiar [2], Brugnano and Trigiante [3], Cash [4], Enright [5], Hairer *et al.* [6], Henrici [7], Butcher [8], Fatunla [9], Jator [10], and Onumanyi *et al.* [11], [12]). Consider the PDE of the form

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, (x, t) \in [0, 1] \times [0 < t \leq T] \quad (1)$$

*Corresponding author.

subject to the initial/boundary conditions

$$u(x, 0) = G(x), x \in [0, 1], u(0, t) = u(1, t) = 0, t \geq 0. \quad (2)$$

We seek a solution in the strip $[0, 1] \times [0 < t \leq T]$ by first fixing the grid in the spatial variable x , then approximating this spatial derivative using the central difference method, and finally solving the resulting system of first order time dependent ODEs. Specifically, we discretize the space variable with mesh spacings $\Delta x = 1/M$,

$$x_m = m\Delta x, m = 0, 1, \dots, M.$$

We then define $u_m(t) \approx u(x_m, t)$, $\mathbf{u}(t) = [u_1(t), \dots, u_m(t)]^T$, and replace the partial derivatives $\frac{\partial^2 u(x, t)}{\partial x^2}$

occurring in (1) by the central difference approximation to obtain

$$\frac{\partial u(x_m, t)}{\partial t} = [u(x_{m+1}, t) - 2u(x_m, t) + u(x_{m-1}, t)] / (\Delta x)^2; \quad m = 0, 1, \dots, M-1, \text{ which reduces the PDE to the semi-discrete problem}$$

$$\frac{du_m}{dt} = \frac{1}{(\Delta x)^2} (u_{m+1} - 2u_m + u_{m-1})$$

which can be written in the form

$$\mathbf{u}' = \mathbf{f}(t, \mathbf{u}), \mathbf{u}(0) = \mathbf{u}_0, \quad (3)$$

where $\mathbf{f}(t, \mathbf{u}) = \mathbf{A}\mathbf{u}$, and \mathbf{A} is an $M \times M$ matrix arising from the central difference approximations to the derivatives of x . The problem (2) is now a system of first order ODEs which is solved by the BHSDA.

The paper is organized as follows. In Section 2, we derive a continuous approximation which is used to obtain the BHSDA. The BHSDA is also analyzed in Section 2. The computational aspects of the method is given in Section 3. Numerical examples are given in Section 4 to show the accuracy of the method. Finally, the conclusion of the paper is discussed in Section 5.

2. Development of the Method

We begin by considering a scalar form of (3)

$$u' = f(t, u), u(t_0) = u_0, t \in [t_0, t_N] \quad (4)$$

where we assume that the function f is Lipschitz continuous and the problem (4) possesses a unique solution. Furthermore, let u_n be an approximation of the theoretical solution $u(t)$ at t_n . Our objective is to simultaneously seek numerical approximations at the points $t_{n+\nu} = t_n + \nu h$ and $t_{n+1} = t_n + h$ respectively, where h is the step size, n the grid index, and $\nu \in (0, 1)$. This approximation u_n is provided by a continuous approximation $U(t)$ as a by-product. Thus, we assume that $U(t)$ is of the form

$$U(t) = \sum_{j=0}^4 \ell_j t^j \quad (5)$$

where ℓ_j are unknown coefficients.

In order to uniquely determine the unknown coefficients ℓ_j , we impose that the interpolating function (4) coincides with the analytical solution at the end point t_n and also satisfies the differential Equation (3) at the points $t_{n+j\nu}$, $j = 0, 1, 2$ to obtain the following system of equations:

$$U(t_n) = y_n, U'(t_{n+j\nu}) = f_{n+j\nu}, U''(t_{n+1}) = g_{n+1}, j = 0, 1, 2. \quad (6)$$

We note that (6) leads to a system of five equations which is solved by Cramer's Rule to obtain ℓ_j . The continuous method is constructed by substituting the values of ℓ_j into Equation (5) which is simplified and expressed in the form

$$U(t) = y_n + h(\beta_0(t)f_n + \beta_1(t)f_{n+1} + \beta_\nu(t)f_{n+\nu}) + h^2\gamma_1(t)g_{n+1} \quad (7)$$

where $\beta_0(t)$, $\beta_1(t)$, $\beta_\nu(t)$, $\gamma_1(t)$, are continuous coefficients, and $g_{n+1} = \frac{df(t, u(t))}{dt} \Big|_{u_{n+1}}^{t_{n+1}}$. The continuous method (7) is then evaluated at $t = \{t_{n+\nu}, t_{n+1}\}$, for $\nu = 1/2$ to yield

$$\begin{cases} y_{n+1/2} = y_n + \frac{h}{96}(17f_n + 44f_{n+1/2} - 13f_{n+1}) + \frac{h^2}{96}(3g_{n+1}) \\ y_{n+1} = y_n + \frac{h}{6}(f_n + 4f_{n+1/2} + f_{n+1}). \end{cases} \quad (8)$$

Remark 2.1 In order to conveniently analyze and implement the method (8), we will express it in block form as given in (9).

$$A^{(0)}Y_\mu = A^{(1)}Y_{\mu-1} + h[B^{(0)}F_\mu + B^{(1)}F_{\mu-1}] + h^2C^{(0)}G_\mu \quad (9)$$

where $Y_\mu = \begin{pmatrix} u_{n+\frac{1}{2}}, u_{n+1} \end{pmatrix}^T$, $Y_{\mu-1} = \begin{pmatrix} u_{n-\frac{1}{2}}, u_n \end{pmatrix}^T$, $F_\mu = \begin{pmatrix} f_{n+\frac{1}{2}}, f_{n+1} \end{pmatrix}^T$, $F_{\mu-1} = \begin{pmatrix} f_{n-\frac{1}{2}}, f_n \end{pmatrix}^T$, $G_\mu = (0, g_{n+1})^T$,

$\mu = 1, \dots, n = 0, 1, \dots$, and the matrices $A^{(0)}$, $A^{(1)}$, $B^{(0)}$, $B^{(1)}$, $C^{(0)}$ are 2 by 2 matrices whose entries are given by the coefficients of (8).

2.1. Local Truncation Error

Define the local truncation error of (4) as

$$\mathbb{L}[z(t); h] = Z_\mu - A^{(1)}Z_{\mu-1} - h[B^{(0)}\bar{F}_\mu + B^{(1)}\bar{F}_{\mu-1}] - h^2C^{(0)}\bar{G}_\mu \quad (10)$$

where

$$Z_\mu = \begin{pmatrix} u\left(t_{n+\frac{1}{2}}\right), u(t_{n+1}) \end{pmatrix}^T, \quad Z_{\mu-1} = \begin{pmatrix} u\left(t_{n-\frac{1}{2}}\right), u(t_n) \end{pmatrix}^T, \quad \bar{F}_\mu = \begin{pmatrix} f\left(t_{n+\frac{1}{2}}, u\left(t_{n+\frac{1}{2}}\right)\right), f(t_{n+1}, u(t_{n+1})) \end{pmatrix}^T, \\ \bar{F}_{\mu-1} = \begin{pmatrix} f\left(t_{n-\frac{1}{2}}, u\left(t_{n-\frac{1}{2}}\right)\right), f(t_n, u(t_n)) \end{pmatrix}^T, \quad \text{and } \mathbb{L}[z(t); h] = (\mathbb{L}_1[z(t); h], \mathbb{L}_2[z(t); h])^T \text{ is a linear difference}$$

operator. Assuming that $z(t)$ is sufficiently differentiable, we can expand the terms in (10) as a Taylor series about the point t_n to obtain the expression for the local truncation error. $\mathbb{L}[z(t); h] = O(h^5)$, hence the method is of order four.

2.2. Stability

Proposition 2.2 The BHSDA (9) applied to the test equations $u' = \lambda u$ and $u'' = \lambda^2 u$ yields.

$$Y_\mu = M(q)Y_{\mu-1}, \quad q = \lambda h, \quad (11)$$

with the amplification matrix

$$M(q) = (A^{(0)} - qB^{(0)} - q^2C^{(0)})^{-1} (A^{(1)} + qB^{(1)}). \quad (12)$$

Remark 2.3 The dominant eigenvalue of $M(q)$ specified by $q_{\max} = \frac{48+18q+2q^2}{48-30q+8q^2-q^3}$ is a rational function called the stability function which determines the stability of the method.

Proof. We begin by applying (2) to the test equations $u' = \lambda u$ and $u'' = \lambda^2 u$ which are expressed as $f(t, u) = \lambda u$ and $g(t, u) = \lambda^2 u$ respectively; letting $q = h\lambda$, we obtain a system of linear equations which is used to solve for Y_μ with (12) as a consequence.

Definition 2.4 The block method (9) is said to be 1) A -stable if for all $q \in \mathbb{C}^-$, $M(q)$ has a dominant eigenvalue q_{\max} such that $|q_{\max}| \leq 1$; moreover, since q_{\max} is a rational function, the real part of the zeros of q_{\max} must be negative, while the real part of the poles of q_{\max} must be positive; 2) L -stable if it is A -stable and $q_{\max} \rightarrow 0$ as $q \rightarrow -\infty$.

Corollary 2.5 The method (9) is A -stable and L -stable.

Proof: The dominant eigenvalue q_{\max} for the method (9) is given by $q_{\max} = \frac{48+18q+2q^2}{48-30q+8q^2-q^3}$ and the proof follows from definition 2.4.

Remark 2.6 The stability region for the method (9) is given in **Figure 1** showing the zeros and poles of the dominant eigenvalue q_{\max} .

3. Computational Aspects

The resulting system of ODEs (3) is then solved on the partition

$$\pi_N : \{t_0 < t_1 < \dots < t_N, t_n = t_0 + nh\}$$

$h = \Delta t = \frac{b-a}{N}$ is a constant step-size of the partition of π_N , $n=1, 2, \dots, N$, N is a positive integer and n the grid index.

Step 1: Use the block method (9) to solve (3) on rectangles $[t_0, t_1] \times [0, 1]$, $[t_1, t_2] \times [0, 1]$, \dots , $[t_{N-1}, t_N] \times [0, 1]$.

Step 2: Let $Y_{m,\mu} = \left(u_{m,n+\frac{1}{2}}, u_{m,n+1} \right)^T$, noting that $u_m(t_n) \approx u_{m,n} \approx u(x_m, t_n)$, then for $m=1, \dots, M$, $n=0$,

and $\mu=1$, the approximations $Y_{m,1} = \left(u_{m,\frac{1}{2}}, u_{m,1} \right)^T$ are simultaneously obtained on $[t_0, t_1] \times [0, 1]$.

Step 3: Step 2 is repeated for $m=1, \dots, M$, $n=1, 2, \dots, N-1$, and $\mu=2, 3, \dots, N$, to generate the approximations $Y_{m,2}, Y_{m,3}, \dots, Y_{m,N}$ on $[t_1, t_2] \times [0, 1]$, \dots , $[t_{N-1}, t_N] \times [0, 1]$.

We note that for linear problems, we solve (3) directly with our Mathematica code enhanced by the feature `NSolve[]`.

4. Numerical Examples

Computations were carried out in Mathematica 9.0 and the errors were calculated as $|u_{m,n} - u(x_m, t_n)|$, where $u_m(t_n) \approx u_{m,n}$. We note that the method is particularly useful, but not limited to solving parabolic partial differential equations where the solution decays very rapidly and where the PDEs are stiff parabolic equations (see Cash [4]).

Example 4.1 As our first test example, we solve the given PDE (see Cash [4])

$$\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2}, u(0, t) = u(1, t) = 0, u(x, 0) = \sin \pi x.$$

The exact solution $u(x, t) = e^{-\pi^2 \kappa t} \sin \pi x$.

In **Table 1**, it is noticed that the method with the BHSDA is the most accurate.

Example 4.2 As our second test example, we solve the given stiff parabolic equation (see Cash [4])

$$\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2}, u(0, t) = u(1, t) = 0, u(x, 0) = \sin \pi x + \sin \omega \pi x, \omega \gg 1.$$

The exact solution $u(x, t) = e^{-\pi^2 \kappa t} \sin \pi x + e^{-\omega^2 \pi^2 \kappa t} \sin \omega \pi x$.

Cash [4] notes that as ω increases, equations of the type given in example 4.2 exhibit characteristics similar to model stiff equations. Hence, the methods such as the Crank-Nicolson method which are not L_0 -stable are expected to perform poorly. The BHSDA is L -stable and perform excellently when applied to this problem. Therefore the BHSDA is competitive with the L_0 -stable methods of Cash [4]. In **Table 2**, we display the results for $\kappa=1$ and a range of values for ω .

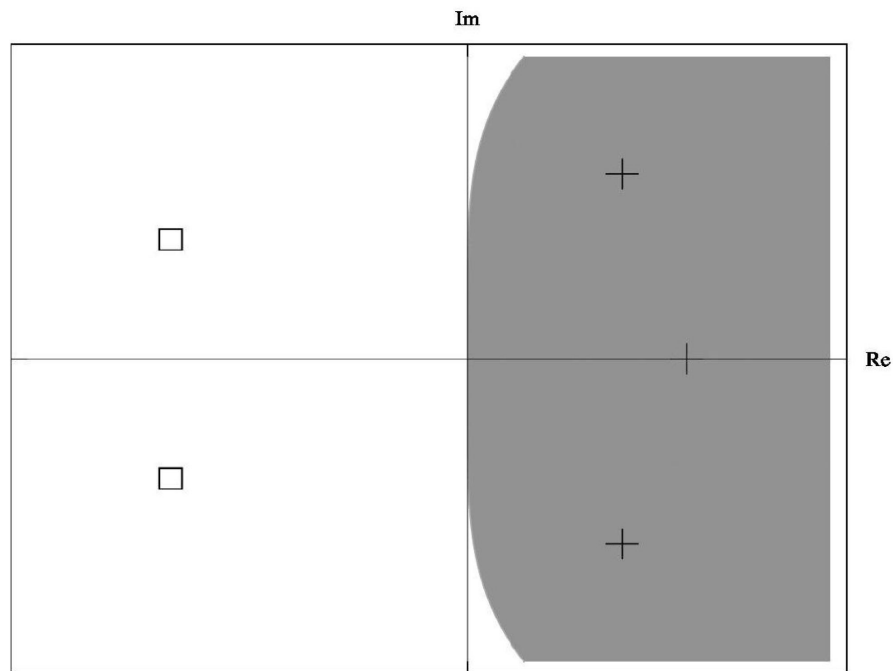


Figure 1. The region of absolute stability of the BHSDA of order 4 is to the left of the dividing line and is symmetric about the real axis; the square and plus symbols to the left and right of the imaginary axis represent the zeros and poles of q_{\max} respectively.

Table 1. A comparison of errors of methods for Example 4.1 at $t = 1$.

Δx	Δt	κ	Crank-Nicolson	Cash (2.6a, b)	Cash (2.13a, b, c)	BHSDA
0.1	0.1	1	3.0×10^{-5}	1.5×10^{-5}	4.5×10^{-6}	1.3×10^{-6}
0.05	0.05	1	9.0×10^{-6}	4.0×10^{-6}	2.7×10^{-7}	1.7×10^{-7}
0.1	0.1	5	2.0×10^{-4}	3.0×10^{-8}	2.0×10^{-10}	2.5×10^{-19}
0.05	0.05	5	1.0×10^{-14}	4.0×10^{-22}	3.7×10^{-22}	7.0×10^{-24}

Table 2. A comparison of errors of methods for Example 4.1 at $t = 1$ and $\omega = 1$, $\Delta x = 0.1$, $\Delta t = 0.1$.

ω	BHSDA	Crank-Nicolson	Cash (2.6a, b)	Cash (2.13a, b, c)
1	2.64×10^{-6}	6.20×10^{-5}	3.7×10^{-5}	1.5×10^{-5}
2	1.32×10^{-6}	3.83×10^{-5}	1.8×10^{-5}	7.4×10^{-6}
3	1.32×10^{-6}	9.30×10^{-3}	1.9×10^{-5}	7.4×10^{-6}
5	1.32×10^{-6}	1.80×10^{-1}	1.8×10^{-5}	7.4×10^{-6}
10	1.32×10^{-6}	6.10×10^{-1}	1.8×10^{-5}	7.4×10^{-6}

5. Conclusion

We have proposed a BHSDA for solving parabolic PDEs via the method of lines. The method is shown to be L -stable and competitive with existing methods in the literature.

References

- [1] Lambert, J.D. (1991) Numerical Methods for Ordinary Differential Systems. John Wiley, New York.
- [2] Vigo-Aguiar, J. and Ramos, H. (2007) A family of A-Stable Collocation Methods of Higher Order for Initial-Value Problems. *IMA Journal of Numerical Analysis*, **27**, 798-817. <http://dx.doi.org/10.1093/imanum/drl040>

- [3] Brugnano, L. and Trigiante, D. (1998) Solving Differential Problems by Multistep Initial and Boundary Value Methods. Gordon and Breach Science Publishers, Amsterdam.
- [4] Cash, J.R. (1984) Two New Finite Difference Schemes for Parabolic Equations. *SIAM Journal of Numerical Analysis*, **21**, 433-446. <http://dx.doi.org/10.1137/0721032>
- [5] Enright, W.H. (2000) Continuous Numerical Methods for ODEs with Defect Control. *Journal of Computational and Applied Mathematics*, **125**, 159-170. [http://dx.doi.org/10.1016/S0377-0427\(00\)00466-0](http://dx.doi.org/10.1016/S0377-0427(00)00466-0)
- [6] Hairer, E. and Wanner, G. (1996) Solving Ordinary Differential Equations II. Springer, New York. <http://dx.doi.org/10.1007/978-3-642-05221-7>
- [7] Henrici, P. (1962) Discrete Variable Methods in ODEs. John Wiley, New York.
- [8] Butcher, J.C. (1987) The Numerical Analysis of Ordinary Differential Equations, Runge-Kutta and General Linear Methods. Wiley, New York.
- [9] Fatunla, S.O. (1991) Block Methods for Second Order IVPs. *International Journal of Computational Mathematics*, **41**, 55-63. <http://dx.doi.org/10.1080/00207169108804026>
- [10] Jator, S.N. (2010) On the Hybrid Method with Three-Off Step Points for Initial Value Problems. *International Journal of Mathematical Education in Science and Technology*, **41**, 110-118 <http://dx.doi.org/10.1080/00207390903189203>
- [11] Onumanyi, P., Sirisena, U.W. and Jator, S.N. (1999) Continuous Finite Difference Approximations for Solving Differential Equations. *International Journal of Computational Mathematics*, **72**, 15-27. <http://dx.doi.org/10.1080/00207169908804831>
- [12] Onumanyi, P., Awoyemi, D.O., Jator, S.N. and Sirisena, U.W. (1994) New Linear Multistep Methods with Continuous Coefficients for First Order Initial Value Problems. *Journal of the Nigerian Mathematics Society*, 37-51.

Derivative of a Determinant with Respect to an Eigenvalue in the Modified Cholesky Decomposition of a Symmetric Matrix, with Applications to Nonlinear Analysis

Mitsuhiro Kashiwagi

School of Industrial Engineering, Tokai University, Kumamoto, Japan
Email: mkashi@tokai-u.jp

Received 8 November 2013; revised 8 December 2013; accepted 15 December 2013

Copyright © 2014 by author and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In this paper, we obtain a formula for the derivative of a determinant with respect to an eigenvalue in the modified Cholesky decomposition of a symmetric matrix, a characteristic example of a direct solution method in computational linear algebra. We apply our proposed formula to a technique used in nonlinear finite-element methods and discuss methods for determining singular points, such as bifurcation points and limit points. In our proposed method, the increment in arc length (or other relevant quantities) may be determined automatically, allowing a reduction in the number of basic parameters. The method is particularly effective for banded matrices, which allow a significant reduction in memory requirements as compared to dense matrices. We discuss the theoretical foundations of our proposed method, present algorithms and programs that implement it, and conduct numerical experiments to investigate its effectiveness.

Keywords

Derivative of a Determinant with Respect to an Eigenvalue; Modified Cholesky Decomposition; Symmetric Matrix; Nonlinear Finite-Element Methods; Singular Points

1. Introduction

The increasing complexity of computational mechanics has created a need to go beyond linear analysis into the realm of nonlinear problems. Nonlinear finite-element methods commonly employ incremental techniques involving local linearization, with examples including load-increment methods, displacement-increment methods,

How to cite this paper: Kashiwagi, M. (2014) Derivative of a Determinant with Respect to an Eigenvalue in the Modified Cholesky Decomposition of a Symmetric Matrix, with Applications to Nonlinear Analysis. *American Journal of Computational Mathematics*, 4, 93-103. <http://dx.doi.org/10.4236/ajcm.2014.42009>

and arc-length methods. Arc-length methods, which seek to eliminate the drawbacks of load-increment methods by choosing an optimal arc-length, are effective at identifying equilibrium paths including singular points.

In previous work [1], we proposed a formula for the derivative of a determinant with respect to an eigenvalue, based on the trace theorem and the expression for the inverse of the coefficient matrix arising in the conjugate-gradient method. In subsequent work [2]-[4], we demonstrated that this formula is particularly effective when applied to methods of eigenvalue analysis. However, the formula as proposed in these works was intended for use with *iterative* linear-algebra methods, such as the conjugate-gradient method, and could not be applied to *direct* methods such as the modified Cholesky decomposition. This limitation was addressed in Reference [5], in which, by considering the equations that arise in the conjugate-gradient method, we applied our technique to the *LDU* decomposition of a nonsymmetric matrix (a characteristic example of a direct solution method) and presented algorithms for differentiating determinants of both dense and banded matrices with respect to eigenvalues.

In the present paper, we propose a formula for the derivative of a determinant with respect to an eigenvalue in the modified Cholesky decomposition of a symmetric matrix. In addition, we apply our formula to the arc-length method (a characteristic example of a solution method for nonlinear finite-element methods) and discuss methods for determining singular points, such as bifurcation points and limit points. When the sign of the derivative of the determinant changes, we may use techniques such as the bisection method to narrow the interval within which the sign changes and thus pinpoint singular values. In addition, solutions obtained via the Newton-Raphson method vary with the derivative of the determinant, and this allows our proposed formula to be used to set the increment. The fact that the increment in the arc length (or other quantities) may thus be determined automatically allows us to reduce the number of basic parameters exerting a significant impact on a nonlinear finite-element method. Our proposed method is applicable to the LDL^T decomposition of dense matrices, as well as to the LDL^T decomposition of banded matrices, which afford a significant reduction in memory requirements compared to dense matrices. In what follows, we first discuss the theoretical foundations of our proposed method and present algorithms and programs that implement it. Then, we assess the effectiveness of our proposed method by applying it to a series of numerical experiments on a three-dimensional truss structure.

2. Derivative of a Determinant with Respect to an Eigenvalue in the Modified Cholesky Decomposition

The derivation presented in this section proceeds in analogy to that discussed in Reference 5. The eigenvalue problem may be expressed as follows. If \tilde{A} is a real-valued symmetric $n \times n$ matrix (specifically, the *tangent stiffness matrix* of a finite-element analysis), then the standard eigenvalue problem takes the form

$$\tilde{A}x = \lambda x, \quad (1)$$

where λ and x denote the eigenvalue and eigenvector, respectively. In order for Equation (1) to have trivial solutions, the matrix $\tilde{A} - \lambda I$ must be singular, *i.e.*,

$$\det[\tilde{A} - \lambda I] = 0. \quad (2)$$

We will use the notation $f(\lambda)$ for the left-hand side of this equation:

$$f(\lambda) = \det[\tilde{A} - \lambda I]. \quad (3)$$

Applying the trace theorem, we find

$$\frac{f'(\lambda)}{f(\lambda)} = \text{trace} \left[[\tilde{A} - \lambda I]^{-1} [\tilde{A} - \lambda I]' \right] = -\text{trace} \left[[\tilde{A} - \lambda I]^{-1} \right], \quad (4)$$

where

$$A = \tilde{A} - \lambda I = [a_{ij}] = \begin{bmatrix} a_{11} & & & \\ a_{21} & a_{22} & & \text{sym.} \\ \vdots & & \ddots & \\ a_{n1} & & \cdots & a_{nn} \end{bmatrix}. \quad (5)$$

In the case of LDL^T decomposition, we have

$$A = LDL^T \quad (6)$$

with factors L and D of the form

$$L = [\ell_{ij}] = \begin{bmatrix} 1 & & & 0 \\ \ell_{21} & 1 & & \\ \vdots & & \ddots & \\ \ell_{n1} & \cdots & \ell_{n(n-1)} & 1 \end{bmatrix}, \quad (7)$$

$$D = [d_{ij}] = \begin{bmatrix} d_{11} & & & 0 \\ & d_{22} & & \\ & & \ddots & \\ 0 & & & d_{nn} \end{bmatrix}. \quad (8)$$

The matrix L^{-1} has the form

$$L^{-1} = [g_{ij}] = \begin{bmatrix} 1 & & & 0 \\ g_{21} & 1 & & \\ \vdots & & \ddots & \\ g_{n1} & \cdots & g_{n(n-1)} & 1 \end{bmatrix}. \quad (9)$$

Expanding the relation $LL^{-1} = I$ (where I is the identity matrix) and collecting terms, we find

$$g_{ij} = -\ell_{ij} - \ell_{i2}g_{2j} - \ell_{i3}g_{3j} - \cdots - \ell_{i(i-1)}g_{(i-1)j}. \quad (10)$$

Equation (10) indicates that g_{ij} must be computed for all matrix elements; however, for matrix elements outside the bandwidth, we have $\ell_{ij} = 0$, and thus the computation requires only elements ℓ_{ij} within the bandwidth. This implies that a narrower bandwidth gives a greater reduction in computation time.

From Equation (4), we see that evaluating the derivative of a determinant requires only the diagonal elements of the inverse matrix (6). Upon expanding the product $[L^T]^{-1} D^{-1} L^{-1}$ using Equations (7-9) and summing the diagonal elements, Equation (4) takes the form

$$\frac{f'(\lambda)}{f(\lambda)} = -\sum_{i=1}^n \left(\frac{1}{d_{ii}} + \sum_{k=i+1}^n \frac{g_{ki}^2}{d_{kk}} \right). \quad (11)$$

This equation demonstrates that the derivative of the determinant may be computed from the elements of the inverses of the matrices D and L obtained from the modified Cholesky decomposition. As noted above, only matrix elements within a certain bandwidth of the diagonal are needed for this computation, and thus computations even for dense matrices may be carried out as if the matrices were banded. Because of this, we expect dense matrices not to require significantly more computation time than banded matrices.

By augmenting an LDL^T decomposition program with an additional routine (which simply adds one additional vector), we easily obtain a program for evaluating the quantity f'/f . The value of this quantity may be put to effective use in Newton-Raphson approaches to the numerical analysis of bifurcation points and limit points in problems such as large-deflection elastoplastic finite-element analysis. Our proposed method is easily implemented as a minor additional step in the process of solving simultaneous linear equations.

3. Algorithms Implementing the Proposed Method

3.1. Algorithm for Dense Matrices

We first present an algorithm for dense matrices. The arrays and variables appearing in this algorithm are as follows.

- 1) Computation of the modified Cholesky decomposition of a matrix together with its derivative with respect

to an eigenvalue

(1) Input data

A : given symmetric coefficient matrix, 2-dimension array as A(n,n)

b : work vector, 1-dimension array as b(n)

n : given order of matrix A and vector b

eps : parameter to check singularity of the matrix output

(2) Output data

A : L matrix and D matrix, 2-dimension array as A(n,n)

fd : differentiation of determinant

ichg : numbers of minus element of diagonal matrix D (numbers of eigenvalue)

ierr : error code

=0, for normal execution

=1, for singularity

(3) LDL^T decomposition

ichg=0

do i=1,n

<d(i,i)>

do k=1,i-1

$A(i,i) = A(i,i) - A(k,k) * A(i,k)^2$

end do

if (A(i,i)<0) ichg=ichg+1

if (abs(A(i,i))<eps) then

ierr=1

return

end if

<l(i,j)>

do j=i+1,n

do k=1,i-1

$A(j,i) = A(j,i) - A(j,k) * A(k,k) * A(i,k)$

end do

$A(j,i) = A(j,i) / A(i,i)$

end do

end do

ierr=0

(4) Derivative of a determinant with respect to an eigenvalue (fd)

fd=0

do i=1,n

<(i,i).>

$fd = fd - 1 / A(i,i)$

<(i,j)>

do j=i+1,n

$b(j) = -A(j,i)$

do k=1,j-i-1

$b(j) = b(j) - A(j,i+k) * b(i+k)$

end do

$fd = fd - b(j)^2 / A(j,j)$

end do

end do

2) Calculation of the solution

(1) Input data

A : L matrix and D matrix, 2-dimension array as A(n,n)

b : given right hand side vector, 1-dimension array as b(n)

n : given order of matrix A and vector b

```

(2) Output data
    b : work and solution vector, 1-dimension array
(3) Forward substitution
    do i=1,n
        do j=i+1,n
            b(j)=b(j)-A(j,i)*b(i)
        end do
    end do
(4) Backward substitution
    do i=1,n
        b(i)=b(i)/A(i,i)
    end do
    do i=1,n
        ii=n-i+1
        do j=1,ii-1
            b(j)=b(j)-A(ii,j)*b(ii)
        end do
    end do

```

3.2. Algorithm for Banded Matrices

We next present an algorithm for banded matrices. The banded matrices considered here are depicted schematically in **Figure 1**. In what follows, nq denotes the bandwidth including the diagonal elements.

1) *Computation of the modified Cholesky decomposition of a matrix together with its derivative with respect to an eigenvalue*

```

(1) Input data
    A : given coefficient band matrix, 2-dimension array as A(n,nq)
    b : work vector, 1-dimension array as b(n)
    n : given order of matrix A
    nq : given half band width of matrix A
    eps : parameter to check singularity of the matrix
(2) Output data
    A : L matrix and D matrix, 2-dimension array
    fd : differential of determinant
    ichg : numbers of minus element of diagonal matrix D (numbers of eigenvalue)
    ierr : error code
           =0, for normal execution
           =1, for singularity
(3)  $LDL^T$  decomposition
    ichg=0
    do i=1,n
        <d(i,i)>
        do j=max(1,i-nq+1),i-1
            A(i,nq)=A(i,nq)-A(j,nq)*A(i,nq+j-i)2
        end do
        if (A(i,nq)<0) ichg=ichg+1
        if (abs(A(i,nq))<eps) then
            ierr=1
            return
        end if
        <l(i,j)>
        do j=i+1,min(i+nq-1,n)
            aa=A(j,nq+i-j)

```

```

        do k=max(1,j-nq+1),i-1
            aa=aa- A(i,nq+k-i)*A(k,nq)*A(j,nq+k-j)
        end do
        A(j,nq+i-j)=aa/A(i,nq)
    end do
end do
ierr=0
(4) Derivative of a determinant with respect to an eigenvalue (fd)
fd=0
do i=1,n
<(i,i)>
    fd=fd-1/A(i,nq)
<(i,j)>
    do j=i+1,min(i+nq-1,n)
        b(j)=-A(j,nq-(j-i))
        do k=1,j-i-1
            b(j)=b(j)-A(j,nq-(j-i)+k)*b(i+k)
        end do
        fd=fd-b(j)2/A(j,nq)
    end do
    do j=i+nq,n
        b(j)=0
        do k=1,nq-1
            b(j)=b(j)-A(j,k)*b(j-nq+k)
        end do
        fd=fd-b(j)2/A(j,nq)
    end do
end do
2) Calculation of the solution
(1) Input data
A : given decomposed coefficient band matrix, 2-dimension array as A(n,nq)
b : given right hand side vector, 1-dimension array as b(n)
n : given order of matrix A and vector b
nq : given half band width of matrix A
(2) Output data
b : solution vector, 1-dimension array
(3) Forward substitution
do i=1,n
    do j=max(1,i-nq+1),i-1
        b(i)=b(i)-A(i,nq+j-i)*b(j)
    end do
end do
(4) Backward substitution
do i=1,n
    ii=n-i+1
    b(ii)=b(ii)/A(ii,nq)
    do j=ii+1,min(n,ii+nq-1)
        b(ii)=b(ii)-A(j,nq+ii-j)*b(j)
    end do
end do

```

4. Numerical Experiments

To demonstrate the effectiveness of the derivative of a determinant in the context of LDL^T decompositions in

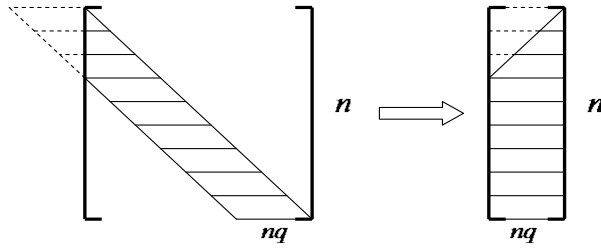


Figure 1. Banded matrix.

the arc-length method of nonlinear analysis, we conducted numerical experiments on a three-dimensional truss (Figure 1). We first describe the nonlinear FEM algorithms used in these numerical experiments.

In nonlinear FEM methods [6] [7], the matrix determinant vanishes at singular points, indicating the presence of a zero eigenvalue. The existence of a formula for the derivative of the determinant with respect to an eigenvalue makes it easy to identify singular points, using the fact that the sign of the derivative changes in the vicinity of a singular point. Within the context of the arc-length method, we apply a search technique (such as the bisection method) to narrow the interval within which the sign of f'/f changes and thus to pinpoint the location of the singular point. Of course, this calculation could also be performed by counting the number of negative elements of the diagonal matrix D arising from the LDL^T decomposition. Moreover, the solution obtained via Newton-Raphson varies as $1/(f'/f)$, and hence we may use the quantity $|1/(f'/f)|$ as an increment. The fact that the increment in the arc length (or other quantities) may thus be determined automatically allows us to reduce the number of basic parameters exerting a significant impact on the nonlinear finite-element method. However, in the numerical experiments that we have conducted so far, we have found that accurate determination of singular points, such as bifurcation points or limit points, requires, in addition to values of the quantity $|1/(f'/f)|$, the imposition of constraints on the maximum values of the strain and/or the relative strain. For example, if we are using steel, we impose a strain constraint. Choosing the smaller of $|1/(f'/f)|$ and this constraint value then allows the singular point to be determined accurately. Aggregating all the considerations discussed above, we arrive at the following arc-length algorithm for identifying singular points along the main pathway.

{1} step=0

(1) Configure or input data parameters (incremental convergence tolerance, maximum number of steps, maximum number of iterations at a single step, choice of elasticity or elastoplasticity analysis, number of subdivisions for the strain value constraint, threshold value for identifying singularity in the LDL^T decomposition, elasticity coefficients and plasticity parameters, node coordinates, characteristics of all elements, boundary conditions, etc.)

(2) Compute bandwidth

(3) Initialize arrays and other variables

(4) Configure or input external force vectors

{2} step=1,2,3,...

1) iteration=0

(5) Recall data from previous step (node coordinates, cross-sectional performance, displacement, strain, stress, f'/f , etc.).

(6) Compute tangent stiffness matrix.

2) iteration=1,2,3,...

(7) Compute the LDL^T decomposition (including the computation of f'/f and the number of negative elements of the diagonal matrix D) and the solution to the simultaneous linear equations.

(8) Choose the new arc length to be the smaller of the absolute value of the following two quantities: (a) the value obtained from the arc-length method and (b) $1/|f'/f|$. Within the iterative process, adjust as necessary to ensure that the maximum value of the strain satisfies the strain constraint. Compute the incremental strain and the total strain.

- (9) Use the constitutive equation to compute the material stiffness and the stress at arbitrary points.
- (10) Compute the tangent stiffness matrix.
- (11) Compute the residual (the extent to which the system is unbalanced) and assess convergence.
- (12) If not converged, return to step 2). If converged and the sign of f'/f has changed, use the bisection method to search for the singular point. Once the singular point has been identified with sufficient accuracy, confirm it by counting the number of negative entries of the diagonal matrix D obtained from the LDL^T decomposition; then proceed to step (2) unless the maximum number of steps has been taken, in which case stop the calculation.

Numerical Experiments

As shown in **Figure 2**, the three-dimensional truss we considered consists of 24 segments and 13 nodes and is symmetric in the xy plane. To ensure that the load results in a symmetric displacement, the load is applied in the downward vertical direction to nodes 1 - 7, with the load at node 1 being half the load at the other nodes. All numerical experiments were carried out in double-precision arithmetic using the algorithm described above. Computations were performed on an Intel(R) Core™ i7 3.2 GHz machine with 12 GB of RAM, running Windows 7 and gcc-4.7-20110723-64 Fortran. We analyzed three computational procedures: Equations (4) and (11) for dense matrices, and Equation (11) for banded matrices. All three procedures yielded identical results. We verified that our proposed formula allows accurate calculation of the quantity f'/f . In what follows, we will discuss results obtained for the banded-matrix case.

The following parameter values were used. The incremental convergence tolerance was $TOLER = 10^{-8}$. The maximum number of steps was $NSTEP = 2000$. The maximum number of iterations at a single step was $NITR = 30$. We used elastoplasticity analysis ($IEP = 1$). The threshold value for identifying singularity in the modified Cholesky decomposition was $EPS = 10^{-12}$. The elasticity coefficients and plasticity parameters were configured as follows: elasticity coefficient $E = 2.058 \times 10^5$ (N/mm²), initial cross-sectional area

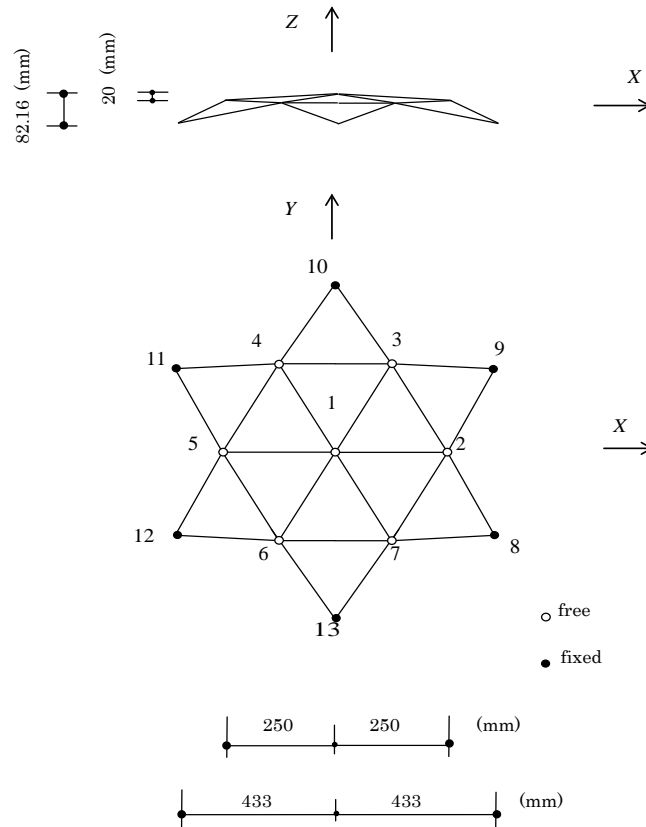


Figure 2. A three-dimensional truss structure model.

$A_0 = 1.0 \text{ (mm}^2\text{)}$, number of subdivisions for strain value constraint $MD = 250$, Poisson ratio $\nu = 0.3$ (in the elastic regime) or $\nu = 0.5$ (in the plastic regime), and yield stress $\sigma_y = 235.2 \text{ (N/mm}^2\text{)}$. To allow the stress to be computed directly from the strain, we adopted the Richard-Abbott model as the constitutive equation. The relationship between the stress σ and the strain ε is given by

$$\sigma = \frac{(E - E_p)\varepsilon}{\left(1 + \left|\frac{(E - E_p)\varepsilon}{\sigma_y}\right|^m\right)^{\frac{1}{m}}} + E_p\varepsilon, \quad (12)$$

$$E_t = \frac{d\sigma}{d\varepsilon} = \frac{E - E_p}{\left(1 + \left|\frac{(E - E_p)\varepsilon}{\sigma_y}\right|^m\right)^{\frac{m+1}{m}}} + E_p, \quad (13)$$

where E_p is the effective strain hardness, which is set to $E_p = 0.01E$ in our numerical experiments; m is a parameter that controls the degree of smoothness, which is set at $m = 18$, close to bilinear point; and E_t is the tangential stiffness at an arbitrary point [8].

The tangent stiffness matrix for the three-dimensional truss takes the form

$$K = \begin{bmatrix} K_0 & -K_0 \\ -K_0 & K_0 \end{bmatrix}, \quad (14)$$

where

$$K_0 = \left[\frac{N}{\ell} I + \frac{E_t A - (1 + 2\nu)N}{\ell} cc^T \right], \quad (15)$$

$$c = \begin{Bmatrix} c_x \\ c_y \\ c_z \end{Bmatrix}, \quad (16)$$

$$c_x = \frac{x_j - x_i}{\ell}, c_y = \frac{y_j - y_i}{\ell}, c_z = \frac{z_j - z_i}{\ell}. \quad (17)$$

Here N is the axial force, i and j are the indices of the nodes at the segment endpoints, (x_i, y_i) and (x_j, y_j) are the x, y coordinates of nodes i, j , and ℓ is the segment length.

Figure 3 plots the load-displacement curve obtained using the proposed method. **Figures 3** and **4** indicate the correct count of eigenvalues. A total of six eigenvalues appear before the limit point, with the sixth eigenvalue corresponding to the limit point itself. λ_1 through λ_5 are bifurcation buckling points, and λ_6 is the limit point. The pair (λ_2, λ_3) is a pair of multiple roots, as is the pair (λ_4, λ_5) . If the number MD of subdivisions for the strain value constraint is too small, discrepancies in the number of zero eigenvalues detected by the proposed method can arise, causing some singular points to be overlooked. For this reason, we have here chosen $MD = 250$, but high-precision nonlinear analyses require large numbers of steps. The method that we have proposed is a simple addition to the process of solving simultaneous linear equations and may be put to effective use in nonlinear analysis.

5. Conclusions

We have presented a formula for computing the derivative of a determinant with respect to an eigenvalue. Our computation proceeds simultaneously with the modified Cholesky (LDL^T) decomposition of the matrix, a characteristic example of a direct solution method. We applied our proposed formula to the determination of singular points, such as branch points and breaking points, in the arc-length method in a nonlinear finite-element

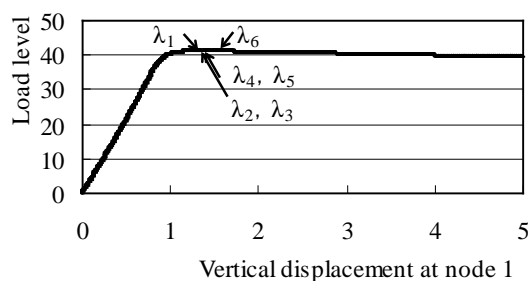


Figure 3. Vertical displacement and load level.

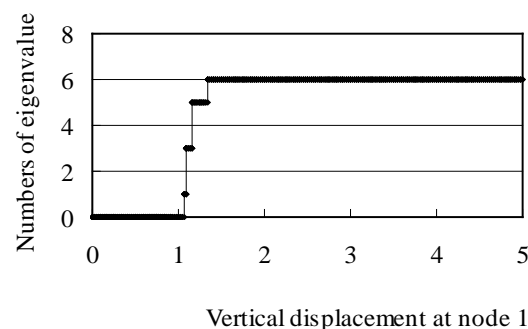


Figure 4. Vertical displacement and numbers of eigenvalue.

method. In this application, we detect changes in the sign of the derivative of the determinant and then use the bisection method to narrow the interval containing the singular point.

Moreover, because the solution obtained via the Newton-Raphson method varies with the derivative of the determinant, it is possible to use this quantity as the arc length. This then allows the arc-length increment to be determined automatically, which in turn allows a reduction in the number of basic parameters impacting the nonlinear analysis. However, as our numerical experiments demonstrated, when using the proposed method, it is necessary to impose a constraint on the absolute value of the maximum values of the strain or the relative strain, and to use the strain constraint to control the increment. The proposed method is designed to work with the LDL^T decomposition and exhibits a significant reduction in memory requirements when applied to the LDL^T decomposition of banded matrices instead of dense matrices.

Numerical experiments on a three-dimensional truss structure demonstrated that the proposed method is able to identify singular points (bifurcation points and limit points) accurately using the derivative with respect to an eigenvalue of the characteristic equation of the stiffness matrix. This method does not require the time-consuming step of solving the eigenvalue problem and makes use of the solution to the simultaneous linear equations arising in incremental analysis, thus making it an extremely effective technique.

References

- [1] Kashiwagi, M. (1999) An Eigensolution Method for Solving the Largest or Smallest Eigenpair by the Conjugate Gradient Method. *Transactions of JSCEs*, **1**, 1-5.
- [2] Kashiwagi, M. (2009) A Method for Determining Intermediate Eigensolution of Sparse and Symmetric Matrices by the Double Shifted Inverse Power Method. *Transactions of JSIAM*, **19**, 23-38.
- [3] Kashiwagi, M. (2008) A Method for Determining Eigensolutions of Large, Sparse, Symmetric Matrices by the Preconditioned Conjugate Gradient Method in the Generalized Eigenvalue Problem. *Journal of Structural Engineering*, **73**, 1209-1217.
- [4] Kashiwagi, M. (2011) A Method for Determining Intermediate Eigenpairs of Sparse Symmetric Matrices by Lanczos and Shifted Inverse Power Method in Generalized Eigenvalue Problems. *Journal of Structural and Construction Engineering: Transactions of AIJ*, **76**, 1181-1188. <http://dx.doi.org/10.3130/aijs.76.1181>
- [5] Kashiwagi, M. (2013) Derivative of a Determinant with Respect to an Eigenvalue in the LDU Decomposition of a

- Non-Symmetric Matrix. *Applied Mathematics*, **4**, 464-468. <http://dx.doi.org/10.4236/am.2013.43069>
- [6] Bathe, K.J. (1982) Finite Element Procedures in Engineering Analysis, Prentice-Hall, Englewood Cliffs, Nj.
- [7] Fujii, F. and Choong, K.K. (1992) Branch Switching in Bifurcation of Structures. *Journal of Engineering Mechanics*. **118**, 1578-1596. [http://dx.doi.org/10.1061/\(ASCE\)0733-9399\(1992\)118:8\(1578\)](http://dx.doi.org/10.1061/(ASCE)0733-9399(1992)118:8(1578))
- [8] Richrd, R.M. and Abotto, B.J. (1975) Versatile Elastic-Plastic Stress-Strain Formula. *Journal of Engineering Mechanics*. **101**, 511-515.

Bifurcations of Travelling Wave Solutions for the B(m,n) Equation

Minzhi Wei^{1*}, Yujian Gan¹, Shengqiang Tang²

¹School of Information and Statistics, Guangxi University of Finance and Economics, Nanning, China

²School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin, China

Email: *weiminzhi@21cn.com

Received 22 December 2013; revised 22 January 2014; accepted 29 January 2014

Copyright © 2014 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Using the bifurcation theory of dynamical systems to a class of nonlinear fourth order analogue of the B(m,n) equation, the existence of solitary wave solutions, periodic cusp wave solutions, compactons solutions, and uncountably infinite many smooth wave solutions are obtained. Under different parametric conditions, various sufficient conditions to guarantee the existence of the above solutions are given. Some exact explicit parametric representations of the above waves are determined.

Keywords

Solitary Wave Solution; Periodic Cusp Wave Solution; Periodic Wave Solution; Smoothness of Wave; B(m,n) Equation

1. Introduction

Recently, Song and Shao [1] employed bifurcation method of dynamical systems to investigate bifurcation of solitary waves of the following generalized (2 + 1)-dimensional Boussinesq equation

$$u_{tt} - \alpha u_{xx} - \beta u_{yy} - \gamma (u^2)_{xx} - \delta u_{xxxx} = 0, \quad (1.1)$$

where α, β, γ and δ are arbitrary constants with $\gamma\delta \neq 0$. Chen and Zhang [2] obtained some double periodic and multiple soliton solutions of Equation (1.1) by using the generalized Jacobi elliptic function method. Further, Li [3] studied the generalized Boussinesq equation:

$$\zeta_t + [(1 + \zeta_t)u]_x = au_{xxxx} \quad (1.2)$$

*Corresponding author.

by using bifurcation method. In this paper, we shall employ bifurcation method of dynamical systems [4]-[11] to investigate bifurcation of solitary waves of the following equation:

$$(u^m)_{tt} = (u^n)_{xx} + (u^n)_{yy} + a(u^2)_{xx} + \delta(u^n)_{xxxx}, \quad (1.3)$$

Numbers of solitary waves are given for each parameter condition. Under some parameter conditions, exact solitary wave solutions will be obtained. It is very important to consider the dynamical bifurcation behavior for the travelling wave solutions of (1.3). In this paper, we shall study all travelling wave solutions in the parameter space of this system. Let $u(x, y, t) = \phi(x + y - ct) = \phi(\xi)$, where c is the wave speed. Then (1.3) becomes to

$$c^2(\phi^m)'' = 2(\phi^n)'' + a(\phi^2)'' + b(u^n)^{(4)}, \quad (1.4)$$

where “'” is the derivative with respect to ξ . Integrating Equation (1.4) twice, using the constants of integration to be zero we find

$$q\phi^m + p\phi^n + \phi^2 + r[n(n-1)\phi^{n-2}(\phi')^2 + n\phi^{n-1}\phi''] = 0, \quad (1.5)$$

where $p = \frac{2}{a}, q = -\frac{c^2}{a}, r = \frac{b}{a}$. Equation (1.5) is equivalent to the two-dimensional systems as follows

$$\frac{d\phi}{d\xi} = y, \quad \frac{dy}{d\xi} = -\frac{q\phi^m + p\phi^n + \phi^2 + rn(n-1)\phi^{n-2}y^2}{rn\phi^{n-1}} \quad (1.6)$$

with the first integral

$$H(\phi, y) = \frac{1}{2}rn\phi^{2(n-1)} + \phi^{n+2} \left[\frac{q}{n+m}\phi^{m-2} + \phi^2 + \frac{p}{2n}\phi^{n-2} + \frac{1}{2+n} \right] = h. \quad (1.7)$$

System (1.6) is a 5-parameter planar dynamical system depending on the parameter group (m, n, p, q, r) . For different m, n and a fixed r , we shall investigate the bifurcations of phase portraits of System (1.6) in the phase plane (ϕ, y) as the parameters p, q are changed. Here we are considering a physical model where only bounded travelling waves are meaningful. So we only pay attention to the bounded solutions of System (1.6).

2. Bifurcations of Phase Portraits of (1.6)

In this section, we study all possible periodic annuluses defined by the vector fields of (1.6) when the parameters p, q are varied.

Let $d\xi = rn\phi^{n-1}d\zeta$, Then, except on the straight lines $\phi = 0$, the system (1.6) has the same topological phase portraits as the following system

$$\frac{d\phi}{d\zeta} = rn\phi^{n-1}, \quad \frac{dy}{d\zeta} = -[p\phi^n + \phi^2 + rn(n-1)\phi^{n-2}y^2] \quad (2.1)$$

Now, the straight lines $\phi = 0$ is an integral invariant straight line of (2.1).

Denote that

$$f(\phi) = 1 + q\phi^{m-2} + p\phi^{n-2}, \quad f'(\phi) = \phi^{n-3}[q(M-2)\phi^{m-n} + p(n-2)] \quad (2.2)$$

For $m - n = 2l$ ($l \in \mathbb{Z}^+$), $m - 1 = 2m_1 - 1$, $n - 1 = 2n_1 - 1$,

When $\phi = \phi_0 = \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{1}{m-n}}$, $f'(\pm\phi_0) = 0$.

We have $f(\pm\phi_0) = 1 + q \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{m-2}{m-n}} + p \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{n-2}{m-n}}$

and which imply respectively the relations in the (p, q) -parameter plane

$$L_a : q = -\frac{n-2}{m-2} p^{\frac{m-2}{n-2}} \left(\frac{m-n}{m-2} \right)^{\frac{m-n}{n-2}}, p > 0, q < 0,$$

$$L_b : q = \frac{n-2}{m-2} (-p)^{\frac{m-2}{n-2}} \left(\frac{m-n}{m-2} \right)^{\frac{m-n}{n-2}}, p < 0, q > 0,$$

For $m-n=2l$ ($l \in \mathbb{Z}^+$), $m-1=2m_1$, $n-1=2n_1-1$, when $\phi = \phi_0 = \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{1}{m-n}}$, $f'(\phi_0)=0$. We have

$f(\phi_0) = 1 + q \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{m-2}{m-n}} + p \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{n-2}{m-n}}$ and which imply respectively the relations in the (p, q) -parameter plane

$$L_b : q = \frac{n-2}{m-2} (-p)^{\frac{m-2}{n-2}} \left(\frac{m-n}{m-2} \right)^{\frac{m-n}{n-2}}, p < 0, q > 0,$$

$$L_c : q = -\frac{n-2}{m-2} (-p)^{\frac{m-2}{n-2}} \left(\frac{m-n}{m-2} \right)^{\frac{m-n}{n-2}}, p < 0, q < 0,$$

For $m-n=2l-1$ ($l \in \mathbb{Z}^+$), $m-1=2m_1-1$, $n-1=2n_1$, when $\phi = \phi_0 = \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{1}{m-n}}$, $f'(\phi_0)=0$. We have

$f(\phi_0) = 1 + q \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{m-2}{m-n}} + p \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{n-2}{m-n}}$, which imply respectively the relations in the (p, q) -parameter plane

$$L_d : q = -\frac{n-2}{m-2} p^{\frac{m-2}{n-2}} \left(\frac{m-n+4}{m-2} \right)^{\frac{m-n}{n-2}}.$$

For $m-n=2l-1$ ($l \in \mathbb{Z}^+$), $m-1=2m_1$, $n-1=2n_1$, when $\phi = \phi_0 = \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{1}{m-n}}$, $f'(\pm\phi_0)=0$. We have

$f(\phi_0) = 1 + q \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{m-2}{m-n}} + p \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{n-2}{m-n}}$ and $f(-\phi_0) = 1 - q \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{m-2}{m-n}} - p \left[-\frac{p(n-2)}{q(m-2)} \right]^{\frac{n-2}{m-n}}$,

which imply respectively the relations in the (p, q) -parameter plane

$$L_e : q = -\frac{n-2}{m-2} p^{\frac{m-2}{n-2}} \left(\frac{m-n}{m-2} \right)^{\frac{m-n}{n-2}}, pq < 0.$$

Let $M(\phi_e, y_e)$ be the coefficient matrix of the linearized system of (2.1) at an equilibrium point (ϕ_e, y_e) . Then, we have

$$J(\phi_i, 0) = \det(M(\phi_e, 0)) = rn\phi_i^{n-3} (q(m-2)\phi_i^{m-3} + p(n-2)\phi_i^{n-3}).$$

By the theory of planar dynamical systems, we know that for an equilibrium point of a planar integrable system, if $J < 0$ then the equilibrium point is a saddle point; if $J > 0$ and $\text{Trace}(M(\phi_e, y_e)) = 0$ then it is a center point; if $J > 0$ and $(\text{Trace}(M(\phi_e, y_e)))^2 - 4J(\phi_e, y_e) > 0$, then it is a node; if $J = 0$ and the index of the equilibrium point is 0 then it is a cusp, otherwise, it is a high order equilibrium point. For the function defined by (1.7), we denote that

$$h_i = H(\phi_i, 0) = \phi_i^{n+2} \left[\frac{m-2}{(n-2)(m+n)} + \frac{p(m-n)}{2n(m+n)} \phi_i^{n-2} \right], i = 1-4.$$

We next use the above statements to consider the bifurcations of the phase portraits of (2.1). In the (p, q) parameter plane, the curves partition it into 4 regions for $m-n=2l-1, m-n=2l$ shown in **Figure 1** (1-1), (1-2), (1-3), and (1-4), respectively.

1) The case $q \neq 0$, We use **Figure 2**, **Figure 3**, **Figure 4**, and **Figure 5** to show the bifurcations of the phase portraits of (2.1).

2) The case $q = 0$. We consider the system

$$\frac{d\phi}{d\zeta} = m\phi^{n-1}, \quad \frac{dy}{d\zeta} = -[p\phi^n + \phi^2 + m(n-1)\phi^{n-2}y^2] \quad (2.3)$$

with the first integral

$$H(\phi, y) = -\frac{1}{2}m\phi^{2(n-1)}y^2 - \phi^{n+2} \left[\frac{p}{2n}\phi^{n-2} + \frac{1}{2+n} \right] = h. \quad (2.4)$$

Figure 6 and **Figure 7** show respectively the phase portraits of (2.3) for $n = 2n_1$ and $n = 2n_1 + 1$.

3. Exact Explicit Parametric Representations of Traveling Wave Solutions of (1.6)

In this section, we give some exact explicit parametric representations of periodic cusp wave solutions.

1). Suppose that $n = 4, m = 6, r < 0, (p, q) \in A_4$, In this case, we have the phase portrait of (2.1) shown in **Figure 2**

(2-5). Corresponding to the orbit defined by $H(\phi, y) = 0$ to the equilibrium point $S_{\pm} \left(\pm \sqrt{\frac{-p + \sqrt{p^2 - 4q}}{2q}}, 0 \right)$, the

arch curve has the algebraic equation

$$y^2 = \frac{1}{4(-r)} \left[\phi^2 - \frac{-5p-5\sqrt{\frac{p^2-4q}{16-15}}}{2q} \right] \left[\phi^2 - \frac{5p-5\sqrt{\frac{p^2-4q}{16-15}}}{2q} \right]. \quad (3.1)$$

Thus, by using the first Equation of (1.6) and (3.1), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = \pm \sqrt{\frac{-5p-5\sqrt{\frac{p^2-4q}{16-15}}}{2q}} \text{cn}^{-1}(\Omega_1 \xi, k_1), \quad (3.2)$$

$$\text{where } \Omega_1 = \sqrt{\frac{5\sqrt{\frac{p^2-4q}{16-15}}}{4qr}}, \quad k_1 = \sqrt{\frac{\frac{p}{4} - \sqrt{\frac{p^2-4q}{16-15}}}{-2\sqrt{\frac{p^2-4q}{16-15}}}}.$$

We will show in Section 4 that (3.10) gives rise to two periodic cusp wave solutions of peak type and valley type of (1.3).

2). Suppose that $n = 2, m = 4, r > 0, (p, q) \in A_3$, In this case, we have the phase portrait of (2.1) shown in **Figure 2** (2-4). corresponding to the orbit defined by $H(\phi, y) = 0$ to the equilibrium point $A(0, 0)$, the arch curve has the algebraic equation

$$y^2 = \frac{1}{r} \phi^2 \left(\frac{q}{6} \phi^2 + \frac{p+1}{4} \right), \quad (3.3)$$

Thus, by using the first equation of (1.6) and (3.3), we obtain the parametric representation of this arch as follows:

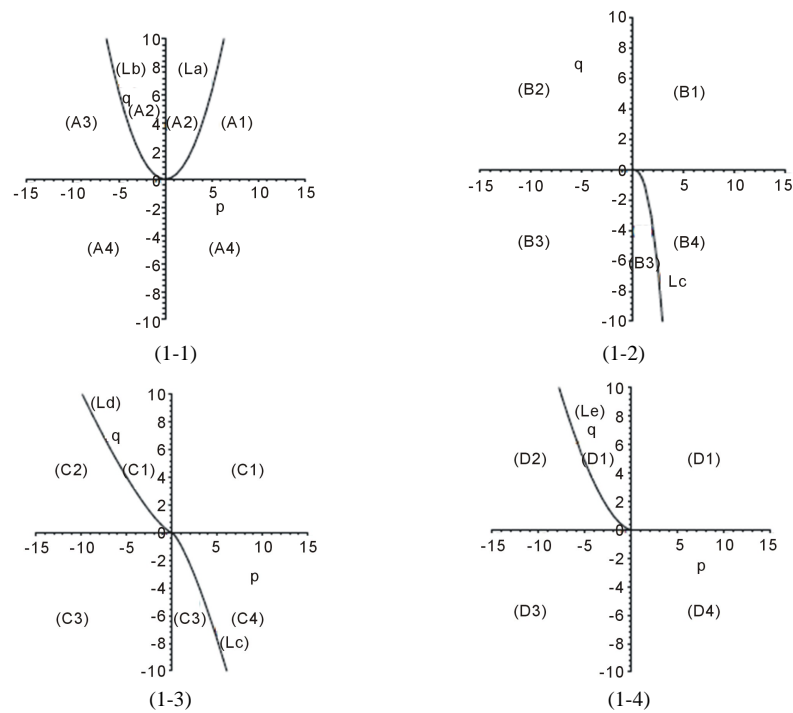


Figure 1. (1-1) $m - n = 2l$, $n = 2n_1$; (1-2) $m - n = 2l - 1$, $n = 2n_1$; (1-3) $m - n = 2l$, $n = 2n_1 + 1$; (1-4) $m - n = 2l - 1$, $n = 2n_1 + 1$.

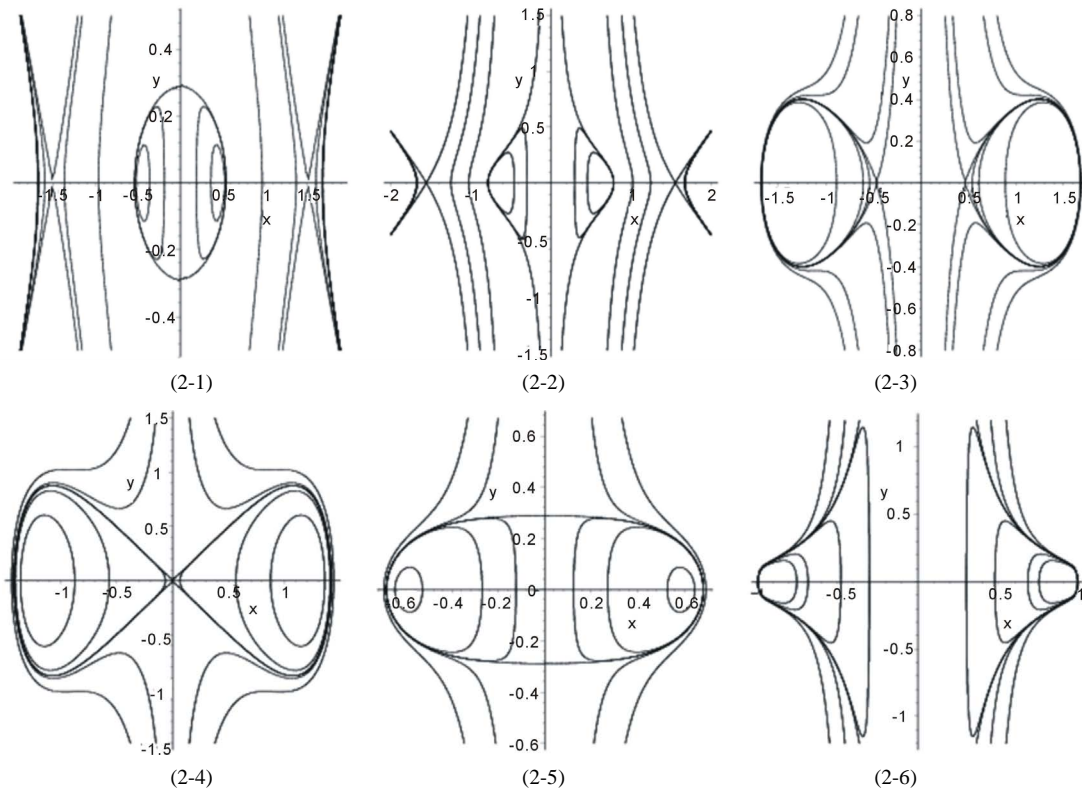


Figure 2. The phase portraits of (1.6) for $m - n = 2l$, $n = 2n_1$, $l, n_1 \in \mathbb{Z}^+$. (2-1) $r < 0$, $n_1 = 2$, $(p, q) \in (A_3)$; (2-2) $r < 0$, $n_1 \geq 2$, $(p, q) \in (A_3)$; (2-3) $r > 0$, $n_1 \geq 2$, $(p, q) \in (A_3)$; (2-4) $r > 0$, $n_1 = 1$, $(p, q) \in (A_3)$; (2-5) $r < 0$, $n_1 = 2$, $(p, q) \in (A_4)$; (2-6) $r < 0$, $n_1 \geq 3$, $(p, q) \in (A_3)$.

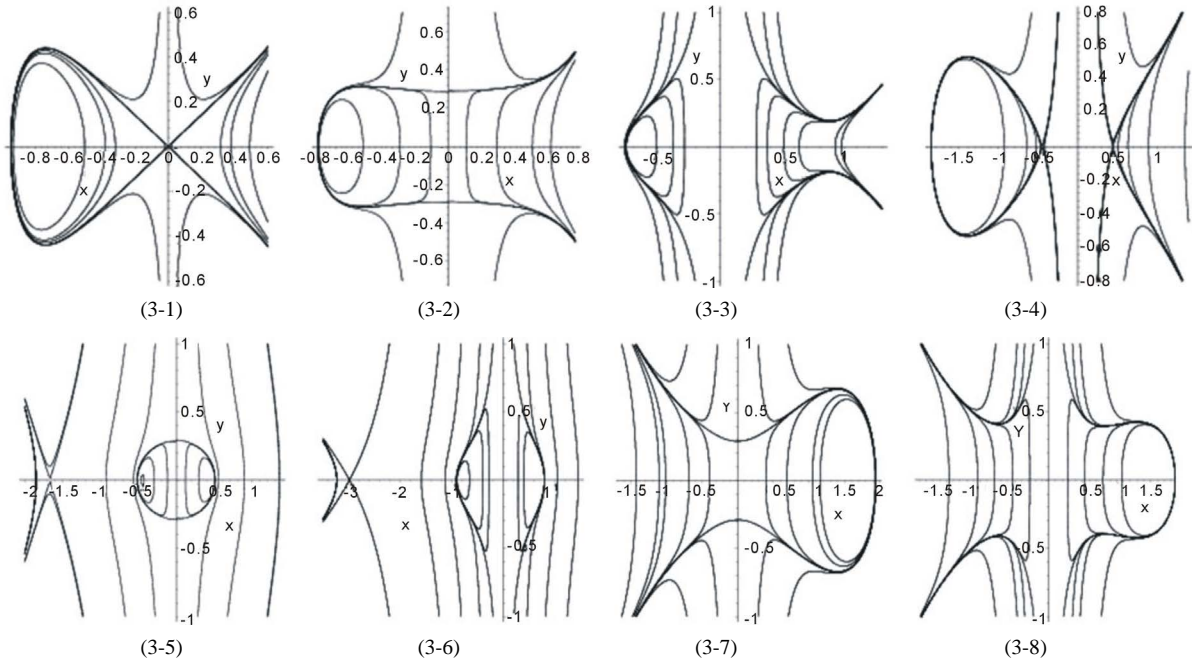


Figure 3. The phase portraits of (1.6) for $m - n = 2l - 1$, $n = 2n_1$, $l, n_1 \in \mathbb{Z}^+$. (3-1) $r < 0$, $n_1 = 1$, $(p, q) \in (B_1)$; (3-2) $r < 0$, $n_1 = 2$, $(p, q) \in (B_1 \cup B_2)$; (3-3) $r > 0$, $n_1 \geq 2$, $(p, q) \in (B_1) \cup (B_2)$; (3-4) $r > 0$, $n_1 \geq 2$, $(p, q) \in (B_3)$; (3-5) $r < 0$, $n_1 = 2$, $(p, q) \in (B_3)$; (3-6) $r < 0$, $n_1 \geq 2$, $(p, q) \in (B_1 \cup B_2)$; (3-7) $r < 0$, $n_1 = 2$, $(p, q) \in (B_4) \cup (B_2)$; (3-8) $r > 0$, $n_1 \geq 2$, $(p, q) \in (B_4)$.

$$\phi(\xi) = \sqrt{\frac{2q}{3(p+1)}} \sin^2 \left(\sqrt{\frac{q}{6r}} \xi \right), \quad (3.4)$$

We will show in Section 4 that (3.10) gives rise to a solitary wave solutions of peak type and valley type of (1.3).

3). Suppose that $n = 3, m = 5, r < 0, (p, q) \in C_2$. In this case, we have the phase portrait of (2.1) shown in **Figure 4** (4-5). corresponding to the orbit defined by $H(\phi, y) = 0$ to the equilibrium point $A(0, 0)$, the arch curve has the algebraic equation

$$y^2 = \frac{2}{3(-r)} (\phi - \phi_1)(\phi - \phi_2)(\phi_3 - \phi)(\phi_4 - \phi), \quad (3.5)$$

where $\phi_1 < \phi_2 < \phi_3 < \phi_4$, $\phi_i \left(\frac{q}{8} \phi_i^3 + \frac{p}{6} \phi_i + \frac{1}{5} \right) = 0, i = 1 - 4$.

Thus, by using the first equation of (1.6) and (3.5), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = \frac{(\phi_4 - \phi_1)\phi_2 - \phi_1(\phi_4 - \phi_2)sn^2(\Omega_2\xi; k_2)}{(\phi_4 - \phi_1) - (\phi_4 - \phi_2)sn^2(\Omega_2\xi; k_2)}, \quad (3.6)$$

where $sn(x; k)$ is the Jacobin elliptic functions with the modulo k ,

$$\Omega_2 = \sqrt{\frac{(\phi_3 - \phi_2)(\phi_3 - \phi_1)}{-6r}}, \quad k_2 = \sqrt{\frac{(\phi_3 - \phi_2)(\phi_4 - \phi_1)}{(\phi_4 - \phi_2)(\phi_3 - \phi_1)}},$$

We will show in Section 4 that (3.6) gives rise to a smooth compacton solution of (1.3).

4). Suppose that $n = 2, m = 3, r < 0, (p, q) \in B_1$. In this case, we have the phase portrait of (2.1) shown in **Figure 3** (3-1), corresponding to the orbit defined by $H(\phi, y) = 0$ to the equilibrium point $A(0, 0)$, the arch curve has the algebraic equation

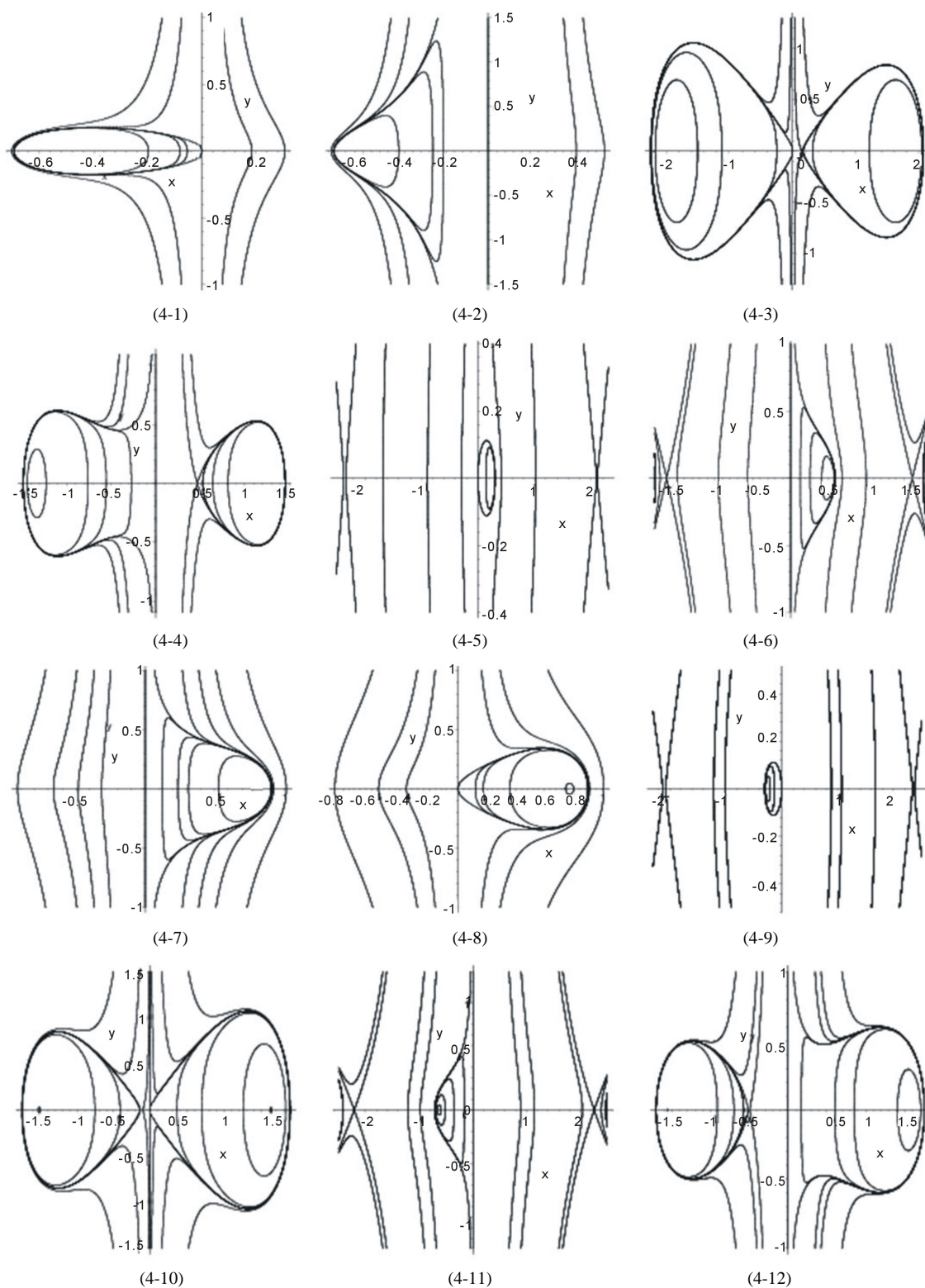


Figure 4. (4-1) $r > 0, n_1 = 1, (p, q) \in (C_1)$; (4-2) $r > 0, n_1 \geq 1, (p, q) \in (C_2)$; (4-3) $r > 0, n_1 = 1, (p, q) \in (C_2)$; (4-4) $r > 0, n_1 \geq 2, (p, q) \in (C_2)$; (4-5) $r < 0, n_1 = 1, (p, q) \in (C_2)$; (4-6) $r < 0, n_1 \geq 2, (p, q) \in (C_2)$; (4-7) $r > 0, n_1 \geq 2, (p, q) \in (C_3)$; (4-8) $r < 0, n_1 = 1, (p, q) \in (C_3)$; (4-9) $r > 0, n_1 = 1, (p, q) \in (C_4)$; (4-10) $r < 0, n_1 = 1, (p, q) \in (C_4)$; (4-11) $r > 0, n_1 \geq 2, (p, q) \in (C_4)$; (4-12) $r < 0, n_1 \geq 2, (p, q) \in (C_4)$.

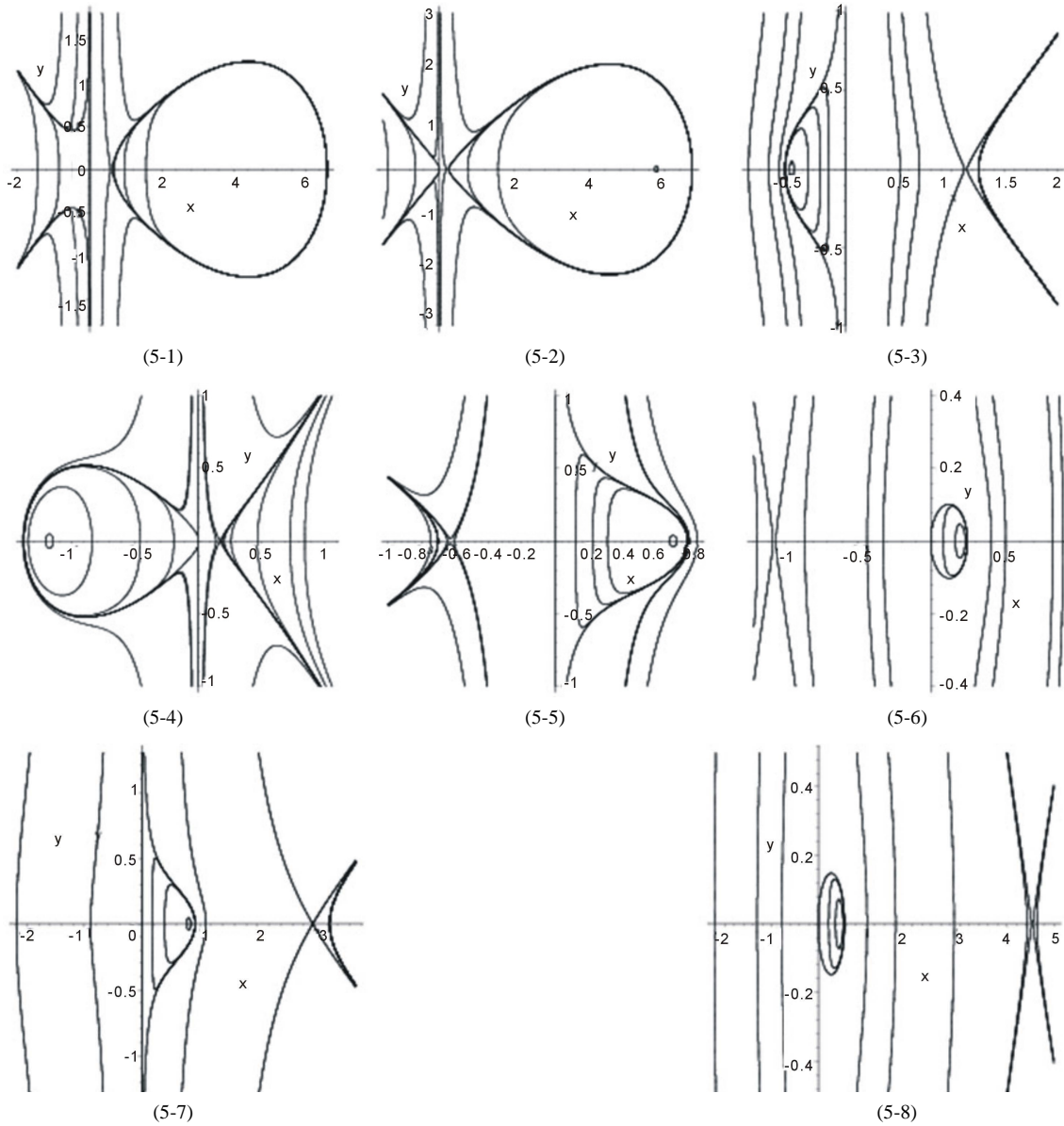


Figure 5. The phase portraits of (1.6) for $m - n = 2l - 1$, $n = 2n_1 + 1$, $l, n_1 \in \mathbb{Z}^+$ (5-1) $r > 0$, $n_1 \geq 2$, $(p, q) \in (D_2)$; (5-2) $r > 0$, $n_1 = 1$, $(p, q) \in (D_2)$; (5-3) $r > 0$, $n_1 \geq 2$, $(p, q) \in (D_3) \cup (D_4)$; (5-4) $r > 0$, $n_1 = 1$, $(p, q) \in (D_3) \cup (D_4)$; (5-5) $r < 0$, $n_1 \geq 2$, $(p, q) \in (D_3) \cup (D_4)$; (5-6) $r < 0$, $n_1 = 1$, $(p, q) \in (D_3) \cup (D_4)$; (5-7) $r < 0$, $n_1 \geq 2$, $(p, q) \in (D_2)$; (5-8) $r < 0$, $n_1 = 1$, $(p, q) \in (D_2)$.

$$y^2 = \frac{q}{-5r} \phi^2 \left(\phi + \frac{5(p+1)}{4q} \right), \quad (3.7)$$

Thus, by using the first equation of (1.6) and (3.7), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = \frac{-\frac{5(p+1)}{4q}}{1 - \tanh^2 \left(\sqrt{-\frac{4q}{5r}} \xi \right)}. \quad (3.8)$$

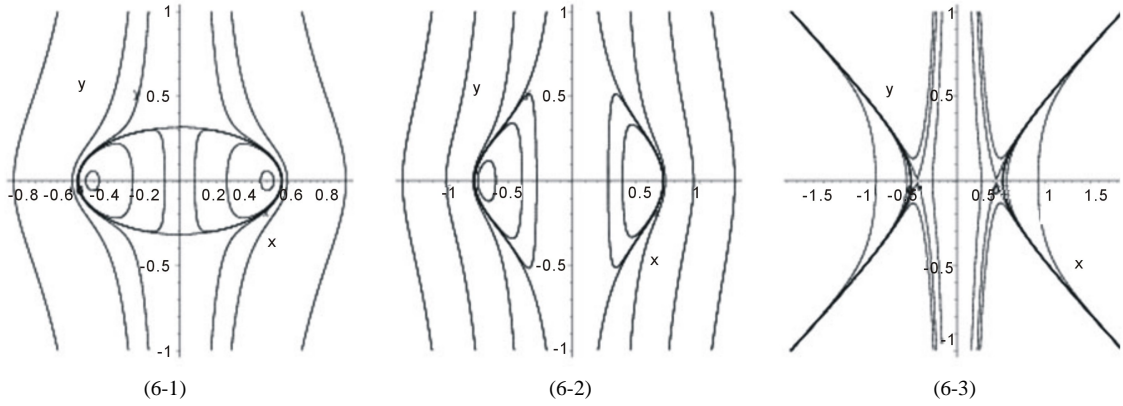


Figure 6. The phase portraits of (1.6) for $n = 2n_1$, $n_1 \in \mathbb{Z}^+$. (6-1) $r < 0$, $n_1 = 2$, $m_1 \geq n_1$, $p < 0$; (6-2) $r < 0$, $n_1 \geq 2$, $m_1 > n_1$, $p < 0$; (6-3) $r > 0$, $m_1 > n_1$, $p < 0$.

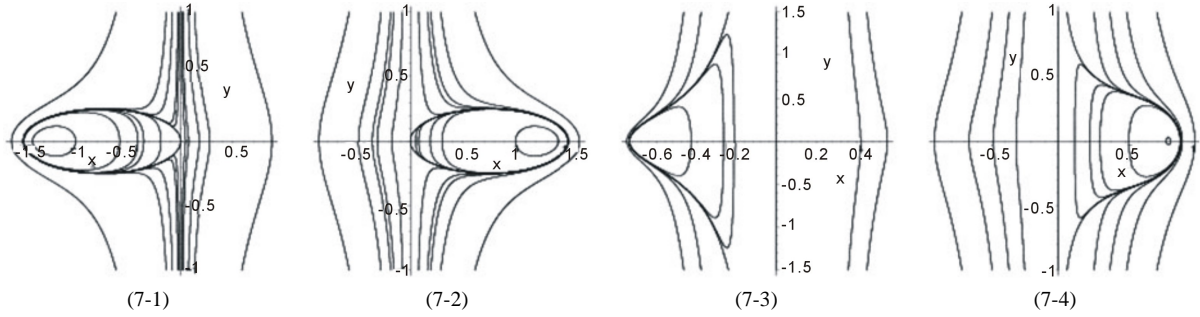


Figure 7. The phase portraits of (1.6) for $n = 2n_1 + 1$, $n_1 \in \mathbb{Z}^+$. (7-1) $r > 0$, $n_1 = 1$, $m_1 \geq n_1$, $p > 0$; (7-2) $r < 0$, $n_1 = 1$, $m_1 > n_1$, $p < 0$; (7-3) $r < 0$, $n_1 \geq 2$, $m_1 > n_1$, $p > 0$; (7-4) $r < 0$, $n_1 \geq 2$, $m_1 > n_1$, $p < 0$.

We will show in Section 4 that (3.8) gives rise to a solitary wave solution of peak type or valley type of (1.3).

5). Suppose that $n = 3, m = 4, r > 0, (p, q) \in D_3 \cup D_4$, in this case, we have the phase portrait of (2.1) shown in Figure 5 (5-4). corresponding to the orbit defined by $H(\phi, y) = 0$ to the equilibrium point

$S_{\pm} \left(\pm \sqrt{\frac{-p + \sqrt{p^2 - 4q}}{2q}}, 0 \right)$, the arch curve has the algebraic equation

$$y^2 = \frac{2}{3r}(\phi - 0) \left(\phi - \left(-\frac{p}{12} + \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} \right) \right) \left(-\frac{p}{12} - \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} - \phi \right). \quad (3.9)$$

Thus, by using the first equation of (1.6) and (3.9), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = -\frac{7p}{12q} - \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} + \left(\frac{7p}{12q} + \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} \right) \text{sn}^2(\Omega_3 \xi; k_3), \quad (3.10)$$

where $\text{sn}(x; k)$ is the Jacobin elliptic functions with the modulo k and

$$\Omega_3 = \sqrt{\frac{-7 \sqrt{\frac{p^2 - 4q}{36 - 35}}}{6qr}}, \quad k_3 = \sqrt{\frac{1}{12 \sqrt{\frac{p^2 - 4q}{36 - 35}}} + \frac{1}{2}},$$

We will show in Section 4 that (3.10) gives rise to a smooth compacton solution of (1.3).

6). Suppose that $n=3, m=4, r<0, (p, q) \in D_3 \cup D_4$, In this case, we have the phase portrait of (2.1) shown in **Figure 5** (5-6), corresponding to the orbit defined by $H(\phi, y)=0$ to the equilibrium point

$S_{\pm} \left(\pm \sqrt{\frac{-p \pm \sqrt{p^2 - 4q}}{2q}}, 0 \right)$, the arch curve has the algebraic equation

$$y^2 = \frac{2}{-3r} (0 - \phi) \left(\phi - \left(-\frac{p}{12} + \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} \right) \right) \left(-\frac{p}{12} - \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} - \phi \right). \quad (3.11)$$

Thus, by using the first equation of (1.6) and (3.11), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = -\frac{7p}{12q} - \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} + \frac{\frac{7}{q} \sqrt{\frac{p^2 - 4q}{36 - 35}} \left(\frac{7p}{12q} - \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} \right)}{\left(\frac{7p}{12q} + \frac{7}{2q} \sqrt{\frac{p^2 - 4q}{36 - 35}} \right) \text{sn}^2(\Omega_4 \xi; k_4)}, \quad (3.12)$$

$$\text{where } \Omega_4 = \sqrt{\frac{-7 \sqrt{\frac{p^2 - 4q}{36 - 35}}}{6qr}}, \quad k_4 = \sqrt{-\frac{1}{12 \sqrt{\frac{p^2 - 4q}{36 - 35}}} + \frac{1}{2}},$$

we will show in Section 4 that (3.20) gives rise to a smooth compacton solution of (1.3)

7). Suppose $n=4, m=5, r<0, (p, q) \in B_1 \cup B_2 \cup B_4$ that. In this case, we have the phase portrait of (2.1) shown in **Figure 3** (3-2) and (3-7), corresponding to the orbit defined by $H(\phi, y)=0$ to the equilibrium point $A(0,0)$, the arch curve has the algebraic equation

$$y^2 = \frac{1}{-2r} \left(\frac{q}{9} \phi^3 + \frac{p}{8} \phi^2 + \frac{1}{6} \right). \quad (3.13)$$

Thus, by using the first equation of (1.6) and (3.13), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = \frac{3p}{-8q} + \wp \left(\sqrt{-\frac{q}{72r}} \xi, g_1, g_2 \right), \quad (3.14)$$

where $g_1 = \frac{27p^2}{16q^2}, g_2 = \frac{27p^3}{128q^3} - \frac{6}{q}$. We will show in Section 4 that (3.14) gives rise to a smooth compacton solution of (1.3).

8). Suppose $n=4, m=6, r<0, (p, q) \in A_3$. In this case, we have the phase portrait of (2.1) shown in **Figure 2** (2-1), corresponding to the orbit defined by $H(\phi, y)=0$ to the equilibrium point $A(0,0)$, the arch curve has the algebraic equation

$$y^2 = \frac{1}{-4r} \left(\phi^2 + \frac{5p}{8q} + \frac{5}{2q} \sqrt{\frac{p^2 - 4q}{16 - 15}} \right) \left(\phi^2 + \frac{5p}{8q} - \frac{5}{2q} \sqrt{\frac{p^2 - 4q}{16 - 15}} \right). \quad (3.15)$$

Thus, by using the first equation of (1.6) and (3.15), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = \pm \frac{\sqrt{-\frac{5p}{8q} + \frac{5}{2q} \sqrt{\frac{p^2 - 4q}{16 - 15}}}}{\text{cn}(\Omega_5 \xi; k_5)}, \quad (3.16)$$

$$\text{where } \Omega_5 = \sqrt{\frac{5\sqrt{\frac{p^2}{16} - \frac{4q}{15}}}{-4qr}}, \quad k_4 = \sqrt{\frac{p}{8\sqrt{\frac{p^2}{16} - \frac{4q}{15}}} + \frac{1}{2}},$$

We will show in Section 4 that (3.16) gives rise to two periodic cusp wave solutions of peak type and valley type of (1.3).

9). Suppose $n=3, m=5, r>0, (p, q) \in C_4$. In this case, we have the phase portrait of (2.1) shown in **Figure 4** (4-5), corresponding to the orbit defined by $H(\phi, y)=0$ to the equilibrium point $A(0,0)$, the arch curve has the algebraic equation

$$y^2 = \frac{2}{3r}(\phi - \phi_1)(\phi - \phi_2)(\phi - \phi_3)(\phi_4 - \phi), \quad (3.17)$$

where $\phi_1 < \phi_2 < \phi_3 < \phi_4$, $\phi_i \left(-\frac{q}{8}\phi_i^3 - \frac{p}{6}\phi_i - \frac{1}{5} \right) = 0, i=1-4$. Thus, by using the first equation of (1.6) and (3.17), we obtain the parametric representation of this arch as follows:

$$\phi(\xi) = \frac{(\phi_4 - \phi_3)\phi_2 - \phi_3(\phi_4 - \phi_2)sn^2(\Omega_6\xi; k_6)}{(\phi_4 - \phi_2) - (\phi_4 - \phi_3)sn^2(\Omega_6\xi; k_6)}, \quad (3.18)$$

where $sn(x; k)$ is the Jacobin elliptic functions with the modulo k and

$$\Omega_6 = \sqrt{\frac{(\phi_4 - \phi_2)(\phi_3 - \phi_1)}{6r}}, \quad k_6 = \sqrt{\frac{(\phi_4 - \phi_3)(\phi_2 - \phi_1)}{(\phi_3 - \phi_2)(\phi_3 - \phi_1)}},$$

We will show in Section 4 that (3.6) gives rise to a smooth compacton solution of (1.3).

4. The Existence of Smooth and Non-Smooth Travelling Wave Solutions of (1.6)

In this section, we use the results of Section 2 to discuss the existence of smooth and non-smooth solitary wave and periodic wave solutions. We first consider the existence of smooth solitary wave solution and periodic wave solutions.

Theorem 4.1

1). Suppose that $m-n=2, n=2n_1+1 \geq 5, l, n_1 \in \mathbb{Z}^+, r>0, (p, q) \in C_2$: Then, corresponding to a branch of the curves $H(\phi, y)=h_2(h_3)$ defined by (1.7), Equation (1.3) has a smooth solitary wave solution of peak type, corresponding to a branch of the curves $H(\phi, y)=h, h \in (h_3, h_2)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see **Figure 4** (4-4)).

2). Suppose that $m-n=2, n=3, m \in \mathbb{Z}^+, r>0, (p, q) \in C_2$: Then, corresponding to a branch of the curves $H(\phi, y)=h_2$ defined by (1.7), equation (1.3) has a smooth solitary wave solution of peak type, corresponding to a branch of the curves $H(\phi, y)=h, h \in (h_3, h_2)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see **Figure 4** (4-3)).

3). Suppose that $m-n=2, n=2n_1+1 \geq 5, l, n_1 \in \mathbb{Z}^+, r>0, (p, q) \in A_2$, Then, corresponding to a branch of the curves $H(\phi, y)=0$ defined by (1.7), equation (1.3) has a smooth solitary wave solution of peak type, corresponding to a branch of the curves $H(\phi, y)=h, h \in (h_1, h_2)$ defined by (1.7), equation (1.3) has a smooth family of periodic wave solutions (see **Figure 4** (4-12)).

4). Suppose that $m-n=2l, n=3, m \in \mathbb{Z}^+, r>0, (p, q) \in C_2$, Then, corresponding to a branch of the curves $H(\phi, y)=0$ defined by (1.7), equation (1.3) has a smooth solitary wave solution of valley type, corresponding to a branch of the curves $H(\phi, y)=h, h \in (h_1, h_3)$ defined by (1.7), equation (1.3) has a smooth family of periodic wave solutions (see **Figure 4** (4-3)).

5). Suppose that $m - n = 2l, n \geq 4, m \in \mathbb{Z}^+, r > 0, (p, q) \in A_2$, Then, corresponding to a branch of the curves $H(\phi, y) = h_2$ defined by (1.7), equation (1.3) has a smooth solitary wave solution of peak type, corresponding to a branch of the curves $H(\phi, y) = h, h \in (h_3, h_4)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see [Figure 2](#) (2-3)).

6). Suppose that $m - n = 2l, n \geq 4, m \in \mathbb{Z}^+, r > 0, (p, q) \in A_2$, Then, corresponding to a branch of the curves $H(\phi, y) = h_3$ defined by (1.7), equation (1.3) has a smooth solitary wave solution of peak type, corresponding to a branch of the curves $H(\phi, y) = h, h \in (h_1, h_2)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see [Figure 2](#) (2-3)).

7). Suppose that $m - n = 2l, n = 3, m \in \mathbb{Z}^+, r > 0, (p, q) \in C_3$, Then, corresponding to a branch of the curves $H(\phi, y) = 0$ defined by (1.7), Equation (1.3) has a smooth solitary wave solution of valley type, corresponding to a branch of the curves $H(\phi, y) = h, h \in (h_1, h_2)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see [Figure 2](#) (2-3)).

8). Suppose that $m - n = 2l - 1, n = 4, l \in \mathbb{Z}^+, r > 0, (p, q) \in B_3$, Then, corresponding to a branch of the curves $H(\phi, y) = h, h \in (h_3, h_2)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see [Figure 3](#) (3-5)).

9). Suppose that: $m - n = 2l - 1, n \geq 4, l \in \mathbb{Z}^+, r > 0, (p, q) \in B_3$, Then, corresponding to a branch of the curves $H(\phi, y) = h_2$ defined by (1.7), equation (1.3) has a smooth solitary wave solution of valley type, corresponding to a branch of the curves $H(\phi, y) = h, h \in (h_1, h_2)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see [Figure 3](#) (3-4)).

10). Suppose that $m - n = 2l, n = 2, l \in \mathbb{Z}^+, r < 0, (p, q) \in B_1$, Then, corresponding to a branch of the curves $H(\phi, y) = 0$ defined by (1.7), equation (1.3) has a smooth solitary wave solution of valley type, corresponding to a branch of the curves $H(\phi, y) = h, h \in (0, h_1)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see [Figure 2](#) (2-4)).

11). Suppose that $m - n = 2l - 1, n = 2, l \in \mathbb{Z}^+, r < 0, (p, q) \in B_1$: Then, corresponding to a branch of the curves $H(\phi, y) = 0$ defined by (1.7), Equation (1.3) has a smooth solitary wave solution of valley type, corresponding to a branch of the curves $H(\phi, y) = h, h \in (0, h_1)$ defined by (1.7), Equation (1.3) has a smooth family of periodic wave solutions (see [Figure 3](#) (3-1)).

We shall describe what types of non-smooth solitary wave and periodic wave solutions can appear for our system (1.6) which correspond to some orbits of (2.1) near the straight line $\phi = 0$. To discuss the existence of cusp waves, we need to use the following lemma relating to the singular straight line.

Lemma 4.2 The boundary curves of a periodic annulus are the limit curves of closed orbits inside the annulus; If these boundary curves contain a segment of the singular straight line $\phi = 0$ of (1.4), then along this segment and near this segment, in very short time interval $y = \phi_\xi$ jumps rapidly.

Base on Lemma 4.2, [Figure 2](#), and [Figure 3](#), we have the following result.

Theorem 4.3

1). Suppose that $m - n = 2l, n = 4, l \in \mathbb{Z}^+$.

a). For $r < 0, (p, q) \in A_4$ corresponding to the arch curve $H(\phi, y) = 0$ defined by (1.7), Equation (1.3) has two periodic cusp wave solutions; corresponding to two branches of the curves $H(\phi, y) = h, h \in (h_1, 0)$ defined by (1.7), Equation (1.3) has two families of periodic wave solutions. When h varies from h_1 to 0, these periodic travelling waves will gradually lose their smoothness, and evolve from smooth periodic travelling waves to periodic cusp travelling waves, finally approach a periodic cusp wave of valley type and a periodic cusp wave of peak type defined by $H(\phi, y) = 0$ of (1.7) (see [Figure 2](#) (2-5)).

b). For $r > 0, (p, q) \in A_3$ corresponding to the arch curve $H(\phi, y) = h, h \in (0, h_2)$ defined by (1.7), Equation (1.3) has two periodic cusp wave solutions; corresponding to two branches of the curves $H(\phi, y) = h, h \in (h_1, 0)$ defined by (1.7), Equation (1.3) has two families of periodic wave solutions. When h varies from h_1 to 0, these periodic travelling waves will gradually lose their smoothness, and evolve from smooth periodic travelling waves to periodic cusp travelling waves, finally approach a periodic cusp wave of valley type and a periodic cusp wave of peak type defined by $H(\phi, y) = 0$ of (1.7) (see [Figure 2](#) (2-1)).

2). Suppose that $m - n = 2l - 1, n = 4, l \in \mathbb{Z}^+$.

c). For $r < 0, (p, q) \in B_4$ corresponding to the arch curve $H(\phi, y) = 0$ defined by (1.7), Equation (1.3) has two periodic cusp wave solutions; corresponding to two branches of the curves $H(\phi, y) = h, h \in (0, h_1)$ defined by (1.7), Equation (1.3) has two families of periodic wave solutions. When h varies from 0 to h_1 , these periodic travelling waves will gradually lose their smoothness, and evolve from smooth periodic travelling waves to periodic cusp travelling waves, finally approach a periodic cusp wave of valley type and a periodic cusp wave of peak type defined by $H(\phi, y) = 0$ of (1.7) (see [Figure 3](#) (3-7)).

d). For $r < 0, (p, q) \in B_3$ corresponding to the arch curve $H(\phi, y) = 0$ defined by (1.7), Equation (1.3) has two periodic cusp wave solutions; corresponding to two branches of the curves $H(\phi, y) = h, h \in (0, h_3)$ defined by (1.7), Equation (1.3) has two families of periodic wave solutions. When h varies from 0 to h_3 , these periodic travelling waves will gradually lose their smoothness, and evolve from smooth periodic travelling waves to periodic cusp travelling waves, finally approach a periodic cusp wave of valley type and a periodic cusp wave of peak type defined by $H(\phi, y) = 0$ of (1.7) (see [Figure 3](#) (3-5)).

d). For $r < 0, (p, q) \in B_1 \cup B_2$ corresponding to the arch curve $H(\phi, y) = 0$ defined by (1.7), Equation (1.3) has two periodic cusp wave solutions; corresponding to two branches of the curves $H(\phi, y) = h, h \in (0, h_1)$ defined by (1.7), Equation (1.3) has two families of periodic wave solutions. When h varies from 0 to h_1 , these periodic travelling waves will gradually lose their smoothness, and evolve from smooth periodic travelling waves to periodic cusp travelling waves, finally approach a periodic cusp wave of valley type and a periodic cusp wave of peak type defined by $H(\phi, y) = 0$ of (1.7) (see [Figure 3](#) (3-3)).

We can easily see that there exist two families of closed orbits of (1.3) in [Figure 2](#) (2-6), [Figure 3](#) (3-6) and in [Figure 6](#) (6-2). There is one family of closed orbits in [Figure 3](#) (3-3), (3-8), [Figure 4](#) (4-2), (4-5) - (4-7), (4-9), (4-11) and in [Figure 5](#) (5-3), (5-5) - (5-7) and in [Figure 7](#) (7-3), (7-4). In all the above cases there exists at least one family of closed orbits (1.3) for which as h from $H(\phi_e, 0)$ to 0, where ϕ_e is the abscissa of the center, the closed orbit will expand outwards to approach the straight line $\phi = 0$ and $|y| = |\phi'|$ will approach to ∞ . As a result, we have the following conclusions.

Theorem 4.4

1). Suppose that $m - n = 2l, n = 2n_1 + 1, l, n_1 \in \mathbb{Z}^+$.

a). If $r < 0, n_1 \geq 1, (p, q) \in C_2$; then when $h \in (h_1, 0)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; where h varies from h_1 to 0, these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see [Figure 4](#) (4-2)).

b). If $r < 0, n_1 \geq 2, (p, q) \in C_3$; then when $h \in (0, h_1)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_1 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see [Figure 4](#) (4-7)).

c). If $r < 0, n_1 \geq 2, (p, q) \in C_2$; then when $h \in (0, h_2)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_2 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see [Figure 4](#) (4-6)).

d). If $r > 0, n_1 \geq 2, (p, q) \in C_4$; then when $h \in (h_2, 0)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from h_2 to 0, these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 4 (4-11)).

e). If $r < 0, n_1 = 1, (p, q) \in C_2$; then when $h \in (0, h_2)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_2 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 4 (4-5)).

f). If $r > 0, n_1 = 1, (p, q) \in C_4$; then when $h \in (0, h_2)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_2 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 4 (4-9)).

2). Suppose that, then when $h \in (0, h_1)$ in (1.7), Equation (1.3) has two family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_1 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 2 (2-6)). Paralleling to Figure 2 (2-6), we can see the periodic travelling wave solutions implied in Figure 3 (3-6) and Figure 6 (6-2) have the same characters.

3). Suppose that $m - n = 2l - 1, n = 2n_1 + 1, l, n_1 \in \mathbb{Z}^+$.

g). If $r < 0, n_1 = 1, (p, q) \in D_2 \cup D_4$; then when $h \in (0, h_2)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_2 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 5 (5-6)).

h). If $r < 0, n_1 \geq 2, (p, q) \in D_2$; then when $h \in (0, h_1)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_1 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 5 (5-7)).

i). If $r < 0, n_1 \geq 2, (p, q) \in D_3 \cup D_4$, then when $h \in (h_1, 0)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from h_1 to 0, these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 5 (5-3)).

j). If $r < 0, n_1 \geq 2, (p, q) \in D_3 \cup D_4$, then when $h \in (0, h_2)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_2 , these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 5 (5-5)).

4). Suppose that $n = 2n_1 + 1, l, n_1 \in \mathbb{Z}^+, q = 0$.

k). If $r < 0, n_1 \geq 2, p > 0$, then when $h \in (h_1, 0)$ in (1.7), Equation (1.3) has a family of uncountably infinite many periodic traveling wave solutions; when h varies from h_1 to 0, these periodic traveling wave solutions will gradually lose their smoothness, and evolve from smooth periodic traveling waves to periodic cusp traveling waves (see Figure 5 (5-5)).

Equation (1.3) has one family of uncountably infinite many periodic traveling wave solutions; when h varies from 0 to h_1 , these periodic travelling wave solutions will gradually lose their smoothness, and evolve from smooth periodic travelling waves to periodic cusp travelling waves (see Figure 7 (7-4)).

Acknowledgements

This work is supported by t NNSF of China (11061010, 11161013). The authors are grateful for this financial support.

References

- [1] Song, M. and Shao, S.G. (2010) Exact Solitary Wave Solutions of the Generalized $(2 + 1)$ Dimensional Boussinesq Equation. *Applied Mathematics and Computation*, **217**, 3557-3563. <http://dx.doi.org/10.1016/j.amc.2010.09.030>
- [2] Chen, H.T. and Zhang, H.Q. (2004) New Double Periodic and Multiple Soliton Solutions of Thegeneralized $(2 + 1)$ -Dimensional Boussinesq Equation. *Chaos, Solitons & Fractals*, **20**, 765-769. <http://dx.doi.org/10.1016/j.chaos.2003.08.006>
- [3] Li, J.B. (2008) Bifurcation of Traveling Wave Solutions for Two Types Boussinesq Equations. *Science China Press*, **38**, 1221-1234. <http://dx.doi.org/10.1007/s11426-008-0129-x>
- [4] Tang, S.Q., Xiao, Y.X. and Wang, Z.J. (2009) Travelling Wave Solutions for a Class of Nonlinear Fourth Order Variant of a Generalized Camassa-Holm equation. *Applied Mathematics and Computation*, **210**, 39-47. <http://dx.doi.org/10.1016/j.amc.2008.10.041>
- [5] Rong, J.H., Tang, S.Q. and Huang, W.T. (2010) Bifurcation of Traveling Wave Solutions for a Class of Nonlinear Fourth Order Variant of a Generalized Camassa-Holm equation. *Communications in Nonlinear Science and Numerical Simulation*, **15**, 3402-3417. <http://dx.doi.org/10.1016/j.cnsns.2009.12.027>
- [6] Tang, S.Q. and Huang, W.T. (2008) Bifurcations of Travelling Wave Solutions for the $K(n, -n, 2n)$ Equations. *Applied Mathematics and Computation*, **203**, 39-49. <http://dx.doi.org/10.1016/j.amc.2008.01.036>
- [7] Li, J.B. and Liu, Z.R. (2002) Travelling Wave Solutions for a Class of Nonlinear Dispersive Equations. *Chinese Annals of Mathematics, Series B*, **23**, 397-418. <http://dx.doi.org/10.1142/S0252959902000365>
- [8] Li, J.B. and Liu, Z.R. (2000) Smooth and Non-Smooth Travelling Waves in a Nonlinearly Dispersive Equation. *Applied Mathematical Modelling*, **25**, 41-56. [http://dx.doi.org/10.1016/S0307-904X\(00\)00031-7](http://dx.doi.org/10.1016/S0307-904X(00)00031-7)
- [9] Bibikov, Y.N. (1979) Local Theory of Nonlinear Analytic Ordinary Differential Equations. *Lecture Notes in Mathematics*, Vol. 702. Springer-Verlag, New York.
- [10] Wang, Z.J. and Tang, S.Q. (2009) Bifurcation of Travelling Wave Solutions for the Generalizedzk Equations. *Communications in Nonlinear Science and Numerical Simulation*, **14**, 2018-2024. <http://dx.doi.org/10.1016/j.cnsns.2008.06.026>
- [11] Takahashi, M. (2003) Bifurcations of Ordinary Differential Equations of Clairaut Type. *Journal of Differential Equations*, **19**, 579-599. [http://dx.doi.org/10.1016/S0022-0396\(02\)00198-5](http://dx.doi.org/10.1016/S0022-0396(02)00198-5)

Call for Papers



American Journal of Computational Mathematics (AJCM)

ISSN: 2161-1203(Print) 2161-1211(Online)
<http://www.scirp.org/journal/ajcm>

American Journal of Computational Mathematics (AJCM) is a journal dedicated to providing a platform for publication of articles about mathematical research in areas of science where computing plays a central and essential role emphasizing algorithms, numerical methods, and symbolic methods.

Subject Coverage

This journal invites original research and review papers that address the following issues. Topics of interest include, but are not limited to:

- Algebraic Theory of Computer Science
- Combinatorics
- Computational Geometry
- Computational Linguistics
- Computational Science
- Computer Simulation
- Computer-Assisted Research
- Discrete Mathematics in Relation to Computer Science
- Logic in Computer Science
- Mathematics of Scientific Computation
- Numerical Methods
- Stochastic Methods
- Symbolic Computation and Computer
- Algebra Systems

We are also interested in short papers (letters) that clearly address a specific problem, and short survey or position papers that sketch the results or problems on a specific topic. Authors of selected short papers would be invited to write a regular paper on the same topic for future issues of *American Journal of Computational Mathematics*.

Notes for Intending Authors

Submitted papers should not have been previously published nor be currently under consideration for publication elsewhere. Paper submission will be handled electronically through the website. All papers are refereed through a peer review process. For more details about the submissions, please access the website.

Website and E-Mail

<http://www.scirp.org/journal/ajcm>

E-mail: ajcm@scirp.org

What is SCIRP?

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science, by delivering superior scientific publications and scientific information solution provider that enable advancement in scientific research.

What is Open Access?

All original research papers published by SCIRP are made freely and permanently accessible online immediately upon publication. To be able to provide open access journals, SCIRP defrays operation costs from authors and subscription charges only for its printed version. Open access publishing allows an immediate, world-wide, barrier-free, open access to the full text of research papers, which is in the best interests of the scientific community.

- High visibility for maximum global exposure with open access publishing model
- Rigorous peer review of research papers
- Prompt faster publication with less cost
- Guaranteed targeted, multidisciplinary audience



Website: <http://www.scirp.org>
Subscription: sub@scirp.org
Advertisement: service@scirp.org