# Iranian Journal of Numerical Analysis and Optimization

Volume 3, Number 1

Winter 2013

Ferdowsi University of Mashhad, Iran

In the Name of God

**Iranian Journal of Numerical Analysis and Optimization (IJNAO)**

This journal is authorized under the registration No. 174/853 dated 1386/2/26, by the Ministry of Culture and Islamic Guidence.

**Volume 3, Number 1, Winter 2013**

**ISSN:** 1735-7144

**Publisher:** Faculty of Mathematical Sciences, Ferdowsi University of Mashhad

**Published by:** Ferdowsi University of Mashhad Press

**Circulation:** 500

**Subscription Information:**

The author(s) receive 10 offprints free of charge. Additional offprints are provided in lots of 50, when the order form is received on time.

**Address:** Iranian Journal of Numerical Analysis and Optimization

Faculty of Mathematical Sciences

Ferdowsi University of Mashhad

P.O. Box 1159, Mashhad 91775, Iran.

Tel-Fax: +98-511-8828606

**E-mail:** mjms@um.ac.ir

**Website:** http://jm.um.ac.ir/index.php/math

**This journal is indexed by:**

- Mathematical Review

- Zenteralblatt

# Iranian Journal of Numerical Analysis and Optimization

Volume 3, Number 1, Winter 2013

Ferdowsi University of Mashhad - Iran

# Iranian Journal of Numerical Analysis and Optimization

This journal is published under the auspices of Ferdowsi University of Mashhad

We would like to acknowledge the help of Atefeh Gooran Orimi and Sayyed Mahdi Mosaic for the typesetting of this issue.

* Full Professor
** Associate Professor
*** Assistant Professor

**Letter from the Editor in Chief**

I would like to welcome you to the Iranian Journal of Numerical Analysis and Optimization (IJNAO). This journal is published biannually and supported by the Faculty of Mathematical Sciences at the Ferdowsi University of Mashhad. Faculty of Mathematical Sciences with three centers of excellence and three research centers is well-known in mathematical communities in Iran.

The main aim of the journal is to facilitate discussions and collaborations between specialists in applied mathematics, especially in the fields of numerical analysis and optimization, in the region and worldwide.

Our vision is that scholars from different applied mathematical research disciplines, pool their insight, knowledge and efforts by communicating via this international journal.

In order to assure high quality of the journal, each article will be reviewed by subject-qualified referees.

Our expectations for IJNAO are as high as any well-known applied mathematical journal in the world. We trust that by publishing quality research and creative work, the possibility of more collaborations between researchers would be provided. We invite all applied mathematicians especially in the fields of numerical analysis and optimization to join us by submitting their original work to the Iranian Journal of Numerical Analysis and Optimization.

Mohammad Hadi Farahi

# Contents

Iranaian J. Numerical Analysis and Optimization
Vol. 3, No. 1, (2013), 1-7

# On convergence of He's variational iteration method for nonlinear partial differential equations

Jafar Saberi-Nadjafi and Asghar Ghorbani

**Abstract**

This paper deals with a novel proof of convergence of He's variational iteration method applied to nonlinear partial differential equations by proposing a new formulation for this technique.

**Keywords:** Variational iteration method; Convergence theorem; Partial differential equations; Burger's equation.

## 1 Introduction

Recently He [8] has written a survey article and some new asymptotic techniques with numerous examples. The limitations of traditional perturbation procedures are illustrated. Various modified perturbation techniques are introduced, and some mathematical tools such as variational theory, homotopy technique, and iteration technique are proposed to overcome the shortcomings. For the nonlinear oscillators, all the reviewed schemes produce high approximate periods, but the accuracy of the amplitudes cannot be ameliorated by iteration. The emphasis of this author [8] is on the variational approaches, parameter-expanding methods, parameterized perturbation technique, homotopy perturbation method, iteration perturbation procedure and ancient Chinese methods. Variational approaches to soliton solution, bifurcation, limit cycle, and period solutions of nonlinear equations including the Ritz method, energy technique, variational iteration method are illustrated in his paper [8].

The variational iteration method (VIM) plays an important role in recent researches in this field. This method is proposed by He [4, 6, 5, 7] as a modification of a general Lagrange multiplier method [11]. It has been shown

Jafar Saberi-Nadjafi
Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: najafi@math.um.ac.ir

Asghar Ghorbani
Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: as_gh56@yahoo.com

that this procedure is a powerful tool for solving various kinds of problems (e.g., see [1, 3, 13, 12]).

In this work, we adapt the technique to nonlinear partial differential equations and we prove the convergence of this method by proposing a new formulation of the method.

## 2 The Variational Iteration Method

The idea of VIM is very simple and straightforward. To explain the basic idea of VIM, we consider an one dimension general first order nonlinear partial differential equation as follows with the assumption that the equation has the unique solution (note that one can consider a nonlinear partial differential equation in higher dimensions and also more general form without loss of generality):

$$\Phi(t, x, u, u_t, u_x, \cdots) = 0, \tag{1}$$

with the specified initial condition (i.e., $u(x, t_0) = f(x)$). We assume that the nonlinear operator $\varPhi$ is continuous with respect to its arguments and $u(x, t)$ is an unknown. We first consider Eq. (1) as follows:

$$\Lambda[u(x, t)] + \mathrm{N}[u(x, t)] = 0, \tag{2}$$

with, for example, the assumption $\alpha(x, t) \neq 0$

$$\Lambda[u] = \alpha(x, t)u_t + \beta(x, t)u$$

and

$$\mathrm{N}[u] = \Phi(t, x, u, u_t, u_x, \cdots) - \alpha(x, t)u_t - \beta(x, t)u, \tag{3}$$

where, as shown above, $\Lambda$ with the property $\Lambda y \equiv 0$ when $y \equiv 0$ denotes the linear operator with respect to $u$ and N is a nonlinear operator with respect to $u$. We then construct a correction functional for Eq. (2) as [7]:

$$\begin{aligned} u_{n+1}(x, t) = u_n(x, t) + \int_{t_0}^{t} \lambda_{(x,t,s)} \; \{ \, & \alpha(x, s)u_{n_s}(x, s) + \beta(x, s)u_n(x, s) \\ & + \mathrm{N}[\tilde{u}_n(x, s)] \; \} \; \mathrm{d}s, \end{aligned} \tag{4}$$

where $u_0(x, t)$ is the initial guess and the subscript $n$ denotes the $n$-th iteration, and $\lambda_{(x,t,s)} \neq 0$ denotes the Lagrange multiplier, which can be identified efficiently via the variational theory, and $\tilde{u}_n$ is considered as a restricted variation [7], i.e., $\delta\tilde{u}_n = 0$.

Taking variation with respect to the independent variable $u_n$, noticing that $\delta u_n(x, t_0) = 0$ and by making the correction functional stationary, we obtain $\delta u_{n+1}(x, t) = 0$ and therefore we have the following:

$$\begin{aligned}\delta u_{n+1}(x,t) =\ & \delta u_n(x,t) + \delta \int_{t_0}^{t} \lambda_{(x,t,s)}\ \{\alpha(x,s)u_{n_t}(x,s) + \beta(x,s)u_n(x,s) \\ & +\mathrm{N}[\tilde{u}_n(x,s)]\}\ \mathrm{d}s \\ =\ & \delta u_n(x,t) + \alpha(x,s)\lambda_{(x,t,s)}\delta u_n(x,s)\big|_{s=t} \\ & - \int_{t_0}^{t} \left\{ \frac{\partial}{\partial s}\left(\alpha(x,s)\lambda_{(x,t,s)}\right) - \beta(x,s)\lambda_{(x,t,s)} \right\} \delta u_n(x,s)\,\mathrm{d}s \\ =\ & [1+\alpha(x,s)\lambda_{(x,t,s)}]\ \delta u_n(x,s)|_{s=t} \\ & - \int_{t_0}^{t} \left\{ \frac{\partial}{\partial s}\left(\alpha(x,s)\lambda_{(x,t,s)}\right) - \beta(x,s)\lambda_{(x,t,s)} \right\} \delta u_n(x,s)\,\mathrm{d}s \\ =\ & 0. \end{aligned} \tag{5}$$

Therefore, we have the following stationary conditions:

$$\alpha(x,s)\lambda_{(x,t,s)}\big|_{s=t} = -1, \tag{6}$$

$$\frac{\partial}{\partial s}\left(\alpha(x,s)\lambda_{(x,t,s)}\right) - \beta(x,s)\lambda_{(x,t,s)} = 0. \tag{7}$$

Hence, the Lagrange multiplier can be readily identified as

$$\lambda_{(x,t,s)} = \frac{-1}{\alpha(x,t)} \exp\left( \int_{s}^{t} \frac{\alpha_\tau(x,\tau) - \beta(x,\tau)}{\alpha(x,\tau)}\,\mathrm{d}\tau \right). \tag{8}$$

As a result, we have the following variational iteration formula:

$$u_{n+1}(x,t) = u_n(x,t) + \int_{t_0}^{t} \lambda_{(x,t,s)}\Phi(s,x,u_n(x,s),u_{n_s}(x,s),u_{n_x}(x,s),\cdots)\,\mathrm{d}s. \tag{9}$$

Accordingly, the successive approximations $u_n(x,t), n \geq 0$ of VIM will be readily obtained by choosing all the above-mentioned parameters. Consequently, the exact solution may be obtained by using

$$u(x,t) = \lim_{n\to\infty} u_n(x,t). \tag{10}$$

The initial guess can be freely chosen with possible unknown constants, it can also be solved from its corresponding linear homogeneous equation $\Lambda[u_0(x,t)] = 0$. It is important to note that for linear problems, the exact solution can be obtained easily by only one iteration due to the fact that the auxiliary function can be suitably identified [9]. For nonlinear problems, in general, one iteration leads to highly accurate solution by VIM if the initial solution is carefully chosen with some unknown parameters.

## 3 Convergence Theorem

The variational iteration formula makes a recurrence sequence $\{u_n(x,t)\}$. Obviously, the limit of the sequence will be the solution of Eq. (1) if the sequence is convergent. In this section, we give a new proof of convergence of VIM in details by introducing a new iterative formulation of this procedure. Here, we suppose that for every positive integer $n$, $u_n \in C[t_0, T]$, and $\{u_{n_t}(x,t)\}$ and $\{u_{n_x}(x,t)\}$ are uniformly convergent.

**Lemma 3.1.** *The variational iteration formula* (9) *is equivalent to the following iterative relation*

$$\Lambda[u_{n+1}(x,t) - u_n(x,t)] = -\Phi(t, x, u_n(x,t), u_{n_t}(x,t), u_{n_x}(x,t), \cdots), \qquad (11)$$

*where* $\Lambda$ *is as noted in* (3).

**Proof**. Suppose $u_n$ and $u_{n+1}$ satisfies the variational iteration formula (9). Applying $\frac{\partial}{\partial t}$ into both sides of (9), results in

$$\frac{\partial}{\partial t}[u_{n+1}(x,t) - u_n(x,t)] = \int_{t_0}^{t} \frac{\partial \lambda_{(x,t,s)}}{\partial t} \Phi \, \mathrm{d}s + \frac{\partial}{\partial t}[\lambda_{(x,t,s)}\big|_{s=t} \ \Phi]. \qquad (12)$$

Now, by using conditions (6) and (7), and $\frac{\partial \lambda_{(x,t,s)}}{\partial t} = -\frac{\beta(x,t)}{\alpha(x,t)}$, we will have

$$\begin{aligned} \alpha(x,t)\tfrac{\partial}{\partial t}[u_{n+1}(x,t) - u_n(x,t)] + \beta(x,t)[u_{n+1}(x,t) - u_n(x,t)] = \\ -\Phi(t, x, u_n(x,t), u_{n_t}(x,t), u_{n_x}(x,t), \cdots \}). \end{aligned} \qquad (13)$$

From the definition (3) of $\Lambda$, we obtain

$$\Lambda[u_{n+1}(x,t) - u_n(x,t)] = -\Phi(t, x, u_n(x,t), u_{n_t}(x,t), u_{n_x}(x,t), \cdots). \qquad (14)$$

Conversely, suppose $u_n$ and $u_{n+1}$ satisfies (11). Multiplying (11) by $\lambda_{(x,t,s)}$, in view of the definition of $\Lambda$ and $\lambda_{(x,t,s)} \neq 0$, and next by integrating on both sides of the resulted term from $t_0$ to $t$, yields

$$\begin{aligned} \textstyle\int_{t_0}^{t} \lambda_{(x,t,s)}\alpha(x,s)[\frac{\partial}{\partial s}u_{n+1}(x,s) - \frac{\partial}{\partial s}u_n(x,s)]\,\mathrm{d}s + \int_{t_0}^{t} \lambda_{(x,t,s)}\beta(x,s)[u_{n+1}(x,s) - u_n(x,s)]\,\mathrm{d}s \\ = -\textstyle\int_{t_0}^{t} \lambda_{(x,t,s)}\Phi(s)\,\mathrm{d}s. \end{aligned} \qquad (15)$$

Using simple integration by parts, the expression (15) becomes

$$\begin{aligned} \alpha(x,t)\lambda_{(x,t,t)}[u_{n+1}(x,t) - u_n(x,t)] - \textstyle\int_{t_0}^{t} \left(\frac{\partial}{\partial s}(\alpha(x,s)\lambda_{(x,t,s)}\right) - \beta(x,s)\lambda_{(x,t,s)})[u_{n+1}(x,s) \\ -u_n(x,s)]\,\mathrm{d}s = -\textstyle\int_{t_0}^{t} \lambda_{(x,t,s)}\,\Phi(s)\,\mathrm{d}s, \end{aligned} \qquad (16)$$

which exactly results (9) upon imposing conditions (6) and (7), i.e.,

$$u_{n+1}(x,t) = u_n(x,t) + \int_{t_0}^{t} \lambda_{(x,t,s)}\Phi(s, x, u_n(x,s), u_{n_s}(x,s), u_{n_x}(x,s), \cdots)\,\mathrm{d}s. \qquad (17)$$

and this ends the proof. □

**Theorem 3.1.** *If the sequence* (10) *converges, where* $u_n(x,t)$ *is produced by the variational iteration formulation of* (9), *then it is the exact solution of the equation* (1).

**Proof**. If the sequence $\{u_n(x,t)\}$ converges, we can write

$$v(x,t) = \lim_{n\to\infty} u_n(x,t), \tag{18}$$

and it holds

$$v(x,t) = \lim_{n\to\infty} u_{n+1}(x,t). \tag{19}$$

Using expressions (18) and (19), and by the definition of $\Lambda$ in (3), we can easily gain

$$\lim_{n\to\infty} \Lambda\left[u_{n+1}(x,t) - u_n(x,t)\right] = \Lambda \lim_{n\to\infty}\left[u_{n+1}(x,t) - u_n(x,t)\right] = 0. \tag{20}$$

From (20) and according to the Lemma 3.1, we obtain

$$\Lambda \lim_{n\to\infty}\left[u_{n+1}(x,t) - u_n(x,t)\right] = -\lim_{n\to\infty} \Phi(t,x,u_n(x,t),u_{n_t}(x,t),u_{n_x}(x,t),\cdots) = 0, \tag{21}$$

which gives us

$$\lim_{n\to\infty} \Phi\left(t,x,u_n(x,t),u_{n_t}(x,t),u_{n_x}(x,t),\cdots\right) = 0. \tag{22}$$

From (22) and the continuity of $\Phi$ operator, it holds

$$\begin{aligned}
&\lim_{n\to\infty} \Phi(t,x,u_n(x,t),u_{n_t}(x,t),u_{n_x}(x,t),\cdots) \\
&\quad = \Phi(t,x,\lim_{n\to\infty} u_n(x,t), \lim_{n\to\infty} u_{n_t}(x,t), \lim_{n\to\infty} u_{n_x}(x,t),\cdots) \\
&\quad = \Phi\left(t,x,\lim_{n\to\infty} u_n(x,t), \left(\lim_{n\to\infty} u_n(x,t)\right)_t, \left(\lim_{n\to\infty} u_n(x,t)\right)_x, \cdots\right) \\
&\quad = \Phi\left(t,x,v(x,t),v_t(x,t),v_x(x,t),\cdots\right).
\end{aligned} \tag{23}$$

Now, from Equations (22) and (23), we have

$$\Phi\left(t,x,v,v_t,v_x,\cdots\right) = 0, \qquad t_0 \le t \le T. \tag{24}$$

On the other hand, using the specified initial conditions and the definition of the initial guess, we have

$$v(x,t_0) = \lim_{n\to\infty} u_n(x,t_0) = f(x), \quad \text{since} \quad u(x,t_0) = u_n(x,t_0) = f(x), \quad n \ge 0, \tag{25}$$

Therefore, according to (24)-(25), $v(x,t)$ must be the exact solution of the equation (1), this ends the proof. □

Note that the above theorem is valid for the linear operator $\Lambda$ defined by (3). This convergence theorem is important. It is because of this theorem we can focus on ensuring that the approximation sequence converges. It is clear that the convergence of the sequence (10) depends upon the initial guess $u_0(x,t)$ and the linear operator $\Lambda$. Fortunately, VIM provides us with great freedom of choosing them. Thus, as long as $u_0(x,t)$ and $\Lambda$ are so properly chosen that the sequence (10) converges in a region $t_0 \le t \le T$, it must converge to the exact solution in this region. Therefore, the combination of the convergence theorem and the freedom of the choice of the initial guess $u_0(x,t)$ and the linear operator $\Lambda$ establishes the cornerstone of the validity and flexibility of VIM.

## 4 An Illustrative Example

In order to illustrate the efficiency of the VIM described in this paper, we present one example.

**Example** A much-considered example is the Burger's equation [12,13]

$$u_t + uu_x - u_{xx} = 0, \tag{26}$$

This equation was only intended as an approach to the study of turbulence because it exhibited some essential characteristics of the more realistic (and difficult) equations. This equation involves nonlinearity, dissipation, and is relatively simple. The VIM solves much more difficult systems. We consider it now to show the simplicity of a proper solution.

According to the VIM procedure, (9), one can obtain the following variational iteration relation ($\lambda_{(x,t,s)} = -1$):

$$u_{n+1}(x,t) = u_n(x,t) - \int_0^t \{u_{n_t}(x,s) + u_n(x,s)u_{n_x}(x,s) - u_{n_{xx}}(x,s)\}\, \mathrm{d}s. \tag{27}$$

The problem is completely defined when the initial condition is specified. If we specify $u = x$ when $t = 0$, we have

$$\begin{aligned}
u_0(x,t) &= x,\\
u_1(x,t) &= x - xt + \mathrm{O}(t^2),\\
u_2(x,t) &= x - xt + xt^2 + \mathrm{O}(t^3),\\
u_3(x,t) &= x - xt + xt^2 - xt^3 + \mathrm{O}(t^4),\\
u_4(x,t) &= x - xt + xt^2 - xt^3 + xt^4 + \mathrm{O}(t^5),\\
&\cdots
\end{aligned} \tag{28}$$

Thus, $u(x,t) = x/(1+t)$ which is the exact solution of (26).

## 5 Conclusion

In this paper, we have given a proof of convergence of He's variational iteration method by presenting a new formulation of He's method for nonlinear partial differential equations. The main property of this method is in its flexibility and ability to solve nonlinear equations accurately and conveniently without decomposing the nonlinear terms, which makes the procedure very complex. This technique is a very powerful tool for solving nonlinear problems. Furthermore, it gives an accurate and easily computable solution by means of a truncated series whose convergence is fast.

## References

1. Abdou, M.A. and Soliman, A.A., *Variational iteration method for solving Burger's and coupled Burger's equations*, J. Comput. Appl. Math. **81** (2005) 245-251.
2. Burgers, J.M., *A mathematical model illustrating the theory of turbulence*, Adv. App. Mech. **1** (1948) 171-199.
3. Dehghan, M. and Tatari, M., *The use of He's variational iteration method for solving the Fokker–Planck equation*, Phys. Scripta **74** (2006) 310-316.
4. He, J.H., *Variational iteration method for delay differential equations*, Comm. Non-Linear. Sci. Numer. Simulation 2 (1997) 235-236.
5. He, J.H., *Approximate analytical solution for seepage flow with fractional derivatives in porous media*, Comput. Methods. Appl. Mech. Engng. **167** (1998) 57-68.
6. He, J.H., *Approximate solution of nonlinear differential equations with convolution product non-linearities*, Comput. Methods. Appl. Mech. Engng. **167** (1998) 69-73.
7. He, J.H., *Variational iteration method- a kind of non-linear analytical technique: some examples*, Internat. J. Non-Linear Mech. **34** (1999) 699-708.
8. He, J.H., *Some asymptotic methods for strongly nonlinear equations*, Internat. J. Modern Phys. B 20 (2006) 1141-1199.
9. He, J.H., *Variational iteration method- some recent results and new interpretations*, J. Comput. Appl. Math., **207** (2007) 3-17.
10. Hood, S., *Next exact solutions of Burgers' equation- An extension to the direct method of Clarkson and Kruskal*, J. Math. Phys. **36** (1995) 1971-1990.
11. Inokuti, M., Sekine, H. and Mura, T., *General use of the Lagrange multiplier in nonlinear mathematical physics, in: Variational Methods in the Mechanics of Solids*, Pergamon Press, NewYork, (1978), pp. 156-162.
12. Momani, S. and Abuasad, S., *Application of He's variational iteration method to Helmholtz equation*, Chaos Solitons Fractals **27** (2006) 1119-1123.
13. Tatari, M. and Dehghan, M., *On the convergence of He's variational iteration method*, J. Comput. Appl. Math. **207** (2007) 121-128.

Iranaian J. Numerical Analysis and Optimization
Vol. 3, No. 1, (2013), 9-12

# Block-Coppels chaos in set-valued discrete systems

Bahman Honary and Mojtaba Jazaeri

**Abstract**

Let $(X, d)$ be a compact metric space and $f : X \to X$ be a continuous map. Consider the metric space $(K(X), H)$ of all non empty compact subsets of $X$ endowed with the Hausdorff metric induced by $d$. Let $\bar{f} : K(X) \to K(X)$ be defined by $\bar{f}(A) = \{f(a) : a \in A\}$ . We show that Block-Coppels chaos in $f$ implies Block-Coppels chaos in $\bar{f}$ if $f$ is a bijection.

**Keywords:** Chaos; Discrete system; Dynamical system.

## 1 Introduction

Let $(X, d)$ be a compact metric space with metric $d$ and $f : X \to X$ be a continuous map. For every positive integer n, we define $f^n$ inductively by $f^n = f \circ f^{n-1}$ , where $f^0$ is the identity map on $X$. A map $f$ is called to be Block-Coppels chaotic [3] if there exist disjoint non-empty compact subsets $J, K$ of $X$ and a positive integer n such that $J \cup K \subseteq f^n(J) \cap f^n(K)$.
Roman-Flores and Chalco-Cano investigated Robinsons chaos in set-valued discrete systems [5]. Gu investigated Katos chaos in set-valued discrete systems [2]. Devaneys chaos in set-valued discrete systems has been studied in several papers. For example see [4], [1].
In this paper, we investigate the relationships between Block-Coppels chaoticity of $(X, f)$ and Block-Coppels chaoticity of $(K(X), \bar{f})$.

Mojtaba Jazaeri
Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: se_ja81@stu-mail.um.ac.ir, seja81@gmail.com

Bahman Honary
Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: honary@sbu.ac.ir

## 2 Preliminaries

Let $(X, d)$ be a compact metric space with metric $d$. The distance of a point $x$ from a set A in $X$ is defined by $d(x, A) = \inf\{d(x, a) : a \in A\}$ if $A \neq \varnothing$, and $d(x, \varnothing) = 1$. Let $K(X)$ be the family of all non-empty compact subsets of $X$.
The Hausdorff metric on $K(X)$ is defined by $H(A, B) = \max\{\sup\{d(a, B) : a \in A\}, \sup\{d(b, A) : b \in B\}\}$ for $A, B \in K(X)$. It is easy to see that $(K(X), H)$ is a compact metric space.
Let $\tau_d$ be the topology of $X$ induced by the metric $d$. The topology $\tau_H$ of $K(X)$ induced by the Hausdorff metric $H$ coincides with the topology $\tau_v$ generated by the basis $\beta_v$ consisting of all sets of the form $G_0^u \cap G_1^l \cap \cdots \cap G_k^l$ where $G_0, G_1, \ldots, G_k \in \tau_d, G_0^u = \{A \in K(X) : A \subseteq G_0\}$ and

$$G_i^l = \{A \in K(X) : A \cap G \neq \varnothing\}, i = 1, 2, \ldots, k.$$

The topology $\tau_\nu$ is also called the Vietoris topology or the exponential topology on $K(X)$.
If $f : X \in X$ is a continuous map then one can define a continuous map $\bar{f} : K(X) \to K(X)$ by letting $\bar{f}(A) = \{f(a) : a \in A\}$ for every $A \in K(X)$.

## 3 Block-Coppels Chaoticity

In this section, we show that Block-Coppels chaoticity of $(X, f)$ implies Block-Coppels chaoticity of $(K(X), \bar{f})$ if $f$ is bijection.

**Definition 3.1.** Let $A$ be a subset of $X$, the extension of $A$ to $K(X)$ is defined by $e(A) = \{K \in K(X) : K \subseteq A\}$.

Remark. It is clear that $e(A) = \varnothing$ if and only if $A = \varnothing$.

**Lemma 3.1.** *Let $A$ be a non-empty compact subset of $X$. Then, $e(A)$ is a non-empty compact subset of $K(X)$.*

**Proof**. It is sufficient to show that $(e(A))^c$ is open because in this case $e(A)$ is closed and a closed subset of a compact space, is compact. If $K \in (e(A))^c$ then $K \notin e(A)$ which means $K \subsetneq A$. Therefore $K \cap A^c \neq \varnothing$, and hence $K \in (A^c)^l$. So that

$$(e(A))^c \subseteq (A^c)^l \tag{1}$$

On the other hand if $K \in (A^c)^l$ then $K \cap A^c \neq \varnothing$, therefore $K \subsetneq A$ and hence $K \notin e(A)$. So that $K \in (e(A))^c$ and therefore

$$(A^c)^l \subseteq (e(A))^c \tag{2}$$

These two relations show that $(e(A))^c = (A^c)^l$ and the proof is completed. □

The following lemma is obvious from definition.

**Lemma 3.2.** *Let $A$ be a subset of $X$. Then,*

*i)* $e(A \cap B) = e(A) \cap e(B)$;
*ii)* $\bar{f}(e(A)) \subseteq e(f(A))$;
*iii)* $\overline{f^n} = \bar{f}^n$.

**Lemma 3.3.** *Let $f : X \to X$ be a continuous bijection and $A$ be a subset of $X$.*
*Then $\bar{f}(e(A)) = e(f(A))$.*

**Proof**. According to previous Lemma $\bar{f}(e(A)) \subseteq e(f(A))$. Conversely if $K \in e(f(A))$, then $f^{-1}(K) \subseteq f^{-1}(f(A))$. Also $f^{-1}(f(A)) = A$ because $f$ is a bijection. Therefore $f^{-1}(K) \subseteq A$ and $f(f^{-1}(K)) \in f(e(A))$. Also $f^{-1}(f(K)) = K$. Hence $K \in f(e(A))$ and therefore $e(f(A)) \subseteq \bar{f}(e(A))$. □

**Theorem 3.1.** *Let $X$ be a compact metric space and $f : X \to X$ be a continuous bijection. If $f$ is chaotic in the sense of Block-Coppel's, then so is $\bar{f}$.*

**Proof**. Let $f$ be Block-Coppel's chaotic, then there exist disjoint non-empty compact subsets $J, K$ of $X$ and a positive integer $n$ such that $J \cup K \subseteq f^n(J) \cap f^n(K)$. We claim that

$$e(J) \cup e(K) \subseteq \bar{f}^n(e(J)) \cap \bar{f}^n(e(K)).$$

Since $J \subseteq f^n(J)$, then $e(J) \subseteq e(f^n(J))$. According to Lemma 3.3 $e(f^n(J)) = \bar{f}^n(e(J))$. Therefore $e(J) \subseteq \bar{f}^n(e(J))$. In a similar way

$$e(J) \subseteq \bar{f}^n(e(K)), e(K) \subseteq \bar{f}^n(e(K)) \text{ and } e(K) \subseteq \bar{f}^n(e(J)).$$

Therefore

$$e(J) \cup e(K) \subseteq \bar{f}^n(e(J)) \cap \bar{f}^n(e(K))$$

On the other hand $e(J) \cap e(K) = e(J \cap K) = \varnothing$ and according to Lemma 3.1 $e(J), e(K)$ are compact, and the proof is completed. □

## References

1. Gu, R. and Guo, W., *On mixing property in set- valued discrete systems.* Chaos, Solitons & Fractals, **28**, (2006), 747-754.
2. Gu, R., *Katos chaos in set-valued discrete systems.* Chaos, Solitons & Fractals, **31**, (2007), 765-771.

3. Palmisani, Ch., *Chaotic Dynamics in Solow-type Growth Models*. Doctoral thesis, University of Naples Federico II, (2008).
4. Roman-Flores, H., *A note on transitivity in set-valued discrete systems*. Chaos, Solitons & Fractals, *17*, (2003), 99104.
5. Roman-Flores, H. and Chalco-Cano Y., *Robinsons chaos in set-valued discrete systems.* Chaos, Solitons & Fractals, **25**, (2005), 33-42.

Iranaian J. Numerical Analysis and Optimization
Vol. 3, No. 1, (2013), 13-30

# A level set moving mesh method in static form for one dimensional PDEs

Maryam Arab Ameri

**Abstract**

In this paper, we propose an adaptive mesh approach for time dependent parial differential equations, based on a so-called moving mesh PDE(MMPDE) and level set method. It means that the velocity of mesh nodes is calculated by MMPDE and is employed as veocity in the level set equation. Then, at each time level, the mesh points are considered as the level contours of the level set function. Finally the method is merged with local time step technique.

**Keywords:** Adaptive grid; Level set function; Level contours; Moving mesh; Local time stepping refinement; MMPDE.

## 1 Introduction

In this paper, we discuss a class of adaptive mesh algorithms for solving time-dependent partial differential equations(PDEs) whose solutions have large variations over a given physical domain, such as shock waves, boundary layer and interior layer.

This method is based on the level set concept. the level set methods are powerful numerical techniques for tracking the evolution of interfaces moving in complex ways. The level set methods were used for the first time to represent the evolution of surfaces implicity by Osher and Sethian [11]. Due to its many advantages, this approach has been used for many cases [1, 5, 9, 13, 19], for example, it has been used for mesh generation around a body(inside or outside) by the level set function [12]. Liao et al. presented some points about using the level set functions for moving grid based on deformation method [6]. In this paper, we also formulate an adaptive mesh method which is based on the level set method. This method is combined with a class of moving mesh methods which employs a moving mesh partial differential equation, MMPDE, to perform mesh adaptation. The MMPDE is formulated in terms of coordinate transformation or mapping,[3, 4, 8, 16, 17]. Several moving mesh equations(MMPDEs) based upon equidistribution principle have been

Maryam Arab Ameri
Department of Mathematics, University of Sistan and Baluchestan, Zahedan, Iran. e-mail: arabameri@math.usb.ac.ir

derived in [3], where in Section 2, we briefly review some of them. Finally after introducing the new method, we present local time stepping refinement technique based on the slope of the level set function for increasing the efficiency of the moving mesh method. The mentioned method is used to solve scalar field satisfying

$$u_t(x,t) = L(u), \tag{1}$$

where $L$ is a differential operator defined on physical domain $\Omega$, $a \leq x \leq b$ and $0 \leq t \leq T_f$.

## 2 Moving Mesh Methods

In this section, we develop a new moving mesh method based on the level set concept. We start with a review of the equidistribution principle(EP)[3, 15] and derive MMPDEs from that principle in subsection 2.1,then a description of adaptive grid based on level set method is given in the subsection 2.2.

### 2.1 MMPDE

The evolution of moving computational grid can be viewed as a discretization of a one-to-one time dependent coordinate mapping. Let $x$ and $\xi$ denote the physical and computational coordinates, with domains $\Omega$ and $\Omega_c$, respectively. Without loss of generality, both of them are assumed to be in $[0,1]$. Define a coordinate transformation by:

$$x = x(\xi,t),\ \ \xi \in [0,1],\ \ x(0,t) = 0, \quad x(1,t) = 1. \tag{2}$$

The computational coordinates is discretized on a uniform mesh given by

$$\xi_j = \frac{j}{N}, \quad j = 0, 1, \cdots, N,$$

where $N$ is a certain positive integer and the corresponding mesh in $x$ denoted by

$$0 = x_0 < x_1(t) < x_2(t) < \cdots < x_{N-1}(t) < x_N = 1.$$

A major factor of moving mesh approach is the monitor function, $\rho(x,t)$, which is chosen to be somehow a measure of the solution error. For a given monitor function, the mesh point locations, $x_j(t)$, could be required to satisfy the following equidistribution principle (EP) for all the values of time $t$ [3]:

$$\int_{x_{j-1}(t)}^{x_j(t)} \rho(x,t)dx = \frac{1}{N}\int_0^1 \rho(x,t)dx = \frac{1}{N}\theta(t),$$

or equivalently

$$\int_0^{x_j(t)} \rho(x,t)dx = \frac{j}{N}\theta(t) = \xi_j\theta(t). \tag{3}$$

By differentiating (3) with respect to $\xi$ once and twice,, we obtain two differential forms of EP,

$$\rho(x(\xi,t))\frac{\partial}{\partial\xi}x(\xi,t) = \theta(t), \tag{4}$$

$$\frac{\partial}{\partial\xi}(\rho(x(\xi,t),t)\frac{\partial}{\partial\xi}x(\xi,t)) = 0. \tag{5}$$

These EPs,(3),(4) and (5), which do not contain the node speed $\dot{x}(\xi,t)$,are called quasi-static EPs(QSEPs). Related to these QSEPs, various MMPDEs were derived in [3] by taking the mesh to satisfy the above EP (Eq. (5)) at a later time $t+\tau$ instead of at $t$. In this case the mesh should satisfy

$$\frac{\partial}{\partial\xi}\rho(x(\xi,t+\tau),t+\tau)\frac{\partial}{\partial\xi}x(\xi,t+\tau) = 0 \tag{6}$$

where the parameter $\tau$ is called a relaxation time. By expanding the term $\frac{\partial}{\partial\xi}x(\xi,t+\tau)$ in Taylor series and dropping certain higher order terms, various MMPDEs can be obtained. Two of them which are used in our work are MMPDE5 and MMPDE6:

$$MMPDE5: \quad -\dot{x} = -\frac{1}{\tau}\frac{\partial}{\partial\xi}(\rho\frac{\partial x}{\partial\xi}), \tag{7}$$

$$MMPDE6: \quad \frac{\partial^2\dot{x}}{\partial\xi^2} = -\frac{1}{\tau}\frac{\partial}{\partial\xi}(\rho\frac{\partial x}{\partial\xi}). \tag{8}$$

These MMPDEs and also the rest of them not only force the mesh ,$x(\xi,t)$, toward equidistribution principle but also prevent the mesh from crossing. More specifically, the term $-(\frac{1}{\tau})(\frac{\partial}{\partial\xi}(\rho\frac{\partial \mathrm{x}}{\partial\xi}))$ plays the fundamental role of a correction term to make the mesh equidistribute the monitor function and it is stabilizing term for the mesh trajectories.

The monitor function, $\rho$, is chosen such that the mesh points are concentrated in regions where more accuracy is needed, and so $\rho$ is usually taken to be some measure of the solution error estimated from discrete solution values. A commonly used monitor function is $\rho = \sqrt{1+\alpha u_x^2}$ ($\alpha$ is regularizing factor), which equidistributes the arclength of the solution $u$.

In this work, we discretize MMPDE5 and MMPDE6 by centered finite difference in space, which yields for MMPDE5

$$\dot{x}_j = \frac{E_j}{\tau}, \quad j = 0,1,\ldots,N$$

and for MMPDE6

$$\dot{x}_{j+1} - 2\dot{x}_j + \dot{x}_{j-1} = -\frac{E_j}{\tau}, \quad j = 0, 1, \ldots, N.$$

The quantity $E_j$ represents a centered approximation to the term on the right hand side of MMPDE5 and MMPDE6 given by

$$E_j = M_{j+1/2}(x_{j+1} - x_j) - M_{j-1/2}(x_j - x_{j-1}), \quad j = 0, 1, \ldots, N$$

where $M_{j+1/2} = \frac{1}{2}(M_j + M_{j+1}), M_j = M(u_j)$, and $u_j \approx u(x_j, t)$ is an approximation of the solution at grid point $x_j$.

## 2.2 Moving Mesh Based on Level Set Approach

In this section, at first a new moving grid method is formulated, and then the algorithm of this method is presented.

Essential point in all applications of the level set method is using the implicit representation. This point is also used in grid generation in such a way that the mesh nodes are represented implicitly by the level contours of the level set function. In this method, at each time level we construct the level set function,$\psi(x, t)$, and the mesh points $x_j, j = 0, 1, \ldots, N$, are obtained by the level contours of $\psi(x, t)$, that means the mesh points are:

$$\{x_j \mid \psi(x, t) = c_j,\ j = 1, 2, \ldots, N\} \tag{9}$$

where $c_j$ is $j$-th component of a constant vector $C = (0, \frac{1}{N}, \frac{2}{N}, \ldots, 1)$ and $N$ is the number of mesh points.

In fact, in each time, we seek a level set function, $\xi = \psi(x, t) : \Omega \to \Omega_c$ , which maps $x_j$ to $c_j = \frac{j}{N},\ j = 0, 1, \ldots, N$ and satisfies the well-known equidistribution principle

$$J\rho = \frac{\sigma}{|\Omega_c|}, \tag{10}$$

where $J$ is the Jacobian of the coordinate transformation $x = \psi^{-1}(\xi, t) : \Omega_c \to \Omega$, and $\rho = \rho(x, t)$ is a given monitor function, and $\sigma = \sigma(t) = \int_\Omega \rho(x, t) dx$.

For updating the level set function or equivalently for updating the position of mesh nodes, the well-known level set equation is applied,

$$\psi_t(x, t) + v\psi_x(x, t) = 0. \tag{11}$$

The ideal initial condition for the above equation is $\psi(x, t) = x$, because in the most mesh adaptation algorithms, the purpose is to convert a uniform mesh at initial time to an equidistributed mesh at the next time levels. The mentioned initial condition gives a uniform mesh at $t = 0$. Also the following boundary conditions are considered for the level set equation,

$$\psi(0,t)=0, \psi(1,t)=1.$$

As we mentioned at each time level, any

$$\psi(x,t)=a \tag{12}$$

has a correspondence point on $x$-axis which is one of the new mesh nodes. By differentiating (12) with respect to $t$, we get

$$\psi_t(x,t)+\psi_x(x,t)\dot{x}=0, \tag{13}$$

and when compared with (11) and (13), we get

$$\dot{x}=v, \tag{14}$$

that means, the nodes velocity is equal to $v$.
So for calculating the nodes velocity in the level set equation, the MMPDE5 or MMPDE6 should be applied which generates an equidistributed mesh through this algorithm.
At each time level, after determining new nodes, the solution of PDE should be determined. For this purpose, let

$$\tilde{U}(\psi(x,t),t)=u(x,t), \tag{15}$$

then

$$\tilde{U}_t=u_x\dot{x}+u_t$$

where $u_t=L(u)$ by (1) and $\dot{x}=v$ by (14). The derivatives in $L(u)$, such as $u_x, u_{xx}$ are also transformed. For example, from (15), we have

$$\tilde{U}_\psi=u_x x_\psi \Rightarrow u_x=\frac{\tilde{U}_\psi}{x_\psi}=\frac{\tilde{U}_\psi}{\frac{\sigma}{\rho|\Omega_c|}}=\frac{\rho|\Omega_c|}{\sigma}\tilde{U}_\psi,$$

$$\tilde{U}_{\psi\psi}=u_{xx}(x_\psi)^2+u_x x_{\psi\psi} \Rightarrow u_{xx}=\frac{\tilde{U}_{\psi\psi}-u_x x_{\psi\psi}}{(x_\psi)^2}=(\frac{\rho|\Omega_c|}{\sigma})^2(\tilde{U}_{\psi\psi}-u_x x_{\psi\psi}).$$

The higher derivatives can be derived similarly. The transformed equation for $\tilde{U}(\psi,t)$ takes the form of

$$\tilde{U}_t=\tilde{L}(\tilde{U}), \tag{16}$$

where $\tilde{L}$ is a differential operator in $\psi$. Finally (16) will be solved on a uniform grid.

## 2.3 Algorithm of Adaptive Level Set Method(ALSM)

In this section, we provide an algorithm to solve the given PDE (1) in a moving grid based on the level set function.

**Step1:** Enter the initial time, $t_0$, the final time, $T_f$, the length of time step, $\Delta t$, and the number of mesh points, $N$.

**Step2:** Set $i = 0$.

**Step3:** If $t_0 = 0$, set

$$\begin{aligned} X^{(t_0)} &= \{x_j^0 = \frac{j}{N}, j = 0, 1, \dots, N\}, \\ \psi^{(t_0)} &= X^{(t_0)}, \\ U^{(t_0)} &= u^{(t_0)} = u(X^{(t_0)}, t_0), \\ \tilde{U}^{(t_0)} &= U^{(t_0)} \end{aligned}$$

**Step4:** Determine $\rho(u(x, t_i))$ by the solution $u$ being calculated.

**Step5:** Compute mesh velocity, $v = \dot{x}$, either by MMPDE5:

$$\dot{x}_j = \frac{E_j}{\tau}, \quad j = 0, 1, \dots, N$$

or by MMPDE6:

$$\dot{x}_{j+1} - 2\dot{x}_j + \dot{x}_{j-1} = -\frac{E_j}{\tau}. \quad j = 0, 1, \dots, N$$

**Step6:** Update the level set function by the following equation:

$$\psi_t(x, t) + \psi_x(x, t).v = 0,$$

which by forward finite difference discretization in time converts to:

$$\psi(t_{i+1}) = \psi(t_i) + \Delta t(v\psi_x)_{t_i},$$

with the boundary conditions $\psi_0(t_i + 1) = 0$ and $\psi_N(t_i + 1) = 1$.

**Step7:** Define the inverse of $\psi(x, t)$ for updating the new nodes, $X(t_{i+1})$, in current time.

**Step8:** Determine the values of $u$ on the current time level by the described procedure in previous subsection and solve (16) or

$$\tilde{U}(t_{i+1}) = \tilde{U}(t_i) + \Delta t.\tilde{L}(\tilde{U}(t_i)),$$

**Step9:** Set $i = i + 1$.

**Step10:** If $t_i \leq T_f$ go to Step3, else break.

## 3 Local Time Step Refinement

Local time stepping for one-dimensional conservation laws was first proposed by Osher and Sanders [10]. Tan et. al. proposed variable time stepping for one and two dimension for conservation laws, which uses semi-dynamic moving mesh method such that the CFL condition is used to obtain time interval refinement to compute the solution [18]. Also, Soheili and Salahshour used this approach for the blow-up problems [14].
In this section, we also present the details of the local time stepping refinement(LTSR) which is performed based on the level set function.
Let initial time steps of the problem have the form

$$t_0 = 0 \to t_1 = t_0 + \Delta t \to \ldots \to t_n = t_0 + n\Delta t \to t_{n+1} = t_0 + (n+1)\Delta t \to \ldots \to T_f,$$

where $\Delta t$ is a specified value for the time step and is constant through using this procedure. On the first interval $[t_0, t_1]$, set $\Delta t_0 = \frac{t_1 - t_0}{k_0}$ where $k_0 \in N$ is constant and depends on the slope of the level set function, (the method of determining $k_0$ will be described). We have

$$t_{0+(k_0-i)\Delta t_0} = t_0 + (k_0 - i)\Delta t_0, \ i = k_0, k_0 - 1, \ldots, 0$$

so the time integration on $[t_0, t_1]$ involves $k_0$ sub-steps such that

$$t_0 = 0 \to t_0 + \Delta t_0 \to \ldots \to t_0 + k_0 \Delta t_0 = t_1.$$

Generally, suppose that we are at time level $t = t_n$ and we want to move towards $t = t_{n+1}$. Similarly consider $\Delta t_n = \frac{t_{n+1} - t_n}{k_n}$ such that

$$t_{n+(k_n-i)\Delta t_n} = t_n + (k_n - i)\Delta t_n, \ i = k_n, k_n - 1, \ldots, 0$$

that means, on interval $[t_n, t_{n+1}]$, we have

$$t_n \to t_n + \Delta t_n \to \ldots \to t_n + k_n \Delta t_n = t_{n+1}.$$

This process continues up to $T_f$. Now, $k_0, k_1, \ldots, k_n, \ldots$ are determined by the following process. At the first step, we start integrate the problem from $t_0$ to $t_1 = t_0 + \Delta t$ without any LTSR, we call this process prediction step, then we determine the level set function, the new mesh nodes of adaptive mesh and solution of PDE at $t_1$. According to the property of the mentioned adaptive mesh method in previous section, "for larger slope of the level set function, more mesh nodes are concentrated". Thus according to slope of the level set function, we define the interval $[t_0, t_1]$. So the slope of $\psi_j^{(1)}$ on $[x_j, x_{j+1}]$, $j = 0, 1, \ldots, N$ is calculated, then for having more efficient solution, $[t_0, t_1]$ is subdivided to $k_0$ parts. Judicious choice for $k_0$ can be

$$k_0 = \min\{\max_j[slope(\psi_j^{(1)}), 10]\}, \quad j = 0, 1, \ldots, N,$$

and the time integration is done on this sub-interval again but with $\Delta t_0 = \frac{t_1 - t_0}{k_0}$. After obtaining the solution at $t_1$ we act on $[t_1, t_2]$ similarly and determine the solution at $t_2$.
Generally, after calculating the solution at $t_n$, at first we have a prediction process concluding time integration from $t_n$ to $t_{n+1}$ with pre-determined value of $\Delta t$ as time step for knowing the slope of $\psi_j^{(n+1)}$ on $[x_j, x_{j+1}], j = 0, 1, \ldots, N$ and subsequently $k_n = \min\{\max_j[slope(\psi_j^{(n+1)})], 10\}$. Then the time integration is performed on this sub-interval $k_n$ times with local time step $\Delta t_n$.

## 3.1 Algorithm of Adaptive Level Set Method with LTSR

**Step1:** Enter the initial time, $t_0$, the final time, $T_f$, the length of time step, $\Delta t$, and the number of mesh points, $N$.
**Step2:** Set $i = 0$.
**Step3:** If $t_0 = 0$, set

$$X^{(t_0)} = \{x_j^0 = \frac{j}{N}, j = 0, 1, \ldots, N\},$$

$$\psi^{(t_0)} = X^{(t_0)},$$

$$\tilde{U}^{(t_0)} = U^{(t_0)} = u^{(t_0)} = u(X^{(t_0)}, t_0).$$

**Step4:** Run ALSM one time with $t_0 = t_i, T_f = t_{i+1}$ and $\Delta t$, then calculate $\psi^{(t_{i+1})}, X^{(t_{i+1})}$ and $\tilde{U}^{(t_{i+1})}$.
**Step5:** Calculate the slope of $\psi^{(i+1)} = \psi^{(t_{i+1})}$ on $[x_j, x_{j+1}], \quad j = 0, 1, \ldots, N$ and subsequently $k_i = \min\{\max_j[slope(\psi_j^{(i+1)})], 10\}$.
**Step6:** Set $\Delta t_i = \frac{t_{i+1} - t_i}{k_i}$.
**Step7:** Run ALSM $k_i$ times with $t_0 = t_i$, $T_f = t_{i+1}$, and $\Delta t = \Delta t_i$, then again calculate $\psi^{(t_{i+1})}, X^{(t_{i+1})}$ and $\tilde{U}^{(t_{i+1})}$.
**Step8:** Set $i = i + 1$.
**Step9:** If $t_i \leq T_f$ go to step4, else break.

# 4 Numerical Experiment

In this section, we implement the present work for two numerical examples where $u(x, t)$ is a given function. We use the arclength monitor function. In order to obtain an accurate and non-oscillatory solution, it is necessary to smooth the mesh points. Following [5], we have applied smoothed monitor function as below:

$$\tilde{\rho}_i = \frac{\sum_{k=i-ip}^{i+ip} \rho_k (\frac{\gamma}{\gamma+1})^{|k-i|}}{\sum_{k=i-ip}^{i+ip} (\frac{\gamma}{\gamma+1})^{|k-i|}},$$

where $ip$ is a nonnegative integer and $\gamma$ is a positive constant. In this paper we select the mentioned formula for smoothing monitor function with $ip = 1$ and $\gamma = 1$. We apply MMPDE5 and MMPDE6 in the evolution equation of the level set function for obtaining nodes velocity.

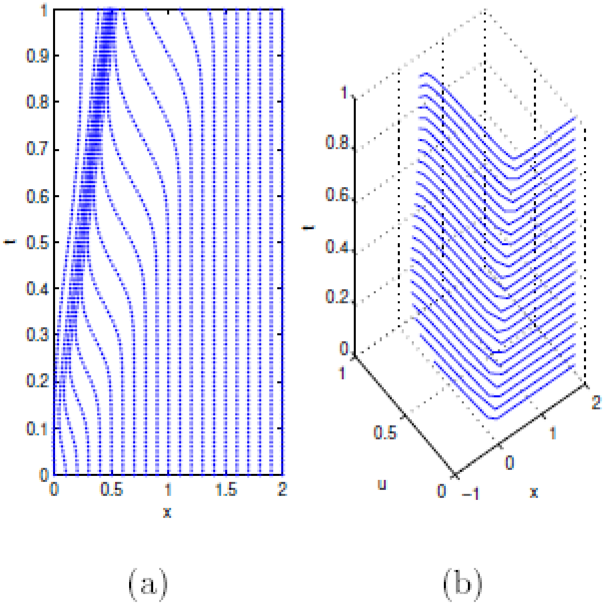


Fig. 1: The mesh trajectory and solution of the first example for $0 \leq t \leq 1.0$ with adaptive level set method(MMPDE5) of 21 mesh point,($\mu = 0.5, \lambda = 0.4, \varepsilon = 0.01, \beta = 0$).

**Example 4.1.** Consider the Burger's equation as first example,

$$u_t + uu_x - \varepsilon u_{xx} = 0,$$

where its exact solution is:

$$u(x,t) = \frac{\mu + \lambda + (\mu - \lambda)e^{\frac{\lambda}{\varepsilon}(x-\mu t-\beta)}}{1 + e^{\frac{\lambda}{\varepsilon}(x-\mu t-\beta)}}.$$

We select $\mu = 0.5, \lambda = 0.4, \varepsilon = 0.01$ and $\beta = 0$. This problem is characterized by moving discontinuities(specially when $\varepsilon$ is very small), that means the discontinuities move in time, and so the solution at a particular point in space

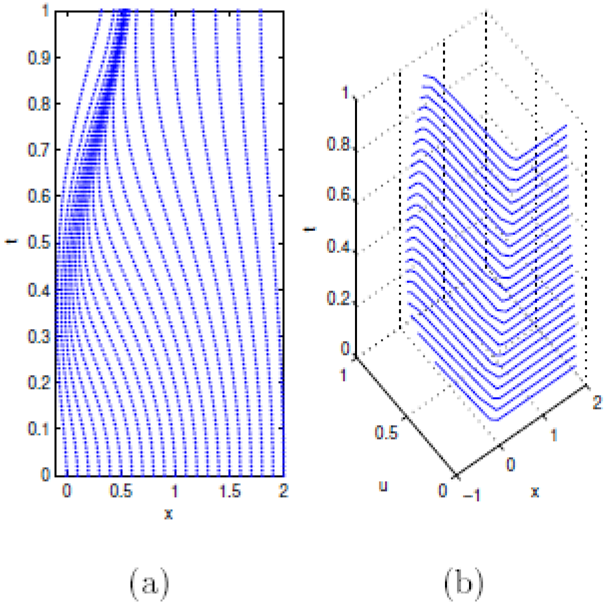


Fig. 2: The mesh trajectory and solution of the first example for $0 \leq t \leq 1.0$ with adaptive level set method(MMPDE6) of 21 mesh point,($\mu = 0.5, \lambda = 0.4, \varepsilon = 0.01, \beta = 0$).

Table 1: The error of ALSM(with MMPDE5) for the first example at $t = 1.0$, obtained by the arclength monitor function with $\alpha = 0$(uniform mesh) and $\alpha = 2$(moving mesh)without LTSR and with LTSR, ($\mu = 0.5, \lambda = 0.4, \varepsilon = 0.01$ and $\beta = 0$).

| MMPDE | Time stepping | Type of mesh | N | $L^\infty$-error | CPU time |
|---|---|---|---|---|---|
| | | | 21 | 0.1924 | 0.504 |
| 5 | without LTSR | fixed mesh | 31 | 0.0782 | 0.612 |
| | | ($\alpha = 0$) | 41 | 0.0564 | 0.8807 |
| | | | 21 | 0.0711 | 0.7548 |
| 5 | without LTSR | Moving mesh | 31 | 0.0602 | 0.8315 |
| | | ($\alpha = 2$) | 41 | 0.0587 | 0.9812 |
| | | | 21 | 0.0510 | 0.917 |
| 5 | with LTSR | Moving mesh | 31 | 0.0347 | 1.019 |
| | | ($\alpha = 2$) | 41 | 0.0298 | 1.079 |

can change very rapidly. The solution of such a problem on a fixed uniform spatial mesh, needs very small time step in order to have sufficient accuracy, but using the adaptive mesh for finding the solution of this problem improves both the accuracy and the efficiency.

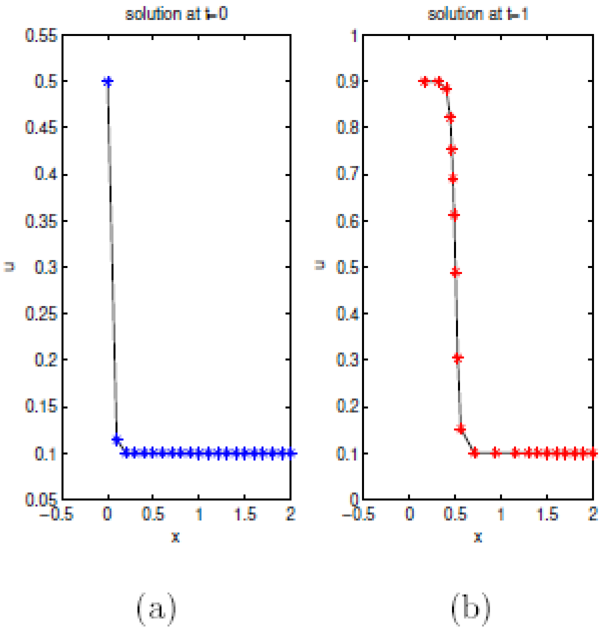


Fig. 3: The solution at $t = 0$ with uniform mesh and at $t = 1$ with adaptive level set method(MMPDE5) of 21 mesh point for the first example, ($\mu = 0.5, \lambda = 0.4, \varepsilon = 0.01, \beta = 0$).

Table 2: The error of ALSM(with MMPDE6) for the first example at $t = 1.0$, obtained by the arclength monitor function with $\alpha = 0$ (uniform mesh) and $\alpha = 2$(moving mesh)without LTSR and with LTSR, ($\mu = 0.5, \lambda = 0.4, \varepsilon = 0.01$ and $\beta = 0$).

| MMPDE | Time stepping | Type of mesh | N | $L^\infty$-error | CPU time |
|---|---|---|---|---|---|
| | | | 21 | 0.0645 | 0.8815 |
| 6 | without LTSR | fixed mesh | 31 | 0.0578 | 0.9205 |
| | | ($\alpha = 0$) | 41 | 0.0512 | 0.9876 |
| | | | 21 | 0.0441 | 0.031 |
| 6 | without LTSR | Moving mesh | 31 | 0.0315 | 1.108 |
| | | ($\alpha = 2$) | 41 | 0.0274 | 1.230 |

For this reason, we have used new adaptive mesh for this problem and in order to demonstrate the efficiency of the adaptive level set method (ALSM)(which has combined with MMPDE5), and also to compare ALSM with or without LTSR, some results are presented in Table1. These results show $L^\infty$-error and CPU times for different number of mesh points. These results certify higher accuracy of the mentioned method(ALSM) in comparison with using uniform mesh. Besides, it shows that using the adaptive method

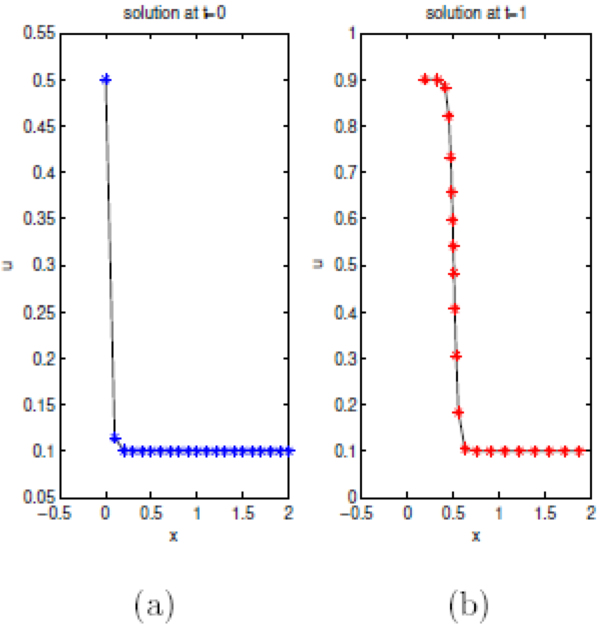


Fig. 4: The solution at $t = 0$ with uniform mesh and at $t = 1$ with adaptive level set method(MMPDE6) of 21 mesh point for the first example, ($\mu = 0.5, \lambda = 0.4, \varepsilon = 0.01, \beta = 0$).

with LTSR gives better results. Similar information, are given in Table 2 when the MMPDE6 is applied for calculating nodes velocity. Figure 1(a) shows mesh trajectories for $0 \leq x \leq 2, 0 \leq t \leq 1.0$ which has been derived by using the new method with MMPDE5, also the solution of PDE, $u$, has been plotted on the moving mesh for $0 \leq t \leq 1.0$ in Figure 1(b). The spatial domain is divide into 21 mesh points. Like above, mesh trajectories and solution of Example 4.1 have been plotted by using the new method with MMPDE6 in Figure 2. Also in Figure 3 we plotted the solution of PDE at $t = 0$ on a uniform initial mesh and at $t = 1$ on a moving mesh with MMPDE5 and similarly, the solution at $t = 0$ and $t = 1$ has been plotted with MMPDE6 in Figure 4.

For demonstrating the efficiency of our method we compare this method with another moving mesh method. For this purpose, we consider moving element free Petrov-Galerkin viscous method, MEFP-GVP, where introduced in [2], for comparing under equal conditions we consider the parameters $\mu, \lambda, \varepsilon$ and $\beta$ in the Burger's equation according to [2], ($\mu = 0.5, \lambda = 0.4, \varepsilon = 1/15$ and $\beta = 0.16$). Also, we solve this equation for $x \in [0, 1]$ and $t \in [0, 0.51]$ again. With above conditions and with $N = 7$ grid points the $L^\infty$-error is 0.03 by

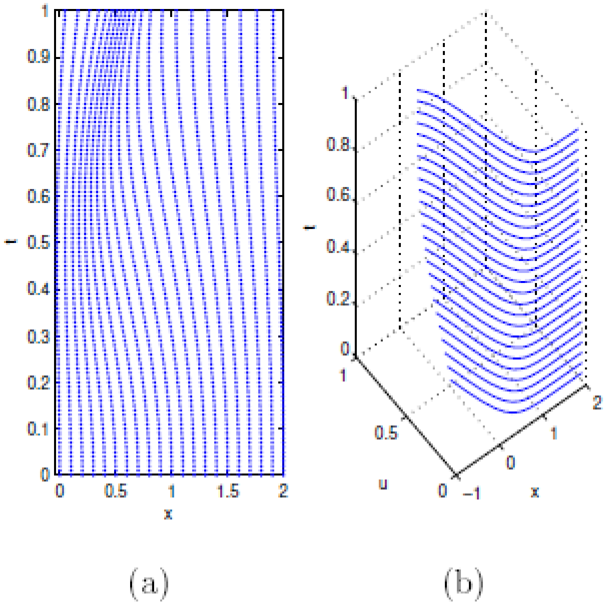


Fig. 5: The mesh trajectory and solution of the first example for $0 \leq t \leq 1.0$ with adaptive level set method(MMPDE5) of 21 mesh point,($\mu = 0.5, \lambda = 0.4, \varepsilon = 1/15, \beta = 0$).

MEFP-GVM but in ALSM we have $L^{\infty}$- error= 0.01. This result shows the preference of our method. In addition, the CPU time in our method is very smaller than the MEFP-GVM. Also we plotted grid motion and the solution for new conditions for 21 grid points in Figure 5.

Table 3: The error of ALSM(with MMPDE5) for the second example at $t = 1.0$, obtained by the arclength monitor function ($\alpha = 2$) without LTSR and with LTSR.

| MMPDE | Time stepping | N | $L^{\infty}$-error | CPU time |
|---|---|---|---|---|
| | | 21 | 0.0060 | 0.7213 |
| 5 | without LTSR | 31 | 0.0052 | 0.8507 |
| | | 41 | 0.0050 | 0.9759 |
| | | 21 | 0.0051 | 0.906 |
| 5 | without LTSR | 31 | 0.0032 | 1.015 |
| | | 41 | 0.0029 | 1.076 |

**Example 4.2.** Consider the equation

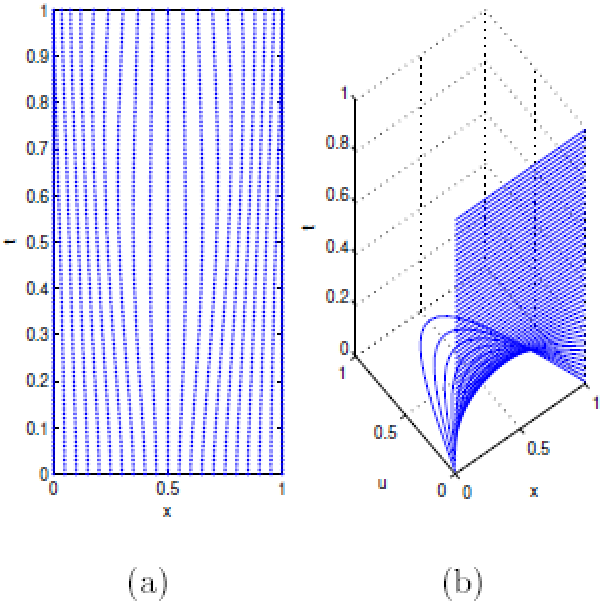


Fig. 6: The mesh trajectory and solution of the second problem for $0 \leq x \leq 1$ and $0 \leq t \leq 1.0$ with adaptive level set method(MMPDE5) of 21 mesh point.

Table 4: The error of ALSM(with MMPDE6) for the second example at $t = 1.0$, obtained by the arclength monitor function ($\alpha = 2$) without LTSR and with LTSR.

| MMPDE | Time stepping | N | $L^{\infty}$-error | CPU time |
|---|---|---|---|---|
| | | 21 | 0.0023 | 0.8706 |
| 6 | without LTSR | 31 | 0.0015 | 0.9844 |
| | | 41 | 0.0009 | 1.029 |
| | | 21 | 0.0048 | 1.025 |
| 6 | without LTSR | 31 | 0.0030 | 1.098 |
| | | 41 | 0.0024 | 1.221 |

$$u_t = u_{xx},$$

with the exact solution:

$$u(x,t) = e^{-\pi^2 t} \sin \pi x$$

This problem has been used in [3] as a numerical example and also in [15] to study the moving mesh with variable relaxation time.
Since $u_x(x,t) \to 0$ in the limit as $t \to +\infty$, then for typical arclength monitor function

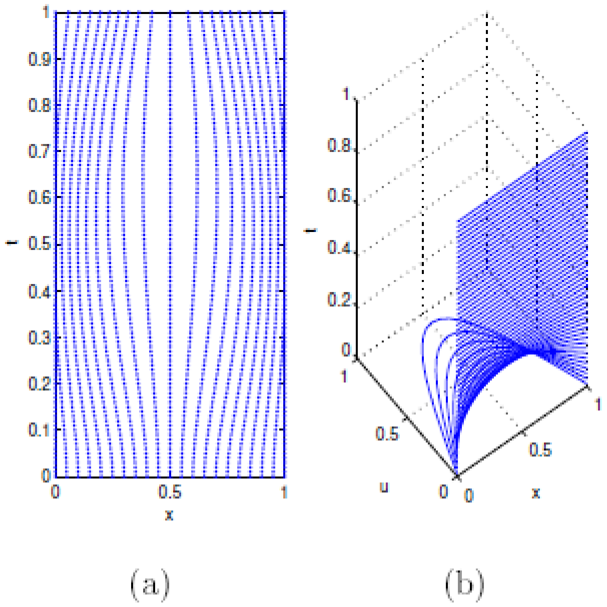


Fig. 7: The mesh trajectory and solution of the second problem for $0 \leq x \leq 1$ and $0 \leq t \leq 1.0$ with adaptive level set method(MMPDE6) of 21 mesh point.

$$M = \sqrt{1 + \alpha u_x^2}$$

we have $M(x,t) \to 1$ as $t \to +\infty$ and so the equidistributed mesh should tend to a uniform mesh in space.

Similar to Tables 1 and 2, some results about the second example are given in Tables 3 and 4 which certify the quality of the new mesh adaptive algorithm. Figure 6(a) shows the mesh trajectories for the above problem using adaptive level set method with MMPDE5 and in Figure 6(b), the numerical solution of PDE is plotted for $0 \leq t \leq 1.0$ by the new method with MMPDE6 have been plotted. We also plotted the solution at $t = 0$ on uniform initial mesh and at the times $t = 0.3, 0.5$ and $t = 1$ on moving mesh in Figure 8. As we expect, the adaptive mesh at $t = 1$ tends towards uniform mesh.

In the second example, because of the smoothness of the solution there is not any major difference between the uniform and the adaptive mesh and also between the adaptive mesh with MMPDE5 and MMPDE6. In this example, ALSM with LTSR does not yield any different result from ALSM without LTSR.

**Example 4.3.** We use the following Burger's equation as the third example,

$$u_t + uu_x - \varepsilon u_{xx} = 0, \quad 0 < x < x_R.$$

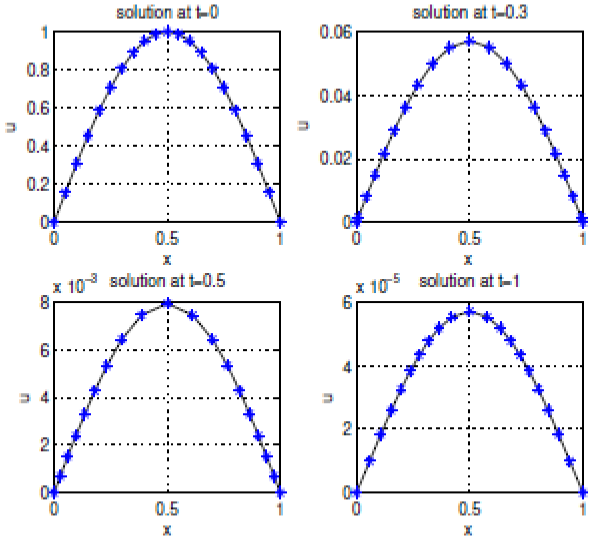


Fig. 8: The solution of the second PDE at $t = 0, 0.3, 0.5$ and $t = 1$ with its corresponding mesh nodes.

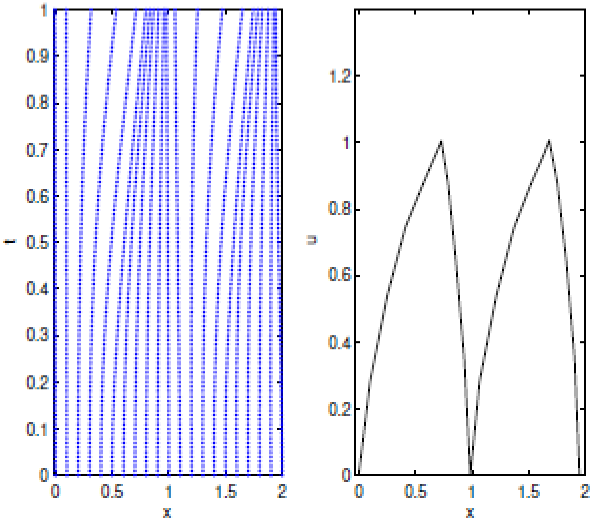


Fig. 9: The mesh trajectory and the computed solution of the third example up to $t = 1$ for $0 < x < 2$ with adaptive level set method(MMPDE5) of 21 mesh points.

The initial condition is considered as follows [7]:

$$u(x,0) = |\sin(\frac{2\pi x}{x_R})|.$$

The boundary conditions are chosen in the form

$$u(0,t) = u(x_R,t) = 0.$$

We select $\varepsilon = 10^{-5}$ and $x_R = 2$ in this example. This example has been solved by the presented method up to $t = 1$.
The mesh trajectory has been plotted along with the nodes of moving grid with $N = 21$ mesh points in different times in Figure 9. Also the computed solution of the third example has been plotted in Figure 9. It is clear that the mesh moves correctly, because the solution has large variations about $x = 0.5$ and $x = 2$ and the mesh also concentrates around these areas.

## 5 Conclusion

In this paper, we have developed a static moving mesh method based on the level set approach for solving one dimensional time dependent PDEs, such that for representing the nodes of adaptive mesh, the level set equation is used. Also, we proposed a strategy for local time step refinement where the local time step is selected by the level set function. This adaptive method is among the static moving mesh. We plan to investigate the dynamical moving mesh method based on the level set function.

## References

1. Aslam, T., *A level set algorithm for tracking discontinuities in hyperbolic conservation laws*,I:*Scalar equations*, J. Comput. Phys. **167**, (2001), 413-438.
2. Ghorbani, M. and Soheili, A.R.,*Moving element free Petrov Galerkian viscous method,special issue on meshless methodes*, Journal of the Chinese Institue of engineers, **27**, (2004), 473-479.
3. Huang, W., Ren, Y. and Russell, R., *Moving mesh partial differential equations(MMPDEs)based on the equiditribution principle*, SIAM J.Number. Anal. **31**, (1994), 709-730.
4. Huang, W., Ren, Y. and Russell, R., *Moving mesh methods based on moving mesh partial differential equations*, J. Compute. Phys., **113**, (1994), 279-290.
5. Jin, S. and Osher, S.,*A level set method for the computation of multivalued solutions to quasi-linear hyperbolic PDEs and Hamilton-Jacobi equations*, SIAM .Numer. Analys. **28**, (1991), 907-921.
6. Liao, G., Liu, F., Pena, G.D., peng, D. and Osher, S., *Level-set-based deformation methods for adaptive grids*, J.Compute. Phys., **159**, (2000), 103-122.

7. Mazhukin, A.V., *Dynamic adaptation in convection-diffusion equations*, Comput. method. Appl. Math, **8**, (2008), 171-186.
8. Mazhukin, A.V. and Chuiko, M.M., *Solution of the multi-interface Stefan problem by the method of dynamic adaptation*, Compute. method. Appl. Math, **2**, (2002), 283-294.
9. Osher, S. and Fedkiw, R., *Level set methods and dynamic implicit surfaces*, Springer-Verlag, NY, (2002)
10. Osher, S. and Sanders, R., *Numerical approxiamtion to nonlinear conservation laws with locally varying time and spaces grid* , Math. Compute. **41**, (1983), 321-336.
11. Osher, S. and Sethian, J., *Front propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations*, J. Compute. Phys. **79**, 12 (1988).
12. Sethian, J.A.*Level set methods and fast marching methods*, New York: Cambridge University Press,(1999).
13. Soheili, A.R. and Ameri, M.A., *Adaptive grid based on geometric conservation law level set method for time dependent PDE* , Numerical methods for PDEs, **25**, (2009), 582-597.
14. Soheili, A.R. and Salahshour, S., *Moving mesh methods with local time step refinement for blow-up problems*, Appl.Math.Copmpute. **195**, (2008), 76-85.
15. Soheili, A.R. and Stockie, J.M.,*A moving mesh method with variable mesh relaxation time*, Appl.Number.Math. **58**, (2008), 249-263.
16. Stockie, J.M., Mackenzie, J.A. and Russell, R., *A moving mesh method for one dimensional hyperbolic conservation laws*, SIAM. J.S.Compute. **22**, (2001), 1791-1831.
17. Tan, H.Z. and Tang, T., *Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws*, SIAM. J.Numer. Anal. **41**, (2003), 487-515.
18. Tan, Z., Zhang, Z., Huang, Y. and Tang, T.,*Moving mesh methods with locally varying time step*, J. Comp. Phys. **200**, (2004), 347-367.
19. Tasi, Y.H.R. and Giga, Y. and Osher, S.,*A level set approach for computing discontinuous solutions of Hamilton-Jacobi equations*, J. Math. Compute. **72**, (2002), 159-181.

Iranaian J. Numerical Analysis and Optimization
Vol. 3, No. 1, (2013), 31-44

# Accelerated normal and skew-Hermitian splitting methods for positive definite linear systems

F. Toutounian and D. Hezari

**Abstract**

For solving large sparse non-Hermitian positive definite linear equations, Bai et al. proposed the Hermitian and skew-Hermitian splitting methods (HSS). They recently generalized this technique to the normal and skew-Hermitian splitting methods (NSS). In this paper, we present an accelerated normal and skew-Hermitian splitting methods (ANSS) which involve two parameters for the NSS iteration. We theoretically study the convergence properties of the ANSS method. Moreover, the contraction factor of the ANSS iteration is derived. Numerical examples illustrating the effectiveness of ANSS iteration are presented.

**Keywords:** Non-Hermitian matrix; Normal matrix; Hermitian matrix; Skew-Hermitian matrix; Splitting iteration method.

## 1 Introduction

Many problems in scientific computation give rise to solving the linear system

$$Ax = b, \tag{1}$$

with $A \in \mathbb{C}^{n\times n}$ a large non-Hermitian positive definite matrix and $x, b \in \mathbb{C}^n$. We observe that the coefficient matrix $A$ naturally possesses the Hermitian/skew-Hermitian (HS) splitting

$$A = H + S,$$

where

$$H = \frac{1}{2}(A + A^*) \quad \text{and} \quad \mathrm{S} = \frac{1}{2}(\mathrm{A} - \mathrm{A}^*),$$

with $A^*$ being the conjugate transpose of $A$. Bai et al. [2] presented the HSS iteration method: Given an initial guess $x^{(0)} \in \mathbb{C}^n$, for $k = 0, 1, 2, \ldots$, until

F. Toutounian
Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: toutouni@math.um.ac.ir

Davood Hezari
Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: hezari_ h@yahoo.com

$\{x^{(k)}\}$ converges, compute

$$\begin{cases} (\alpha I + H)x^{(k+\frac{1}{2})} = (\alpha I - S)x^{(k)} + b, \\ (\alpha I + S)x^{(k+1)} = (\alpha I - H)x^{(k+\frac{1}{2})} + b, \end{cases} \tag{2}$$

where $\alpha$ is a given positive constant. They have also proved that for any positive $\alpha$, the HSS method converges unconditionally to the unique solution of the system of linear equations.

Moreover, based on the HS splitting, Li et al. [5] presented the asymmetric Hermitian/skew-Hermitian splitting (AHSS) iteration method: Given an initial guess $x^{(0)} \in \mathbb{C}^n$, for $k = 0, 1, 2, \ldots$, until $\{x^{(k)}\}$ converges, compute

$$\begin{cases} (\alpha I + H)x^{(k+\frac{1}{2})} = (\alpha I - S)x^{(k)} + b, \\ (\beta I + S)x^{(k+1)} = (\beta I - H)x^{(k+\frac{1}{2})} + b, \end{cases} \tag{3}$$

where $\alpha$ is a given nonnegative constant and $\beta$ is a given positive constant. They proved that if the coefficient matrix A is positive definite (Hermitian or non-Hermitian) the AHSS iteration converges to the unique solution of linear system (1) with any given nonnegative $\alpha$, if $\beta$ is restricted to an appropriate region.

Bai et al. [1] recently generalized the HS splitting to the normal/skew-Hermitian (NS) splitting

$$A = N + S, \tag{4}$$

where $N \in \mathbb{C}^{n\times n}$ is a normal matrix and $S \in \mathbb{C}^{n\times n}$ is a skew-Hermitian matrix, and obtained the following normal/skew-Hermitian splitting (NSS) method to iteratively compute a reliable and accurate approximate solution for the system of linear equations (1):
*The NSS iteration method:* Given an initial guess $x^{(0)} \in \mathbb{C}^n$. For $k = 0, 1, 2 \ldots$ until $\{x^{(k)}\}$ converges, compute

$$\begin{cases} (\alpha I + N)x^{(k+\frac{1}{2})} = (\alpha I - S)x^{(k)} + b, \\ (\alpha I + S)x^{(k+1)} = (\alpha I - N)x^{(k+\frac{1}{2})} + b, \end{cases} \tag{5}$$

where $\alpha$ is a given positive constant. They have also proved that for any positive $\alpha$ the NSS method converges unconditionally to the unique solution of the system of linear equations.

In this paper, we introduce two constants for the NSS iteration and present a different approach to solve Eq. (1), called the *accelerated normal and skew-Hermitian splitting iteration*, shortened to the ANSS iteration. Moreover, theoretical analysis shows that if the coefficient matrix $A$ is positive definite (Hermitian or non-Hermitian) the ANSS method can converge to the unique solution of the linear system (1) with any given nonnegative $\alpha$, if $\beta$ is restricted to an appropriate region. In addition the upper bound of the contraction factor of the ANSS iteration is dependent on the choice of $\alpha$ and $\beta$, the spectrum of the normal matrix $N$ and the singular-values of the skew-

Hermitian, but it is not dependent on the eigenvectors of the matrices $N$, $S$ and $A$.

The organization of this paper is as follows. In section 2, we establish the ANSS iteration and study its convergence properties. Numerical experiments are presented in section 3 to show the effectiveness of our method. Finally, in section 4, some concluding remarks are given.

## 2 The ANSS Method

Throughout the paper, the non-Hermitian matrix $A \in \mathbb{C}^{n\times n}$ ( i.e. $A \neq A^*$) is positive definite if its Hermitian part is Hermitian positive definite.

Based on the NSS iteration (5), in this paper we present a new approach to solve the system of linear equations (1), called the ANSS iteration, and it is as follows.

*The ANSS iteration method:* Given an initial guess $x^{(0)} \in \mathbb{C}^n$, for $k = 0, 1, 2 \ldots$ until $\{x^{(k)}\}$ converges, compute

$$\begin{cases} (\alpha I + N)x^{(k+\frac{1}{2})} = (\alpha I - S)x^{(k)} + b, \\ (\beta I + S)x^{(k+1)} = (\beta I - N)x^{(k+\frac{1}{2})} + b, \end{cases} \tag{6}$$

where $\alpha$ is a given nonnegative constant and $\beta$ is a given positive constant.

The ANSS iteration alternates between the normal matrix $N$ and the skew-Hermitian matrix $S$. In fact, we can reverse the roles of the matrices $N$ and $S$ in the above ANSS iteration so that we may first solve the system of linear equations with coefficient matrix $\beta I + S$ and then solve the system of linear equations with coefficient matrix $\alpha I + N$.

Note that both $\alpha I + N$ and $\beta I + S$ are normal matrices. Therefore, the linear systems with the coefficient matrices $\alpha I + N$ and $\beta I + S$ may be solved accurately and efficiently by some Krylov subspace iteration methods, e.g. GMRES. It is known that the GMRES method naturally reduces to an iterative process of the three-term recurrence. See [4, 3] for other iteration methods about solving large sparse normal system of linear equations.

In matrix-vector form, the ANSS iteration method can be equivalently rewritten as

$$x^{(k+1)} = M(\alpha,\beta)x^{(k)} + G(\alpha,\beta)b, \qquad k = 0, 1, 2, \ldots, \tag{7}$$

where

$$M(\alpha,\beta) = (\beta I + S)^{-1}(\beta I - N)(\alpha I + N)^{-1}(\alpha I - S) \tag{8}$$

and

$$G(\alpha,\beta) = (\alpha + \beta)(\beta I + S)^{-1}(\alpha I + N)^{-1}.$$

Here, $M(\alpha,\beta)$ is the iteration matrix of the ANSS iteration. In fact, (7) may also result from the splitting

$$A = B(\alpha,\beta) - C(\alpha,\beta)$$

of the coefficient matrix $A$, with

$$\begin{cases} B(\alpha,\beta) = \dfrac{1}{\alpha+\beta}(\alpha I + N)(\beta I + S) \\ \\ C(\alpha,\beta) = \dfrac{1}{\alpha+\beta}(\beta I - N)(\alpha I - S). \end{cases} \tag{9}$$

Obviously

$$M(\alpha,\beta) = B(\alpha,\beta)^{-1}C(\alpha,\beta) \quad \text{and} \quad G(\alpha,\beta) = B(\alpha,\beta)^{-1}.$$

To study the convergence properties of the ANSS iteration and derive the upper bound of the contraction factor, we first represent the following lemmas.

**Lemma 2.1.** *Let $\alpha$ be a nonnegative constant and $\beta$ be a positive constant. If $(\gamma,\eta) \in \Omega$, where $\Omega = [\gamma_{\min},\gamma_{\max}] \times [\eta_{\min},\eta_{\max}]$, $\gamma_{\min} > 0$ and $\eta_{\min} \geq 0$, then*

$$f(\alpha,\beta) \equiv \max_{(\gamma,\eta)\in\Omega} \left\{ \frac{(\beta-\gamma)^2+\eta^2}{(\alpha+\gamma)^2+\eta^2} \right\}$$

$$= \begin{cases} \displaystyle\max_{\gamma_{\min}\leq\gamma\leq\frac{\beta-\alpha}{2}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\min}^2}{(\alpha+\gamma)^2+\eta_{\min}^2} \right\} & \text{for} \quad \gamma_{\min} \leq \dfrac{\beta-\alpha}{2} \\ \\ \displaystyle\max_{\gamma_{\min}\leq\gamma\leq\gamma_{\max}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\max}^2}{(\alpha+\gamma)^2+\eta_{\max}^2} \right\} & \text{for} \quad \dfrac{\beta-\alpha}{2} \leq \gamma_{\min}. \end{cases} \tag{10}$$

**Proof**. Let us define the function $g(\eta)$ by

$$g(\eta) = \frac{(\beta-\gamma)^2+\eta^2}{(\alpha+\gamma)^2+\eta^2}.$$

Differentiation gives

$$g'(\eta) = \frac{2\eta(\alpha+\beta)(\alpha-\beta+2\gamma)}{[(\alpha+\gamma)^2+\eta^2]^2}.$$

Since $(\alpha+\beta) > 0$, it follows that the function $g(\eta)$ is an increasing function if $\gamma \geq \dfrac{\beta-\alpha}{2}$ and is a decreasing function if $\gamma \leq \dfrac{\beta-\alpha}{2}$.

If $\dfrac{\beta-\alpha}{2} \leq \gamma_{\min}$, then for all $\gamma$ satisfying $\gamma_{\min} \leq \gamma \leq \gamma_{\max}$, we have $\dfrac{\beta-\alpha}{2} \leq \gamma$. So, for $\gamma_{\min} \leq \gamma \leq \gamma_{\max}$, the function $g(\eta)$ is an increasing function, and

$$f(\alpha,\beta) = \max_{\gamma_{\min}\leq\gamma\leq\gamma_{\max}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\max}^2}{(\alpha+\gamma)^2+\eta_{\max}^2} \right\} \quad \text{if} \quad \gamma_{\min} \leq \frac{\beta-\alpha}{2}. \tag{11}$$

If $\gamma_{\min} \leq \dfrac{\beta-\alpha}{2}$, then, by using $\beta+\alpha > 0$, for all $\gamma$ satisfying $\gamma_{\min} \leq \dfrac{\beta-\alpha}{2} \leq \gamma$, we obtain $(\beta-\gamma)^2 \leq (\alpha+\gamma)^2$, which implies that

$$\frac{(\beta-\gamma)^2+\eta^2}{(\alpha+\gamma)^2+\eta^2} \leq 1.$$

Similarly, for all $\gamma$ satisfying $\gamma_{\min} \leq \gamma \leq \dfrac{\beta-\alpha}{2}$, we obtain $(\beta-\gamma)^2 \geq (\alpha+\gamma)^2$ and

$$\frac{(\beta-\gamma)^2+\eta^2}{(\alpha+\gamma)^2+\eta^2} \geq 1.$$

Therefore,

$$f(\alpha,\beta) = \max_{\gamma_{\min}\leq\gamma\leq\frac{\beta-\alpha}{2},\eta_{\min}\leq\eta\leq\eta_{\max}} \left\{ \frac{(\beta-\gamma)^2+\eta^2}{(\alpha+\gamma)^2+\eta^2} \right\}, \quad \text{if} \quad \gamma_{\min} \leq \frac{\beta-\alpha}{2}$$

From the fact that, for $\gamma_{\min} \leq \gamma \leq \dfrac{\beta-\alpha}{2}$, the function $g(\eta)$ is a decreasing function, we can conclude that

$$f(\alpha,\beta) = \max_{\gamma_{\min}\leq\gamma\leq\frac{\beta-\alpha}{2}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\min}^2}{(\alpha+\gamma)^2+\eta_{\min}^2} \right\}, \quad \text{if} \quad \gamma_{\min} \leq \frac{\beta-\alpha}{2} \tag{12}$$

Therefore (11) and (12) immediately result relation (10). □

**Lemma 2.2.** *Let $\alpha$ be a nonnegative constant and $\beta$ be a positive constant. If $\dfrac{\beta-\alpha}{2} \leq \gamma_{\min}$, where $0 < \gamma_{\min}$, then*

$$\max_{\gamma_{\min}\leq\gamma\leq\gamma_{\max}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\max}^2}{(\alpha+\gamma)^2+\eta_{\max}^2} \right\} = \max\left\{ \frac{(\beta-\gamma_{\min})^2+\eta_{\max}^2}{(\alpha+\gamma_{\min})^2+\eta_{\max}^2}, \frac{(\beta-\gamma_{\max})^2+\eta_{\max}^2}{(\alpha+\gamma_{\max})^2+\eta_{\max}^2} \right\} \tag{13}$$

**Proof**. Let us define the function $g(\gamma)$ by

$$g(\gamma) = \frac{(\beta-\gamma)^2+\eta_{\max}^2}{(\alpha+\gamma)^2+\eta_{\max}^2}.$$

Differentiation gives

$$g'(\gamma) = \frac{-2(\alpha+\beta)\left[-\gamma^2 + (\beta-\alpha)\gamma + \beta\alpha + \eta_{\max}^2\right]}{\left[(\alpha+\gamma)^2 + \eta_{\max}^2\right]^2}.$$

The smallest root of $g'(\gamma)$ is negative and is not in the interval $[\gamma_{\min}, \gamma_{\max}]$. The largest root of $g'(\gamma)$ is

$$\gamma_1 = \frac{(\beta-\alpha) + \sqrt{(\beta-\alpha)^2 + 4(\beta\alpha + \eta_{\max}^2)}}{2}.$$

By simple computation, we can show that this root is a minimum point for the function $g(\gamma)$. Hence (13) holds and the proof of Lemma is completed. □

**Lemma 2.3.** *Let $\alpha$ be a nonnegative constant and $\beta$ be a positive constant. If $0 < \gamma_{\min} \le \dfrac{\beta-\alpha}{2}$, then*

$$f(\alpha,\beta) = \max_{\gamma_{\min} \le \gamma \le \frac{\beta-\alpha}{2}} \left\{ \frac{(\beta-\gamma)^2 + \eta_{\min}^2}{(\alpha+\gamma)^2 + \eta_{\min}^2} \right\} = \frac{(\beta-\gamma_{\min})^2 + \eta_{\min}^2}{(\alpha+\gamma_{\min})^2 + \eta_{\min}^2}.$$

**Proof**. Let us define the function $h(\gamma)$ by

$$h(\gamma) = \frac{(\beta-\gamma)^2 + \eta_{\min}^2}{(\alpha+\gamma)^2 + \eta_{\min}^2}.$$

Differentiation gives

$$h'(\gamma) = \frac{-2(\alpha+\beta)\left[(\beta-\gamma)(\alpha+\gamma) + \eta_{\min}^2\right]}{\left[(\alpha+\gamma)^2 + \eta_{\min}^2\right]^2}$$

Since $(\alpha+\beta) > 0$ and $\gamma \le \beta$, for all $\gamma$ satisfying $\gamma_{\min} \le \gamma \le \gamma_{\max}$ and $\gamma \le \dfrac{\beta-\alpha}{2}$, we have $h'(\gamma) < 0$. Thus

$$f(\alpha,\beta) = \frac{(\beta-\gamma_{\min})^2 + \eta_{min}^2}{(\alpha+\gamma_{\min})^2 + \eta_{\min}^2}.$$

□

The following theorem describes the convergence property of the ANSS iteration.

**Theorem 2.1.** *Let $A \in \mathbb{C}^{n\times n}$ be a positive definite matrix, $N \in \mathbb{C}^{n\times n}$ be a normal matrix and $S \in \mathbb{C}^{n\times n}$ be a skew-Hermitian matrix such that $A = N + S$, and $\alpha$ be a nonnegative constant and $\beta$ be a positive constant. Then the spectral radius $\rho(M(\alpha,\beta))$ of the iteration matrix $M(\alpha,\beta)$ of the ANSS iteration is bounded by*

$$\delta(\alpha,\beta) \equiv \max_{\sigma_j\in\sigma(S)} \frac{\sqrt{\alpha^2+\sigma_j^2}}{\sqrt{\beta^2+\sigma_j^2}} \max_{\gamma_j+i\eta_j\in\lambda(N)} \sqrt{\frac{(\beta-\gamma_j)^2+\eta_j^2}{(\alpha+\gamma_j)^2+\eta_j^2}} \tag{14}$$

*where $\lambda(N)$ is the spectral set of $N$ and $\sigma(S)$ is the singular-value set of $S$. Let $\gamma_{\min}$ and $\gamma_{\max}$, $\eta_{\min}$ and $\eta_{\max}$ be the lower and the upper bound of the real, the absolute values of the imaginary parts of the eigenvalues of the matrix $N$, respectively, and $\sigma_{\min}$, $\sigma_{\max}$ be the lower and the upper bound of the singular-value set of the matrix $S$, respectively. Then $\delta(\alpha,\beta)<1$ if one of the following conditions holds:*

*(a) Any given parameter $\alpha$ and $\beta$ satisfies*

$$\max\left\{\frac{\alpha(\gamma_{\min}^2+\eta_{\max}^2)}{2\alpha\gamma_{\min}+\gamma_{\min}^2+\eta_{\max}^2}, \frac{\alpha(\gamma_{\max}^2+\eta_{\max}^2)}{2\alpha\gamma_{\max}+\gamma_{\max}^2+\eta_{\max}^2}\right\} < \beta \le \alpha+2\gamma_{\min}$$

*(b) Any given parameter $\alpha$ and $\beta$ satisfies*

$$\alpha+2\gamma_{\min}<\beta$$

*if $\sigma_{\max} \le \sqrt{\gamma_{\min}+\eta_{\min}+2\gamma_{\min}\alpha}$.*
*(c) Any given parameter $\alpha$ and $\beta$ satisfies*

$$\alpha+2\gamma_{\min}<\beta\le\frac{\alpha(\gamma_{\min}^2+\eta_{\min}^2-\sigma_{\max}^2)-2\sigma_{\max}^2\gamma_{\min}}{\gamma_{\min}^2+\eta_{\min}^2-\sigma_{\max}^2+2\alpha\gamma_{\min}}$$

*if $\sigma_{\max} \ge \sqrt{\gamma_{\min}+\eta_{\min}+2\gamma_{\min}\alpha}$.*

**Proof**. By the similarity invariance of the matrix spectrum, we have

$$\begin{aligned}\rho(M(\alpha,\beta)) &= \rho((\beta I-N)(\alpha I+N)^{-1}(\alpha I-S)(\beta I+S)^{-1})\\ &\le \|(\beta I-N)(\alpha I+N)^{-1}\|_2\|(\alpha I-S)(\beta I+S)^{-1}\|_2.\end{aligned}$$

Letting $Q(\alpha,\beta)=(\alpha I-S)(\beta I+S)^{-1}$ and noting that $S^*=-S$, we have

$$\begin{aligned}Q(\alpha,\beta)^*Q(\alpha,\beta) &= \left[(\alpha I-S)(\beta I+S)^{-1}\right]^*\left[(\alpha I-S)(\beta I+S)^{-1}\right]\\ &= (\beta I-S)^{-1}(\alpha I+S)(\alpha I-S)(\beta I+S)^{-1}\\ &= (\alpha I-S)(\beta I+S)^{-1}(\beta I-S)^{-1}(\alpha I+S)\\ &= Q(\alpha,\beta)Q(\alpha,\beta)^*\end{aligned}$$

That is to say, $Q(\alpha,\beta)$ is a normal matrix. Therefore, there exists a unitary matrix $U\in\mathbb{C}^{n\times n}$ and a complex diagonal matrix $\bigwedge_q=\mathrm{diag}(\tilde\lambda_1,\tilde\lambda_2,\ldots,\tilde\lambda_n)\in\mathbb{C}^{n\times n}$ such that $Q(\alpha,\beta)=U^*\bigwedge_q U$. Suppose that $\tilde\lambda$ be an eigenvalue of $Q(\alpha,\beta)$ and $x$ be an associated eigenvector, we have

$$\begin{aligned} Q(\alpha,\beta)x &= \tilde{\lambda}x \\ (\alpha I - S)(\beta I + S)^{-1}x &= \tilde{\lambda}x \\ (\beta I + S)^{-1}(\alpha I - S)x &= \tilde{\lambda}x \\ (\alpha I - S)x &= \tilde{\lambda}(\beta I + S)x \end{aligned}$$

If $\tilde{\lambda} \neq -1$, then

$$Sx = \frac{\alpha - \tilde{\lambda}\beta}{1 + \tilde{\lambda}}x \tag{15}$$

If $\tilde{\lambda} = -1$, then

$$(\alpha + \beta)x = 0.$$

Since $\alpha + \beta > 0$, it implies $x = 0$, and this contradicts the definition of eigenvector. Therefore $\tilde{\lambda} = -1$ can not be an eigenvalue of $Q(\alpha,\beta)$.

From (15), $\dfrac{\alpha - \tilde{\lambda}\beta}{1 + \tilde{\lambda}}$ is an eigenvalue of $S$ and $x$ is an associated eigenvector. Since $S$ is a skew-Hermitian matrix, its eigenvalues are pure imaginary and thus of the form $i\tau_j$, $j = 1, \ldots, n$, where $\tau_j \in \mathbb{R}$, So

$$\tilde{\lambda_j} = \frac{\alpha - i\tau_j}{\beta + i\tau_j},$$

where $i\tau_j$ is an eigenvalue of $S$. Therefore

$$\|Q(\alpha,\beta)\|_2 = \|U^* \wedge_q U\|_2 = \| \wedge_q \|_2 = \max_{\sigma_j \in\ \sigma(S)} \frac{\sqrt{\alpha^2 + \sigma_j^2}}{\sqrt{\beta^2 + \sigma_j^2}}. \tag{16}$$

Because $N$ is a normal matrix, there exists a unitary matrix $V \in \mathbb{C}^{n\times n}$ and a complex diagonal matrix $\bigwedge_N = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \in \mathbb{C}^{n\times n}$ such that $N = V^* \bigwedge_N V$. Hence, we have

$$\begin{aligned} \|(\alpha I + N)^{-1}(\beta I - N)\|_2 &= \max_{\lambda_j \in\ \lambda(N)} \frac{\mid \beta - \lambda_j \mid}{\mid \alpha + \lambda_j \mid} \\ &= \max_{\lambda_j = \gamma_j + i\eta_j \in\ \lambda(N)} \sqrt{\frac{(\beta - \gamma_j)^2 + \eta_j^2}{(\alpha + \gamma_j)^2 + \eta_j^2}} \end{aligned} \tag{17}$$

Now, from (16) and (17), we see that

$$\rho(M(\alpha,\beta)) \leq \max_{\sigma_j \in\ \sigma(S)} \frac{\sqrt{\alpha^2 + \sigma_j^2}}{\sqrt{\beta^2 + \sigma_j^2}} \max_{\lambda_j = \gamma_j + i\eta_j \in\ \lambda(N)} \sqrt{\frac{(\beta - \gamma_j)^2 + \eta_j^2}{(\alpha + \gamma_j)^2 + \eta_j^2}}.$$

Then the bound for $\rho(M(\alpha,\beta))$ is given by (14).

To prove (a), we note that, if $\frac{\beta-\alpha}{2} \leq \gamma_{\min}$, then $\beta-\alpha \leq 2\gamma$ for $\gamma_{\min} \leq \gamma \leq \gamma_{\max}$. By using $0 < (\beta+\alpha)$, we obtain $(\beta-\gamma)^2 \leq (\alpha+\gamma)^2$. Thus, by Lemma 2.1, we have

$$f(\alpha,\beta) = \max_{\gamma_{\min}\leq\gamma\leq\gamma_{\max}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\max}^2}{(\alpha+\gamma)^2+\eta_{\max}^2} \right\} \leq 1, \quad \text{for} \quad \frac{\beta-\alpha}{2} \leq \gamma_{\min} \tag{18}$$

Moreover, if $\beta > \alpha$, then $\max_{\sigma_j\in\ \sigma(S)} \frac{\sqrt{\alpha^2+\sigma_j^2}}{\sqrt{\beta^2+\sigma_j^2}} < 1$, and therefore

(i) if $\alpha < \beta \leq \alpha + 2\gamma_{\min}$ then $\delta(\alpha,\beta) < 1$.

If $\beta \leq \alpha$, then $\max_{\sigma_j\in\ \sigma(S)} \frac{\sqrt{\alpha^2+\sigma_j^2}}{\sqrt{\beta^2+\sigma_j^2}} \leq \frac{\alpha}{\beta}$. By using (18), we have

$$\delta(\alpha,\beta) \leq \frac{\alpha}{\beta} \max_{\gamma_{\min}\leq\gamma\leq\gamma_{\max}} \sqrt{\frac{(\beta-\gamma)^2+\eta_{\max}^2}{(\alpha+\gamma)^2+\eta_{\max}^2}}.$$

So, in order to have the bound $\delta(\alpha,\beta) < 1$, the following inequality must hold

$$\max_{\gamma_{\min}\leq\gamma\leq\gamma_{\max}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\max}^2}{(\alpha+\gamma)^2+\eta_{\max}^2} \right\} < \frac{\beta^2}{\alpha^2}. \tag{19}$$

By using the results of Lemma 2.2, the following inequalities must hold

$$\frac{(\beta-\gamma_{\min})^2+\eta_{\max}^2}{(\alpha+\gamma_{\min})^2+\eta_{\max}^2} < \frac{\beta^2}{\alpha^2} \quad \text{and} \quad \frac{(\beta-\gamma_{\max})^2+\eta_{\max}^2}{(\alpha+\gamma_{\max})^2+\eta_{\max}^2} < \frac{\beta^2}{\alpha^2} \tag{20}$$

By simple computation, we can show that, for $\alpha+\beta > 0$, these two inequalities hold if $\beta$ satisfies the following inequalities.

$$\frac{\alpha(\gamma_{\min}^2+\eta_{\max}^2)}{2\alpha\gamma_{\min}+\gamma_{\min}^2+\eta_{\max}^2} < \beta \quad \text{and} \quad \frac{\alpha(\gamma_{\max}^2+\eta_{\max}^2)}{2\alpha\gamma_{\max}+\gamma_{\max}^2+\eta_{\max}^2} < \beta.$$

Therefore

(ii) if $\max\left\{ \frac{\alpha(\gamma_{\min}^2+\eta_{\max}^2)}{2\alpha\gamma_{\min}+\gamma_{\min}^2+\eta_{\max}^2}, \frac{\alpha(\gamma_{\max}^2+\eta_{\max}^2)}{2\alpha\gamma_{\max}+\gamma_{\max}^2+\eta_{\max}^2} \right\} < \beta \leq \alpha$ then $\delta(\alpha,\beta) < 1$.

Combining (i) and (ii), we have

(iii) if $\max\left\{ \frac{\alpha(\gamma_{\min}^2+\eta_{\max}^2)}{2\alpha\gamma_{\min}+\gamma_{\min}^2+\eta_{\max}^2}, \frac{\alpha(\gamma_{\max}^2+\eta_{\max}^2)}{2\alpha\gamma_{\max}+\gamma_{\max}^2+\eta_{\max}^2} \right\} < \beta \leq \alpha + 2\gamma_{\min}$ then $\delta(\alpha,\beta) < 1$

To prove parts (b) and (c), we note that, if $\gamma_{\min} \leq \dfrac{\beta-\alpha}{2}$, by Lemmas 2.1 and 2.3, we have

$$f(\alpha,\beta) = \max_{\gamma_{\min}\leq\gamma\leq\frac{\beta-\alpha}{2}} \left\{ \frac{(\beta-\gamma)^2+\eta_{\min}^2}{(\alpha+\gamma)^2+\eta_{\min}^2} \right\} = \frac{(\beta-\gamma_{\min})^2+\eta_{\min}^2}{(\alpha+\gamma_{\min})^2+\eta_{\min}^2} \geq 1,$$

since $(\alpha+\gamma_{\min}) \leq (\beta-\gamma_{\min})$.

On the other hand,

$$\max_{\sigma_j\in\ \sigma(S)} \frac{\sqrt{\alpha^2+\sigma_j^2}}{\sqrt{\beta^2+\sigma_j^2}} = \frac{\sqrt{\alpha^2+\sigma_{\max}^2}}{\sqrt{\beta^2+\sigma_{\max}^2}} < 1,$$

since $\alpha < \beta$. So, the relation

$$\delta(\alpha,\beta) = \frac{\sqrt{\alpha^2+\sigma_{\max}^2}}{\sqrt{\beta^2+\sigma_{\max}^2}} \sqrt{\frac{(\beta-\gamma_{\min})^2+\eta_{\max}^2}{(\alpha+\gamma_{\min})^2+\eta_{\max}^2}} < 1$$

will hold if $\alpha$ and $\beta$ satisfy the following inequality,

$$\frac{(\beta-\gamma_{\min})^2+\eta_{\max}^2}{(\alpha+\gamma_{\min})^2+\eta_{\max}^2} < \frac{\beta^2+\sigma_{\max}^2}{\alpha^2+\sigma_{\max}^2}. \tag{21}$$

For $\alpha+\beta>0$, this inequality is equivalent to

$$0 < (\beta-\alpha)(\gamma_{\min}^2+\eta_{\min}^2-\sigma_{\max}^2+2\alpha\gamma_{\min}) + 2\gamma_{\min}(\alpha^2+\sigma_{\max}^2). \tag{22}$$

Since $(\beta-\alpha)>0$, (22) holds if $\sigma_{\max} \leq \sqrt{\gamma_{\min}^2+\eta_{\min}^2+2\alpha\gamma_{\min}}$. Thus

(iv) if $\sigma_{\max} \leq \sqrt{\gamma_{\min}^2+\eta_{\min}^2+2\alpha\gamma_{\min}}$ and $\alpha+2\gamma_{\min}\leq\beta$, then $\delta(\alpha,\beta)<1$.

If $\sigma_{\max} > \sqrt{\gamma_{\min}^2+\eta_{\min}^2+2\alpha\gamma_{\min}}$, then (22) holds if $\beta$ satisfies the following inequality

$$\beta < \frac{\alpha(\gamma_{\min}^2+\eta_{\min}^2-\sigma_{\max}^2+2\alpha\gamma_{\min}) - 2\gamma_{\min}(\alpha^2+\sigma_{\max}^2)}{\gamma_{\min}^2+\eta_{\min}^2-\sigma_{\max}^2+2\alpha\gamma_{\min}}.$$

Thus

(v) if $\sigma_{\max} > \sqrt{\gamma_{\min}^2+\eta_{\min}^2+2\alpha\gamma_{\min}}$ and

$$\alpha+2\gamma_{\min} \leq \beta \leq \frac{\alpha(\gamma_{\min}^2+\eta_{\min}^2-\sigma_{\max}^2+2\alpha\gamma_{\min}) - 2\gamma_{\min}(\alpha^2+\sigma_{\max}^2)}{\gamma_{\min}^2+\eta_{\min}^2-\sigma_{\max}^2+2\alpha\gamma_{\min}}$$

then $\delta(\alpha,\beta)\leq 1$.

□

Theorem 2.1 mainly discusses the available $\beta$ for a convergent ANSS iteration for any given nonnegative $\alpha$. It also shows that the choice of $\beta$ is dependent on the choice of $\alpha$, the spectrum of the matrix $N$, the singular-values of $S$, but is not dependent on the spectrum of $A$. Notice that

$$\alpha + 2\gamma_{\min} - \frac{\alpha(\gamma_{\min}^2 + \eta_{\max}^2)}{2\alpha\gamma_{\min} + \gamma_{\min}^2 + \eta_{\max}^2} = \frac{2\alpha\gamma_{\min}^2 + 2\gamma_{\min}^3 + 2\gamma_{\min}\eta_{\max}^2 + 2\alpha^2\gamma_{\min}}{2\alpha\gamma_{\min} + \gamma_{\min}^2 + \eta_{\max}^2} > 0$$

and

$$\alpha + 2\gamma_{\min} - \frac{\alpha(\gamma_{\max}^2 + \eta_{\max}^2)}{2\alpha\gamma_{\max} + \gamma_{\max}^2 + \eta_{\max}^2} = \frac{2\alpha\gamma_{\min}\gamma_{\max} + 2\gamma_{\min}\gamma_{\max}^2 + 2\gamma_{\min}\eta_{\max}^2 + 2\alpha^2\gamma_{\max}}{2\alpha\gamma_{\max} + \gamma_{\max}^2 + \eta_{\max}^2} > 0,$$

we remark that for any given nonnegative $\alpha$ the available $\beta$ always exists. The bound $\delta(\alpha, \beta)$ of the convergence rate depends on the spectrum of $N$ and $S$ and the choice of $\alpha$ and $\beta$. Moreover, $\delta(\alpha, \beta)$ is also an upper bound of the contraction factor of the ANSS iteration.

## 3 Numerical Example

In this section, we give a numerical example to illustrate the effectiveness of ANSS iteration.

We consider the differential equation

$$-u^{''} + qu^{'} = f,$$

on the inteval $[0, 1]$, with the constant coefficient $q$ and the homogeneous boundary condition. When the finite difference discretization, for example, the centered difference is applied to the above equation, we get the system of linear equations (1) with the coefficient matrix

$$A = \text{tridiag}(-1 - \frac{qh}{2}, 2, -1 + \frac{qh}{2})$$

where the equidistant step-size $h = \dfrac{1}{n+1}$ is used.

Let $H = \frac{1}{2}(A + A^*)$ and $S_0 = \frac{1}{2}(A - A^*)$ be Hermitian and skew-Hermitian parts of A, respectively. We consider a NS splitting

$$A = N + S$$

where

$$N = H + icI \quad \text{and} \quad S = S_0 - icI$$

and $c$ is a real number. We test the spectral radius of the iteration matrix $M(\alpha, \beta)$ (8) with different values of $qh$. All the tested matrices are $64 \times 64$.

In Figs. 1 and 2, we show the spectral radius of the iteration matrix of the ANSS method and the NSS method with different values of $\alpha$. ANSS represents the spectral radius of the iteration matrix of the ANSS method, where parameter $\beta$ is tested to be the optimal one, and NSS represents that of the NSS method.

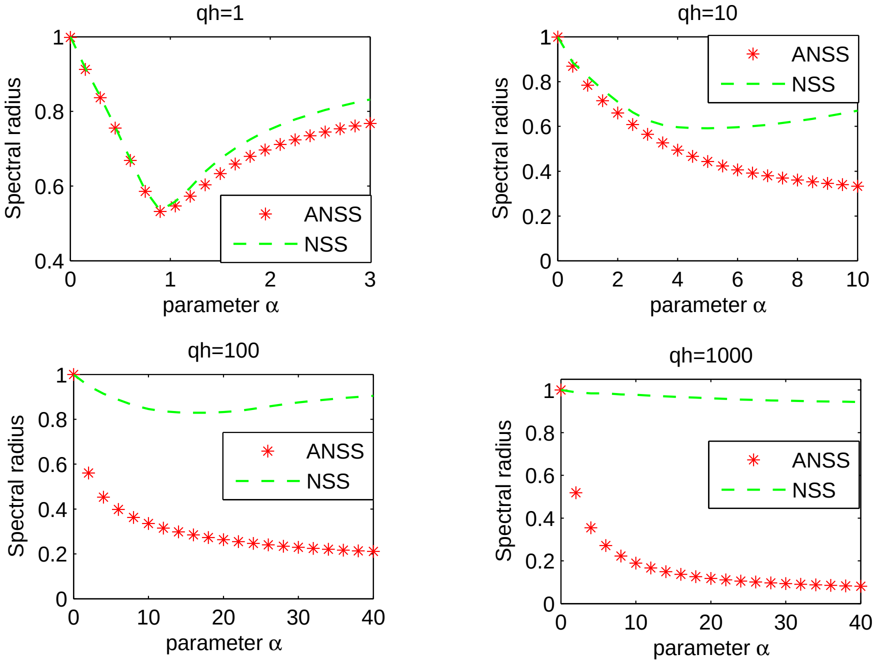


Fig. 1: Spectral radius of iteration matrices of ANSS and NSS methods for $c = .1$

We find that if $c = 0.1$ is used, the spectral radius of the iteration matrix of the ANSS method is always smaller than that of the NSS method, and when $qh$ is large, the spectral radius of the iteration matrix of the ANSS method is much smaller than that of the NSS method, but if $c = 10$ is used, these two spectral radius of the iteration matrices are almost the same.

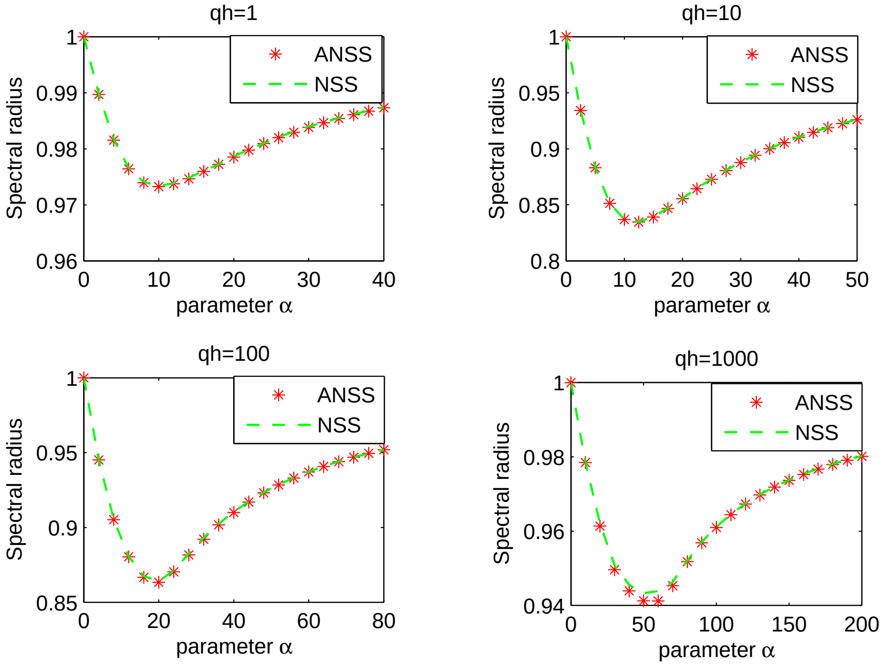


Fig. 2: Spectral radius of iteration matrices of ANSS and NSS methods for $c = 10$

## 4 Conclusion

In this paper, we have introduced two constants for the NSS iteration and presented a different approach to solve the system of linear equations (1), called ANSS method.

Theoretical analysis showed that if the coefficient matrix $A$ is positive definite (Hermitian or non-Hermitian) the ANSS method can converge to the unique solution of the linear system (1) with any given nonnegative $\alpha$, if $\beta$ is restricted to an appropriate region. In addition the upper bound of the contraction factor of the ANSS iteration is dependent on the choice of $\alpha$ and $\beta$, the spectrum of the normal matrix $N$ and the singular-values of the skew-Hermitian, but is not dependent on the eigenvectors of the matrices $N$, $S$ and $A$. Numerical examples illustrated the effectiveness of ANSS iteration and showed that the spectral radius of the iteration matrix of the ANSS method is always smaller than or equal to that of the NSS method.

## Acknowledgment

The authors wish to thank the referee for valuable comments and suggestions.

## References

1. Bai, Z-Z. and Golub, G.H. and Ng, M.K., *On successive-overrelaxation acceleration of the Hermitian and skew-Hermitian splitting iterations*, Numer. Linear. Algebra. Appl., **14** (2007) 319-335.
2. Bai, Z-Z., Golub, G.H. and Ng, M.K., *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems* , SIAM J. Matrix. Anal. Appl., **24** (2003) 603-626.
3. Huhtanen, M., *Aspects of nonnormality for iterative methods*, Linear Algebra Appl., **394** (2005) 119-144.
4. Huhtanen, M.,*A Hermitain Lanczos method for normal matrices.*, SIAM J. Matrix. Anal. Appl., **23** (2002) 1092-1108.
5. Li, L., Huang, T-Z. and Liu, X. P., *Asymmetric Hermitian and skew-Hermitian splitting methods for positive definite linear systems*, Comput. Math. Appl., **54** (2007) 147-159

Iranaian J. Numerical Analysis and Optimization
Vol. 3, No. 1, (2013), 45-54

# An alternative 2-phase method for evaluating of DMUs using DEA

Mohammadreza Alirezaee

**Abstract**

Computationally, selection of a proper numerical value for infinitesimal non Archimedean epsilon in DEA models has some difficulties. Although there are several algorithms for selecting the proper non-Archimedean epsilon, it is important to introduce methods in order to calculate the efficiency of DMUs without using epsilon. One of these methods is a two-phase method, which obtains the efficiency of each DMU through solving two LPs, which the second LP is depended to the first. This paper proposes a method, which is able to compute the efficiency of DMUs by two LPs, which are not depended to each other and computationally can solve in a parallel computation. The major of this method is to find two references for each unit and combine them to obtain actual reference.

**Keywords:** Data Envelopment Analysis (DEA); Decision Making Units (DMUs); Non-Archimedean; Two-phase method; Reference point.

## 1 Introduction

Since the first mathematical model of Data Envelopment Analysis (DEA) by Charnes et al. (1978), (known as CCR), and Banker et al. (1984) (known as BCC), there have been many theoretical and applied researches in DEA (Emrouznejad ea al., 2008). In 1979 the first version of DEA model has been updated by adding the non-Archimedean $\varepsilon$ as a lower bound for weights of inputs and outputs of the corresponding DMUs(Charnes et al. 1979). Different methods have been proposed for computing a suitable value for $\varepsilon$. Ali and Seiford (1993) introduce a method to find an acceptable value for $\varepsilon$. Mehrabian et al. (2000) modify this method and propose an LP to select a proper value for $\varepsilon$. Up to now, some researchers have published methods and discussions about the non-Archimedean $\varepsilon$ such as Amin and Toloo (2004), MirHassani and Alirezaee (2004), Alirezaee and Khalili (2006).

As the first and most important substitution method for epsilon-based DEA solving methods, Cooper et al. (1999) introduce the two-phase method

Mohammadreza Alirezaee
School of Mathematics, Iran University of Sciences and Technology, Tehran, Iran. e-mail: mralirez@iust.ac.ir

for evaluating the efficiency in DEA without using $\varepsilon$. In this method , two LPs must be solved respectively for each DMU.

In this paper, we introduce an alternative algorithm for evaluating the efficiency of DMUs without using $\varepsilon$. In the proposed method, firstly, we find the references of DMUs and then inherited the references in a way that we can find out if the reference point of the unit is located on the weak frontier. The most advantage of this method is reducing the overall running time, because we can use parallel computation for our independent LPs in the algorithm.

## 2 Classification of DMUs

In DEA a set of DMUs are partitioned into two main classes: efficient and inefficient. The efficient units make the efficiency frontier. Figure 1 shows the classification of DMUs, based on the position of their reference on the efficiency frontier (Charnes et al., 1991).

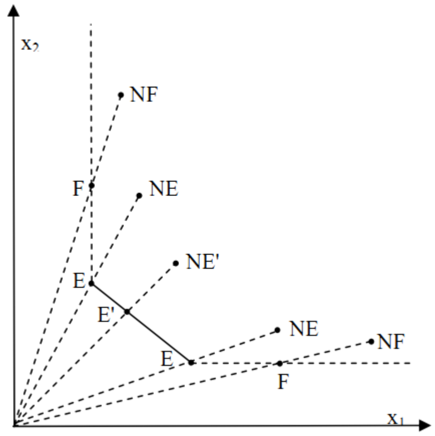


Fig. 1: Classifying DMUs.

In this classification, $E$ and $E^{'}$ are efficient and $NE$ and $NE^{'}$ are inefficient DMUs. In addition, $F$ and $NF$ are weakly efficient and weakly inefficient, respectively.

Suppose that there are $n$ Decision Making Units (DMUs) each consumes m inputs to produce $s$ outputs. Let $x_j = (x_{1j}, x_{2j}, \ldots, x_{mj})$ and $y_j = (y_{1j}, y_{2j}, \ldots, y_{sj})$ are input vector and output vector of DMUj $(j = 1, \ldots, n)$, respectively. Hence, the CCR model corresponding to DMUp is as follow:

**CCR Model:**

$$\min z = \theta - \varepsilon(\Sigma_{i=1}^{m}(s_i^-) + \Sigma_{r=1}^{s}(s_r^+))$$

$$s.t.$$

$$x_{ip}\theta - s_i^- - \Sigma_{j=1}^{n} x_{ij}\lambda_j = 0, \forall i$$

$$-s_r^+ + \Sigma_{j=1}^{n} y_{rj}\lambda_j = y_{rp}, \forall r$$

$$\lambda_j, s_i^-, s_r^+ \geq 0, \forall i, r, j$$

To introduce the new method, all of the values of variables for all DMUs are needed. So consider the following integrated model.

**CCR$_P$ Model:**

$$\min z = \theta_p - \varepsilon(\Sigma_{i=1}^{m} s_{ip}^- + \Sigma_{r=1}^{s} s_{rp}^+)$$

$$s.t.$$

$$x_{ip}\theta_p - s_{ip}^- - \Sigma_{j=1}^{n} x_{ij}\lambda_{jp} = 0, \quad \forall i$$

$$-s_{rp}^+ + \Sigma_{j=1}^{n} y_{rj}\lambda_{jp} = y_{rp}, \quad \forall r$$

$$\lambda_{jp}, s_{ip}^-, s_{rp}^+ \geq 0, \quad \forall i, r, j$$

In the above model, for each variable an index has been added. To simplify the notation, let $u_j = (y_j, x_j)$ be DMUj. Hence one can present members of productivity possibility set as $u = (y, -x)$. Clearly, $u_j \in PPS$,$(j = 1, \ldots, n)$. Let $(\theta_p^*, \lambda_p^*, {s_p^+}^*, {s_p^-}^*)$ is an optimal solution of $CCR_p$, then efficiency of $u_p$ is equal to $\theta_p^*$. We also denote the efficiency of $u$ by $\theta_u^*$. The dominate space and reference of $u_p$ are denoted by $DS_p$ and $u(p)$, respectively, and the set of reference indices of $u_p$ is denoted by E(p).

$$DS_p = \{u \in PPS | u \geq u_p\}, u(p) = \Sigma_{j=1}^{n} l_{jp}^* u_j, E(p) = \{j : \lambda_{jp}^* > 0\}$$

It is clear that $u(p) \in DS_p$ and $u(p) = \Sigma_{j \in E(p)} \lambda_{jp}^* u_j$. These concepts are illustrated in Figure 2.

The shading pattern in this figure and other figure in the paper represents the dominant space of $u_p$, so every point in that region has inputs less than or equal and outputs more than or equal to $u_p$. And dashed line in the frontier is weakly frontier.

Therefore, we have:

1. $u_p$ is efficient $\Leftrightarrow \theta_p^* = 1, \Sigma_{i=1}^{m} {s_{ip}^-}^* + \Sigma_{r=1}^{s} {s_{rp}^+}^* = 0$.
2. $u_p$ is weak efficient $\Leftrightarrow \theta_p^* = 1, \Sigma_{i=1}^{m} {s_{ip}^-}^* + \Sigma_{r=1}^{s} {s_{rp}^+}^* > 0$.
3. $u_p$ is inefficient $\Leftrightarrow \theta_p^* < 1, \Sigma_{i=1}^{m} {s_{iu(p)}^-}^* + \Sigma_{r=1}^{s} {s_{ru(p)}^+}^* = 0$.
4. $u_p$ is weak inefficient $\Leftrightarrow \theta_p^* < 1, \Sigma_{i=1}^{m} {s_{iu(p)}^-}^* + \Sigma_{r=1}^{s} {s_{ru(p)}^+}^* > 0$.

Notice that $u(p)$ lies on the efficiency frontier, and so $\theta_{u(p)}^* = 1$.

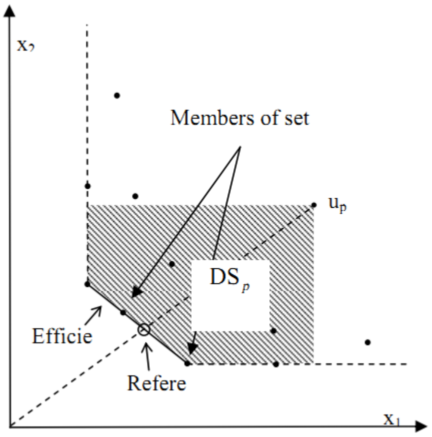


Fig. 2: Efficient frontier, dominate space, reference point and set of reference index.

## 3 New Method

As explained in previous section, for classifying the DMUs, the values of $\theta$ and slacks of reference point must be used. For each $u_p$, two reference points are determined, the first reference belongs to $DS_p$ which minimize $\theta_p$. The second reference belongs to $DS_p$ which maximize $\Sigma_{i=1}^{m} s_{ip}^{-^*} + \Sigma_{r=1}^{s} s_{rp}^{+^*}$. Consider the following models:

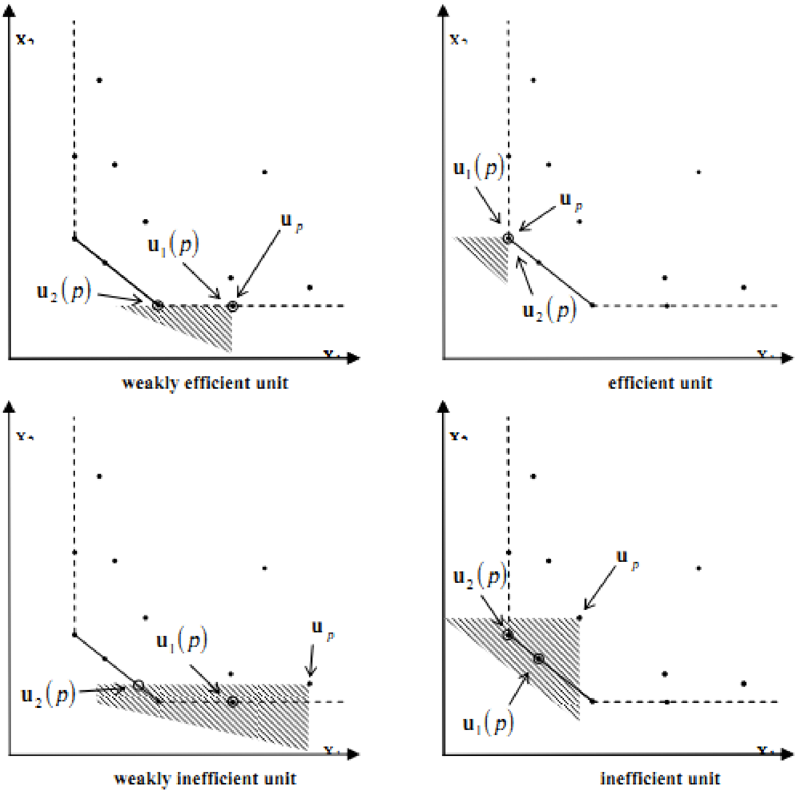
Model P1:

$$\begin{aligned} &\min z = \theta_p s.t. \\ &\quad x_{ip}\theta_p - \Sigma_{j=1}^{n} x_{ij}\lambda_{jp}^{l} \geq 0, \quad \forall i \\ &\quad \Sigma_{j=1}^{n} y_{rj}\lambda_{jp}^{l} \geq y_{rp}, \quad \forall r \\ &\quad \lambda_{jp}^{l} \geq 0, \quad \forall j \end{aligned}$$

Model P2:

$$\begin{aligned} &\min z = -(\Sigma_{i=1}^{m} s_{ip}^{-} + \Sigma_{r=1}^{s} s_{rp}^{+}) \\ &s.t. \\ &\quad s_{ip}^{-} + \Sigma_{j=1}^{n} x_{ij}\lambda_{jp}^{2} = x_{ip}, \quad \forall i \\ &\quad - s_{rp}^{+} + \Sigma_{j=1}^{n} y_{ij}\lambda_{jp}^{2} = y_{rp}, \quad \forall i \\ &\quad \lambda_{jp}^{2}, s_{ip}^{-}, s_{rp}^{+} \geq 0, \quad \forall j, i, r \end{aligned}$$

The first reference for $u_p$ is determined by Model P1 as $u_1(P) = \Sigma_{j=1}^{n}\lambda_{jp}^{l^*}u_j$. Similarly, the second reference for $u_p$ is determined by the model p2 as $u_2(p) = \Sigma_{j=1}^{n}\lambda_{jp}^{2^*}u_j$. These references are illustrated in Figure 3:



weakly efficient unit

efficient unit

weakly inefficient unit

inefficient unit

Fig. 3: $u_p$ and its first and second references.

$u_1(p)$ is not always the same as the reference point which is defined as $(\theta^* x_j, y_j)$ for $(x_j, y_j)$. If $u_p$ be an efficient or inefficient unit then its reference point and $u_1(p)$ are the same and if it is a weakly efficient or inefficient unit then probably its reference probably its reference point and $u_1(p)$ are not equal to each other. For example, in the Figure 4(a), $u_1(p)$ and reference point of $u_p$ are the same, but if the weakly efficient unit $u_1(p)$ is removed as illustrated in the Figure 4(b), the two definitions become different. In this case the reference point can be improved to the $u_1(p)$, therefore we must improve the reference point, too. Reference of each unit must be on the (strong) frontier and not on the weakly frontier. Weakly frontier is shown by dashed line in the figures in the paper.

**Definition 3.1.** The revised reference for $u_p$ is defined as bellow:

$$\hat{u}(p) = \mathit{\Sigma}_{k=1}^{n}(\mathit{\Sigma}_{j=1}^{n}\lambda_{jp}^{1^*}\lambda_{kj}^{2^*})$$

If $u_p$ be an efficient or an inefficient unit then the revised reference is $u_1(p)$, but if it is a weekly efficient or inefficient unit then the reference for $u_p$ on the frontier is moved by $\hat{u}(p)$ using the weights belonging to E(P). Revised reference is simply the second reference of the first reference of the *pth* unit. We introduce a simple algorithm to identify the revised reference of $u_p$ and describe the idea of revised reference. It woks as follows:

1. Find the first reference of $u_p$. There are one or more units that create the first reference of $u_p$. They are efficient or weakly efficient units.
2. For each unit that participating the construction of the first reference of $u_p$, find the second reference. For each of them, there are one or more units that are efficient (and not weakly efficient).
3. The second reference of the first reference of $u_p$ is the actual reference of it, so calculate values of the new variable that create the second reference of the first reference of $u_p$. The revised reference is on the efficient frontier and is the actual reference of $u_p$.

   This algorithm computed the revised reference of $u_p$, in other word we move of $u_p$ to $u_1(p)$ and then move to $u_2(u_1(p))$, as illustrated in Figure 4.

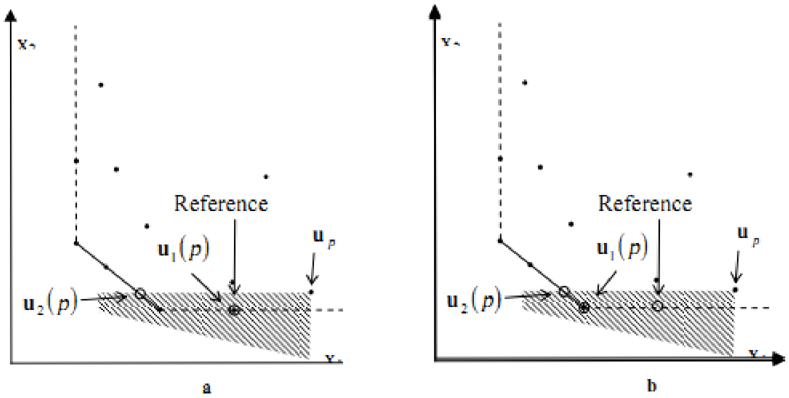


Fig. 4: Reference point and $u_1(p)$ may be different.

These concepts lead to the next theorems 1 proves that only efficient (not weakly efficient) units participating in construction of revised reference. And theorem 2 proves that comparing $u_p$ to the revised reference results the actual efficiency.

**Theorem 3.1.** *If* $u_k$ *is not an efficient unit then* $\sum_{j=1}^{n}\lambda_{jp}^{1^*}\lambda_{kj}^{2^*} = 0$.

**Proof**. Since $\lambda_{kj}^{2^*}$ is the optimum weight of $u_k$ in the model P2 for evaluating $u_j$, if $u_k$ is a weakly efficient, an inefficient or a weakly inefficient unit then for all j, $\lambda_{kj}^{2^*} = 0$ and $\sum_{j=1}^{n} \lambda_{jp}^{1^*} \lambda_{kj}^{2^*} = 0$. □

**Theorem 3.2.** *If we select $\hat{u}(p)$ as a reference for $u_p$ in the model P1, then efficiency value of $u_p$ is equal to $\theta_p^*$.*

**Proof**. We knew that $u_1(p)$ is on the efficient frontier and $\theta_p^*$ is the minimum value of $\theta_p$, so we must prove that $\hat{u}(p) \geq u_1(p)$. We rewrite the $\hat{u}(p)$ as follow:

$\hat{u}(p) = \sum_{j=1}^{n} \lambda_{jp}^{1^*} \Big( \sum_{j=1}^{n} \lambda_{kj}^{2^*} u_k \Big) = \sum_{j=1}^{n} \lambda_{jp}^{1^*} u_2(j)$, where $u_2(j)$ is the reference of $u_j$ in the model P2.

There are three possibilities for $s_j$:

1. If $u_j$ is an inefficient or a weakly inefficient unit, then $\lambda_{jp}^{1^*} = 0$.
2. If $u_j$ is an inefficient unit, then $\lambda_{jp}^{1^*} > 0$ and $u_2(j) = u_j$.
3. If $u_j$ is an weakly unit, then $\lambda_{jp}^{1^*} > 0$ and $u_2(j) \geq u_j$.

In all cases we have $\hat{u}(p) = \sum_{j=1}^{n} \lambda_{jp}^{1^*} u_2(j) \geq \sum_{j=1}^{n} \lambda_{jp}^{1^*} u_j = u(p)$.

Since $\theta_p^*$ is the minimum value of $\theta_p$, therefore, if we select $\hat{u}(p)$ as the reference of $u_p$ in model P1, the efficiency value of $u_p$ is equal to $\theta_p^*$. □

This shows that $\hat{u}(p)$ is on the efficient frontier, and $\sum_{j=1}^{n} \lambda_{jp}^{1^*} \lambda_{kj}^{2*} > 0$ if and only if $u_i$ is an efficient unit.

Based on Theorems 1 and 2, $\hat{u}(p)$ is a combination of (only) efficient units and its corresponding efficiency value is the same as in model P1. After applying models P1 and P2 for all units, it is possible to compute the revised references of all units, which are on the efficient (and not on the weakly efficient) frontier. It sounds that this method is similar two phase method, but in the new method, the two models are independent.

Based on theorems 1 and 2, $\hat{u}(p)$ is on the efficient frontier and as a reference for $u_p$, the efficiency value is not changed. $\hat{\lambda}_{kp}^* = \sum_{j=1}^{n} \lambda_{jp}^{1^*} \lambda_{kj}^{2*}$, where the related slacks are computed as follows:

$$\hat{s}_{ip}^{-*} = \theta_p^* x_{ip} - \sum_{j=1}^{n} \hat{\lambda}_{jp}^* x_{ij}, \ \hat{s}_{rp}^{+*} = \sum_{j=1}^{n} \hat{\lambda}_{jp}^* y_{rj} - y_{rp}.$$

Therefore, after applying P1 and P2 models for all units, we can compute the revised references, which are on the efficiency frontier, and the related slacks. According to classifying the DMUs, we have:

1. If $\theta_p^* = 1$ and $\sum_{i=1}^{m} \hat{s}_{ip}^{-*} + \sum_{r=1}^{s} \hat{s}_{rp}^{+*} = 0$, then $u_p$ is efficient,
2. If $\theta_p^* = 1$ and $\sum_{i=1}^{m} \hat{s}_{ip}^{-*} + \sum_{r=1}^{s} \hat{s}_{rp}^{+*} > 0$, then $u_p$ is weakly efficient,
3. If $\theta_p^* < 1$ and $\sum_{i=1}^{m} \hat{s}_{ip}^{-*} + \sum_{r=1}^{s} \hat{s}_{rp}^{+*} = 0$, then $u_p$ is inefficient,
4. If $\theta_p^* < 1$ and $\sum_{i=1}^{m} \hat{s}_{ip}^{-*} + \sum_{r=1}^{s} \hat{s}_{rp}^{+*} > 0$, then $u_p$ is weakly inefficient.

The reference for $u_p$ is $\hat{u}(p) = \sum_{j=1}^{n} \hat{\lambda}_{jk}^{*} u_j$.

The basic role of this method is removing the non-Archimedean epsilon of models and using models that have no dependency to each others and could be solved separately while the models of two-phase method needs to be solved respectively, because second model uses of result of first model.

The basis of traditional two-phase method is to find the optimum value of $\theta$ in the first stage and fix it to the second LP and solve it to find the maximum value for sum of slacks.

## 4 Numerical Example

In this section we solve a simple numerical example. We add the constraints $\sum_{j=1}^{n} \lambda_{jp}^{1} = 1$ and $\sum_{j=1}^{n} \lambda_{jp}^{2} = 1$ to models P1 and P2 respectively and create the BCC versions of DEA models.

Consider the following example:

Table 1: Data for the numerical example

| | *Output* | *Input* |
|---|---|---|
| | $y$ | $x$ |
| **DMU1** | 1 | 1 |
| **DMU2** | 2 | 1 |
| **DMU3** | 0.2 | 2 |
| **DMU4** | 3 | 3 |
| **DMU5** | 4 | 3 |

These data are illustrated in Figure 5.

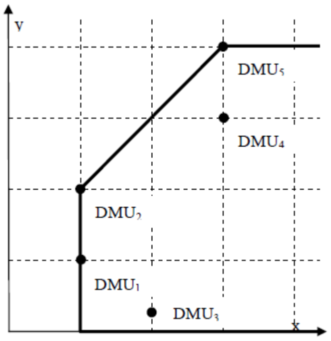


Fig. 5: DMUs of the example.

Table 2 shows the optimum solutions of models P1 and P2 in the BCC format:

Table 2: Results of the example using model P1 and P2 with variable returns to scale

| **DMU** | $\theta^*$ | $\lambda^{1^*}$ | | | | | $\lambda^{2^*}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 |
| **DMU1** | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **DMU2** | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **DMU3** | 0.50 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **DMU4** | 0.6667 | 0 | 0.5 | 0 | 0 | 0.5 | 0 | 0 | 0 | 0 | 1 |
| **DMU5** | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |

For example for DMU4, we have $\lambda^{2^*}_{24} = 0.5$, $\lambda^{2^*}_{54} = 0.5$, and $\lambda^{2^*}_{54} = 1$. The following table presents $\hat{s}^{-^*}$, $\hat{s}^{+^*}$ and $\hat{\lambda}^*$:

Table 3: $\hat{s}^{-^*}$, $\hat{s}^{+^*}$ and $\hat{\lambda}^*$

| **DMU** | $\theta^*$ | $\hat{s}^{-^*}$ | $\hat{s}^{+^*}$ | $\hat{\lambda}^*$ | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 |
| 1 | 1 | 0.0 | 1.0 | 0 | 1 | 0 | 0 | 0 |
| 2 | 1 | 0.0 | 0.0 | 0 | 1 | 0 | 0 | 0 |
| 3 | 0.50 | 0.0 | 1.80 | 0 | 1 | 0 | 0 | 0 |
| 4 | 0.6667 | 0.0 | 0.0 | 0 | 0.5 | 0 | 0 | 0.5 |
| 5 | 1 | 0.0 | 0.0 | 0 | 0 | 0 | 0 | 1 |

Thus, the revised references of DMUs are as follow:

Table 4: The revised references of DMUs

| **DMU** | $u_j$ | $u(j)$ | $E(j)$ |
|---|---|---|---|
| 1 | (1, -1) | $1 \times u_2$ | $\{2\}$ |
| 2 | (2, -1) | $1 \times u_2$ | $\{2\}$ |
| 3 | (0.2, -2) | $1 \times u_2$ | $\{2\}$ |
| 4 | (3, -3) | $0.5 \times u_2 + 0.5 \times u_5$ | $\{2, 5\}$ |
| 5 | (4, -3) | $1 \times u_5$ | $\{5\}$ |

It is concluded that we can compute the results of the efficiency evaluation of DMUs by applying a linear programming for each unit.

## 5 Conclusion

There are two major methods for solving the basic DEA models: the epsilon based method, which selects a real number for epsilon, and the two phases method, which is used two LPs for each DMU. In this paper, we presented another method that determines two references for each DMU and then combines them and computes new lambdas and slack variables. Solving models without using non-Archimedean epsilon is an advantage of the new method and ability of computing the models in parallel can reduce the overall running time.

## References

1. Ali, A.I. and Seiford, L. M., *Computational Accuracy and Infinitesimals in Data Envelopment Analysis.* INFOR, *37*, (1993), 290-297.
2. Alirezaee, M. R. and Khalili, M., *Recognizing the efficiency, weak efficiency and inefficiency of DMUs with an epsilon independent linear program.* Applied Mathematics and Computation, **183**, (2006), 1323-1327.
3. Amin, G. R. and Toloo, M., *A polynomial-time algorithm for finding Epsilon in DEA models.* Computers Operations Research, **31**, (2004), 803-805.
4. Banker, R. D., Charnes, A., and Cooper, W.W., *Some models for estimating technical and scale inefficiencies in data envelopment analysis*, Management Science, **30**, (1984), 1078-1092.
5. Charnes, A., Cooper, W. and Rhodes, E., *Measuring the Efficiency of Decision Making Units.* European Journal of Operational Research, **2**, (1978), 429-444.
6. Charnes, A., Cooper, W. and Thrall, R. M., *A Structure for Classifying and Characterizing Efficiency and Inefficiency in Data Envelopment Analysis.* The Journal of Productivity Analysis, **2**, (1991), 197-237.
7. Charnes, A. and Rhodes, E., *Short Communication Measuring the Efficiency of DMU's.* EJOR (1979), 339-339.
8. Cooper, W. W., Seiford, L .M. and Tone, K., *Data EnvelopmentAnalysis A Comprehensive Text with Models.*, Applications, References and DEA-Solver Software, Springer Science, (1999).
9. Emrouznejad, A., Parer, B. R. and Tavarese, G., *Evaluation of research in efficiency and productivity: A survery and analysis of the first 30 years of scholarly literature in DEA*. Socio-Economic Planning Sciences, **42**, (2008), 151-157.
10. Mehrabian, S., Jahanshahloo, G. R., Alirezaee, M. R. and Amin, G. R., *An Assurance Interval for the Non-Archimedean Epsilon in DEA Models.* Operations Research, **48**, (2000), 344-347.
11. MirHassani, S. A. and Alirezaee, M. R., *An efficient approach for computing non-Archimedean (Epsilon) in DEA based on integrated models.* Applied Mathematics and Computation, **166**, (2004), 449-456 .

**Iranaian J. Numerical Analysis and Optimization**
**Vol. 3, No. 1, (2013), 55-72**

# Finite volume method for one dimensional biot poroelasticity system in multilayered domains

M. Namjoo and H. Atighi Lorestani

**Abstract**

R. Ewing, O. Liev, R. Lazarov and A. Naumovich in [1] proposed a finite volume discretization for one dimensional Biot poroelasticity system in multilayer domains. Their discretization and exact solution are invalid. We derive valid discretization and exact solution. Finally, our numerical solution is compared with known exact solution in discrete $L_2$ norm.

**Keywords:** Biot poroelasticity system; Interface problem; Finite volume discretization.

## 1 Introduction

The presence of a moving fluid in a porous medium affects its mechanical response. At the same time, the change in the mechanical state of the porous skeleton influences the behavior of the fluid inside the pores. These two coupled deformation-diffusion phenomena lie at the heart of the theory of poroelasticity. More precisely, the two key phenomena can be summarized as follows:

1. fluid-to-solid coupling: occurs when a change in the fluid pressure or the fluid mass induces a deformation of the porous skeleton.
2. solid-to-fluid coupling: occurs when modifications in the stress of the porous skeleton induce change in the fluid pressure or the fluid mass.

In accordance with these two phenomena, the fluid-filled porous medium acts in a time-dependent manner. Indeed, suppose that the porous medium is compressed. This will result in an increment of the fluid pressure inside the pores and consequent fluid flow. The time dependence of the fluid pressure will induce a time dependence of the poroelastic stresses, which in turn will respond

M. Namjoo
Department of Mathematics, Vali-e-Asr University of Rafsanjan, Rafsanjan, Iran. e-mail: namjoo@vru.ac.ir

H. Atighi Lorestani
Department of Mathematics, Vali-e-Asr University of Rafsanjan, Rafsanjan, Iran. e-mail: hatighi@gmail.com

back to the fluid pressure field. The earliest theories, which is related to Terzaghi, accounted for the fluid-to-solid coupling only. In this case, the problem is mathematically much easier. This kind of theory can model successfully some of the poroelastic processes in the case of highly compressible fluids such as air. However, when one deals with slightly compressible or incompressible fluids, the solid-to-fluid coupling cannot be neglected since the changes in stress field can influence significantly the pore pressure. Maurice Biot was the first who, by means of phenomenological approach, developed a detailed mathematical theory of poroelasticity which successfully incorporated both basic phenomena mentioned above. In this paper, assumption of only vertical subsidence is invoked and this leads to the one dimensional model of poroelasticity. We consider a finite volume discretization for one dimensional Biot poroelasticity system in multilayer domains. For stability reasons, staggered grids are used. The discretization takes into account discontinuity of the coefficients across the interfaces between layers with different physical properties.

## 2 Biot model in one dimension

In one dimension, the domain of consideration $\Omega$ is an interval $(0, L)$ where the boundary $\Gamma$ is $\{0, L\}$. The Biot model, which describes poroelastic process in $\Omega$ can be written as a system of partial differential equations for the unknown fluid pressure $p(x, t)$ and displacement of the porous medium $u(x, t)$ consisting of the equilibrium equation and the diffusion equation

$$\begin{cases} -\dfrac{\partial}{\partial x}((\lambda + 2\mu)\dfrac{\partial u}{\partial x}) + \dfrac{\partial p}{\partial x} = 0, \quad x \in (0, L), \quad t \in (0, T], \\ \dfrac{\partial}{\partial t}(\phi\beta p + \dfrac{\partial u}{\partial x}) - \dfrac{\partial}{\partial x}(\dfrac{\kappa}{\eta}\dfrac{\partial p}{\partial x}) = q(x, t), \quad x \in (0, L), \quad t \in (0, T], \end{cases}$$

where $\lambda$ (dilation moduli) and $\mu$ (shear moduli) are Lame coefficients of the porous medium. Here $\phi$, $\beta$, $\kappa$, $\eta$ and $q(x, t)$ are porosity of porous medium, compressibility of the fluid, permeability of the porous medium, viscosity of the fluid and source term, respectively. We define stress tensor and fluid velocity, respectively by the following relationships

$$S = (\lambda + 2\mu)\frac{\partial u}{\partial x}, \quad V = -\frac{\kappa}{\eta}\frac{\partial p}{\partial x}.$$

In classical formulation, the one-dimensional Biot model describes, fluid flow and skeleton deformation caused by the constant vertical load applied on the top of column of soil, which is bounded with rigid and impermeable bottom and lateral walls, and a top wall which is free to drain. The following boundary and initial conditions supplement this model

$$p = 0, \quad (\lambda + 2\mu)\frac{\partial u}{\partial x} = -s_0, \quad at \quad x = 0.$$

This means that the upper boundary is free to drain and a load with the value $s_0$ is applied to it. Also

$$u = 0, \quad \frac{\partial p}{\partial x} = 0, \quad at \quad x = L,$$

corresponds to a rigid and impermeable lower boundary. The initial condition

$$\phi\beta p + \frac{\partial u}{\partial x} = 0, \quad at \quad t = 0,$$

means that the variation in water content is zero at the beginning of the process. Now, consider the case when the porous medium is not homogeneous but has a layered structure, each layer being characterized by different porosity, permeability and Lame coefficients. For the simplicity of presentation, let us restrict ourselves to the case of only two layers. In the case of the considered two-layered medium, coefficients of the governing equations are discontinuous across the interface $\xi$

$$\lambda(x) = \begin{cases} \lambda_1 & x \leq \xi, \\ \lambda_2 & x > \xi, \end{cases} \qquad \mu(x) = \begin{cases} \mu_1 & x \leq \xi, \\ \mu_2 & x > \xi, \end{cases}$$

$$\kappa(x) = \begin{cases} \kappa_1 & x \leq \xi, \\ \kappa_2 & x > \xi, \end{cases} \qquad \phi = \begin{cases} \phi_1 & x \leq \xi, \\ \phi_2 & x > \xi. \end{cases}$$

Assuming a perfect contact, the interface conditions look as follows

$$[u] = 0, \quad [p] = 0, \tag{1}$$

which express continuity of the displacement and of the fluid pressure across the interface. Also

$$[S] = 0, \quad [V] = 0, \tag{2}$$

which means continuity of the stress of the porous skeleton and continuity of the fluid flux, respectively. In the formulae (1) and (2), we have

$$[q] = q \mid_{x=\xi+0} - q \mid_{x=\xi-0},$$

where $q$ is a symbol for quantities $u$, $p$, $S$ and $V$. As it is shown in [3], the set of interface conditions (1) and (2) can also be derived directly from the Biot equations if they are written for a general inhomogeneous medium. Now, the following dimensionless dependent and independent functions are introduced

$$x := \frac{x}{L}, \quad \xi := \frac{\xi}{L}, \quad t := \frac{(\lambda_0 + 2\mu_0)\kappa_0 t}{\eta_0 L^2}, \quad p := \frac{p}{s_0}, \quad u := \frac{(\lambda_0 + 2\mu_0)u}{s_0 L},$$

$$\nu := \frac{\lambda + 2\mu}{\lambda_0 + 2\mu_0}, \quad \kappa := \frac{\frac{\kappa}{\eta}}{\frac{\kappa_0}{\eta_0}}, \quad a := \phi\beta(\lambda_0 + 2\mu_0), \quad f(x,t) := \frac{L^2\eta_0}{s_0\kappa_0} q(x,t).$$

Then, the governing equations together with the boundary, initial and interface conditions can be transformed to dimensionless form

$$\begin{aligned}
&-\frac{\partial}{\partial x}(\nu\frac{\partial u}{\partial x}) + \frac{\partial p}{\partial x} = 0, \quad x \in (0,1), \ \ t \in (0,T],\\
&\frac{\partial}{\partial t}(ap + \frac{\partial u}{\partial x}) - \frac{\partial}{\partial x}(\kappa\frac{\partial p}{\partial x}) = f(x,t), \quad x \in (0,1), \ \ t \in (0,T],\\
&\nu\frac{\partial u}{\partial x} = -1, \quad p = 0, \quad at \quad x = 0, \ \ t \in [0,T], \qquad (3)\\
&u = 0, \quad \kappa\frac{\partial p}{\partial x} = 0, \quad at \quad x = 1, \ \ t \in [0,T],\\
&ap + \frac{\partial u}{\partial x} = 0, \quad at \quad t = 0, \ \ x \in (0,1),\\
&[u] = 0, \quad [\nu\frac{\partial u}{\partial x}] = 0, \quad [p] = 0, \quad [\kappa\frac{\partial p}{\partial x}] = 0, \quad at \quad x = \xi, \ \ t \in [0,T].
\end{aligned}$$

Further, the possible discontinuities of the dimensionless coefficients at $x = \xi$ are distinguished

$$\nu(x) = \begin{cases} \nu_1 & x \le \xi, \\ \nu_2 & x > \xi, \end{cases} \qquad \kappa(x) = \begin{cases} \kappa_1 & x \le \xi, \\ \kappa_2 & x > \xi, \end{cases}$$

$$a(x) = \begin{cases} a_1 & x \le \xi, \\ a_2 & x > \xi. \end{cases}$$

For the convenience of the theoretical analysis, the problem (3) is transformed into a problem with homogeneous boundary conditions, by the following substitution

$$u(x,t) := u(x,t) - \frac{1}{\nu}x + \frac{1}{\nu}.$$

According to this substitution, problem (3) is reformulated as follows:

$$
\begin{aligned}
&-\frac{\partial}{\partial x}(\nu\frac{\partial u}{\partial x})+\frac{\partial p}{\partial x}=0, \quad x\in(0,1),\ \ t\in(0,T],\\
&\frac{\partial}{\partial t}(ap+\frac{\partial u}{\partial x})-\frac{\partial}{\partial x}(\kappa\frac{\partial p}{\partial x})=f(x,t), \quad x\in(0,1),\ \ t\in(0,T],\\
&\nu\frac{\partial u}{\partial x}=0, \quad p=0, \quad at \quad x=0,\ \ t\in[0,T],\\
&u=0, \quad \kappa\frac{\partial p}{\partial x}=0, \quad at \quad x=1,\ \ t\in[0,T],\\
&ap+\frac{\partial u}{\partial x}=\frac{1}{\nu}, \quad at \quad t=0,\ \ x\in(0,1),\\
&[u]=0, \quad [\nu\frac{\partial u}{\partial x}]=0, \quad [p]=0, \quad [\kappa\frac{\partial p}{\partial x}]=0, \quad at \quad x=\xi,\ \ t\in[0,T].
\end{aligned}
\tag{4}
$$

Due to the complexity of the Biot system, analytical solutions in closed form are available only in very special cases. Certainly, the situation gets complicated in the case of inhomogeneous porous media. The choice of the numerical method for the discretization of the poroelasticity system is not obvious. The finite element method currently dominates in solving poroelasticity system, especially when dealing with complex domains (see [4] for further details). Although finite element methods can be applied to the interface problems, however, they usually work on grids which resolve the interfaces. Hence this fact leads to that imposes certain restriction on the method. Moreover, even when the grids resolve the interfaces, standard finite element methods do not provide good approximation for the flux variables. On the other hand, there is variety of successful finite difference and finite volume approaches, where the interfaces are allowed to cross the grid cells (see [5]).

## 2.1 Grids and notations

For the interval $(0,1)$ and $N>1$, we define stepsize $h$ in the following form

$$h:=\frac{2}{2N-1}.$$

To overcome stability difficulties, which often arise when the discretization of the Biot model is done on the collocate grids, the use of staggered grids was proposed in [2], [7]. Two different spatial grids, $\bar{\omega}_p$ to discretize the pressure equation and $\bar{\omega}_u$ to discretize the displacement equation, are employed

$$
\begin{aligned}
\bar{\omega}_p&=\{x_i:\quad x_i=ih,\ i=0,1,...,N-1\},\\
\bar{\omega}_u&=\{x_{i-0.5}:\quad x_{i-0.5}=x_i-0.5h,\ i=1,2,...,N\}.
\end{aligned}
$$

Further, the grids $\omega_p$ and $\omega_u$ are also used

$$\omega_p = \{x_i \in \bar{\omega}_p, \quad i = 1,2,...,N-1\},$$
$$\omega_u = \{x_{i-0.5} \in \bar{\omega}_u, \quad i = 1,2,...,N-1\}.$$

A grid in time with a stepsize $\tau$ is also defined

$$\omega_T = \{t_n : \quad t_n = n\tau, \quad n = 1,2,...,M\}.$$

These grids as designed to represent the values of the pressure $p$ at the grid points $x_i \in \bar{\omega}_p$ and the values of the displacement $u$ at the midpoints $x_{i-0.5} \in \bar{\omega}_u$ of the subintervals $(x_{i-1}, x_i)$. According to these grids, position of the interface $\xi$ could be represented in the form

$$\xi = x_{i_{int}-0.5} + \theta h,$$

where $0 < i_{int} < N$ is an integer and $0 \leq \theta < 1$. Now, the following notations for discrete functions, defined on $\bar{\omega}_u \times \omega_T$ and $\bar{\omega}_p \times \omega_T$, respectively, are introduced

$$\begin{aligned} u &:= u^n := u_i^n := u(x_{i-0.5}, t_n), \\ p &:= p^n := p_i^n := p(x_i, t_n), \\ p^\sigma &:= \sigma p^{n+1} + (1-\sigma)p^n, \\ p^\wedge &:= p^{n+1}. \end{aligned}$$

Moreover we use some notations for the first order forward and backward finite differences on a uniform mesh in the following form

$$\begin{aligned} p_x := p_{x,i} &= \frac{p(x_{i+1}, t) - p(x_i, t)}{h}, \\ p_{\bar{x}} := p_{\bar{x},i} &= \frac{p(x_i, t) - p(x_{i-1}, t)}{h}. \end{aligned}$$

In a similar way we define

$$\begin{aligned} u_x := u_{x,i} &= \frac{u(x_{i+0.5}, t) - u(x_{i-0.5}, t)}{h}, \\ u_{\bar{x}} := u_{\bar{x},i} &= \frac{u(x_{i-0.5}, t) - u(x_{i-1.5}, t)}{h}. \end{aligned}$$

Finally, the finite differences in time are defined

$$\begin{aligned} u_t := u_t^n := u_t(x_{i-0.5}, t_n) &= \frac{u_i^{n+1} - u_i^n}{\tau}, \quad x_{i-0.5} \in \omega_u, \\ p_t := p_t^n := p_t(x_i, t_n) &= \frac{p_i^{n+1} - p_i^n}{\tau}, \quad x_i \in \omega_p. \end{aligned}$$

## 2.2 Finite volume discretization

In order to approximate the differential problem (4) by finite volume method. Firstly the Biot equations are rewritten in the following way

$$\begin{aligned} &-\frac{\partial S}{\partial x}+\frac{\partial p}{\partial x}=0, \quad x\in(0,1),\ \ t\in(0,T], \\ &\frac{\partial}{\partial t}(ap+\frac{\partial u}{\partial x})+\frac{\partial V}{\partial x}=f(x,t), \quad x\in(0,1),\ \ t\in(0,T]. \end{aligned} \tag{5}$$

Now, the first equation in (5) is integrated over the interval $(x_{i-1},x_i)$

$$-\int_{x_{i-1}}^{x_i}\frac{\partial S}{\partial x}dx+\int_{x_{i-1}}^{x_i}\frac{\partial p}{\partial x}dx=0, \tag{6}$$

and the second equation over the interval $(x_{i-0.5},x_{i+0.5})$

$$\int_{x_{i-0.5}}^{x_{i+0.5}}\frac{\partial}{\partial t}(ap+\frac{\partial u}{\partial x})dx+\int_{x_{i-0.5}}^{x_{i+0.5}}\frac{\partial V}{\partial x}dx=\int_{x_{i-0.5}}^{x_{i+0.5}}f(x,t)dx. \tag{7}$$

Hence, in accordance with the interface conditions (1) and (2), some integrals from (6) and (7) can be rewritten as

$$\begin{aligned} &\int_{x_{i-1}}^{x_i}\frac{\partial S}{\partial x}dx=S(x_i)-S(x_{i-1}), \quad \int_{x_{i-0.5}}^{x_{i+0.5}}\frac{\partial V}{\partial x}dx=V(x_{i+0.5})-V(x_{i-0.5}), \\ &\int_{x_{i-1}}^{x_i}\frac{\partial p}{\partial x}dx=p(x_i)-p(x_{i-1}), \quad \int_{x_{i-0.5}}^{x_{i+0.5}}\frac{\partial u}{\partial x}dx=u(x_{i+0.5})-u(x_{i-0.5}). \end{aligned} \tag{8}$$

Using the rectangular quadratic formula, we can write

$$\int_{x_{i-0.5}}^{x_{i+0.5}}\frac{\partial}{\partial t}(ap)dx\approx\frac{\partial p}{\partial t}(x_i)\int_{x_{i-0.5}}^{x_{i+0.5}}a(x)dx\approx a_i\frac{p_i^{n+1}-p_i^n}{\tau},$$

where

$$a_i=\int_{x_{i-0.5}}^{x_{i+0.5}}a(x)dx. \tag{9}$$

In order to approximate the fluxes $S(x)$ and $V(x)$ in (8) in the grid points, with integrating the equation

$$\frac{S(x)}{\nu}=\frac{\partial u}{\partial x},$$

over the interval $(x_{i-0.5},x_{i+0.5})$ and the equation

$$\frac{V(x)}{\kappa} = -\frac{\partial p}{\partial x},$$

over the interval $(x_{i-1}, x_i)$ we will have the following integral equations

$$\int_{x_{i-0.5}}^{x_{i+0.5}} \frac{S(x)}{\nu} dx = \int_{x_{i-0.5}}^{x_{i+0.5}} \frac{\partial u}{\partial x} dx, \quad \int_{x_{i-1}}^{x_i} \frac{V(x)}{\kappa} dx = -\int_{x_{i-1}}^{x_i} \frac{\partial p}{\partial x} dx.$$

Then, by applying approximate formulae for integrals, we can transform these equations into the following form

$$S(x_i) \int_{x_{i-0.5}}^{x_{i+0.5}} \frac{dx}{\nu(x)} dx \approx u_{i+0.5} - u_{i-0.5}, \quad V(x_{i-0.5}) \int_{x_{i-1}}^{x_i} \frac{dx}{\kappa(x)} dx \approx -(p_i - p_{i-1}).$$

From these two formulae, approximate expressions for fluxes can be found

$$S(x_i) \approx S_i = \nu_i \frac{u_{i+0.5} - u_{i-0.5}}{h}, \quad V(x_{i-0.5}) \approx V_i = -\kappa_i \frac{p_i - p_{i-1}}{h},$$

where

$$\nu_i = \left(\frac{1}{h} \int_{x_{i-0.5}}^{x_{i+0.5}} \frac{dx}{\nu(x)}\right)^{-1}, \quad \kappa_i = \left(\frac{1}{h} \int_{x_{i-1}}^{x_i} \frac{dx}{\kappa(x)}\right)^{-1}. \tag{10}$$

After the substitution of approximate expressions for all the integrals into equations (6) and (7), weighted discretization in time with the weight parameter $\sigma$ is applied. This procedure produces a finite difference scheme, which is a discrete analogue of the problem (4). The obtained finite difference scheme is theoretically investigated and detailed convergence analysis is presented in [6]. Using non-index notations, this scheme for the discrete approximate solution $u = u_i^n$ at point $(x_{i-0.5}, t_n) \in \omega_u \times \omega_T$ and $p = p_i^n$ at grid point $(x_i, t_n) \in \omega_p \times \omega_T$ can be written as in the following form

$$-\tfrac{\nu}{h} u_x^{\wedge} + p_{\bar{x}}^{\wedge} = 0, \quad x = x_{0.5}\ (i = 1), \quad t \in \omega_T, \quad \nu(x) = \begin{cases} \nu_1 & x \le \xi, \\ \nu_2 & x > \xi. \end{cases}$$

$$-\nu_1 (u_{\bar{x}}^{\wedge})_x + p_{\bar{x}}^{\wedge} = 0, \quad x_i \le \xi, \ (i = 2, 3, ..., N-1), \quad t \in \omega_T,$$

$$-\tfrac{\nu_2}{h} u_x^{\wedge} + \tfrac{\nu_1}{h} u_{\bar{x}}^{\wedge} + p_{\bar{x}}^{\wedge} = 0, \quad x_{i-1} \le \xi, \quad x_i > \xi, \ (i = 2, 3, ..., N-1), \quad t \in \omega_T,$$

$$-\nu_2 (u_{\bar{x}}^{\wedge})_x + p_{\bar{x}}^{\wedge} = 0, \quad x_{i-1} > \xi, \ (i = 2, 3, ..., N-1), \quad t \in \omega_T,$$

$$(ap + u_x)_t - \kappa_1 (p_{\bar{x}}^{\sigma})_x = f^{\sigma}, \quad x_{i-0.5} \le \xi, \ (i = 1, 2, ..., N-2), \quad t \in \omega_T, \tag{11}$$

$$(ap + u_x)_t - \tfrac{\kappa_2}{h} p_x^{\sigma} + \tfrac{\kappa_1}{h} p_{\bar{x}}^{\sigma} = f^{\sigma}, \quad x_{i-0.5} \le \xi, \ x_{i+0.5} > \xi, \ (i = 1, 2, ..., N-2), \quad t \in \omega_T,$$

$$(ap + u_x)_t - \kappa_2 (p_{\bar{x}}^{\sigma})_x = f^{\sigma}, \quad x_{i-0.5} > \xi, \ (i = 1, 2, ..., N-2), \quad t \in \omega_T,$$

$$(ap + u_x)_t + \tfrac{\kappa}{h} p_{\bar{x}}^{\sigma} = f^{\sigma}, \quad x = x_{N-1}, \ (i = N-1), \quad t \in \omega_T, \quad \kappa(x) =$$

$$\begin{cases} \kappa_1 & x_{N-1.5} \leq \xi, \\ \kappa_2 & x_{N-1.5} > \xi. \end{cases}$$

$p_0 = 0, \quad u_N = 0, \quad t \in \omega_T,$

$ap + u_x = \frac{1}{\nu}, \quad x = x_i \in \bar{\omega}_p, \ (i = 1, 2, ..., N-1), \quad t = 0,$

where coefficients $a$, $\kappa$ and $\nu$ are calculated according to the formula (9), (10) and the right hand side $f$ is defined as

$$f_i(t) = \frac{1}{h} \int_{x_{i-0.5}}^{x_{i+0.5}} f(x,t) dx.$$

## 3 Numerical results

In this section, results of the numerical experiment are presented. Convergence of all unknowns of the system, i.e., $u$ and $p$ produced by the scheme (11) with respect to the exact solution of the continuous problem are shown. The numerical solution is compared to the known exact solution in discrete $L_2$ norm, which is calculated according to the following form

$$\|\varepsilon_w\|_{L_2} = h \sum_{x_i \ \in \ \omega_w} |w_{ext}(x_i, t_{n+1}) - w_{app}(x_i, t_{n+1})|^2,$$

where $w_{ext}$ and $w_{app}$ stand for the exact and numerical solutions, respectively and $w = \{u, p\}$. In the following experiment, weight parameter is $\sigma = 0.5$.

**Example 3.1** Suppose the following values of the parameters are used:

$$\begin{aligned} \nu_1 &= 1, \quad \nu_2 = \frac{\tan(\frac{1}{12}) \tan(\frac{10\pi}{3})}{8\pi}, \\ \kappa_1 &= 1, \quad \kappa_2 = \frac{1}{8\pi \tan(\frac{1}{12}) \tan(\frac{10\pi}{3})}, \\ a_1 &= 0, \quad a_2 = 0, \quad f(x,t) = 0. \end{aligned}$$

The position of the interface is at $\xi = \frac{1}{6}$. There is no exact solution of problem (4). Now, consider the following initial condition

$$(a-1)p + \nu \frac{\partial u}{\partial x} = 0 \quad at \quad t = 0.$$

If we substitute the above condition in problem (4), the exact solution is as the following forms

$$p(x,t) = \begin{cases} \cos(\frac{10\pi}{3})\sin(\frac{x}{2})\exp(-0.25t) & x \leq \xi, \\ \\ \sin(\frac{1}{12})\cos(4\pi(1-x))\exp(-0.25t) & x > \xi, \end{cases}$$

$$u(x,t) = \begin{cases} -2\cos(\frac{10\pi}{3})\cos(\frac{x}{2})\exp(-0.25t) & x \leq \xi, \\ \\ -2\frac{\cos(\frac{1}{12})}{\tan(\frac{10\pi}{3})}\sin(4\pi(1-x))\exp(-0.25t) & x > \xi. \end{cases}$$

Note that the mesh size $h$ is decreased in a way, preserving a constant value for the parameter $\theta$ in the expression $\xi = x_{i-0.5} + \theta h$. The convergence results are given for times $t = 0.1$ and $t = 1$. All numerical results are shown in Tables 1 and 2. In Figures 3.1(a-e) and 3.2(f-j), we have the convergence of displacement for given stepsizes $h$=0.1, 1/40, 1/160, 1/640 and 1/2560 at $t$=0.1 and $t$=1, respectively. Also, Figures 3.3(k-o) and 3.4(p-t) are prepared for representation of convergence of pressure for given stepsizes $h$=0.1, 1/40, 1/160, 1/640 and 1/2560 at $t$=0.1 and $t$=1, respectively. In all of these figures, exact and approximate solutions are represented by continuous lines and broken lines, respectively.

Table 1: Convergence in discrete $L_2$ norm at the time $t = 0.1$.

| $h = \tau$ | $\|\varepsilon_u\|$ | $\|\varepsilon_p\|$ |
|---|---|---|
| $\frac{1}{10}$ | 0.010978550724151 | $2.584356430578406 \times 10^{-5}$ |
| $\frac{1}{40}$ | $4.299657739862227 \times 10^{-7}$ | $1.565023743780048 \times 10^{-6}$ |
| $\frac{1}{160}$ | $5.255105309209073 \times 10^{-9}$ | $4.642021997862129 \times 10^{-7}$ |
| $\frac{1}{640}$ | $5.263842780036844 \times 10^{-10}$ | $2.411707866280830 \times 10^{-7}$ |
| $\frac{1}{2560}$ | $3.408942895693747 \times 10^{-11}$ | $6.152003882582387 \times 10^{-8}$ |

Table 2: Convergence in discrete $L_2$ norm at the time $t = 1$.

| $h = \tau$ | $\|\varepsilon_u\|$ | $\|\varepsilon_p\|$ |
|---|---|---|
| $\frac{1}{10}$ | $6.010657532287119 \times 10^{-5}$ | $2.069371443954429 \times 10^{-5}$ |
| $\frac{1}{40}$ | $2.741582817269723 \times 10^{-7}$ | $9.979031969906696 \times 10^{-7}$ |
| $\frac{1}{160}$ | $3.350803084864384 \times 10^{-9}$ | $2.959883906284094 \times 10^{-7}$ |
| $\frac{1}{640}$ | $3.356374342242523 \times 10^{-10}$ | $1.537772829030507 \times 10^{-7}$ |
| $\frac{1}{2560}$ | $2.173637957624389 \times 10^{-11}$ | $3.922690864486794 \times 10^{-8}$ |

Figure 3.1(a)

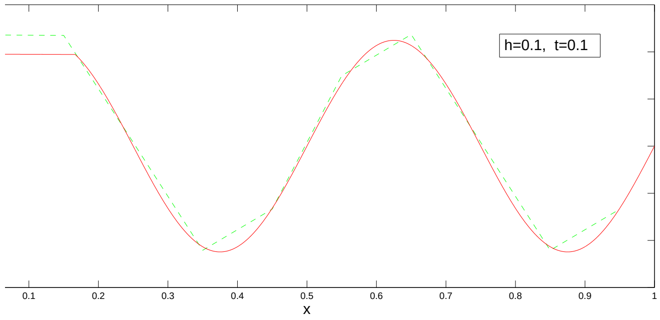


Figure 3.1(b)

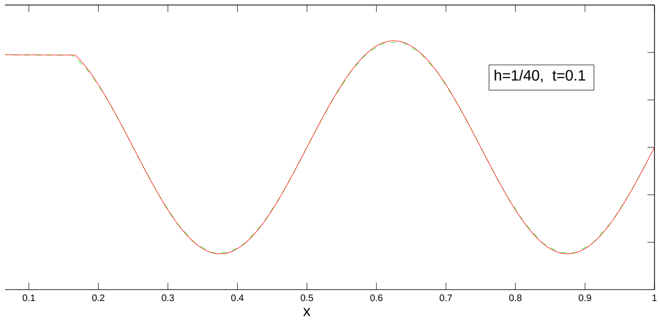


Figure 3.1(c)

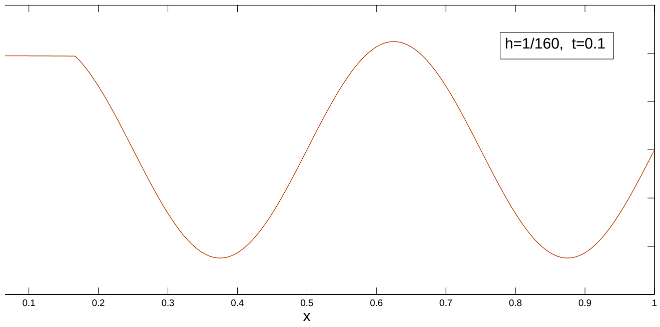


Figure 3.1(d)

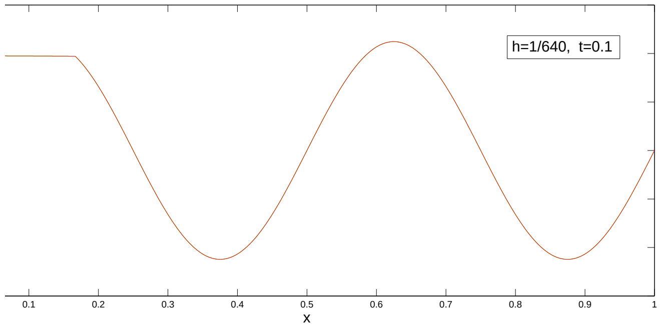


Figure 3.1(e)

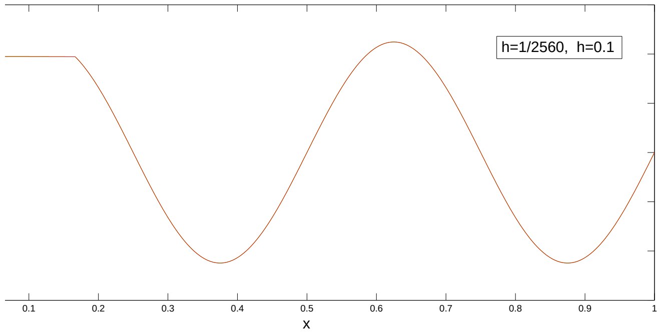
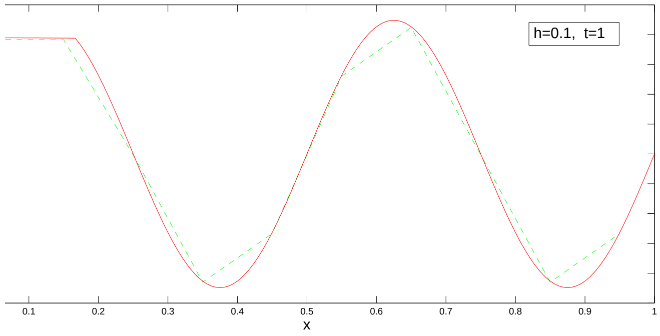


Figure 3.2(f)



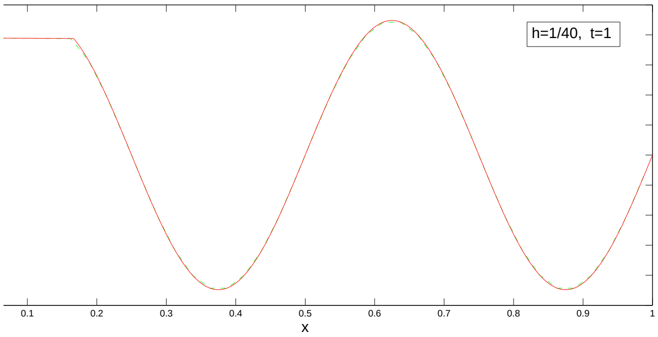


Figure 3.2(g)

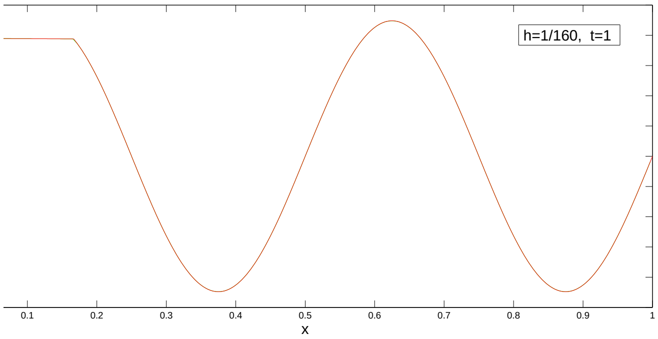


Figure 3.2(h)

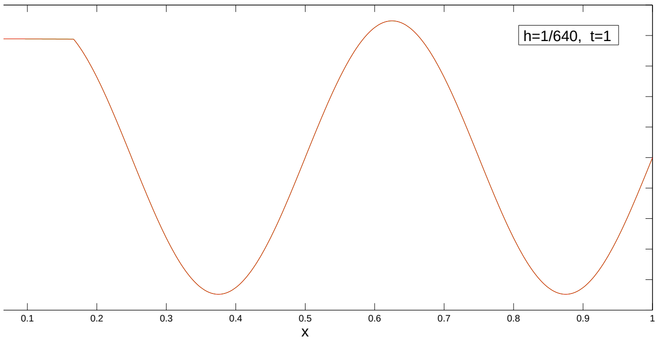


Figure 3.2(i)

Figure 3.2(j)

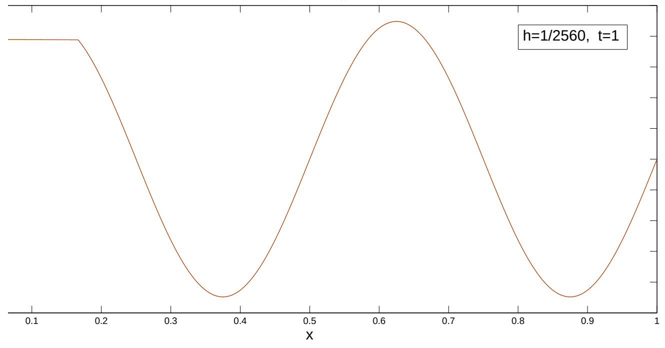


Figure 3.3(k)

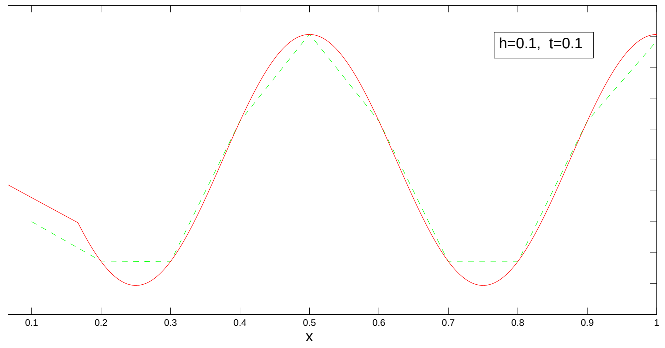


Figure 3.3(l)

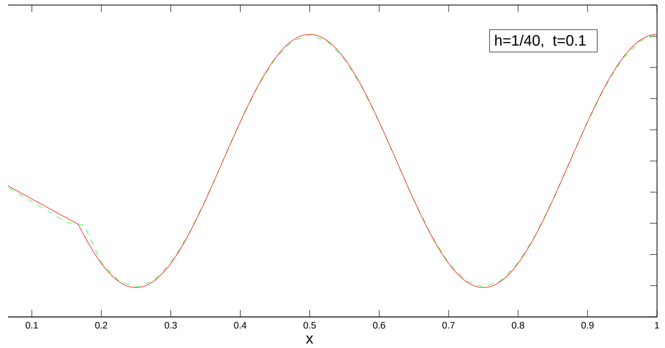


Figure 3.3(m)

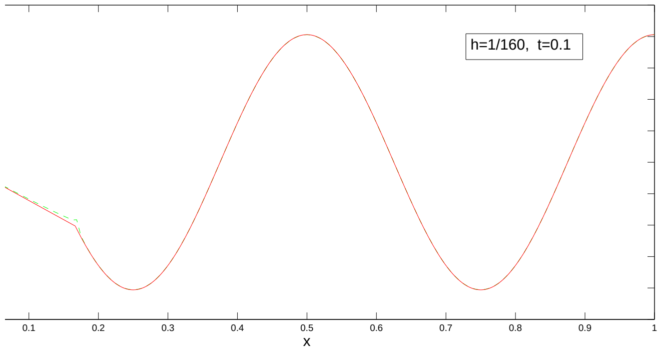


Figure 3.3(n)

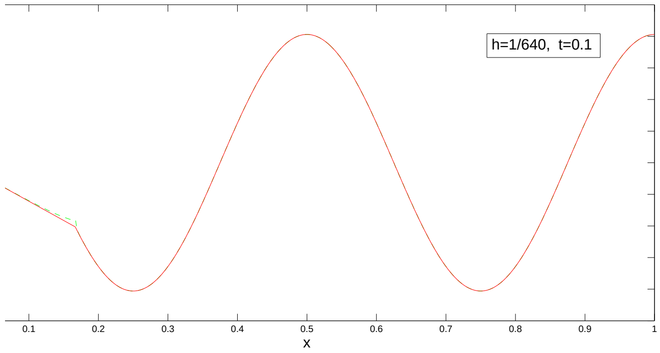


Figure 3.3(o)

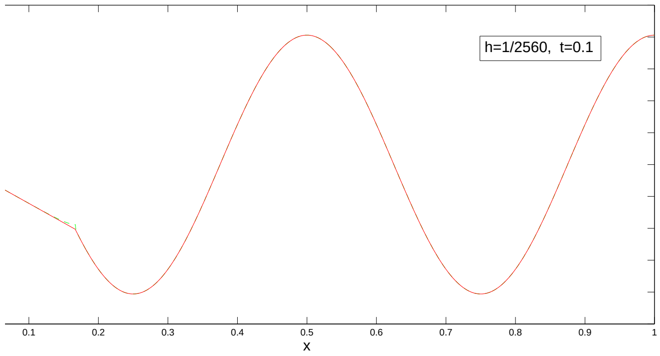


Figure 3.4(p)

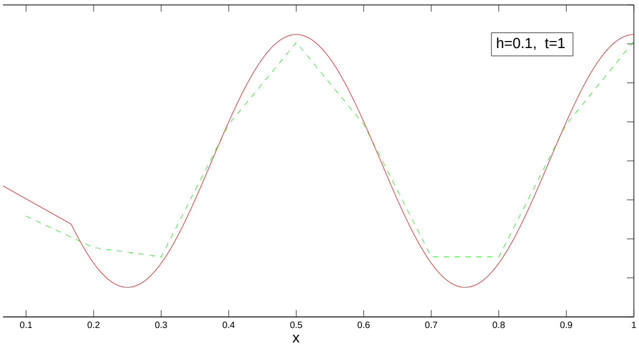


Figure 3.4(q)

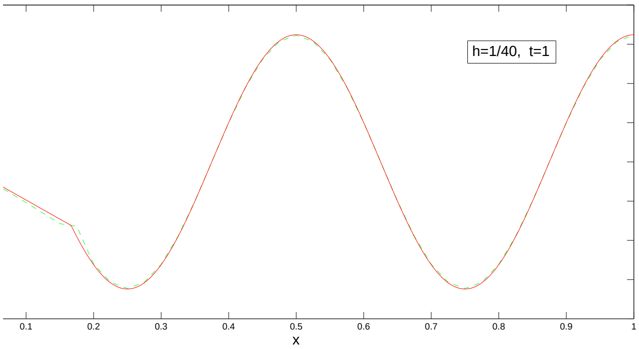


Figure 3.4(r)

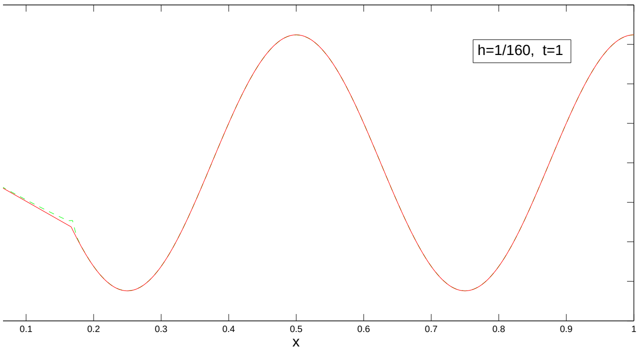


Figure 3.4(s)

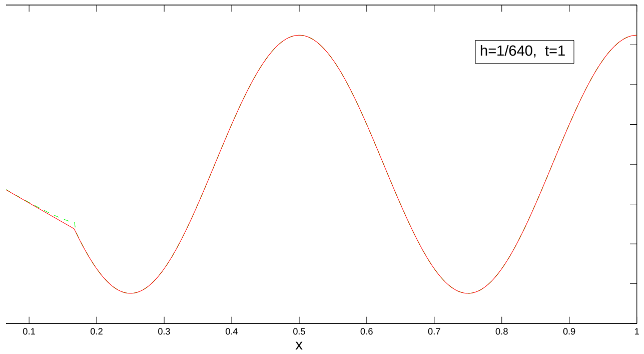
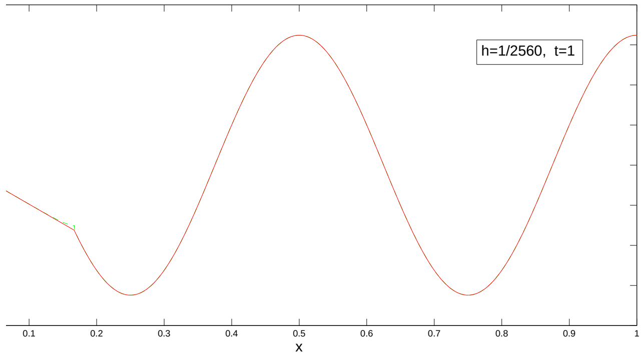


Figure 3.4(t)



# 4 Conclusion

In this paper, we derived a finite volume discretization for the Biot system with continuous coefficients. In example 3.1, numerical solution was compared to the known exact solution in descrete $L_2$ norm. In Tables 1 and 2 and derived figures, when the stepsize $h$ becames smaller, we derived better results. In fact, our numerical experiments confirmed the theoretical considerations.

# References

1. Ewing, R., Liev, O., Lazarov, R. and A. Naumovich, *On convergence of certain finite volume difference discretizations for 1D poroelasticity interface problems*, Wiley Inter-Science, DOI 10. 1002/ num. 20184, (2006).
2. Gaspar, F. J., Lisbona, F. J. and Vabishchevich, P. N., *A finite difference analysis of Biot' s consolidation model*, Departamento de Matematica Aplicada, University of Zaragoza, Zaragoza, Spain., Applied numerical mathematics **4**, (2003), 487-506.
3. Gurevich, B. and Schoenberg, M., *Interface conditions for Biot' s equations of poroelasticity*, J. Acoust. Soc. Am., **105** (5), (1999), 2585-2589.
4. Korsawe, J. and Starke, G., *A least - squares mixed finite element method for Biot's consolidation problem in porous media* , SIAM Journal on numerical analysis, **43** (1), (2005), 318 - 339.
5. Liev, O., *Finite volume discretizations for elliptic problems with discontinuous coefficients*, Habilitation, University of Kaiserslautern, (2002).
6. Naumovich, A., *Efficient numerical methods for the Biot poroelasticity system in multilayered domains*, Vom fachbereich mathematik, Der universitat kaiserslautern, Zur verleihung des akademischen grades, (2007).
7. Versteeg, H. K. and Malalaskera, W., *An introduction to computational fluid dynamics*: The finite volume method, Longman scientific and technical, (1995).

## Aims and scope

Iranian Journal of Numerical Analysis and Optimization (IJNAO) is published twice a year by the Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad. Papers dealing with different aspects of numerical analysis and optimization, theories and their applications in engineering and industry are considered for publication.

## Journal Policy

After receiving an article, the editorial committee will assign referees. Refereeing process can be followed via the web site of the Journal.
The manuscripts are accepted for review with the understanding that the work has not been published and also it is not under consideration for publication by any other journal. All submissions should be accompanied by a written declaration signed by the author(s) that the paper has not been published before and has not been submitted for consideration elsewhere.

## Instruction for Authors

The Journal publishes all papers in the fields of numerical analysis and optimization. Articles must be written in English.
All submitted papers will be refereed and the authors may be asked to revise their manuscripts according to the referee's reports. The Editorial Board of the Journal keeps the right to accept or reject the papers for publication.
The papers with more than one authors, should determine the corresponding author. The e-mail address of the corresponding author must appear at the end of the manuscript or as a footnote of the first page.
It is strongly recommended to set up the manuscript by Latex or Tex, using the template provided in the web site of the Journal. Manuscripts should be typed double-spaced with wide margins to provide enough room for editorial remarks.
References should be arranged in alphabetical order by the surname of the first author as examples below:
[1] Stoer, J. and Bulirsch, R. *Introduction to Numerical Analysis*, Springer-verlag, New York, 2002.
[2] Brunner, H. *A survey of recent advances in the numerical treatment of Volterra integral and integro-differential equations*, J. Comput. Appl. Math. 8 (1982), 213-229.

## Submission of Manuscripts

Authors may submit their manuscripts by either of the following ways:

a) Online submitssion (pdf or dvi files) via the website of the journals at:

http://jm.um.ac.ir/index.php/math

b) Via journal's email mjms@um.ac.ir

## Copyright Agreement

Upon the acceptance of an article by the Journal, the corresponding author will be asked to sign a "Copyright Transfer Agreement" (see the web site) and send it to the Journal address. This will permit the publisher to publish and distribute the work.

**Iranian Journal of Numerical Analysis and Optimization**

Former : MRJMS

CONTENTS Vol. 3, No.1, pp 1-72, Winter 2013

**Web site :** **http://jm.um.ac.ir**

**Email :** mjms@um.ac.ir

**ISSN : 1735-7144**