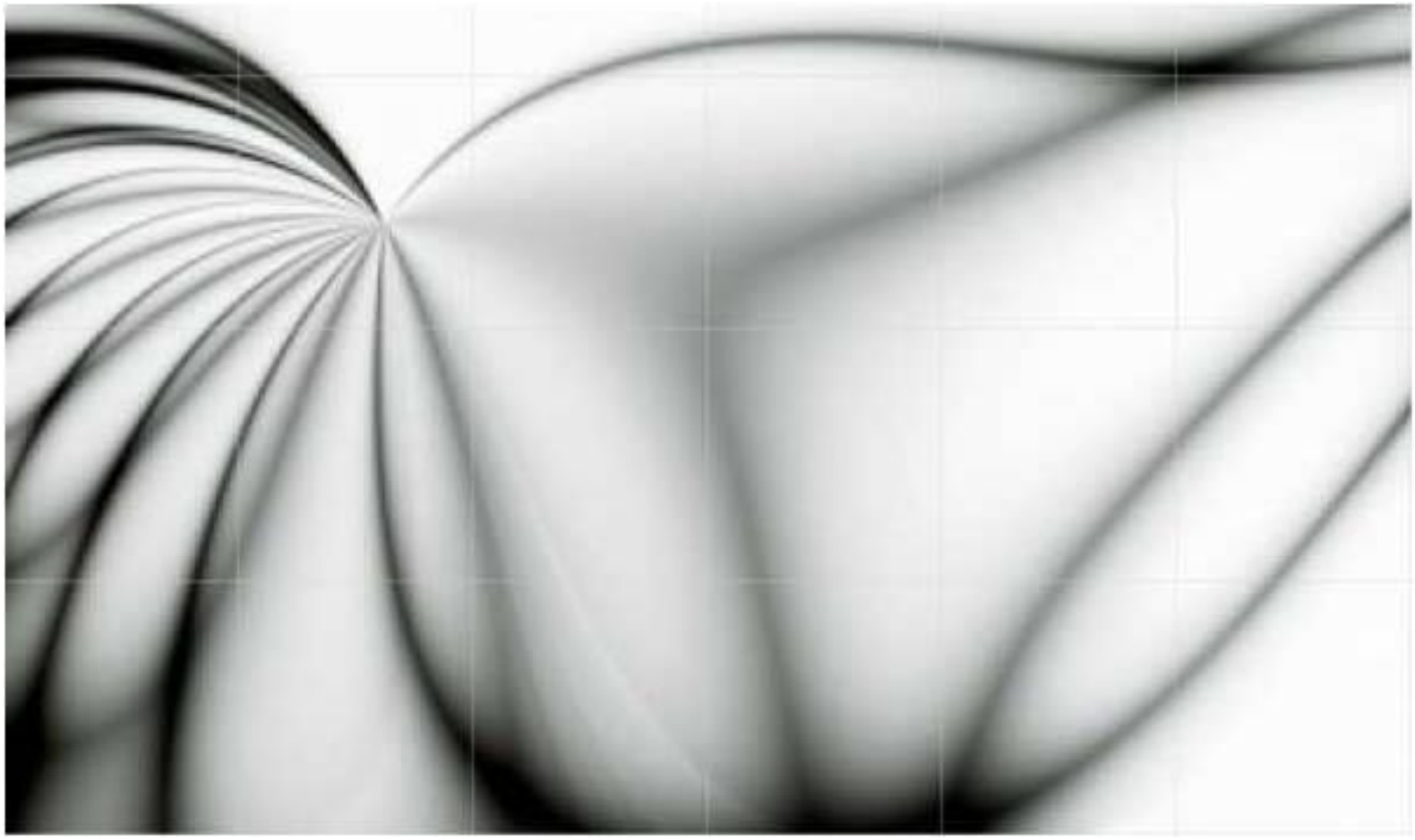
IJOCTA
An International Journal of Optimization and Control: Theories & Applications
2010

ISSN:2146-0957
eISSN:2146-5703

Volume:11 Number:2
July 2021

An International Journal of Optimization and Control:

Theories & Applications



www.ijocta.org

IJOCTA
An International Journal of Optimization and Control: Theories & Applications
2010

http://www.ijocta.org
editor@ijocta.org

An International Journal of Optimization and Control: Theories & Applications

Volume: 11, Number: 2

July 2021

Publisher & Owner *(Yayımcı & Sahibi)*:
Prof. Dr. Ramazan YAMAN
Atlas Vadi Campus 2020, Anadolu St. No. 40, Istanbul, Turkey
Atlas Vadi Kampüsü 2020, Anadolu Cad. No. 40, Avcılar, İstanbul, Türkiye

ISSN: 2146-0957
eISSN: 2146-5703

Press *(Basımevi)*:
Bizim Dijital Matbaa (SAGE Publishing), Kazım Karabekir Street, Kültür Market, No:7 / 101-102, İskitler, Ankara, Turkey
Bizim Dijital Matbaa (SAGE Yayıncılık), Kazım Karabekir Caddesi, Kültür Çarşısı, No:7 / 101-102, İskitler, Ankara, Türkiye

Date Printed *(Basım Tarihi)*:
July 2021
Temmuz 2021

Responsible Director (Sorumlu Müdür):
Prof. Dr. Ramazan YAMAN

IJOCTA is an international, bi-annual, and peer-reviewed journal indexed/ abstracted by *(IJOCTA, yılda iki kez yayımlanan ve aşağıdaki indekslerce taranan/dizinlenen uluslararası hakemli bir dergidir)*:
Cabell's Directories, DOAJ, EBSCO Databases, JournalSeek, Google Scholar, Index Copernicus, International Abstracts in Operations Research, JournalTOCs, Mathematical Reviews (MathSciNet), ProQuest, SCOPUS, ULAKBIM Engineering and Basic Sciences Database (Tubitak), Ulrich's Periodical Directory, and Zentralblatt Math.

crossref

We Use Plagiarism Detection iThenticate Learn More »

iThenticate plagiarism check service is granted by Balikesir University.

An International Journal of Optimization and Control: Theories & Applications
Volume: 11 Number: 2
July 2021

CONTENTS

RESEARCH ARTICLES

An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.11, No.2, pp.123-127 (2021)
https://doi.org/10.11121/ijocta.01.2021.00973

RESEARCH ARTICLE

Kink and anti-kink wave solutions for the generalized KdV equation with Fisher-type nonlinearity

Hüseyin Koçak*

Quantitative Methods Division, Pamukkale University, 20160, Denizli, Turkey
hkocak@pau.edu.tr

ARTICLE INFO

Article history:
Received: 20 April 2020
Accepted: 10 November 2020
Available Online: 2 April 2021

Keywords:
The gKdV-Fisher equation
Dispersion-convection-reaction model
Travelling wave solutions

AMS Classification 2010:
35C07, 76B15, 76V05

ABSTRACT

This paper proposes a new dispersion-convection-reaction model, which is called the gKdV-Fisher equation, to obtain the travelling wave solutions by using the Riccati equation method. The proposed equation is a third-order dispersive partial differential equation combining the purely nonlinear convective term with the purely nonlinear reactive term. The obtained global and blow-up solutions, which might be used in the further numerical and analytical analyses of such models, are illustrated with suitable parameters.

1. Introduction

This study focuses on the travelling wave solutions of a newly introduced dispersion-convection-reaction model

$$u_t + \varepsilon u^n u_x + \mu u_{xxx} = ru\left(1-u^n\right), \tag{1}$$

where $n>0$, ε is a parameter for the purely nonlinear convection term, μ is a parameter for the linear dispersion term and r is a parameter for the purely nonlinear reaction term. One can easily obtain the talented equations, such as the generalized KdV (gKdV) equation [1-9] and the dispersive-Fisher equation [10] by taking $r=0$ and $\varepsilon=0$ in Eq. (1), respectively. There have been many cooperative combinations of dispersion with the different terms, such as dissipation, convection, diffusion and reaction [11-15]. Here, by combining the gKdV equation [2-8] with the Fisher-type (or KPP-type) nonlinearity [16-23], we call Eq. (1) as the gKdV-Fisher equation.

Recently, Galaktionov has focused on the higher-order versions of the KPP (or Fisher) type problem in the parabolic, dispersive and hyperbolic equations, see [21-23] for fruitful discussions. Fortunately, the third-order dispersive partial differential equation including the Fisher-type nonlinearity

$$u_t = u_{xxx} + u\left(1-u\right), \tag{2}$$

with two travelling wave solutions

$$u_1(x,t) = \frac{1}{2} - \frac{9}{4}\tanh\left(\frac{19}{60}t - \frac{1}{60}9900^{1/3}\left(x+\xi_0\right)\right) + \frac{11}{4}\tanh^3\left(\frac{19}{60}t - \frac{1}{60}9900^{1/3}\left(x+\xi_0\right)\right),$$

$$u_2(x,t) = \frac{1}{2} + \frac{3}{4}\tanh\left(\frac{19}{60}t + \frac{1}{60}30^{2/3}\left(x+\xi_0\right)\right) - \frac{1}{4}\tanh^3\left(\frac{19}{60}t + \frac{1}{60}30^{2/3}\left(x+\xi_0\right)\right),$$

has been proposed in [10]. More recently, the exact travelling waves of the KdV-Burgers-Fisher equation

$$u_t + \varepsilon u u_x - \nu u_{xx} + \mu u_{xxx} = ru\left(1-u\right), \tag{3}$$

have been investigated in [24].

We would also like to remind the neighbouring nonlinear parabolic equation, which is a diffusion-convection-reaction model and called the generalized Burgers-Fisher equation [25,26],

$$u_t + \varepsilon u^n u_x - \nu u_{xx} = ru\left(1-u^n\right), \tag{4}$$

with the exact solution

$$u(x,t) = \left(\frac{\tanh\left(\xi\right)+1}{2}\right)^{\frac{1}{n}},$$

where

$$\xi = -\frac{n\varepsilon}{2\nu\left(n+1\right)}x + \frac{n\left(\varepsilon^2 + \nu r\left(n+1\right)^2\right)}{2\nu\left(n+1\right)^2}t + \xi_0.$$

*Corresponding author

In the next section, we use the Riccati equation method [27-36] to reveal the travelling wave solutions of the gKdV-Fisher equation, which are the cooperative results of the proposed combined model. Here, it is reasonable to expect kink and antikink wave solutions because of the reaction term in the proposed equation.

2. Method and application

Let us first take the wave variable $\xi = kx + wt + \xi_0$ and $u(x,t) = u(\xi)$ in Eq. (1) to obtain the reduced nonlinear ODE as

$$wu' + \varepsilon k u^n u' + \mu k^3 u''' - ru\left(1-u^n\right) = 0 . \tag{5}$$

The solution of Eq. (5) is assumed to be expressed as

$$u = \sum_{i=0}^{M} a_i z^i , \tag{6}$$

where the parameters, a_0, …, a_M and M, can be determined later and $z = z(\xi)$ is the solution of the following classical Riccati equation [27-35]:

$$z' = 1 - z^2 , \tag{7}$$

which has the forms

$$z = \tanh(\xi) \text{ and } z = \coth(\xi) . \tag{8}$$

Here, using the advantage of the Riccati equation, higher-order derivatives of Eq. (7) can be obtained as

$$z'' = -2z + 2z^3 , \tag{9}$$

$$z''' = -2 + 8z^2 - 6z^4 . \tag{10}$$

If we next substitute Eq. (6) with Eqs. (7), (9) and (10) into Eq. (5) and balance z''' with $z^n z'$, we have

$$M + 3 = nM + M + 1 \text{ resulting in } M = \frac{2}{n} . \tag{11}$$

In order to obtain the positive integer M values for Eq. (6), we use the transformation

$$u = v^{\frac{2}{n}} \tag{12}$$

in Eq. (5), which yields

$$2wn^2v^2v' + 2\varepsilon kn^2v^4v' + 2\mu k^3n^2v^2v''' - 6\mu k^3n^2vv'v''$$
$$+12\mu k^3nvv'v'' + 4\mu k^3n^2\left(v'\right)^3 - 12\mu k^3n\left(v'\right)^3$$
$$-rn^3v^3 + rn^3v^5 = 0 . \tag{13}$$

If we now apply the same procedure by using the expression for v as

$$v = \sum_{i=0}^{M} a_i z^i \tag{14}$$

in Eq. (13) with Eqs. (7), (9) and (10), and balancing the highest power of z, we have $M = 1$, which yields the solution of Eq. (13) to be in the form

$$v = a_0 + a_1 z . \tag{15}$$

Let us next use Eq. (15) in Eq. (13) and collect the coefficients for the same powers of z as follows:

$$z^0 : \; 4a_1^3k^3\mu n^2 - 4k^3\mu n^2a_0^2a_1 + 2kn^2\varepsilon a_0^4a_1 + n^3ra_0^5$$
$$-12a_1^3k^3\mu n + 8a_1^3k^3\mu - n^3ra_0^3 + 2n^2wa_0^2a_1 = 0 ,$$
$$z^1 : \; 4k^3\mu n^2a_0a_1^2 + 8kn^2\varepsilon a_0^3a_1^2 + 5n^3ra_0^4a_1 + 4n^2wa_0a_1^2$$
$$-24k^3\mu na_0a_1^2 - 3n^3ra_0^2a_1 = 0 ,$$
$$z^2 : \; 16k^3\mu n^2a_0^2a_1 - 4a_1^3k^3\mu n^2 - 2kn^2\varepsilon a_0^4a_1 - 24a_1^3k^3\mu$$
$$+12kn^2\varepsilon a_0^2a_1^3 + 10n^3ra_0^3a_1^2 + 12a_1^3k^3\mu n + 2n^2wa_1^3$$
$$-3n^3ra_0a_1^2 - 2n^2wa_0^2a_1 = 0 , \tag{16}$$
$$z^3 : \; 8k^3\mu n^2a_0a_1^2 - 8kn^2\varepsilon a_0^3a_1^2 + 8kn^2\varepsilon a_0a_1^4 - n^3ra_1^3$$
$$+10n^3ra_0^2a_1^3 + 48k^3\mu na_0a_1^2 - 4n^2wa_0a_1^2 = 0 ,$$
$$z^4 : \; 4a_1^3k^3\mu n^2 - 12k^3\mu n^2a_0^2a_1 - 12kn^2\varepsilon a_0^4a_1^3 - 2n^2wa_1^3$$
$$+2kn^2\varepsilon a_1^5 + 5n^3ra_0a_1^4 + 12a_1^3k^3\mu n + 24a_1^3k^3\mu = 0 ,$$
$$z^5 : \; n^3ra_1^5 - 12k^3\mu n^2a_0a_1^2 - 8kn^2\varepsilon a_0a_1^4 - 24k^3\mu na_0a_1^2 ,$$
$$z^6 : \; -4a_1^3k^3\mu n^2 - 2kn^2\varepsilon a_1^5 - 12a_1^3k^3\mu n - 8a_1^3k^3\mu = 0 .$$

As the final step, solving the nonlinear system (16) for a_0, a_1 and nonzero k, w, ε, μ, r, n parameters by using Maple, we have the following solutions for $\varepsilon\mu < 0$:

$$u_{1,2} = \left(\frac{\tanh\left(kx + wt + \xi_0\right) \pm 1}{2}\right)^{\frac{2}{n}} \tag{17}$$

and

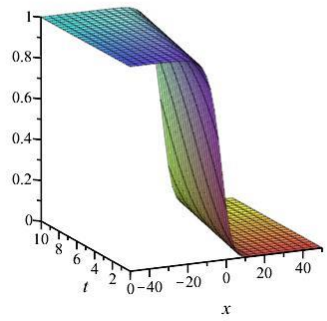
$$u_{3,4} = \left(\frac{\coth\left(kx + wt + \xi_0\right) \pm 1}{2}\right)^{\frac{2}{n}} , \tag{18}$$

where

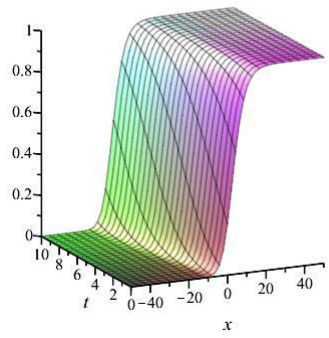
$$r = \mp \frac{\varepsilon\left(n+4\right)\sqrt{-2\varepsilon\mu\left(n+1\right)\left(n+2\right)}}{2\mu\left(n+1\right)^2\left(n+2\right)} ,$$
$$k = \pm \frac{r\left(n+1\right)n}{2\varepsilon\left(n+4\right)} , \tag{19}$$
$$w = \pm \frac{r\left(n^2+6n+12\right)n}{4\left(n+2\right)\left(n+4\right)} .$$

Figure 1 exhibits the long-time behaviour for the global solution $u_1(x,t)$ of the gKdV-Fisher equation in Eq. (17) for the different values of the parameters, which represent kink and antikink waves. One can easily see that the propagations of waves are backward in Figure 1-(b), i.e. $k, w > 0$, and forward in Figure 1-(a) and Figure 1-(c). Another global solution $u_2(x,t)$ in Eq. (17) is displayed in Figure 2 for the given parameters, which also exhibit kink and antikink waves.

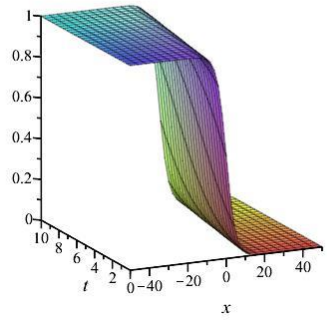
On the other hand, the blow-up solutions $u_3(x,t)$ and $u_4(x,t)$ in Eq. (18) are displayed in Figure 3 and Figure 4, respectively.



(a) $\varepsilon=-1,\ \mu=1,\ r=5\sqrt{3}/12,\ n=1,\ \xi_0=0.$



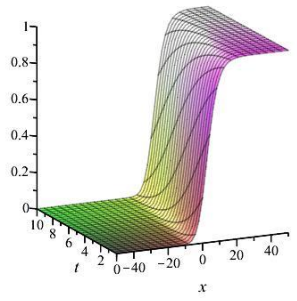
(b) $\varepsilon=1,\ \mu=-1,\ r=5\sqrt{3}/12,\ n=1,\ \xi_0=0.$



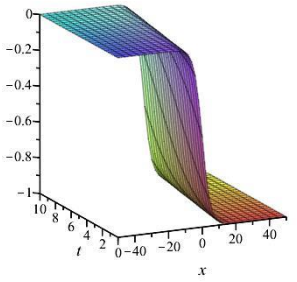
(c) $\varepsilon=-1,\ \mu=1,\ r=\sqrt{6}/6,\ n=2,\ \xi_0=0.$

Figure 1. The solution $u_1(x,t)$ of the gKdV-Fisher equation for the different suitable parameters.

Because of the nature of the complex nonlinear phenomena, it is reasonable to find the blow-up solutions for the mix of the different entities with nonlinearity. Fortunately, we can reveal the cooperative combinations of dispersion, convection and reaction with the parameters in Eq. (19) for the global solutions of the gKdV-Fisher equation, which represent kink and antikink waves, see Figure 1 and 2.

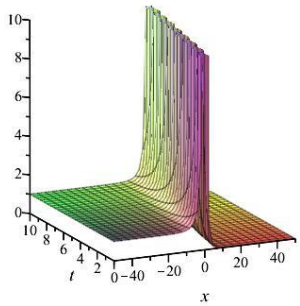


(a) $\varepsilon=-1,\ \mu=1,\ r=-5\sqrt{3}/12,\ n=1,\ \xi_0=0.$

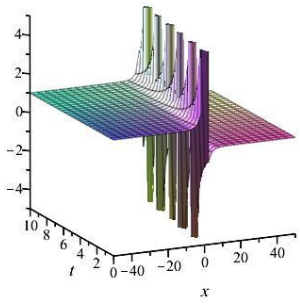


(b) $\varepsilon=-1,\ \mu=1,\ r=-\sqrt{6}/6,\ n=2,\ \xi_0=0.$

Figure 2. The solution $u_2(x,t)$ of the gKdV-Fisher equation for the different suitable parameters.

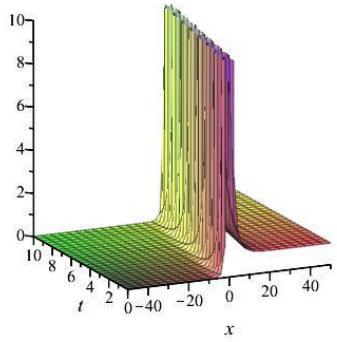


(a) $\varepsilon=-1,\ \mu=1,\ r=5\sqrt{3}/12,\ n=1,\ \xi_0=0.$

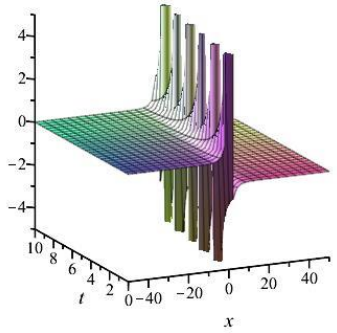


(b) $\varepsilon=-1,\ \mu=1,\ r=\sqrt{6}/6,\ n=2,\ \xi_0=0.$

Figure 3. The solution $u_3(x,t)$ of the gKdV-Fisher equation for the different suitable parameters.



(a) $\varepsilon=-1$, $\mu=1$, $r=-5\sqrt{3}/12$, $n=1$, $\xi_0=0$.



(b) $\varepsilon=-1$, $\mu=1$, $r=-\sqrt{6}/6$, $n=2$, $\xi_0=0$.

Figure 4. The solution $u_4(x,t)$ of the gKdV-Fisher equation for the different suitable parameters.

3. Conclusion

A nonlinear dispersion-convection-reaction model, called the gKdV-Fisher equation, has been introduced to investigate the travelling wave solutions. A classical and efficient the Riccati equation method has been used to investigate two new global and two new blow-up solutions. One can easily see that the reaction term in the proposed equation yields kink and antikink wave solutions, which can be used in the other various numerical and analytical investigations on the application of such combined model to scientific problems. Further research would be based on investigating N-soliton solutions of the third and higher odd-order PDEs including Fisher-type nonlinearity.

Acknowledgments

The author would like to thank Prof. Dr. V. A. Galaktionov for the guidance in the PDE world.

References

[1] Korteweg, D. J., & de Vries, G. (1895). On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 39(240), 422-443.

[2] Tsutsumi, M., Mukasa, T., & Iino, R. (1970). On the generalized Korteweg-de Vries Equation. *Proceedings of the Japan Academy*, 46(9), 921-925.

[3] Gardner, C. S., Greene, J. M., Kruskal, M. D., & Miura, R. M. (1974). Korteweg-devries equation and generalizations. VI. methods for exact solution. *Communications on Pure and Applied Mathematics*, 27(1), 97-133.

[4] Tagare, S. G., & Chakrabarti, A. (1974). Solution of a generalized Korteweg-de Vries equation. *Physics of Fluids*, 17, 1331-1332.

[5] Rosenau, P., & Hyman, M. (1993). Compactons: Solitons with Finite Wavelength. *Physical Review Letters*, 70, 564-567.

[6] Colliander, J. E., & Kenig, C. E. (2002). The generalized Korteweg-de Vries equation on the half line. *Communications in Partial Differential Equations*, 27, 2187-2266.

[7] Wazwaz, A. M. (2006). Kinks and solitons for the generalized KdV equation with two power nonlinearities. *Applied Mathematics and Computation*, 183, 1181-1189.

[8] Tao, T. (2007). Two remarks on the generalised Korteweg de-Vries equation. *Discrete & Continuous Dynamical Systems - A*, 18, 1-14.

[9] Tao, T. (2008). Global behaviour of nonlinear dispersive and wave equations. *Current Developments in Mathematics*, 2006, 255-340.

[10] Pinar, Z., & Koçak, H. (2018). Exact solutions for the third-order dispersive-Fisher equations. *Nonlinear Dynamics*, 91(1), 421-426.

[11] Rosenau, P. (1998). On a class of nonlinear dispersive-dissipative interactions. *Physica D: Nonlinear Phenomena*, 123(1-4), 525-546.

[12] Wazwaz, A. M. (2006). The tanh method for compact and noncompact solutions for variants of the KdV-Burger and the K(n,n)-Burger equations. *Physica D: Nonlinear Phenomena*, 213, 147-151.

[13] Galaktionov, V. A., Miditieri, E. L., & Pohozaev, S. I. (2014). *Blow-up for higher-order parabolic, hyperbolic, dispersion and Schrödinger equations*. Monographs and Research Notes in Mathematics, Chapman and Hall/CRC, Boca Raton.

[14] Koçak, H. (2017). Similarity solutions of nonlinear third-order dispersive PDEs: The first critical exponent. *Applied Mathematics Letters*, 74, 108-113.

[15] Koçak, H., & Pinar Z. (2018). On solutions of the fifth-order dispersive equations with porous medium type non-linearity. *Waves in Random and Complex Media*, 28(3), 516-522.

[16] Fisher, R. A. (1937). The wave of advance of advantageous genes. *Annals of Eugenics*, 7(4), 355-369.

[17] Kolmogorov, A. N., Petrovskii, I. G., & Piskunov,

N. S. (1937). Study of the diffusion equation with growth of the quantity of matter and its application to a biological problem. *Byull. Moskov. Gos. Univ.*, Sect. A, 1, 1–26. (English. transl. In: *Dynamics of Curved Fronts*, P. Pelce, Ed., Acad. Press, Inc., New York, 1988, 105–130.)

[18] Ablowitz, M. J., & Zeppetella, A. (1979). Explicit solutions of Fisher's equation for a special wave speed. *Bulletin of Mathematical Biology*, 41, 835–840.

[19] Wazwaz, A. M. (2008). Analytic study on Burgers, Fisher, Huxley equations and combined forms of these equations. *Applied Mathematics and Computation*, 195(2), 754-761.

[20] Gilding, B. H., & Kersner, R. (2012). *Travelling waves in nonlinear diffusion-convection-reaction*. Vol 60, Birkhäuser Basel.

[21] Galaktionov, V. A. (2012). Towards the KPP-Problem and log t-Front Shift for Higher-Order Nonlinear PDES II. Quasilinear Bi-and Tri-Harmonic Equations. *arXiv:1210.5063*.

[22] Galaktionov, V. A. (2012). Towards the KPP--Problem and log t-Front Shift for Higher-Order Nonlinear PDEs III. Dispersion and Hyperbolic Equations. *arXiv:1210.5084*.

[23] Galaktionov, V. A. (2013). The KPP-Problem and log t-Front Shift for Higher-Order Semilinear Parabolic Equations. *Proceedings of the Steklov Institute of Mathematics*, 283, 44–74.

[24] Koçak, H. (2020). Travelling Waves in Nonlinear Media with Dispersion, Dissipation and Reaction. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 30(9), 093143.

[25] Chen, H., & Zhang, H. (2004). New multiple soliton solutions to the general Burgers-Fisher equation and the Kuramato-Sivashinsky equation. *Chaos, Soltions and Fractals*, 19, 71-76.

[26] Wazwaz, A. M. (2005). The tanh method for generalized forms of nonlinear heat conduction and Burgers-Fisher equations. *Applied Mathematics and Computation*, 169, 321-338.

[27] Huibin, L., & Kelin, W. (1990). Exact solutions for nonlinear equations. I. *Journal of Physics A: Mathematical and General*, 23(17), 3923-3928.

[28] Malfliet, W. (1992). Solitary wave solutions of nonlinear wave equations. *American Journal of Physics*, 60(7), 650-654.

[29] Malfliet, W., & Hereman, W. (1996). The Tanh Method: I. Exact Solutions of Nonlinear Evolution and Wave Equations. *Physica Scripta*, 54, 563-568.

[30] Fan, E. G. (2002). Traveling Wave Solutions for Nonlinear Equations Using Symbolic Computation. *Computers and Mathematics with Applications*, 43, 671-680.

[31] Kudryashov, N. A. (2005). Simplest equation method to look for exact solutions of nonlinear differential equations. *Chaos, Solitons & Fractals*, 24 (5), 1217-1231.

[32] Kudryashov, N. A. (2009). Seven common errors in finding exact solutions of nonlinear differential equations. *Communications in Nonlinear Science and Numerical Simulation*, 14, 3507-3529.

[33] Wazwaz, A. M. (2009). *Partial Differential Equations and Solitary Wave Theory*. Higher Education Press, Beijing and Springer-Verlag Berlin Heidelberg.

[34] Griffiths, G. W., & Schiesser, W. E. (2011). *Traveling Wave analysis of Partial Differential Equations: Numerical and Analytical Methods with MATLAB and MAPLE*. Academic Press, New York.

[35] Polyanin, A. D., & Zaitsev, V. F. (2012). *Handbook of Nonlinear Partial Differential Equations*. Chapman and Hall/CRC, Boca Raton.

[36] Manafian, J., & Lakestani, M. (2017). A new analytical approach to solve some of the fractional-order partial differential equations. *Indian Journal of Physics*, 91, 243-258.

Hüseyin Koçak *obtained his PhD degree in Mathematical Sciences from the University of Bath, UK in 2015. He has been working as Asst. Prof. of Quantitative Methods at the Pamukkale University since 2016.*

http://orcid.org/0000-0001-9683-6096

An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.11, No.2, pp.128-138 (2021)
https://doi.org/10.11121/ijocta.01.2021.001023

RESEARCH ARTICLE

UAV routing with genetic algorithm based matheuristic for border security missions

Muhammed Kaya and Omer Ozkan*

Department of Industrial Engineering, National Defence University, Turkish Air Force Academy
34149, Yeşilyurt , Istanbul, Turkey
3410kaya@harbiyeli.hho.edu.tr, o.ozkan@hho.edu.tr

ARTICLE INFO

Article history:
Received: 30 September 2020
Accepted: 26 March 2021
Available Online: 19 April 2021

Keywords:
UAV routing
Genetic algorithm
Matheuristic
Border security
Homeland security

AMS Classification 2010:
90C11, 90C59, 90C90

ABSTRACT

In recent years, Unmanned Aerial Vehicles (UAVs) are a good alternative for the problem of ensuring the security of the borders of the countries. UAVs are preferred because of their speed, ease of use, being able to observe many points at the same time, and being more cost-effective in total compared to other security tools. This study is dealt with the problem of the use of UAVs for the security of the Turkey-Syria borderline which becomes sensitive in recent years and the problem is modeled as a UAV routing problem. To solve the problem, a Genetic Algorithm Based Matheuristic (GABM) approach has been developed and 12 scenarios have been created covering the departure bases, daily patrol numbers, and ranges of UAVs. GABM finds the minimum number of UAVs to use in scenarios with the help of a GA run first and tries to find the optimal routes for these UAVs. If GABM can find an optimal route for the determined UAV number, it decreases the UAV number and tries to solve the problem again. GABM proposes a hybrid approach in which a metaheuristic with a mathematical model works together and the metaheuristic sets an upper limit for the number of UAVs in the model. In computational studies, when compared GA with GABM it is seen that GABM has obtained good results and decreased the utilized number of UAVs (up to 400%) and their flight distances (up to 85.99%) for the problem in very short CPU times (max. 122.17 s. for GA and max. 46.39 s. for GABM in addition to GA).

1. Introduction

Ensuring the border security for the countries is one of the critical necessities to maintain homeland security. Creating smart borders by using intelligence and technology with national and international cooperation and coordination can increase to achievement possibility of the border security missions. Smart borders can prevent terrorist attacks, organized crimes, cyber-crimes, the passage of illegal drugs, and illegal migrants through borders [1].

Traditional border security can be maintained by fences, barriers, walls, towers, manned, animal and vehicle patrols, etc. When technology is added to the process in recent years; binoculars, cameras, radars, mobile surveillance equipment, radio, and cell phone data surveillance equipment, helicopters, zeppelins, planes, satellites, wireless sensors, Wireless Sensor Networks (WSNs), autonomous ground vehicles and robots, and Unmanned Aerial Vehicles (UAVs) are the main instruments that are proposed by the literature and implemented by the authorities.

Sensors and WSNs are studied to secure borders. “BorderSense”, a hybrid WSN concept is introduced by [2]. “OptaSense”, a distributed acoustic sensing system that uses acoustic and seismic sensing with fiber optic cables is developed for border security missions and presented by [3]. Decentralized smart sensor scheduling for multiple target tracking for border surveillance is studied by [4]. The deployment of the sensor is optimized by [5] for border surveillance. Large scale border security systems are modeled and simulated with “OPNET” by [6]. By [7], a brief survey about using WSNs for border security and intruder detection is presented and a bi-level exposure-oriented sensor location problem for border security missions is covered by [8].

A method for guidance and control of an autonomous vehicle in problems of border patrolling and obstacle avoidance is proposed by [9]. The “TALOS” project

*Corresponding author

aiming to guard European Union borders with autonomous robotic vehicles is shared by [10,11]. UAVs are proposed for border security, for instance, hierarchical control architecture for a system involving multiple UAVs is proposed by [12]. A paper [13] is published to discuss the implementation of UAVs at borders and the navigation of UAVs on borders is studied by [14]. A report [15] is examined the strengths and limitations of using UAVs along the borders and related subjects for USA Congress.

The study [16] explores the different features when a four-rotor UAV is deployed to the USA's borders and territories. The paper [17] examined the threats and counter responsibilities that require the utilization of UAVs in homeland security. A report [18] is prepared to determine the effectiveness and cost of the UAV programs for border protection by the U.S. Department of Homeland Security. Border surveillance using multiple UAVs in coordination with alert stations including ground sensors along the borderline/fence is proposed by [19]. The usage of UAVs is also examined for search and rescue operations by [20]. Especially, the hybrid systems that combine some of these instruments above with intelligence and data analysis are studied in the last decade.

In the literature, mathematical models [5,8,21], simulation models [6,12,19,22,23] and heuristics/ metaheuristics [1,5,24] are used to solve border security based problems. U.S.-Mexico border is the most studied border [13,15,18,25,26]. There are also studies about Turkey [22] and Spain [27]. The first and the only paper (as we may found) about the border security of Turkey is mainly using a simulation approach to model the border security system of Turkey and including border patrols, ambushes, sentries, thermal cameras, and askarad in the simulation model. However, in our paper, we tried to integrate the UAVs into the border security system as with real-life scenarios.

The usage of UAVs increased in recent years for missions in both military and civilian fields; such as intelligence, surveillance, reconnaissance, monitoring, destruction, communication, search and rescue, transportation, etc. The UAVs are chosen for that kind of mission because they are reliable, secure, long-ranged, remote-controlled (if needed), easy to use, and cheap. The technological details and the opportunities about the UAVs can be found in [28], a literature survey on quadrotor UAVs is presented in [29] and two review papers about UAVs are summarized in [30,31].

In this paper, the usage of the UAVs at the borderline between Turkey and Syria is studied. The internal conflicts in Syria have started in 2011 and since then the importance of the border between these countries is increased from the perspective of Turkey. Millions of civilian immigrants escaping from battles have come to the border. This situation forced Turkey to increase the security precautions on the border to prevent the passage of terrorists hidden in the civilian crowds through the border. A concrete wall through the border has been built. In addition to these precautions, this study is aiming to use UAVs for the security of the borderline between Turkey and Syria (i.e. over the concrete wall). Therefore, the problem can be thought of as an implementation of a UAV routing problem (UAVRP) or in general a vehicle routing problem (VRP).

VRP is a well-studied problem in the literature and there are optimal and approximate solution methods to solve the VRP. The first literature survey about VRP is presented by [32] published in 2009 and the second one is by [33] in 2016. The collaborative VRP is summarized in [34]. Capacitated [35,36], multi-vehicle, and multi-depot [37] versions are also studied in the past years. The traveling salesman problem (TSP) and its multiple salesman versions can also be examined to understand the simple nature of the problem [38,39]. Since the VRP and its versions have high-complexity, quite a lot of types of heuristics and metaheuristics are proposed to solve the VRPs in the literature.

After the increase of the usage of the UAVs for several missions, UAVRP in both 2D and 3D domains has also been studied in the literature. Similar to VRPs, UAVRP and its versions have complex nature; therefore mainly geometric methods, dynamic programming, mathematical models, heuristics, metaheuristics, artificial neural networks, learning-based methods, fuzzy logic, simulation models, etc. are studied in the literature [40,41]. Computational-intelligence based algorithms [42], evolutionary algorithms including differential evolution and genetic algorithm (GA) [43,44], particle swarm optimization, ant colony optimization, simulated annealing [45], tabu search are mainly the proposed metaheuristics to solve the UAVRP and its versions [41]. 3-D UAVRP is reviewed by [46] and Flying Ad-Hoc Networks are also examined by [47]. The first difference between VRP and UAVRP is the distance calculation methods (i.e. Euclidean distance is preferred in UAVRP since the flight between two points can be done directly) and the second one is the usage of distance range constraint is a binding constraint for UAVRPs (if the UAV has not a capability to refuel in the air). Before the fuel of a UAV is finished, the UAV should be landed; therefore the distance range constraint should be a hard constraint. This hard constraint makes the model harder to be solved. In VRP on the land, the distance range constraint can be increased by refueling, therefore; it can be a soft constraint. These differences are considered in the model that this paper proposed.

In summary for the introduction and literature review section, there are different systems to secure borders in the literature. In addition to the traditional security systems, in recent years the usage of unmanned and robotic systems on borders is increased. UAVs are very good alternatives to be included in the border security systems in a hybrid manner but there are not many studies in the literature about this topic and we summarized the very few existing papers above.

Therefore, as a contribution to the literature as far as we know this paper is the first paper that proposes a matheuristic approach (i.e. Genetic Algorithm Based Matheuristic (GABM)) to solve a UAVRP defined for real border security missions. The mathematical model is inspired by literature and adapted to the border security missions. A real UAV is preferred in the case study section and its specifications are used in real case scenarios. In this manner, this work tries to find optimal routes over the borderline of Turkey and Syria for real UAVs considering checkpoints, number of UAVs, number of daily patrols, flight ranges, and main bases that the UAVs take off and land. Generated 12 scenarios are considering real military airports near the border as main bases. The proposed GABM is trying to minimize the used number of UAVs and the traveling distances of the UAVs simultaneously.

This paper has been divided into sections as follows. The next section gives the problem definition and the mathematical model for the problem. The third section covers the details about the proposed GABM and the used GA in the GABM. The fourth section includes the case study part and provides the results of the computations on the case scenarios. The final section concludes the paper.

2. The problem definition and the mathematical model

The UAVRP can be described as a graph $G=(N,A)$ with a set of nodes (N=1..n) and a set of arcs (A=1..a) with a set of UAVs (M=1..m). The UAVRP can be formulated as a Mixed Integer Linear Programming (MILP) model, it is inspired and revised from [38,39] and presented below.

$$\text{min. } z = c_d \sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij} \sum_{k=1}^{m} x_{ijk} + m\, c_m \quad (1)$$

subject to:

$$\sum_{j=2}^{n} x_{1jk} = 1\,; \qquad \forall k;\ 1 \le k \le m \quad (2)$$

$$\sum_{i=2}^{n} x_{i1k} = 1\,; \qquad \forall k;\ 1 \le k \le m \quad (3)$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m} x_{ijk} = 1\,; \qquad \forall i;\ 2 \le i \le n \quad (4)$$

$$\sum_{i=1}^{n}\sum_{k=1}^{m} x_{ijk} = 1\,; \qquad \forall j;\ 2 \le j \le n \quad (5)$$

$$\sum_{i=1}^{n} x_{irk} = \sum_{j=1}^{n} x_{rjk}\,; \quad \forall r;\ 2 \le r \le n\,; \quad \forall k;\ 1 \le k \le m \quad (6)$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}\, x_{ijk} \le R_k\,; \quad \forall k;\ 1 \le k \le m \quad (7)$$

$$v_i - v_j + (n-m)\sum_{k=1}^{m} x_{ijk} \le n - m - 1\,; \quad \forall i;\ \forall j;\ 2 \le i \ne j \le n \quad (8)$$

$$x_{ijk} \in \{0,1\}\,; \quad \forall i,j \in V;\ \forall k \in U$$

$$v_i \ge 0 \text{ and} \in Z; \quad \forall i \in N \quad (9)$$

In the model, there are "m" numbers of UAVs and "n" number of nodes including 1 base station (i.e. first numbered node is the base station) and n-1 numbers of grid (i.e. check) points on the border. The arcs are the flight connection paths between nodes. The d_{ij} is the Euclidean flight distance between nodes i and j. The decision variables x_{ijk} are used to represent the used route between nodes, that x_{ijk} = 1 if the arc between i and j is used, otherwise x_{ijk} = 0. The R_k is the distance range limit of UAVs to finish the tour. In the model, v_i and v_j are the positions of node i and j in the path that is used to prevent the sub-tours.

The objective function (1) minimizes the total cost. The flight cost and the used UAV cost are creating the total cost. The c_d is the 1-kilometer flight cost for a single UAV and c_m is the unit UAV cost. Constraint (2) ensures that "m" number of UAVs take off from node #1 (i.e. base station) and Constraint (3) guarantees the UAVs turn back to the departure base. Constraint (4) and Constraint (5) ensure that it is necessary to visit all checkpoints. Constraint (6) guarantees that if a UAV visits a node, it also departures from that node and Constraint (7) limits the total tour distance according to the flight range of the UAVs. Constraint (8) is essential to prevent the sub-tours. Constraint (9) is defining the decision variables.

One of the main assumptions in this model is that the UAV(s) takes off from any base station and land at any base station. When UAV visits a node that means it can visualize and observe the borderline on that point. The UAV directly flies from one point to another without changing the route in any event that occurs, for instance, the detection of an intruder from the borderline. The UAV just informs the authorities when an event happens. The meteorological effects on the flight times are ignored and the regular speed of UAVs is assumed as constant. The breakdowns of UAVs on the air are also ignored and assumed that the UAVs are always operating. It is also assumed that the grid points have the same importance weight, since in the border security missions even one border violation may cause terrorist attacks in the homeland. Therefore, the security overall of the border should be maintained. However, if there are more difficult areas to be secured via UAVs, the precautions can be increased by using other traditional security tools (patrols, fences, concrete walls, sensors, cameras, etc.).

In the implementation of scenarios, the proposed model has been revised and able to solve multi-base station UAVRPs via changes in Constraints (2) and (3). In the revised model, there are "o" numbers of base stations, and the "m" numbers of UAVs can take off from any of these base stations and can land at any of them.

3. The proposed genetic algorithm based matheuristic

The GA is a well-known and well-studied metaheuristic in the literature. According to the literature [41,43], the GA is the most-preferred metaheuristic to solve UAV routing-based problems and the effectiveness of this algorithm is presented in these studies. Therefore the GA is chosen for this study. The GABM is starting with running a problem-specific GA 30 times and the pseudo-code of the designed GA is presented below in Figure 1.

Begin
Set initial parameters (μ, λ, k, P_c, P_m, e, g_{max})
Generate μ # of individual (initial population)
Evaluate initial population by using f
$g \rightarrow 0$ /* g: generation number */
repeat
Apply tournament to select parents by using k
Generate λ # of new children with crossover by using P_c
Apply mutation by using P_m
Evaluate new children (new population) by using f
Apply elitism by using e
Apply environmental selection (update population)
$g \rightarrow g + 1$
until $g = g_{max}$
Show best solution
End

Figure 1. Pseudo-code of developed GA.

The designed GA starts searching with a group of solutions (i.e. a population) and generates better individuals (i.e. solutions) by implementing crossover, mutation, and selection operators through generations. The details about the GA are described in the following subtitles.

The proposed GA and the proposed mathematical model are hybridized in the GABM. The hybridization of optimization methods aims to use powerful sides of them in a single algorithm. The GA is used in GABM to determine the "*m*" value of the proposed mathematical model. Since in the proposed model, if the "*m*" value becomes a decision variable, the model becomes more complex. The GA helps the model to use the "*m*" value as a constant parameter. Therefore, the GABM is trying to find an optimal solution to the UAVRP as described in Figure 2.

The proposed GABM acts to decrease the complexity of the UAVRP, and helps to find a faster solution. The used GA and its specifications are described in the following subtitles.

3.1. Representation and fitness function

In the GA, a problem-specific permutation-based representation is used to indicate the solutions. A simple example of the representation is presented in Figure 3.

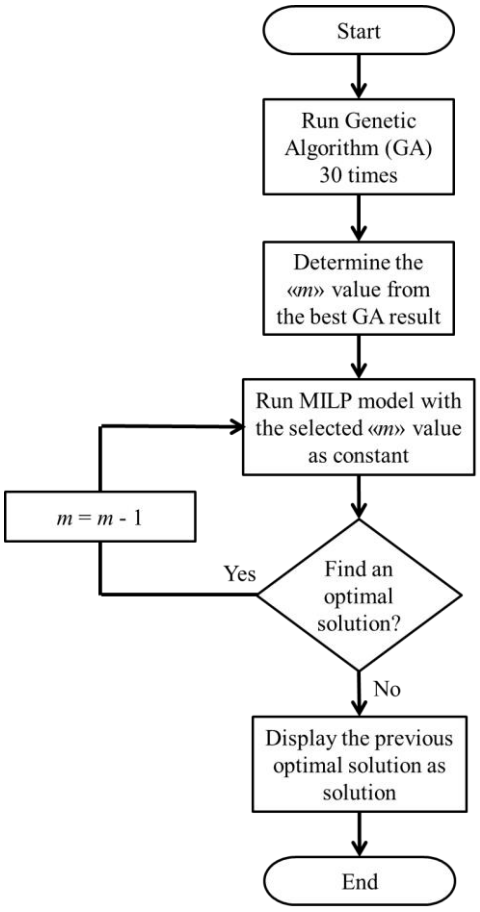


Figure 2. The flowchart of the proposed GABM.

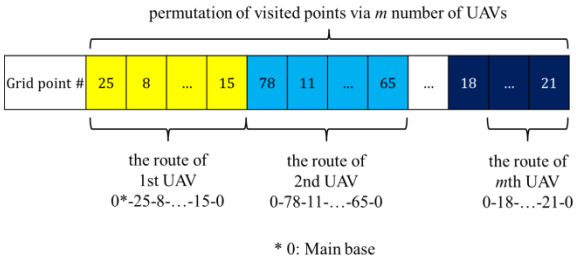


Figure 3. Permutation-based representation.

The permutation of the visited grid (i.e. check) points is showing the routes of the "*m*" number of UAVs. As seen in the example, the first UAV is taking off from a base station, visiting the grid points #25,#8,…,#15 (i.e. the yellow points) in order and turning back to a base station. The first UAV cannot exceed its flight range; therefore it cannot visit grid point #78 and turns back to a base station. The second UAV continues to fly in a route from the grid points #78,#11,…,#65 (i.e. the blue points) in order similarly. The UAVs always take off from or land to the nearest active base station. The order of the grid points indicates the routes of the all "*m*" number of UAVs. The total cost (i.e. objective function) described in Eq.(1) is used as a fitness function for the solutions. The individuals with lower costs mean that they are better solutions.

3.2. Initial population and parental selection

While generating μ number of solutions as initial population, five different strategies are used in GA. In the first strategy, when a UAV is monitoring the border on the field it is logical to fly from one grid point to its one of two neighbor points, therefore one individual of the initial population is simply starting from the first grid point and the route follows forward through the nearest neighbor points. The second solution is beginning from the last grid point and the route continues backward through the nearest neighbor points. The third strategy is to generate random individuals. For fourth and fifth strategies, random initial grid points are selected and the route continues forward or backward till the end or beginning points, respectively as type-1 and 2 heuristics. The sample initial solutions are presented in Figure 4 when μ=20.

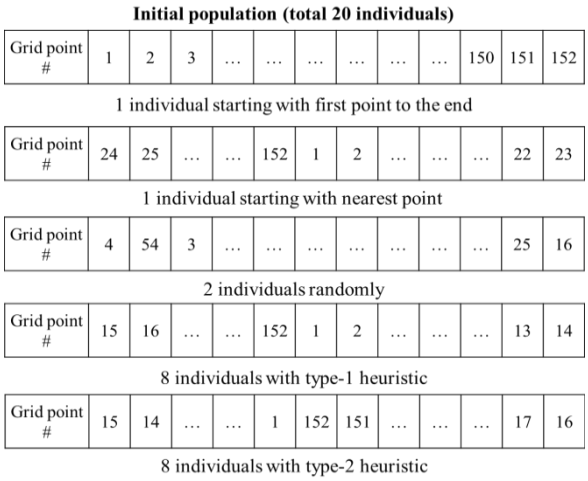


Figure 4. Sample initial population.

The tournament selection operator is used to determine parents in the proposed GA. In the tournament selection, the operator selects k number of individuals randomly from the current population, and the solution with the lowest fitness wins the tournament and becomes a parent. The μ and k parameters are tuned before the case study.

3.3. Crossover and mutation

The crossover probability (P_c) is used to determine whether the selected parent can be a candidate in the crossover. The order-1 crossover operator is selected to generate new children as seen in Figure 5.

The operator uses two parents and it selects two random breakpoints in the representation of the parents (i.e. yellow parts). The operator copies the grid points in the interval of selected breakpoints from the first parent to the child. The copying process continues with the second parent from the outside of the last breakpoint. The operator prevents generating infeasible children according to the permutation-based solution type and copies only the uncovered grid points in the current child from the second parent. When the last grid point of the second parent is reached, the operator turns back to the beginning point of the second parent. The process finishes when all grid points are covered in the child.

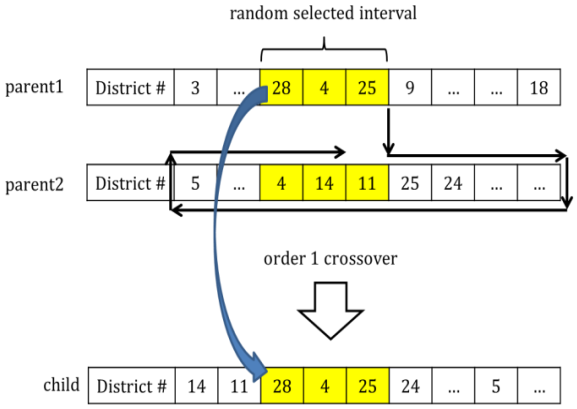


Figure 5. Order-1 crossover.

The swap mutation is used as the mutation operator as seen in Figure 6 and mutation probability (P_m) is used to determine whether the generated new child can be mutated. The operator selects two random points from the child and swaps their positions. The P_c and P_m are tuned before the case study.

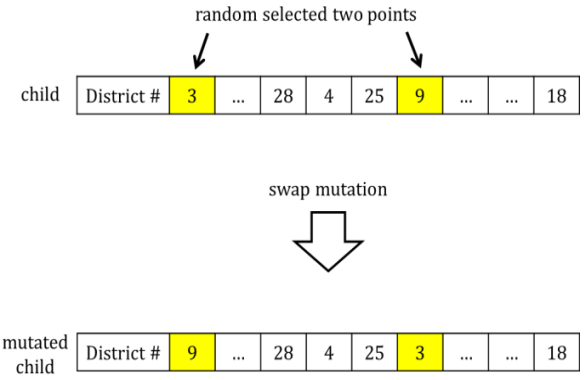


Figure 6. Swap mutation.

3.4. Environmental selection and stopping condition

The GA generates μ-e=λ number of new children and with an elitist strategy; the best e number of individuals of the previous population is added to the new generation. The μ stays constant through generations in the environmental selection. The GA stops when it reaches to g_{max} number of generations. The parameters are tuned before the case study.

4. The case study and the results

The border between Turkey and Syria is selected because the conflicts started in 2011 and not ended till 2021 in Syria. Millions of refugees migrated from Syria through Turkey. It is critical to observe the border to prevent the passage of terrorists hidden in the civilian crowds through the border. The borderline between

Turkey and Syria is 910 km as seen in Figure 7.

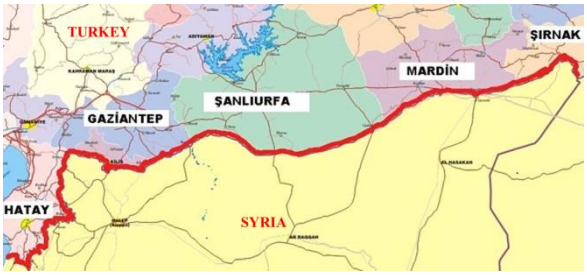


Figure 7. The borderline between Turkey and Syria (910km).

The borderline is not a straight line and there are indentations and protrusions. Therefore 152 grid (i.e. check) points and 4 base stations nearest through the borderline (Adana, Diyarbakır, Batman, and Muş) are selected conveniently to the selected UAV. The Euclidean distance between the grid points is five kilometers and the grid points are covering the overall Turkey-Syria borderline. The base stations are representing the military airports in those cities. The selected bases and grid points are presented in Figure 8.

Figure 8. The selected 152 grid points and 4 main bases (Adana, Diyarbakır, Batman, and Muş).

The Bayraktar UAV [48,49] is selected for the case study because it is already using in these missions in real life. The Bayraktar is a long-ranged (3000 kilometers) Turkish made tactical UAV that is convenient for that kind of missions. We assumed that the unit cost of one UAV is $4,000,000 and the unit cost of the 1-kilometer flight is $20 to be used in the model. The Bayraktar can get the desired images and videos from that visited grid points by flying at a proper altitude. A sample figure for the coverage of the zones by visiting the grid points (i.e. red signs are representing the grid points in the borderline) in the center of the grid zones via a Bayraktar UAV is presented in Figure 9.

The 152 grid points (can be seen in Figure 8) are located based on the observing capabilities of UAV as in Figure 9 and covering all borderline from beginning to end. The scenarios in the case study section are generated according to the real-life necessities. 12 scenarios are considered covering active base station(s) and the number of daily patrols can be made by UAVs. The number of daily patrols is affecting the flight range of UAVs as seen in Table 1.

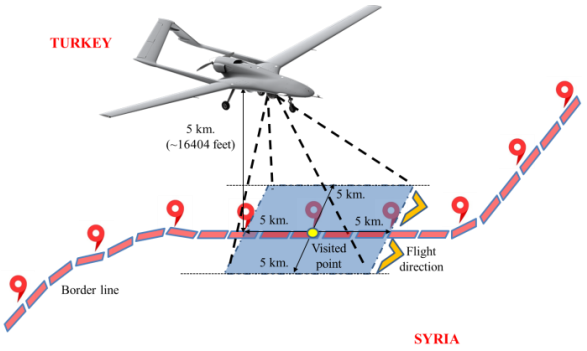


Figure 9. The usage of Bayraktar UAV for a border security mission.

Table 1. The scenarios.

Sce. #	Active main base(s)	# of daily patrols	Flight range (km.)
1	Adana	1	3000
2	Batman	1	3000
3	Diyarbakır	1	3000
4	Muş	1	3000
5	Adana	2	1500
6	Batman	2	1500
7	Diyarbakır	2	1500
8	Muş	2	1500
9	All bases	3	1000
10	All bases	4	750
11	All bases	5	600
12	All bases	6	500

In eight of twelve scenarios, it is considered that just one base station is active which means the UAVs can only be settled in those base stations. For the remaining four scenarios, all bases are thought of as active. Therefore, the UAVs can take off from one base station and can land at another base.

The parameters of the GA are tuned with experiments. 30 runs are done for each combination of the parameters by running with the candidate values in a selected scenario. The best-resulted values in 30 runs are used to determine the values for the parameters as seen in Table 2. The μ and g_{max} parameters are considered as pairs to make a fair comparison. P_c is assumed as 1.

The GA is coded in MATLAB, the MILP model is developed in ILOG and solved in CPLEX. The runs of GA and the MILP model are made on the same computer that has Intel Core I7-7700HQ CPU, 2.80 GHz, and 16 GB RAM specifications. The scenarios are solved 30 times with GA and GABM used the best solution of GA to determine the "m" value.

The summary results are presented in Table 3. According to the results with the one active based initial eight scenarios, for six solutions the GABM found the same optimal solutions with the GA. But for two

scenarios (i.e. scenario #5 and #8), the GABM improved the GA solutions and found the optimal solutions. Especially for scenario #8, the GABM solution is using much less UAV than GA, since Muş is the furthest base station to the borderline. However, for the last four more complex scenarios, the GABM outperformed and improved the GA results much more. The used UAVs and the flown distances in GA results are extremely decreased in GABM results. The flight distances are reduced by more than near 1000 kilometers in some scenarios and at least one UAV is saved. According to the used active bases that are listed in the last four results; mostly Adana, Diyarbakır, and Batman bases are preferred to take off and land by the UAVs. Since Muş is the furthest base station to the borderline, it is not used in optimal routes.

Since the range distance is the most effective constraint in the results, the details about the flight distances, remaining flight ranges, and flight routes of each UAV for the results of the GABM are provided in Table 4 with the details about the scenarios (i.e. the # of variables and the # of constraints) are also covered. In scenarios, the # of variables is changing for 1 to 4 UAVs as 23562, 46971, 48825, and 97497, respectively. The # of constraints is also ranging 23411 to 23876 for 1 to 4 UAVs. According to the detailed results of GABM, when the # of daily patrols increases and the flight range of UAVs decreases, the utilization based on flight distances of UAVs also increases as seen that the remaining flight ranges are smaller in the last scenarios (i.e. especially scenarios #10, #11, and #12). For scenarios with long flight ranges the Bayraktar UAV can accomplish missions with smaller utilization ratios based on flight distances.

The CPU times of the algorithms are presented in Table 5. The CPU times seem very reasonable for a UAVRP. The CPU times for GA are ranging between 71.89 to 122.17 seconds. The CPU times of GABM is varying from 2.92 to 46.39 seconds. The CPU times of the GA should be added to the GABM times in reality since the GABM is firstly using the GA results.

The optimal solution found by GABM for the 12th scenario is presented in Figure 10. The used active bases and the routes of used four UAVs can be seen in the figure. The first UAV is taking off from Adana, monitoring the west side of the border, and turning back to Adana. The second UAV is also taking off from Adana, monitoring the remaining west side of the border, and landing to Diyarbakır. The third UAV is departing from Diyarbakır and landing at Batman. The last UAV is taking off and landing to Batman with covering the east side of the border. As seen all grid points are visited by one of the four UAVs.

Table 2. The tuning results.

Parameters					Fitness values for 30 runs			
μ	g_{max}	e	P_m	k	Min.	Mean	Max.	Std.dev.
80	125	0.1	0.1	3	8019991.1	8022027.8	8025389.9	1091.3
80	125	0.1	0.1	5	8019097.6	8021529.4	8023435.9	896.5
80	125	0.1	0.2	3	8019434.1	8021719.4	8023313.1	808.1
80	125	0.1	0.2	5	8019113.8	8021858.7	8023700.7	1163.1
80	125	0.2	0.1	3	8016928.8	8020533.4	8023854.4	1998.4
80	125	0.2	0.1	5	8016608.8	8021417.1	8024069.0	1629.1
80	125	0.2	0.2	3	8019478.7	8021155.0	8023131.7	718.9
80	125	0.2	0.2	5	8016845.3	8021317.1	8025543.3	1024.2
40	250	0.1	0.1	3	8020456.5	8022085.3	8024880.2	1000.4
40	250	0.1	0.1	5	8019328.7	8021403.9	8023808.3	899.1
40	250	0.1	0.2	3	8019150.8	8023526.2	8027532.6	2387.7
40	250	0.1	0.2	5	8019166.4	8022191.1	8024364.9	1320.9
40	250	0.2	0.1	3	8019137.3	8021383.1	8023118.5	758.2
40	250	0.2	0.1	5	8018223.7	8021876.6	8025219.9	1169.1
40	250	0.2	0.2	3	8017936.9	8021321.2	8024272.5	1584.4
40	250	0.2	0.2	5	8017811.6	8021217.6	8023878.5	1442.1
20	500	0.1	0.1	3	8016797.7	8023057.7	8026972.6	2276.1
20	**500**	**0.1**	**0.1**	**5**	**8016518.3***	8022263.2	8025954.5	1872.1
20	500	0.1	0.2	3	8017019.3	8023629.8	8027532.6	3046.1
20	500	0.1	0.2	5	8017472.8	8023218.3	8026627.4	2299.4
20	500	0.2	0.1	3	8019458.8	8023270.7	8027532.6	2575.1
20	500	0.2	0.1	5	8017343.7	8021898.6	8024866.8	2096.1
20	500	0.2	0.2	3	8017324.2	8021759.5	8025381.9	2080.7
20	500	0.2	0.2	5	8017885.9	8021795.8	8026330.2	1916.3

* In the best solution according to the fitness value calculation 2 UAVs are flying 825.9 km.

Table 3. The results.

Sce. #	GA Best # of UAV(s)	GA Best Total flight dist. (km.)	GA Mean # of UAV(s)	GA Mean Total flight dist. (km.)	GA Worst # of UAV(s)	GA Worst Total flight dist. (km.)	GABM # of UAV(s)	GABM Flight dist. (km.)	Improvement (%) # of UAV(s)	Improvement (%) Total flight dist. (km.)
1	**1**	**1504.9**	**1**	**1504.9**	**1**	**1504.9**	**1**	**1504.9**	**-**	**-**
2	**1**	**1419.3**	**1**	**1419.3**	**1**	**1419.3**	**1**	**1419.3**	**-**	**-**
3	**1**	**1415.7**	**1**	**1415.7**	**1**	**1415.7**	**1**	**1415.7**	**-**	**-**
4	**1**	**1554.4**	**1**	**1554.4**	**1**	**1554.4**	**1**	**1554.4**	**-**	**-**
5	**2**	3618.8	**2**	3618.8	**2**	3618.8	**2**	**1718.4**	**-**	**52.51**
6	**1**	**1419.3**	**1**	**1419.3**	**1**	**1419.3**	**1**	**1419.3**	**-**	**-**
7	**1**	**1415.7**	**1**	**1415.7**	**1**	**1415.7**	**1**	**1415.7**	**-**	**-**
8	8	13474.8	8	13474.8	8	13474.8	**2**	**1887.5**	**400**	**85.99**
9	2*	1710.3*	2	1919.4	2	1986.6	**2****	**1204.5****	**-**	**29.57**
10	3***	2126.6***	3	2184.6	3	2234.4	**2*****	**1298.7*****	**33**	**38.93**
11	4***	2406.8***	4	2567.1	4	2718.8	**3*****	**1444.1*****	**25**	**39.99**
12	5***	2840.2***	5	2900.1	5	2955.5	**4*****	**1793.9*****	**20**	**36.83**

* The used bases through active bases are Adana and Diyarbakır
** The used bases through active bases are Adana and Batman
*** The used bases through active bases are Adana, Diyarbakır, and Batman

Table 4. The details about the GABM results and the scenarios.

Sce. #	# of variables	# of constraints	Flight range (km.)	GABM # of UAV(s)	GABM Singular flight dist. (km.)	GABM Singular remaining flight range (km.)	GABM Singular flight route
1	23562	23411	3000	1	1504.9	1495.1	1-…-152
2	23562	23411	3000	1	1419.3	1580.7	1-…-152
3	23562	23411	3000	1	1415.7	1584.3	1-…-152
4	23562	23411	3000	1	1554.4	1445.6	1-…-152
5	46971	23566	1500	2	390.9	1109.1	33-34-…-152
					1327.5	172.4	32-31-…-1
6	23562	23411	1500	1	1419.3	1580.7	1-…-152
7	23562	23411	1500	1	1415.7	1584.3	1-…-152
8	46971	23566	1500	2	1490.7	9.3	144-143-…-1
					396.8	1103.2	145-146-…-152
9	48825	23566	1000	2	820.8	179.2	121-120-…-1
					383.7	616.3	122-123-…-152
10	48825	23566	750	2	747.4	2.6	1-…-96-97
					551.3	198.7	152-…-99-98
11	73161	23721	600	3	599.5	0.5	112-111-…-36-35
					441.6	158.4	152-…-114-113
					403.0	197.0	34-33-…-1
12	97497	23876	500	4	499.2	0.8	120-121-…-152
					497.2	2.8	35-36-…-74-75
					403.0	97.0	76-77-…-118-119
					394.5	105.5	34-33-…-1

5. Conclusion

In this study, the UAVRP is adapted for border security missions. A new mathematical model is developed inspired by literature. The model is combined with a problem-specific GA to solve the UAVRP. The designed GABM is trying to solve the problem by initializing GA 30 times in the beginning. Then the model is using the "*m*" value of the best GA result as a constant. If the model solves the problem with the determined "*m*" value, then the algorithm decreases the "*m*=*m*-1" and rerun the model. If the model cannot find a solution with the determined "*m*" value, then GABM presents the previous solution as optimal. As far as known, GA and GABM algorithms are used for the first time in the literature in a UAVRP defined for border monitoring missions.

Table 5. The CPU times.

Sce. #	GA (s.)			GABM (s.)
	Best	Mean	Worst	
1	79.39	91.39	111.86	3.23
2	74.65	88.17	98.47	3.78
3	71.89	82.94	97.49	3.11
4	74.90	82.04	92.51	2.92
5	82.61	91.57	100.99	4.44
6	80.84	91.51	105.95	3.78
7	81.61	95.75	111.58	3.11
8	81.76	88.97	96.19	3.16
9	89.55	100.71	116.73	5.04
10	92.64	101.32	114.92	5.65
11	98.87	106.29	122.17	24.60
12	91.81	104.87	121.09	46.39

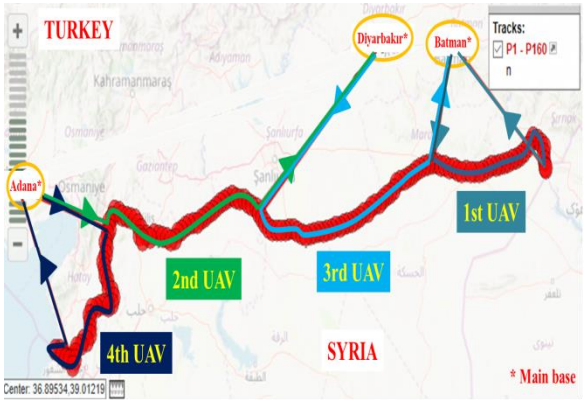


Figure 10. The optimal route for scenario #12 with 4 UAVs

The borderline between Turkey and Syria is selected for the case study. Bayraktar UAV is chosen considering the usage of it in real life. 12 realistic scenarios are generated covering four airbases and 152 grid points on the borderline. While considering scenarios, the usage of active bases and the number of daily patrols done by UAVs are changed. The scenarios are solved with GA and GABM. In half of the scenarios, GABM improved the GA results and decreased the used number of UAVs and the total travel distances. Especially for more complex scenarios, GABM developed the results much effectively in reasonable CPU times.

For future research, more realistic 3-D scenarios can be tried. Simulation models can be integrated with proposed hybrid algorithms to simulate the intruder detections in the border with a more dynamic perspective.

References

[1] Kaza, S., Wang, Y., & Chen, H. (2007). Enhancing border security: Mutual information analysis to identify suspect vehicles, *Decision Support Systems*, 43, 199-210.

[2] Sun, Z., Wang, P., Vuran, M.C., Al-Rodhaan, M. A., Al-Dhelaan, A.M., & Akyildiz, I.F. (2011). BorderSense: Border patrol through advanced wireless sensor networks, *Ad Hoc Networks*, 9, 468-477.

[3] Owen, A., Duckworth, G., & Worsley, J. (2012). OptaSense: Fibre optic distributed acoustic sensing for border monitoring, in *Proc. of the 2012 European Intelligence and Security Informatics Conference, 22-24 Aug., Odense, Denmark* [Online]. Available: IEEE Xplore, http://www.ieee.org. [Accessed: 16 September 2020].

[4] Hare, J., Gupta, S., & Wilson, J. (2015). Decentralized smart sensor scheduling for multiple target tracking for border *surveillance, in Proc. of the 2015 IEEE Int. Conf. on Robotics and Automation, ICRA 2015, 26-30 May, Seattle, Washington*, [Online]. Available: IEEE Xplore, http://www.ieee.org. [Accessed: 16 September 2020].

[5] Karabulut, E., Aras, N., & Altınel, İ.K. (2017). Optimal sensor deployment to increase the security of the maximal breach path in border surveillance, *European Journal of Operational Research*, 259, 19-36.

[6] Alkhathami, M., Alazzawi, L., & Elkateeb, A. (2017). Large scale border security systems modeling and simulation with OPNET, in *Proc. of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference, CCWC 2017, 09-11 Jan., Las Vegas, USA*.

[7] Arjun, D., Indukala, P.K., & Unnikrishna Menon, K.A. (2017). Border surveillance and intruder detection using wireless sensor networks: A brief survey, *in Proc. of the 2017 Int. Conf. on Communication and Signal Processing, ICCSP 2017, 06-08 April, Chennai, India*.

[8] Lessin, A.M., Lunday, B.J., & Hill, R.R. (2018). A bilevel exposure-oriented sensor location problem for border security, *Computers and Operations Research*, 98, 56-68.

[9] Matveev, A.S., Teimoori, H., & Savkin, A.V. (2011). A method for guidance and control of an autonomous vehicle in problems of border patrolling and obstacle avoidance, *Automatica*, 47, 515-524.

[10] Tanas, M., Holubowicz, W., Adamczyk, A., & Taberski, G. (2011). The TALOS Project. EU wide robotic border guard system, in *Proc. of the 2011 16th Int. Conf. on Methods & Models in Automation & Robotics, 22-25 Aug., Miedzyzdroje, Poland*.

[11] Tanas, M., Taberski, G., Hołubowicz, W., Samp, K., Sprońska, A., Główka, J., & Maciaś, M. (2012). The TALOS project – autonomous robotic patrol vehicles, in *Proc. of the 2012 European Intelligence and Security Informatics Conference, 22-24 Aug., Odense, Denmark.*

[12] Girard, A.R., Howell, A.S., & Hedrick, J.K. (2004). Border patrol and surveillance missions using multiple unmanned air vehicles, *in Proc. of the 43rd IEEE Conference on Decision and Control, 2004, 14-17 Dec., Atlantis, Bahamas*.

[13] Blazakis, J. (2006). Border security and unmanned aerial vehicles, *Connections,* 5(2), 154-159.

[14] Matveev, A.S. Teimoori, H., & Savkin, A.V. (2010). A method for navigation of an autonomous vehicle for border patrol, *in Proc. of the 2010 American Control Conference, 30 June-02 July, Marriott Waterfront, Baltimore, USA.*

[15] Haddal, C.C., & Gertler, J. (2010). Homeland security: Unmanned aerial vehicles and border surveillance, *Congressional Research Service Report for Congress, RS21698,* USA, [Online]. Available: https://nsarchive2.gwu.edu/NSAEBB/NSAEBB527-Using-overhead-imagery-to-track-domestic-US-targets/documents/EBB-Doc24.pdf, [Accessed: 16 September 2020].

[16] Ortiz-Rivera, E.I., Estela, A., Romero, C., & Valentin, J.A. (2012). The use of UAVS in USA's security by an engineering education approach, *in Proc. of the 2012 IEEE Conference on Technologies for Homeland Security (HST), 13-15 Nov., Waltham, USA.*

[17] Moss, V., Jones, D., & Nwaneri, S. (2012). Analysis of homeland security and economic survey using special missions unmanned aerial vehicle utilities, *in Proc. of the 2012 IEEE International Geoscience and Remote Sensing Symposium, 22-27 July, Munich, Germany*.

[18] Office of Inspector General (2014). U.S. customs and border protection's unmanned aircraft system program does not achieve intended results or recognize all costs of operations", *Department of Homeland Security Report, OIG-15-17,* USA, [Online]. Available: https://www.oig.dhs.gov/assets/Mgmt/2015/OIG_15-17_Dec14.pdf, [Accessed: 16 September 2020].

[19] Bein, D., Bein, W., Karki, A., & Madan, B.B. (2015). Optimizing border patrol operations using unmanned aerial vehicles, *in Proc. of the 2015 12th International Conference on Information Technology - New Generations, 13-15 April, Las Vegas, USA.*

[20] Półłka, M., Ptak, S., & Kuziora, L. (2017). The use of UAV's for search and rescue operations, *Procedia Engineering,* 192, 748-752.

[21] Cică, C., & Filipoaia, L. (2016). Border surveillance optimization using a multi-objective mathematical model, in *Proc. of the 2016 IEEE Int. Conf. on Electronics, Computers and Artificial Intelligence, ECAI 2016, 30 June-02 July, Ploiesti, Romania*.

[22] Çelik, G., & Sabuncuoğlu, İ. (2007). Simulation modelling and analysis of a border security system, *European Journal of Operational Research,* 180, 1394-1410.

[23] Jenkins, J.L., Marquardson, J., Proudfoot, J.G., Valacich, J.S., Golob, E., & Nunamaker, Jr., J.F. (2013). The Checkpoint Simulation: A tool for informing border patrol checkpoint design and resource allocation, in *Proc. of the 2013 European Intelligence and Security Informatics Conference, 12-14 Aug., Uppsala, Sweden*.

[24] Muaafa, M., & Ramirez-Marquez, J.E. (2017). Bi-objective evolutionary approach to the design of patrolling schemes for improved border security", *Computers & Industrial Engineering,* 107, 74-84.

[25] Ackleson, J. (2003). Directions in border security research, *The Social Science Journal,* 40, 573-581.

[26] Gravelle, T.B. (2018). Politics, time, space, and attitudes toward US–Mexico border security, *Political Geography,* 65, 107-116.

[27] Fisher, D.X.O. (2018). Situating border control: Unpacking Spain's SIVE border surveillance assemblage, *Political Geography,* 65, 67-76.

[28] Dalamagkidis, K. Valavanis, K.P., & Piegl, L.A. (2008). On unmanned aircraft systems issues, challenges and operational restrictions preventing integration into the National Airspace System, *Progress in Aerospace Sciences,* 44, 503-519.

[29] Gupte, S., Mohandas, P.I.T., & Conrad, J.M. (2012). A survey of quadrotor unmanned aerial vehicles, *in Proc. of the 2012 IEEE Southeastcon, 15-18 March, Orlando, USA.*

[30] Yu, X., & Zhang, Y. (2015). Sense and avoid technologies with applications to unmanned aircraft systems: Review and prospects, *Progress in Aerospace Sciences,* 74, 152-166.

[31] Mcfadyen, A., & Mejias, L. (2016). A survey of autonomous vision-based see and avoid for unmanned aircraft systems, *Progress in Aerospace Sciences,* 80, 01-17.

[32] Eksioglu, B., Vural, A.V., & Reisman, A. (2009). The vehicle routing problem: A taxonomic review, *Computers & Industrial Engineering,* 57, 1472-1483.

[33] Braekers, K., Ramaekers, K., & Nieuwenhuyse, I.V. (2016). The vehicle routing problem: State of the art classification and review, *Computers & Industrial Engineering,* 99, 300-313.

[34] Gansterer, M., & Hartl, R.F. (2018). Collaborative vehicle routing: A survey, *European Journal of Operational Research,* 268, 1-12.

[35] Tlili, T., Faiz, S., & Krichen, S. (2014). A hybrid metaheuristic for the distance-constrained capacitated vehicle routing problem, *Procedia - Social and Behavioral Sciences,* 109, 779-783.

[36] Letchford, A.N., & Salazar-González, J.-J. (2019). The capacitated vehicle routing problem: Stronger bounds in pseudo-polynomial time, *European Journal of Operational Research,* 272, 24-31.

[37] Montoya-Torres, J.R., Franco, J.L., Isaza, S.N., Jiménez, H.F., & Herazo-Padilla, N. (2015). A literature review on the vehicle routing problem with multiple depots, *Computers & Industrial*

Engineering, 79, 115-129.

[38] Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures", *Omega*, 34, 209-219.

[39] Kaempfer, Y., & Wolf, L. (2018). Learning the multiple traveling salesmen problem with permutation invariant pooling networks, *CORR*, abs/1803.09621, 1-17.

[40] Coutinho, W.P., Battarra, M., & Fliege, J. (2018). The unmanned aerial vehicle routing and trajectory optimisation problem, a taxonomic review, *Computers & Industrial Engineering*, 120, 116-128.

[41] Pandey, P., Shukla, A., & Tiwari, R. (2017). Aerial path planning using meta-heuristics: A survey, in *Proc. of the 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), 22-24 Feb., Coimbatore, India*.

[42] Zhao, Y., Zheng, Z., & Liu, Y. (2018). Survey on computational-intelligence-based UAV path planning, *Knowledge-Based Systems*, 158, 54-64.

[43] Seyis, A., Karacin, Y., & Ozkan, O. (2016). Optimal path planning with minimum number of UAVs by using genetic algorithm", *in Proc. of the 28th European Conference on Operational Research (EURO 2016), 03-06 July, Poznan, Poland.*

[44] Kaya, M., & Ozkan, O. (2018). Sınır koruma görevi için insansız hava araçlarının rotalanması probleminin genetik algoritma ile eniyilenmesi, *in Proc. of the 38. Ulusal Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi (YAEM 2018), 26-29 June, Eskişehir, Turkey.*

[45] Ozkan, O. (2018). İnsansız hava araçları ile Türkiye'deki orman yangınlarının tespiti probleminin tavlama benzetimi ile eniyilenmesi, *in Proc. of the 38. Ulusal Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi (YAEM 2018), 26-29 June, Eskişehir, Turkey.*

[46] Yang, L., Qi, J., Xiao, J., & Yong, X. (2014). A literature review of UAV 3D path planning, *in Proc. of the 2014 11th World Congress on Intelligent Control and Automation, 29 June-04 July, Shenyang, China*.

[47] Khan, M.A., Safi, A., Qureshi, I.M., & Khan, I.U. (2017). Flying Ad-Hoc Networks (FANETs): A review of communication architectures, and routing protocols, *in Proc. of the 2017 First International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT), 15-16 Nov., Karachi, Pakistan*.

[48] T.C. Cumhurbaşkanlığı, Türk Savunma Sanayii Başkanlığı (2021). *SSB – Türk Savunma Sanayii Ürün Kataloğu [online]*. Available from: SSB - TÜRK SAVUNMA SANAYİİ ÜRÜN KATALOĞU, [Accessed: 16 September 2020].

[49] Baykar Savunma (2021). *Bayraktar TB-2 [online]*. Available from: BAYKAR İnsansız Hava Aracı Sistemleri (baykarsavunma.com), [Accessed: 16 September 2020].

Muhammed Kaya *received his B.Sc. degree in Industrial Engineering from Turkish Air Force Academy, National Defence University, Istanbul, in 2018. He is a First Lieutenant in Turkish Air Force.*

https://orcid.org/0000-0001-9410-7367

Omer Ozkan *received his B.Sc. degree in Industrial Engineering from Turkish Air Force Academy, Istanbul, in 2002. He received his M.Sc. degree in Industrial Engineering from Dokuz Eylul University, Izmir, in 2010 and the Ph.D. degree in Industrial Engineering from National Defence University, Istanbul, in 2016. He is currently an Assistant Professor of Operations Research in Industrial Engineering Department, Turkish Air Force Academy, National Defence University. His research interests include metaheuristics, optimization, and network analysis.*

https://orcid.org/0000-0002-3839-2754

An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.11, No.2, pp.139-151 (2021)
http://doi.org/10.11121/ijocta.01.2021.001001

RESEARCH ARTICLE

Conic reformulations for Kullback-Leibler divergence constrained distributionally robust optimization and applications

Burak Kocuk*

Industrial Engineering Program, Faculty of Engineering and Natural Sciences, Sabancı University, Istanbul, Turkey
burak.kocuk@sabanciuniv.edu

ARTICLE INFO

Article History:
Received 12 July 2020
Accepted 11 January 2021
Available 19 April 2021

Keywords:
Distributionally robust optimization
Stochastic programming
Conic programming
Newsvendor problem
Uncapacitated facility location problem

AMS Classification 2010:
90C15; 90C25; 90C90

ABSTRACT

In this paper, we consider a Kullback-Leibler divergence constrained distributionally robust optimization model. This model considers an ambiguity set that consists of all distributions whose Kullback-Leibler divergence to an empirical distribution is bounded. Utilizing the fact that this divergence measure has an exponential cone representation, we obtain the robust counterpart of the Kullback-Leibler divergence constrained distributionally robust optimization problem as a dual exponential cone constrained program under mild assumptions on the underlying optimization problem. The resulting conic reformulation of the original optimization problem can be directly solved by a commercial conic programming solver. We specialize our generic formulation to two classical optimization problems, namely, the Newsvendor Problem and the Uncapacitated Facility Location Problem. Our computational study in an out-of-sample analysis shows that the solutions obtained via the distributionally robust optimization approach yield significantly better performance in terms of the dispersion of the cost realizations while the central tendency deteriorates only slightly compared to the solutions obtained by stochastic programming.

1. Introduction

Decision making under uncertainty is one of the most challenging tasks in operations research. Two paradigms are predominantly used in the literature to address uncertainty: stochastic programming and robust optimization. In the classical stochastic programming [1, 2], a predefined set of scenarios (or samples) are determined, either taken directly from observed data or after fitting an appropriate distribution. Then, the objective function is replaced with an expectation taken with respect to the random elements, and constraints are copied for each scenario. In addition to the assumption about knowing the underlying distribution, this basic stochastic programming approach has some limitations: Firstly, the size of the resulting deterministic equivalent formulation grows larger with the size of the scenarios. Secondly, an expectation may not be an appropriate performance measure for risk-averse decision makers. Thirdly, satisfying constraints for each scenario might be too restrictive. The respective remedies for these shortcomings are proposed such as sample average approximation to limit the problem size, risk-averse objective function for a more appropriate performance measure and chance constraints to allow constraint satisfaction with high probability. However, the implicit assumption of stochastic programming remains, which is the need to *assume* a distribution by analyzing the data or *fitting* one. Unfortunately, this step may not be performed satisfactorily in all cases.

*Corresponding author

In robust optimization [3–5], a predefined uncertainty set, which includes all possible values of the uncertain elements, is used. Then, the optimization is performed with the aim of optimizing with respect to the worst possible realization from the uncertainty set. There are two main advantages of using robust optimization. Firstly, the decision maker does not need to make any assumptions about the distribution of the uncertain elements in the problem as opposed to the stochastic programming approach. Secondly, the deterministic equivalent (or so-called the *robust counterpart*) formulation of the robust optimization problem typically has the same computational complexity as the deterministic version of the problem under reasonable assumptions on the uncertainty sets. On the other hand, the main disadvantage of the robust optimization approach is that depending on the construction of the uncertainty set, it might lead to overly conservative solutions, which might have poor performance in central tendency such as expectation.

Distributionally robust optimization (DRO) is a relatively new paradigm that aims to combine stochastic programming and robust optimization approaches. The main modeling assumption of DRO is that some *partial* information about the distribution governing the underlying uncertainty is available, and the optimization is performed with respect to the worst distribution from an *ambiguity set*, which contains all distributions consistent with this partial information. There are mainly two streams in the DRO literature based on how the ambiguity set is defined: moment-based and distance-based.

In moment-based DRO, ambiguity sets are defined as the set of distributions whose first few moments are assumed to be known or constrained to lie in certain subsets. If certain structural properties hold for the ambiguity sets such as convexity (or conic representability), then tractable convex (or conic reformulations) can be obtained [6–8]. In distance-based DRO, ambiguity sets are defined as the set of distributions whose distance (or divergence) from a reference distribution is constrained. For Wasserstein distance [9–12] and $\phi-$divergence [13–17] constrained DRO, tractable convex reformulations have been proposed. Recently, chance constrained DRO problems have also drawn attention [18–21].

As summarized above, in many cases, tractable convex robust counterparts or reformulations can be obtained for robust and distributionally robust (DR) optimization problems. However, an even more special structure such as conic representability can be preferred whenever available. Especially, if the robust counterpart can be expressed as a conic program for which the underlying cone admits a self-concordant barrier, then efficient polynomial-time interior point methods can be applied directly [22]. This desired property holds for linear programs, second-order cone programs and semidefinite programs, which appear extensively in both robust and DR optimization literature. We note that the efficiency of the conic programming solvers specialized in these three problem classes has improved considerably.

There is some recent interest in conic programs for which the underlying cone is not self-dual, such as exponential cone. There are two main reasons: i) exponential cone has extensive expressive power that is useful to model optimization problems involving the exponential and logarithm functions (see e.g. [23]), and ii) a practical implementation of a primal-dual interior point method is developed [24] although its polynomial-time complexity has not been proven yet. Our paper will exploit both the expressive power of the exponential cone and the practical implementation that can be used to solve the resulting optimization problem, as detailed below.

The Kullback-Leibler (KL) divergence [25] is a popular divergence measure in information theory that can be used to quantify the divergence of one distribution from another (see Definition 8) and we prove that it is exponential cone representable (see Definitions 3 and 6, and Proposition 2). Although the robust counterpart of KL divergence constrained DRO is proven to be a tractable convex program [26], to the best of our knowledge, its exponential cone representability has not been exploited in the literature before. Also, its practical performance against stochastic programming has not been analyzed in detail except for a limited number of applications from power systems [27, 28].

In this paper, we consider KL divergence constrained DRO problems and propose their dual exponential cone constrained reformulation under the mild assumption of conic representability. This allows us to solve the corresponding robust counterpart using a conic programming solver such as MOSEK [29]. We also present how the generic formulation can be specialized for two classical problems: Newsvendor and Uncapacitated Facility Location. Although the DRO methodology has been applied to variations of these problems [30–35], to the best of our knowledge, their KL divergence constrained versions have not been studied in detail. Our computational results suggest that solutions obtained via

a DR approach give slightly higher cost realizations when central tendencies such as mean and median are considered compared to solutions obtained via stochastic programming in an out-of-sample analysis. However, the dispersion (measured by the standard deviation and range of the cost realizations) and the risk (measured by the average of worst cost realizations and the third quartile) metrics improve significantly with solutions obtained via a DR approach.

Our main contribution in this paper is to exploit the exponential cone representation of the KL-divergence to solve DRO problems with ambiguity sets defined using this measure[1]. Unlike the previous literature, which treats such problems as "general convex programs" (see, for instance, [13]), we utilize the general-purpose conic programming solver MOSEK in our computations. The conic representability property can be quite useful in practice since one can then include other (mixed-integer) conic representable sets and functions to the underlying optimization problem and, hence, model a wide variety of real-life situations.

Although we carry out the computational experiments for two classical problems, we remark that our approach is rather general and it can be adapted to various settings. To give a few examples, one can use our framework in applications from portfolio optimization [7,10], image processing [9, 16], asset pricing [13], multidimensional knapsack problem [12,20] and logistics [35,36].

The rest of the paper is organized as follows: In Section 2, we review basic concepts from convex analysis and probability theory which serve as the basis of our main result about conic reformulation of KL divergence constrained DRO problems in Section 3. Then, we analyze two applications, namely, the Newsvendor Problem in Section 4 and the Uncapacitated Facility Location Problem in Section 5, and present the results of our computational study. Finally, we conclude our paper in Section 6.

2. Preliminaries

Before stating our main result in Section 3, we will first review some important concepts from convex analysis in Section 2.1 and probability theory in Section 2.2.

2.1. Convex analysis

For a set $X \subseteq \mathbb{R}^m$, we denote its interior as $\text{int}(X)$, its relative interior as $\text{ri}(X)$ and its closure as $\text{cl}(S)$. We use the shorthand notation $[n]$ for the set $\{1, \ldots, n\}$.

We will first review some basic concepts from convex analysis related to cones.

Definition 1 (Regular cone)**.** *A cone $K \subseteq \mathbb{R}^m$ is called regular if it is closed, convex, pointed and full-dimensional.*

Examples of regular cones include the nonnegative orthant, Lorentz (or second-order) cone and the cone of positive semidefinite matrices. We will refer to these cones as *canonical cones* in this paper.

Definition 2 (Dual cone)**.** *The dual cone to a cone $K \subseteq \mathbb{R}^m$ is defined as $K_* = \{y \in \mathbb{R}^m : x^T y \geq 0,\ \forall x \in K\}$.*

It is well-known that the dual cone to a regular cone is also regular. In addition, the three canonical cones mentioned above are self-dual.

We will now define the exponential cone, which is the key ingredient of this paper.

Definition 3 (Exponential cone)**.** *The exponential cone, denoted as K_{exp}, is defined as*

$$K_{\text{exp}} = \text{cl}\big(\{x \in \mathbb{R}^3 : x_1 \geq x_2 e^{x_3/x_2},\ x_2 > 0\}\big).$$

As opposed to the three canonical cones mentioned above, the exponential cone is not self-dual although it is a regular cone.

Proposition 1 (See e.g. [23])**.** *The dual cone to the exponential cone (or simply the dual exponential cone) is given as*

$$(K_{\text{exp}})_* = \text{cl}\big(\{s \in \mathbb{R}^3 : s_1 \geq -s_3 e^{(s_2 - s_3)/s_3},\ s_3 < 0\}\big).$$

The following definitions are instrumental in the description of conic programming problems:

Definition 4 (Conic inequality)**.** *A conic inequality with respect to a regular cone K is defined as $x \succeq_K y$, meaning that $x - y \in K$. We will denote the relation $x \in \text{int}(K)$ alternatively as $x \succ_K 0$.*

Definition 5 (Conic representability of a set)**.** *A set $X \subseteq \mathbb{R}^n$ is called conic representable if it can be expressed as*

$$X = \{x \in \mathbb{R}^n : \exists y \in \mathbb{R}^k : Ax + By \succeq_K b\},$$

for some appropriately chosen regular cone K.

[1]We also note that two other $\phi-$divergence measures called Burg entropy and $J-$divergence have the same representation property.

If any of the variables in this representation is integer, then X is called a mixed-integer conic representable set.

Definition 6 (Conic representability of a function)**.** *A function is called conic representable if its epigraph is a conic representable set.*

In this paper, when we say that "a set or function is conic representable", we will implicitly assume that the cone used in its representation is either one of the three canonical cones or the (dual) exponential cone.

2.2. Probability theory

Definition 7 (Probability simplex)**.** *The probability simplex in dimension S is denoted as*

$$\Delta^S := \big\{p \in \mathbb{R}^S_+ : \sum_{s=1}^S p_s = 1\big\}.$$

The following function can be used to measure the "divergence" of one distribution from another.

Definition 8 (KL Divergence)**.** *For two discrete probability distributions $p \in \Delta^S$ and $q \in \text{ri}(\Delta^S)$, the KL divergence from p to q is defined as*

$$D_{KL}(p||q) := \sum_{s=1}^S p_s \log(p_s/q_s).$$

We note that the KL Divergence does not define a distance metric between two probability distributions since it is not symmetric. However, it has the following useful property.

Proposition 2. *Let $p \in \Delta^S$ and $q \in \text{ri}(\Delta^S)$. Then, the function $D_{KL}(p||q)$ is exponential cone representable.*

Proof. Due to Definition 6, it suffices to show that the epigraph of the function $D_{KL}(p||q)$ is an exponential cone representable set. Since the set $\{(x,y,t) \in \mathbb{R}^3 : t \geq x\log(x/y)\}$ has the exponential cone representation $(y,x,-t) \in K_{\text{exp}}$ [37], we obtain an exponential cone representation for the function $D_{KL}(p||q)$ as follows:

$$\begin{aligned}&\{(p,q,\epsilon) \in \Delta^S \times \text{ri}(\Delta^S) \times \mathbb{R} : D_{KL}(p||q) \leq \epsilon\}\\ =&\big\{(p,q,\epsilon) : \exists \delta \in \mathbb{R}^S : \sum_{s=1}^S \delta_s \leq \epsilon,\\ &\qquad (q_s, p_s, -\delta_s) \in K_{\text{exp}},\ s \in [S]\big\}.\end{aligned}$$

□

The following proposition gives an upper bound on the KL-divergence of a given distribution from any other distribution.

Proposition 3. *Let $q \in \text{ri}(\Delta^S)$. Then,*

$$\bar{\epsilon}(q) := \sup_{p \in \Delta^S}\{D_{KL}(p||q)\} = \log(1/\min_{s\in[S]}\{q_s\}).$$

Proof. Notice that the objective function of the optimization problem $\sup_{p\in\Delta^S}\{D_{KL}(p||q)\}$ is convex and its feasible region is a polytope. Therefore, there exists an optimal solution which is an extreme point of Δ^S. Observe that the extreme points of Δ^S are the unit vectors in $\mathbb{R}^S$, denoted by $\tilde{e}^s$ for $s \in [S]$ (note that $\tilde{e}^s_{s'} = 1$ for $s = s'$, and $\tilde{e}^s_{s'} = 0$ otherwise).
Let us now compute $D_{KL}(\tilde{e}^s||q)$ for some $s \in [S]$. In fact, we have

$$\begin{aligned}D_{KL}(\tilde{e}^s||q) &= \sum_{s'=1}^S \tilde{e}^s_{s'}\log(\tilde{e}^s_{s'}/q_s)\\ &= \tilde{e}^s_s\log(\tilde{e}^s_s/q_s) + \sum_{\substack{s'=1\\ s'\neq s}}^S \tilde{e}^s_{s'}\log(\tilde{e}^s_{s'}/q_s)\\ &= \log(1/q_s).\end{aligned}$$

Here, we use the fact that $\lim_{x\to 0^+} x\log(x/y) = 0$ for $y > 0$ in the last equality (recall that $q \in \text{ri}(\Delta^S)$, which implies that $q_s > 0$, $s \in [S]$).
Finally, we have

$$\begin{aligned}\bar{\epsilon}(q) &= \sup_{p\in\Delta^S}\{D_{KL}(p||q)\}\\ &= \max_{s\in[S]}\{D_{KL}(\tilde{e}^s||q)\}\\ &= \max_{s\in[S]}\{\log(1/q_s)\}\\ &= \log(1/\min_{s\in[S]}\{q_s\}).\end{aligned}$$

□

Proposition 3 is useful to quantify the ambiguity sets in KL divergence constrained DRO problems as we will see later.

3. Main results

In this section, we present our main result about the reformulation of a KL divergence constrained DRO problem as a conic program under mild conditions.

3.1. Generic problem formulation

We first give the generic problem setting considered in this paper. Suppose that there are m random variables $\xi^i \in \mathbb{R}$, $i \in [m]$, each with a discrete distribution $q^i \in \text{ri}(\Delta^{S_i})$ estimated from the historical data as

$$\Pr(\xi^i = d^i_s) = q^i_s \quad s \in [S_i],$$

where $\{d^i_s : s \in [S_i]\}$ is the set of observed realizations of ξ^i, $i \in [m]$. Under this probabilistic

setting, we define the ambiguity set

$$\mathcal{P}^i(q^i, \epsilon^i) := \{p^i \in \Delta^{S_i} : \ D_{KL}(p^i||q^i) \leq \epsilon^i\},$$

for $i \in [m]$, where $\epsilon^i \in \mathbb{R}_+$ controls the divergence from the historical data (or robustness level).

Then, we consider the following KL divergence constrained DRO problem,

$$\min_{y \in \mathcal{Y}} \Big\{h(y) + \sum_{i=1}^{m} \max_{p^i \in \mathcal{P}^i(q^i, \epsilon^i)} \mathbb{E}_{\xi^i}[H^i(y, \xi^i)]\Big\}, \quad (1)$$

where each expectation is taken with respect to an ambiguous distribution $p^i \in \mathcal{P}^i(q^i, \epsilon^i)$. In problem (1), h is a real-valued function defined on $\mathbb{R}^n$; H^i is a real-valued function defined on $\mathbb{R}^n \times \mathbb{R}$; and $\mathcal{Y}$ is a subset of $\mathbb{R}^n$. Observe that given y decisions, the inner maximization problem is decomposable over the random elements ξ^i, $i \in [m]$. Although we are assuming discrete probability distributions in this paper, our framework can be used to solve problems involving continuous distributions via a finite support approximation.

3.2. Robust counterpart and conic reformulation

We will now obtain the robust counterpart [4] of problem (1) utilizing Conic Duality under mild conditions.

Theorem 1. *Consider the KL divergence constrained DRO problem* (1) *as described in Section 3.1, and assume that $\epsilon^i > 0$, $i \in [m]$. Then, the robust counterpart is given as follows:*

$$\min\ h(y) + \sum_{i=1}^{m} \left[\alpha^i + \epsilon^i \beta^i + \sum_{s=1}^{S_i} q_s^i u_s^i\right] \quad (2a)$$

$$s.t.\ \alpha^i - v_s^i \geq H^i(y, d_s^i) \quad i \in [m]; s \in [S_i] \quad (2b)$$

$$\beta^i + w_s^i = 0 \quad i \in [m]; s \in [S_i] \quad (2c)$$

$$\alpha^i \in \mathbb{R}, \beta^i \in \mathbb{R}_+ \quad i \in [m] \quad (2d)$$

$$(u_s^i, v_s^i, w_s^i) \in (K_{\exp})_* \ i \in [m]; s \in [S_i] \quad (2e)$$

$$y \in \mathcal{Y}. \quad (2f)$$

Proof. We will start the proof by analyzing the inner maximization problems. Given a vector $y \in \mathcal{Y}$, let us write the i-th inner maximization problem explicitly as the following exponential cone constrained program:

$$\max\ \sum_{s=1}^{S_i} H^i(y, d_s^i) p_s^i \quad (3a)$$

$$\text{s.t.}\ \sum_{s=1}^{S_i} p_s^i = 1 \quad (3b)$$

$$\sum_{s=1}^{S_i} \delta_s^i \leq \epsilon^i \quad (3c)$$

$$\begin{bmatrix} 0 & 0 \\ -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_s^i \\ \delta_s^i \end{bmatrix} \preceq_{K_{\exp}} \begin{bmatrix} q_s^i \\ 0 \\ 0 \end{bmatrix} \quad s \in [S_i] \quad (3d)$$

$$p_s^i \in \mathbb{R}_+,\ \delta_s^i \in \mathbb{R} \quad s \in [S_i]. \quad (3e)$$

Here, constraints (3b)-(3e) model the relation $p^i \in \mathcal{P}^i(q^i, \epsilon^i)$, as stated in Proposition 2.

Recall that $\epsilon^i > 0$ for each $i \in [m]$. Then, each inner maximization problem (3) satisfies essential strict feasibility [38] (e.g. consider $p_s^i = q_s^i$ and $\delta_i^s = \epsilon^i/|S_i|$ for $s \in [S_i]$), and its optimal value is bounded above (e.g. by $\max_{s \in [S_i]} H^i(y, d_s^i)$). Therefore, strong duality holds between problem (3) and its conic dual given as follows:

$$\min\ \alpha^i + \epsilon^i \beta^i + \sum_{s=1}^{S_i} q_s^i u_s^i \quad (4a)$$

$$\text{s.t.}\ \alpha^i - v_s^i \geq H^i(y, d_s^i) \quad s \in [S_i] \quad (4b)$$

$$\beta^i + w_s^i = 0 \quad s \in [S_i] \quad (4c)$$

$$\alpha^i \in \mathbb{R}, \beta^i \in \mathbb{R}_+ \quad (4d)$$

$$(u_s^i, v_s^i, w_s^i) \in (K_{\exp})_* \quad s \in [S_i]. \quad (4e)$$

Here, α^i, β^i and (u_s^i, v_s^i, w_s^i) are the dual variables associated with the primal constraints (3b), (3c) and (3d), respectively. Notice that problem (4) is a dual exponential cone constrained program.

As the final step in the proof, we write the dual of each inner maximization problem and obtain the robust counterpart of problem (1) as problem (2). □

We will now discuss the consequences of Theorem 1 under additional structural properties such as convexity and conic representability.

Corollary 1. *Consider the KL divergence constrained DRO problem* (1) *as described in Theorem 1. In addition, let us assume that $\mathcal{Y}$ is a convex set, $h(y)$ and $H^i(y, \xi^i)$ are convex functions in y, $i \in [m]$. Then, the robust counterpart* (2) *is a convex program.*

Corollary 2. *Consider the KL divergence constrained DRO problem* (1) *as described in Theorem 1. In addition, let us assume that $\mathcal{Y}$ is a (mixed-integer) conic representable set, $h(y)$ and $H^i(y, \xi^i)$ are conic representable functions, $i \in [m]$. Then, the robust counterpart* (2) *is a dual exponential cone constrained (mixed-integer) program.*

As an application of Corollary 2, we will consider the Newsvendor Problem in Section 4 and the Uncapacitated Facility Location Problem in Section 5. The common characteristic of these two problems is that the set $\mathcal{Y}$ is a mixed-integer linear set, the function h is a linear function and the

functions H^i are the maxima of linear functions (hence, they are polyhedrally representable).

3.3. Extension to joint discrete probability distributions

We will now extend our analysis to the case of joint discrete probability distributions with a finite support. Suppose that we have a vector of random variables $\Xi \in \mathbb{R}^m$ with a joint probability distribution $q \in \text{ri}(\Delta^S)$ estimated from the historical data as

$$\Pr(\Xi = \tilde{D}_s) = q_s \quad s \in [S],$$

where $\{\tilde{D}_s : s \in [S]\} \subseteq \mathbb{R}^m$ is the set of observed realizations of Ξ.

Consider the ambiguity set

$$\mathcal{P}(q, \epsilon) := \{p \in \Delta^S : \ D_{KL}(p||q) \leq \epsilon\},$$

where $\epsilon \in \mathbb{R}_+$, and the following KL divergence constrained DRO problem:

$$\min_{y \in \mathcal{Y}} \big\{h(y) + \max_{p \in \mathcal{P}(q,\epsilon)} \mathbb{E}_\Xi[H(y, \Xi)]\big\}. \tag{5}$$

Here, h is a real-valued function defined on $\mathbb{R}^n$; H is a real-valued function defined on $\mathbb{R}^n \times \mathbb{R}^m$; and $\mathcal{Y}$ is a subset of $\mathbb{R}^n$. Then, we have the following result:

Theorem 2. *Consider the KL divergence constrained DRO problem* (5) *as described above, and assume that $\epsilon > 0$. Then, the robust counterpart is given as follows:*

$$\begin{aligned}
\min\ & h(y) + \left[\alpha + \epsilon\beta + \sum_{s=1}^{S} q_s u_s\right] \\
s.t.\ & \alpha - v_s \geq H(y, \tilde{D}_s) && s \in [S] \\
& \beta + w_s = 0 && s \in [S] \\
& \alpha \in \mathbb{R}, \beta \in \mathbb{R}_+ \\
& (u_s, v_s, w_s) \in (K_{\text{exp}})_* && s \in [S] \\
& y \in \mathcal{Y}.
\end{aligned}$$

We will omit the proof of Theorem 2 due to its similarity to the proof of Theorem 1. We remark that results similar to Corollaries 1 and 2 can be also obtained in this case under the assumptions of convexity and conic representability, respectively.

4. Application to the Newsvendor problem

In this section, we analyze a toy example, the KL divergence constrained DR version of the single-period, single-product Newsvendor Problem. In this case, since there is only one random variable ξ (that is, $m = 1$), representing the unknown demand, we will omit the superscript i for convenience.

4.1. Problem formulation

Consider the generic formulation (1) with the following specifications: We let $y \in \mathcal{Y} := \mathbb{Z}_+$ be the order quantity, and consider functions

$$h(y) := cy,$$

where c is the variable order cost, and

$$H(y, \xi) := c_b \max\{\xi - y, 0\} + c_h \max\{y - \xi, 0\},$$

where c_b is the back-order penalty for unsatisfied demand and c_h is the inventory cost. Notice that $H(y, \xi)$ is a piecewise linear convex function in y and can be rewritten as

$$H(y, \xi) = \max\{-c_b y + c_b \xi, c_h y - c_h \xi\}.$$

This observation will be useful to linearize constraint (2c).

By omitting i indices and simplifying the notation of problem (2) by taking into account the special structure of the newsvendor problem, we obtain the following dual exponential cone constrained MIP as its robust counterpart:

$$\begin{aligned}
\min\ & cy + \left[\alpha + \epsilon\beta + \sum_{s=1}^{S} q_s u_s\right] && \text{(6a)} \\
\text{s.t. } & \alpha - v_s \geq -c_b y + c_b d_s && s \in [S] \quad \text{(6b)} \\
& \alpha - v_s \geq c_h y - c_h d_s && s \in [S] \quad \text{(6c)} \\
& \beta + w_s = 0 && s \in [S] \quad \text{(6d)} \\
& y \in \mathbb{Z}_+, \alpha \in \mathbb{R}, \beta \in \mathbb{R}_+ && \text{(6e)} \\
& (u_s, v_s, w_s) \in (K_{\text{exp}})_* && s \in [S]. \quad \text{(6f)}
\end{aligned}$$

4.2. Computations

4.2.1. Experimental setup

To compare the effect of robustness level of KL divergence constrained DR version of the Newsvendor Problem, we propose Algorithm 1. Note that setting $\epsilon = 0$ in problem (6) reduces it to the stochastic programming approach while larger values of ϵ lead to more robustness (and conservativeness) in solutions.

Algorithm 1. ***Input:*** *A probability distribution $\mathcal{D}$, the number of samples R, the set of robustness levels $\mathcal{T}$.*

1: *Sample R random variates from $\mathcal{D}$ for training, and obtain the empirical distribution q and the maximum KL divergence $\bar{\epsilon}(q)$ as computed in Proposition 3.*

2: *Solve problem* (6) *with $\epsilon := \theta\bar{\epsilon}(q)$ for each $\theta \in \mathcal{T}$ to obtain a decision $y^*(\theta)$.*

3: Sample R random variates from $\mathcal{D}$ for testing, and then compute the cost realizations for each realization under the decision $y^*(\theta)$.

We implement Algorithm 1 in the Python programming language and use MOSEK 9.2 [29] to solve the dual exponential cone constrained MIP problem (6).

4.2.2. Results

For this illustration, we choose the following cost coefficients:

$$c = 1,\ c_b = 2,\ c_h = 1.$$

We will now specify the parameters of Algorithm 1. Firstly, we experiment with three different discrete distributions:

- Discrete Uniform Distribution with parameters 0 and 10, denoted as `Uniform(0, 10)`.
- Binomial Distribution with parameters 10 and 0.5, denoted as `Binomial(10, 0.5)`.
- Poisson Distribution with parameter 5, denoted as `Poisson(5)`.

We sample $R = 100$ random variates separately to obtain "training" and "test" datasets. Then, we repeat the experiments for the following "robustness" levels:

$$\mathcal{T} := \{0.00, 0.05, 0.10, 0.15, 0.20, 0.25\}.$$

The summary statistics of our experiments are reported in Tables 1-3 for Uniform, Binomial and Poisson distributions, respectively. In particular, we report the average and standard deviation of the cost realizations, abbreviated as "Avg." and "St. Dev.", respectively. In addition, we compute the average of the worst 10% of the realizations, abbreviated as "Worst 10%", to quantify the risk.

We observe that as the robustness level θ increases, the optimal order quantity y^* increases (recall that $\theta = 0.00$ corresponds to the stochastic programming approach). Moreover, with increasing θ, the average cost increases while the standard deviation and the average of worst realizations decrease for each distribution. This is an expected behavior when robust optimization is utilized. We note that Binomial distribution is the least sensitive with respect to θ as the order quantity (and performance measures) do not change after $\theta \geq 0.05$. On the other hand, Uniform and Poisson distributions are more sensitive with respect to this parameter.

We also repeat the experiments with even higher values of θ and observe that only the results corresponding to the Poisson distribution changes, which we attribute to its right-skewness. However, the order quantities in those cases are very high, which result in overly conservative policies and deteriorated performance measures.

Table 1. Summary results for the Newsvendor Problem with `Uniform(0, 10)` and $R = 100$.

θ	y^*	Avg.	St. Dev.	Worst 10%
0.00	4	8.08	2.92	13.80
0.05	4	8.08	2.92	13.80
0.10	5	8.67	2.22	12.80
0.15	5	8.67	2.22	12.80
0.20	5	8.67	2.22	12.80
0.25	6	9.53	1.94	12.00

Table 2. Summary results for the Newsvendor Problem with `Binomial(10, 0.5)` and $R = 100$.

θ	y^*	Avg.	St. Dev.	Worst 10%
0.00	4	6.76	2.38	11.40
0.05	5	7.02	1.67	10.40
0.10	5	7.02	1.67	10.40
0.15	5	7.02	1.67	10.40
0.20	5	7.02	1.67	10.40
0.25	5	7.02	1.67	10.40

Table 3. Summary results for the Newsvendor Problem with `Poisson(5)` and $R = 100$.

θ	y^*	Avg.	St. Dev.	Worst 10%
0.00	4	7.59	3.04	13.60
0.05	5	7.73	2.40	12.60
0.10	5	7.73	2.40	12.60
0.15	5	7.73	2.40	12.60
0.20	6	8.44	1.86	12.00
0.25	6	8.44	1.86	12.00

In addition to the summary statistics, we also provide the box plots of the cost realizations in Figures 1-3 for Uniform, Binomial and Poisson distributions, respectively. We observe that as the robustness level θ increases, the median of the cost realizations increases while the range shrinks for each distribution. We also note that the maximum and upper quartile values decrease for $\theta \in [0.05, 0.15]$. This is a desired property since it implies that the risk of the stochastic programming approach ($\theta = 0.00$) can be lowered.

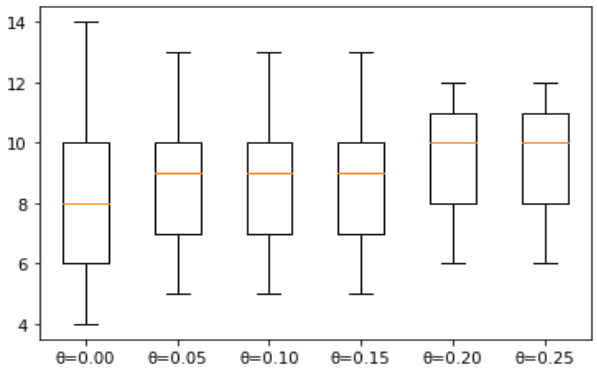


Figure 1. Box plot of the results for the Newsvendor Problem with `Uniform(0, 10)` and $R = 100$.

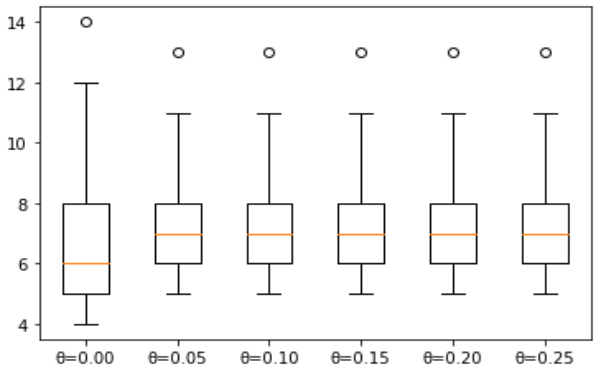


Figure 2. Box plot of the results for the Newsvendor Problem with `Binomial(10, 0.5)` and $R = 100$.

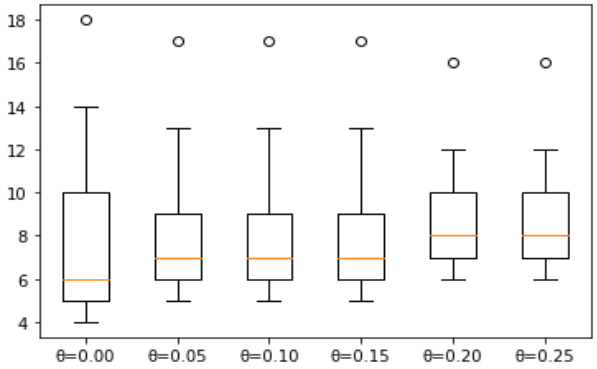


Figure 3. Box plot of the results for the Newsvendor Problem with `Poisson(5)` and $R = 100$.

As a final comparison, we formulate the DR version of the Newsvendor Problem assuming that the first and second moments are known (in our case, estimated from the training data) following [39]. We observe that the optimal order quantity y^* obtained from this approach turns out to be identical to the optimal order quantity obtained from the stochastic programming approach in our experimental setting.

5. Application to the uncapacitated facility location Problem

In this section, we analyze the KL divergence constrained DR version of the Uncapacitated Facility Location (UFL) Problem.

5.1. Deterministic version

We first remind the reader the deterministic version of the well-known UFL Problem. Suppose that we have m customers, each with demand d^i, $i \in [m]$. The demand must be satisfied by opening new facilities. There are n potential facilities, each with a fixed cost of f_j, $j \in [n]$. The unit transportation cost between each customer i and facility j is given as t_{ij}, $i \in [m]$, $j \in [n]$. The objective is to minimize the total fixed cost and transportation cost.

The UFL Problem can be modeled as an integer program by defining two sets of binary decision variables. The first set of decision variables, denoted as y_j, represent the status of each facility j, and the second set of decision variables, denoted as x_{ij}, represents the assignment of a customer i to a facility j. The complete model is given as follows:

$$\min \sum_{j=1}^{n} \left[f_j y_j + \sum_{i=1}^{m} d^i t_{ij} x_{ij} \right] \tag{7a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1 \qquad i \in [m] \tag{7b}$$

$$x_{ij} \leq y_j \qquad i \in [m]; j \in [n] \tag{7c}$$

$$x_{ij} \in \{0,1\} \qquad i \in [m]; j \in [n] \tag{7d}$$

$$y_j \in \{0,1\} \qquad i \in [m]; j \in [n]. \tag{7e}$$

Here, constraint (7b) guarantees that each customer is served by exactly one facility while constraint (7c) ensures that each customer is served by an open facility.

We point out two useful observations about the UFL Problem. Firstly, in any feasible solution to problem (7), at least one facility must be opened. Therefore, we must have

$$\sum_{j=1}^{n} y_j \geq 1. \tag{8}$$

Secondly, given any optimal y^* vector, the optimal objective function value can be computed as

$$\sum_{j=1}^{n} f_j y_j^* + \sum_{i=1}^{m} d^i \min_{j: y_j^* = 1} \{t_{ij}\}, \tag{9}$$

since each customer can be served by the closest open facility.

5.2. Distributionally robust version

Now, suppose that we replace the deterministic demand d^i with a random variable ξ^i having an empirical distribution $q^i \in \text{ri}(\Delta^{S_i})$, with realizations d^i_s, $s \in [S_i]$. Then, the DR version of the UFL Problem can be modeled as an instance of the generic model (1) with $\mathcal{Y} := \{y \in \{0,1\}^n : (8)\}$ as follows: We choose functions

$$h(y) := \sum_{j=1}^{n} f_j y_j$$

and

$$H^i(y, \xi^i) := \xi^i \min_{j:y_j=1}\{t_{ij}\},\ i = 1, \ldots, m,$$

due to (9). In the remainder of this subsection, we will obtain the robust counterpart of the KL divergence constrained DR UFL Problem as a dual exponential cone constrained MIP by the help of Lemma 1.

5.2.1. A lemma

The following lemma will be critical to linearize constraint (2c).

Lemma 1. *Let $t \in \mathbb{R}^n_+$ be a given vector and consider the function $g(y) : \{0,1\}^n \to \mathbb{R}$ defined as $g(y) := \min\{t_j : y_j = 1\}$. Then, for any $y \in \{0,1\}^n$ satisfying (8), we have*

$$g(y) = \max_{l=1,\ldots,n} \big\{t_l - \sum_{j=1}^{n} y_j \max\{t_l - t_j, 0\}\big\}. \quad (10)$$

Proof. Let $y \in \{0,1\}^n$ satisfying (8) be given. We define the following nonempty sets $T := \{j : y_j = 1\}$ and $T^* := \text{argmin}\{t_j : j \in T\}$. Notice that $g(y) = t_l$ for $l \in T^*$. Also, let us define the quantity

$$z_l := t_l - \sum_{j=1}^{n} y_j \max\{t_l - t_j, 0\},\ l \in [n],$$

for convenience. Notice that we have

$$y_j \max\{t_l - t_j, 0\} = \begin{cases} 0 & \text{if } j \notin T \\ 0 & \text{if } j \in T \text{ and } t_{il} \le t_{ij} \\ t_{il} - t_{ij} & \text{if } j \in T \text{ and } t_{il} > t_{ij} \end{cases}.$$

This observation helps us to rewrite z_l as

$$z_l = t_l - \sum_{j \in T: t_l > t_j} (t_l - t_j),\ l \in [n].$$

Now, we will look at the following cases to compute or bound z_l:

Case 1: Let $l^* \in T^*$. Then, we have $z_{l^*} = t_{l^*}$.

Case 2: Let $l \notin T^*$, and choose any $j^* \in T^*$. Then, we have

$$\begin{aligned} z_l &= t_l - (t_l - t_{j^*}) - \sum_{j \in T \setminus \{j^*\}: t_{il} > t_{ij}} (t_l - t_j) \\ &= t_{j^*} - \sum_{j \in T \setminus \{j^*\}: t_{il} > t_{ij}} (t_l - t_j) \\ &\le t_{j^*}. \end{aligned}$$

This analysis indicates that

$$\max_{l \in [n]} \big\{t_l - \sum_{j=1}^{n} y_j \max\{t_l - t_j, 0\}\big\} = \max_{l \in [n]} z_l = t_{l^*},$$

where $l^* \in T^*$. Hence, we conclude that equation (10) holds true. □

An alternative proof of Lemma 1 can be obtained via LP duality: First, one would write the problem $\min\{t_j : y_j = 1\}$ as an IP by introducing additional binary variables x_j. Secondly, this IP can be relaxed as an LP due to the totally unimodular structure. Then, the extreme points of the feasible region of the dual LP can be characterized, enabling the dual LP to be solved in closed form (see dual based arguments in [40, 41]).

5.2.2. The final formulation

Taking into account the special structure of the UFL Problem and utilizing Lemma 1 by setting $g := H^i$ for each $i \in [m]$, we obtain the following dual exponential cone constrained MIP:

$$\min \sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m} \Big[\alpha^i + \epsilon^i \beta^i + \sum_{s=1}^{S_i} q^i_s u^i_s\Big] \quad (11a)$$

$$\text{s.t. } \alpha^i - v^i_s \ge d^i_s \big(t_{il} - \sum_{j=1}^{n} y_j \max\{t_{il} - t_{ij}, 0\}\big)$$
$$i \in [m]; s \in [S_i]; l \in [n] \quad (11b)$$
$$(2c) - (2e),\ (7e),\ (8).$$

5.3. Computations

5.3.1. Experimental setup

We utilize Algorithm 2 to compare the effect of robustness level to KL divergence constrained DR version of the UFL Problem. This algorithm is quite similar to Algorithm 1 used for the analysis of the Newsvendor Problem.

Algorithm 2. ***Input:*** *A probability distribution $\mathcal{D}$, the number of samples R, the set of robustness levels $\mathcal{T}$.*

1: *Sample R random variates from $\mathcal{D}$ for each customer $i \in [m]$ for training, and obtain the empirical distribution q^i and the maximum KL divergence $\overline{\epsilon}(q^i)$ for each $i \in [m]$.*

2: *Solve problem* (11) *with* $\epsilon^i := \theta\bar{\epsilon}(q^i)$ *for each* $i \in [m]$ *and* $\theta \in \mathcal{T}$ *to obtain a decision vector* $y^*(\theta)$.
3: *Sample* R *random variates from* $\mathcal{D}$ *for each* $i \in [m]$ *for testing, and then compute the cost realizations for each realization under the decision vector* $y^*(\theta)$.

5.3.2. Results

For this illustration, we assume that there are 12 customers and three potential facilities located in the unit interval. Their precise locations are given as

$$\left\{\frac{2\omega-1}{36} : \omega \in [6]\right\} \cup \left\{\frac{35-2\omega}{36} : \omega \in [6]\right\},$$

and

$$\left\{\frac{2\Omega-1}{6} : \Omega \in [3]\right\},$$

respectively, and are shown in Figure 4.

Figure 4. Locations of the customers (circles) and potential facilities (diamonds).

As evident from the figure, potential facilities are located evenly across the unit interval and there are two clusters of customers which are also distributed evenly in their respective regions. The fixed cost of opening facilities are given as $f_1 = f_3 = 10$ for the two facilities in the middle of these clusters (marked by a large diamond), and $f_2 = 5$ for the other facility (marked by a small diamond). Finally, the unit transportation cost between a facility-customer pair is assumed to be equal to the their distance from each other.

We specify the parameters of Algorithm 2 regarding the generation of random variates similar to that of Algorithm 1 as described in Section 4.2.2.

The summary statistics of our experiments are reported in Tables 4-6 for Uniform, Binomial and Poisson distributions, respectively. We first observe that the optimal solutions and the performance measures are similar for every distribution, therefore, we will summarize our observations together. Due to the choice of parameters and the locations of the facilities and customers as can be seen from Figure 4, there is a fundamental trade-off in this instance: We can i) either open a single facility at the middle of the line segment with the lower fixed cost and serve customers via longer distances, or ii) open two facilities at the middle of two customer clusters with higher fixed cost and serve customers via shorter distances. In the stochastic programming approach ($\theta = 0.00$), the first policy becomes optimal whereas in the DR approach ($\theta \geq 0.05$), the second policy becomes optimal. We note that considering the ambiguity of the demand distributions increases the average cost only slightly whereas both the standard deviation and the average of the worst 10% of the realizations decrease significantly. We remind the reader that the total fixed cost of the stochastic programming approach is only 5 while the fixed cost of the DR approach is 20. This also shows that the corresponding transportation cost, which is affected by the random uncertainty, is significantly smaller in the DR approach.

Table 4. Summary results for the UFL Problem with `Uniform(0, 10)` and $R = 100$.

θ	y^*	Avg.	St. Dev.	Worst 10%
0.00	0,1,0	23.11	3.47	29.19
0.05	1,0,1	24.52	0.92	26.10
0.10	1,0,1	24.52	0.92	26.10
0.15	1,0,1	24.52	0.92	26.10
0.20	1,0,1	24.52	0.92	26.10
0.25	1,0,1	24.52	0.92	26.10

Table 5. Summary results for the UFL Problem with `Binomial(10, 0.5)` and $R = 100$.

θ	y^*	Avg.	St. Dev.	Worst 10%
0.00	0,1,0	24.87	1.88	28.15
0.05	1,0,1	24.97	0.52	25.86
0.10	1,0,1	24.97	0.52	25.86
0.15	1,0,1	24.97	0.52	25.86
0.20	1,0,1	24.97	0.52	25.86
0.25	1,0,1	24.97	0.52	25.86

Table 6. Summary results for the UFL Problem with `Poisson(5)` and $R = 100$.

θ	y^*	Avg.	St. Dev.	Worst 10%
0.00	0,1,0	24.96	2.67	29.89
0.05	1,0,1	24.99	0.74	26.34
0.10	1,0,1	24.99	0.74	26.34
0.15	1,0,1	24.99	0.74	26.34
0.20	1,0,1	24.99	0.74	26.34
0.25	1,0,1	24.99	0.74	26.34

In addition to the summary statistics, we also provide the box plots of the cost realizations in Figures 5-7 for Uniform, Binomial and Poisson distributions, respectively. We observe that the median of the cost realizations either stays the same or increases slightly in the DR approach while the range shrinks significantly compared to the

stochastic programming approach for each distribution. We also note that the maximum and upper quartile values decrease considerably with the DR approach (especially for Binomial and Poisson distributions).

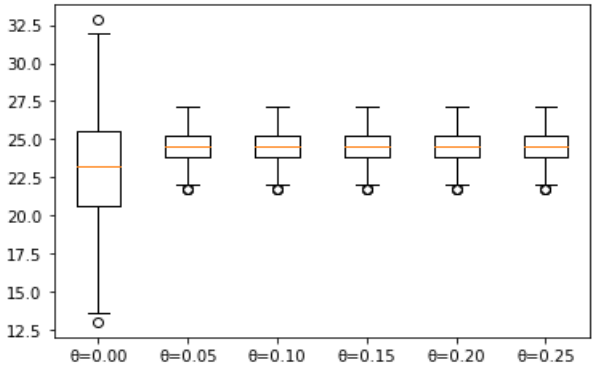


Figure 5. Box plot of the results for the UFL Problem with `Uniform(0, 10)` and $R = 100$.

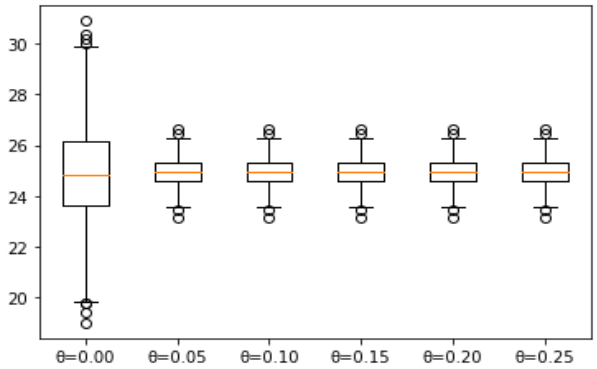


Figure 6. Box plot of the results for the UFL Problem with `Binomial(10, 0.5)` and $R = 100$.

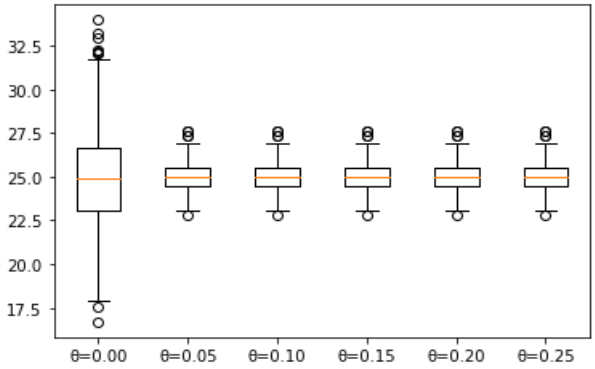


Figure 7. Box plot of the results for the UFL Problem with `Poisson(5)` and $R = 100$.

6. Conclusion

In this paper, we analyzed the KL divergence constrained DRO problems and proposed their dual exponential cone constrained reformulations utilizing the exponential cone representability property of KL divergence and Conic Duality. The resulting robust counterpart can be solved by a commercial conic programming solver directly. We specialized our results to the Newsvendor and UFL Problems by providing problem specific reformulations, and conducted a computational analysis comparing the performance of the solutions obtained via DR approach and stochastic programming from different aspects. We observed that although the mean and median of the cost realizations deteriorate slightly when the DR approach is preferred; the range, standard deviation and worst case values of the cost realizations improve significantly compared to stochastic programming approach.

Some future research directions seem promising. Firstly, by utilizing the semidefinite programming approximations of the matrix logarithm [42], we can try to formulate and solve KL-divergence constrained DRO problems that involve multivariate normal distributions. Secondly, we would like to test the success of the proposed method on different problems with real datasets. Lastly, we may adapt our results to the decision-dependent setting, which is a recent active research area in the DRO literature [35, 36, 43].

Acknowledgments

The author would like to thank Dr. Beste Basciftci for her comments on an earlier version of this paper.

References

[1] Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming*. Springer Science & Business Media.

[2] Shapiro, A., Dentcheva, D., & Ruszczyński, A. (2014). *Lectures on stochastic programming: modeling and theory*. Society for Industrial and Applied Mathematics.

[3] Ben-Tal, A., & Nemirovski, A. (2002). Robust optimization–methodology and applications. *Mathematical Programming*, 92(3), 453-480.

[4] Ben-Tal, A., El Ghaoui, L., & Nemirovski, A. (2009). *Robust optimization*. Princeton University Press.

[5] Bertsimas, D., Brown, D. B., & Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM Review*, 53(3), 464-501.

[6] Popescu, I. (2007). Robust mean-covariance solutions for stochastic optimization. *Operations Research*, 55(1), 98–112.

[7] Delage, E. and Ye, Y. (2010). Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3), 595-612.

[8] Wiesemann, W., Kuhn, D., & Sim, M. (2014). Distributionally robust convex optimization. *Operations Research*, 62(6), 1358-1376.

[9] Gao, R. & Kleywegt, A. J. (2016). Distributionally robust stochastic optimization with Wasserstein distance. *Optimization Online*.

[10] Mohajerin Esfahani, P. & Kuhn, D. (2018). Data-driven distributionally robust optimization using the Wasserstein metric: performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1), 115-166.

[11] Hanasusanto, G. A., & Kuhn, D. (2018). Conic programming reformulations of two-stage distributionally robust linear programs over Wasserstein balls. *Operations Research*, 66(3), 849-869.

[12] Xie, W. (2019). On distributionally robust chance constrained programs with Wasserstein distance. *Mathematical Programming*, 1-41.

[13] Ben-Tal, A., Den Hertog, D., De Waegenaere, A., Melenberg, B., & Rennen, G. (2013). Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2), 341-357.

[14] Klabjan, D., Simchi-Levi, D., & Song, M. (2013). Robust stochastic lot-sizing by means of histograms. *Production and Operations Management*, 22(3), 691-710.

[15] Jiang, R., & Guan, Y. (2016). Data-driven chance constrained stochastic program. *Mathematical Programming*, 158(1-2), 291-327.

[16] Yanıkoğlu, İ., den Hertog, D., & Kleijnen, J. P. (2016). Robust dual-response optimization. *IIE Transactions*, 48(3), 298-312.

[17] Lam, H. (2019). Recovering best statistical guarantees via the empirical divergence-based distributionally robust optimization. *Operations Research*, 67(4), 1090-1105.

[18] Zymler, S., Kuhn, D., & Rustem, B. (2013). Distributionally robust joint chance constraints with second-order moment information. *Mathematical Programming*, 137(1-2), 167-198.

[19] Yanıkoğlu, İ., & den Hertog, D. (2013). Safe approximations of ambiguous chance constraints using historical data. *INFORMS Journal on Computing*, 25(4), 666-681.

[20] Xie, W., & Ahmed, S. (2018). On deterministic reformulations of distributionally robust joint chance constrained optimization problems. *SIAM Journal on Optimization*, 28(2), 1151-1182.

[21] Yanıkoğlu, İ. (2019). Robust reformulations of ambiguous chance constraints with discrete probability distributions. *An International Journal of Optimization and Control: Theories & Applications*, 9(2), 236-252.

[22] Nesterov, Y., & Nemirovski, A. (1994). *Interior-point polynomial algorithms in convex programming*. Society for Industrial and Applied Mathematics.

[23] Serrano, S. A. (2015). Algorithms for unsymmetric cone optimization and an implementation for problems with the exponential cone. PhD Thesis. Stanford University.

[24] Dahl, J., & Andersen, E. D. (2019). A primal-dual interior-point algorithm for nonsymmetric exponential-cone optimization. *Optimization Online*.

[25] Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22(1), 79-86.

[26] Hu, Z., & Hong, L. J. (2013). Kullback-Leibler divergence constrained distributionally robust optimization. *Optimization Online*.

[27] Chen, Y., Guo, Q., Sun, H., Li, Z., Wu, W., & Li, Z. (2018). A distributionally robust optimization model for unit commitment based on Kullback–Leibler divergence. *IEEE Transactions on Power Systems*, 33(5), 5147-5160.

[28] Li, Z., Wu, W., Zhang, B., & Tai, X. (2018). Kullback–Leibler divergence-based distributionally robust optimisation model for heat pump day-ahead operational schedule to improve PV integration. *IET Generation, Transmission & Distribution*, 12(13), 3136-3144.

[29] MOSEK ApS. (2020). MOSEK optimizer API for Python.

[30] Hanasusanto, G. A., Kuhn, D., Wallace, S. W., & Zymler, S. (2015). Distributionally robust multi-item newsvendor problems with multimodal demand distributions. *Mathematical Programming*, 152(1-2), 1-32.

[31] Natarajan, K., Sim, M., & Uichanco, J. (2018). Asymmetry and ambiguity in newsvendor models. *Management Science*, 64(7), 3146-3167.

[32] Lee, S., Kim, H., & Moon, I. (2020). A data-driven distributionally robust newsvendor model with a Wasserstein ambiguity set. *Journal of the Operational Research Society*, 1-19.

[33] Lu, M., Ran, L., & Shen, Z. J. M. (2015). Reliable facility location design under uncertain correlated disruptions. *Manufacturing & Service Operations Management*, 17(4), 445-455.

[34] Santiváñez, J. A., & Carlo, H. J. (2018). Reliable capacitated facility location problem with service levels. *EURO Journal on Transportation and Logistics*, 7(4), 315-341.

[35] Basciftci, B., Ahmed, S., & Shen, S. (2020). Distributionally robust facility location problem under decision-dependent stochastic demand. *European Journal of Operational Research*. doi:10.1016/j.ejor.2020.11.002.

[36] Noyan, N., Rudolf, G., & Lejeune, M. (2018). Distributionally robust optimization with decision-dependent ambiguity set. *Optimization Online*.

[37] MOSEK ApS. (2020). MOSEK modeling cookbook.

[38] Ben-Tal, A., & Nemirovski, A. (2001). *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*. Society for Industrial and Applied Mathematics.

[39] Scarf, H. A. (1958). A min-max solution of an inventory problem. In: K. J. Arrow, S. Karlin and H. E. Scarf, Eds., *Studies in the Mathematical Theory of Inventory and Production*, Stanford University Press, California, 201-209.

[40] Erlenkotter, D. (1978). A dual-based procedure for uncapacitated facility location. *Operations Research*, 26(6), 992-1009.

[41] Conn, A. R., & Cornuejols, G. (1990). A projection method for the uncapacitated facility location problem. *Mathematical Programming*, 46(1-3), 273-298.

[42] Fawzi, H., Saunderson, J., & Parrilo, P. A. (2019). Semidefinite approximations of the matrix logarithm. *Foundations of Computational Mathematics*, 19(2), 259-296.

[43] Luo, F., & Mehrotra, S. (2020). Distributionally robust optimization with decision dependent ambiguity sets. *Optimization Letters*, 14, 2565-2594.

Burak Kocuk *is an assistant professor at the Industrial Engineering Program, Sabancı University. He obtained his BS degrees in Industrial Engineering and Mathematics, and MS degree in Industrial Engineering from Boğaziçi University. He obtained his PhD degree of Operations Research at the School of Industrial and Systems Engineering, Georgia Institute of Technology. Before joining Sabancı University, he was a postdoctoral fellow at the Tepper School of Business, Carnegie Mellon University. His current research focuses on mixed-integer nonlinear programming and stochastic optimization problems, from both theoretical and methodological aspects.*

https://orcid.org/0000-0002-4218-1116

An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957eISSN: 2146-5703
Vol.11, No.2, pp.152-157 (2021)
https://doi.org/10.11121/ijocta.01.2021.00990

RESEARCH ARTICLE

Taguchi's method of optimization of fracture toughness parameters of Al-SiC_p composite using compact tension specimens

Hareesha Guddhur [a *], Chikkanna Naganna [b], Saleemsab Doddamani [c]

[a] *Department of Mechanical Engineering, Government Engineering College, HuvinaHadagali,VTU, Belagavi, India*
[b] *Department of Aerospace Propulsion Technology, VTU, VIAT, Muddenahalli, Bangalore, VTU, Belagavi, India*
[c] *Department of Mechanical Engineering, Jain Institute of Technology, Davangere, VTU, Belagavi, India*
harishssb@gmail.com, nchikkanna1967@gmail.com, saleemsabdoddamani@gmail.com

ARTICLE INFO

Article history:
Received: 14 June 2020
Accepted: 18 April 2021
Available Online: 20 April 2021

Keywords:
Al-SiC_p
Fracture Toughness
Compact tension specimens
Taguchi's method
ANOVA

AMS Classification 2010:
62J10, 74A45, 62K86

ABSTRACT

The objective of this work is to investigate the process parameters which influence the fracture toughness of aluminum-silicon carbide particulate composite prepared using the stir casting technique. The Taguchi's design of experiments is conducted to analyze the process parameters. Three parameters considered are composition of material, grain size and a/W ratio. From the Taguchi's analysis, on compact tension specimens, aluminum 6061 reinforced with 9 wt% of the silicon carbide particles composite and a/W ratio of 0.45 are considered to be optimized parameters. Taguchi's technique result shows that the increment in the a/W ratio causes decrement in the load carrying capacity of the composite. Whereas the fine grain size of silicon carbide have better toughness values. From the ANOVA outcomes it is clear that the composition and a/W ratio of the geometry has more influence on the fracture toughness than the grain size of reinforcement.

1. Introduction

A common expression for measurement of materials ability to resist the crack propagation is generally referred as fracture toughness. Estimation and examination of fracture toughness has been a serious problem being the growth for the fracture mechanics approach and their applications in engineering. The concept of fracture mechanics [1] consists of some of the significant parameters like stress intensity factor (K), energy release rate (G), the crack-tip opening displacement (CTOD) and the J integral. The resistance to the crack growth is known as fracture toughness which can be determined experimentally using many testing methods. American society for testing and materials (ASTM) [2] proposed many standard testing methods to test the fracture toughness of the metallic materials in E399-17. As per ASTM many standard specimens [3] were utilized for K_{Ic} testing such as compact tension (CT) specimen, single edge notch bend (SENB) specimens etc.

These specimens, now a days, widely used to test the metallic composites such as metal matrix composites (MMCs). These composites have been utilized when it required in the application of weight reduction, wear and corrosion resistance and thermal management. Aluminum as a matrix and silicon carbide, alumina etc as reinforcements are widely used, present day, metal matrix composites [4]. The properties of the particulate type metal matrix composites were influenced by the many factors like particle size, weight/volume fraction, inter particle spacing etc.

To prepare these metal matrix composites, among many methods, stir casting technique is widely utilized. Literature shows that the Al-SiC [5], Al-graphite [6,7] and also aluminum based hybrid [8-10] composites were prepared from the stir casting technique. Mechanical properties of the particulate reinforced with aluminum for varying weight/mass fractions has been carried out using Universal Testing Machine [5,6] (UTM). Fracture toughness [5], Indentation Fracture Toughness [11,12] tensile fracture behaviour on circumferential notched tensile (CNT) specimens [5, 13] and Compact Tension (CT) test [14] specimen and single edge notch bend (SENB) specimen [15] of aluminum alloy with different reinforcements were studied by different researchers.

Also the effect of reinforcement addition on the base metal [16], effect of specimen thickness [17], aging [18] and fatigue crack growth behavior [19] on the fracture toughness using CT specimens has been

*Corresponding author

examined. The different fracture toughness testing methods were compared [20] and found that the results obtained from all the testing techniques agree with each other. The researchers also conducted different tensile [21], fracture toughness [5,22,23] investigation using CT specimens on aluminum silicon carbide composites.

Literature review reveals that the mechanical characterization of the aluminum silicon carbide has been studied extensively. In this background, there is a scope for the study of aluminum-silicon carbide particulate composite in the area of fracture mechanics. Through this investigation, an attempt has been made to investigate impact of process parameters on the fracture toughness of the aluminum-silicon carbide particulate (Al-SiC$_p$) composite. The Taguchi's design of experiments and ANOVA are intended to use to analyze the process parameters such as composition of the material, a/W ratio of the geometry and the grain size of the reinforcement.

2. Materials and processing

In the present work aluminum 6061 is used as a matrix and silicon carbide particles are used as reinforcement. A precipitation-hardened aluminium (Al6061) alloy has its main alloying elements as magnesium (0.81 wt%) and silicon (0.70 wt%). Some general chrestersitcs of Al6061 are mentioned in the Table 1.

Table 1. General charecterstics of Al6061

Sl no	Charecterstics	Value
1	Hardness	95 BHN
2	Yield Strength	275MPa
3	Elastic modulus	68.9GPa
4	Tensile Stength	315MPa
5	Elongation	17%
6	Density	2.65g/cc
7	Melting Temp	650^{o}C

Silicon carbide (SiC), a form of carborundum, is a ceramic material. It is a combination of silicon and carbon. SiC is one of exceptional abrasive materials used to manufacture abrasive wheels. Now a days the SiC available is of high quality technical grade ceramic with excellent physical properties. Density = 3.1g/cc, melting point – 2730°C, Appearance –Black in color, Hardness = 45.8 GPa [21] were some of the key properties of silicon carbide.

The aluminum silicon carbide particulate (Al-SiC$_p$) composites demonstrate isotropic properties [5] as well as exceptional combination of structural and physical properties. The particle size of the silicon carbide, among many factors, is the most significant variable considered which will influence the microstructure of the composite. The particle sizes of silicon carbide utilized in this work are 44 μm, 75 μm and 150 μm.

Stir casting method [6-20] is used to cast the Al-SiC$_p$ MMCs at 6, 9 and 12% weight fractions of SiC. The aluminum super heated above its melting point (i.e.720°C) and predetermined quantity of reinforcement particles and degassifiers are added while stirring at speed of 500 rpm [6-10, 24]. The molten Al-SiC is poured to the graphite mold and it is allows for solidification. The block took from the mold were machined to the required size of the specimen.

3. Design of experimentation

Taguchi strategy of optimization is one of the best strategies in view of its straightforwardness to do the design of experiments [24]. The objective of the design of experiments is to determine the significant factors which influence the fracture toughness to optimize the process parameters from which can increase the toughness, minimise the crack initiation and propagation. The procedure given in the Taguchi's design of experiments is to examine the various parameters and their effect on the mean and variance. From the results of the Taguchi's design of experiments, ANOVA (analysis of variance) [25] has been carried out, to optimize the performance behavior, and to choose the new process parameter.

In the current work, optimizing the parameters of the fracture toughness tests is done utilizing the Taguchi's technique. Three factors and three levels for each are considered to analyze their performance behaviour. Factors considered are composition of material, grain size of reinforcement and a/W ratio. Levels considered are composition of materials considered are 6, 9 and 12 wt% of SiC reinforcement, grain size of reinforcement considered are 44μm, 75μm and 150μm and a/W ratio = 0.45, 0.47 and 0.50, and. The Taguchi's L9 orthogonal array has been given in Table 1.

4. Experimentation

The CT specimen's geometry used, as given in the Figure 1, is as per the standard testing procedures for fracture toughness of metallic materials as prescribed by the ASTM. Fracture toughness testing for Al-SiC$_p$ is conducted using the standard universal testing machine (UTM) as per the available testing procedure. Some of the standard specifications are mentioned in the Table 2.

The servo-hydraulic testing machine is used to carry out, in the room temperature, the fracture toughness experiments. The crack length developed, for a/W ratios varied from 0.45 to 0.50, is measured using visual method. By this way the specimens are fatigue pre-cracked under the mode 1 (tensile) loading. Below equation (Eq.1) [3] is utilized to determine the fracture toughness of Al-SiC$_p$ composites using the load at fracture (P$_f$). The plot of load applied versus crack opening displacement gives the type III curve [2,3,14]. Hence the load at fracture, for type III curve, is itself is critical load (P$_Q$).

Table 2. Standard specifications of UTM

Sl no	Charecterstics	Value
1	Capacity, Ton	20
2	Test Speed, mm/min	0.01 to 500
3	Test Temperature ºC	Room Temp
4	Display	Digital
5	Testing Standard	ASTM E83
6	Accuracy	0.5µm
7	Transmission	Hydraulic
8	Loading frequency	5 Hz
	Calibration standard	
9	Crosshead speed	ASTM E2658
10	Crosshead displacement	ASTM E2309
11	Strain and load rate	ASTM E2309
12	Measurement of tension	ASTM E4

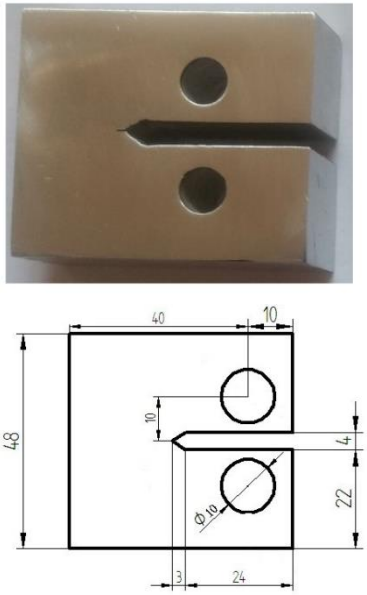


Figure 1. CT Specimen with geometry

$$K_{Ic} = \frac{P_Q}{B\sqrt{W}} f\left(\frac{a}{W}\right) \quad (1)$$

Where f(a/W) is expressed as follows:

$$f\left(\frac{a}{W}\right) = \frac{\left(2+\frac{a}{W}\right)}{\left(1-\frac{a}{W}\right)^{3/2}}\left(0.886 + 4.64\left(\frac{a}{W}\right) - 13.32\left(\frac{a}{W}\right)^2 + 14.72\left(\frac{a}{W}\right)^3 - 5.6\left(\frac{a}{W}\right)^4\right) \quad (2)$$

5. Results and discussions

5.1. Experimental results

In the Table 3, the results of the fracture toughness testing as per the Taguchi's design of experiments (DOE) are listed. From the results, it might be uncovered that with an addition in substance of the SiC in Al-SiC composite the increment in the value of fracture toughness. The improvement in the fracture toughness is a direct result of the impact of the additional SiC particulates which goes about as an inward blockade to the internal microstructural cracks. The values in the Table 3 shows, the decrement in the fracture toughness with the increase in the a/W ratio.

Table 3. Taguchi's DOE and fracture toughness of Al-SiC_p composite

Sl no	Composition % of SiCp	a/W ratio	Grain Size µm	Load at Fracture (P_Q) kN	Fracture Toughness MPa√m
1	6	0.45	44	2.260	9.42
2	6	0.47	75	2.013	8.89
3	6	0.50	150	1.786	8.62
4	9	0.45	75	2.367	9.87
5	9	0.47	150	2.114	9.33
6	9	0.50	44	1.839	8.87
7	12	0.45	150	2.148	8.96
8	12	0.47	44	2.084	9.20
9	12	0.50	75	1.725	8.32

From the Table 3 it is seen that as a/W proportion increases there is a decrement in the value of the fracture toughness. Also it is observed that the increment in the fracture toughness for the fine sized grains of the silicon carbide reinforcement. Taguchi's design has been used to analyze the two input functions viz., values of the experimental fracture toughness and the load carrying capacity which were the major input functions. The results of the examination are appeared in Figure 2(a) and (b).

Taguchi's technique result shows that the increment in the a/W proportion causes decrement in composite's load carrying capacity. As the addition in the SiC reinforcement in Al6061 matrix the load carrying capacity increases up to 9 wt% of SiC and reduces for 12 wt% of SiC. Also from Figure 2(a) it is also apparent that as the grain size of the SiC increases, load carrying capacity decreases. It is obvious that the bigger particle size of reinforcement causes weak bonding between the matrix and reinforcement hence reduces its load carrying capacity.

Figure 2(b) shows the performance of Al-SiC_p composite for different process parameters. For the increment in the parameter a/W ratio, there is the reduction in fracture toughness of material. The increment in a/W ratio is nothing but the increase of crack length for the given width, which causes the decrease of load carrying capacity, hence reduces the fracture toughness. The fine grain size of the reinforcement enhances the matrix and reinforcement bonding and acts as the barrier to the crack initiation and propagation which in turn increases the fracture toughness.

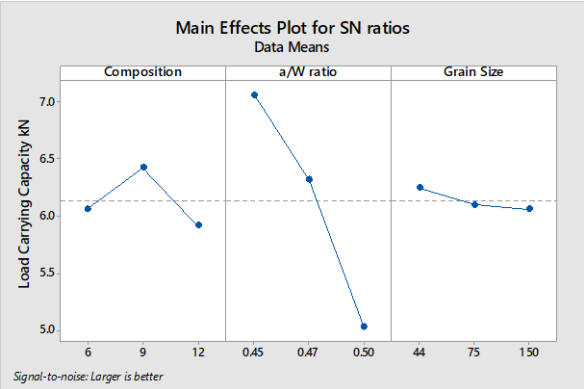


Figure 2(a). Taguchi's design results for load carrying capacity

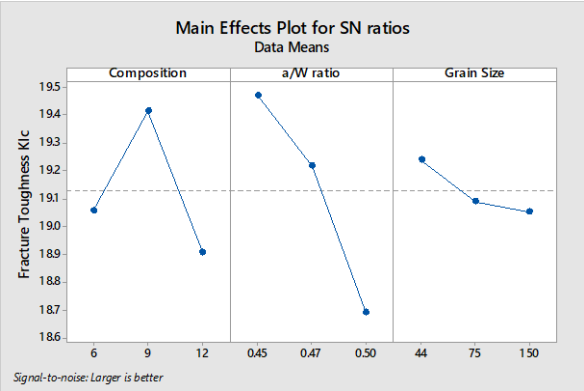


Figure 2(b). Taguchi's design results for fracture toughness

In view of fracture toughness and load carrying capacity the optimized composition is considered as Al6061-9 wt% of SiC, a/W ratio = 0.45 and finer grain size of reinforcement.

5.2. ANOVA (analysis of varience)

Statistical tool utilized to examine the level of individual contribution of the process parameter on the responses is ANOVA. Example: the toughness and load carrying capacity and moreover it gives exact plan of the process parameters. Using ANOVA technique one can analyze and optimize the individual process parameters and their influence on the process. The results of the analysis of the parameters toughness and load carrying capacity are displayed in the Table 4 (a) and (b).

Table 4(a). Analysis of variance for Load carrying capacity

Source	DF	Seq.SS	Adj.MS	P.Value	% contribution
Composition	2	0.0234	0.011	0.288	6.16
a/W ratio	2	0.3433	0.171	0.027	90.54
Grain Size	2	0.0031	0.001	0.755	0.81
Error	2	0.0094	0.004		2.49
Total	8	0.3792			

Table 4(b). Analysis of variance for Fracture Toughness

Source	DF	Seq.SS	Adj.MS	P.Value	% contribution
Composition	2	0.447	0.223	0.264	26.43
a/W ratio	2	1.026	0.513	0.136	60.57
Grain Size	2	0.059	0.029	0.731	3.50
Error	2	0.160	0.080		9.50
Total	8	1.694			

From the Table 4(a-b), the it is observed that the P_value for the a/W ratio is 0.027 which is less than the 0.05. Thus the parameter a/W ratio is considered to be statistically significant. It is also true that, as the a/W ratio increases, crack length (a) increases, thus the load carrying capacity of the material decreases. However, for the fracture toughness, the P_value is slightly higher for the a/W ratio, still affect more on the fracture behaviour of the material.

From the Table 4(a), it is observed that the crack length to width (a/W) ratio majorly effect the load carrying capacity by 90.54% whereas grain size and composition of SiC have a little impact. It is obvious that as crack length (a) increases load carrying capacity decreases.

Also, factors influencing fracture toughness is a/W ratio (60.57%) followed by the composition (26.43%) whereas the grain size of the reinforcement has the least influence on the fracture toughness. This might be due to the use of fine grained reinforcement. The bigger size of the particles (i.e. >150µm) may gives comparably lesser fracture toughness values.

6. Conclusion

From the outcomes of the study, the following conclusions are made: The improvement in the fracture toughness is a direct result of the impact of the addition of fine sized SiC particulates which goes about as an inward blockade to the internal microstructural cracks [16]. Taguchi's technique result shows that the increment in the a/W ratio causes decrement in the composite's load carrying capacity [20,24]. It is obvious that the bigger particle size of reinforcement causes weak bonding between matrix and reinforcement hence reduces the load carrying capacity which in turn decreases fracture toughness of the material. The ANOVA analysis reveals that the a/W ratio [25] followed by the material composition will influence more on the fracture toughness than grain size of the SiC.

References

[1] Anderson, T.L. (2013). *Fracture Mechanics-Fundamentals and Applications*. 3rd Edition, Taylor & Francis Group, New York,

[2] Zhu, X.K. Joyce, J.A. (2012). Review of Fracture Toughness (G, K, J, CTOD, CTOA) Testing and

Standardization, *Engineering Fracture Mechanics, Elsevier*, 85, 1–46.

[3] ASTM Standards. (2017). Standard Test Method for Plane-Strain Fracture Toughness of Metallic Materials. *ASTM International*, E 399-17.

[4] ASM Handbook. (2001). *Composites*. 21. *ASM International*.

[5] Alaneme, K.K., Aluko, A.O. (2012). Fracture toughness (K1C) and tensile properties of as-cast and age-hardened aluminium (6063)–silicon carbide particulate composites. *Scientia Iranica A*, 19(4), 992–996.

[6] Doddamani, S., Kaleemulla, M., Begum, Y. (2015). Experimental Investigation on Tensile Properties of Al6061-Graphite Particulate Composites. *International Journal of Composite Material and Matrices*, 1(2), 1-8.

[7] Taj, A., Doddamani, S., Vijaykumar, T.N. (2017). Vibrational Analysis of Aluminium Graphite Metal Matrix Composite, *International Journal of Engineering Research & Technology (IJERT)*, 6(4), 1072-1078. http://dx.doi.org/10.17577/IJERTV6IS040720.

[8] Rajesh A M, Mohammed Kaleemulla, Saleemsab Doddamani. (2019). Material characterization of SiC and Al2O3 reinforced hybrid aluminum metal matrix composites on wear behavior", *Advanced Composite Letters, SAGE*, 28, 1-10. DOI: 10.1177/0963693519856356.

[9] Rajesh, A.M, Kaleemulla, M., Doddamani, S. (2019). Effect of heat treatment on wear behavior of hybrid aluminum metal matrix composites, *Tribology in Industry*, 41(3), 344-354. DOI: 10.24874/ti.2019.41.03.04.

[10] Rajesh, A.M, Kaleemulla, M., Doddamani, S. (2019). Generation of Mechanically Mixed Layer (MML) in Hybrid Aluminum Metal Matrix Composites under As-cast and Age Hardened Conditions. *SN Aplied Science, Springer*, 1(8). DOI:10.1007/s42452-019-0906-5.

[11] Doddamani, S., Kaleemulla, M. (2015). Review of experimental fracture toughness (K_{Ic}) of aluminium alloy and aluminium MMCs. *International Journal of Fracture and Damage Mechanics*, 1(2), 38-51.

[12] Doddamani, S., Kaleemulla, M. (2016). Indentation Fracture Toughness of Alumnum6061-Graphite Composites, *International Journal of Fracture and Damage Mechanics*, 1(1), 40-46.

[13] Doddamani, S., Kaleemulla, M. (2017). Experimental investigation on fracture toughness of Al6061–graphite by using Circumferential Notched Tensile Specimens. *Frattura ed Integrità Strutturale*, 11(39), 274-281. DOI: 10.3221/IGF-ESIS.39.25.

[14] Doddamani, S., Kaleemulla, M. (2017). Fracture toughness investigations of Al6061- Graphite particulate composite using compact specimens, *Frattura ed Integrità Strutturale*, 11(41), 484-490. DOI:10.3221/IGF-ESIS.41.61.

[15] Doddamani, S., Kaleemulla, M., Kiran, J.O., Bakkappa, B. (2019). Fracture toughness testing of 6061Al-graphite composites using SENB specimens, *Journal of The Institute of Engineers (India)-series D, Springer*, 100(2), 195-201. DOI: 10.1007/s40033-019-00188-z.

[16] Doddamani, S., Kaleemulla, M. (2018). Effect of graphite on fracture toughness of 6061Al-Graphite, *Strength, Fracture and Complexity*, 11(4), 295-308. DOI:10.3233/SFC-180230.

[17] Doddamani, S., Kaleemulla, M. (2019). Effect of Thickness on fracture toughness of Al6061-Graphite, *Journal of Solid Mechanics*, 11(3), 635-643. DOI: 10.22034/jsm.2019.666695.

[18] Doddamani, S., Kaleemulla, M. (2019). Effect of aging on fracture toughness of Al6061-Graphite particulate composites. *Mechanics of Advanced Composite Structures*, 6(2), 139-146 DOI: 10.22075/macs.2019.16436.1177.

[19] Doddamani, S., Kaleemulla, M. (2018). Effect of graphite addition on the fracture and fatigue crack growth behavior of Al6061-Graphite, *Structural Integrity and Life*, 18(3), 185–192. UDC: 66.018.9:539.42.

[20] Doddamani, S., Kaleemulla, M. (2019). Comparisons of experimental fracture toughness testing methods of Al6061-graphite particulate composites, *Journal of Failure Analysis and Prevention, Springer*, 19(3), 730-737. DOI: 10.1007/s11668-019-00652-8.

[21] Singh, V., Prasad, R.C. (2004). Tensile and fracture behavior of 6061 al-sicp metal matrix composites. *International Symposium of Research Students on Materials Science and Engineering*, December 20-22,.

[22] Ranjbaran, M.M. (2010). Experimental investigation of fracture toughness in Al 356-SiCp aluminium matrix composite. *American Journal of Scientific and Industrial Research*, 1(3), 549-557.

[23] Manigandan, K., Srivatsan, T.S., Quick, T. (2012). Influence of silicon carbide particulates on tensile fracture behavior of an aluminum alloy, *Materials Science and Engineering A*, 534, 711–715.

[24] Dhummansure, V., Kalyanrao, A.A., Doddamani, S. (2020). Optimization of process parameters for fracture toughness of Al6061-graphite composites. *Structural Integrity and Life*, 20(1), 51–55.

[25] Begum, Y., Bharath, K.N., Doddamani, S., Rajesh A.M., Kaleemulla, M. (2020). Optimization of process parameters of fracture toughness using simulation technique considering aluminum-graphite composites, Transactions of the Indian Institute of Metals, Springer, 73(12), 3095 – 3103. DOI: 10.1007/s12666-020-02113-5.

Hareesha Guddhur *completed M.Tech in Machine Design in 2008. Presently working as an assistant professor in Government Engineering College, Huvinahadagali, Karnataka, India. He presently perusing the Ph.D in the Visvesvaraya Technological University, Belagavi, Karnataka, India.*
http://orcid.org/0000-0003-1274-9716

Chikkanna Naganna *completed Ph.D in Mechanical Engineering. Presently working as a professor in Department of Aerospace Propulsion Technology, VTU, VIAT, Muddenahalli, Chickballapur, Karnataka, India.*
http://orcid.org/0000-0003-1003-803X

Saleemsab Doddamani *completed Ph.D in Mechanical Engineering in 2019. Presently working as an assistant professor in Department of Mechanical Engineering, Jain Institute of Technology, Karnataka, India. Area of research is fracture mechanics, composites, material science and wear behavior of hybrid composites.*
http://orcid.org/0000-0002-8498-1488

An International Journal of Optimization and Control: Theories & Applications (http://ijocta.balikesir.edu.tr)

An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.11, No.2, pp.158-177 (2021)
https://doi.org/10.11121/ijocta.01.2021.001077

RESEARCH ARTICLE

Differential gradient evolution plus algorithm for constraint optimization problems: A hybrid approach

Muhammad Farhan Tabassum[a], Sana Akram[a], Saadia Hassan[b], Rabia Karim[b], Parvaiz Ahmad Naik[c], Muhammad Farman[d], Mehmet Yavuz[e,*], Mehraj-ud-din Naik[f], Hijaz Ahmad[g,h]

[a] *Department of Mathematics, University of Management and Technology, Lahore, 54000, Pakistan.*
[b] *Department of Sports Sciences, Faculty of Allied Health Science, University of Lahore, Lahore, 54000, Pakistan.*
[c] *School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, Shaanix. People's Rebublic of China.*
[d] *Department of Mathematics and Statistics, University of Lahore, Lahore, 54000, Pakistan*
[e] *Department of Mathematics and Computer Sciences, Faculty of Science, Necmettin Erbakan University, 42090 Konya, Turkey*
[f] *Department of Chemical Engineering, College of Engineering, Jazan University, Jazan 45142, Saudi Arabia*
[g] *Section of Mathematics, International Telematic University Uninettuno, Corso Vittorio Emanuele II, 39, 00186 Roma, Italy*
[h] *Department of Basic Sciences, University of Engineering and Technology, Peshawar, Pakistan*

ARTICLE INFO

Article history:
Received: 19 January 2021
Accepted: 25 March 2021
Available Online: 2 May 2021

Keywords:
Meta-heuristic algorithms
Hybridization
Differential evolution
Gradient evolution
Constraint optimization problems

AMS Classification 2010:
90C15, 90C26, 90C30, 90C59, 90C90

ABSTRACT

Optimization for all disciplines is very important and applicable. Optimization has played a key role in practical engineering problems. A novel hybrid meta-heuristic optimization algorithm that is based on Differential Evolution (DE), Gradient Evolution (GE) and Jumping Technique named Differential Gradient Evolution Plus (DGE+) are presented in this paper. The proposed algorithm hybridizes the above-mentioned algorithms with the help of an improvised dynamic probability distribution, additionally provides a new shake off method to avoid premature convergence towards local minima. To evaluate the efficiency, robustness, and reliability of DGE+ it has been applied on seven benchmark constraint problems, the results of comparison revealed that the proposed algorithm can provide very compact, competitive and promising performance.

1. Introduction

Optimization is the best fit solution for all possible solutions to a given problem. Many modern optimization approaches fail to solve complex problems. Several researchers then started proposing new approaches to solve complex optimization problems in reasonable time and cost. There are two groups for optimizing methods: deterministic algorithms and stochastic algorithms [2]. If the same initial values are used, Deterministic methods may obtain the same results. Such algorithms have good efficacy for certain problems, but for all forms of optimization problems, it is difficult to generalize them [3]. One disadvantage of these search algorithms, they can simply be trapped in the local optimum [4]. For their strategies, stochastic algorithms usually use some randomness and avoid striking at a local optimum. Although they can have high-quality solutions in a reasonable amount of time for hard optimization problems, they do not ensure that the best solution will be found always.

The complexity of real-world problems has risen over the last few decades. To resolve these problems, a new meta-heuristic technique needs to be developed that is used to achieve optimal solutions with a low computational cost. Meta-heuristics are broadly divided into three categories: algorithms based on evolution theory, physical phenomena and swarm intelligence. A population-based meta-heuristic, inspired by the biological evolution based on mutation, reproduction, selection, and recombination.

*Corresponding author

Algorithms derived from physical phenomena are the second category. In these algorithms, search agents will move around the search space according to the rules of gravity, inertia, and electromagnetism. The final class is swarming intelligence algorithms that are based on social creatures' collective behavior. There are also other meta-heuristic approaches influenced by human behavior. Modern meta-heuristic algorithms having two main components, exploration and exploitation [5, 6]. Exploration makes sure the algorithm hits various promising search space regions while exploitation concentrating on the local area's search [7]. To achieve optimal solutions, both components must be optimized. Schematic view of the classification of the meta-heuristic algorithms is as follows:

Evolutionary Algorithms: Biogeography Based Optimizer [8], Differential Evolution [9], Evolution Strategy [10], Genetic Algorithms [11], Genetic Programming [12].

Physics-Based Algorithms: Artificial Chemical Reaction Optimization Algorithm [13], Big-Bang Big Crunch [14], Gravitational Search Algorithm [15], Ray Optimization Algorithm [16], Simulated Annealing [17], Small-World Optimization Algorithm [18], Nonlinear Optimization Algorithm [19,20], Constrained Optimization Problem [21], Fractional Gradient Based Algorithm [22], Optimization Problems Based on Hyperbolic Penalty Dynamic Framework [23], Jaya Optimization Algorithm [24,25], Feedback Controller Algorithm [26].

Swarm-Based Algorithms: Ant Colony Optimization [27], Bat-Inspired Algorithm [28], Bee Collecting Pollen Algorithm [29], Cuckoo Search [30], Particle Swarm Optimization [31].

Human Behaviors-Based Algorithms: Colliding Bodies Optimization [32], Mine Blast Algorithm [33], Seeker Optimization Algorithm [34], Soccer League Competition Algorithm [35], Social-Based Algorithm [36].

Differential Evolution (*DE*) is one of Price and Storn's most suitable and commonly used evolutionary algorithms [9]. Several methodologies were suggested and used to solve the various optimization problems in literature with the classic *DE* algorithm, such as Adaptive Chaotic *DE* [37], Adaptive Hybrid *DE* [38], *DE* with Ant Colony Optimization [39], *DE* with Firefly Algorithm [40], Modified Teaching–Learning Algorithm [41], Hybrid differential evolution with biogeography-based optimization [42].

The system for gradient evolution uses a series of vectors and consists of three main steps: updating, jumping and refreshing the search space. The major rule for gradient evolution is vector updating. Using the Newton–Raphson method search direction has been determined. The jumping and refreshing vector system allows local optima to be avoided [43]. This concept is based on gradient-based methods of search, such as the newton method, the conjugate direction and the Quasi-Newton method [44].

This paper introduces a new metaheuristic algorithm to optimize unconstrained and chemical design problems. The main characteristic of this paper are as follows: 1) A novel hybrid meta-heuristic optimization algorithm based on local and global search. This algorithm is the best combination of exploration and exploitation. 2) The proposed hybridized algorithm works with the help of an improvised dynamic probability distribution. 3) Additionally, it provides a novel shake off method to avoid premature convergence towards local minima. 4) It has been applied on several benchmark unconstraint problems and four complex practical engineering problems to evaluate the efficiency of proposed algorithm. The remaining of this paper is organized as follows: in section 2, the comprehensive detail of Differential Evolution and Gradient Evolution. In section 3, the proposed DGE+ and the concepts behind it are introduced in details. In section 4, the performance of the proposed optimizer is validated on different constrained optimization problems. Finally, conclusions and future directions are given in section 5.

2. Conventional algorithms

2.1. Differential evolution algorithm

Differential evolution is a relatively efficient meta-heuristic technique designed to optimize existing problems. Through applying mutation, crossover and selection operators, the population is successively improved over generations to achieve an optimal solution [45, 46]. The comprehensive detail of *DE* is present in [9, 47] and the main steps of the *DE* algorithm are given below in the form of a self-explanatory flow diagram shown in Figure 1.

2.2. Gradient evolution algorithm

Gradient evolution (*GE*) is an optimization algorithm based on the concept of gradients. The vector updating operator was driven from the Tylor series expansion and transforms the updating law for population-based search. The vector jumping operator prevents local optima and the vector refreshing operator is implemented in multiple iterations when a vector cannot move to a different location. The detail of this idea and the mathematical formulation of the *GE* algorithm is in [43, 48] the main steps of the *GE* algorithm are given below in the form of the self-explanatory flow diagram shown in Figure 2.

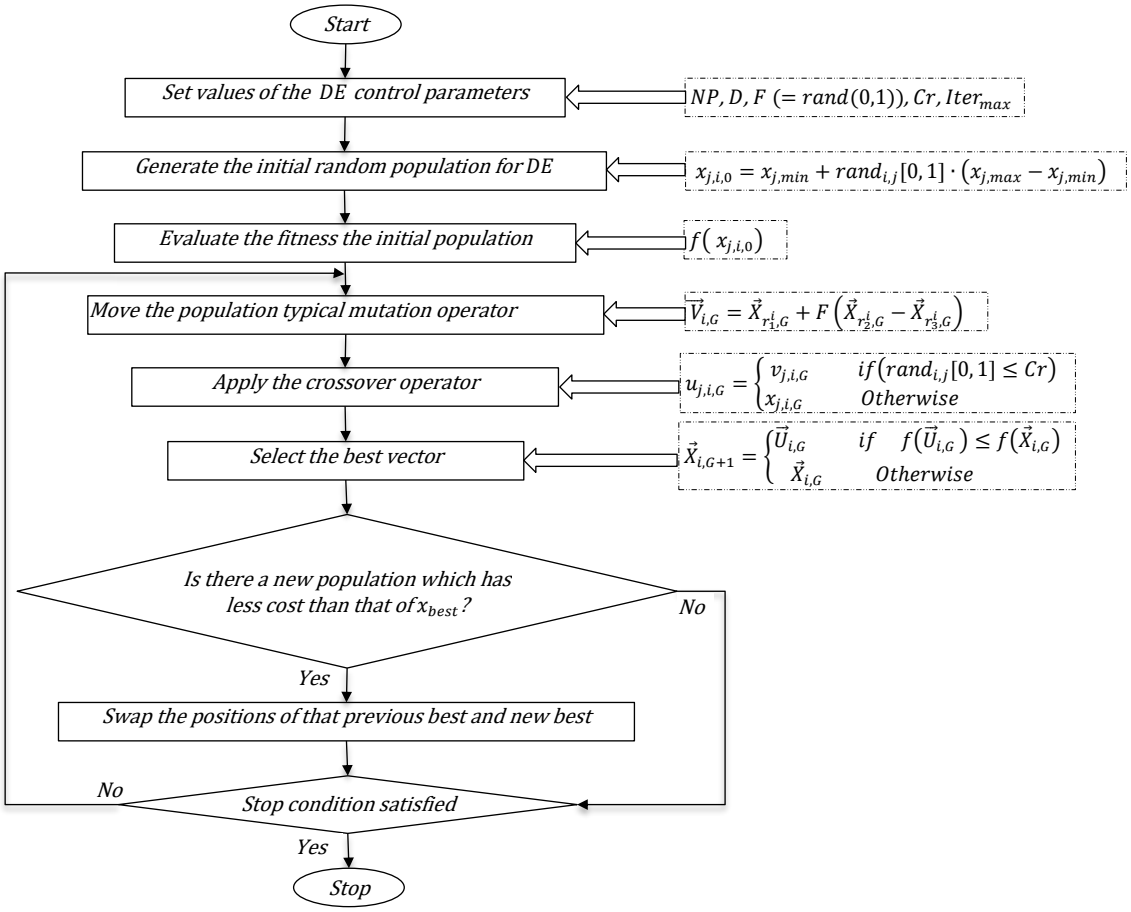


Figure 1. Flowchart for differential evolution

3. Differential gradient evolution plus

Differential Evolution is a powerful search technique to solve optimization problems with non-discreet variables. Differential Evolution is known for its excellent coverage of global search space and its tendency to find optimum solutions in higher dimensional optimization problems. On the other hand Gradient Evolution (*GE*) is a well-known technique that converges towards local minima by the use of instantaneous gradient information. In this way, *GE* is an effective method to explore local search space. The proposed algorithm hybridizes the above-mentioned algorithms with the help of an improvised dynamic probability distribution. The proposed algorithm additionally provides a new shake off' method to avoid premature convergence towards local minima. In this proposed method, the best solution of the last generation is maintained as a solution vector Y, this vector Y is used in the differential algorithm to generate new solutions. The proposed algorithm constantly monitors the best solution produced in each completed generation and if no significant improvement against best solution Y of previous generations is observed over a specified number of generations then a shake-off sequence is initiated which slightly changes the position of Y in solution space. In this way, the search direction of all individual members of the population is changed which results in an increased probability of escaping local minima and finding the optimum solution. During the search, best solution found in any iteration is preserved and reported after the search. Combination of these three above mentioned techniques resulted in a novel algorithm, named *DGE+* ($DE = Differential\ Evolution, GE = Gradient\ Evolution$ and $+ = Jumping\ Technique$), to solve unconstrained and constrained problems of any size and complexity. Each solution is represented with the symbol ${}^tX_i$, where $t = 1, 2, 3, \ldots, G_N$ and $i = 1, 2, 3, \ldots, P_s$ denotes generation and iteration respectively. Here G_N and P_s are user parameters which specify the total number of generation to be run and population size respectively.

$${}^tX_i = x_m, \text{ where } m = 1, 2, 3, \ldots, D. \quad (1)$$

In the above equation, x represents values of variables and D is the dimensions of search space and it is equal to the numbers of independent variables of the problem to

be solved. The proposed algorithm starts with the initialization of the population with random values of independent variables. Each solution vector is initialized randomly by using the following formula;

$$X = \{LB + random(0 \cdots 1) \times (UB - LB)\}, \quad (2)$$

where LB and UB are lower and upper bounds of the particular variable in specified problem and $random$ number is generated between 0 and 1. This formula ensures uniform distribution of initial values of variables within upper and lower bounds which results in no need for any repair strategy.

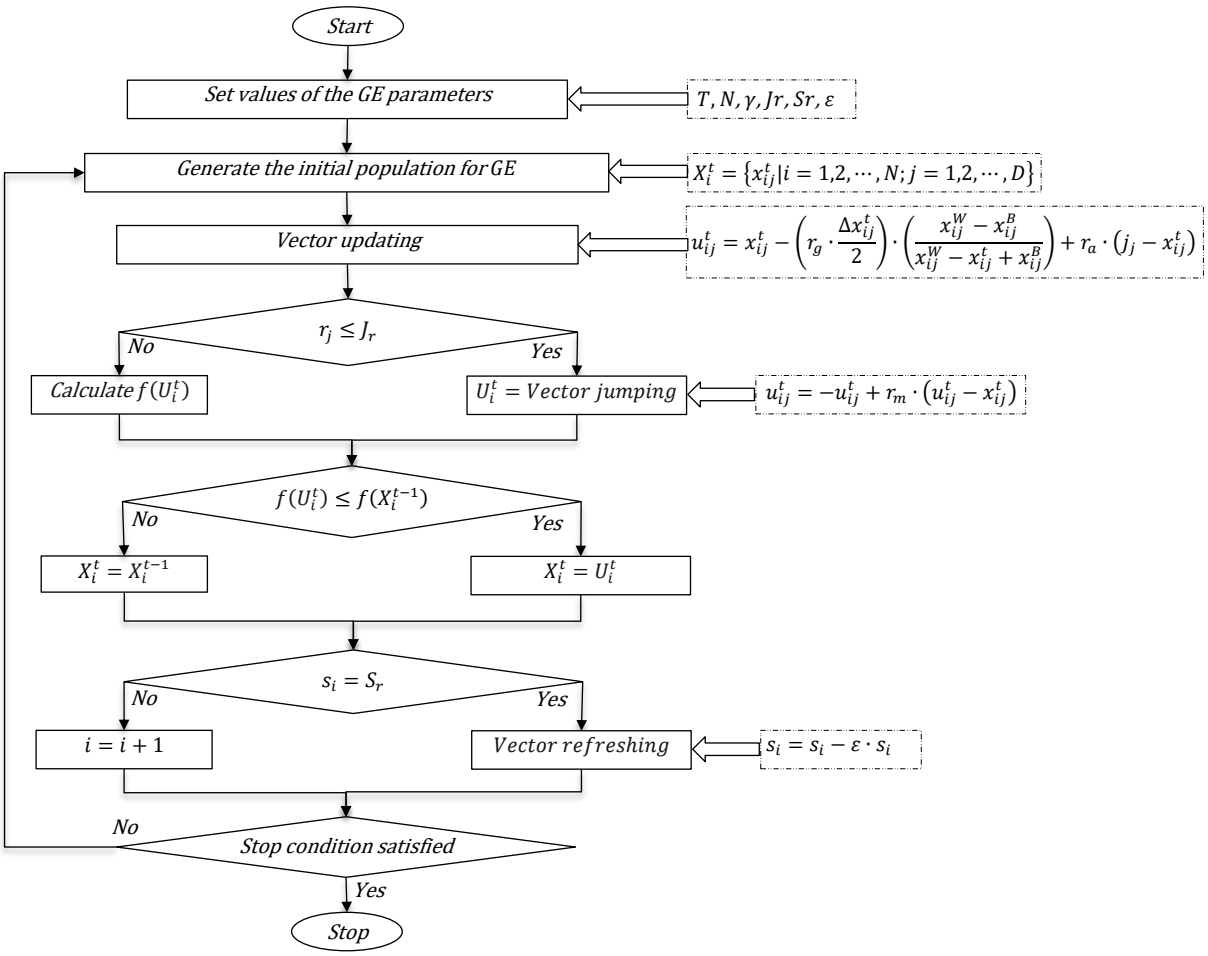


Figure 2. Flowchart for gradient evolution

After initialization, the complete population is evaluated for objective and constraints functions. At this stage, a solution vector Y is selected which is currently the best solution of this initial population. This initial population is then fed to the main body of the search loops. The new solutions are built using DE or GE, the selection of the algorithm to be used is dependent upon the following formula given in Eq. (3). In the following equation ${}^tU_i$ the new solution generated by the application of DE or GE at i^{th} iteration of t^{th} generation.

$${}^tU_i = \begin{cases} DE({}^tP), & if\ random\ (0 \cdots 1) > \frac{S_P}{G_N} \times t \\ GE({}^tP), & else \end{cases} \quad (3)$$

Algorithm selection probability of user parameter is represented by S_P. If differential evolution is to be used for the generation of new solutions then the following formula is used:

$${}^tU_i = DE({}^tP) = Y + S_F(X_{r_1} - X_{r_2}) + S_F(X_{r_3} - X_{r_4}), \quad (4)$$

where S_F is scaling factor and r_1, r_2, r_3 & r_4 are random integer numbers and their values range between 1 to P_s,

such that $r_1 \neq r_2 \neq r_3 \neq r_4$. In case when a new solution is to be generated by the use of gradient evolution following formula is used:

$$\delta_x = \frac{\gamma + \left| {}^{t}X_i - {}^{t}X_{i-1} \right|}{2} \quad (5)$$

$$b = {}^{t}X_i - \delta_x \quad (6)$$

$$w = {}^{t}X_i + \delta_x \quad (7)$$

$${}^{t}U_i = \begin{cases} {}^{t}X_i - \frac{(rand \times \delta_x)}{2}\left(\frac{{}^{t}X_{i+1} - b}{{}^{t}X_{i+1} - {}^{t}X_i + b}\right), & if\ i = 1 \\ {}^{t}X_i - \frac{(rand \times \delta_x)}{2}\left(\frac{w - {}^{t}X_{i-1}}{w - {}^{t}X_i + {}^{t}X_{i-1}}\right), & if\ i = P_S \\ {}^{t}X_i - \frac{(rand \times \delta_x)}{2}\left(\frac{{}^{t}X_{i+1} - {}^{t}X_{i-1}}{{}^{t}X_{i+1} - {}^{t}X_i + {}^{t}X_{i-1}}\right), & otherwise \end{cases} \quad (8)$$

In the above expressions, gamma γ is a gradient evolution user parameter. The newly generated solution ${}^{t}U_i$ is compared with the available solution at i^{th} location of the current population, if this solution is found better then this solution is inserted into the population at i^{th} location. Additionally, this algorithm allows acceptance of solutions with poorer performance into the main population to maintain diversity. This insertion probability of poorer solution is depended on a user control parameter A_R. A random number is generated between 0 and 1 if this number is less than A_R then the poorer solution is accepted in the main population.

To maintain diversity in population, fresh vectors are regularly inserted into the main population. The rate of insertion of a new random vector in population is dependent upon a parameter R_R. After scanning all the members of the population, existing solution vector Y is compared with the best solution of the current population, if this new best solution is better than Y then this new solution is selected as Y and a variable which track changes in Y is reset to 0. For every failed attempt to update Y, this variable is incremented by 1 and if its count becomes equal to user control parameter S_T then the value of current Y is shaken off randomly as per following equations;

$$d = Min\left(\left| {}^{t}X_i - UB \right|, \left| {}^{t}X_i - LB \right|\right), \quad (9)$$

$$Y = Y + d \times rand(-1 \cdots +1) \times \frac{G_N - t}{G_N}. \quad (10)$$

The above-mentioned cycles are repeated continuously for all generations and in the end, the best solution, which is preserved during the whole search, is reported as the solution to the given optimization problem.

3.1. Parameter selection

A wrong selection of algorithm parameters may result in a higher tendency to diverge, pre-mature convergence to a local minimum value, or undesired solutions. Therefore, the following considerations should be taken into account to fine-tune the algorithm parameters.

3.1.1. Population size $\mathbf{P_s}$

Optimization problems of low to medium complexity may require a population size of 30 to 50 individual solutions which are sufficient enough to solve the problem optimally. For the problem with a higher number of dimensions more individual members may be required to maintain diversity and room to explore global solution space. But on the other hand, larger population size results in higher computation time and increased number function evaluations. The benchmark problem set, selected for this study, of constrained and unconstrained problems contain optimization problems from low to high complexity. The experiments on the proposed algorithm show that $P_s = 50$ is sufficient enough to solve the entire problem set with excellent solution quality and in reasonable computational time.

3.1.2. Number of generations $\mathbf{G_N}$

The number of generations required to solve a problem optimally is directly proportional to the number of independent variables of the optimization problem. A lower value of the G_N produces non-optimal solutions and an unrealistically high value of G_N results in unnecessary high computational cost. The experiments on the proposed algorithm show that for unconstraint problems with up to 10 variables $G_N = 6000$, up to 20 variables $G_N = 12000$ and up to 30 variables $G_N = 20000$ is sufficient to produce optimal results. For constrained problems $G_N = 600$ is sufficient to solve all the selected Problems with excellent optimal values of objective functions.

3.1.3. Gradient evolution parameter gamma $\boldsymbol{\gamma}$

This parameter is used to control the performance of the gradient evolution part of the proposed algorithm. This number ensures that the value of change in any variable is non-zero; a zero value may lead to stagnation at the same point in solution space. The experiments on the proposed algorithm show that the complexity of the problem does not affect the value of this variable and for the chosen set of constrained and unconstrained problems $\gamma = 0.4$ has produced optimal results.

3.1.4. Differential evolution parameter scale factor $\mathbf{S_F}$

This parameter acts as a control of acceleration of convergence and has the most prominent effect on the performance of the differential evolution algorithm. The value of this parameter is dependent on the complexity of objective and constraint functions, a lower value of S_F, may result in non-optimal solutions due to the slower rate of convergence and conversely a higher value of S_F may cause DE to jump over optimal solutions in search space. The experiments with the proposed algorithm suggest

that for constrained problems $S_F = 0.5$ and for unconstrained problems $S_F = 0.48\ to\ 0.62$ has produced optimal results for all selected benchmark problems.

3.1.5. Differential evolution parameter crossover rate C_R

This parameter controls how much change, produced by DE should be passed on to the next generations. If the value of this parameter is set to a lower value then the convergence rate of the algorithm drops and vice versa. The value of this parameter should be set at a higher value to pass on the effect of DE to the next generations. The experiments on the proposed algorithm show that a value of $C_R = 0.91$ is good enough to produce optimal results for all selected benchmark constrained and unconstrained optimization problems.

3.1.6. Selection probability S_P

The proposed algorithm uses a differential evolution algorithm to explore (global search) and gradient evolution to exploit (local search) the given search space of the optimization problem. The decision when to use DE or GE is made by a dynamic probability function. At the start of the search, the probability of usage of DE is maximum and as the generations go the probability of DE usage drops and the probability of GE usage increases. In other words, in the beginning, more resources are utilized to perform a global search and in the end, relatively more computation is performed for local search. This dynamic probability distribution is controlled by the parameter S_P. A un-optimized low value of S_P usually causes less exploitation of local search space which results in poorer solution quality and a un-optimized higher value of S_P causes less exploration of global search space which in turn results in premature convergence to local minima. As both of these scenarios are undesirable therefore the value of this variable should be chosen carefully. The experiments conducted on all the constrained and unconstrained problems shows that $S_P = 0.2$ is good value to solve the entire set of benchmark problems optimally. This value $S_P = 0.2$ results in usage probability GE to increase from 0 to 20%, and consequentially the usage of DE drops from 100% to 80% during execution.

3.1.7. Sub-optimal solution acceptance rate A_R

All the new solutions which are produced either by DE or GE are tested for fitness against the corresponding member of the current population. If this new solution is better than the existing solution in the current population then this member of the population is killed and replaced by the newly generated solution. The proposed algorithm additionally allows for the acceptance of poorer solutions with a probability of A_R. This additional feature of the proposed algorithm maintains diversity in future populations and increases the probability of escaping local minima. The value of this parameter should be chosen carefully, in the case when the value of this parameter is set too high then the quality of search degrade and algorithm does not converge to the optimal values. The experiments on the proposed algorithm suggest that A_R between 0.01 and 0.05 is a good value to produce statistically better results in comparison to $A_R = 0$ for the given set of constrained and unconstrained problems.

3.1.8. Refresh rate R_R

For all population base algorithms regular supply of new individual solutions is essential to preserve diversity which in turn results in better solution quality. This fresh supply of new random solutions is controlled by R_R. A lower value of this parameter R_R causes the loss of diversity and poorer solution quality and a higher value of this parameter results in loss of better solutions and divergence of the optimization algorithm. The experiments with the proposed algorithm demonstrate that $R_R = 0.02$ is a decent value to solve the entire benchmark set of constrained and unconstrained problems.

3.1.9. Shake off threshold S_T

As an attempt to escape from local minima this proposed algorithm provides a shake off technique. The algorithm keeps monitoring the best solution of every subsequent generation and if no new improvement is observed then a counter is incremented by one. If the value of this variable becomes equal to shake off threshold S_T then shake off is initiated. A un-optimized high value of this threshold S_T will make this shake off ineffective and in contrast a low value of this parameter will result in poorer solution quality. The experiments conducted on our proposed algorithm indicates that the value of $S_T = 500$ and $S_T = 60$ for all unconstrained and constrained problems respectively can provide optimal results.

3.2. Constraint handling

Constraint handling of the problem is done as per rules given by Mottos & Coello [49]. The following four rules are used:

3.2.1. Rule 1

Whatever the value of the objective function is any feasible solution will always be preferred over infeasible solutions.

3.2.2. Rule 2

Infeasible solutions having a slight violation of 0.001 are considered as feasible solutions.

3.2.3. Rule 3

If two solutions are feasible then the one with better objective function value will be preferred.

3.2.4. Rule 4

If two solutions are infeasible then the one with less violation of feasibility will be preferred.

By incorporating first and fourth rules, the search is guided towards feasible regions rather than wasting resources by exploring infeasible regions of search space, the third rule forces the algorithm to both keep the search within the feasible regions and attempt to find a solution with a better value of objective function [49]. If the optimal solution lies near the boundary of the feasible region then the second rule facilitates the search of boundaries of the feasible region [50]. The algorithm of *DGE+* is as follows:

Algorithm: Differential Gradient Evolution Plus

Step 1: Initialize population
Step 2: Calculate objective and constraint functions
Step 3: Select Y which is the best solution in the current population
Step 4: Check the current generation is equal to G_N if yes then go to step 11. Otherwise, go to step 5
Step 5: Check current iteration is equal to P_s if yes then go to step 9. Otherwise, go to step 6
Step 6: Calculate U by using equations 3-8
Step 7: Evaluate U if it is acceptable then replace current solution of the population with this new solution U
Step 8: Go to step 5
Step 9: Check for shake off conditions, if true then change Y as per equations 9 and10
Step 10: Go to step 4
Step 11: Report the best solution and stop

The detail of the idea and the mathematical formulation of the *DGE+* algorithm is in the last section, the main steps of the *DGE+* algorithm are given below in the form of the self-explanatory flow diagram shown in Figure 3.

4. Experiments on constrained optimization problems

The comparison of the results produced by each constraint problem has been reported and listed in Table 1 which provided the comparative methods with references.

4.1. Experimental setup

The performance of the proposed novel and dynamic algorithm (*DGE+*) is exhibited by solving several optimization problems that are widely used to test optimization methods and considered as the benchmark problems in the literature. These test cases consist of seven benchmark constraint test problems [33]. All analyses are implemented in Matlab® environment on the computer equipped with the Intel CORE i5 @ 1.8 GHz CPU and 4 GB of RAM. The parameter settings of the proposed algorithm are:

Number of runs are 30, population size is 50, generations are 600, gamma value is 0.4, scale factor is 0.5 and cross over is 0.91. In the following subsections, *DGE+* is implemented on seven benchmark constraint problems and eight complex practical engineering problems.

4.2. Constrained optimization problems

4.2.1. Constrained problem 1

Braken and McCormick [84] originally introduced this problem which is a relatively simple constrained problem of minimization, having two variables and two constraints, one is equality constraint and the other is inequality constraint.

$$\min\ f(x) = (x_1 - 2)^2 + (x_2 - 1)^2$$

$$subject\ to \begin{cases} h_1(x) = x_1 - 2x_2 + 1 = 0 \\ h_2(x) = -\left(\frac{{x_1}^2}{4}\right) - {x_2}^2 + 1 \geq 0 \end{cases}$$

$$-10 \leq x_{1,} x_2 \leq 10$$

Table 2 demonstrates the comparison of the best solution among the different optimizers and the corresponding design variables. The results obtained by *DGE+* are compared with 4 state-of-the-art algorithms that are abbreviated and listed in Table 1.

Evolutionary programming violets both the constraints and remaining methods violet first constraint for the final solution but *DGE+* satisfies all constraints for the final solution. It is evident from Table 2 that the proposed *DGE+* algorithm performed better and superior to all the state-of-the-art methods without any violation.

The convergence curve shows the function values versus the number of generations for the constrained problem 1. The 30 trials of the best solution obtained from the *DGE+* algorithm are given in Figure 4.

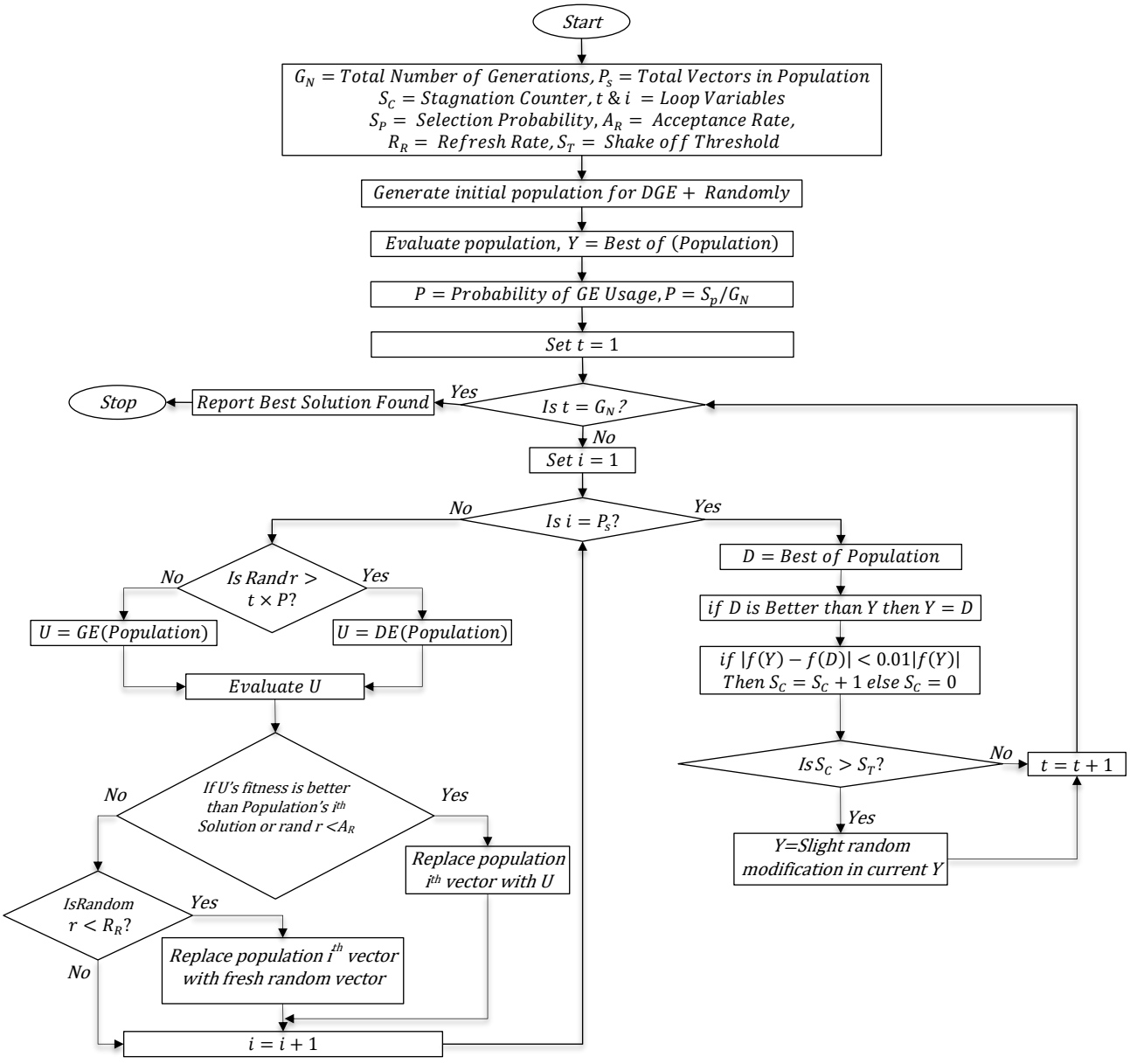


Figure 3. Flowchart for differential gradient evolution plus

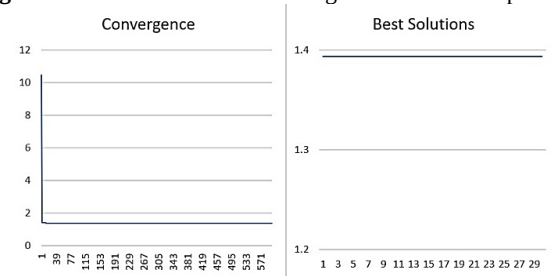


Figure 4. Convergence curve and 30 best solutions for constraint problem 1

Table 1. Comparative algorithms with references

Key	Algorithm Name	Key	Algorithm Name
MBA [33]	Mine Blast Algorithm	HM [51]	Homomorphous Mappings
ISR [52]	Improved Stochastic Ranking	HPSO [53]	Hybrid Particle Swarm Optimization
ABC [54, 55]	Artificial Bee Colony	HS [56, 57]	Harmony Search
IGA [58]	Interactive Genetic Algorithm	CRGA [59]	Changing Range Genetic Algorithm
ASCHEA [60]	Adaptive Segregational Constraint Handling Evolutionary Algorithm	CPSO-GD [61]	Co-evolutionary Particle Swarm Optimization Using Gaussian Distribution
CAEP [62]	Cultural Algorithm using Evolutionary Programming	Co-DE [63]	Effective Co-Evolutionary Differential Evolution
NM-PSO [64]	Nelder-Mead Particle Swarm Optimization	PSO [53]	Particle Swarm Optimization
CULDE [65]	Cultured Differential Evolution	PESO [66]	Particle Evolutionary Swarm Optimization
SAPF [67]	Self-Adaptive Penalty Function	EP [67]	Evolutionary Programming
DE [68]	Differential Evolution	GA [69-71]	Genetic Algorithms
DEDS [73]	Differential Evolution with Dynamic Stochastic	DELC [74]	Differential Evolution with Level Comparison
FSA [75]	Filter Simulated Annealing	SR [52]	Stochastic Ranking
GA with TS, PS [75]	Efficient Constraint Handling Method For Genetic Algorithms	α-Simples [77]	A Constrained Method
GA1 [76]	Genetic Algorithms 1	SMES [78]	Simple Multi-membered Evolution Strategy
GA2 [79]	Genetic Algorithms 2	TLBO [80]	Teaching-Learning-Based Optimization
HEAA [81]	Hybrid Evolutionary Algorithm and Adaptive technique	PSO-DE [82] [83]	Particle Swarm Optimization with Differential Evolution

Table 2. Reported results for constrained problem 1 from different optimizers

$\boldsymbol{Methods}$	$\boldsymbol{Design\ variables}$		$\boldsymbol{f(x)}$	$\boldsymbol{Constraints}$	
	$\boldsymbol{x_1}$	$\boldsymbol{x_2}$		$\boldsymbol{h_1(x)}$	$\boldsymbol{h_2(x)}$
HS	0.8343	0.9121	1.3770	$5E-03$	$5.4E-03$
GA	0.8080	0.8854	1.4339	$3.7E-02$	$5.2E-02$
MBA	0.822875	0.911437	1.3934649	$1.11E-06$	0
EP	0.8350	0.9125	1.3772	$1.0E-02$	$-7.0E-02$
$DGE+$	$\mathbf{0.822875656}$	$\mathbf{0.911437828}$	$\mathbf{1.393464981}$	**0**	**0**

4.2.2. Constrained problem 2

This problem is taken from [33] which is a relatively simple constrained problem of minimization having two variables and one equality constraint.

$\min f(x) = {x_1}^2 + (x_2 - 1)^2$

$subject\ to \{h(x) = x_2 - {x_1}^2 = 0,$

$-1 \leq x_1, x_2 \leq 1.$

Table 3 demonstrates the comparison of the best solution among the different optimizers and the corresponding design variables. *CULDE*, *SAPF*, *PSO-DE*, and *MBA* violates the constraint but *DGE+* satisfies constraint for the final solution. The results obtained by $DGE+$ are also compared with 10 state-of-the-art algorithms that are abbreviated and listed in Table 1. The comparison of statistical results for constrained problem 2 is given in Table 4. It is evident from Tables 3 and 4 that the proposed *DGE+* algorithm performed better and superior to all the state-of-the-art methods without any violation.

Table 3. Reported results for constrained problem 2 from different optimizers

$\boldsymbol{Methods}$	$\boldsymbol{Design\ variables}$		$\boldsymbol{f(x)}$	$\boldsymbol{Constraint}$
	$\boldsymbol{x_1}$	$\boldsymbol{x_2}$		$\boldsymbol{h(x)}$
$PSO-DE$	-0.7069	0.49975	0.749957673	$4.2E-05$
$CULDE$	-0.707036	0.5	0.749899905	0.0001
$SAPF$	-0.706	0.4996	0.74883616	0.00116
MBA	-0.706958	0.49979	0.749999658	$3.9E-07$
$\boldsymbol{DGE+}$	$\mathbf{-0.707106782}$	$\mathbf{0.5}$	$\mathbf{0.75}$	**0**

Table 4. Statistical comparison of results for constrained problem 2 of various algorithms

Method	***Worst***	***Mean***	***Best***	***SD***
HM	0.75	0.75	0.75	$N.A$
$ASCHEA$	$N.A$	0.75	0.75	$N.A$
$CRGA$	0.757	0.752	0.750	$2.5E-03$
$SMES$	0.75	0.75	0.75	$1.52E-04$
PSO	0.998823	0.860530	0.750000	$8.4E-02$
SR	0.750	0.750	0.750	$8E-05$
$DELC$	0.750	0.750	0.750	0
$HEAA$	0.750	0.750	0.750	$3.4E-16$
ISR	0.750	0.750	0.750	$1.1E-16$
ABC	0.75	0.75	0.75	0
$\boldsymbol{DGE+}$	$\mathbf{0.75}$	$\mathbf{0.75}$	$\mathbf{0.75}$	$\mathbf{0}$

"N.A" means not available.

The convergence curve shows the function values versus the number of generations for the constrained problem 2. The 30 trials of the best solution obtained from the *DGE+* algorithm are given in Figure 5.

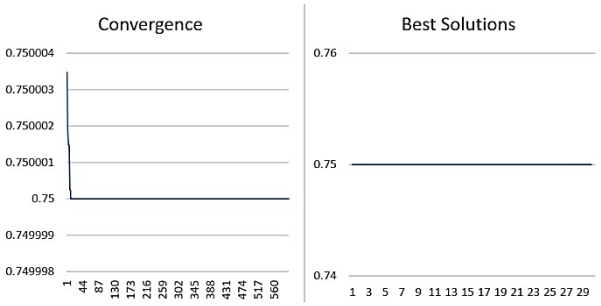


Figure 5. Convergence curve and 30 best solutions for constraint problem 2

4.2.3. Constrained problem 3

This problem is taken from [33] which is a relatively simple constrained problem of minimization having two variables and two inequality constraints.

$$min\ f(x) = ({x_1}^2 + x_2 - 11)^2 + (x_1 + {x_2}^2 - 7)^2$$

$$subject\ to \begin{cases} h_1(x) = 4.84 - (x_1 - 0.05)^2 - (x_2 - 2.5)^2 \geq 0 \\ h_2(x) = {x_1}^2 + (x_2 - 2.5)^2 - 4.84 \geq 0 \end{cases}$$

$$0 \leq x_1, x_2 \leq 6.$$

Table 5 demonstrates the comparison of the best solution among the different optimizers and the corresponding design variables. The results obtained by *DGE+* are compared with 5 state-of-the-art algorithms that are abbreviated and listed in Table 1. Harmony search violets both the constraints and mine blast algorithm violet second constraint for the final solution but *DGE+* satisfies all constraints for the final solution.

Table 5. Reported results for constrained problem 3 from different optimizers

Methods	***Design variables***		$\boldsymbol{f(x)}$	***Constraints***	
	$\boldsymbol{x_1}$	$\boldsymbol{x_2}$		$\boldsymbol{h_1(x)}$	$\boldsymbol{h_2(x)}$
$GA\ with\ PS\ (R = 0.01)$	$N.A$	$N.A$	13.58958	$N.A$	$N.A$
$GA\ with\ PS\ (R = 1)$	$N.A$	$N.A$	13.59108	$N.A$	$N.A$
$GA\ with\ TS$	2.246826	2.381865	13.59085	$N.A$	$N.A$
HS	2.24684	2.382136	13.590845	$-2.09E-06$	-0.222181
MBA	2.246833	2.381997	13.590842	0	-0.222183
$\boldsymbol{DGE+}$	$\mathbf{2.246825837}$	$\mathbf{2.381863455}$	$\mathbf{13.59084169}$	$\mathbf{0.027912486}$	$\mathbf{0.222182584}$

It is evident from Table 5 that the proposed *DGE+* algorithm performed better and superior to all the state-of-the-art methods without any violation.

The convergence curve shows the function values versus the number of generations for the constrained problem 3. The 30 trials of the best solution obtained from the *DGE+* algorithm are given in Figure 6.

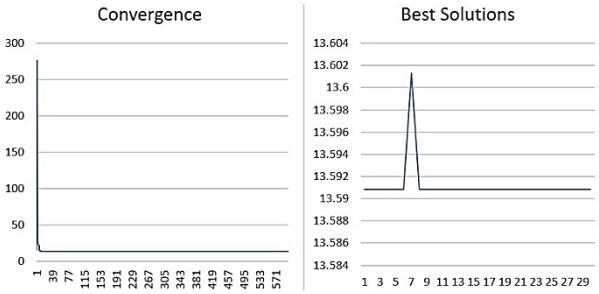


Figure 6. Convergence curve and 30 best solutions for constraint problem 3

4.2.4. Constrained problem 4

This problem taken from [33] which is a relatively simple constrained problem of minimization having two variables and two inequality constraints.

$$\min f(x) = -\frac{Sin^3(2\pi x_1)Sin(2\pi x_2)}{{x_1}^3(x_1+x_2)}$$

$$subject\ to \begin{cases} h_1(x) = {x_1}^2 - x_2 + 1 \le 0 \\ h_2(x) = 1 - x_1 + (x_2 - 4)^2 \le 0 \end{cases}$$

$$0 \le x_1, x_2 \le 10$$

Table 6. Reported result for constrained problem 4 from DGE+

Methods	***Design variables***		$f(x)$	***Constraints***	
	x_1	x_2		$h_1(x)$	$h_2(x)$
$DGE+$	1.227971353	4.245373367	−0.0958250	−1.737459724	−0.167763263

Table 7. Statistical comparison of results for constrained problem 4 of various algorithms

Method	***Worst***	***Mean***	***Best***	***SD***
HM	−0.0291438	−0.0891568	−0.0958250	$N.A$
$ASCHEA$	$N.A$	−0.095825	−0.095825	$N.A$
SR	−0.0958250	−0.0958250	−0.0958250	$2.6E-17$
$CAEP$	−0.0958250	−0.0958250	−0.0958250	0
DE	−0.0958250	−0.0958250	−0.0958250	$N.A$
$HPSO$	−0.0958250	−0.0958250	−0.0958250	$1.2E-10$
$NM-PSO$	−0.0958250	−0.0958250	−0.0958250	$3.5E-08$
$CRGA$	−0.095808	−0.095819	−0.095825	$4.40E-06$
$SAPF$	−0.092697	−0.095635	−0.095825	$1.055E-03$
GA	−0.0958250	−0.0958250	−0.0958250	$2.70E-09$
$SMES$	−0.095825	−0.095825	−0.095825	0
$CULDE$	−0.095825	−0.095825	−0.095825	$1E-07$
$DELC$	−0.095825	−0.095825	−0.095825	$1.0E-17$
$DEDS$	−0.095825	−0.095825	−0.095825	$4.0E-17$
$HEAA$	−0.095825	−0.095825	−0.095825	$2.8E-17$
ISR	−0.095825	−0.095825	−0.095825	$2.7E-17$
$Simplex$	−0.095825	−0.095825	−0.095825	$3.8E-13$
ABC	−0.0958250	−0.095825	−0.095825	0
MBA	−0.0958250	−0.0958250	−0.0958250	0
$DGE+$	**−0.093743605**	**−0.095748202**	**−0.0958250**	**0.00037334**

Table 6 represents the best solution and the value of corresponding design variables by using the *DGE+* algorithm. The results obtained by *DGE+* satisfies all constraints for the final solution, also compared with 19 state-of-the-art algorithms which are abbreviated and listed in Table 1.

It is evident from Table 7 that the proposed *DGE+* algorithm performed better and superior to all the state-of-the-art methods without any violation. The convergence curve shows the function values versus the number of generations for the constrained problem 4. The 30 trials of the best solution obtained from the *DGE+* algorithm are given in Figure 7.

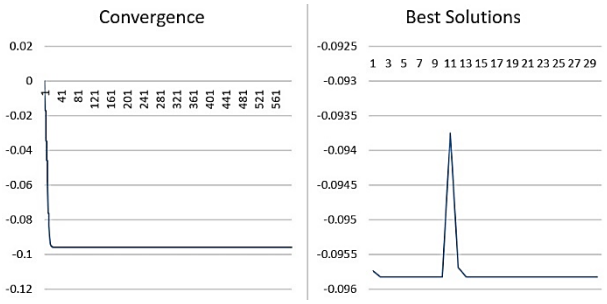


Figure 7. Convergence curve and 30 best solutions for constraint problem 4

4.2.5. Constrained problem 5

This problem is taken from [33] which is a relatively simple constrained problem of minimization having two variables and two inequality constraints.

$$\min f(x) = (x_1 - 10)^3 + (x_2 - 20)^3$$
$$subject\ to \begin{cases} h_1(x) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \geq 0 \\ h_2(x) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0 \end{cases}$$
$$13 \leq x_1 \leq 100, 0 \leq x_2 \leq 100.$$

Table 8. Reported result for constrained problem 5 from DEG+

Methods	***Design variables***		$f(x)$	***Constraints***	
	x_1	x_2		$h_1(x)$	$h_2(x)$
DGE +	14.095	0.842961	−6961.813644	165.4380518	$-1.75248E-06$

Table 9. Statistical comparison of results for constrained problem 5 of various algorithms

Method	***Worst***	***Mean***	***Best***	***SD***
HM	−5473.9	−6342.6	−6952.1	*N.A*
PSO − DE	−6961.81388	−6961.81388	−6961.81388	$2.3E-09$
ISR	−6961.814	−6961.814	−6961.814	$1.9E-12$
HEAA	−6961.814	−6961.814	−6961.814	$4.6E-12$
ABC	−6961.805	−6961.813	−6961.814	$2E-03$
FSA	−6961.8139	−6961.8139	−6961.8139	0
PSO	−6961.81381	−6961.81387	−6961.81388	$6.5E-06$
CRGA	−6077.123	−6740.288	−6956.251	$2.70E+2$
DEDS	−6961.814	−6961.814	−6961.814	0
MBA	−6961.813875	−6961.813875	−6961.813875	0
ASCHEA	*N.A*	−6961.81	−6961.81	*N.A*
SR	−6350.262	−6875.940	−6961.814	160
SMES	−6962.482	−6961.284	−6961.814	1.85
DELC	−6961.814	−6961.814	−6961.814	$7.3E-10$
SAPF	−6943.304	−6953.061	−6961.046	5.876
GA	−6961.8139	−6961.8139	−6961.8139	0
DE	−6961.814	−6961.814	−6961.81	*N.A*
CULDE	−6961.813876	−6961.813876	−6961.813876	$1E-07$
NM − PSO	−6961.8240	−6961.8240	−6961.8240	0
Simplex	−6961.814	−6961.814	−6961.814	$1.3E-10$
DGE +	**−6961.813894**	**−6961.813894**	**−6961.813894**	**0**

Table 8 represents the best solution and the value of corresponding design variables by using the *DGE+* algorithm. The results obtained by *DGE+* satisfies all constraints for the final solution, also compared with 21 state-of-the-art algorithms which are abbreviated and listed in Table 1.

It is evident from Table 9 that the proposed *DGE+* algorithm performed better and superior to all the state-of-the-art methods without any violation. The convergence curve shows the function values versus the number of generations for the constrained problem 4. The 30 trials of the best solution obtained from the *DGE+* algorithm are given in Figure 8.

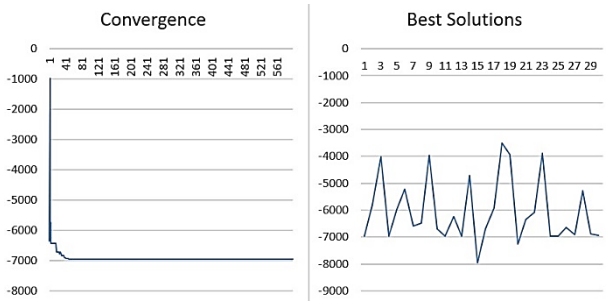


Figure 8. Convergence curve and 30 best solutions for constraint problem 5

4.2.6. Constrained problem 6

This problem is taken from [33] which is a relatively complex constrained problem of minimization having seven variables and four inequality constraints.

$$minf(x) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + {x_3}^4 + 3(x_4 - 11)^2 + 10{x_5}^6 + 7{x_6}^2 + {x_7}^4 - 4x_6x_7 - 10x_6 - 8x_7,$$

$$subject\ to$$
$$\begin{cases} h_1(x) = 127 - 2{x_1}^2 - 3{x_2}^4 - x_3 - 4{x_4}^2 - 5x_5 \geq 0, \\ h_2(x) = 282 - 7x_1 - 3x_2 - 10{x_3}^2 - x_4 + x_5 \geq 0, \\ h_3(x) = 196 - 23x_1 - {x_2}^2 - 6{x_6}^2 + 8x_7 \geq 0, \\ h_4(x) = -4{x_1}^2 - {x_2}^2 + 3x_1x_2 - 2{x_3}^2 - 5x_6 + 11x_7 \geq 0, \end{cases}$$
$$-10 \leq x_1, x_2, x_3, x_4, x_5, x_6, x_7 \leq 10.$$

Table 10 demonstrates the comparison of the best solution among the different optimizers and the corresponding design variables. The results obtained by *DGE+*satisfies all constraints for the final solution are compared with 25 state-of-the-art algorithms that are abbreviated and listed in Table 1.

Table 10. Reported results for constrained problem 6 from different optimizers

Methods	**Design variables**							$f(x)$
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
IGA	2.330499	1.951372	−0.477541	4.365726	−0.624487	1.038131	1.594227	680.63006
HS	2.323456	1.951242	−0.448467	4.361919	−0.630075	1.03866	1.605348	680.6413574
MBA	2.326585	1.950973	−0.497446	4.367508	−0.618578	1.043839	1.595928	680.6322202
$\boldsymbol{DGE+}$	**2.330404**	**1.95135**	**−0.47779**	**4.365786**	**−0.62427**	**1.038215**	**1.594204**	**680.63**

Table 11. Reported results for constrained problem 6 from different optimizers (continued)

Methods	$f(x)$	**Constraints**			
		$h_1(x)$	$h_2(x)$	$h_3(x)$	$h_4(x)$
IGA	680.63006	$4.46E-05$	252.561723	144.878190	$7.63E-06$
HS	680.6413574	0.208928	252.878859	145.123347	0.263414
MBA	680.6322202	$1.17E-04$	252.400363	144.912069	$1.39E-04$
$\boldsymbol{DGE+}$	**680.63**	$\mathbf{7.90E-8}$	**252.5603**	**144.8792**	$\mathbf{2.42E-07}$

Table 12. Statistical comparison of results for constrained problem 6 of various algorithms

Method	**Worst**	**Mean**	**Best**	**SD**
GA	680.6538	680.6381	680.6303	$6.61E-03$
$ASCHEA$	$N.A$	680.641	680.630	$N.A$
$CULDE$	680.630057	680.630057	680.630057	$1E-07$
$CRGA$	682.965	681.347	680.726	$5.70E-01$
$Simplex$	680.630	680.630	680.630	$2.9E-10$
HM	683.1800	681.1600	680.9100	$4.11E-02$
$GA1$	680.6508	680.6417	680.6344	$N.A$
MBA	680.7882	680.6620	680.6322	$3.30E-02$
$GA2$	$N.A$	$N.A$	680.642	$N.A$
$SAPF$	682.081	681.246	680.773	0.322
SR	680.763	680.656	680.63	0.034
HS	$N.A$	$N.A$	680.6413	$N.A$
DE	680.144	680.503	680.771	0.67098
IGA	680.6304	680.6302	680.6301	$1.00E-05$
PSO	684.5289146	680.9710606	680.6345517	$5.1E-01$
$CPSO-GD$	681.371	680.7810	680.678	0.1484
$SMES$	680.719	680.643	680.632	$1.55E-02$
$DELC$	680.630	680.630	680.630	$3.2E-12$
$DEDS$	680.630	680.630	680.630	$2.9E-13$
$HEAA$	680.630	680.630	680.630	$5.8E-13$
ISR	680.630	680.630	680.630	$3.2E-13$
$PESO$	680.630	680.630	680.631	$N.A$
$CoDE$	685.144	681.503	680.771	$N.A$
ABC	680.638	680.640	680.634	$4E-03$
$TLBO$	680.638	680.633	680.630	$N.A$
$\boldsymbol{DGE+}$	**680.6974951**	**680.6340181**	**680.63**	**0.012065102**

It is evident from Tables 10 & 11 that the proposed *DGE+* algorithm performed better and superior to all the state-of-the-art methods without any violation. The convergence curve shows the function values versus the number of generations for the constrained problem 1. The 30 trials of the best solution obtained from the *DGE+* algorithm are given in Figure 9.

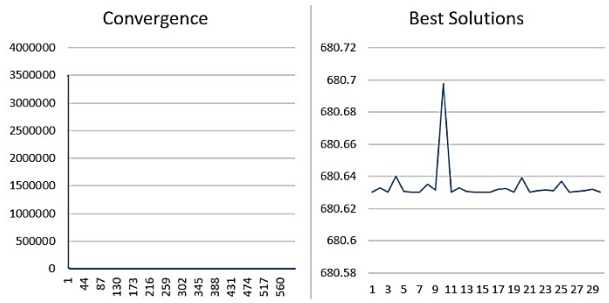


Figure 9. Convergence curve and 30 best solutions for constraint problem 6

4.2.7. Constrained problem 7

This problem is taken from [33] which is a relatively complex constrained problem of minimization having five variables and six inequality constraints. Table 12 demonstrates the comparison of the best solution among the different optimizers and the corresponding design variables. The results obtained by *DGE+* are compared with 5 state-of-the-art algorithms that are abbreviated and listed in Table 1. *CULDE*, Harmony search and *GA2* violet two constraints and remaining methods violet first constraint for the final solution but *DGE+* satisfies all constraints for the final solution.

$$\min f(x) = 5.3578547{x_3}^3 + 0.8356891\,x_1x_5 + 37.293239x_1 + 40729.141,$$

$$subject\ to \begin{cases} h_1(x) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \le 0, \\ h_2(x) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 - 0.0022053x_3x_5 \le 0, \\ h_3(x) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813{x_3}^2 - 110 \le 0, \\ h_4(x) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813{x_3}^2 + 90 \le 0, \\ h_5(x) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \le 0, \\ h_6\{x\} = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \le 0, \end{cases}$$

$$78 \le x_1 \le 102, 33 \le x_2 \le 45, 27 \le x_3, x_4, x_5 \le 45.$$

Table 13. Reported results for constrained problem 7 from different optimizers

Methods	***Design variables***					$f(x)$
	x_1	x_2	x_3	x_4	x_5	
CULDE	78.000000	33.000000	29.995256	45.000000	36.775813	−30665.5386
HS	78.0	33.0	29.995	45.0	36.776	−30665.500
GA1	80.39	35.07	32.05	40.33	33.34	−30005.700
GA2	78.0495	33.007	27.081	45.00	44.94	−31020.859
MBA	78.00000	33.00000	29.99526	44.99999	36.77581	−30665.5386
DGE +	**78**	**33**	**29.99525603**	**45**	**36.77581**	**−30665.5386**

Table 14. Reported results for constrained problem 7 from different optimizers (continued)

Methods	$f(x)$	***Constraints***					
		$h_1(x)$	$h_2(x)$	$h_3(x)$	$h_4(x)$	$h_5(x)$	$h_6(x)$
CULDE	−30665.5386	$1.35E-08$	−92.00000001	−11.15945	−8.840500	−4.999999	$4.12E-09$
HS	−30665.500	$4.34E-05$	−92.000043	−11.15949	−8.840510	−5.000064	$6.49E-05$
GA1	−30005.700	−0.343809	−91.656190	−10.463103	−9.536896	−4.974473	−0.025526
GA2	−31020.859	1.283813	−93.283813	−9.592143	−10.407856	−4.998088	$1.91E-03$
MBA	−30665.5386	$1.33E-08$	−91.99999	−11.159499	−8.84050	−4.99999	$-3.06E-09$
DGE +	**−30665.5386**	**0**	**−92**	**−11.15949969**	**−8.840500309**	**−5**	**0**

The results obtained by *DGE+* are also compared with 20 state-of-the-art algorithms, the comparison of statistical results for constrained problem 7 is given in Table 13.

It is evident from Table 12 & 13 that the proposed *DGE+* algorithm performed better and superior to all the state-of-the-art methods without any violation. The convergence curve shows the function values versus the number of generations for the constrained problem 1. The 30 trials of the best solution obtained from the *DGE+* algorithm are given in Figure 10.

Table 15. Statistical comparison of results for constrained problem 7 of various algorithms

Method	***Worst***	***Mean***	***Best***	***SD***
MBA	-30665.3300	-30665.5182	-30665.5386	$5.08E-02$
$ASCHEA$	$N.A$	-30665.5	-30665.5	$N.A$
SR	-30665.539	-30665.539	-30665.539	$2E-05$
ISR	-30665.539	-30665.539	-30665.539	$1.1E-11$
$CAEP$	-30662.200	-30662.500	-30665.500	9.3
$HEAA$	-30665.539	-30665.539	-30665.539	$7.4E-12$
$SAPF$	-30656.471	-30655.92	-30665.401	2.043
$HPSO$	-30665.539	-30665.539	-30665.539	$1.7E-06$
HS	$N.A$	$N.A$	-30665.500	$N.A$
DE	-30665.509	-30665.536	-30665.539	$5.067E-03$
$SMES$	-30665.539	-30665.539	-30665.539	0
$CRGA$	-30660.313	-30664.398	-30665.520	1.6
ABC	-30665.539	-30665.539	-30665.539	0
$CULDE$	-30665.5386	-30665.5386	-30665.5386	$1E-07$
$DEDS$	-30665.539	-30665.539	-30665.539	$2.7E-11$
$PSO-DE$	-30665.5387	-30665.5387	-30665.5387	$8.3E-10$
HM	-30645.900	-30665.300	-30664.500	$N.A$
$DELC$	-30665.539	-30665.539	-30665.539	$1.0E-11$
$Simplex$	-30665.539	-30665.539	-30665.539	$4.2E-11$
PSO	-30252.3258	-30570.9286	-30663.8563	81
$\boldsymbol{DGE+}$	$\mathbf{-30665.53823}$	$\mathbf{-30665.539}$	$\mathbf{-30665.53867}$	$\mathbf{9.26E-05}$

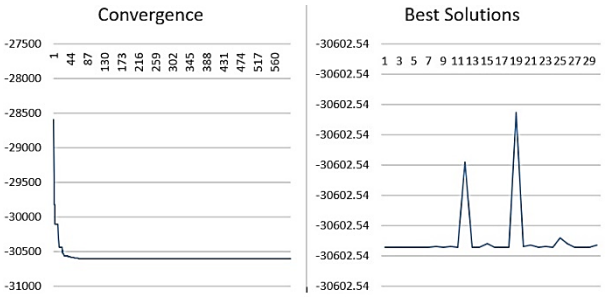


Figure 10. Convergence curve and 30 best solutions for constraint problem 7

5. Conclusions

A new hybrid meta-heuristic has been presented in this paper, called *DGE+*, for dealing with seven benchmark constraint optimization problems. The main motivation behind the present study is to combine the desirable explorative features of *DE* with exploitative features of *GE* algorithms. The proposed method is mainly based on Differential Evolution, Gradient Evolution, and novel jumping technique. The proposed algorithm hybridizes the above-mentioned algorithms with the help of an improvised dynamic probability distribution, additionally provides a new shake off method to avoid premature convergence towards local minima. To evaluate the efficiency and robustness of *DGE+* it has been applied on seven benchmark constraint optimization problems, the results of comparison revealed that *DGE+* can provide very compact, competitive and promising results. As future works, various research directions can be followed. Based on certain preliminary observations, the parameter values for *DGE+* are modified. A full sensitivity analysis on the impact of parameters may, therefore, be a guideline for future research. The implementation of the proposed algorithm to several real-world problems is also extremely valuable.

Acknowledgment

The authors thank the reviewers and editors for their useful comments, which led to the improvement of the content of the paper.

References

[1] Khalilpourazari, S. & Khalilpourazary. S. (2018). Optimization of production time in the multi-pass milling process via a robust grey wolf optimizer. *Neural Computing and Applications*. 29(12), 1321-1336.

[2] Yang, X.-S. (2010). *Nature-inspired metaheuristic algorithms*. Luniver press.

[3] Gandomi, A. H., Yang, X.-S. & Alavi, A. H. (2011). Mixed variable structural optimization using firefly algorithm. *Computers & Structures*. 89(23-24), 2325-2336.

[4] Zhang, L., et al. (2016). A novel hybrid firefly algorithm for global optimization. *PloS one*. 11(9), e0163230.

[5] Alba, E. & Dorronsoro, B. (2005). The exploration/exploitation tradeoff in dynamic cellular genetic algorithms. *IEEE transactions on evolutionary computation*. 9(2), 126-142.

[6] Olorunda, O. and Engelbrecht, A. P. (2008). *Measuring exploration/exploitation in particle swarms using swarm diversity*. in *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*.

[7] Lozano, M. & García-Martínez, C. (2010). Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification: Overview and progress report. *Computers & Operations Research*. 37(3), 481-497.

[8] Simon, D. (2008). Biogeography-based optimization. *IEEE transactions on evolutionary computation*. 12(6), 702-713.

[9] Storn, R. (1996). *On the usage of differential evolution for function optimization*. in *Proceedings of North American Fuzzy Information Processing*. IEEE.

[10] Beyer, H.-G. & Schwefel, H.-P. (2002). Evolution strategies–a comprehensive introduction. *Natural computing*. 1(1), 3-52.

[11] Bonabeau, E., Dorigo, M. & Theraulaz, G. (1999). *From natural to artificial swarm intelligence*. Oxford university press, UK.

[12] Koza, J.R. & J.R. Koza. (1992). *Genetic programming: On the programming of computers by means of natural selection*. MIT press.

[13] Alatas, B. (2011). Acroa: Artificial chemical reaction optimization algorithm for global optimization. *Expert Systems with Applications*. 38(10), 13170-13180.

[14] Erol, O. K. & I. Eksin. (2006). A new optimization method: Big bang–big crunch. *Advances in Engineering Software*. 37(2), 106-111.

[15] Rashedi, E., H. Nezamabadi-Pour, & S. Saryazdi. (2009). Gsa: A gravitational search algorithm. *Information sciences*. 179(13), 2232-2248.

[16] Kaveh, A. & M. Khayatazad. (2012). A new meta-heuristic method: Ray optimization. *Computers & Structures*. 112: p. 283-294.

[17] Kirkpatrick, S., C. D. Gelatt, & M. P. Vecchi. (1983). Optimization by simulated annealing. *science*. 220(4598), 671-680.

[18] Du, H., X. Wu, & J. Zhuang. (2006) *Small-world optimization algorithm for function optimization*. in *International Conference on Natural Computation*. Springer.

[19] Evirgen, F., & Yavuz, M. (2018). An alternative approach for nonlinear optimization problem with Caputo-Fabrizio derivative. *In ITM Web of Conferences* (Vol. 22, p. 01009). EDP Sciences.

[20] Evirgen, F., & Özdemir, N. (2012). A fractional order dynamical trajectory approach for optimization problem with HPM. *In Fractional Dynamics and Control* (pp. 145-155). Springer, New York, NY.

[21] Evirgen, F. (2017). Conformable Fractional Gradient Based Dynamic System for Constrained Optimization Problem. *Acta Physica Polonica A*, 132(3), 1066-1069.

[22] Evirgen, F. (2016). Analyze the optimal solutions of optimization problems by means of fractional gradient based system using VIM. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, 6(2), 75-83.

[23] Evirgen, F. (2017). Solution of a Class of Optimization Problems Based on Hyperbolic Penalty Dynamic Framework. *Acta Physica Polonica A*, 132(3), 1062-1065.

[24] Jumani, T. A., et al. (2020). Jaya optimization algorithm for transient response and stability enhancement of a fractional-order PID based automatic voltage regulator system. *Alexandria Engineering Journal*, 59(4), 2429-2440.

[25] Al-Dhaifallah, M., et al. (2018). Optimal parameter design of fractional order control based INC-MPPT for PV system. *Solar Energy*, 159, 650-664.

[26] Bitirgen, R., Hancer, M., & Bayezit, I. (2018). All Stabilizing State Feedback Controller for Inverted Pendulum Mechanism. *IFAC-PapersOnLine*, 51(4), 346-351.

[27] Stützle, T., et al. (2011). *Parameter adaptation in ant colony optimization*, in Autonomous search. Springer. 191-215.

[28] Yang, X.-S. (2010). *A new metaheuristic bat-inspired algorithm*, in Nature inspired cooperative strategies for optimization (nicso 2010). Springer. 65-74.

[29] Lu, X. and Y. Zhou. (2008). A novel global convergence algorithm: Bee collecting pollen algorithm. in *International Conference on Intelligent Computing*. 2008. Springer.

[30] Singh, H., et al. (2019). A reliable numerical algorithm for the fractional klein-gordon equation. *Engineering Transactions*. 67(1), 21–34.

[31] Kennedy, J. & R. Eberhart. *Particle swarm optimization (pso)*. in *Proc IEEE International Conference on Neural Networks, Perth, Australia*. 1995.

[32] Kaveh, A. & V. Mahdavi. (2014). Colliding bodies optimization: A novel meta-heuristic method. *Computers & Structures*. 139: p. 18-27.

[33] Sadollah, A., et al. (2013). Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems. *Applied Soft Computing*. 13(5), 2592-2612.

[34] Wang, B., C. Liu, & H. Wu. (2014). *The research of pattern synthesis of linear antenna array based on seeker optimization algorithm*. in *2014 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*. IEEE.

[35] He, S., Q. H. Wu, & J. Saunders. (2009). Group search optimizer: An optimization algorithm inspired by animal searching behavior. *IEEE transactions on evolutionary computation*. 13(5), 973-990.

[36] Ramezani, F. & Lotfi, S. (2013). Social-based algorithm (sba). *Applied Soft Computing*. 13(5), 2837-2856.

[37] Lu, Y., et al. (2010). An adaptive chaotic differential evolution for the short-term hydrothermal generation scheduling problem. *Energy Conversion and Management*. 51(7), 1481-1490.

[38] Lu, Y., et al. (2010). An adaptive hybrid differential evolution algorithm for dynamic economic dispatch with valve-point effects. *Expert Systems with Applications*. 37(7), 4842-4849.

[39] Chang, L., et al. (2012). A hybrid method based on differential evolution and continuous ant colony optimization and its application on wideband antenna design. *Progress in electromagnetics research*. 122: p. 105-118.

[40] Abdullah, A., et al. (2013). An evolutionary firefly algorithm for the estimation of nonlinear biological model parameters. *PloS one*. 8(3), e56310.

[41] Niknam, T., Azizipanah-Abarghooee, R. & Aghaei, J. (2012). A new modified teaching-learning algorithm for reserve constrained dynamic economic dispatch. *IEEE Transactions on power systems*. 28(2), 749-763.

[42] Bhattacharya, A. and Chattopadhyay, P. K. (2010). Hybrid differential evolution with biogeography-based optimization for solution of economic load dispatch. *IEEE Transactions on power systems*. 25(4), 1955-1964.

[43] Kuo, R. and Zulvia, F. E. (2015). The gradient evolution algorithm: A new metaheuristic. *Information Sciences*. 316: p. 246-265.

[44] Bazaraa, M. S., H. D. Sherali, & C. M. Shetty. (2013). *Nonlinear programming: Theory and algorithms*. John Wiley & Sons.

[45] Wang, S.-K., J.-P. Chiou, & C.-W. Liu. (2007). Non-smooth/non-convex economic dispatch by a novel hybrid differential evolution algorithm. *IET Generation, Transmission & Distribution*. 1(5), 793-803.

[46] Chiou, J.-P. (2007). Variable scaling hybrid differential evolution for large-scale economic dispatch problems. *Electric Power Systems Research*. 77(3-4), 212-218.

[47] Storn, R. & K. Price. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*. 11(4), 341-359.

[48] Kuo, R. & F. E. Zulvia. *Cluster analysis using a gradient evolution-based k-means algorithm*. in *2016 IEEE Congress on Evolutionary Computation (CEC)*. 2016. IEEE.

[49] Mezura-Montes, E. & C. A. C. Coello. (2008). An empirical study about the usefulness of evolution strategies to solve constrained optimization problems. *International Journal of General Systems*. 37(4), 443-473.

[50] Kaveh, A. & S. Talatahari. (2009). A particle swarm ant colony optimization for truss structures with discrete variables. *Journal of Constructional Steel Research*. 65(8-9), 1558-1568.

[51] Koziel, S. & Z. Michalewicz. (1999). Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. *Evolutionary computation*. 7(1), 19-44.

[52] Runarsson, T. P. & X. Yao. (2000). Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation*. 4(3), 284-294.

[53] Parsopoulos, K. E. & M. N. Vrahatis. (2005). *Unified particle swarm optimization for solving constrained engineering optimization problems*. in *International conference on natural computation*. Springer.

[54] Karaboga, D. & B. Basturk. (2007). *Artificial bee colony (abc) optimization algorithm for solving constrained optimization problems*. in *International fuzzy systems association world congress*. Springer.

[55] Akay, B. & D. Karaboga. (2012). Artificial bee colony algorithm for large-scale problems and engineering design optimization. *Journal of intelligent manufacturing*. 23(4), 1001-1014.

[56] Geem, Z. W., J. H. Kim, & G. V. Loganathan. (2001). A new heuristic optimization algorithm: Harmony search. *Simulation*. 76(2), 60-68.

[57] Lee, K. S. & Z. W. Geem. (2005). A new meta-heuristic algorithm for continuous engineering optimization: Harmony search theory and practice. *Computer Methods in Applied Mechanics and Engineering*. 194(36-38), 3902-3933.

[58] Farooq, H. & M. T. Siddique. (2014). A comparative study on user interfaces of interactive genetic algorithm. *Procedia Computer Science*. 32: p. 45-52.

[59] Amirjanov, A. (2008). Investigation of a changing range genetic algorithm in noisy environments. *International journal for numerical methods in engineering*. 73(1), 26-46.

[60] Hamida, S. B. & M. Schoenauer. (2002). *Aschea: New results using adaptive segregational constraint handling*. in *Proceedings of the 2002 Congress on Evolutionary Computation CEC'02 (Cat No 02TH8600)*. IEEE.

[61] Krohling, R. A. & L. dos Santos Coelho. (2006). Coevolutionary particle swarm optimization using gaussian distribution for solving constrained optimization problems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*. 36(6), 1407-1416.

[62] Coello Coello, C. A. & R. L. Becerra. (2004). Efficient evolutionary optimization through the use of a cultural algorithm. *Engineering Optimization*. 36(2), 219-236.

[63] Huang, F.-z., L. Wang, & Q. He. (2007). An effective co-evolutionary differential evolution for constrained optimization. *Applied Mathematics and Computation*. 186(1), 340-356.

[64] Zahara, E. & Y.-T. Kao. (2009). Hybrid nelder–mead simplex search and particle swarm optimization for constrained engineering design problems. *Expert Systems with Applications*. 36(2), 3880-3886.
[65] Becerra, R. L. & C. A. C. Coello. (2006). Cultured differential evolution for constrained optimization. *Computer Methods in Applied Mechanics and Engineering*. 195(33-36), 4303-4322.
[66] Muñoz Zavala, A. E., A. H. Aguirre, & E. R. Villa Diharce. (2005). *Constrained optimization via particle evolutionary swarm optimization algorithm (peso)*. in *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. ACM.
[67] Tessema, B. & G. G. Yen. (2006). *A self adaptive penalty function based algorithm for constrained optimization*. in *2006 IEEE International Conference on Evolutionary Computation*. IEEE.
[68] Lampinen, J. (2002). *A constraint handling approach for the differential evolution algorithm*. in *Proceedings of the 2002 Congress on Evolutionary Computation CEC'02 (Cat No 02TH8600)*. IEEE.
[69] Fogel, D. B. (1995). A comparison of evolutionary programming and genetic algorithms on selected constrained optimization problems. *Simulation*. 64(6), 397-404.
[70] Amirjanov, A. (2006). The development of a changing range genetic algorithm. *Computer Methods in Applied Mechanics and Engineering*. 195(19-22), 2495-2508.
[71] Chootinan, P. & A. Chen. (2006). Constraint handling in genetic algorithms using a gradient-based repair method. *Computers & operations research*. 33(8), 2263-2281.
[72] Gupta, S., R. Tiwari, & S. B. Nair. (2007). Multi-objective design optimisation of rolling bearings using genetic algorithms. *Mechanism and Machine Theory*. 42(10), 1418-1443.
[73] Zhang, M., W. Luo, & X. Wang. (2008). Differential evolution with dynamic stochastic selection for constrained optimization. *Information Sciences*. 178(15), 3043-3074.
[74] Wang, L. & L.-p. Li. (2010). An effective differential evolution with level comparison for constrained engineering design. *Structural and Multidisciplinary Optimization*. 41(6), 947-963.
[75] Hedar, A.-R. & M. Fukushima. (2006). Derivative-free filter simulated annealing method for constrained continuous global optimization. *Journal of global optimization*. 35(4), 521-549.
[76] Deb, K. (2000). An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering*. 186(2-4), 311-338.
[77] Runarsson, T. P. & X. Yao. (2005). Search biases in constrained evolutionary optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*. 35(2), 233-243.
[78] Mezura-Montes, E. & C. A. C. Coello. (2005). A simple multimembered evolution strategy to solve constrained optimization problems. *IEEE Transactions on Evolutionary Computation*. 9(1), 1-17.
[79] Michalewicz, Z. (1995). *Genetic algorithms, numerical optimization, and constraints*. in *Proceedings of the sixth international conference on genetic algorithms*. Citeseer.
[80] Rao, R. V., V. J. Savsani, & D. Vakharia. (2011). Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Computer-Aided Design*. 43(3), 303-315.
[81] Wang, Y., et al. (2009). Constrained optimization based on hybrid evolutionary algorithm and adaptive constraint-handling technique. *Structural and Multidisciplinary Optimization*. 37(4), 395-413.
[82] de Fátima Araújo, T. & W. Uturbey. (2013). Performance assessment of pso, de and hybrid pso–de algorithms when applied to the dispatch of generation and demand. *International Journal of Electrical Power & Energy Systems*. 47: p. 205-217.
[83] Liu, H., Z. Cai, & Y. Wang. (2010). Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. *Applied Soft Computing*. 10(2), 629-640.
[84] Bracken, J. & G. P. McCormick (1968). Selected applications of nonlinear programming. Research Analysis Corp Mclean.

Muhammad Farhan Tabassum *is working as Assistant Professor at the University of Lahore, Pakistan and currently pursuing his PhD from UMT Lahore. He has published more than 30 research papers. His research interests are Operations research, Optimization, Numerical analysis, Algorithmic development and Multicriteria decision making. He has more than eight years of teaching experience at the university level and supervised the thesis of M.Phil. Mathematics students.*

http://orcid.org/0000-0002-9958-5015

Sana Akram *is working as Assistant Professor in Lahore Garrison University, Lahore, also doing a Ph.D form UMT Lahore, She published more than 40 research papers. Research field is Graph theory, Operations research, Optimization, Numerical Analysis. He has more than seven year teaching experience at University Level also supervised the thesis of M.Phil Mathematics Students.*

https://orcid.org/0000-0003-2038-9511

Saadia Hassan *is currently working as a senior lecturer at the University of Lahore, Pakistan. After completing her MS in linguistics with nine research publications in stylistics, discourse analysis, sports sciences and physical education and translation analysis. With ample experience of research which she gained as a research scholar at the University of Punjab and co-supervision of more than 5 scholars helped her to enliven her hidden potentials. As a PhD scholar, she has plans to endeavour excellence in the domain of applied linguistics.*

http://orcid.org/0000-0003-1852-2854

Rabia Karim *is currently serving as Senior Lecturer at the University of Lahore, Pakistan and teaching sports management, sports modern technology and Sports sociology. She has 4 publications at her credit in different journals. Before devoting herself to this field, she has worked as Manager National Sports Events at the Sports Board Punjab. She has also played a leading role in developing and establishing several Cricket Coaching Academies, both for Girls and Boys, in various cities of Pakistan.*

http://orcid.org/0000-0001-7343-4262

Parvaiz Ahmad Naik *received his PhD in Mathematics from Maulana Azad National Institute of Technology, a leading institute of India, in December 2015 and currently working as Assistant Professor at the Department of Applied Mathematics, Xi'an Jiaotong University P. R. China. Earlier, he was a postdoctoral research fellow and worked with Prof. Jian Zu at the school of Mathematics and Statistics, Xi'an Jiaotong University, from December 2018-December 2019. His research interests mainly focus on infectious disease dynamics, fractional mathematical modeling, fractional mathematical theory and method and bifurcation analysis. He has published more than 20 SCI research papers in international repute journals like World Scientific, Elsevier, Springer, American Scientific, Taylor & Francis etc. He has received two young scientist awards (gold medals) for his outstanding research work in mathematical biology. Furthermore, he presided over one scientific research project at the national level from the China Postdoctoral Science Foundation under grant no. 2019M663653.*

http://orcid.org/ 0000-0001-8967-5992

Muhammad Farman *did his PhD at the University of Lahore, Pakistan. His research field is mathematical biology, Control theory, Numerical analysis. He has more than seven years of teaching and research experience at the university level and supervised the thesis of M.Phil. and PhD Mathematics students. He has published more than 65 research papers in a national and international journal. He completes several projects in fractional order nonlinear dynamical system with the collaboration of global universities.*

http://orcid.org/0000-0001-7616-0500

Mehmet Yavuz *received his PhD in Mathematics from Balıkesir University, Turkey. He visited the University of Exeter, UK. for post-doctoral research in mathematical biology and optimal control theory for a year. He is currently serving as associate professor at Necmettin Erbakan University, Turkey. His research interests mainly focus on infectious disease dynamics, fractional mathematical modeling, fractional mathematical theory and method, optimal control theory and bifurcation analysis. He has published more than 40 research papers in international esteemed journals and he is a reviewer for about seventy international repute journals.*

http://orcid.org/ 0000-0002-3966-6518

Mehraj-ud-din Naik *is currently working as Assistant Professor at the Department of Chemical Engineering, College of Engineering, Jazan University, Saudi Arabia. He received his PhD in Chemical Engineering from Chonbuk National University, South Korea, in August 2009. Besides this, he worked as a postdoctoral research fellow at the Department of Chemical Engineering and Applied Chemistry, Chungnam National University South Korea from September 2009-October 2010 and Department of Physics and Mechanical Engineering, University of Padova, Italy, from January 2011-February 2013. His research interests mainly focus on chemical engineering, catalysis, nanotechnology, nanomaterials, nanoparticles. He has published more than 15 SCI research papers in the journals of international repute and serving as a reviewer to many SCI-indexed journals.*

http://orcid.org/0000-0001-8192-4843

Hijaz Ahmad *works in a number of mathematical areas, but he is primarily interested in developing new numerical techniques for the solution of differential equations. Recently, he has published many papers in high quality journals on modifications of varaitional iteration algorithm-I, algorithm-II and fractional iteration algorithm. He has Ms in Computational Mathematics from COMSATS University, Pakistan and PhD in Computational Mathematics from the*

University of Engineering and Technology Peshawar, Pakistan. He is an associate member of Section of Mathematics, Uninettuno University, Rome, Italy. He is a reviewer for at least fifty international journals, and also serves on the editorial boards for many good international journals.

http://orcid.org/ 0000-0002-5438-5407

An International Journal of Optimization and Control: Theories & Applications (http://ijocta.balikesir.edu.tr)

An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.11, No.2, pp.178-185 (2021)
http://doi.org/10.11121/ijocta.01.2021.00987

RESEARCH ARTICLE

Performance comparison of approximate dynamic programming techniques for dynamic stochastic scheduling

Yasin Göçgün*

Department of Industrial Engineering, Istinye University, Istanbul, Turkey
yasin.gocgun@istinye.edu.tr

ARTICLE INFO

Article History:
Received 08 June 2020
Accepted 01 March 2021
Available 09 May 2021

Keywords:
Dynamic stochastic scheduling
Markov decision processes
Approximate dynamic programming

AMS Classification 2010:
90-08; 60Jxx

ABSTRACT

This paper focuses on the performance comparison of several approximate dynamic programming (ADP) techniques. In particular, we evaluate three ADP techniques through a class of dynamic stochastic scheduling problems: Lagrangian-based ADP, linear programming-based ADP, and direct search-based ADP. We uniquely implement the direct search-based ADP through basis functions that differ from those used in the relevant literature. The class of scheduling problems has the property that jobs arriving dynamically and stochastically must be scheduled to days in advance. Numerical results reveal that the direct search-based ADP outperforms others in the majority of problem sets generated.

1. Introduction

Approximate dynamic programming (ADP) is a method to solve large-scale Markov decision processes (MDPs), which are used to model systems that evolve stochastically over time. The term approximate refers to the fact that the solution obtained by the underlying ADP technique is an approximate to the optimal solution. ADPs have been used to solve problems arising in diverse fields such as healthcare, manufacturing, transportation, and revenue management.

In the last few decades, various ADP techniques have been proposed to approximately solve computationally intractable MDPs. The state-of-the-art ADP techniques include the Lagrangian-based ADP ([1], [2]), the linear programming-based ADP ([3], [4]), and the direct search-based ADP techniques ([5], [6]). However, the performances of those techniques have not been evaluated in the literature.

To this end, we evaluate the performances of the aforementioned ADP techniques through a class of dynamic stochastic scheduling problems. These problems have the following main features: 1) Jobs arrive dynamically and stochastically at the system over time; 2) Arriving jobs from different types must be scheduled to future time slots such as days. These problems are termed as dynamic stochastic advanced scheduling problems (DSASPs) and arise in various fields such as manufacturing, healthcare, and transportation. We perform a comparison analysis, considering diverse scenarios obtained by different levels of crucial problem parameters. In particular, we approximately solve various problem sets generated for a class of DSASPs introduced in [3] using the aforementioned ADP techniques. The way we implement the direct search-based ADP is unique in that we use new basis functions for value function approximation.

The rest of the paper is structured as follows. Section 2 discusses the relevant literature. In Section 3, we introduce a class of dynamic stochastic advanced scheduling problems, describe the ADP techniques to be compared, and present our computational work. Section 4 includes concluding remarks.

*Corresponding author

2. Literature review

We review the literature on both approximate dynamic programming (ADP) and dynamic stochastic advanced scheduling. A summary of work done on ADP is given first.

2.1. Literature review on approximate dynamic programming

There are many studies focusing approximate dynamic programming (ADP). Powell [7] provides a good review on ADP-based techniques. We briefly review prominent work on ADP below.

Farias and Roy [8] addressed the curse of dimensionality of large-sized stochastic control problems by developing linear programming (LP)-based ADP for solving such problems. In the heart of their approach, a linear combination of basis functions is fitted to cost-to-go function. The authors developed error bounds that ensure performance guarantees. In another study, Farias and Van Roy [9] improved linear programming approach to ADP through the development of constraint sampling. They showed that a subset of constraints can be chosen independently of total number of constraints the problem contains under certain conditions.

Maxwell et al. [5] proposed ADP-based algorithms for ambulance redeployment. In particular, the authors introduced direct search based ADP for solving the underlying problem. Owing to the computationally intensive nature of direct search, they utilized a "post-decision state dynamic programming formulation of ambulance redeployment".

Shechter et al. [10] studied an optimal search problem where the location of a target is only known probabilistically. The authors aimed to minimize the probability of having a failed search and considered the "unconstrained search" and the "constrained search". They developed ADP approaches for larger instances of their problem. Numerical results showed that ADP-based algorithms perform well. In a follow-up study, Gocgun [11] worked on a class of optimal search problems that contain a target and an obstacle. The author provided Markov decision process (MDP) formulations of these problems and proposed a direct search-based ADP for obtaining approximate solutions.

Gocgun and Ghate [12] studied a class of dynamic resource allocation problems where"multiple renewable resources must be dynamically allocated to different types of jobs arriving randomly". The objective is to select "which jobs to service in each time-period so as to maximize total infinite-horizon discounted expected profit." The authors developed a Lagrangian relaxation-based ADP method for obtaining approximate solutions to those problems. In a follow-up study, Gocgun and Ghate [1] proposed an ADP approach based on Lagrangian relaxation for dynamic stochastic scheduling problems. Their computational results demonstrated that the ADP approach outperforms myopic decision rules.

Yin et al. [13] studied a class of metro train scheduling problems, considering performance metrics such as time delay of passengers and operational costs. They proposed a stochastic programming model for this problem and approximately solved it through an ADP-based algorithm.

Wang et al. [14] introduced ADP-based methods through iterated Bellman inequalities. Their methods solve linear and semidefinite programs and provide a bound on optimal value as well as a reasonably good suboptimal policy.

Li and Womer [15] studied a class of project scheduling problems that contain resource constraints and task durations that are uncertain. Differing from the existing research, the authors found a dynamic and adaptive policy through ADP-based algorithms. Specifically, they developed a hybrid ADP framework that makes use of the rollout policy as well as a lookup table approach.

Nozhati et al. [16] developed a framework for recovery management. Their approach utilizes ADP and heuristics for determining recovery actions. Their approach efficiently manages multi-state systems following disasters.

Yang et al. [17] proposed ADP-based algorithms for optimization problems with nonlinear constraints. In particular, they introduced a policy-iteration algorithm to solve the underlying problem, and validated their control method through the simulation of an interconnected plant.

Kanj et al. [18] employed ADP for a problem faced in a ride-hailing system that consists of a fleet of autonomous electric vehicles. Through ADP, the authors developed dispatch strategies to determine, for instance, which car is the most appropriate for a particular trip. Their work showed that the problem contains monotone value functions.

Ou et al. [19] studied a class of gantry scheduling problems where the material transfer is handled by gantries. The authors introduced a method that makes use of reinforcement learning and ADP. Numerical results showed that the proposed

method outperforms a standard Q-learning algorithm.

2.2. Literature review on dynamic stochastic advanced scheduling

One prominent feature of dynamic stochastic advanced scheduling problems (DSASPs) is that they are formulated as Markov decision processes (MDPs). We review research on DSASPs below.

Patrick et al. [3] introduced an MDP formulation of DSASPs where patients from different types are scheduled to future days. Due to intractable state and action spaces, the authors employed an ADP approach; specifically they developed an LP-based ADP to provide approximate solutions. In a follow-up study, Saure et al. [4] studied a DSASP faced in radiation therapy units. The authors provided an MDP formulation of the underlying problem and solved it using an ADP method that is based on linear programming.

Gocgun and Puterman [20] worked on an appointment booking problem faced in chemotherapy settings. Differing from the similar problems, it has the property that patients have target dates along with tolerance limits. The authors proposed an LP-based ADP for acquiring an optimal solution. In a follow-up study, Gocgun [6] worked on a DSASP faced in chemotherapy settings and allows for cancellation of jobs. The author employed a direct search-based ADP for solving larger instances of the underlying problem.

Akhavizadegan et al. [21] addressed appointment scheduling in a nuclear medical center, considering patient choice and different no-show rates. The authors formulated the problem as an MDP and compared the optimal solution with heuristic decision rules. Wang and Fung [22] studied a class of dynamic appointment scheduling problems considering patient preferences and choices. The authors developed a column generation-based approximation algorithm to solve these problems. Lu et al. [23] worked on a class of dynamic appointment scheduling problems taking into account "wait-dependent abandonment". They formulated these problems as MDPs and investigated the properties of the optimal policy theoretically.

In a recent work, Saure et al. [24] studied a DSASP where service times are stochastic. The authors put forth theoretical results for the deterministic case with "multi-class, multi-priority" jobs, and then developed methods for the stochastic case.

2.3. Contribution

Table 1 indicates research work in which a comparative analysis was performed using any of the LP-based ADP (LP-A), the direct search-based ADP (DS-A), and the Lagrangian-based ADP (LGR-A) techniques. To the best of our knowledge, the relevant literature does not contain any work that deals with a comparative analysis of all the three state-of-the-art ADP techniques.

Table 1. The list of work in which any of LP-A, DS-A, and LGR-A was developed. "S-based A" refers to Simulation-based ADP.

Study	Proposed ADP	Comparison
[5]	DS-A	against S-based A
[10]	DS-A	against heuristics
[11]	DS-A	against heuristics
[12]	LGR-A	against myopic
[1]	LGR-A	against myopic
[3]	LP-A	against myopic
[4]	LP-A	against myopic
[20]	LP-A	against myopic
[6]	DS-A	against myopic
[2]	LGR-A	against myopic

In this research, we evaluate the performance of three state-of-the-art ADP techniques, employing them for solving a class of DSASPs. In particular, the Lagrangian-based ADP, the LP-based ADP, and the direct-search based ADP were used to solve the DSASP introduced in [3]. Our contribution is twofold: 1) We employ the direct search-based ADP through basis functions that are different as compared to those used in the literature, 2) We close a gap in the literature, addressing the question of which of those techniques perform the best in an important class of dynamic scheduling problems.

The features of the DSASP we studied are given next.

3. Dynamic stochastic advanced scheduling

The dynamic stochastic advanced scheduling problem introduced in [3] has the following features.

- Heterogeneous job types are considered.
- Arrivals of jobs to the system are random. In addition, arrivals across job types are independent.

- Jobs arriving at the system must be scheduled to a day within a booking horizon. Rejecting (or outsourcing or serving through overtime) jobs is allowed.
- There is a deadline for jobs of each type. Scheduling a job to a day after its deadline results in delay cost.
- Rejecting jobs results in a penalty cost.
- The goal is to make decisions of scheduling and rejecting arriving jobs so as to "minimize the total discounted expected cost over an infinite horizon" ([20]).

3.1. The Markov decision process model

The following notations are used in the mathematical model of the aforementioned problem.

- I: the number of job types
- N: the length of the booking horizon
- x_n, $n = 1, \ldots, N$: number of jobs that are already scheduled to day n
- u_i, $i = 1, \ldots, I$: number of type-i jobs waiting to be scheduled
- y_{in}, $i = 1, \ldots, I$, $n = 1, \ldots, N$: number of type-i jobs to be scheduled to day n
- z_i, $i = 1, \ldots, I$: number of type-i jobs that are rejected
- C_1: daily capacity
- C_2: upper bound on number of jobs rejected each day
- $p(u_i')$: probability that u_i' jobs of type-i arrive on a given day
- D^i: a deadline associated with type-i job
- $C^i(n, D^i)$: delay cost of scheduling a type-i job on day n
- $r(i)$: rejection cost of a type-i job
- F^i, $i = 1, \ldots, I$: unit delay cost for a type-i job
- D: the set of all possible demand vectors

The Markov decision process (MDP) model of the aforementioned problem is provided next (see [3] for an equivalent formulation).

State Space: $s = (x, u) = (x_n, u_i)$, $i = 1, \ldots, I$ and $n = 1, \ldots, N$. The state of the system consists of the number of jobs that are already scheduled to each day in a booking horizon, and the number of jobs of each type waiting to be scheduled.

The Action Set: $(y, z) = (y_{in}, z_i)$, $i = 1, \ldots, I$ and $n = 1, \ldots, N$. The action to be made at a given state is to decide the number of jobs of each type to be scheduled to each day of the booking horizon, and the number of jobs of each type that are rejected. Note that z_i does not have the day index, as it represents the number of jobs of type-i that will not be scheduled to any day of the booking horizon and hence are rejected. Any action must satisfy certain constraints, which are provided below ([3]).

$$x_n + \sum_{i=1}^{I} y_{in} \leq C_1, \ n = 1, \ldots, N, \tag{1}$$

$$\sum_{i=i}^{I} z_i \leq C_2, \tag{2}$$

$$\sum_{n=1}^{N} y_{in} + z_i \leq u_i, \ i = 1, \ldots, I. \tag{3}$$

Constraint 1 ensures that the sum of the number of jobs that are already scheduled to day n and total number of jobs to be scheduled to day n does not exceed the daily capacity. Constraint 2 guarantees that total number of jobs rejected is bounded by C_2. Finally, Constraint 3 ensures that the sum of the total number of type-i jobs to be scheduled and the number of type-i jobs rejected cannot be greater than the number of type-i jobs waiting to be scheduled.

Transition Probabilities: Stochasticity in the system arises only due to the number of new arrivals of jobs from each type. Hence, once an action is chosen at a given state $(x_1, x_2, \ldots, x_N, u_1, u_2, \ldots, u_I)$, the system switches to the following state with probability $\prod_{i=1}^{I} p(u_i')$ due to the assumption of independent arrivals:
$(x_2 + \sum_{i=1}^{I} y_{i2}, x_3 + \sum_{i=1}^{I} y_{i3}, \ldots, x_N + \sum_{i=1}^{I} y_{iN}, 0,$
$u_1', u_2', \ldots, u_I')$. Here, for instance, $x_2 + \sum_{i=1}^{I} y_{i2}$ represents x_1'.

Costs: The immediate cost of choosing an action at a given state consists of total delay cost and total rejection cost. It is mathematically expressed as follows.
$c(y, z) = \sum_{i=1}^{I} \sum_{n=1}^{N} C^i(n, D^i) y_{in} + \sum_{i=1}^{I} r(i) z_i$.
$C^i(n, D^i)$ for $i = 1, \ldots, I$ is expressed as

$$C^i(n, D^i) = \max(n - D^i, 0) \times F^i, \ n = 1, \ldots, N. \tag{4}$$

Bellman's Equations: The cost-to-go function of a given state is given by

$$v(x, u) = \min_{(y,z)} \left\{ c(y, z) + \lambda \sum_{u' \in D} (u') v(x', u') \right\}. \tag{5}$$

Owing to extremely large number of states and actions, the underlying MDP model is computationally intractable. The three approximate dynamic programming techniques are briefly described next.

3.2. Approximate dynamic programming techniques

Due to curse of dimensionality, Bellman equations given in Eqn. 5 cannot be solved. The fundamental theme behind approximate dynamic programming (ADP) is to approximate the value function (i.e., cost-to-go function) through a combination of basis functions, thereby eliminating the computational intractability.

ADPs are mainly categorized as mathematical programming (MP)-based ADP, simulation-based ADP, and direct search-based ADP. MP-based ADPs transform the underlying MDP model into the "equivalent linear programming (LP) version of Bellman equations. Approximate value function is then used to avoid intractability" ([20]). Examples of MP-based ADPs are linear programming (LP)-based ADP and Lagrangian-based ADP. Simulation-based ADP techniques, however, simulate "the evolution of the system over a number of initial states in order to tune the parameters" ([6]), thereby finding an approximate solution to the Bellman's equations. Simulation models such as reinforcement learning and statistical sampling are used to estimate value functions. On the other hand, ADP based on direct search tackles an optimization problem where the decision variables are tuning parameters, and the goal is to minimize "the expected cost of the policy induced by the corresponding parameter vector" ([6]). The optimization problem is solved through direct search.

As stated earlier, in ADP, basis functions that possess certain important features of the system state are used to approximate the value function. One example of utilizing basis functions is linear approximation, which is given by

$$V(s) \approx \sum_{k=1}^{K} r_k \Phi_k(s),$$

where "r_k for $k = 1, \ldots, K$ are tuning parameters and $\Phi_k(s)$ for $k = 1, \ldots, K$ are basis functions" ([11]). The approximation parameters are tuned iteratively to acquire an ADP policy after the approximation of the value function is performed. In this context, ADP approaches aim to find the optimal parameter vector through which a certain performance metric is minimized ([11]).

The parameter tuning phase enables us to have the approximate value of a given state. We then retrieve the ADP policy through the computation of a decision vector for any given state.

We briefly describe the three approximate dynamic programming techniques, without delving into all mathematical details. (refer to [3], [1] and [6] for technical details of these methods).

3.2.1. Linear programming-based ADP

The LP approximation is provided below (see [3] for the complete steps of the LP-based ADP).

"For a discounted infinite-horizon MDP (where the objective function is in minimization form as in (5) and $\alpha(\vec{s})$ are positive numbers indexed by states $\vec{s} \in S$), the equivalent LP formulation is given" ([20]):

$$\begin{aligned} &\max \sum_{\vec{s} \in S} \alpha(\vec{s}) v(\vec{s}) \\ &s.t.\ c(\vec{s}, \vec{a}) + \\ &\lambda \sum_{\vec{s'} \in S} p(\vec{s'} | \vec{s}, \vec{a}) v(\vec{s'}) \geq v(\vec{s}),\ \forall \vec{s} \in S, \vec{a} \in A_{\vec{s}}. \end{aligned} \tag{6}$$

Using an affine approximation, the value function can be approximated as:

$$\tilde{v}(\vec{x}, \vec{u}) = W_0 + \sum_{n=1}^{N} V_n x_n + \sum_{i=1}^{I} W_i u_i. \tag{7}$$

The LP formulation of our MDP model is then:

$$\begin{aligned} &\max_{v} \sum_{(\vec{x}, \vec{u}) \in S} \alpha(\vec{x}, \vec{u}) v(\vec{x}, \vec{u}) \\ &s.t.\ c(\vec{y}, \vec{z}) + \\ &\lambda \sum_{d \in D} p(d) v(x_2 + \sum_i y_{i2}, \ldots, x_N + \sum_i y_{iN}, 0, u_i^{'}) \\ &\geq v(\vec{x}, \vec{u}),\ \forall (\vec{x}, \vec{u}) \in S,\ \forall (\vec{y}, \vec{z}) \in A_{(\vec{x}, \vec{u})}. \end{aligned} \tag{8}$$

We substitute (7) into (6) and obtain the following LP after rearranging terms ([20]):

$$\begin{aligned} &\max_{\vec{V}, \vec{W}} \quad W_0 + \sum_{n=1}^{N} E_\alpha(X_n) V_n + \sum_{i=1}^{I} E_\alpha(U_i) W_i \\ &s.t.\ (1 - \lambda) W_0 + \\ &\sum_{n=1}^{N} V_n (x_n - \lambda x_{n+1} - \lambda \sum_{i=1}^{I} y_{i(n+1)}) + \\ &\sum_{i=1}^{I} W_i (u_i - \lambda E_\alpha(U_i)) \leq c(\vec{y}, \vec{z}),\ \forall (\vec{x}, \vec{u}) \in S, \\ &\forall (\vec{y}, \vec{z}) \in A_{(\vec{x}, \vec{u})}, \\ &V_n \geq 0,\ n = 1, \ldots, N, \\ &W_i \geq 0,\ i = 1, \ldots, I. \end{aligned} \tag{9}$$

As the above LP "still has a very large number of constraints" ([20]), its dual is solved through column generation (see [3]).

3.2.2. Lagrangian relaxation-based ADP

The Lagrangian approach is similar to the LP-based in that it transforms the underlying MDP and tackles the equivalent LP formulation of the MDP through the problem decomposition obtained by Lagrange multipliers. As the resulting LP is still intractable, a hybrid Lagrangian relaxation - LP approach is employed to tackle intractability. In particular, the Lagrangian value functions are approximated through affine functions. The resulting approximate LP is solved using a column generation method. ([25], [1])

3.2.3. Direct-search based ADP

As part of the direct search-based ADP, we tune approximation parameters using direct search with the goal of finding good policies. To be more specific, an optimization problem where feasible r's constitute the variables and the goal is to minimize "the expected cost of the policy induced by the corresponding parameter vector" ([6]) is solved by direct search. As a result, we have the following optimization problem ([11]):

$$\min_{r \in R^N} \sum_{t=0}^{\infty} c(s_t, \pi_r(s_t)), \tag{10}$$

where "s_t is the state at stage t of the system, π_r is the policy obtained by the parameter vector r, $\pi_r(s_t)$ is the action dictated by the policy π_r in the state at stage t, and $c(s_t, \pi_r(s_t))$ is immediate cost incurred at step t as a result of choosing $\pi_r(s_t)$." [11]

We use the following basis functions during the implementation of the direct-search based ADP.

$$\Phi_1(s) = C_1 - \sum_{n=1}^{N} \sum_{i=1}^{I} (x_{in} + y_{in}), \tag{11}$$

$$\Phi_2(s) = -(\sum_{i=1}^{I} z_i).$$

The first basis function represents available capacity (see [6] for a somewhat similar basis function), whereas the second one allows us to consider different values of z_i for the underlying optimization problem.

Because of having two basis functions, two tuning parameters are used, which are r_1 and r_2. We let r_1 and r_2 range from 1 to 5 in increments of 1, and 0 to 40 in increments of 2, respectively. For each problem instance, the combination of (r_1, r_2) that yields the best value is used for computing average cost values.

3.3. Numerical experiments

Data generation was performed by taking into account the way data is generated in the literature ([1]). Number of types was set to 5 and 10. Arrival probabilities of jobs are assumed to follow Poisson with a parameter DU (1,5) (DU means discrete uniform). Discount factor was set to 0.9 and 0.99. Resource availability was set to 10 and 20. Two levels were considered for booking horizon: 7 and 14. As a result, we have 16 scenarios for the comparison analysis.

As the arrival process is random, we estimate the discounted expected cost accrued by any of the three ADP techniques by averaging the total discounted cost through simulation. Simulation run length was set to 50, and number of replications was set to 20, which means that the total discounted cost is averaged over 20 independent simulations. For each problem set, we ran 10 problem instances.

3.3.1. Results

We provide results in tables 2 and 3. For each problem set determined by the combination of I,C_1, and N, columns 2 to 4 of each table give the average discounted cost values over 10 independent problem instances obtained for the Lagrangian-based, the LP-based, and the direct search-based ADP, respectively. The last column of each table gives the percentage difference between the best and next best techniques. The bolded percentage difference values correspond to problem sets where the Lagrangian-based ADP outperforms others whereas other values correspond to problem sets where the direct search-based ADP outperforms others. When the discount factor (λ) has a high level (i.e., 0.99); the Lagrangian-based ADP turns out to be the best approach in 5 out of 8 problem sets, whereas the direct search-based ADP outperforms others in two problem sets. When the discount factor was set to a low level (i.e., 0.9), the direct search-based ADP outperforms others in all problem sets. (Paired t-tests revealed that the respective percentage differences were statistically significant at the 0.05 level.)

Table 2. Results for $\lambda = 0.99$.

(I, C_1, N)	LGR-A	LP-A	DS-A	Per. d.
(5,5,7)	15379	20208	16991	**9.5**
(5,5,14)	15273	20127	15763	**3.1**
(5,10,7)	6546	10380	7345	**10.9**
(5,10,14)	6831	9556	6676	2.3
(10,10,7)	31110	41513	33120	**6.1**
(10,10,14)	31256	40179	31035	0.7
(10,20,7)	12310	21844	13062	**5.7**
(10,20,14)	12627	16746	11174	11.5

Table 3. Results for $\lambda = 0.90$.

(I, C_1, N)	LGR-A	LP-A	DS-A	Per. d.
(5,5,7)	3758	3785	3346	11
(5,5,14)	3758	3197	2762	13.6
(5,10,7)	1708	1685	1327	21.2
(5,10,14)	1708	1172	1022	12.8
(10,10,7)	7276	7514	6451	11.3
(10,10,14)	7276	5978	5185	13.3
(10,20,7)	2923	2827	2351	16.8
(10,20,14)	2923	1541	1461	5.2

4. Conclusions

In this paper, we aimed to close a gap in the literature by comparing the performances of the state-of-the-art approximate dynamic programming (ADP) techniques through a class of dynamic stochastic advanced scheduling problems (DSASPs). These problems are modeled as Markov decision process and their large instances are approximately solved via ADP techniques. We solved a class of these problems using three ADP approaches: 1)Lagrangian-based ADP, 2) Linear programming-based ADP, and 3) direct search-based ADP, which we uniquely implemented through new basis functions.

Our numerical experiments reveal that the direct search-based ADP outperforms others in 10 out of 16 problem sets. On the other hand, the Lagrangian-based ADP outperforms others in 5 out of 16 problem sets. Future research may focus on the performance comparison of such techniques through variants of DSASPs that include extensions such as cancellations of jobs, multiple resources, and overbooking.

References

[1] Gocgun, Y., & Ghate, A. (2012). Lagrangian relaxation and constraint generation for allocation and advance scheduling. *Computers & Operations Research*, 39, 2323-2336.

[2] Parizi, M. S., & Ghate, A. (2016). Multi-class, multi-resource advance scheduling with no-shows, cancellations and overbooking. *Computers & Operations Research*, 67, 90-101.

[3] Patrick, J., Puterman, M. L., & Queyranne, M. (2008). Dynamic multi-priority patient scheduling for a diagnostic resource. *Operations Research*, 56, 1507-1525.

[4] Saure, A., Patrick, J., Tyldesley, S., & Puterman, M. L. (2012). Dynamic multi-appointment patient scheduling for radiation therapy. *European Journal of Operational Research*, 223, 573-584.

[5] Maxwell, M.S., Henderson, S. G., & Topaloglu, H. (2013). Tuning approximate dynamic programming policies for ambulance redeployment via direct search. *Stochastic Systems*, 3(2), 322-361.

[6] Gocgun, Y. (2018). Dynamic scheduling with cancellations: An application to chemotherapy appointment booking. *An International Journal of Optimization and Control: Theories and Applications*, 8, 2, 161-169.

[7] Powell, W. B. (2007). *Approximate dynamic programming: solving the curses of dimensionality*, Hoboken, New Jersey, USA: John Wiley and Sons.

[8] De Farias, D.P. & Roy, B.V. (2003). The linear programming approach to approximate dynamic programming. *Operations Research*, 51, 850-865.

[9] De Farias, D. P. & Roy, B. V. (2004). On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of Operations Research*, 29(3), 462–478.

[10] Shechter, S. M., Ghassemi, F., Gocgun, Y., & Puterman, M. L. (2015). Technical note – trading off quick versus slow actions in optimal search. *Operations Research*, 63(2), 353–362.

[11] Gocgun, Y. (2019). Approximate dynamic programming for optimal search with an obstacle. *Selcuk University Journal of Engineering, Science, and Technology*, 7, 80-95.

[12] Gocgun, Y., & Ghate, A. (2010). A Lagrangian approach to dynamic resource allocation. *Proceedings of the Winter Simulation Conference*, 3330-3338.

[13] Yin, J., Tang, T., Yang, L., Gao, Z., & Ran, B. (2016). Energy-efficient metro train rescheduling with uncertain time-variant passenger demands: an approximate dynamic programming approach. *Transportation Research, Part B*, 91, 178-210.

[14] Wang, Y., O'Donoghue, B., & Boyd, S. (2015). Approximate dynamic programming via iterated bellman inequalities. *International Journal of Robust and Nonlinear Control*, 25, 1472–1496.

[15] Li, H., & Womer, N. K. (2015). Solving stochastic resource-constrained project scheduling problems by closed-loop approximate dynamic programming. *European Journal of Operational Research*, 246, 20-33 2015.

[16] Nozhati, S., Sarkale, Y., Ellingwood, B., Chong, E. K. P., & Mahmoud, H. (2019). Near-optimal planning using approximate dynamic programming to enhance post-hazard community resilience management. *Reliability Engineering and System Safety*, 181, 116-126.

[17] Yang, X., He, H., & Zhong, X. (2019). Approximate dynamic programming for nonlinear-constrained optimizations. *IEEE Transactions on Cybernetics*.

[18] Al-Kanj, L., Nascimento, J., & Powell, W. B. (2020). Approximate dynamic programming for planning a ride-sharing system using autonomous fleets of electric vehicles. *European Journal of Operational Research*, 284, 1088-1106.

[19] Ou, X., Chang, Q., & Chakraborty, N. (2021). A method integrating Q-Learning with approximate dynamic programming for gantry work cell scheduling. *IEEE Transactions on Automation Science and Engineering*, 18, 1, 85-93.

[20] Gocgun, Y., & Puterman, M. L. (2014). Dynamic scheduling with due dates and time windows: an application to chemotherapy patient appointment booking. *Health Care Management Science*, 17, 60-76.

[21] Akhavizadegan, F., Ansarifar, J., & Jolai, F. (2017). A novel approach to determine a tactical and operational decision for dynamic appointment scheduling at nuclear medical center. *Computers & Operations Research*, 78, 267–277.

[22] Wang, J., Fung, R.Y., & Chan, H.K. (2015). Dynamic appointment scheduling with patient preferences and choices. *Industrial Management & Data Systems*, 115(4), 700-717.

[23] Lu, Y., Xie, X., & Jiang, Z. (2018). Dynamic appointment scheduling with wait-dependent abandonment. *European Journal of Operational Research*, 265(3), 75-984.

[24] Saure, A., Begen, M.A. & Patrick, J. (2020). Dynamic multi-priority, multi-class patient scheduling with stochastic service times. *European Journal of Operational Research*, 280(1), 254–265.

[25] Gocgun, Y. (2010). *Approximate dynamic programming for dynamic stochastic resource allocation with applications to healthcare.* PhD Thesis. The University of Washington.

Yasin Göçgün *received his B.S. degree and M.S. degree from the Industrial Engineering Department at Bilkent University in 2003 and 2005, respectively. After completing his doctoral studies in the Industrial and Systems Engineering Department at the University of Washington in 2010, Dr. Göçgün worked as a postdoctoral fellow in the Sauder School of Business at the University of British Columbia between 2010 and 2012. After his first post-doc, Dr. Göçgün carried out another post-doc study in the Mechanical and Industrial Engineering Department at the University of Toronto between 2012 and 2014. He has been working as an assistant professor in the Industrial Engineering Department at Istinye University. His research interests primarily focus on dynamic stochastic optimization, operations research in healthcare, Markov decision processes, approximate dynamic programming, and scheduling.*

https://orcid.org/0000-0003-3005-7596

An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.11, No.2, pp.186-198 (2021)
http://doi.org/10.11121/ijocta.01.2021.01090

RESEARCH ARTICLE

Reconstruction of potential function in inverse Sturm-Liouville problem via partial data

Mehmet Açil[a] and Ali Konuralp[b*]

[a] *Department of Mathematics, Van Yüzüncü Yıl University, Turkey*
[b] *Department of Mathematics Manisa Celal Bayar University, Turkey*
mehmet.acil@yyu.edu.tr, ali.konuralp@cbu.edu.tr

ARTICLE INFO

Article History:
Received 26 February 2021
Accepted 09 May 2021
Available 12 May 2021

Keywords:
Sturm-Liouville theory
Numerical approximation of eigenvalues and of other parts of the spectrum
Optimization

AMS Classification 2010:
31A25; 34B24; 34L16

ABSTRACT

In this paper, three different uniqueness data are investigated to reconstruct the potential function in the Sturm-Liouville boundary value problem in the normal form. Taking account of Röhrl's objective function, the steepest descent method is used in the computation of potential functions. To decrease the volume of computation, we propose a theorem to precalculate the minimization parameter that is required in the optimization. Further, we propose a novel time-saving algorithm in which the obligation of using the asymptotics of eigenvalues and eigenfunctions and the appropriateness of selected boundary conditions are also eliminated. As partial data, we take two spectra, the set of the jth elements of the infinite numbers of spectra obtained by changing boundary conditions in the problem, and one spectrum with the set of terminal velocities. In order to show the efficiency of the proposed method, numerical results are given for three test potentials which are smooth, nonsmooth continuous, and noncontinuous, respectively.

1. Introduction

The inverse Sturm-Liouville (S-L) reconstruction problems consist of the calculation of the potential function from known data. These data which can be referred as spectrum, normalized constants, spectral functions, etc. are based on uniqueness. Although there are many studies on the uniqueness of the inverse S-L problem, it is observed from the literature that only a few studies have been conducted on the reconstruction of the potential function from data that can be obtained experimentally. In this study, we aim to contribute to the literature in this direction.

Let's consider the Sturm-Liouville problem

$$\begin{aligned} -y''(x)+(\lambda+q(x))y(x)=0, \qquad & x\in[0,1], \\ y'(0)\sin\alpha+y(0)\cos\alpha=0, \quad & (1) \\ y'(1)\sin\beta+y(1)\cos\beta=0, \quad & \end{aligned}$$

where $q\in L_2[0,1]$ is potential function, λ is the eigenvalue parameter, and α,β are real constants.

In 1978, Hald [1] considered the inverse problem of (1) for symmetric potential q with Dirichlet conditions, and by using Rayleigh-Ritz method he reduced the problem consist of Fourier expansion with finite terms to an eigenvalue problem for a matrix. Thus, he showed that the solution for matrix problem converged to the solution for the inverse problem as the dimension of the matrix increased. In 1984, Paine [2] used an algorithm to deal with a similar problem assuming $q\in C^2[0,\pi]$ in which a perisymmetric tridiagonal matrix obtained from first N eigenvalues. The errors arising from results obtained by using a correction term were analyzed according to the increasing the number N. In 1988, by the use of characteristic values $\{\lambda_n,\rho_n\}$, Sacks [3] generated the iteration algorithm

*Corresponding Author

$$q_{n+1}(x) = q_n(x) + 2g(2x) - 2F(q_n)(2x),$$

for inverse S-L problem subject to Dirichlet boundary condition where

$$g(t) = \sum_{k=1}^{\infty} \left(2k\pi \sin(k\pi t) - \frac{1}{\lambda_k^{1/2} \rho_k} \sin(\lambda_k^{1/2} t) \right),$$
$$F(q)(t) = w_{x,t}(0,t),$$

and $w(x,t)$ was considered to be the solution to related Goursat problem. Then some numerical examples were considered by noticing the convergence of q_n to the potential q. In 1992, Lowe et al. [4] investigated the vector $\vec{q} = (q_1, q_2, ...)$ by proposing the potential in the form of

$$q^N = q_N + \sum_{k=1}^{N-1} q_k \phi_k(x). \tag{2}$$

for the S-L problem with Dirichlet boundary condition and with also general separable boundary conditions where

$$\{\phi_k(x)\}_{k=1}^{k=2N} = \{\sin 2\pi x, \cos 2\pi x, ... \\ , \sin 2N\pi x, \cos 2N\pi x\}.$$

To determine the Fourier coefficients, they took the advantage of Newton's method, proved a convergence theorem of the method, and gave some examples. In the same year, Rundell and Sacks [5] handled two spectra to determine Cauchy data for a transformed hyperbolic partial differential problem, and then they used successive approximation method and Quasi-Newton method respectively. Finally, they considered the same reconstruction problem for different data and controlled effectiveness of their approach on smooth, non-smooth continuous, and noncontinuous test potentials. In 1994, Neher [6] studied the investigation of potential function q when the eigenvalues and symmetric base functions were given for the problem having symmetric potential with Dirichlet condition. His study was based on the determination of a_j constants in which the potential function was written in the form of

$$q(x) = q(x;a) := \hat{q}(x) + \sum_{j=1}^{n} a_j q_j(x), \tag{3}$$

where $a = (a_j) \in \mathbb{R}^n$. He used Newton's method to find zeros of the function $f(a) = (f_i(a)) = (\lambda_i(q(x;a)) - v_i)$ where the eigenvalues were v_i; $i = 1, 2, ..., n$. In fact, the formulae (2) and (3) used in [4] and [6], respectively, was considered in several earlier papers, notably the paper of O.H. Hald [1]. However, [4] deals with the difficulties arising from the limitation of data available in real applications, while [6] addresses the problem of attempting to enclose q within an interval-valued function. In 1995, Fabiano et al. [7] considered Dirichlet problem with symmetric and general potential, then converted the problem to a matrix equation through a partition of the interval. In this equation, the finite set of eigenvalues and terminal velocities were used to determine the matrix of coefficients. In 2004, Andrew [8] used the same method, which is Modified Newton's method, with Fabiano et al. [7]. The difference between the approaches used in [7] and [8] is that [7] uses a second order finite difference approximation of the differential equation, whereas [8] uses the more accurate Numerov method. The advantages of the latter approach are discussed in [9]. The author generalized the case in 2005 [9], and in 2011 he used a similar method to solve the problem corresponding to a different set of data [10]. In 2003, Brown et al [11] considered a finite number of linear dependence coefficients between some appropriate solutions to the Sturm-Liouville problem and eigenvalues. They used the steepest descent method for the objective functional

$$G(q) = \sum_{n=0}^{N} \{\omega_n \int_0^1 [(u_q' - C_n v_q')^2 \\ + (u_q - C_n v_q))^2]\},$$

where $\{\lambda_n, C_n\}$ was the given set of finite data, and the functions u_q, v_q were the solutions corresponding to λ_n for the Sturm-Liouville equation with some special initial conditions. In 2005, Röhrl [12] handled two spectra and used Polak-Ribiere conjugate gradient method to minimize his objective functional given with (4). In his ongoing study [13], he also generalized his work to boundary conditions. In 2007, Rafler and Böckmann [14] modified the Rundell-Sacks method to deal more effectively with potentials having jump discontinuities and gave some numerical examples in L_2 and L_∞. Additionally, some other methods such as the boundary value method and the finite difference method to find the solutions of

the inverse Sturm-Liouville problems can be seen in [15–18].

This paper is summarized as follows. Section 2.1 discusses the improvement of efficiency of the study given in [12]. In the method described in Sections 2 and 3 of [12], asymptotic formulas were used for the minimization of the objective functional in each iteration so that the original potential was approached as a Fourier series. Here, however, by using an estimate of minimization parameter which is calculated approximately for a random potential, the necessity of using asymptotics is eliminated. So, the original potential is approached as a linear combination of the eigenfunctions which is calculated with the help of an initial value problem. Two important advantages of this statement are expressed below:

i) In the reconstruction problems, the errors in calculating the potential increase as the data at hand decreases. Besides, there are two factors that cause errors. The first of these is the error resulting from not being able to solve the direct problem analytically. It is not included in this study. The second is, however, the error that arises from the use of asymptotics in each iteration. As it can be observed from the asymptotics, though this error is small in the calculation of large eigenvalues, one cannot say the same for small eigenvalues. Thus, in the case where the number of experimentally obtained data is limited, one would like to minimize the error which occurs in small eigenvalues. As a result, the elimination of the necessity of the use of asymptotics minimizes the above-mentioned error. In this respect, better results are achieved by taking a small number of data pairs in our study. Moreover, even though increments on the iteration numbers is comprehended as a disadvantage for the case of two spectra, an important gain in terms of time is achieved because the calculation volume is decreased.

ii) The asymptotics of eigenfunctions corresponding to each element of different spectra can be the same (See [12]). Thus, in the calculation, a challenge occurs between the terms included in the gradient. This leads to significant increases in the number of iterations (Please see Figure 1-3 and the first paragraph of page 2013 in [12]). However, the elimination of asymptotics prevents such a challenge. Therefore, a gain in both the number of iterations and the time of calculation is achieved in our paper.

In Section 2.2 and Section 2.3, the process for the two spectra case in the previous section is also applied to the case of two different data sets. The first of these data is McLaughlin-Rundell data which was considered in [19]. When the studies about this problem is surveyed, while almost all of them are about generalizations of the uniqueness problem of [19], this study discusses the numerical solution of the problem. The second one is a spectrum and a set of terminal velocities. Finally, Section 3 is devoted to demonstrating the numerical results obtained for each uniqueness data in Section 2.

2. Reconstruction of the potential

2.1. Two spectra

We propose to remove the use of asymptotic formulae of eigenvalues and corresponding eigenfunctions while reconstructing the potential. As a result of this suggestion, small eigenvalues and their corresponding eigenfunctions can be used more effectively. Moreover, different eigenvalues do not have to fight against each other when the corresponding asymptotic formulas for eigenfunctions are the same. Indeed, we especially handle the steepest descent method for Röhrl's objective functional.

It is well known that two spectra obtained by changing boundary condition determine the potential uniquely for (1) [20].

Let $I = \mathbb{M} \times \{1,2\}$, $\mathbb{M} \subset \mathbb{N}$ and $\{\lambda_{i,j,Q}\} = \{\lambda_i(\alpha, \beta_j, Q)\}$ be two spectra of the inverse S-L problem subject to different two boundary conditions for test potential Q. The Röhrl's objective functional is

$$G(q) = \sum_{(i,j)\in I} \omega_{i,j}(\lambda_{i,j,q} - \lambda_{i,j,Q})^2, \tag{4}$$

where $\omega_{i,j}$ is positive weight constant. Since $\frac{\partial \lambda_{i,j,q}}{\partial q(x)} = \dot{\lambda}_{i,j,q} = g^2_{i,j}(x,q)$ [12, 21],

$$\nabla G(q) = 2 \sum_{(i,j)\in I} \omega_{i,j}(\lambda_{i,j,q} - \lambda_{i,j,Q}) g^2_{i,j}$$

is in $H^1([0,1])$- Sobolev space where $g_{i,j}(x,q)$ are normalized eigenfunctions which correspond to the eigenvalues $\lambda_{i,j,q}$ [12].

It is seen from the literature that this type of objective function was firstly considered by Brown et al. [11]. Röhrl handled two spectra instead of Brown's data $\{\lambda_n, C_n\}$ and generalized the problem to determine both the boundary conditions and the potential function [13]. Although it

seems that Röhrl's functional is much simple than Brown's, it contains more computation than those in [11] because Röhrl needs to solve the boundary value problem instead of the initial value problem. Here, it is aimed to minimize functional $G(q)$ step by step so that one can approach potential Q by using iteration functions q_n. If I is infinite and the positive weights $(i\omega_{i,j})$ are summable, then the series given by $G(q)$ is convergent (see [12]). Furthermore $G(q) = 0$ if and only if $q = Q$ from uniqueness [20]. The convergence is obvious if I is finite.

Theorem 1. *If I is finite or $(i\omega_{i,j})$ is summable, the functional $G(q)$ has no local minima at q with $G(q)$, i.e.$\nabla G(q) = 0 \Leftrightarrow G(q) = 0$. Thus a conjugate gradient algorithm will not get trapped in local minima [12].*

By Theorem 1, the minimization process leads to $G(q_n) \to 0$ so that $q_n \to Q$. The algorithm to be used is the steepest descent algorithm which is described as follows [22]:

Step 0: Choose an initial potential as q_0 and set $n = 0$,
Step 1: If $G(q_n)$ is small enough, stop; otherwise go to step 2,
Step 2: Compute gradient $\nabla G(q_n)$,
Step 3: Minimize $G(q_n - h_n \nabla G(q_n))$ with respect to h_n,
Step 4: Set $q_{n+1} := q_n - h_n \nabla G(q_n)$ and replace n by $n + 1$, go to step 1.

There are three cases for the asymptotics of eigenvalues according to $\{\alpha, \beta\}$ in the eqn. (1) (See [12]). To compute asymptotics of squared normalized eigenfunctions corresponding to eigenvalues with respect to $\{\alpha, \beta\}$, one can consider the study done by Hochstadt [23] which contains asymptotic solution to the normal form Sturm-Liouville differential equation with initial values $y(0) = \sin\alpha$, $y'(0) = -\cos\alpha$. For example, if $\alpha = \beta = \frac{\pi}{4}$ and $\alpha = \frac{\pi}{4}, \beta = -\frac{\pi}{4}$, then the squared normalized eigenfunctions have the same asymptotics $g^2_{i,j} = 1 + \cos 2i\pi x + O\left(\frac{1}{i}\right)$ for all $i \in \mathbb{M}$ and $j = 1, 2$. These values of $\{\alpha, \beta\}$ were considered in [13]. When one chooses the values of $\{\alpha, \beta\}$ which have the same asymptotic form for corresponding the squared normalized eigenfunctions, the number of $g^2_{i,j}$ in the gradient drops to half, and so the coefficients of the same asymptotics $g^2_{i,j}$ fight each other for each index j. Thus, eliminating the necessity of the use of asymptotics is proposed to overcome such a conflict. Actually, because the asymptotics are not required until Step 3, any solver for eigenvalues and eigenfunctions can be used. Since the eqn. (1) cannot be solved in h_n for Step 3, as long as a command that computes the parameter h_n, is used, the asymptotics needs to be considered. Therefore, we obtain an estimate for h_n by making some omissions:

Let us consider $\lambda_{i,q}(\alpha, \beta) = F_{i,\alpha,\beta} + \int_0^1 q ds + a_i$ where $(a_i) \in \ell_2$ and $F_{i,\alpha,\beta}$ can be seen from asymptotics for eigenvalues (See [12]). Then, for the test potential Q and the iterative potential q_n, it can be written as

$$\lambda_{i,q_{n+1}}(\alpha, \beta_j) - \lambda_{i,Q}(\alpha, \beta_j) = \lambda_{i,q_n - h_n \nabla G(q_n)}(\alpha, \beta_j)$$

$$-\lambda_{i,Q}(\alpha, \beta_j) = F_{i,\alpha,\beta_j} + \int_0^1 q_n ds - h_n \int_0^1 \nabla G(q_n(s)) ds$$

$$+a_{q_{n+1},j,i} - \lambda_{i,Q}(\alpha, \beta_j) = [\lambda_{i,q_n}(\alpha, \beta_j) - \lambda_{i,Q}(\alpha, \beta_j)]$$

$$-h_n \int_0^1 \nabla G(q_n(s)) ds + (a_{q_{n+1},j,i} - a_{q_n,j,i}),$$

and so we have

$$G(q_n - h_n \nabla G(q_n)) = \sum_{(i,j)\in I} \{\omega_{i,j}\big([\lambda_{i,q_n}(\alpha, \beta_j) - \lambda_{i,Q}(\alpha, \beta_j)] - h_n \int_0^1 \nabla G(q_n(s)) ds\big)^2\},$$

by neglecting the term $A_{i,j,n} = a_{q_{n+1},j,i} - a_{q_n,j,i}$. Thus by differentiating $G(q_{n+1})$ with respect to the parameter h_n and then equating it zero, we obtain

$$\frac{d}{dh_n} G(q_{n+1}) = -2 \int_0^1 \nabla G(q_n) ds \Big[\sum_{(i,j)\in I} \{\omega_{i,j} \times \big([\lambda_{i,q_n}(\alpha, \beta_j) - \lambda_{i,Q}(\alpha, \beta_j)] - h_n \int_0^1 \nabla G(q_n) ds\big)\} \Big],$$

and

$$h_n = \frac{\sum_{(i,j)\in I} \{\omega_{i,j}[\lambda_{i,q_n}(\alpha, \beta_j) - \lambda_{i,Q}(\alpha, \beta_j)]\}}{\sum_{(i,j)\in I} \left\{\omega_{i,j} \int_0^1 \nabla G(q_n) ds\right\}}.$$

Assume $\omega_{i,j} = 1$. Since $\int_0^1 g^2_{i,q_n}(\alpha, \beta_j) = 1$,

$$h_n = \frac{\sum\limits_{(i,j)\in I}(\lambda_{i,q_n}(\alpha,\beta_j)-\lambda_{i,Q}(\alpha,\beta_j))}{2k\int\limits_0^1 \nabla G(q_n)ds}$$
$$= \frac{\sum\limits_{(i,j)\in I}\{(\lambda_{i,q_n}(\alpha,\beta_j)-\lambda_{i,Q}(\alpha,\beta_j))\int\limits_0^1 g^2_{i,q_n}(\alpha,\beta_j)\}}{2k\int\limits_0^1 \nabla G(q_n)ds}$$
$$= \frac{\int\limits_0^1[\sum\limits_{(i,j)\in I}\{(\lambda_{i,q_n}(\alpha,\beta_j)-\lambda_{i,Q}(\alpha,\beta_j))g^2_{i,q_n}(\alpha,\beta_j)\}]ds}{2k\int\limits_0^1 \nabla G(q_n)ds}$$
$$= \frac{(1/2)\int\limits_0^1 \nabla G(q_n)ds}{2k\int\limits_0^1 \nabla G(q_n)ds} = \frac{1}{4k}.$$

It could be wondered that how would the calculation be if $A_{i,j,n}$ had not been neglected. For

$$A_{i,j,n} = a_{q_{n+1},j,i} - a_{q_n,j,i}$$
$$= (\lambda_{i,q_{n+1}}(\alpha,\beta_j)-\lambda_{i,q_n}(\alpha,\beta_j)) + h_n\int\limits_0^1 \nabla G(q_n)ds$$

by taking its derivative with respect to the parameter h_n, we obtain

$$\frac{dA_{i,j,n}}{dh_n} = \frac{d(\lambda_{i,j,q_{n+1}}-\lambda_{i,j,q_n})}{dh_n} + \int\limits_0^1 \nabla G(q_n)ds$$
$$= \frac{d\lambda_{i,j,q_{n+1}}}{dq_{n+1}}\frac{dq_{n+1}}{dh_n} - \frac{d\lambda_{i,j,q_n}}{dq_n}\frac{dq_n}{dh_n} + \int\limits_0^1 \nabla G(q_n)ds$$
$$= g^2_{i,j,q_n}\nabla G(q_{n-1}) - g^2_{i,j,q_{n+1}}\nabla G(q_n) + \int\limits_0^1 \nabla G(q_n)ds.$$

Since different expressions of $g^2_{i,j,q_n}\nabla G(q_{n-1}) - g^2_{i,j,q_{n+1}}\nabla G(q_n)$ for all $\{i,n\}$ cannot be equal to the same constant $-\int\limits_0^1 \nabla G(q_n)ds$ when $j=1$ and $j=2$, $\frac{dA_{i,j,n}}{dh_n}$ as a multiplier does not affect our computation. The value of the n^{th} iterative potential which makes this multiplier zero shows that we have already reached the global minimum at $(n-1)th$ iteration. So there is no need to calculate an extra iteration. Besides, $A_{i,j,n}$ remains in the other factor in the equality of $\frac{d}{dh_n}G(q_{n+1})$. This means that $(\sum\limits_{(i,j)\in I} A_{i,j,n})/(2k\int\limits_0^1 \nabla G(q_n)ds)$ is added to the value of the parameter h_n which is determined for each n when $\omega_{i,j}=1$. Once $A_{i,j,n}$ and the gradient $\nabla G(q_n)$ are considered, it is easy to see that it does not make a difference except very small values of integral $\int\limits_0^1 \nabla G(q_n)ds$. We observe from our experiments in calculations of numerical examples that this value does not have an inhibiting effect on the downsizing of $G(q_n)$ to about 10^{-18}. Because we approach the original potential Q by using smooth iterative potentials q_n, the small effect of this ratio does not change even if the original potential is not smooth. It should be noted that this neglect has no effect on the eigenvalues or the eigenfunctions. It just affects the number of iteration in the calculation.

2.2. McLaughlin-Rundell's data

In this part, we use the data handled by McLaughlin and Rundell in their article [19]. We consider the reconstruction of the potential q when we have a first few eigenvalues for a spectrum by making some modification in their uniqueness theorem. Firstly, we assume that the boundary conditions can be changed sufficiently, i.e., let us consider the problem

$$\begin{aligned} -y''(x)+(\lambda+q(x))y(x)=0, \quad 0<x<1, \\ y(0)=0, \quad y'(1)+\beta_k y(1)=0, \end{aligned} \tag{5}$$

where $q\in L_2(0,1)$ and β_k for $k=1,2,...$ are distinct real numbers. Let $\lambda_j(q,\beta)$ be the jth eigenvalue of (5) for β instead of β_k.

Theorem 2. *Consider the problem (5). Let $q_1,q_2\in L_2(0,1)$ and j is a fixed positive integer. Suppose that $\lambda_j(q_1,\beta_k)=\lambda_j(q_2,\beta_k)$ for $k=1,2,...$, then $q_1=q_2$ almost everywhere [19].*

Theorem 3. *Suppose that $\{\beta_{k_n}\}$ are disjoint sets such that $\{\beta_k\}_{k=1}^{\infty}=\bigcup\limits_n\{\beta_{k_n}\}$. Let j_n be a fixed positive integer, $q_1,q_2\in L_2(0,1)$ and $\lambda_{j_n,k_n}(q)=\lambda_{j_n}(q,\beta_{k_n})$ for each n. If $\lambda_{j_n,k_n}(q_1)=\lambda_{j_n,k_n}(q_2)$ for each $k_n,n=1,2,...,n_0$, then $q_1=q_2$ almost everywhere.*

Proof. Since $\{\beta_k\}_{k=1}^{\infty}=\bigcup\limits_n\{\beta_{k_n}\}$, the set $\{\lambda_{k_n,j_n}\}$ is an infinite bounded set of real numbers. Therefore it has at least one finite accumulation point. The rest of this proof is similar as in the proof of Theorem 2. □

Taking $k=k_n$ for each n allows us an extension for Theorem 3. So the eigenvalues can be used more than one for each spectrum. Because the theory is similar to those in the two spectra case (see [12]), we do not give it again here.

2.3. One spectrum and set of the terminal velocities

In this part, we use finite subsets of one spectrum $\{\lambda_n\}_{n\geq 1}$ and the set of the terminal velocities $\kappa_n = \log\left|\frac{g_n'(1,q)}{g_n'(0,q)}\right|$ as uniqueness data for Dirichlet problem. It is easy to see that the key is to have derivative of λ_n and κ_n with respect to potential q_n in the steepest descent method. To see the details and to generalize the problem to all separable conditions, one can read the nice book written by Pöschel and Trubowitz [21].

Let us consider the Sturm-Liouville problem with Dirichlet conditions

$$-y''(x) + (\lambda + q(x))y(x) = 0, \qquad 0 \leq x \leq 1, \quad (6)$$

$$y(0) = 0 \quad , \quad y(1) = 0, \quad (7)$$

where $q \in L_2[0,1]$. Let $y_1(x,\lambda,q)$ and $y_2(x,\lambda,q)$ be the solutions of (6) satisfying the initial conditions

$$y_1(0,\lambda,q) = y_2'(0,\lambda,q) = 1,$$
$$y_2(0,\lambda,q) = y_1'(0,\lambda,q) = 0.$$

They are the fundamental solutions. It means that any solutions of (6) satisfying the initial conditions $y(0) = a$, $y'(0) = b$ can be written by using these solutions as follows [21]:

$$y(x) = ay_1(x) + by_2(x).$$

So from condition $y(0) = 0$ in (7), the normalized eigenfunctions and the terminal velocities can be obtain as follows [21]:

$$g_n = g(\lambda_n, x) = \frac{y_2(x,\lambda_n)}{\|y_2(x,\lambda_n)\|_{L_2}},$$
$$\kappa_n = \log|y_2'(1,\lambda_n)| = \log[(-1)^n y_2'(1,\lambda_n)].$$

Theorem 4. *Each κ_n, $n \geq 1$, is a compact, real analytic function on L_2 with asymptotic behaviour*

$$\kappa_n(q) = \frac{1}{2n\pi}\int_0^1 \sin(2n\pi x)q(x)dx + O\big(\frac{1}{n^2}\big).$$

Its gradient is

$$\frac{\partial \kappa_n}{\partial q}(x) = y_1(x,\lambda_n)y_2(x,\lambda_n) - [a_n]g_n^2(x)$$
$$= \frac{\sin(2n\pi x)}{2n\pi} + O\big(\frac{1}{n^2}\big),$$

where $[a_n] = \int_0^1 y_1(t,\lambda_n)y_2(t,\lambda_n)dt$ [21].

Theorem 5. *$\kappa \times \lambda$ is one-to-one on L_2 where $\kappa(q) = (\kappa_1(q), \kappa_2(q), ...)$ and $\lambda(q) = (\lambda_1(q), \lambda_2(q), ...)$ [21].*

For these data, the new objective functional and its gradient become

$$G(q) = \sum_{i\in I} \omega_i\big((\lambda_{i,q} - \lambda_{i,Q})^2 + (\kappa_{i,q} - \kappa_{i,Q})^2\big),$$

and

$$\nabla G(q) = 2\sum_{i\in I} \omega_i\{(\lambda_{i,q} - \lambda_{i,Q})g_i^2(x)$$
$$+(\kappa_{i,q} - \kappa_{i,Q})\frac{\partial \kappa_i}{\partial q}(x)\big\},$$

where $I \subset \mathbb{N}$, respectively. Since $y_{1,n} = y_1(x,\lambda_n)$ and $y_{2,n} = y_2(x,\lambda_n)$ are the solutions to eqn. (6), $\nabla G(q)$ is in H^1. It is obvious that $0 = G(Q) < G(q)$ for $q \neq Q \in L_2[0,1]$. In other words, $Q(x) \in L_2[0,1]$ is the global minimum for $G(q)$.

Now, let us consider the bilinear form $\Gamma : H^1 \times H^1 \to \mathbb{R}$ with $\Gamma(f,g) = \int_0^1 [f,g]dx$ where $[\cdot,\cdot]$ is the Wronskian operator such that $[f,g] = f(x)g'(x) - f'(x)g(x)$ for the differentiable functions $f, g : [0,1] \to \mathbb{R}$. This transformation is bounded by $\|f\|_{H^1}\|g\|_{H^1}$, that is, $|\Gamma(f,g)| \leq \|f\|_{H^1}\|g\|_{H^1}$. In particular Γ is continuous on H^1 [12]. Also, it is easy to see that Γ is antisymmetric because of the Wronskian. Some properties for the Wronskian are given as follows [12]:

***i*:** $[fg, FG] = fF[g,G] + gG[f,F]$ for differentiable functions f, g, F, G.

***ii*:** For two arbitrary solutions f_1 and f_2 of the eqn. (6) with eigenvalue parameters λ_1 and λ_2 we have $f_1 f_2 = \frac{1}{\lambda_1 - \lambda_2}[f_1, f_2]'$.

Since $y_1(x,\lambda)$ and $y_2(x,\lambda)$ satisfy the eqn. (6), we obtain

$$\frac{d}{dx}[y_1, y_2] = \frac{d}{dx}(y_1 y_2' - y_1' y_2)$$
$$= y_1 y_2(q - \lambda) - y_1 y_2(q - \lambda) = 0,$$

and so we have

$$[y_1, y_2] = y_1(0)y_2'(0) - y_1'(0)y_2(0) = 1. \quad (8)$$

Lemma 1. *Let $\{\lambda_n\}_{n\geq 1}$ be spectrum of the Sturm-Liouville problem (6)-(7). Then the following properties are satisfied:*

i: $\Gamma(g_n^2, g_m^2) = 0$.
ii: $\Gamma(y_{1n}y_{2n}, y_{1m}y_{2m}) = 0$.
iii: $\Gamma(y_{1n}y_{2n}, g_m^2) = \begin{cases} 1 & , \quad m = n \\ 0 & , \quad m \neq n \end{cases}$.

Proof. i) A more general proof was given by Röhrl [12].
ii) It is obvious that $\Gamma(y_{1n}y_{2n}, y_{1m}y_{2m}) = 0$ for $m = n$. For $m \neq n$, we obtain

$$\begin{aligned}\Gamma(y_{1n}y_{2n}, y_{1m}y_{2m}) &= \int_0^1 (y_{1n}y_{1m}[y_{2n}, y_{2m}] \\ &\quad + y_{2n}y_{2m}[y_{1n}, y_{1m}])dx \\ &= \frac{1}{\lambda_n - \lambda_m}\int_0^1 ([y_{1n}, y_{1m}]'[y_{2n}, y_{2m}] \\ &\quad + [y_{2n}, y_{2m}]'[y_{1n}, y_{1m}])dx \\ &= \frac{1}{\lambda_n} - \lambda_m \int_0^1 \frac{d}{dx}([y_{1n}, y_{1m}][y_{2n}, y_{2m}])dx \\ &= \frac{1}{\lambda_n - \lambda_m}[y_{1n}, y_{1m}][y_{2n}, y_{2m}]\big|_{x=0}^{x=1} = 0,\end{aligned}$$

by using Dirichlet conditions. ***iii)*** For $m \neq n$,

$$\begin{aligned}\Gamma(y_{1n}y_{2n}, g_m^2) &= \frac{1}{\|y_{2m}\|_{L_2}^2}\int_0^1 [y_{1n}y_{2n}, y_{2m}y_{2m}]dx \\ &= \frac{1}{\|y_{2m}\|_{L_2}^2}\int_0^1 (y_{1n}y_{2m}[y_{2n}, y_{2m}] + y_{2n}y_{2m}[y_{1n}, y_{2m}])dx \\ &= \frac{1}{\|y_{2m}\|_{L_2}^2(\lambda_n - \lambda_m)}\int_0^1 \frac{d}{dx}([y_{1n}, y_{2m}][y_{2n}, y_{2m}])dx \\ &= \frac{([y_{1n}, y_{2m}][y_{2n}, y_{2m}])\big|_0^1}{\|y_{2m}\|_{L_2}^2(\lambda_n - \lambda_m)} = 0.\end{aligned}$$

For $m = n$, because of $[y_{2n}, y_{2m}] = 0$ and in the view of (8), we have

$$\begin{aligned}\Gamma(y_{1n}y_{2n}, g_n^2) &= \frac{1}{\|y_{2n}\|_{L_2}^2}\int_0^1 (y_{2n}y_{2n}[y_{1n}, y_{2n}])dx \\ &= \int_0^1 g_n^2[y_{1n}, y_{2n}]dx = \int_0^1 g_n^2 dx = 1.\end{aligned}$$

□

Corollary 1. *For all $m, n \in \mathbb{N}$,*

i: $\Gamma\big(\frac{\partial\kappa_n}{\partial q}, \frac{\partial\kappa_m}{\partial q}\big) = 0$.
ii: $\Gamma(\frac{\partial\kappa_n}{\partial q}, g_m^2) = \begin{cases} 1 & , \quad m = n \\ 0 & , \quad m \neq n \end{cases}$.

Proof. Let $m, n \in \mathbb{N}$.
i)

$$\begin{aligned}\Gamma\big(\frac{\partial\kappa_n}{\partial q}, \frac{\partial\kappa_m}{\partial q}\big) &= \Gamma(y_{1n}y_{2n} + [a_n]g_n^2, y_{1m}y_{2m} + [a_m]g_m^2) \\ &= \Gamma(y_{1n}y_{2n}, y_{1m}y_{2m}) + [a_m]\Gamma(y_{1n}y_{2n}, g_m^2) \\ &\quad + [a_n]\Gamma(g_n^2, y_{1m}y_{2m}) + [a_n][a_m]\Gamma(g_n^2, g_m^2).\end{aligned}$$

So we see that

$$\Gamma\big(\frac{\partial\kappa_n}{\partial q}, \frac{\partial\kappa_n}{\partial q}\big) = [a_n] - [a_n] = 0,$$

and

$$\Gamma\big(\frac{\partial\kappa_n}{\partial q}, \frac{\partial\kappa_m}{\partial q}\big) = 0,$$

by considering Lemma 1 for $m = n$ and $m \neq n$, respectively.
ii) From Lemma 1, it can be concluded that

$$\begin{aligned}\Gamma\big(\frac{\partial\kappa_n}{\partial q}, g_m^2\big) &= \Gamma(y_{1n}y_{2n}, g_m^2) - [a_n][g_n^2, g_m^2] \\ &= \begin{cases} 1 & , \quad m = n \\ 0 & , \quad m \neq n \end{cases}.\end{aligned}$$

□

Theorem 6. *The set $\{g_n^2\}_{n\geq 1} \cup \{\frac{\partial\kappa_n}{\partial q}\}_{n\geq 1}$ is linearly independent in H^1.*

Proof. Assume, to the contrary, that it is not true; that is,

$$g_n^2 = \sum_{k\in\mathbb{M}} b_k g_k^2 + \sum_{m\in\mathbb{M}\cup\{n\}} c_m \frac{\partial\kappa_m}{\partial q}$$

and

$$\frac{\partial\kappa_n}{\partial q} = \sum_{k\in\mathbb{M}} b_k \frac{\partial\kappa_k}{\partial q} + \sum_{m\in\mathbb{M}\cup\{n\}} c_m g_k^2,$$

for some fixed n $(n \neq k)$, where b_k, c_m are real numbers and $\mathbb{M} \subset \mathbb{N}$. Then by Corollary 1 and Lemma 1, we have the contradictions

$$\begin{aligned}1 &= \Gamma\big(\frac{\partial\kappa_n}{\partial q}, g_n^2\big) \\ &= \Gamma\big(\frac{\partial\kappa_n}{\partial q}, \sum_{k\in\mathbb{M}} b_k g_k^2\big) + \Gamma\big(\frac{\partial\kappa_n}{\partial q}, \sum_{m\in\mathbb{M}\cup\{n\}} c_m \frac{\partial\kappa_m}{\partial q}\big) \\ &= \sum_{k\in\mathbb{M}} b_k \Gamma\big(\frac{\partial\kappa_n}{\partial q}, g_k^2\big) + \sum_{m\in\mathbb{M}\cup\{n\}} c_m \Gamma\big(\frac{\partial\kappa_n}{\partial q}, \frac{\partial\kappa_m}{\partial q}\big) = 0,\end{aligned}$$

and

$$
\begin{aligned}
-1 &= \Gamma\big(g_n^2, \frac{\partial \kappa_n}{\partial q}\big) \\
&= \Gamma\big(g_n^2, \sum_{k\in\mathbb{M}} b_k \frac{\partial \kappa_k}{\partial q}\big) + \Gamma\big(g_n^2, \sum_{m\in\mathbb{M}\cup\{n\}} c_m g_m^2\big) \\
&= \sum_{k\in\mathbb{M}} b_k \Gamma\big(g_n^2, \frac{\partial \kappa_k}{\partial q},\big) \\
&+ \sum_{m\in\mathbb{M}\cup\{n\}} c_m \Gamma\big(g_n^2, g_m^2\big) = 0.
\end{aligned}
$$

These complete the proof. □

Theorem 7. *If I is finite or $i\omega_i$ is summable, the functional $G(q)$ has no local minima at q with $G(q) > 0$. In other words, $G(q) = 0 \Leftrightarrow \nabla G(q) = 0$.*

Proof. ($\Leftarrow$:) If $G(q) = 0$, then it is obvious that $\nabla G(q) = 0$. ($\Rightarrow$:) If $\nabla G(q) = 0$, then it is concluded that $\lambda_{i,q} - \lambda_{i,Q} = 0$ and $\kappa_{i,q} - \kappa_{i,Q} = 0$ from Theorem 6. Therefore, $G(q) = 0$. □

The computation to find the parameter h_n is similar to that in Section 2.1. Since

$$
\begin{aligned}
G(q_{n+1}) \cong \sum_{i=1}^{k} \Big\{ &\big(\lambda_{i,q_n} - \lambda_{i,Q} - h_n \int_0^1 \nabla G(q_n) dx\big)^2 \\
&\big(\kappa_{i,q_n} - \kappa_{i,Q} - h_n \int_0^1 \frac{\sin(2i\pi x)\nabla G(q_n)}{2i\pi} dx\big)^2 \Big\},
\end{aligned}
$$

we obtain

$$
\begin{aligned}
\sum_{i=1}^{k} \Big\{ &(\lambda_{i,q_n} - \lambda_{i,Q}) \int_0^1 \nabla G(q_n) dx \\
&+ (\kappa_{i,q_n} - \kappa_{i,Q}) \int_0^1 \frac{\nabla G(q_n)\sin(2i\pi x)}{2i\pi} dx \\
&- h_n \big[(\int_0^1 \nabla G(q_n) dx)^2 + (\int_0^1 \frac{\nabla G(q_n)\sin(2i\pi x)}{2i\pi} dx)^2 \big] \Big\} = 0,
\end{aligned}
$$

and consequently

$$
h_n \cong \frac{I_{1n}}{I_{2n}} \tag{9}
$$

where

$$
I_{1n} = \sum_{i=1}^{k} \Big\{ (\lambda_{i,q_n} - \lambda_{i,Q}) \int_0^1 \nabla G(q_n) dx + (\kappa_{i,q_n} - \kappa_{i,Q}) \int_0^1 \frac{\nabla G(q_n)\sin(2i\pi x)}{2i\pi} dx \Big\}
$$

and

$$
I_{2n} = \sum_{i=1}^{k} \Big\{ (\int_0^1 \nabla G(q_n) dx)^2 + (\int_0^1 \frac{\nabla G(q_n)\sin(2i\pi x)}{2i\pi} dx)^2 \Big\}
$$

Because of $\|g_{i,q_n}\|_{L_2}^2 = 1$,

$$
\begin{aligned}
\int_0^1 \frac{\partial \kappa_{i,q_n}}{\partial q_n} dx &= \int_0^1 y_{1i}(x,q_n) y_{2i}(x,q_n) \\
&- [a_i] \int_0^1 g_{i,q_n}^2 dx = 0,
\end{aligned}
$$

and therefore

$$
\int_0^1 \nabla G(q_n) dx = 2 \sum_{i=1}^{k} (\lambda_{i,q_n} - \lambda_{i,Q}). \tag{10}
$$

On the other hand, for all $m, n \in \mathbb{N}$ since

$$
\int_0^1 \sin(2m\pi x)(1 - \cos(2n\pi x)) dx = 0,
$$

and

$$
\int_0^1 \sin(2m\pi x)\sin(2n\pi x) dx = \begin{cases} 1/2 & , \quad m = n \\ 0 & , \quad m \neq n \end{cases},
$$

we find

$$
\int_0^1 \Big(\frac{\nabla G(q_n)\sin(2i\pi x)}{2i\pi} \Big) dx \cong \frac{\kappa_{i,q_n} - \kappa_{i,Q}}{4i^2\pi^2}, \tag{11}
$$

for $1 \leq i \leq k$ by using asymptotic formulas. Finally, by substituting (10) and (11) in (9), we obtain

$$
h_n \cong \frac{2\Big(\sum_{i=1}^{k} (\lambda_{i,q_n} - \lambda_{i,Q})\Big)^2 + \sum_{i=1}^{k} \frac{(\kappa_{i,q_n} - \kappa_{i,Q})^2}{4i^2\pi^2}}{4\Big(\sum_{i=1}^{k} (\lambda_{i,q_n} - \lambda_{i,Q})\Big)^2 + \sum_{i=1}^{k} \frac{(\kappa_{i,q_n} - \kappa_{i,Q})^4}{(2i\pi)^4}}.
$$

By the theory set out in this section, numerical calculations can be performed for three different data sets.

3. Numerical experiments

In this study numerical computations are obtained using Mathematics 11 with Eigen package [24]. The calculations are conducted on a desktop PC with a processor of Intel(R) Core i5-3470 CPU @3.2 GHz for Windows 7. We consider the initial potential $q_0(x) = 0$ and the test potentials $Q^1(x), Q^2(x), Q^3(x)$ as follows:

$$Q^1(x) = 75.16x^6 - 176.44x^5 + 129.35x^4 - 30.67x^3 + 2.6x^2 + 0.001x,$$

$$Q^2(x) = \begin{cases} -35.2x^2 + 17.6x & , 0 \le x < 0.25 \\ 35.2x^2 - 35.2x + 8.8 & , 0.25 \le x < 0.75 \\ 32.5x^2 + 52.8x - 17.6 & , 0.75 \le x \le 1 \end{cases},$$

$$Q^3(x) = \begin{cases} 0 & , \quad 0 \le x < 0.1 \\ 7x - 0.7 & , \quad 0.1 \le x < 0.3 \\ 3.5 - 7x & , \quad 0.3 \le x < 0.5 \\ 0 & , \quad 0.5 \le x < 0.7 \\ 4 & , \quad 0.7 \le x < 0.9 \\ 2 & , \quad 0.9 \le x \le 1 \end{cases},$$

which were used in [11–13].

3.1. Numerical results for two spectra

To obtain two spectra data we choose $\alpha = 0$, $\beta_1 = 0$ and $\beta_2 = \frac{\pi}{2}$ in the S-L problem (1). The steepest descent method is used to find the potential function for two spectra numerically via the methodology exhibited in Section 2.1. The numerical results that are calculated by our effective algorithm written in Mathematica are demonstrated in figures in which dashed curves show the numerical results while the solid lines show the exact values of potential function. We obtain these results in 381.59 and 3669.09 seconds with 41 and 110 iterations for the first two graphics in Fig. 1, respectively. We have also handled a few different test potential and different boundary conditions. We observed that although the number of iteration or the cost of computation of CPU may increase, there isn't any remarkable difference. However, a very small difference is occurred when we just consider two pairs data. In the numerical calculations of the potentials, the accuracy of $G(q_n) \approx 10^{-6}$ is taken, and Fig. 1 and Fig. 2 are drawn according to this. We also take into account noise for five pairs data by adding random numbers in the interval $(-0.01, 0.01)$ to each eigenvalue, it is demonstrated in Fig. 1c. For uniform noise, we have not seen a remarkable difference. The distribution of $G(q_n)$ as n increases is shown by Fig. 1d.

3.2. Numerical results for McLaughlin-Rundell's data

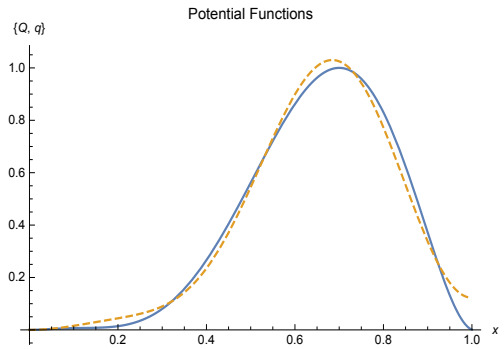
To obtain data, we choose $\beta_k = \tan\left(\frac{\pi}{k^2}\right)$ in the S-L problem (5). Similar to the way that is followed in Section 3.1, to approximate test potential functions with McLaughlin-Rundell's data, we use the methodology given in Section 2.2 with the same algorithm in Section 2.1. The upper bound of the value of $G(q_n)$ is approximately 5×10^{-5}. The numerical values of q for $Q^1(x)$ and $Q^2(x)$ are demonstrated in (a), (b), (c) and (d) of Fig. 3 with the dashed curves while the solid lines show the exact values of test potentials Q^1 and Q^2. Fig. 3c is plotted to show the effectiveness of our extension on McLaughlin-Rundell's data which is plotted in Fig. 3b. It can be seen also that numerical results in Fig. 3c are very close to test potential and are better than of those in Fig. 3b. It is concluded that the numerical result we obtained is more accurate for smooth test potential $Q^1(x)$. However, it can be observed from Fig. 3d that for other test potentials we need more than two eigenvalues in calculations to have better approximation. Because we have a few first eigenvalues in real life problems, we can say that it will be more appropriate to prefer extra eigenvalues as auxiliary data with original data.

3.3. Numerical results for one spectrum and set of the terminal velocities

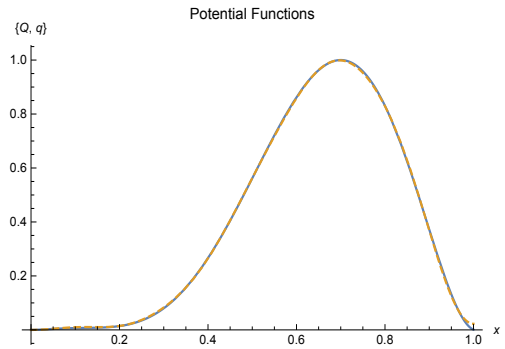
In this section, we consider a finite set consisted of the elements from one spectrum and terminal velocities of the S-L problem (6)- (7) to approximate the test potentials $Q^1(x)$ and $Q^2(x)$. The upper bound of the value of $G(q_n)$ is approximately 10^{-6} in calculations. By using the analogue Mathematica program performed in Section 3.1, we obtain the numerical results which are displayed in Fig. 4. When this data is compared with the first data considered in Section 3.1, it is seen from Fig. 4 that there is no remarkable difference for test potentials Q^1 and Q^2. However when our computer program runs to obtain required numerical results for Q^3, the calculation needs some time to get the same results in Section 3.1.

4. Conclusion

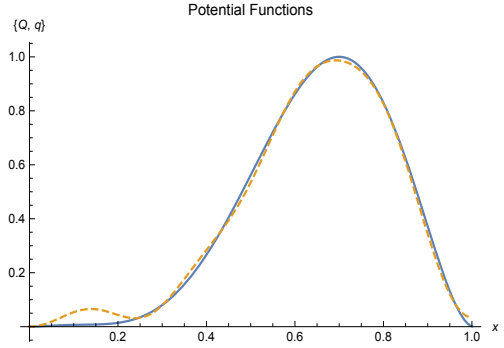
The inverse Sturm-Liouville reconstruction problem is an interesting subject that arises in many physical phenomena and it involves the reconstruction of coefficient functions and boundary conditions by aid of some data which determine these functions uniquely. In this study, we consider inverse S-L problem in the normal form and reconstruct the potential function. Actually, one of the main goal of our study is to eliminate the obligation of using the asymptotics for $\lambda_{i,q_n}(\alpha,\beta), g^2_{i,j,q_n}, \kappa_n(q_n)$, and $\frac{\partial \kappa_n}{\partial q_n}$ in the steepest descent algorithm by estimating the parameter h_n which appears in each iteration when the program runs. The other is to apply our approach to three



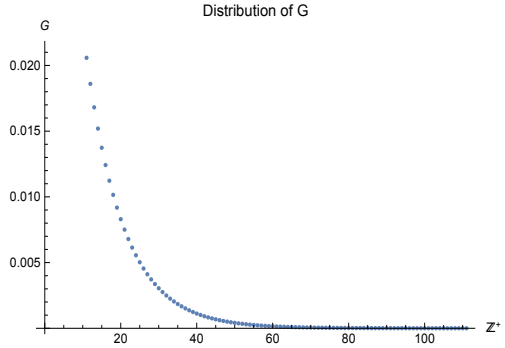
(a) Two pairs of eigenvalues



(b) Five pairs of eigenvalues

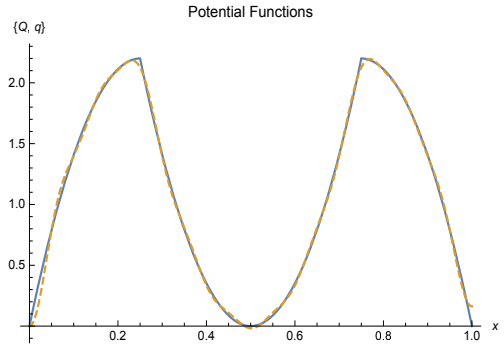


(c) Random noise for five pairs data

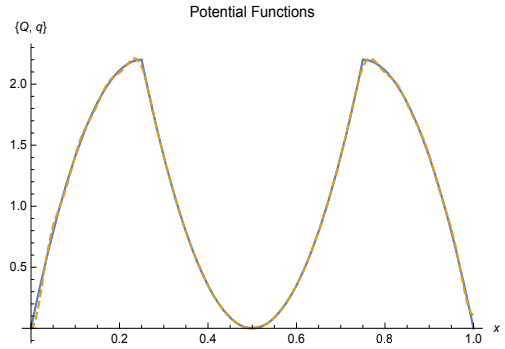


(d) The distribution of G for five pairs data

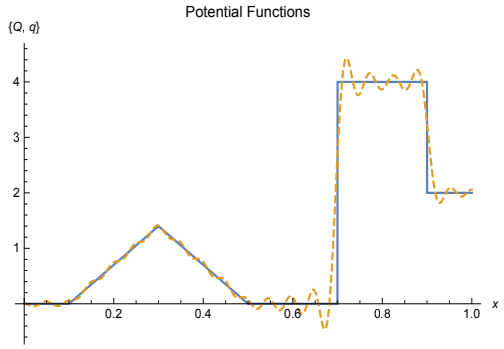
Figure 1. The graphics (a), (b), and (c) show the numerical results for $Q^1(x)$. The graphic (d) shows the distribution of $G(q)$ for $Q^1(x)$.



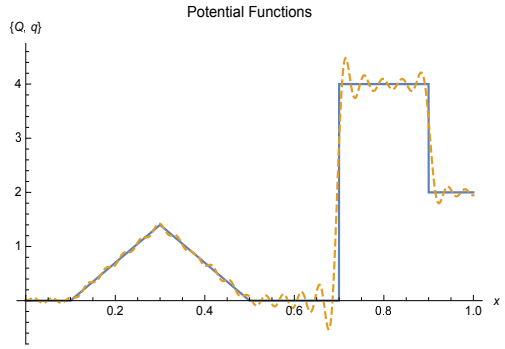
(a) Ten pairs of eigenvalues



(b) Fifteen pairs of eigenvalues



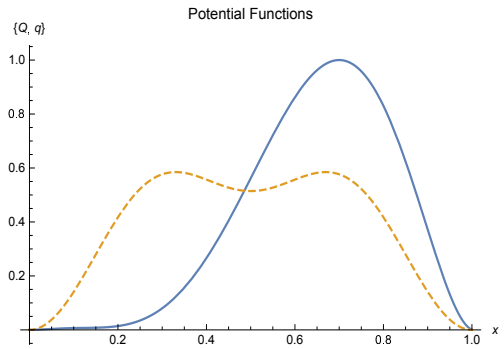
(c) Twenty pairs of eigenvalues



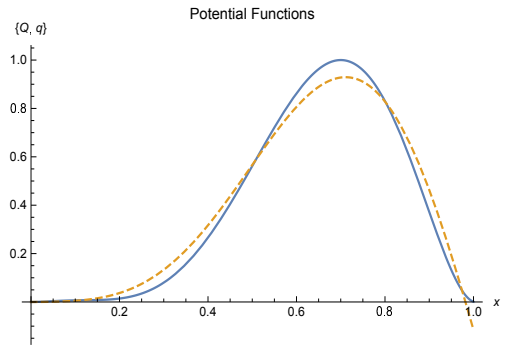
(d) Twenty five pairs of eigenvalues

Figure 2. The graphics obtained for $Q^2(x)$ in (a), (b) and $Q^3(x)$ in (c), (d).

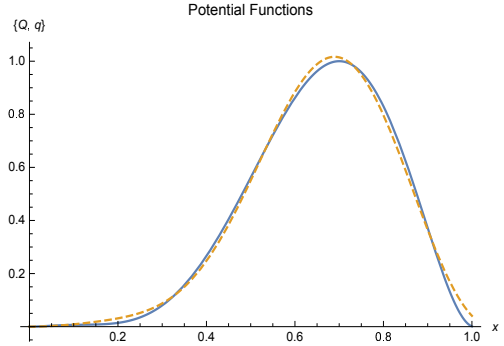
different types of data which are described in Section 2.1, 2.2 and 2.3. It is seen in the literature that nearly all studies on McLaughlin-Rundell data which is described in Section 2.2 deal with minor generalizations of the uniqueness result and say nothing about the numerical solution of this



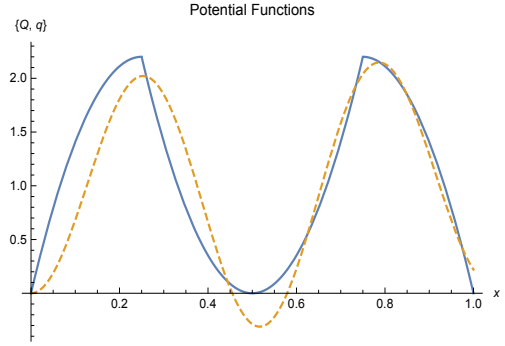
(a) First two eigenvalues from one spectrum



(b) First eigenvalues from five spectra

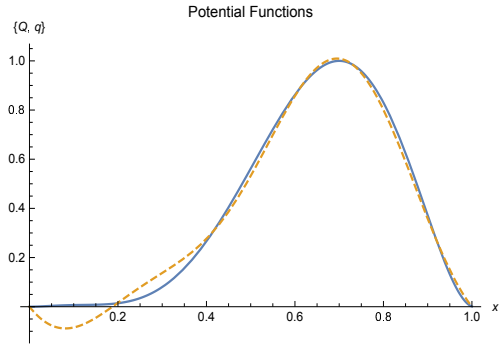


(c) First two eigenvalues from five spectra

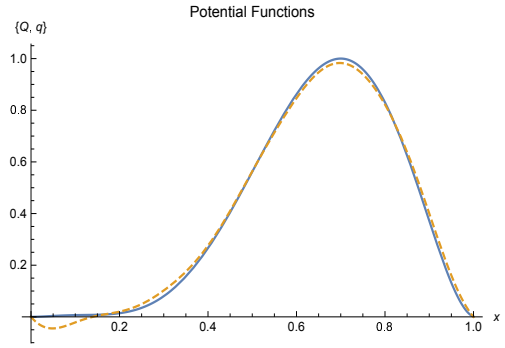


(d) First two eigenvalues from ten spectra

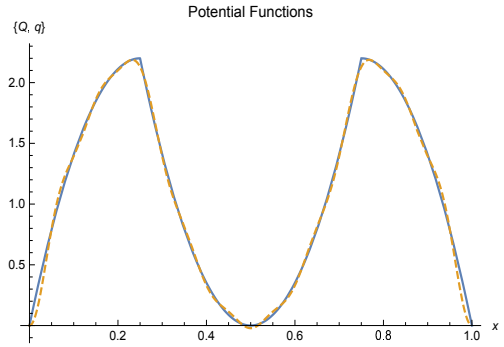
Figure 3. The graphics obtained for $Q^1(x)$ in (a), (b), (c) and $Q^2(x)$ in (d).



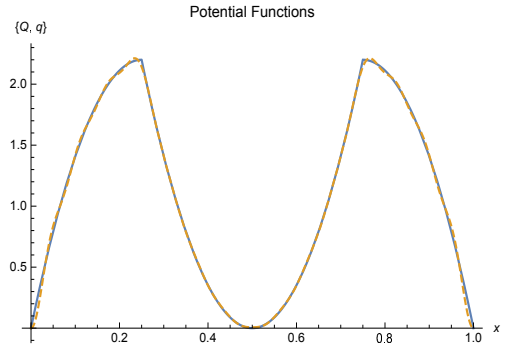
(a) Two pairs data for the Q^1



(b) Five pairs data for the Q^1



(c) Ten pairs data for the Q^2



(d) Fifteen pairs data for the Q^2

Figure 4. The graphics obtained for $Q^1(x)$ in (a), (b) and $Q^2(x)$ in (c), (d).

problem. Therefore, we think especially this part is worth considering.

Although the reconstruction of the potential by using alternative methods for the data in Section 2.3 appears in the literature, proposed approach in Section 2.1 is firstly considered here. The obtained numerical results are given in Section 3. It can be concluded that the approach for two spectra in Section 3.1 is more effective than other approaches for other data in Section 3.2 and Section 3.3.

The theory in this study can be generalized to other forms of the S-L problems, and it can be used not only for the reconstruction of potential function but also for the recovery of boundary conditions. Recently, there have been studies on fractional Sturm-Liouville problems [25, 26]. We think that our approaches will also be useful for fractional Sturm-Liouville problems.

Acknowledgments

The author thanks the anonymous reviewers.

References

[1] Hald, O.H. (1978). Sturm-Liouville problem and the Rayleigh-Ritz method. *Math. Comp.*, 32, 687–705.

[2] Paine, J. (1984). A Numerical method for the inverse Sturm-Liouville problem. *SIAM J. Sci. Stat.Comput.*, 5, 149–156.

[3] Sacks, P.E. (1988). An iterative method for the inverse Dirichlet problem. *Inverse Problems*, 4, 1055–1069.

[4] Lowe, B.D., Pilant, M., & Rundell, W. (1992). The recovery of potentials from finite spectral data. *SIAM J. Math. Anal.*, 23, 482–504.

[5] Rundell, W., Sacks, P.E. (1992). Reconstruction techniques for classical inverse Sturm-Liouville problems. *Math. Comp.*, 58, 161–183.

[6] Neher, M. (1994). Enclosing solutions of an inverse Sturm-Liouville problem with finite data. *Computing*, 53, 379–395.

[7] Fabiano, R.H., Knobel, R., & Lowe, B.D. (1995). A finite-difference algorithm for an Sturm-Liouville problem. *IMA J. Num. Anal.*, 15 , 75–88.

[8] Andrew, A.L. (2004). Numerical solution of inverse Sturm-Liouville problems. *Anziam J.*, 45, C326–C337.

[9] Andrew, A.L. (2005). Numerov's method for inverse Sturm-Liouville problems. *Inverse Problems*, 21, 223–238.

[10] Andrew, A.L. (2011). Finite difference methods for half inverse Sturm-Liouville problems. *App. Math. and Comp.*, 218, 445–457.

[11] Brown, B.M., Samko, V.S., Knowles, I.W., & Marletta, M. (2003). Inverse spectral problem for the Sturm-Liouville equation. *Inverse Problems*, 19, 235–252.

[12] Röhrl, N. (2005). A least-squares functional for solving inverse Sturm-Liouville problems. *Inverse Problems*, 21, 2009–2017.

[13] Röhrl, N. (2006). Recovering boundary conditions in inverse Sturm-Liouville problems. Recent advances in differential equations and Mathematical physics, *Contemp. Math., Amer. Math. Soc.*, Providence, RI, 412, 263–270.

[14] Rafler, M., Böckmann, C. (2007). Reconstruction method for inverse Sturm-Liouville problems with discontinuous potentials. *Inverse Problems*, 23, 933–946.

[15] Kammanee, A., Böckmann, C. (2009). Boundary value method for inverse Sturm-Liouville problems. *Appl. Math. Comput.*, 214, 342–352.

[16] Ghelardoni, P., Magherini,C. (2010). BVMs for computing Sturm-Liouville symmetric potentials. *App. Math. Comp.*, 217, 3032–3045.

[17] Gao, Q., Huang, Z., & Cheng, X. (2015). A finite difference method for an inverse Sturm-Liouville problem in impedance form. *Numer. Algor.*, 70, 669–690.

[18] Tuz, M. (2017). Boundary values for an eigenvalue problem with a singular potential. *An International Journal of Optimization and Control: Theories & Applications*, 7(3), 293–300.

[19] McLaughlin, J.R., Rundell, W. (1987). A uniqueness theorem for an inverse Sturm-Liouville problem. *Math. Phys.*, 28, 1471–1472.

[20] Levinson, N. (1949). The inverse Sturm-Liouville problem. *Mat. Tideskr. B.*, 25, 25–30.

[21] Pöschel, J., Trubowitz, E. (1987). *Inverse spectral theory*. Pure and Applied Mathematics, Academic Press, Inc., Boston, MA, 130, x+192 pp, ISBN: 0-12-563040-9.

[22] Polak, E. (1997). *Optimization. Algorithms and consistent approximations*. Applied Mathematical Sciences, Springer-Verlag, New York 124, xx+779 pp, ISBN: 0-387-94971-2 297–317.

[23] Hoschtadt, H. (1973). The inverse Sturm-Liouville problem. *Commun. Pure Appl. Math.*, 26, 715–729.

[24] Squire, J. (2013). Eigenvalue differential equation solver. http://library.wolfram.com /infocenter/MathSource/8762/#downloads.

[25] Al-Mdallal, Q.M., Al-Refai, M., Syam, M., & Al-Srihin, M.K. (2018). Theoretical and computational perspectives on the eigenvalues of fourth-order fractional Sturm–Liouville problem. *International Journal of Computer Mathematics*, 95(8) , 1548–1564.

[26] Mert, R., Abdeljawad, T., & Peterson, A. (2018). A Sturm-Liouville approach for continuous and discrete Mittag-Leffler kernel

fractional operators. *Discrete and Continuous Dynamical Systems Series S*, 1–17.

Mehmet Açil *is currently an Assistant Professor of Applied Mathematics at Van Yüzüncü Yıl University (Van YYU), Türkiye since 2019. He obtained his PhD degree in applied mathematics from Van YYU in 2018 specialization in inverse Sturm-Liouville problems and the master degree in applied mathematics from Manisa Celal Bayar University in 2013. His research interests include inverse Sturm-Liouville problems, potential theory and Lie symmetries.*
https://orcid.org/0000-0002-3875-7709

Ali Konuralp *is currently an Associate Professor at Department of Mathematics, Manisa Celal Bayar University in Türkiye, since 2016. He received the master and Ph.D degree in applied mathematics at MCBU. His research interests include in numerical analysis of differential equations and fractional differential equations.*
https://orcid.org/0000-0001-9983-5742

An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.11, No.2, pp.199-205 (2021)
http://doi.org/10.11121/ijocta.01.2021.001015

RESEARCH ARTICLE

On the solutions of boundary value problems

Ali Akgül [a*], Mir Sajjad Hashemi[b] and Negar Seyfi[b]

[a] *Siirt University, Art and Science Faculty, Department of Mathematics, Siirt, Turkey*
[b] *Department of Mathematics, Basic Science Faculty, University of Bonab, Bonab, Iran*
aliakgul00727@gmail.com, hashem_math396@yahoo.com, n.seifi2017@gmail.com

ARTICLE INFO

Article History:
Received 26 August 2020
Accepted 05 January 2021
Available 12 May 2021

Keywords:
Reproducing kernel Hilbert space method
Boundary value problems
Bounded linear operator
Reproducing kernel functions

AMS Classification 2010:
47B32; 34B05

ABSTRACT

We investigate the nonlinear boundary value problems by reproducing kernel Hilbert space technique in this paper. We construct some reproducing kernel Hilbert spaces. We define a bounded linear operator to obtain the solutions of the problems. We demonstrate our numerical results by some tables. We compare our numerical results with some results exist in the literature to present the efficiency of the proposed method.

1. Introduction

We investigate the following boundary value problems by reproducing kernel method.

$$\left(p(x)y'\right)' = f(x,y) \tag{1}$$

subject to the boundary values

$$y(0) = A, \quad y(1) = B. \tag{2}$$

Reproducing kernel space is a special Hilbert space. Many problems have been investigated by reproducing kernel Hilbert space method in the literature.

Safari et al. [1] have investigated the rainfall-runoff modeling through regression in the reproducing kernel Hilbert space algorithm. Najafi et al. [2] have worked on the combining fractional differential transform method and reproducing kernel Hilbert space method to solve fuzzy impulsive fractional differential equations. Sahihi et al. [3] have searched the system of second-order boundary value problems using a new algorithm based on the reproducing kernel Hilbert space. Agud et al. [4] have investigated the weighted p-regular kernels for reproducing kernel Hilbert spaces. Mundayadan et al. [5] have studied on the linear dynamics in the reproducing kernel Hilbert spaces. Arqub et al. [6] have constructed the modulation of reproducing kernel technique successfully. Emamjome et al. [7] have presented the reproducing kernel pseudospectral technique in details. Foroutan et al. [8] have investigated this technique for the nonlinear three-point boundary value problems. Akgül et al. [9] have worked on the representation for the reproducing kernel Hilbert space method for a nonlinear system. Allahviranloo et al. [14] have investigated the reproducing kernel method to solve parabolic partial differential equations with nonlocal conditions. For more details see [15–23].

We organize our manuscript as: We construct the reproducing kernel Hilbert spaces in Section 2. We apply the reproducing kernel method in this section. We demonstrate the numerical results in Section 3. We give the conclusion in the last section.

*Corresponding Author

2. Reproducing kernel Hilbert spaces

We define the reproducing kernel Hilbert spaces and find some reproducing kernel functions in these spaces in this section.

Definition 1. *Let $(H, \langle\cdot,\cdot\rangle)$ be a Hilbert space of real functions defined on a nonempty set E. A function $K : E \times E \to \mathbb{R}$ is called a* reproducing kernel *for H if and only if*

(a) $K(\cdot, z) \in H$ *for each fixed* $z \in E$,
(b) $\langle\varphi, K(\cdot, z)\rangle = \varphi(z)$ *for all* $z \in E$ *and all* $\varphi \in H$.

We will refer to such a Hilbert space H for which there exists a reproducing kernel function K as a reproducing kernel Hilbert space.

Condition (b) is called "the reproducing property" of the kernel K because the value of an arbitrary function $\varphi \in H$ at an arbitrary point $z \in E$ is reproduced by the inner product of φ with $K(\cdot, z)$.

Definition 2. $\mathcal{W}_2^1[0,1]$ *is given as:*

$$\mathcal{W}_2^1[0,1] = \{y : y \in AC[0,1] \text{ and } y' \in L^2[0,1]\},$$

with

$$\langle y, g\rangle_{\mathcal{W}_2^1} = \int_0^1 \big[y(x)g(x) + y'(x)g'(x)\big]\mathrm{d}x, \qquad y, g \in \mathcal{W}_2^1[0,1],$$

and

$$\|y\|_{\mathcal{W}_2^1} = \sqrt{\langle y, y\rangle_{\mathcal{W}_2^1}}, \quad y \in \mathcal{W}_2^1[0,1], \tag{3}$$

as the inner product and the norm in $\mathcal{W}_2^1[0,1]$ respectively. Reproducing kernel function $T_x(y)$ of $\mathcal{W}_2^1[0,1]$ is presented as:

$$T_x(y) = \frac{1}{2\sinh(1)}\left[\cosh(x+y-1) + \cosh(|x-y|-1)\right]. \tag{4}$$

Definition 3. *The space ${}^o\mathcal{W}_2^3[0,1]$ is given by*

$${}^o\mathcal{W}_2^3[0,1] = \{y \in AC[0,1] : \ y', y'' \in AC[0,1], \ y^{(3)} \in L^2[0,1], \ y(0) = 0 = y(1)\}.$$

$$\begin{aligned}\langle y, v\rangle_{{}^o\mathcal{W}_2^3[0,1]} &= y(0)v(0) + y'(0)v'(0) + y(1)v(1) \\ &\quad + \int_0^1 y^{(3)}(x)v^{(3)}(x)\mathrm{d}x, \\ &\quad y, v \in {}^o\mathcal{W}_2^3[0,1],\end{aligned}$$

and

$$\|y\|_{{}^o\mathcal{W}_2^3[0,1]} = \sqrt{\langle y, y\rangle_{{}^o\mathcal{W}_2^3[0,1]}}, \quad y \in {}^o\mathcal{W}_2^3[0,1],$$

are the inner product and the norm in ${}^o\mathcal{W}_2^3[0,1]$ respectively.

Theorem 1. *Reproducing kernel function $y(x)$ of ${}^o\mathcal{W}_2^3[0,1]$ is given as*

$$\mathcal{R}_\xi(x) = \begin{cases} -x^2\xi - x\xi^2 + x\xi - \frac{1}{120}x^2\xi^2 + \frac{21}{20}x^2\xi^2 \\ +\frac{1}{24}x^2\xi^4 - \frac{1}{12}x^2\xi^3 \\ +\frac{1}{24}x^4\xi^2 - \frac{1}{24}x^4\xi - \frac{1}{120}x^5\xi^2 + \frac{x^5}{120}, \\ 0 \le x \le \xi \le 1, \\ \\ -\xi^2 x - \xi x^2 + \xi x - \frac{1}{120}\xi^2 x^2 + \frac{21}{20}\xi^2 x^2 \\ +\frac{1}{24}\xi^2 x^4 - \frac{1}{12}\xi^2 x^3 \\ +\frac{1}{24}\xi^4 x^2 - \frac{1}{24}\xi^4 x - \frac{1}{120}\xi^5 x^2 + \frac{\xi^5}{120}, \\ 0 \le \xi < x \le 1. \end{cases} \tag{5}$$

Proof. First, let us suppose

$$\mathcal{R}_\xi(x) = \begin{cases} \sum_{i=1}^6 c_i(\xi)x^{i-1}, & 0 \le x \le \xi \le 1, \\ \\ \sum_{i=1}^6 d_i(\xi)x^{i-1}, & 0 \le \xi < x \le 1. \end{cases} \tag{6}$$

Then from $y \in {}^o\mathcal{W}_2^3[0,1]$, we get

$$\begin{aligned}&\langle y(x), \mathcal{R}_\xi(x)\rangle_{{}^o\mathcal{W}_2^3[0,1]} = y(0)\mathcal{R}_\xi(0) + y'(0)\mathcal{R}'_\xi(0) + y(1)\mathcal{R}_\xi(1) \\ &+\int_0^1 y^{(3)}(x)\frac{\partial^3\mathcal{R}_\xi(x)}{\partial x^3}\mathrm{d}x \\ &= y(0)\mathcal{R}_\xi(0) + y'(0)\frac{\partial\mathcal{R}_\xi(0)}{\partial x} \\ &+y(1)\mathcal{R}_\xi(1) + y''(1)\frac{\partial^3\mathcal{R}_\xi(1)}{\partial x^3} \\ &-y''(0)\frac{\partial^3\mathcal{R}_\xi(0)}{\partial x^3} - y'(1)\frac{\partial^4\mathcal{R}_\xi(1)}{\partial x^4} \\ &+y'(0)\frac{\partial^4\mathcal{R}_\xi(0)}{\partial x^4} + y(1)\frac{\partial^5\mathcal{R}_\xi(1)}{\partial x^5} - y(0)\frac{\partial^5\mathcal{R}_\xi(0)}{\partial x^5} \\ &-\int_0^1 y(x)\frac{\partial^6\mathcal{R}_\xi(x)}{\partial x^6}\mathrm{d}x.\end{aligned}$$

Solving the coefficients, we get the reproducing kernel function as:

$$\mathcal{R}_\xi(x)=\begin{cases}-x^2\xi-x\xi^2+x\xi-\frac{1}{120}x^2\xi^2+\frac{21}{20}x^2\xi^2\\+\frac{1}{24}x^2\xi^4-\frac{1}{12}x^2\xi^3\\+\frac{1}{24}x^4\xi^2-\frac{1}{24}x^4\xi-\frac{1}{120}x^5\xi^2+\frac{x^5}{120},\\0\le x\le\xi\le 1,\\\\-\xi^2x-\xi x^2+\xi x-\frac{1}{120}\xi^2x^2\\+\frac{21}{20}\xi^2x^2+\frac{1}{24}\xi^2x^4-\frac{1}{12}\xi^2x^3\\\\+\frac{1}{24}\xi^4x^2-\frac{1}{24}\xi^4x-\frac{1}{120}\xi^5x^2+\frac{\xi^5}{120},\\0\le\xi<x\le 1.\end{cases}\tag{7}$$

2.1. Solutions in ${}^o\mathcal{W}_2^3[0,1]$

We consider the solution of Eq.(1) in the reproducing kernel space ${}^o\mathcal{W}_2^3[0,1]$ in this section. On defining the operator

$$\mathcal{L}:\ {}^o\mathcal{W}_2^3[0,1]\to\mathcal{W}_2^1[0,1],$$

problem (1) converts as:

$$\begin{cases}\mathcal{L}y=\mathfrak{f}(x,u),\quad x\in[0,1],\\y(0)=A,y(1)=B,\end{cases}\tag{8}$$

We should homogenize the conditions. Put

$$u(x)=y(x)+(A-B)x-A,$$

then we can obtain homogeneous boundary-value conditions of problem (1)

$$\mathcal{L}u(x)=p^{'}(x)u^{'}+p(x)u^{''}.\tag{9}$$

With the boundary conditions:

$$\begin{cases}\mathcal{L}u=\mathfrak{g}(x,u),\quad x\in[0,1],\\u(0)=u(1)=0,\end{cases}\tag{10}$$

where

$$\mathfrak{g}(x,u)=f(x,u)+p^{'}(x)(A-B).\tag{11}$$

Theorem 2. *The operator $\mathcal{L}$ is a bounded linear operator.*

Proof. Firstly, we present $\|\mathcal{L}u\|^2_{\mathcal{W}_2^1}\le\mathcal{M}\|u\|^2_{{}^o\mathcal{W}_2^3}$, with $\mathcal{M}>0$. By
(3) and (3), we get

$$\begin{aligned}\|\mathcal{L}u\|^2_{\mathcal{W}_2^1}&=\langle\mathcal{L}u,\mathcal{L}u\rangle_{\mathcal{W}_2^1}\\&=\int_0^1\left[((\mathcal{L}u)(x))^2+((\mathcal{L}u)'(x))^2\right]\mathrm{d}x.\end{aligned}$$

Moreover, by reproducing property we have:

$$y(x)=\langle y(\cdot),\mathcal{R}_x(\cdot)\rangle_{{}^o\mathcal{W}_2^3}.$$

Then, we get

$$\begin{aligned}\mathcal{L}u(x)&=\langle u(\cdot),\mathcal{L}\mathcal{R}_x(\cdot)\rangle_{{}^o\mathcal{W}_2^3}\\&=\langle u(\cdot),(\mathcal{L}_1+\mathcal{L}_2)\mathcal{R}_x(\cdot)\rangle_{{}^o\mathcal{W}_2^3}\\&=\langle u(\cdot),\mathcal{L}_1\mathcal{R}_x(\cdot)\rangle_{{}^o\mathcal{W}_2^3}+\langle u(\cdot),\mathcal{L}_2\mathcal{R}_x(\cdot)\rangle_{{}^o\mathcal{W}_2^3}.\end{aligned}$$

With condition to $p(x)\in C^2[0,1]$, $\mathcal{M}_p=\max\{|p(x)|,\left|p^{'}(x)\right|,\left|p^{''}(x)\right|\quad 0\le x\le 1\}$, $\mathcal{M}_1=\max\{\frac{\partial}{\partial x}\mathcal{R}_x(\xi)\quad 0\le\xi\le 1\}$, and $\mathcal{M}_2=\max\{\frac{\partial}{\partial x^2}\mathcal{R}_x(\xi)\quad 0\le\xi\le 1\}$ then

$$\begin{aligned}|\mathcal{L}u(x)|&\le\|u\|_{{}^o\mathcal{W}_2^3}\|\mathcal{L}_1\mathcal{R}_x\|_{{}^o\mathcal{W}_2^3}+\|u\|_{{}^o\mathcal{W}_2^3}\|\mathcal{L}_2\mathcal{R}_x\|_{{}^o\mathcal{W}_2^3}\\&=\mathcal{M}_1\mathcal{M}_p\|u\|_{{}^o\mathcal{W}_2^3}+\mathcal{M}_2\mathcal{M}_p\|u\|_{{}^o\mathcal{W}_2^3}\\&=(\mathcal{M}_1+\mathcal{M}_2)\mathcal{M}_p\|u\|_{{}^o\mathcal{W}_2^3}\end{aligned}$$

where $\mathcal{M}_1>0,\mathcal{M}_2>0,\mathcal{M}_p>0$. Therefore

$$\int_0^1\left[(\mathcal{L}y)(x)\right]^2\mathrm{d}x\le(\mathcal{M}_1+\mathcal{M}_2)^2\mathcal{M}_p^2\|u\|^2_{{}^o\mathcal{W}_2^3}.$$

Also, from

$$\begin{aligned}&(\mathcal{L}u)'(x)=\left\langle u(\cdot),(\mathcal{L}\mathcal{R}_x)'(\cdot)\right\rangle_{{}^o\mathcal{W}_2^3}\\&=\left\langle u(\cdot),(\mathcal{L}_1+\mathcal{L}_2)\mathcal{R}_x)'(\cdot)\right\rangle_{{}^o\mathcal{W}_2^3}\\&=\left\langle u(\cdot),(\mathcal{L}_1\mathcal{R}_x)'(\cdot)\right\rangle_{{}^o\mathcal{W}_2^3}+\left\langle u(\cdot),(\mathcal{L}_2\mathcal{R}_x)'(\cdot)\right\rangle_{{}^o\mathcal{W}_2^3}\end{aligned}$$

we have condition to $p(x)\in C^2[0,1]$, $\mathcal{M}_p=\max\{|p(x)|,\left|p^{'}(x)\right|,\left|p^{''}(x)\right|0\le x\le 1\}$ $\mathcal{M}_1=\max\{\frac{\partial}{\partial x}\mathcal{R}_x(\xi)\quad 0\le\xi\le 1\},\mathcal{M}_2=\max\{\frac{\partial}{\partial x^2}\mathcal{R}_x(\xi)\quad 0\le\xi\le 1\}$ and $\mathcal{M}_3=\max\{\frac{\partial}{\partial x^2}\mathcal{R}_x(\xi)\quad 0\le\xi\le 1\}$ then

$$\begin{aligned}\left|(\mathcal{L}u)'(x)\right|&\le\|u\|_{{}^o\mathcal{W}_2^3}\left\|(\mathcal{L}_1\mathcal{R}_x)'\right\|_{{}^o\mathcal{W}_2^3}\\&\quad+\|u\|_{{}^o\mathcal{W}_2^3}\left\|(\mathcal{L}_2\mathcal{R}_x)'\right\|_{{}^o\mathcal{W}_2^3}\\&=(\mathcal{M}_p\mathcal{M}_1+\mathcal{M}_p\mathcal{M}_2)\|u\|_{{}^o\mathcal{W}_2^3}\\&\quad+(\mathcal{M}_p\mathcal{M}_2+\mathcal{M}_p\mathcal{M}_3)\|u\|_{{}^o\mathcal{W}_2^3}\\&=\mathcal{M}_p(\mathcal{M}_1+\mathcal{M}_2+\mathcal{M}_3)\|u\|_{{}^o\mathcal{W}_2^3}\end{aligned}$$

where $\mathcal{M}_1>0,\mathcal{M}_2>0,\mathcal{M}_3>0,\mathcal{M}_p>0$. Thus, we get

$$\left[(\mathcal{L}y)'(x)\right]^2\le\mathcal{M}_p^2(\mathcal{M}_1+\mathcal{M}_2+\mathcal{M}_3)^2\|u\|^2_{{}^o\mathcal{W}_2^3}$$

and

$$\int_0^1\left[(\mathcal{L}u)'(x)\right]^2\mathrm{d}x\le\mathcal{M}_p^2(\mathcal{M}_1+\mathcal{M}_2+\mathcal{M}_3)^2\|u\|^2_{{}^o\mathcal{W}_2^3},$$

that is

$$\begin{aligned}
&\|\mathcal{L}y\|^2_{\mathcal{W}_2^1} = \int_0^1 \left[\left[(\mathcal{L}u)(x)\right]^2 + \left[(\mathcal{L}u)'(x)\right]^2 \right] \mathrm{d}x \\
&\leq (\mathcal{M}_1+\mathcal{M}_2)^2 \mathcal{M}_p^2 \|u\|^2_{{}^o\mathcal{W}_2^3} \\
&+ \mathcal{M}_p^2(\mathcal{M}_1+\mathcal{M}_2+\mathcal{M}_3)^2 \|u\|^2_{{}^o\mathcal{W}_2^3} \\
&= \mathcal{M}_p^2((\mathcal{M}_1+\mathcal{M}_2+\mathcal{M}_3)^2 + (\mathcal{M}_1+\mathcal{M}_2)^2) \|u\|^2_{{}^o\mathcal{W}_2^3} \\
&= \mathcal{M} \|u\|^2_{{}^o\mathcal{W}_2^3}
\end{aligned}$$

where $\mathcal{M} = \mathcal{M}_p^2((\mathcal{M}_1+\mathcal{M}_2+\mathcal{M}_3)^2 + (\mathcal{M}_1+\mathcal{M}_2)^2) > 0$. □

2.2. Solutions of the problems

Obviously, defined operator in (9) as $\mathcal{L} : {}^o\mathcal{W}_2^3[0,1] \to \mathcal{W}_2^1[0,1]$ is a bounded linear operator.

Let us define $\varphi_i(x) = T_{x_i}(x)$ and $\psi_i(x) = \mathcal{L}^*\varphi_i(x)$, where $\mathcal{L}^*$ is conjugate operator of $\mathcal{L}$. The orthonormal system $\{\hat{\psi}_i(x)\}_1^\infty \subseteq {}^o\mathcal{W}_2^3[0,1]$ can be attained by the Gram-Schmidt orthogonalization process of $\{\psi_i(x)\}_1^\infty$:

$$\hat{\psi}_i(x) = \sum_{k=1}^{i} \beta_{ik}\psi_k(x), \ (\beta_{ii} > 0, i = 1,2,...). \quad (12)$$

Theorem 3. *If $y(x)$ is the exact solution of (10), then*

$$y(x) = \sum_{i=1}^{\infty}\sum_{k=1}^{i} \beta_{ik}\mathfrak{f}(x_k,y_k)\hat{\psi}_i(x), \quad (13)$$

where $\{x_i\}_1^\infty$ is dense in $[0,1]$.

Proof. By the (12) and uniqueness of solution of (10) we obtain:

$$\begin{aligned}
y(x) &= \sum_{i=1}^{\infty} \left\langle y(x), \hat{\psi}_i(x) \right\rangle_{{}^o\mathcal{W}_2^3} \hat{\psi}_i(x) \\
&= \sum_{i=1}^{\infty}\sum_{k=1}^{i} \beta_{ik} \langle y(x), \psi_k(x)\rangle_{{}^o\mathcal{W}_2^3} \hat{\psi}_i(x) \\
&= \sum_{i=1}^{\infty}\sum_{k=1}^{i} \beta_{ik} \langle y(x), \mathcal{L}^*\varphi_k(x)\rangle_{{}^o\mathcal{W}_2^3} \hat{\psi}_i(x) \\
&= \sum_{i=1}^{\infty}\sum_{k=1}^{i} \beta_{ik} \langle \mathcal{L}y(x), \varphi_k(x)\rangle_{\mathcal{W}_2^1} \hat{\psi}_i(x) \\
&= \sum_{i=1}^{\infty}\sum_{k=1}^{i} \beta_{ik} \langle \mathfrak{f}(x,y), T_{x_k}\rangle_{\mathcal{W}_2^1} \hat{\psi}_i(x) \\
&= \sum_{i=1}^{\infty}\sum_{k=1}^{i} \beta_{ik}\mathfrak{f}(x_k,y_k)\hat{\psi}_i(x).
\end{aligned}$$

□

Finite terms of (13) concludes the approximate solution:

$$y_n(x) = \sum_{i=1}^{n}\sum_{k=1}^{i} \beta_{ik}\mathfrak{f}(x_k,y_k)\hat{\psi}_i(x). \quad (14)$$

Lemma 1. *If $\|y_n - y\|_{{}^o\mathcal{W}_2^3} \to 0$, $x_n \to x$,$(n \to \infty)$ and $\mathfrak{f}(x,y)$ is continuous w.r.t. $x \in [0,1]$, then*

$$\mathfrak{f}(x_n, y_{n-1}(x_n)) \to \mathfrak{f}(x,u(x)), \quad as\ n \to \infty.$$

Theorem 4. *Let for any fixed $y_0(x) \in {}^o\mathcal{W}_2^3[0,1]$ we have*

(i)
$$y_n(x) = \sum_{i=1}^{n} A_i\hat{\psi}_i(x), \quad (15)$$
where
$$A_i = \sum_{k=1}^{i} \beta_{ik}\mathfrak{f}(x_k, y_{k-1}(x_k)), \quad (16)$$

(ii) $\|y_n\|_{{}^o\mathcal{W}_2^3}$ *is bounded;*

(iii) $\{x_i\}_1^\infty$ *is dense in* $[0,1]$*;*

(iv) $\mathfrak{f}(x,y) \in \mathcal{W}_2^1[0,1]$ *for any* $y(x) \in {}^o\mathcal{W}_2^3[0,1]$. *Then the approximate solution $y_n(x)$ converges to the exact solution of (13) in ${}^o\mathcal{W}_2^3$ and we have*

$$y(x) = \sum_{i=1}^{\infty} A_i\hat{\psi}_i(x).$$

Proof. First, we prove the convergence of $y_n(x)$. From (15), we have

$$y_{n+1}(x) = y_n(x) + A_{n+1}\hat{\psi}_{n+1}(x). \quad (17)$$

Also, orthonormality of $\{\hat{\psi}_i\}_{i=1}^\infty$, yields

$$\begin{aligned}
&\|y_{n+1}\|^2_{{}^o\mathcal{W}_2^3} = \|y_n\|^2_{{}^o\mathcal{W}_2^3} + A^2_{n+1} \\
&= \|y_{n-1}\|^2_{{}^o\mathcal{W}_2^3} + A_n^2 + A^2_{n+1} = ... = \sum_{i=1}^{n+1} A_i^2,
\end{aligned} \quad (18)$$

and from boundedness of $\|y_n\|_{{}^o\mathcal{W}_2^3}$, we obtain

$$\sum_{i=1}^{\infty} A_i^2 < \infty,$$

i.e.,

$$\{A_i\} \in l^2 \quad (i = 1,2,...).$$

Let $m > n$, in view of $(y_m - y_{m-1}) \perp (y_{m-1} - y_{m-2}) \perp ... \perp (y_{n+1} - y_n)$, we get

$$\|y_m - y_n\|^2_{{}^o\mathcal{W}_2^3} = \|y_m - y_{m-1} + y_{m-1} - y_{m-2} + ... + y_{n+1} - y_n\|^2_{{}^o\mathcal{W}_2^3}$$
$$= \|y_m - y_{m-1}\|^2_{{}^o\mathcal{W}_2^3} + ... + \|y_{n+1} - y_n\|^2_{{}^o\mathcal{W}_2^3}$$
$$= \sum_{i=n+1}^{m} A_i^2 \to 0, \quad m, n \to \infty.$$

Considering the completeness of ${}^o\mathcal{W}_2^3[0,1]$, there exists $y(x) \in {}^o\mathcal{W}_2^3[0,1]$, such that

$$y_n(x) \to y(x) \quad as\ n \to \infty.$$

(ii) Now, we show $y(x)$ is the exact solution of (10). Tacking limits in (15) we get

$$y(x) = \sum_{i=1}^{\infty} A_i \hat{\psi}_i(x).$$

Thus, we reach

$$(\mathcal{L}y)(x_2) = \mathfrak{f}(x_2, y_1(x_2)).$$

Moreover, by induction we conclude

$$(\mathcal{L}y)(x_j) = \mathfrak{f}(x_j, y_{j-1}(x_j)). \tag{19}$$

From $\overline{\{x_i\}_{i=1}^{\infty}} = [0,1]$, it can be presented that for any $\xi \in [0,1]$, there exists $\{x_{n_j}\}_1^{\infty} \subseteq \{x_i\}_1^{\infty}$, such that $\lim_{j\to\infty} x_{n_j} = \xi$. Therefore, convergence of $y_n(x)$ and Lemma 4.3 yields

$$(\mathcal{L}y)(\xi) = \mathfrak{f}(\xi, y(\xi)).$$

So, $y(x)$ is the exact solution of (10) given by

$$y(x) = \sum_{i=1}^{\infty} A_i \hat{\psi}_i(x),$$

where A_i are given by (16). □

Theorem 5. *If $y \in {}^o\mathcal{W}_2^3[0,1]$ then*

$$\|y_n - y\|_{{}^o\mathcal{W}_2^3} \to 0, \quad n \to \infty.$$

Moreover a sequence $\|y_n - y\|_{{}^o\mathcal{W}_2^3}$ is monotonically decreasing in n.

Proof.
By (13) and (14), we acquire

$$\|y_n - y\|_{{}^o\mathcal{W}_2^3} = \left\| \sum_{i=n+1}^{\infty} \sum_{k=1}^{i} \beta_{ik} \mathfrak{f}(x_k, y_k) \hat{\psi}_i \right\|_{{}^o\mathcal{W}_2^3}.$$

Therefore, we obtain

$$\|y_n - y\|_{{}^o\mathcal{W}_2^3} \to 0, \quad n \to \infty.$$

In addition

$$\|y_n - y\|^2_{\mathcal{W}_2^3} = \left\| \sum_{i=n+1}^{\infty} \sum_{k=1}^{i} \beta_{ik} \mathfrak{f}(x_k, y_k) \hat{\psi}_i \right\|^2_{{}^o\mathcal{W}_2^3}$$
$$= \sum_{i=n+1}^{\infty} \left(\sum_{k=1}^{i} \beta_{ik} \mathfrak{f}(x_k, y_k) \hat{\psi}_i \right)^2.$$

□

3. Numerical examples

We consider the following problems by reproducing kernel Hilbert space method in this section. We computed our results by MAPLE. We showed our results by tables.

Example 1. We investigate:

$$\left(\frac{1}{1+t} u(t)'\right)' = 2\exp(3u(t)) \quad 0 \le t \le 1 \tag{20}$$

$$u(0) = 0 \quad u(1) = -\log_e(2)$$

We have the exact solution of the above problem as:

$$u(t) = \log_e(1/(1+t)).$$

We searched the boundary value problem (20) by the proposed method and gave corresponding error-norms by Table 1.

Example 2. We solved the following problem for the second example in the reproducing kernel Hilbert space.

$$((1+t^2)u(t)')' - (1+t-t^2)u(t) = h(t) \tag{21}$$

$$u(0) = 0 \quad u(1) = 0$$

We get the exact solution of the above problem as:

$$u(t) = 1 + (t-1)\exp(-t) - t\exp(-(1-t)).$$

In Table 2, we computed absolute errors for (21).

Table 1. Maximum absolute errors (MAE) of the first example.

	$N = 64,\ \sigma = 1.02$	$N = 64,\ \sigma = 1.02$
RKHSM	$5.46E-12$	$5.46E-12$
[10]	$1.70E-08$	$2.85E-06$
[11]	$8.49E-04$	$1.21E-02$
[12]	$2.43E-03$	$1.88E-02$
[13]	$5.63E-03$	$2.70E-02$

Table 2. Maximum absolute errors (MAE) of the second example.

	$N = 64,\ \sigma = 1.02$	$N = 64,\ \sigma = 1.02$
RKHSM	$9.20E-11$	$9.20E-11$
[10]	$8.07E-06$	$6.09E-05$
[11]	$1.86E-03$	$1.08E-02$
[12]	$2.16E-03$	$9.87E-03$
[13]	$5.19E-04$	$2.64E-03$

4. Conclusions

In this work, we gave a new application of the reproducing kernel Hilbert space method. We obtained very useful reproducing kernel functions in the reproducing kernel Hilbert spaces. We proved the accuracy of the method. We compared the reproducing kernel Hilbert space method with the techniques existed in the literature. We concluded that the proposed technique is very effective for solving nonlinear two-point boundary value problems.

References

[1] Safari, M.J.S., Rahimzadeh Arashloo, S. & Danandeh, M. (2020). A. Rainfall-runoff modeling through regression in the reproducing kernel Hilbert space algorithm. *Journal of Hydrology* 587, 125014.

[2] Najafi, N. & Allahviranloo, T. (2020). Combining fractional differential transform method and reproducing kernel Hilbert space method to solve fuzzy impulsive fractional differential equations. *Computational and Applied Mathematics*. 39(2), 122.

[3] Sahihi, H., Allahviranloo, T. & Abbasbandy, S. (2020). Solving system of second-order BVPs using a new algorithm based on reproducing kernel Hilbert space. *Applied Numerical Mathematics*, 151, pp. 27-39.

[4] Agud, L., Calabuig, J.M. & Sánchez Pérez, E.A. (2020). Weighted p -regular kernels for reproducing kernel Hilbert spaces and Mercer Theorem. *Analysis and Applications*, 18(3), pp. 359-383.

[5] Mundayadan, A. & Sarkar, J. (2020). Linear dynamics in reproducing kernel Hilbert spaces. *Bulletin des Sciences Mathematiques*, 159, 2020 102826.

[6] Abu Arqub, O. & Maayah, B. (2019). Modulation of reproducing kernel Hilbert space method for numerical solutions of Riccati and Bernoulli equations in the Atangana-Baleanu fractional sense. *Chaos, Solitons and Fractals*, 125, 163-170.

[7] Emamjome, M., Azarnavid, B. & Ghehsareh, H.R. (2019). A reproducing kernel Hilbert space pseudospectral method for numerical investigation of a two-dimensional capillary formation model in tumor angiogenesis problem. *Neural Computing and Applications*, 31(7), 2233-2241.

[8] Foroutan, M., Asadi, R. & Ebadian, A. (2019). A reproducing kernel Hilbert space method for solving the nonlinear three-point boundary value problems. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 32(3), e2573.

[9] Karatas Akgül, E., Akgül, A., Khan, Y. & Baleanu, D. (2019). Representation for the reproducing kernel hilbert space method for a nonlinear system. *Hacettepe Journal of Mathematics and Statistics*, 48(5), 1345-1355.

[10] Jain, M.K., Sharma, S. & Mohanty, R.K. (2016). High accuracy variable mesh method for nonlinear two-point boundary value problems in divergence form. *Applied Mathematics and Computation*, 273, 885–896.

[11] Mohanty, R.K. (2005). A family of variable mesh methods for the estimates of (du/dr) and solution of non-linear two point boundary value problems with singularity. *Journal of Computational and Applied Mathematics*, 182, 173–187.

[12] Mohanty, R.K., Evans, D.J. & Khosla, N. (2005). An $O(h_k^3)$ non-uniform mesh cubic spline TAGE method for non-linear singular two-point boundary value problems. *International Journal of Computer Mathematics*, 82, 1125–1139.

[13] Mohanty, R.K. & Khosla, N. (2006). Application of TAGE iterative algorithms to an efficient third order arithmetic average variable mesh discretization for two-point non-linear boundary value problems. *Applied Mathematics and Computation*, 172 148–162.

[14] Allahviranloo, T. & Sahihi, H. (2020). Reproducing kernel method to solve parabolic partial differential equations with nonlocal conditions. *Numerical Methods for Partial Differential Equations*, 36(6), 1758–1772.

[15] Abu Arqub, O. & Al-Smadi, M. (2018). Atangana–Baleanu fractional approach to the solutions of Bagley–Torvik and Painlevé equations in Hilbert space. *Chaos, Solitons and Fractals*, 117, 161-167.

[16] Abu Arqub, O. & Maayah, B. (2018). Numerical solutions of integrodifferential equations of Fredholm operator type in the sense of the Atangana–Baleanu fractional operator. *Chaos, Solitons and Fractals*, 117, 117-124.

[17] Abu Arqub, O. & Maayah, B. (2019). Fitted fractional reproducing kernel algorithm for the numerical solutions of ABC – Fractional Volterra integro-differential equations. *Chaos, Solitons and Fractals*, 126, 394-402.

[18] Abu Arqub, O. & Maayah, B. (2019). Modulation of reproducing kernel Hilbert space method for numerical solutions of Riccati and Bernoulli equations in the Atangana-Baleanu fractional sense. *Chaos, Solitons and Fractals*, 125, 163-170.

[19] Abu Arqub, O. &, Al-Smadi, M. (2020). Fuzzy conformable fractional differential equations: novel extended approach and new numerical solutions. *Soft Computing*, 24, 12501–12522.

[20] Yavuz, M. & Evirgen, F. (2018). An alternative approach for nonlinear optimization problem with Caputo-Fabrizio derivative. *In ITM Web of Conferences*, Vol.22, p.01009.

[21] Yavuz, M. & Ozdemir, N. (2018). On the solutions of fractional Cauchy problem featuring conformable derivative. *In ITM Web of Conferences*, Vol.22, p.01045.

[22] Evirgen, F. (2016). Analyze the optimal solutions of optimization problems by means of fractional gradient based system using VIM. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, 6(2), 75-83.

[23] Yavuz, M. & Sene, N. (2020). Approximate solutions of the model describing fluid flow using generalized -Laplace transform method and heat balance integral method. *Axioms*, 9(4), 123.

Ali Akgül *is an associate professor at Siirt University, Turkey. He has many publications on fractional calculus and numerical methods.*
https://orcid.org/0000-0001-9832-1424

Mir Sajjad Hashemi *is an associate professor at Bonab University, Iran. He has many publications on fractional calculus and numerical methods.*
https://orcid.org/0000-0002-5529-3125

Negar Seyfi *is a researcher from Bonab University, Iran.*
https://orcid.org/0000-0002-1760-0745

An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.11, No.2, pp.206-215 (2021)
http://doi.org/10.11121/ijocta.01.2021.001056

RESEARCH ARTICLE

The optimality principle for second-order discrete and discrete-approximate inclusions

Sevilay Demir Sağlam*

Department of Mathematics, University of Istanbul,Turkey
sevilay.demir@istanbul.edu.tr

ARTICLE INFO

Article History:
Received 14 December 2020
Accepted 21 February 2021
*Available ** May 2021*

Keywords:
Discrete inclusion
Approximation
Optimality condition
Locally adjoint mapping

AMS Classification 2010:
49K15; 49M25; 49J53 ; 37M15

ABSTRACT

This paper deals with the necessary and sufficient conditions of optimality for the Mayer problem of second-order discrete and discrete-approximate inclusions. The main problem is to establish the approximation of second-order viability problems for differential inclusions with endpoint constraints. Thus, as a supplementary problem, we study the discrete approximation problem and give the optimality conditions incorporating the Euler-Lagrange inclusions and distinctive transversality conditions. Locally adjoint mappings (LAM) and equivalence theorems are the fundamental principles of achieving these optimal conditions, one of the most characteristic properties of such approaches with second-order differential inclusions that are specific to the existence of LAMs equivalence relations. Also, a discrete linear model and an example of second-order discrete inclusions in which a set-valued mapping is described by a nonlinear inequality show the applications of these results.

1. Introduction

Optimal control problems with discrete and differential inclusions are increasingly studied in mathematical theory [1, 2]. Also, discrete and continuous-time processes have wide applications in the field of mathematical economics and the problems of control dynamic system optimization [3–6]. Thus, the optimal control problems described by discrete inclusions with endpoint constraints and approximation play a very significant role in both theory and applications of control theory [7, 8].

Second-order discrete and differential inclusions have been studied by many authors when the set-valued mapping is both convex and nonconvex valued (see [9–11] and references therein).

Auslender and Mechler [12] establish necessary and sufficient conditions to ensure the existence of solutions to the second-order differential inclusions with state constraints via interior tangent sets.

An approximation to a linear differential inclusion using N-stage single step discrete inclusions is described in the paper [13]. The result is applied to the discretization of control constrained optimal control problems in the second-order and the use of dynamic programming for approximate feedback design.

The paper [14] deals with the discrete approximations of nonconvex differential inclusions in Hilbert spaces and dynamic optimization-optimal control problems concerning such differential inclusions.

Agarwal and O'Regan [15] present new fixed-point theorems for weakly sequentially upper semicontinuous maps. These results are used to establish existing principles for second-order differential equations and inclusions.

The optimal control of discrete-time systems are given in the book of Boltyanskii [16]. That book explores some results from the classical control

*Corresponding author

point of view for a linear, time-invariant, discrete-time, optimal control system with infinite-time case.

In general, several studies on the differential inclusions of the second-order are devoted to problems of existence and viability. In the case where the multifunction is upper semicontinuous and has compact convex values, Haddad and Yarou [17] provided the first viability result for second-order differential inclusions. The Cauchy problem for the infinite-dimensional case and second-order differential inclusions are considered in that paper.

As is pointed out in the paper [18], Marco and Murillo analyzed the existence of Lyapunov functions for second-order differential inclusions by using the methods of the viability theory. A necessary assumption on the initial states and sufficient conditions are obtained for the existence of local and global Lyapunov functions.

Lupulescu [19] demonstrated the existence of viable solutions for autonomous second-order functional-differential inclusions in the case where the multifunction defining the inclusion is upper semicontinuous, compact valued, and contained in the subdifferential of a proper lower semicontinuous convex function.

Due to the higher-order derivatives and their discrete analogs, the problems accompanied by the higher-order discrete and differential inclusions are more complicated. A convenient procedure for eliminating this complication in optimal control theory involving higher-order derivatives is a formal reduction of the problems by substitution to the system of first-order differential inclusions or equations. But in practice, returning to the higher-order problem and expressing the arising optimality conditions in terms of the original problem data, in general, is very difficult. Although the construction of adjoint inclusions and transversality conditions is more complicated, Mahmudov formulates the conditions of optimality for the optimal control problem of higher-order differential inclusions with functional constraints in the paper [20].

The principle approach we use is that of locally adjoint mapping(LAM), which facilitates obtaining necessary and sufficient conditions for all types of discrete and differential inclusions. Optimization of various forms of discrete inclusions can be reduced to finite-dimensional problems of mathematical programming, namely, to the minimization of functions on the intersection of a finite number of sets.

We use difference approximations of ordinary derivatives and grid functions on a uniform grid to approximate differential inclusions and to derive necessary and sufficient conditions of optimality for discrete-approximation problems. It turns out that this requires some particular equivalence theorems of a LAM, which arise in discrete and discrete-approximation problems.

One of the central objects of this paper is the relationship between continuous and discrete systems. By using particular equivalence theorems of the LAM, which play a significant role in the following investigations and without which few necessary or sufficient conditions would be obtained, the transition to the optimal conditions for discrete-approximation problems from their discrete counterparts is realized. The point argument is that discrete and discrete-approximation problems naturally are described by different set-valued mappings (say F and G, respectively). Therefore, we have to express the LAM G^* by F^* to formulate the optimality conditions for each discrete and discrete-approximation problem associated with the continuous problem.

The rest of this paper is organized as follows.

For the convenience of the reader, the necessary facts and supplementary results from the book of Mahmudov [1] are summarized in Section 2. In particular, the Hamiltonian function, argmaximum set of a set-valued mapping, and the locally-adjoint mapping are introduced and the viability problems for second-order discrete and differential inclusions are described with endpoint constraints.

In Section 3, the discrete problem with the second-order discrete inclusions posed in Section 2 is reduced to a convex problem of a finite number of geometric constraints. We prove the necessary and sufficient conditions of optimality in terms of $LAMs$ by using constructions of convex and nonsmooth analysis.

In Section 4, we use difference approximations of derivatives and grid functions on a uniform grid to approximate problems with second-order differential inclusions. Then we derive the necessary and sufficient condition for optimality in both forms of Euler-Lagrange inclusions and transversality conditions for the discrete-approximation problem.

In Section 5, we present some applications of the results obtained for problems with second-order discrete inclusions and set-valued mappings.

2. Needed facts and problem statement

For the convenience of the reader, the books Mahmudov [1] and Mordukhovich [2] contain all the necessary notions and results from set-valued

analysis theory. Let $\mathbb{R}^n$ be an n-dimensional Euclidean space, $\langle x, v\rangle$ be an inner product of elements $x, v \in \mathbb{R}^n, (x, v)$ be a pair of x, v. Assume that $F : \mathbb{R}^n \times \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is a set-valued mapping from $\mathbb{R}^{2n} = \mathbb{R}^n \times \mathbb{R}^n$ into the set of subsets of $\mathbb{R}^n$. The set-valued mapping F is said to be convex if its graph $gphF = \{(x, v_1, v_2) : v_2 \in F(x, v_1)\}$ is a convex subset of $\mathbb{R}^{3n}$. A set-valued mapping F is convex-closed if its $gphF$ is a convex-closed set in $\mathbb{R}^{3n}$. It is convex-valued if $F(x, v_1)$ is a convex set for each $(x, v_1) \in domF$, where $domF = \{(x, v_1) : F(x, v_1) \neq \emptyset\}$.

Let us introduce the Hamiltonian function and argmaximum set for a set-valued mapping F as

$$H_F(x, v_1, v_2^*) = \sup_{v_2}\Big\{\langle v_2, v_2^*\rangle : v_2 \in F(x, v_1)\Big\},$$

$$F(x, v_1; v_2^*) = \Big\{v_2 \in F(x, v_1) : \langle v_2, v_2^*\rangle = H_F(x, v_1, v_2^*)\Big\}$$

$v_2^* \in \mathbb{R}^n$, respectively. For convex F, we set $H_F(x, v_1, v_2^*) = -\infty$ if $F(x, v_1) = \emptyset$. Clearly, the Hamiltonian function $H_F(\cdot, \cdot, v_2^*)$ is concave for the convex set-valued mapping F.

Definition 1. *The convex cone $K_A(z_0)$, $z_0 = (x_0, u_0, v_0)$ is called the cone of tangent directions at a point $z_0 \in A$ to the set A if from $\overline{z} = (\overline{x}, \overline{u}, \overline{v}) \in K_A(z_0)$ it follows that $\overline{z}$ is a tangent vector to the set A at point $z_0 \in A$, i.e., there exists such function $\gamma(\lambda) \in \mathbb{R}^{3n}$ such that $z_0 + \lambda\overline{z} + \gamma(\lambda) \in A$ for sufficiently small $\lambda > 0$ and $\lambda^{-1}\gamma(\lambda) \to 0$ as $\lambda \downarrow 0$.*

It should be pointed out that the cone $K_A(z_0)$ is not uniquely defined. Since $\lambda\overline{z},\ \lambda \geq 0$ is a vector of tangent direction, if $\overline{z}$ is the same, then it is clear that such vectors form a cone. In any case we can see that the wider a cone of tangent directions we have the essentially necessary condition for a minimum [1].

Obviously, for a convex mapping F at a point $(x^0, v_1^0, v_2^0) \in gphF$ setting $\gamma(\lambda) \equiv 0$, we have

$$K_{gphF}(x^0, v_1^0, v_2^0) = cone\big[gphF - (x^0, v_1^0, v_2^0)\big]$$
$$= \Big\{(\overline{x}, \overline{v_1}, \overline{v_2}) :\ \overline{x} = \lambda(x - x^0), \overline{v_1} = \lambda(v_1 - v_1^0), \overline{v_2} = \lambda(v_2 - v_2^0)\Big\},\ \forall\ (x, v_1, v_2) \in gphF.$$

For a convex mapping F a set-valued function defined by

$$F^*\big(v_2^*; (x^0, v_1^0, v_2^0)\big) := \Big\{(x^*, v_1^*) : (x^*, v_1^*, -v_2^*) \in K^*_{gphF}(x^0, v_1^0, v_2^0)\Big\}$$

is a locally adjoint set-valued mapping (LAM) to F at a point $(x^0, v_1^0, v_2^0) \in gphF$, where $K^*_{gphF}(x^0, v_1^0, v_2^0)$ is the dual to the cone of tangent vectors $K_{gphF}(x^0, v_1^0, v_2^0)$.

Let $intA$ be the interior of the set $A \subset \mathbb{R}^{3n}$ and riA be the relative interior of the set A, i.e. the set of interior points of A with respect to its affine hull $AffA$. A function φ is called a proper function, if it does not assume the value $-\infty$ and is not identically equal to $+\infty$. Obviously φ is proper function if and only if $dom\varphi \neq \emptyset$ and $\varphi(x, y)$ is finite for $(x, y) \in dom\varphi$.

We note that a convex function is continuous on the relative interior of its domain, it may have discontinuities only on its relative boundary (see, for example, Theorem 1.18 [1]).

In this paper, we study the following main second-order discrete model:

$$\text{infimum } \varphi(x_{N-1}, x_N), \tag{1}$$

$$x_{t+2} \in F(x_t, x_{t+1}, t)\ ,\ t = 0, \dots, N-2, \tag{2}$$

$$x_0 = \alpha_0\ ,\ x_1 = \alpha_1,$$

$$x_t \in A,\ t = 0, \dots, N,\ x_N \in B, \tag{3}$$

where $x_t \in \mathbb{R}^n$, $F(\cdot, \cdot, t) : \mathbb{R}^{2n} \rightrightarrows \mathbb{R}^n$ is a time dependent set-valued mapping, $\varphi : \mathbb{R}^{2n} \to \mathbb{R}$ is continuous function, α_0, α_1 are fixed vectors and N is fixed natural number, $A, B \subset \mathbb{R}^n$ are nonempty subsets and $A \cap B \neq \emptyset$. We label this problem as (P_D).

A sequence $\{x_t\}_{t=0}^N = \{x_t : t = 0, 1, \dots, N\}$ is called a feasible trajectory for the stated problem (P_D). The problem is to find a solution $\{\widetilde{x}_t\}_{t=0}^N$ of the problem (P_D) for the second-order discrete inclusions satisfying $(2) - (3)$ and minimizing the Mayer functional $\varphi(x_{N-1}, x_N)$.

The problem (P_D) is said to be convex, if the $F(\cdot, \cdot, t), \varphi$ and A, B are convex set-valued mapping, proper convex function, and convex sets respectively.

Definition 2. *We say that for the convex problem (P_D) satisfies the regularity condition, if for points $x_t \in \mathbb{R}^n$, one of the following cases is fulfilled:*

(i) $(x_t, x_{t+1}, x_{t+2}) \in rigphF(\cdot, \cdot, t)$, $t = 0, \dots, N-2$, $x_t \in riA$, $t = 0, \dots, N$, $x_N \in riB$, $(x_{N-1}, x_N) \in ridom\varphi$,

(ii) $(x_t, x_{t+1}, x_{t+2}) \in intgphF(\cdot, \cdot, t)$, $t = 0, \dots, N-2$, $x_t \in intA$; $x_N \in intB$, *(with the possible exception of one fixed t) and φ is continuous at (x_{N-1}, x_N).*

It follows from the regularity condition that, if $\{\widetilde{x}_t\}_{t=0}^N$ is the optimal trajectory in the problem (P_D), then the cones of tangent directions $K_{gphF(\cdot,t)}(\tilde{x}_t, \tilde{x}_{t+1}, \tilde{x}_{t+2})$ are not separable and

consequently the condition of Theorem 3.2 [1] is satisfied.

We are interested in the approximate problem for the following evolution differential inclusions with endpoint constraint

$$\begin{aligned} &\text{infimum } \phi\big(x(1), x'(1)\big), \\ &x''(t) \in F(x(t), x'(t), t) \text{ , a.e. } t \in [0,1], \\ &x(0 = \beta_0 \text{ , } x'(0) = \beta_1, \\ &x(t) \in A, \ t \in [0,1], \ x(1) \in B, \end{aligned} \tag{4}$$

where $F(\cdot,\cdot,t) : \mathbb{R}^{2n} \rightrightarrows \mathbb{R}^n$ is a time dependent set-valued mapping, $\phi : \mathbb{R}^{2n} \to \mathbb{R}$ is continuous function and β_0, β_1 are fixed vectors and $A, B \subset \mathbb{R}^n$ are nonempty subsets and $A \cap B \neq \emptyset$. That problem is called the viability problem.

The problem is to find an arc $\widetilde{x}(\cdot)$ of the viability problem satisfying (4) almost everywhere on $[0,1]$ and the initial and endpoint constraints that minimizes the cost functional ϕ. Here a feasible trajectory $x(\cdot)$ is understood to be an absolutely continuous function on a time interval $[0,1]$ together with the first-order derivatives for which $x''(\cdot) \in L_1^n([0,1])$. Clearly, such a class of functions is a Banach space, endowed with the different equivalent norms.

Clearly, the condition (2) in the main discrete model is a discrete analog of second-order differential inclusions (4).

Definition 3. *We say $F(\cdot,\cdot,t)$ is viable if for every $x(0) = \beta_0$, $x'(0) = \beta_1$, (4) has an absolutely continuous solution $x(\cdot) : [0,1] \to \mathbb{R}^n$ such that $x(t) \in A$ for all $t \in [0,1]$ and the inclusion in (4) is satisfied for almost everywhere on* $[0,1]$.

Let us formulate the conditions of optimality for the discrete problem (P_D) before we begin the discussion on the approximation problem with second-order inclusions.

3. Necessary and sufficient conditions of optimality for discrete problem

We consider the discrete problem (P_D) in this section. First, let us introduce a vector $u = (x_0, x_1, \ldots, x_N) \in \mathbb{R}^{n(N+1)}$ and define the following convex sets in the space $\mathbb{R}^{n(N+1)}$

$$\begin{aligned} M_t &= \Big\{u = (x_0, \ldots, x_N) : \ (x_t, x_{t+1}, x_{t+2}) \\ &\in gphF(\cdot,\cdot,t)\Big\}, \ t = 0, 1, \ldots, N-2, \\ P_t &= \Big\{u = (x_0, \ldots, x_N) : \ x_t \in A\Big\}, \ t = 0, \ldots, N, \\ Q &= \Big\{u = (x_0, \ldots, x_N) : \ x_N \in B\Big\}, \\ Q_0 &= \Big\{u = (x_0, \ldots, x_N) : \ x_0 = \alpha_0\Big\}, \\ Q_1 &= \Big\{u = (x_0, \ldots, x_N) : \ x_1 = \alpha_1\Big\}. \end{aligned}$$

The discrete problem (P_D) is now reduced to solving a convex programming problem by setting $f(u) = \varphi(x_{N-1}, x_N)$. In fact, it is easy to see that the problem (P_D) is equivalent to the following one:

$$\text{minimize } f(u) \tag{5}$$

$u \in C := \Big(\bigcap\limits_{t=0}^{N-2} M_t\Big) \cap \Big(\bigcap\limits_{t=0}^{N} P_t\Big) \cap Q \cap Q_0 \cap Q_1$, where C is the convex set. This transformation allows us to prove rigorously that if $\{\widetilde{x}_t\}_{t=0}^N$ is the optimal solution to the problem (P_D), then $\widetilde{u}$ is the solution to the problem (5). It should be noted that under the regularity condition Definition 2 the dual cone associated with intersection of cones of tangent directions is equal to the algebraic sum of their dual cones. Thus, from Theorem 1.11 [1], we have

$$\begin{aligned} K_C^*(\widetilde{u}) \ &= \sum_{t=0}^{N-2} K_{M_t}^*(\widetilde{u}) + \sum_{t=0}^{N} K_{P_t}^*(\widetilde{u}) + K_Q^*(\widetilde{u}) \\ &\quad + K_{Q_0}^*(\widetilde{u}) + K_{Q_1}^*(\widetilde{u}). \end{aligned} \tag{6}$$

Moreover, the results taken from [1] provide necessary conditions of optimality for the convex mathematical programming (5). We can prove necessary conditions of optimality for problem (5) with geometric constraints based on results concerning convex mathematical programming. Thus, by continuity of φ at points of some feasible solution $\{\widetilde{x}_t\}_{t=0}^N$ it follows from Theorem 3.4 [1], there exist vectors $u^*(t) \in K_{M_t}^*(\tilde{u}), t = 0, 1, \ldots, N-2, w^*(t) \in K_{P_t}^*(\tilde{u})$, $t = 0, \ldots, N$ and $v^*(0) \in K_{Q_0}^*(\tilde{u})$, $v^*(1) \in K_{Q_1}^*(\tilde{u})$, $v^*(N) \in K_Q^*(\tilde{u})$ and the number $\mu \in \{0, 1\}$, not all equal to zero, such that

$$\begin{aligned} \mu\hat{u}^* = \sum_{t=0}^{N-2} u^*(t) + \sum_{t=0}^{N} w^*(t) + v^*(N) \\ + v^*(0) + v^*(1), \end{aligned} \tag{7}$$

where $\hat{u}^* \in \partial_u f(\tilde{u})$. From definition of the function f it is easy to see that the vector $\hat{u}^* \in \partial_u f(\tilde{u})$ has a form $\hat{u}^* = \Big(\underbrace{0, \ldots, 0}_{N-1}, \hat{x}_{N-1}^*, \hat{x}_N^*\Big)$, $\Big(\hat{x}_{N-1}^*, \hat{x}_N^*\Big) \in \partial_{(x,v_1)}\varphi(\hat{x}_{N-1}, \hat{x}_N)$. First of all it is not hard to compute the dual cones $K_{P_t}^*(\widetilde{u})$, $K_Q^*(\widetilde{u})$, $K_{Q_0}^*(\widetilde{u})$ and $K_{Q_1}^*(\widetilde{u})$ as follows

$$\begin{aligned} K_{P_t}^*(\widetilde{u}) = \{w^*(t) : w_t^*(t) \in K_A^*(\widetilde{x}_t), \ w_k^*(t) = 0 \text{ , } \\ k \neq t\}, t = 0, \ldots, N \\ K_Q^*(\widetilde{u}) = \{v^*(N) : v_k^*(N) = 0, \ k \neq N\}, \\ K_{Q_0}^*(\widetilde{u}) = \{v^*(0) : v_k^*(0) = 0, \ k \neq 0\}, \\ K_{Q_1}^*(\widetilde{u}) = \{v^*(1) : v_k^*(1) = 0, \ k \neq 1\}. \end{aligned}$$

Then we should compute the dual cone $K^*_{M_t}(\widetilde{u})$ in the following lemma.

Lemma 1. *Let $K_{gphF}(x_t, x_{t+1}, x_{t+2})$, $(x_t, x_{t+1}, x_{t+2}) \in gphF$ be cone of tangent directions. Then*

$$K^*_{M_t}(u) = \Big\{u^* = (x_0^*, \dots, x_N^*) : (x_t^*, x_{t+1}^*, x_{t+2}^*) \in K^*_{gphF}(x_t, x_{t+1}, x_{t+2}), \quad x_k^* = 0,\ k \neq t, t+1, t+2\Big\}.$$

Proof. From the dual cone definition, $u^* \in K^*_{M_t}(u)$ is valid if and only if $\langle u^*, \overline{u}\rangle = \sum_{k=0}^{N} \langle x_k^*, \overline{x}_k\rangle \geq 0,\ \forall\ \overline{u} \in K_{M_t}(u)$. Clearly, $K_{M_t}(u) = \big\{\overline{u} : (\overline{x}_t, \overline{x}_{t+1}, \overline{x}_{t+2}) \in K_{gphF}(x_t, x_{t+1}, x_{t+2})\big\}$ and from the arbitrariness of components $\overline{x}_k, k \neq t, t+1, t+2$ of vectors $\overline{u}$ it follows that $x_k^* = 0,\ k \neq t, t+1, t+2$. Therefore the inequality $\langle x_t^*, \overline{x}_t\rangle + \langle x_{t+1}^*, \overline{x}_{t+1}\rangle + \langle x_{t+2}^*, \overline{x}_{t+2}\rangle \geq 0$ yields $(x_t^*, x_{t+1}^*, x_{t+2}^*) \in K^*_{gphF}(x_t, x_{t+1}, x_{t+2})$. □

Now, we are ready to give the conditions of optimality for the discrete problem (P_D).

Theorem 1. *Let $F(\cdot,\cdot,t)$ be a convex set-valued mapping and φ be proper convex functional and continuous at the points of some feasible trajectory. Then for optimality of the trajectory $\{\tilde{x}_t\}_{t=0}^N$ in the Mayer problem $(1)-(3)$ with second-order discrete inclusions, initial and endpoint constraints, it is necessary that there exist a number $\mu \in \{0,1\}$ and vectors x_t^*, ξ_t^*, η_t^* , $t = 0, \dots, N$ not all equal zero satisfying the Euler-Lagrange discrete inclusions and transversality conditions:*

(i) $(x_t^* - \xi_t^* - \eta_t^*, \xi_{t+1}^*) \in F^*(x_{t+2}^*; (\tilde{x}_t, \tilde{x}_{t+1}, \tilde{x}_{t+2}), t)$, $t = 0, 1, \dots, N-2$,

(ii) $\eta_t^* \in K_A^*(\tilde{x}_t);\ t = 0, \dots, N,\ \xi_N^* \in K_B^*(\tilde{x}_N)$,

(iii) $\Big(\xi_{N-1}^* - x_{N-1}^* + \eta_{N-1}^*\ ,\ \xi_N^* + \eta_N^* - x_N^*\Big) \in \mu\partial_{(x,v_1)}\varphi(\tilde{x}_{N-1}, \tilde{x}_N)$.

In addition, under the regularity condition these conditions are sufficient for optimality of the trajectory $\{\tilde{x}_t\}_{t=0}^N$.

Proof. Obviously, (5) is a convex problem with geometric constraints and by hypotheses of the theorem, $\tilde{u} = (\tilde{x}_0, \dots, \tilde{x}_N)$ is a solution of the problem (5). According to Lemma 1, one has

$$u^*(t) = \Big(0, \dots, 0, x_t^*(t), x_{t+1}^*(t), x_{t+2}^*(t), 0, \dots, 0\Big),$$

where

$$\Big(x_t^*(t), x_{t+1}^*(t), x_{t+2}^*(t)\Big) \in K^*_{gphF}\Big(\tilde{x}_t, \tilde{x}_{t+1}, \tilde{x}_{t+2}\Big), \quad t = 0, 1, \dots, N-2,$$

$$v^*(0) = \big(v_0^*(0), 0, \dots, 0\big),$$
$$v^*(1) = \big(0, v_1^*(1), 0 \dots, 0\big),$$
$$v^*(N) = \big(0, 0 \dots, 0, v_N^*(N)\big).$$

Now using the component-wise representation of (7), we deduce that

$$x_0^*(0) + w_0^*(0) + v_0^*(0) = 0,$$
$$x_1^*(0) + x_1^*(1) + w_1^*(1) + v_1^*(1) = 0, \tag{8}$$
$$x_t^*(t) + x_t^*(t-1) + x_t^*(t-2) + w_t^*(t) = 0, \quad t = 2, \dots, N-2, \tag{9}$$
$$x_{N-1}^*(N-2) + x_{N-1}^*(N-3) + w_{N-1}^*(N-1) = \mu\hat{x}_{N-1}^*,$$
$$x_N^*(N-2) + w_N^*(N) + v_N^*(N) = \mu\hat{x}_N^*,$$
$$(\hat{x}_{N-1}^*, \hat{x}_N^*) \in \partial_{(x,v_1)}\varphi(\hat{x}_{N-1}, \hat{x}_N),\ \mu \in \{0,1\}.$$

On the other hand, by definition of LAM we derive that

$$\Big(x_t^*(t), x_{t+1}^*(t)\Big) \in F^*\big(-x_{t+2}^*(t); (\tilde{x}_t, \tilde{x}_{t+1}, \tilde{x}_{t+2}), t\big), \quad t = 2, \dots, N-2. \tag{10}$$

Introducing the new notations $-x_{t+2}^*(t) \equiv x_{t+2}^*$, $x_{t+1}^*(t) \equiv \xi_{t+1}^*$, $t = 0, \dots, N-2$ and $w_t^*(t) \equiv \eta_t^*$, $t = 0, \dots, N$, $v_N^*(N) \equiv \xi_N^*$, we find from the formulas (9) and (10) that

$$\Big(x_t^* - \xi_t^* - \eta_t^*, \xi_{t+1}^*\Big) \in F^*\Big(x_{t+2}^*; (\tilde{x}_t, \tilde{x}_{t+1}, \tilde{x}_{t+2}), t\Big), \quad t = 2, \dots, N-2, \tag{11}$$

and

$$\eta_t^* \in K_A^*(\tilde{x}_t);\ t = 0, \dots, N,\ \xi_N^* \in K_B^*(\tilde{x}_N). \tag{12}$$

Moreover, it is easy to see that setting, $v_0^*(0) = \xi_0^* - x_0^*$ and $v_1^*(1) = -x_1^*$ in the first relationships of (8), we can generalize the formula (11) to the cases $t = 0, 1$. Finally, for $t = N-1$ and $t = N$ we have $\mu\hat{x}_{N-1}^* = -x_{N-1}^* + \xi_{N-1}^* + \eta_{N-1}^*$, $\mu\hat{x}_N^* = -x_N^* + \xi_N^* + \eta_N^*$ which imply

$$\Big(-x_{N-1}^* + \xi_{N-1}^* + \eta_{N-1}^*, -x_N^* + \xi_N^* + \eta_N^*\Big) \in \mu\partial_{(x,v_1)}\varphi(\tilde{x}_{N-1}, \tilde{x}_N). \tag{13}$$

Thus taking into account the formulas (11), (12) and (13), we complete the first part of the proof of theorem. The proof of the sufficiency conditions, is based on the Theorem 1.30 [1], under the regularity condition, the representation (13) holds with parameter $\mu = 1$ for the point $\hat{u}^* \in \partial_u f(\tilde{u}) \cap K_C^*(\tilde{u})$. □

4. The conditions of optimality for discrete approximation problem

In what follows, we give an idea of the construction of the discrete-approximation problem for the

viability problem (4) with second-order differential inclusions by introducing the first and second-order difference operators:

$$\Delta x(t) = \frac{1}{\delta}\big(x(t+\delta) - x(t)\big),$$

$$\Delta^2 x(t) = \frac{1}{\delta}\big(\Delta x(t+\delta) - \Delta x(t)\big), t = 0, \delta, \dots 1-\delta,$$

where δ is a step on the t-axis and $x(t) \equiv x_\delta(t)$ is a grid functions on a uniform grid on $[0,1]$. We define the following discrete-approximation problem associated with the continuous problem (4)

$$\text{minimize } \phi_0\big(x(1-\delta), \Delta x(1-\delta)\big), \tag{14}$$

$$\Delta^2 x(t) \in F(x(t), \Delta x(t), t),$$

$$t = 0, \delta, 2\delta, \dots, 1-2\delta, \tag{15}$$

$$x(0) = \beta_0,\ \Delta x(0) = \beta_1,$$

$$x(t) \in A,\ t = \delta, \dots, 1,\ x(1) \in B. \tag{16}$$

We apply the results of Theorem 1 to the problem (14) – (16). To do this we rewrite the discrete-approximation inclusion (15) in the relevant form by introducing the following auxiliary set-valued mapping

$$G(x, v_1, t) := 2v_1 - x + \delta^2 F(x, \frac{v_1 - x}{\delta}, t). \tag{17}$$

Then we rewrite the problem (14)−(16) as follows :

$$\text{minimize } \varphi_0\big(x(1-\delta), x(1)\big), \tag{18}$$

$$x(t+2\delta) \in G(x(t), x(t+\delta), t), \tag{19}$$

$$x(0) = \beta_0,\ \Delta x(0) = \beta_1,$$

$$x(t) \in A, t = \delta, \dots, 1,\ x(1) \in B,$$

where $\phi_0\big(x(1-\delta), \Delta x(1-\delta)\big) \equiv \varphi_0\big(x(1-\delta), x(1)\big)$. By Theorem 1 for optimality of the trajectory $\{\tilde{x}(t)\} := \{\tilde{x}(t) : t = 0, \delta, \dots, 1\}$ in the problem, (18) – (19) it is necessary that there exist vectors $\overline{x}^*(t), \overline{\xi}^*(t), \overline{\eta}^*(t)$ and a number $\mu = \mu_\delta \in \{0, 1\}$, not all zero, such that

$$\Big(\overline{x}^*(t) - \overline{\eta}^*(t) - \overline{\xi}^*(t)\ ,\ \overline{\xi}^*(t+\delta)\Big)$$

$$\in G^*\Big(\overline{x}^*(t+2\delta); (\tilde{x}(t), \tilde{x}(t+\delta), \tilde{x}(t+2\delta)), t\Big),$$

$$t = 0, \dots, 1-2\delta, \tag{20}$$

$$\overline{\eta}^*(t) \in K_A^*(\tilde{x}(t))\ ,\ \overline{\xi}^*(1) \in K_B^*(\tilde{x}(1)), \tag{21}$$

$$\Big(\overline{\eta}^*(1-\delta) + \overline{\xi}^*(1-\delta) - \overline{x}^*(1-\delta),$$

$$\overline{\eta}^*(1) + \overline{\xi}^*(1) - \overline{x}^*(1)\Big)$$

$$\in \mu\partial_{(x,v_1)}\varphi_0(\tilde{x}(1-\delta), \tilde{x}(1)). \tag{22}$$

We should express the *LAM* G^* in the above relationship (20) in terms of *LAM* F^*, which plays a central role in our developments in the next results.

Usually, some equivalence theorems are needed for any development in problems given by differential inclusions. Let us first prove two propositions concerning the Hamiltonian functions of the set-valued mappings F and G, and the sets of subdifferential of the Hamiltonian functions H_G and H_F.

Lemma 2. *Let F and G be formula-specified convex set-valued mappings (17). Then there is the following relation between the Hamiltonian H_G and H_F functions:*

$$H_G(x, v_1, v_2^*) = \big\langle 2v_1 - x\ ,\ v_2^*\big\rangle + \delta^2 H_F(x, \frac{v_1 - x}{\delta}, v_2^*).$$

Proof. We get the lemma proof immediately as follows, keeping in mind the definition of the Hamiltonian functions of the set-valued mappings G, F

$$H_G(x, v_1, v_2^*) = \sup\Big\{\langle v_2, v_2^*\rangle : v_2 \in G(x, v_1)\Big\}$$

$$= \big\langle 2v_1 - x, v_2^*\big\rangle + \delta^2 \sup\Big\{\langle v_3, v_2^*\rangle : v_3 \in F(x, \frac{v_1 - x}{\delta})\Big\}$$

$$= \big\langle 2v_1 - x\ ,\ v_2^*\big\rangle + \delta^2 H_F(x, \frac{v_1 - x}{\delta}, v_2^*).$$

□

Lemma 3. *The following relation holds for subdifferentials of the Hamiltonian functions H_G and H_F:*

$$\partial H_G(x, v_1, v_2^*) = \{-v_2^*\} \times \{2v_2^*\}$$

$$+\delta^2 \Theta^* \partial H_F(x, \frac{v_1 - x}{\delta}, v_2^*),$$

where $\Theta = \begin{pmatrix} I & 0 \\ \frac{-I}{\delta} & \frac{I}{\delta} \end{pmatrix}$ *is a $2n \times 2n$ matrix partitioned into submatrices, I, $\frac{-I}{\delta}$, $\frac{-I}{\delta}$ and $n \times n$ zero matrix, where I is an $n \times n$ identity matrix and Θ^* is transposes of Θ.*

Proof. The subdifferential $\partial H_F(x, \frac{v_1 - x}{\delta}, v_2^*)$ should be computed. Notice that Hamiltonian function is concave and it is understood as $\partial_{(x,v_1)} H_G(x, v_1, v_2^*) = -\partial_{(x,v_1)}[-H_G(x, v_1, v_2^*)]$. Let $\psi_i : \mathbb{R}^{2n} \to \mathbb{R}$, $i = 1, 2$, be a convex functions at a point (x, v_1) and $g : \mathbb{R}^{2n} \to \mathbb{R}$ be a convex function continuous at a point $(\psi_1(x, v_1), \psi_2(x, v_1))$. Then for subdifferential of composition $f(x, v_1) = g\big(\psi_1(x, v_1), \psi_2(x, v_1)\big)$ the following formula is valid

$$\partial f(x, v_1) = \Theta^* \partial g\big(\psi_1(x, v_1), \psi_2(x, v_1)\big) \tag{23}$$

where

$$\Theta = \begin{pmatrix} \dfrac{\partial \psi_1}{\partial x} & \dfrac{\partial \psi_1}{\partial v_1} \\ \dfrac{\partial \psi_2}{\partial x} & \dfrac{\partial \psi_2}{\partial v_1} \end{pmatrix}$$

is $2n \times 2n$ matrix $\frac{\partial \psi_1}{\partial x}$, $\frac{\partial \psi_1}{\partial v_1}$, $\frac{\partial \psi_2}{\partial x}$, $\frac{\partial \psi_2}{\partial v_1}$ are Jacobian matrices and Θ^* is transposes of Θ. Taking $\psi_1(x, v_1) \equiv x$, $\psi_2(x, v_1) \equiv \frac{v_1 - x}{\delta}$ in Θ, then it is easy to compute that $\Theta = \begin{pmatrix} I & 0 \\ \frac{-I}{\delta} & \frac{I}{\delta} \end{pmatrix}$ where I is an $n \times n$ identity matrix. Then, for fixed v_2^* setting $f(x, v_1) = H_F(x, \frac{v_1 - x}{\delta}, v_2^*)$ and using Moreau-Rockafeller Theorem 1.29 [1], Lemma 2 and the formula (23), we obtain the desired result. □

In the construction of LAM for the original discrete approximation problem (14)-(16), the following theorem which is definitely of independent interest plays a crucial role.

Theorem 2. *If G is a convex set-valued mapping defined by (17), then the following statements for the $LAMs$ are equivalent:*

(a) $(x^*, v_1^*) \in G^*(v_2^*; (x, v_1, v_2))$, $v_2 \in G(x, v_1; v_2^*)$,

$$(b) \quad \Big(\frac{x^* + v_1^* - v_2^*}{\delta^2}, \frac{v_1^* - 2v_2^*}{\delta}\Big) \in F^*\Big(v_2^*; (x, \tfrac{v_1 - x}{\delta}, \tfrac{x - 2v_1 + v_2}{\delta^2})\Big),$$
$$\tfrac{x - 2v_1 + v_2}{\delta^2} \in F(x, \tfrac{v_1 - x}{\delta}; v_2^*),\ v_2^* \in \mathbb{R}^n$$

where $G(x, v_1; v_2^*)$ is the argmaximum set for mapping G.

Proof. By useful Frobenius formula, the following inverse of 2×2 block matrices holds if A, B are invertible

$$\begin{pmatrix} A & D \\ C & B \end{pmatrix}^{-1} = \begin{pmatrix} (A - DB^{-1}C)^{-1} & -A^{-1}D\Delta^{-1} \\ -\Delta^{-1}CA^{-1} & \Delta^{-1} \end{pmatrix}$$

where $\Delta = B - CA^{-1}D$ is the Schur complement of A. Therefore, denoting $A = \delta^2 I, B = \delta I$, $C = 0, D = -\delta I$, we compute that

$$(\delta^2 \Theta^*)^{-1} = \begin{pmatrix} \delta^2 I & -\delta I \\ 0 & \delta I \end{pmatrix}^{-1} = \begin{pmatrix} \frac{I}{\delta^2} & \frac{I}{\delta^2} \\ 0 & \frac{I}{\delta} \end{pmatrix}. \quad (24)$$

Furthermore, by Lemma 3 it is easy to see that $(x^*, v_1^*) \in \partial_{(x,v_1)} H_G(x, v_1, v_2^*)$ and

$$(\delta^2 \Theta^*)^{-1}(x^* + v_2^*, v_1^* - 2v_2^*) \in \partial_{(x,v_1)} H_F(x, \frac{v_1 - x}{\delta}, v_2^*) \quad (25)$$

are equivalent. Then from (24) and (25), it means that $(x^*, v_1^*) \in \partial_{(x,v_1)} H_G(x, v_1, v_2^*)$ if and only if

$$\Big(\frac{x^* + v_1^* - v_2^*}{\delta^2}, \frac{v_1^* - 2v_2^*}{\delta}\Big) \in \partial_{(x,v_1)} H_F(x, \frac{v_1 - x}{\delta}, v^*). \quad (26)$$

Since $F^*(v_2^*; (x, v_1, v_2)) = \partial_{(x,v_1)} H_F(x, v_1, v_2^*)$, $v_2 \in F(x, v_1; v_2^*)$ and by using the fact that $\partial_{(x,v_1)} H_F(x, v_1, v_2^*) = -\partial_{(x,v_1)}(-H_F(x, v_1, v_2^*))$, we express (26) in term of $LAMs$. Here take into account that $v_2 \in G(x, v_1; v_2^*)$ and $\frac{x - 2v_1 + v_2}{\delta^2} \in F(x, \frac{v_1 - x}{\delta}; v_2^*)$ ensure that the $LAMs$ are nonempty at a given points. □

Lemma 4. *It turns out that the following inclusions are equivalent:*

$$(1) \quad \Big(\overline{x}^*,\ \overline{y}^*\Big) \in \partial\varphi_0\big(z_0\big),$$
$$(2) \quad \Big(\overline{x}^* + \overline{y}^*,\ \delta\overline{y}^*\Big) \in \partial\phi_0\big(w_0\big).$$

where $z_0 = (x^0, y^0) \in dom\varphi_0$ and $w_0 = (x^0, \frac{y^0 - x^0}{\delta}) \in dom\phi_0$.

Proof. By the definition of subdifferential sets we can write

$$\partial\varphi_0(z_0) = \Big\{\big(\overline{x}^*, \overline{y}^*\big) : \varphi_0(x, y) - \varphi_0(x^0, y^0) \geq \langle \overline{x}^*, x - x^0\rangle + \langle \overline{y}^*, y - y^0\rangle, \quad \forall\ (x, y) \in \mathbb{R}^{2n}\Big\}. \quad (27)$$

$$\partial\phi_0\big(w_0\big) = \Big\{\big(x^*, y^*\big) : \phi_0\big(w\big) - \varphi_0\big(w_0\big) \geq \langle x^*, x - x^0\rangle + \big\langle y^*, \frac{y - x}{\delta} - \frac{y^0 - x^0}{\delta}\big\rangle \quad \forall\ w = \big(x, \frac{y - x}{\delta}\big) \in \mathbb{R}^{2n}\ \Big\}.$$

This latter relationship gives that

$$\partial\phi_0\big(w_0\big) = \Big\{\big(x^*, y^*\big) : \varphi_0(x, y) - \varphi_0(x^0, y^0) \geq \Big\langle x^* - \frac{y^*}{\delta},\ x - x^0\Big\rangle + \Big\langle \frac{y^*}{\delta}\ ,\ y - y^0\Big\rangle, \quad \forall\ w \in \mathbb{R}^{2n}\ \Big\}. \quad (28)$$

When (27) and (28) are compared, we deduce that

$$\overline{x}^* = x^* - \frac{y^*}{\delta}, \quad \overline{y}^* = \frac{y^*}{\delta}$$

or, in other words,

$$x^* = \overline{x}^* + \overline{y}^*, \quad y^* = \delta\overline{y}^*.$$

The proof of theorem is completed. □

Theorem 3. *Let F be a convex set-valued mapping and ϕ_0 be proper convex functional and continuous at the points of some feasible trajectory. Then for optimality of the trajectory $\{\tilde{x}(t)\}$ in the discrete approximation problem, it is necessary that there exist a number $\mu = \mu_\delta \in \{0, 1\}$ and vectors $x^*(t), \eta^*(t), v^*(t)$ which are not all equal zero, satisfying the approximate Euler-Lagrange and transversality inclusions:*

$$(a) \ \Big(\Delta^2 x^*(t) + \Delta v^*(t) - \eta^*(t), v^*(t + \delta)\Big) \in F^*(x^*(t + 2\delta); (\tilde{x}(t), \Delta\tilde{x}(t), \Delta^2\tilde{x}(t), t)), \quad t = 0, \delta, \ldots, 1 - 2\delta,$$
$$(b)\ \eta^*(t) \in K_A^*(\tilde{x}(t))\ ,\ \xi^*(1) \in K_{A \cap B}^*(\tilde{x}(1)),$$

$$(c)\ \Big(v^*(1-\delta)+\Delta x^*(1-\delta)+\xi^*(1)$$
$$+\delta\eta^*(1-\delta)\ ,\ \delta\xi^*(1)-x^*(1)\Big)$$
$$\in \mu\partial\phi_0(\tilde{x}(1-\delta),\Delta\tilde{x}(1-\delta)).$$

And, under the regularity condition these conditions are also sufficient for optimality of the trajectory $\{\tilde{x}(t)\}$.

Proof. By taking into account the condition (20) and by Theorem 2, it can be shown that

$$\Big(\frac{\overline{x}^*(t)-\overline{\eta}^*(t)-\overline{\xi}^*(t)+\overline{\xi}^*(t+\delta)-x^*(t+2\delta)}{\delta^2},$$
$$\frac{\overline{\xi}^*(t+\delta)-2x^*(t+2\delta)}{\delta}\Big)$$
$$\in F^*\Big(\overline{x}^*(t+2\delta);(\tilde{x}(t),\Delta\tilde{x}(t),\Delta^2\tilde{x}(t),t)\Big),$$
$$t=0,\dots,1-2\delta.$$

Denoting $\overline{v}^*(t)=\frac{\overline{\xi}^*(t)-2x^*(t+\delta)}{\delta}$, we get

$$\Big(\Delta^2\overline{x}^*(t)+\Delta\overline{v}^*(t)-\frac{\overline{\eta}^*(t)}{\delta^2},\overline{v}^*(t+\delta)\Big)$$
$$\in F^*(\overline{x}^*(t+2\delta);(\tilde{x}(t),\Delta\tilde{x}(t),\Delta^2\tilde{x}(t),t)),$$
$$t=0,\dots,1-2\delta.$$

Now observe that $LAMF^*$ is positive homogeneous on the first argument and setting $\delta\overline{x}^*(t)$, $\delta\overline{v}^*(t)$ and $\overline{\eta}^*(t)$ are denoted by $x^*(t)$, $v^*(t)$ and $\delta\eta^*(t)$, respectively, we derive the approximate Euler-Lagrange inclusion of theorem. Moreover, by the formula (22) and Lemma 4, we have

$$\Big(\overline{\eta}^*(1-\delta)+\overline{\xi}^*(1-\delta)-\overline{x}^*(1-\delta)+\overline{\eta}^*(1)+\overline{\xi}^*(1)-\overline{x}^*(1),$$
$$\delta\big(\overline{\eta}^*(1)+\overline{\xi}^*(1)-\overline{x}^*(1)\big)\Big)$$
$$\in \mu\partial\phi_0(\tilde{x}(1-\delta),\Delta\tilde{x}(1-\delta)).$$

Then setting $\overline{\xi}^*(1)=\xi^*(1)-\delta\eta^*(1)$, we find that the transversality condition of theorem:

$$\Big(v^*(1-\delta)+\Delta x^*(1-\delta)+\xi^*(1)+\delta\eta^*(1-\delta)\ ,$$
$$\delta\xi^*(1)-x^*(1)\Big)$$
$$\in \mu\partial\phi_0(\tilde{x}(1-\delta),\Delta\tilde{x}(1-\delta)).$$

By the condition (21), we have $\delta\eta^*(t)\in K_A^*(\widetilde{x}(t))$ and $\xi^*(1)-\delta\eta^*(1)\in K_B^*(\widetilde{x}(1))$. Then $\xi^*(1)\in K_A^*(\widetilde{x}(1))+K_B^*(\widetilde{x}(1))=K_{A\cap B}^*(\widetilde{x}(1))$ which completes the proof of theorem. □

Remark 1. *For future directions, we note that the results obtained in this section could be useful for deriving optimality conditions for the second-order viability problem (4) given by differential inclusions with endpoint constraint. At least we emphasize that the key to our success is to pass formally to the limit in the conditions of Theorem 3. Consequently, by setting* $\mu=1$ *and passing formally to the limit as* $\delta\to 0$*, the sufficient conditions of optimality can be formulated for the continuous problem (4). Moreover, by using the functional analysis approach in the convex problem, the necessity of these conditions for optimality can be justified.*

Corollary 1. *Let* F *be a convex set-valued mapping and* ϕ *be proper convex functional and continuous at the points of some feasible trajectory. Then for optimality of the trajectory* $\tilde{x}(t)$ *in the problem (4), it is sufficient that there exist a number* μ *and vectors* $x^*(t),\eta^*(t),v^*(t)$ *which are not all equal zero, satisfying the second-order Euler-Lagrange type adjoint*

$$(i)\ (x^{*''}(t)+v^{*'}(t)-\eta^*(t),v^*(t))$$
$$\in F^*(x^*(t);(\tilde{x}(t),\tilde{x}'(t),\tilde{x}''(t),t)),\ t\in[0,1]$$
$$(ii)\quad \tilde{x}''(t)\in F(x(t),x'(t);x^*(t),t)\ ,a.e.\ t\in[0,1],$$
$$\eta^*(t)\in K_A^*(\tilde{x}(t)),\ t\in[0,1)\ ,$$

and the transversality conditions at the endpoint $t=1$ *consist of the following*

$$(iii)\ \Big(v^*(1)+x^{*'}(1)+\xi^*(1),-x^*(1)\Big)$$
$$\in \mu\partial\phi(\tilde{x}(1),\tilde{x}'(1)),$$
$$\xi^*(1)\in K_{A\cap B}^*(\tilde{x}(1)).$$

Note that the transformation to the continuous problem (4) is in any case, a separate subject of discussion and is therefore omitted. Note also that this construction of optimality conditions in the presented paper for second-order discrete and discrete approximation inclusions can be useful for the optimization of an arbitrary order discrete and ordinary differential inclusions.

5. Applications

In this section, we describe some interesting applications of the problem (1)-(3). At first, let us consider the problem with the "linear discrete" structure

$$\text{minimum}\ \varphi(x_{N-1},x_N),$$
$$x_{t+2}\in F(x_t,x_{t+1},t)\ ,\ t=0,\dots,N-2,$$
$$F(x,v_1)=C_0x+C_1v_1+Du,\ u\in U \qquad (29)$$
$$x_0=\alpha_0\ ,\ x_1=\alpha_1,$$
$$x_t\in A,\ t=0,\dots,N,\ x_N\in B,$$

where C_0,C_1 are $n\times n$, D is $n\times r$ matrix, $D\subset\mathbb{R}^r$ is convex closed set, φ is continuously differentiable function. It is required to find controlling parameters $\widetilde{u}_t\in U$ $(t=0,\dots,N)$ such that the corresponding trajectory $\{\widetilde{x}_t\}_{t=0}^N$ minimizes φ. We observe that in this case $F(x,v_1)=\{v_2=C_0x+C_1v_1+Du:\ u\in U\}$. It is not hard to

compute that

$$F^*(v_2^*;(x,v_1,v_2)) = \begin{cases} \big(C_0^*v_2^*, C_1^*v_2^*\big), & -D^*v_2^* \in K_U^*(u), \\ \emptyset, & -D^*v_2^* \notin K_U^*(u), \end{cases} \quad (30)$$

where $v_2 = C_0x + C_1v_1 + Du$, $u \in U$, C_0^*, C_1^* and D^* are transposed matrices. Then by using Theorem 1 and from the formula (30), we have $x_t^* - \xi_t^* - \eta_t^* = C_0^*x_{t+2}^*$ and $\xi_{t+1}^* = C_1^*x_{t+2}^*$, $t = 0,\dots,N-2$ and $-D^*x_{t+2}^* \in K_U^*(u)$, $\eta_t^* \in K_A^*(\widetilde{x}_t)$ and $\xi_N^* \in K_B^*(\widetilde{x}_N)$. Clearly the transversality conditions consist of the following:

$$\xi_{N-1}^* - x_{N-1}^* + \eta_{N-1}^* = \mu\varphi_x$$
$$\xi_N^* + \eta_N^* - x_N^* = \mu\varphi_{v_1}.$$

Moreover $-D^*v_2^* \in K_U^*(u)$ means that the Weierstrass Pontryagin maximum condition

$$\langle D\widetilde{u}_t, x_t^*\rangle = \sup_{u\in U}\langle Du, x_t^*\rangle \quad (31)$$

is satisfied. The regularity condition is superfluous for linear problems. It is, therefore, necessary and sufficient for the optimality of the trajectory $\{x_t\}_{t=0}^N$ in linear problem that there exists $\{x_t^*\}_{t=0}^N$ satisfying the second-order Euler-Lagrange differential equation with $\mu = 1$ and Weierstrass Pontryagin maximum principle (31).

Now, let us consider another example where a set-valued mapping is defined by nonlinear inequality:

$$\begin{gathered} \text{minimum } \varphi(x_{N-1}, x_N), \\ x_{t+2} \in F(x_t, x_{t+1}, t)\ ,\ t = 0,\dots,N-2, \\ F(x,v_1) = \{v_2 : \Psi(x,v_1,v_2) \le 0\} \quad (32) \\ x_0 = \alpha_0\ ,\ x_1 = \alpha_1, \\ x_t \in A,\ t = 0,\dots,N,\ x_N \in B, \end{gathered}$$

where the function Ψ and φ are continuously differentiable functions. Then according to the Theorem 2.13. [1] by elementary computations we find that

$$\begin{aligned} F^*(v_2^*;(x,v_1,v_2)) = \Big\{ \big(&-\mu\Psi'_x(x,v_1,v_2), \\ -\mu\Psi'_{v_1}(x,v_1,v_2)\big) &: v_2^* = \mu\Psi'_{v_2}(x,v_1,v_2), \\ \mu\Psi(x,v_1,v_2) = 0\ ,\mu \ge 0,\Big\} \end{aligned} \quad (33)$$

where $\Psi'_x(x,v_1,v_2)$, $\Psi'_{v_1}(x,v_1,v_2)$ and $\Psi'_{v_2}(x,v_1,v_2)$ are gradient vectors with respect to x, v_1, v_2, correspondingly. So by using Theorem 1, we get the following relations

$$\begin{gathered} x_t^* - \xi_t^* - \eta_t^* = -\mu_t\Psi'_x(\widetilde{x}_t, \widetilde{x}_{t+1}, \widetilde{x}_{t+2}), \\ \xi_{t+1}^* = -\mu_t\Psi'_{v_1}(\widetilde{x}_t, \widetilde{x}_{t+1}, \widetilde{x}_{t+2}), \\ x_{t+2}^* = \mu_t\Psi'_{v_2}(\widetilde{x}_t, \widetilde{x}_{t+1}, \widetilde{x}_{t+2}), \\ \mu_t\Psi(\widetilde{x}(t), \widetilde{x}_{t+1}, \widetilde{x}_{t+2}) = 0, \quad (34) \\ \mu_t \ge 0,\ t = 0,\dots,N-2, \end{gathered}$$

$$\eta_t^* \in K_A^*(\widetilde{x}_t),\ t = 0,\dots,N,\ \xi_N^* \in K_B^*(\widetilde{x}_N), \quad (35)$$

and transversality conditions

$$\begin{gathered} \xi_{N-1}^* - x_{N-1}^* + \eta_{N-1}^* = \varphi_x(\widetilde{x}_{N-1}, \widetilde{x}_N) \\ \xi_N^* + \eta_N^* - x_N^* = \varphi_{v_1}(\widetilde{x}_{N-1}, \widetilde{x}_N). \quad (36) \end{gathered}$$

So we have proved the following theorem.

Theorem 4. *If Ψ and φ are continuously differential functions and the cone $K_{gphF}(x,v_1,v_2)$ is non-empty for all $(\widetilde{x}_t, \widetilde{x}_{t+1}, \widetilde{x}_{t+2})$ then the existence of $\{x_t^*, \xi_t^*, \eta_t^*\}$ satisfying the adjoint discrete inclusions (34)-(36) is necessary and sufficient for the optimality of the trajectory $\{\widetilde{x}_t\}_{t=0}^N$ of problem (32).*

Acknowledgments. The author wishes to express her sincere thanks to anonymous reviewers for valuable suggestions which improved the final manuscript.

References

[1] Mahmudov, E.N. (2011). *Approximation and optimization of discrete and differential inclusions*. Elsevier, Boston, USA.

[2] Mordukhovich, B.S. (2006). *Variational analysis and generalized differentiation, I: Basic theory; II: Applications,* Grundlehren Series (Fundamental Principles of Mathematical Sciences), Vol. 330 and 331, Springer.

[3] Bohner, M., Ding, Y., & Došly, O. (2015). *Difference equations, discrete dynamical systems and applications*. Springer, Switzerland.

[4] Mahmudov, E.N. (2014). Approximation and optimization of higher order discrete and differential inclusions. *Nonlin. Differ. Equat. Appl.*, 21, 1-26.

[5] Mahmudov, E.N. (1991). A two-parameter optimal control problem for systems of discrete inclusions. *Automat. Remote Control*, 52(3), 353-362.

[6] Mahmudov, E.N., & Mardanov, M.J. (2020). On duality in optimal control problems with second-order differential inclusions and initial-point constraints. *Proceed. Institute Math. Mech. Nat. Acad. Sci. Azerb.*, 46(1), 115-128.

[7] Özdemir, N. & Evirgen, F. (2010). A dynamic system approach to quadratic programming problems with penalty method. *Bulletin of the Malaysian Mathematical Sciences Society. Second Series*, 33(1), 79-91.

[8] Ulus, A.Y. (2018). On discrete time infinite horizon optimal growth problem. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, 8(1), 102-116.

[9] Mahmudov, E.N. (2015). Optimization of second order discrete approximation inclusions. *Numeric. Funct. Anal. Optim.*, 36, 624-643.

[10] Mahmudov, E.N. (2018). Optimization of Mayer problem with Sturm-Liouville-type differential inclusions. *J. Optim. Theory Appl.*, 177, 345-375.

[11] Mahmudov, N.I., Vijayakumar, V., & Murugesu, R. (2016). Approximate controllability of second-order evolution differential inclusions in Hilbert spaces. *Mediterr. J. Math.*, 13, 3433-3454.

[12] Auslender, A., & Mechler, J. (1994). Second order viability problems for differential inclusions. *J. Math. Anal. Appl.*, 181, 205-218.

[13] Veliov, V. (1992). Second-order discrete approximation to linear differential inclusions. *SIAM Journal on Numerical Analysis*, 29(2), 439–451.

[14] Donchev, T., Farkhi, E., & Mordukhovich, B.S. (2007). Discrete approximations, relaxation, and optimization of one-sided Lipschitzian differential inclusions in Hilbert spaces. *Journal of Differential Equations*, 243(2), 301-328.

[15] Agarwal, R.P., & O'Regan, D. (2002). Fixed-point theory for weakly sequentially upper-semicontinuous maps with applications to differential inclusions. *Nonlinear Oscillat.*, 5(3), 277-286.

[16] Boltyanskii, V.G. (1978). *Optimal control of discrete systems*. John Wiley, New York, USA.

[17] Haddad, T., & Yarou, M. (2006). Existence of solutions for nonconvex second-order differential inclusions in the infinite dimensional space. *Electron. J. Differ. Equat.*, 2006(33), 1-8.

[18] Marco, L., & Murillo, J.A. (2001). Lyapunov functions for second-order differential inclusions: a viability approach. *J. Math. Anal. Appl.*, 262(1), 339-354.

[19] Lupulescu, V. (2005). Viable solutions for second order nonconvex functional differential inclusions. *Electron. J. Differ. Equat.*, 110, 1-11.

[20] Mahmudov, E.N. (2020). Optimal control of higher order differential inclusions with functional constraints. *ESAIM: Control, Optimisation and Calculus of Variations*, 26, 1-23.

Sevilay Demir Sağlam *is currently working as a research assistant at the Department of Mathematics, Istanbul University, Istanbul, Turkey. She received M.Sc. and Ph.D. degrees from the Department of Mathematics, Istanbul University in 2012 and 2017, respectively. Her current research interests are the polyhedral optimization, control theory, duality theory and the conditions of optimality for discrete and differential inclusions given by set-valued mappings.*
https://orcid.org/0000-0003-4615-6863

An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.11, No.2, pp.216-226 (2021)
https://doi.org/10.11121/ijocta.01.2021.001091

RESEARCH ARTICLE

An application of the whale optimization algorithm with Levy flight strategy for clustering of medical datasets

Ayşe Nagehan Mat, Onur İnan and Murat Karakoyun *

Department of Computer Engineering, Necmettin Erbakan University, Turkey
nagehanmat@gmail.com, oinan@erbakan.edu.tr, mkarakoyun@erbakan.edu.tr

ARTICLE INFO

Article history:
Received: 4 March 2021
Accepted: 25 May 2021
Available Online: 22 June 2021

Keywords:
Clustering
Whale optimization algorithm
Levy flight
K-means
K-medoids
Fuzzy c-means

AMS Classification 2010:
68T01; 68T05; 68T20

ABSTRACT

Clustering, which is handled by many researchers, is separating data into clusters without supervision. In clustering, the data are grouped using similarities or differences between them. Many traditional and heuristic algorithms are used in clustering problems and new techniques continue to be developed today. In this study, a new and effective clustering algorithm was developed by using the Whale Optimization Algorithm (WOA) and Levy flight (LF) strategy that imitates the hunting behavior of whales. With the developed WOA-LF algorithm, clustering was performed using ten medical datasets taken from the UCI Machine Learning Repository database. The clustering performance of the WOA-LF was compared with the performance of k-means, k-medoids, fuzzy c-means and the original WOA clustering algorithms. Application results showed that WOA-LF has more successful clustering performance in general and can be used as an alternative algorithm in clustering problems.

1. Introduction

Due to the increasingly widespread digitalization processes at global and local level, large-scale data are obtained in many different fields. It is very important to process the data accurately and quickly in order to make more effective use of the large-scale data and extract meaningful information. For this reason, new methods are constantly being developed for the efficient use of knowledge in many areas such as industry, banking, marketing, medicine, engineering and economics, where data mining techniques are being applied.

With the ever-developing science and technology, optimization problems are also increasing. These problems are considered to be more complex and difficult because they are large-scale and have many factors. Existing optimization algorithms can be low-performance and slow in solving complex optimization problems. For this reason, many researchers have focused on improving existing optimization algorithms. Hybridization is one of these improvement efforts, and it means using two or more algorithms as a hybrid. The purpose of hybridization is to use the advantages of each algorithm at the highest level and to minimize the disadvantages. When compared with its components a hybrid algorithm generally has a stronger and robust structure and can effectively solve complex optimization problems [1-3].

Data mining reveals hidden patterns and relationships in the data by using advanced analysis techniques such as machine learning, artificial intelligence, and statistics. Clustering, which is one of the data mining techniques, collects data of similar characteristics together and ensures the data community to be divided into clusters / groups. Looking at literature, the use of heuristic algorithms in clustering emerges as an alternative to the traditional clustering techniques [4, 5].

Selim and El-Sultan [6], used simulated annealing approach for clustering problem. The predetermined parameters of the algorithm are discussed and shown to converge to the global solution in the clustering problem. Mualik and Mukhopadhyay [7] presented a unified clustering algorithm. They combined the simulated annealing algorithm with artificial neural networks to improve solution quality. The proposed hybrid algorithm was used to cluster three true microarray datasets, and the results of the proposed approach were compared with some commonly used clustering algorithms. The results showed the superiority of the new algorithm. Mualik and Bundyopadyay [8] presented a genetic algorithm-based

*Corresponding author: mkarakoyun@erbakan.edu.tr

approach to solve the cluster problem. They evaluated the performance of the approach using synthetic and real datasets. Shelokar et al. [9] proposed a clustering algorithm based on ant colony optimization (ACO). The proposed algorithm has been tested on some artificial and real datasets. The performance of this technique was promising compared to popular algorithms such as genetic algorithm, simulated annealing and Tabu search. Merwe et al. [10] used the particle swarm optimization (PSO) algorithm to solve the clustering problem. They used PSO clustering, where the swarm particles were selected by the k-means algorithm, and a hybrid method. Both methods were compared with the k-means algorithm and the proposed algorithms were observed to have better results. Tunchan [11] introduced a new PSO approach that is effective in clustering problem, easy to adjust, applicable in cases where cluster number is known or unknown. Karaboğa et al. [12] used the artificial bee colony algorithm (ABC) to solve the clustering problem. The results on the test data showed that the proposed algorithm was superior performance compared to the PSO algorithm and some other approaches. Additionally, the authors found that the ABC algorithm may be suitable for solving multivariate clustering problem. Zhang et al. [5] proposed an artificial bee colony (ABC) clustering algorithm. In this algorithm, Deb's rule was used instead of greedy approach in the selection process. The algorithm tested in several well-known datasets was compared with other popular heuristic scanning algorithms in clustering. The results were successful in terms of the quality of the clusters. Armando and Farmani [13] proposed a method that uses k-means and ABC algorithms together. In the proposed algorithm ABC algorithm assist to increase the efficiency of the k-means algorithm in finding the global optimum solution. Karthikeyan and Christopher [14] proposed an algorithm with a combination of PSO and ABC algorithms. The performance of the proposed approach was determined by comparison with other clustering algorithms. Sandeep and Pankaj [15] proposed a new hybrid sequential clustering approach that uses PSO and fuzzy k-means algorithms sequentially in data clustering. Experimental results show that the new approach improves the quality of the generated clusters and avoids local minima.

Recently, Mirjalili and Lewis [16] introduced a new metaheuristic optimization algorithm called the whale optimization algorithm (WOA) that imitates the bubble-hunting strategy of humpback whales. The WOA algorithm was tested with 29 mathematical comparison optimization problems, and the performance of the algorithm was compared to other metaheuristic algorithms such as PSO [17], Differential Evolution [18], Gravitational Search Algorithm [19] and Fast Evolutionary Programming [20]. As a result of comparisons, WOA was accepted to be able to compete with other well-known metaheuristic methods. Nasiri and Khiyabani [21] used WOA in clustering in their work. They compared the results with other popular algorithms such as k-means, PSO, artificial bee colony, differential evolution and genetic algorithm. The intra-cluster distance function and standard deviation values showed that the whale optimization algorithm could be successfully applied in solving the clustering problem. Canayaz and Özdağ [22] obtained a feature vector called BEST in their proposed method by applying clustering with WOA. They applied clustering by calculating the Manhattan distance between this vector and the test data. The results showed that WOA can be used in clustering problem and is faster than the artificial atom algorithm when evaluated in terms of running time.

Levy flight is a random walking class. It can be defined as a generalized Brownian motion to include non-Gaussian randomly distributed step lengths for moving distance [23]. Levy flight can depict many natural and artificial conditions such as liquid dynamics, earthquake analysis, diffusion of fluorescent molecules, cooling behavior, noise [24]. Levy flight was also used by Pereyra and Hajj [25] in skin tissue by ultrasound and Al-teemy [23] for Ladar (Laser Detection and Ranging) screening. In addition to these fields, it has played an important role in many fields in computer science. Internet traffic models by Terdik and Gyres [26], delay and interruption tolerance network by Chen [24], Sutantyo et al. [27] has been used by the multi-robot search procedure and by Rhee [28] in areas of human mobility. In addition, Levy flight, which is similar to the food search behavior of many animals such as albatross, wasps and deer, has been used in conjunction with natural algorithms to improve the performance of algorithms [29, 30]. Yang and Deb [31] used the Levy flight distribution to create a new cuckoo in Cuckoo Search. In addition, Yang [32] introduced a new Levy flight-Firefly algorithm (LFA) by combining the firefly algorithm (FA) with the Levy flight search strategy to improve randomness.

In this article, the advantages of the whale optimization algorithm such as having a small number of parameters and avoiding local minima trap, have been effective in selecting it for the clustering problem. The aim of the study is to achieve better results with simple solutions and to cluster unlabeled data. It has been used in conjunction with the Levy flight search strategy to strengthen WOA's global search and perform a complete search compared to existing methods. The proposed algorithm has been tested on ten medical datasets selected from the UCI [33] database. Clustering performance of the proposed algorithm is given comparatively with k-means, k-medoids and fuzzy c-means algorithms.

The rest of this paper organized as follow: Clustering problem is presented with details in section 2 and the algorithms used in the study are mentioned with details in section 3. Experimental application is explained in section 4 and experimental results are given comparatively in section 5. Finally, conclusions and some future research direction are in section 6.

2. Clustering problem

Clustering, which is one of the data mining applications, is based on grouping the elements in the data collection according to their similarities (or dissimilarities). Clustering methods help to divide the data that is not known to which group it belongs to sub-sections according to similar characteristics [34, 35]. The main purpose of clustering is to group objects using their characteristic properties. While clustering, attention is paid to cluster the individuals in the same group similar to each other, and individuals from different groups to be in a separate group. As seen in Figure 1, it wants to be that the individuals within the same cluster have little distance and different clusters have more distance from each other [36].

Clustering is done in a population where precise information about grouping of variables in the dataset is not available. Observation results of n data taken from this population are regarded for p variables. In clustering, individuals of similar nature are combined and divided into clusters. Clustering allows gathering observation results with little loss [37].

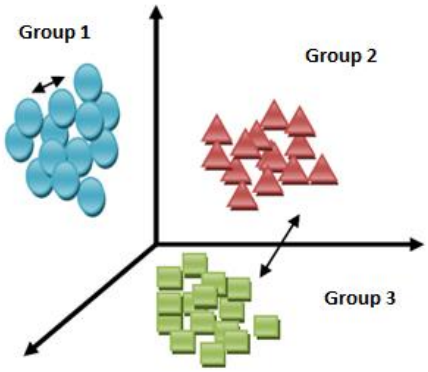


Figure 1. Distance between objects and clusters.

While there are many approaches to the clustering problem in the literature, clustering algorithms are mainly divided into two as hierarchical and non-hierarchical clustering algorithms. When applying hierarchical clustering methods, the cluster number does not need to be determined. In these methods, the cluster number is determined after the clustering process is finished. However, in non-hierarchical clustering methods, cluster number information is required to make clustering. In summary, the purpose of non-hierarchical clustering is to divide n data samples into k sets. In terms of time complexity, it is seen that hierarchical clustering methods are quadratic, while non-hierarchical clustering methods are linear [38, 39]. Clustering algorithms can be categorized as given in Figure 2.

When performing cluster analysis, a similarity or distance criterion should be selected in the first stage. Then, which clustering technique (such as hierarchical or non-hierarchical) should be determined. In the next step, the type of clustering method to be used for the selected technique is chosen and in the last stage, the number of clusters is determined in order to interpret the cluster result [41]. In this study, the distance between each data sample and the cluster center to which the data sample belongs was calculated according to the Euclidean distance [42].

3. Reflective process

Clustering has been applied to ten medical datasets using the Whale Optimization Algorithm, which has been introduced in the literature in recent years and Levy flight search strategy. Clustering results are given in comparison with results of k-means, k-medoids and fuzzy c-means clustering algorithms.

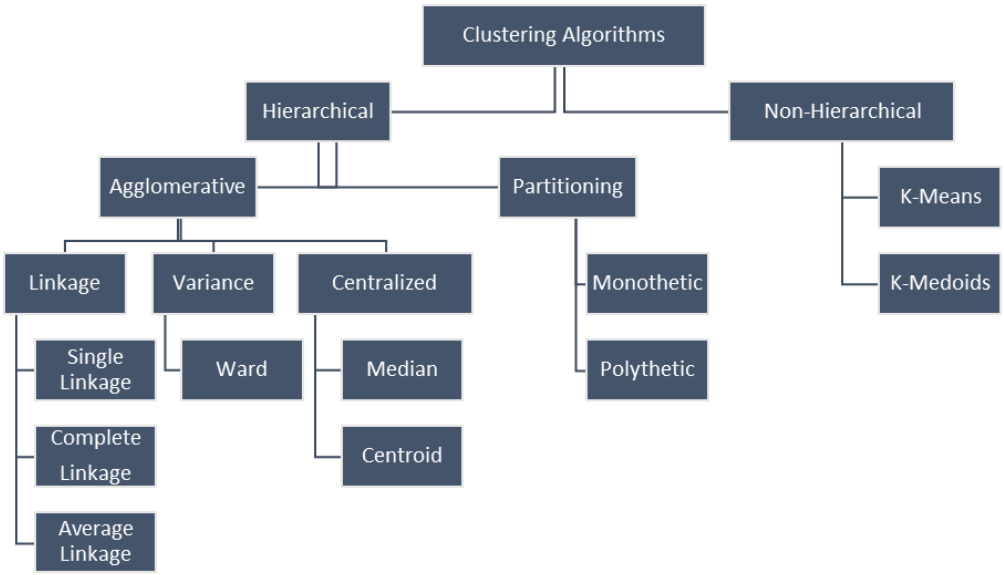


Figure 2. The general categorization of the clustering algorithms [40].

3.1. K-means clustering algorithm

Developed in 1967 by J. B. MacQueen, k-means is one of the oldest clustering algorithms [43]. One of the widely used unsupervised learning algorithms, k-means allows each object to be included in only one cluster. Therefore, it is a sharp clustering algorithm. This method is based on the idea that the cluster center represents the cluster [44].

In the evaluation of the K-means clustering method, the sum of squared errors (SSE) is often used. The lowest value of SSE represents the best result. The sum of squares of objects distance (x) from cluster centers (m_i) is calculated by the Eq. (1) [45].

$$SSE = \sum_{i=1}^{K} \sum_{x \, \epsilon \, C_i} dist^2 \ (m_i \, , x \,) \tag{1}$$

The k-means algorithm is basically based on the principle of dividing n objects into k sets. It is aimed here that the clusters are as dense as possible in terms of objects and at a maximum distance from the other clusters. Cluster similarity is measured by the average value of the objects in the cluster, which represents the center of gravity of the cluster [46].

3.2. K-medoids clustering algorithm

The k-medoids algorithm is based on finding k objects that represent various structural features of the data [47]. The representative object is the closest point to the cluster center and is called the medoid. When dividing the group of objects into k clusters, the aim is to gather similar objects together and create a structure in which the objects in different sets are unalike. Although there are many different types of the k-medoids algorithm, the first introduced k-medoids algorithm is the PAM (Partitioning Around Medoids) algorithm. PAM, similar to the k-means algorithm, determines the randomly chosen k numbers as the cluster center. With each new element joining the cluster, the elements of the cluster are tested and the point that can contribute the most to the cluster development is determined. A swap operation is performed so that this determined point is the new center of the cluster and the old cluster center is the ordinary cluster element [48].

3.3. Fuzzy c-means clustering algorithm

Fuzzy C-Means (FCM) algorithm can be called fuzzy version of the K-means clustering algorithm [49]. FCM algorithm is the best known and widely used method among fuzzy partitioning clustering techniques. It was proposed by Dunn in 1973 and was developed by Bezdek in 1981 [50]. Fuzzy c-means algorithm works with purpose function logic. Objects have a membership value in the range [0, 1] for each cluster, and according to fuzzy logic they belong to that cluster depending on these membership values. The sum of the membership values of an object calculated for all clusters must be 1. The cluster with the highest membership value refers to the cluster center where the object is closest. Clustering is completed when the fitness function converge to the specified value.

The algorithm uses it to minimize the objective function in Eq. (2), which is the generalization of the least squares method.

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^{m} \left\| x_{i-} \, c_j \right\|^2 , 1 \leq m < \infty \tag{2}$$

The algorithm is started with the randomly generated U membership matrix. In the second step, center vectors are calculated using Eq. (3) [50].

$$c_{j=} \frac{\sum_{i=1}^{N} u_{ij}^{m} \, x_i}{\sum_{i=1}^{N} u_{ij}^{m}} \tag{3}$$

The U matrix is reconstructed with the help of Eq. (4) according to the calculated cluster centers. The previous state of the U matrix is compared with the final state, and the operations are repeated until the difference is less than ε [51].

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\| x_i \, - \, c_i \|}{\| x_i - c_k \|} \right)^{2/(m-1)}} \tag{4}$$

When clustering is complete, the fuzzy values in the U membership matrix show the clustering result. When these values are defuzzified, their equivalents can be found as 0 and 1.

3.4. Whale optimization algorithm

Metaheuristic optimization algorithms have been used frequently in engineering applications in recent years. These algorithms have several advantages, such as easy conversion, non-derivative structures, and avoiding the local minima. In addition, they have a wide range of uses, as they can produce solutions to problems in different areas.

The whale optimization algorithm is a metaheuristic optimization algorithm. The algorithm imitates the hunting behavior of humpback whales and it is inspired by the bubble hunting strategy they used during hunting [16].

Humpback whales, which usually feed on small fish flocks, can create air bubble clouds by exhaling under water with their unique air bubble behavior. These large air bubble clouds, which are interconnected, are very effective at gathering prey. Then the whale starts to rise towards the surface in the bubbles formed. As it rises, it also continues to create bubbles, and as it gets closer to its prey, it narrows the bubble circle and shrinks the target. This behavior is useful in finding, immobilizing and taking prey by surprise, as well as making it possible for the hunter to hide from its prey [52].

Figure 3 (a) represents the hunting method with bubble strategy [16]. Figure 3 (b) is a real picture of hunting [16]. In the whale optimization algorithm, the hunting strategy is divided into three parts: encircling prey, bubble-net attacking method, and searching for prey.

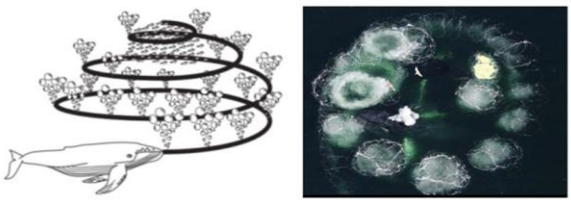


Figure 3. a) Representative hunting picture **b)** Real hunting picture.

3.4.1. Encircling prey

In the whale optimization algorithm, prey is the optimum solution to be reached. Since the optimum solution is not known in optimization problems, best result achieved or a point close to it is considered as the optimum solution. The states of other solutions are updated according to the best solution value determined. The behavior to wrap around prey is shown mathematically in Eq. (5) and (6) [16].

$$\vec{D} = \left| \vec{C} . \overrightarrow{X*}_{(t)} - \vec{X}_{(t)} \right| \tag{5}$$

$$\vec{X}_{(t+1)} = \left| \overrightarrow{X*}_{(t)} - \vec{A} . \vec{D} \right| \tag{6}$$

where t represents current iteration, $\vec{A}$ ve $\vec{C}$ convergence vectors, $\overrightarrow{X^*}$ represents the best solution vector.

$$\vec{A} = 2\,\vec{\alpha} . \vec{r} - \vec{\alpha} \tag{7}$$

$$\vec{C} = 2.\vec{r} \tag{8}$$

In Eq. (7) and Eq. (8), $\vec{r}$ represents a random vector, and $\vec{\alpha}$ represents a decreasing vector that is linear from 2 to 0 during iterations [16].

3.4.2. Bubble-net attacking method

This stage is modeled in two parts as spiral movement and narrowing the circle around the hunt. When the value of α in Eq. (7) is reduced, the circle around the prey also shrinks. Figure 4 shows the spiral motion and the position of the best solution. For the spiral movement shown, the distance between the target position (best solution candidate) and the solution candidate was calculated and Eq. (9) was created [53].

$$\vec{X}_{(t+1)} = \overrightarrow{D'} .\ e^{bl} . \cos\ (2\pi l) + \overrightarrow{X*}_{(t)} \tag{9}$$

The equation is $\overrightarrow{D'} = \overrightarrow{X*}_{(t)} - \vec{X}_{(t)}$ in Eq. (9). This expression gives the distance between the search agent and the best known position. b, refers to logarithmic spiral constant, while l refers to a random number in the range [-1, 1]. Whether the algorithm will select spiral motion or linear motion is determined by probability ½ as shown in Eq. (10). p, represents a random number in range [0, 1].

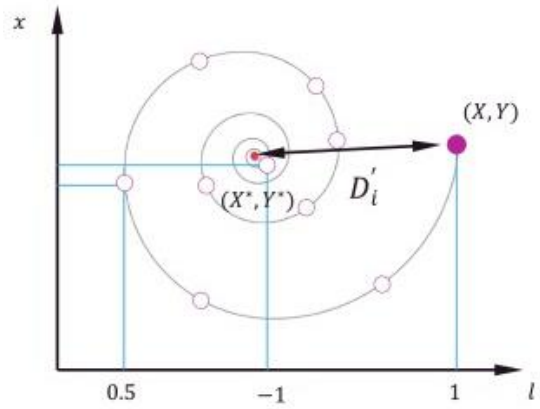


Figure 4. Spiral movement [13]

$$\vec{X}_{(t+1)} = \begin{cases} \overrightarrow{X*}_{(t)} - \vec{A} . \vec{D} , & p < 0{,}5 \\ \overrightarrow{D'} .\ e^{bl} . \cos\ (2\pi l) + \overrightarrow{X*}_{(t)} , & p \geq 0{,}5 \end{cases} \tag{10}$$

3.4.3. Search for prey

The mathematical model of the global solution is shown in Eq. (11) and (12). The new positions of candidate solutions are created around a randomly chosen candidate solution rather than the best known candidate.

$$\overrightarrow{D'} = \overrightarrow{C} . \vec{X}_{rand} - \vec{X} \tag{11}$$

$$\vec{X}_{(t+1)} = \vec{X}_{rand} - \vec{A} . \vec{D} \tag{12}$$

$\vec{X}_{rand}$, represents a randomly chosen solution vector. The value of vector $\vec{A}$ is the determinant for global or local search. A random search agent is chosen when $|\vec{A}|>1$, while the best solution is selected when $|\vec{A}|<1$ for updating the position of the search agents. The global search behavior of the algorithm is shown in Figure 5 and the pseudo code is given in Figure 6.

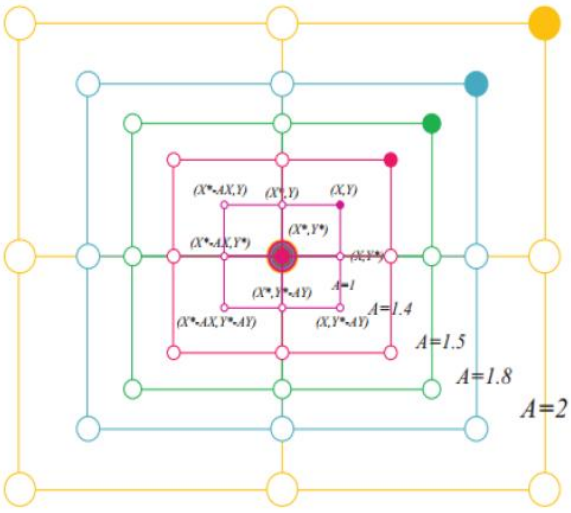


Figure 5. Global search [16].

Set initial population X_i $(i = 1,2,...,n)$
Calculate the fitness value of each search agent
X^*= the best search agent
while (t < maximum number of iterations)
for (each search agent)
Update α, A, C, l, ve p
  if (p < 0.5)
   if (|A| < 1)
    Update the position of the current search agent by Eq. (6)
   else if (|A| ≥ 1)
    Select a random search agent ($\vec{X}_{rand}$)
    Update the position of the current search agent by Eq. (12)
   end if
  else if (p ≥ 0.5)
   Update the position of the current search agent by Eq. (9)
  end if
end for
Check if any search agent goes beyond the search space and amend it
Calculate the fitness value of each search agent
Update X^* if there is a better solution
t = t + 1
end while
return X^*

Figure 6. Pseudo code of the WOA [53].

3.5. Levy flight

Levy flight was introduced in 1937 by the French mathematician Paul Levy. It is a statistical search strategy that was discovered a century or so before him and goes beyond the more traditional Brownian movement. Each of generation starts from the best known position using a series of objects and a new generation is produced at randomly distributed distances. Then, the next generation is evaluated to select the most promising one, and this process is repeated iteratively until certain stopping criterion is met.

In general, animals' foraging behavior is a kind of random movement. Since the next move depends on the current position and the possibility of moving to the next position, every random move made is of great importance. Recent studies show that Levy flight is one of the best search strategies in random motion model [54-56].

When the previous studies are examined, it is seen that Levy flight is used in applications with the modified version as well as the original version. Many researchers have used modified Levy flight techniques such as cut Levy flight, smooth cut Levy flight, and staggered cut Levy flight in optimization operations. In this study, Levy flight technique was used in its original form.

Levy flight [57] is a class of stable non-Gaussian random processes. Random motion is generated by taking advantage of the levy stable distribution. This distribution is actually a simple power law formula. Mathematically, a simple version of the Levy distribution can be defined as in Eq. (13) [31, 58]:

$$L = \begin{cases} \sqrt{\frac{\gamma}{2\pi}} \exp\left[-\frac{\gamma}{2(s-\mu)}\right] \frac{1}{(s-\mu)^{3/2}} & if\ 0 < \mu < s < \infty \\ 0 & if\ s \le 0 \end{cases} \quad (13)$$

μ is the position or displacement parameter and γ> 0 parameter is the scale parameter that controls the distribution scale. In general, the Levy distribution should be defined as the Fourier transform.

$$F(k) = \exp\left[-\alpha|k|^{\beta}\right], 0 < \beta \le 2 \quad (14)$$

α is a parameter in the range [−1, 1] and is known as skewness or scale factor An index of stability $\beta \in (0, 2]$ is also called the Levy index. The analytic form of the integral is unknown for general β, except for a few special cases. While the parameters β and α play a major role in determining the distribution, the γ and μ parameters have a small effect. The β parameter controls the shape of the probability distribution to obtain different shapes of probability distribution, especially in the tail region. Therefore, the smaller parameter β causes the scatter to make longer jumps because it will create a longer tail [59]. The sign of the skewness parameter α indicates the skew direction of the curve. Positive values represent the right direction and negative values represent the left direction. When α = 0, the distribution is symmetrical. The last two parameters, γ width and μ change, are the peaks of the distribution [23]. Different values of the β parameter change the distribution. It makes longer jumps for smaller values, while it makes shorter jumps for larger values.

4. Experimental Application

Datasets with different number of features and number of clusters were selected to compare the performance of the clustering algorithms used in the study. Table 1 contains summary information about the datasets. Algorithms were implemented on an Intel® Core ™ i5-2400 CPU @ 3.1 GHz processor, 4 GB RAM and Windows 7 (64-bit) Professional operating system.

Table 1. Properties of datasets

Dataset	#Clusters	#Attributes	#Data
Dermatology	6	34	366
Cancer-Int	2	9	699
Cancer	2	30	569
Thyroid	3	5	215
Heart	2	13	270
Spect	2	22	267
Diabetes	2	8	768
Hepatitis	2	21	155
Breast Tissue	6	9	106
Parkinson	2	22	195

Since the WOA clustering algorithm has exploration and exploitation capabilities, it can be considered as a global optimizer. The WOA algorithm starts with a series of random solutions and updates the position of search agents according to the above equations at each iteration. The adaptive change of the search vector $\vec{A}$ allows the algorithm to switch easily between exploration and exploitation. This means that some iterations focus on exploration and the rest on exploitation. The exploration ability of the algorithm is due to the position (Eq. (12)) update mechanism used. The purpose of the WOA is to find the search agent that provides the best evaluation of a particular fitness function. In the study, the WOA fitness function value calculated during the iterations was compared with the Levy flight fitness function value. If the SSE value of the position found is better after applying the Levy flight strategy, the WOA position has been updated. Thus, it was tried to achieve a better clustering result by increasing the efficiency of global search. In all algorithms, the number of iterations was selected as 1000 and the population size as 100. K-means, k-medoids and fuzzy c-means algorithms have only the maximum number of iterations as parameter. However, the WOA algorithm includes a number of parameters. The α parameter starts with a value of 2 and decreases linearly towards 0. The parameters r_1 and r_2 are randomly generated in the range [0, 1].

5. Experimental results

In application, the SSE value given in Eq. (1) is used as a fitness function for all algorithms. The purpose of the algorithms is to find the best cluster centers that minimize the SSE value. The results of the 30 runs for the algorithms are given in Table 2 while B is best, W is worst, A is average and S is standard deviation. Considering the average values, WOA-LF had the best result on eight datasets (cancer-int, thyroid, heart, spect, diabetes, hepatitis, breast tissue, parkinson). While in dermatology dataset k-means algorithm has found the best result, three algorithms (WOA-LF, WOA, k-means) have also gained the best result in cancer dataset. K-medoids and fuzzy c-means algorithms did not achieve better results in any dataset compared to other algorithms. As a result, it is clearly seen that the clustering results obtained in the original WOA algorithm are improved in the WOA-LF hybrid method. The results show that the WOA generally has better results than other three algorithms and by using the Levy flight strategy the performance of the WOA has been improved. In this case, it is possible to clearly say that the Levy flight strategy causes positive affects for the clustering problem on the WOA algorithm.

Table 3 shows the average results of the SSE value of the 30 runs for each algorithm and the average success ranking on datasets based on these values. Since there are five algorithms, the algorithms have a rank value between 1 and 5. According to the average rankings, WOA-LF has the best rank value with 1.1. The original WOA, k-means, k-medoids, fuzzy c-means algorithms have respectively obtained ranking values as 2.0, 2.6, 4.0 and 4.3. Here it has been clearly seen the positive affect of the LF on the WOA.

6. Conclusion

In this study, clustering was applied by using Whale Optimization Algorithm (WOA), which based on the whales' hunting strategy, and Levy flight (LF) search strategy. The Levy flight strategy has been used to improve the global search of WOA algorithm and enhance clustering results. The performance of WOA-LF has been examined comparatively with three basic clustering algorithms (k-means, k-medoids, fuzzy c-means) and original WOA. Compared to these algorithms, WOA-LF has been shown to produce better results overall. WOA-LF can be used as an alternative clustering algorithm due to its success in clustering problem.

Comparing the WOA-LF clustering algorithm to hybrid clustering approaches or using other fitness functions during clustering may be the subject of future research.On the other hand, the algorithm can be improved and apply on different optimization problems.

Table 2. The results of the algorithms for clustering problem

		K-means	**K-medoids**	**F.C-means**	**WOA**	**WOA-LF**
Dermatology	B	2.03E+03	2.83E+03	5.20E+03	2.48E+03	2.41E+03
	W	2.10E+03	3.07E+03	5.20E+03	2.66E+03	2.55E+03
	A	**2.06E+03**	2.96E+03	5.20E+03	2.58E+03	2.49E+03
	S	3.56E+01	8.74E+01	4.42E-11	4.77E+01	3.95E+01
Cancer-Int	B	2.99E+03	3.46E+03	3.29E+03	2.96E+03	2.96E+03
	W	2.99E+03	4.46E+03	3.29E+03	2.97E+03	2.96E+03
	A	2.99E+03	3.79E+03	3.29E+03	2.97E+03	**2.96E+03**
	S	8.03E-01	4.33E+02	3.94E-13	7.17E-01	8.75E-02
Cancer	B	1.34E+154	1.79e+308	7.62E+156	1.34E+154	1.34E+154
	W	1.34E+154	1.79e+308	7.62E+156	1.34E+154	1.34E+154
	A	**1.34E+154**	1.79e+308	7.62E+156	**1.34E+154**	**1.34E+154**
	S	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Thyroid	B	2.00E+03	2.14E+03	2.81E+03	1.89E+03	1.89E+03
	W	2.02E+03	2.40E+03	2.81E+03	2.07E+03	1.99E+03
	A	2.01E+03	2.24E+03	2.81E+03	1.97E+03	**1.92E+03**
	S	6.89E+00	1.02E+02	7.20E-11	4.39E+01	2.10E+01
Heart	B	1.07E+04	1.16E+04	1.39E+04	1.06E+04	1.06E+04
	W	1.07E+04	1.39E+04	1.39E+04	1.07E+04	1.07E+04
	A	1.07E+04	1.26E+04	1.39E+04	1.07E+04	**1.07E+04**
	S	0.00E+00	9.79E+02	2.34E-11	1.94E+01	2.25E+01
Spect	B	5.58E+02	6.34E+02	5.58E+02	5.55E+02	5.55E+02
	W	5.58E+02	6.34E+02	5.58E+02	5.58E+02	5.55E+02
	A	5.58E+02	6.34E+02	5.58E+02	5.55E+02	**5.55E+02**
	S	0.00E+00	0.00E+00	0.00E+00	6.71E-01	1.24E-05
Diabetes	B	7.46E+04	7.32E+04	7.46E+04	7.21E+04	7.21E+04
	W	7.46E+04	7.38E+04	7.46E+04	7.42E+04	7.21E+04
	A	7.46E+04	7.34E+04	7.46E+04	7.22E+04	**7.21E+04**
	S	0.00E+00	3.02E+02	0.00E+00	3.75E+02	4.92E+00
Hepatitis	B	9.97E+03	1.04E+04	1.28E+04	9.44E+03	9.44E+03
	W	1.14E+04	1.23E+04	1.28E+04	1.02E+04	9.48E+03
	A	1.09E+04	1.09E+04	1.28E+04	9.50E+03	**9.46E+03**
	S	6.99E+02	8.12E+02	1.20E-11	1.81E+02	1.33E+01
Breast Tissue	B	1.31E+05	1.73E+05	1.46E+05	1.26E+05	1.25E+05
	W	1.90E+05	3.94E+05	1.46E+05	1.35E+05	1.36E+05
	A	1.44E+05	2.75E+05	1.46E+05	1.30E+05	**1.28E+05**
	S	2.62E+04	7.99E+04	4.37E-11	2.47E+03	2.23E+03
Parkinson	B	1.71E+04	1.71E+04	1.71E+04	1.65E+04	1.65E+04
	W	1.71E+04	1.71E+04	1.71E+04	1.65E+04	1.65E+04
	A	1.71E+04	1.71E+04	1.71E+04	1.65E+04	**1.65E+04**
	S	0.00E+00	0.00E+00	0.00E+00	5.47E-01	5.49E-01

Table 3. Average ranking values of the algorithms

	K-means	K-medoids	F.C-means	WOA	WOA-LF
Dermatology	**2.06E+03**	2.96E+03	5.20E+03	2.58E+03	2.49E+03
	1	4	5	3	2
Cancer-Int	2.99E+03	3.79E+03	3.29E+03	2.97E+03	**2.96E+03**
	1	5	4	2	1
Cancer	**1.34E+154**	1.79e+308	7.62E+156	**1.34E+154**	**1.34E+154**
	1	4	5	1	1
Thyroid	2.01E+03	2.24E+03	2.81E+03	1.97E+03	**1.92E+03**
	3	4	5	2	1
Heart	1.07E+04	1.26E+04	1.39E+04	1.07E+04	**1.07E+04**
	3	4	5	2	1
Spect	5.58E+02	6.34E+02	5.58E+02	5.55E+02	**5.55E+02**
	3	4	3	2	1
Diabetes	7.46E+04	7.34E+04	7.46E+04	7.22E+04	**7.21E+04**
	4	3	4	2	1
Hepatit	1.09E+04	1.09E+04	1.28E+04	9.50E+03	**9.46E+03**
	4	3	5	2	1
Breast Tissue	1.44E+05	2.75E+05	1.46E+05	1.30E+05	**1.28E+05**
	3	5	4	2	1
Parkinson	1.71E+04	1.71E+04	1.71E+04	1.65E+04	**1.65E+04**
	3	4	3	2	1
Avg. Rank	2.6	4	4.3	2	**1.1**

References

[1] Evirgen, F. (2016). Analyze the optimal solutions of optimization problems by means of fractional gradient based system using VIM. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, 6(2), 75-83.

[2] Evirgen, F. (2017). Conformable fractional gradient based dynamic system for constrained optimization problem. *Special issue of the 3rd International Conference on Computational and Experimental Science and Engineering (ICCESEN 2016)*,1066-1069.

[3] Evirgen, F. and Yavuz, M. (2018). An alternative approach for nonlinear optimization problem with Caputo-Fabrizio derivative. *ITM Web of Conferences*,01009.

[4] Cui, D. (2017). Application of whale optimization algorithm in reservoir optimal operation. *Advances in Science and Technology of Water Resources*, 37(3), 72-79.

[5] Zhang, C., Ouyang, D., and Ning, J. (2010). An artificial bee colony approach for clustering. *Expert systems with applications*, 37(7), 4761-4767.

[6] Selim, S. Z. and Alsultan, K. (1991). A simulated annealing algorithm for the clustering problem. *Pattern recognition*, 24(10), 1003-1008.

[7] Maulik, U. and Mukhopadhyay, A. (2010). Simulated annealing based automatic fuzzy clustering combined with ANN classification for analyzing microarray data. *Computers & operations research*, 37(8), 1369-1380.

[8] Maulik, U. and Bandyopadhyay, S. (2000). Genetic algorithm-based clustering technique. *Pattern recognition*, 33(9), 1455-1465.

[9] Shelokar, P., Jayaraman, V. K., and Kulkarni, B. D. (2004). An ant colony approach for clustering. *Analytica Chimica Acta*, 509(2), 187-195.

[10] Van der Merwe, D. and Engelbrecht, A. P. (2003). Data clustering using particle swarm optimization. *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*,215-220.

[11] Cura, T. (2012). A particle swarm optimization approach to clustering. *Expert Systems with Applications*, 39(1), 1582-1588.

[12] Karaboga, D. and Ozturk, C. (2011). A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Applied soft computing*, 11(1), 652-657.

[13] Armano, G. and Farmani, M. R. (2014). Clustering analysis with combination of artificial bee colony algorithm and k-means technique. *International Journal of Computer Theory and Engineering*, 6, 141-145.

[14] Karthikeyan, S. and Christopher, T. (2014). A hybrid clustering approach using artificial bee colony (ABC) and particle swarm optimization. *International Journal of Computer Applications*, 100(15).

[15] Mane, S. U. and Gaikwad, P. G. (2014). Hybrid particle swarm optimization (HPSO) for data clustering. *International Journal of Computer Applications*, 97(19).

[16] Mirjalili, S. and Lewis, A. (2016). The whale optimization algorithm. *Advances in engineering software*, 95, 51-67.

[17] Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95-international conference on neural networks*,1942-1948.

[18] Storn, R. and Price, K. (1997). Differential evolution–a simple and efficient heuristic for global

optimization over continuous spaces. *Journal of global optimization*, 11(4), 341-359.
[19] Rashedi, E., Nezamabadi-Pour, H., and Saryazdi, S. (2009). GSA: a gravitational search algorithm. *Information sciences*, 179(13), 2232-2248.
[20] Yao, X., Liu, Y., and Lin, G. (1999). Evolutionary programming made faster. *IEEE Transactions on Evolutionary computation*, 3(2), 82-102.
[21] Nasiri, J. and Khiyabani, F. M. (2018). A whale optimization algorithm (WOA) approach for clustering. *Cogent Mathematics & Statistics*, 5(1), 1483565.
[22] Canayaz, M. and Özdağ, R. (2017). Data clustering based on the whale optimization. *Middle East Journal of Technic*, 2(2), 178-187.
[23] Al-Temeemy, A. A., Spencer, J., and Ralph, J. (2010). Levy flights for improved ladar scanning. *2010 IEEE International Conference on Imaging Systems and Techniques*,225-228.
[24] Chen, Y. (2010). Research and simulation on Levy flight model for DTN. *2010 3rd International Congress on Image and Signal Processing*,4421-4423.
[25] Pereyra, M. A. and Batatia, H. (2010). A Levy flight model for ultrasound in skin tissues. *2010 IEEE International Ultrasonics Symposium*,2327-2331.
[26] Terdik, G. and Gyires, T. (2008). Lévy flights and fractal modeling of internet traffic. *IEEE/ACM Transactions on Networking*, 17(1), 120-129.
[27] Sutantyo, D. K., Kernbach, S., Levi, P., and Nepomnyashchikh, V. A. (2010). Multi-robot searching algorithm using Lévy flight and artificial potential field. *2010 IEEE Safety Security and Rescue Robotics*,1-6.
[28] Rhee, I., Shin, M., Hong, S., Lee, K., Kim, S. J., and Chong, S. (2011). On the levy-walk nature of human mobility. *IEEE/ACM transactions on networking*, 19(3), 630-643.
[29] Edwards, A. M., Phillips, R. A., Watkins, N. W., Freeman, M. P., Murphy, E. J., Afanasyev, V., *et al.* (2007). Revisiting Lévy flight search patterns of wandering albatrosses, bumblebees and deer. *Nature*, 449(7165), 1044-1048.
[30] Viswanathan, G. M., Afanasyev, V., Buldyrev, S., Murphy, E., Prince, P., and Stanley, H. E. (1996). Lévy flight search patterns of wandering albatrosses. *Nature*, 381(6581), 413-415.
[31] Yang, X.-S. and Deb, S. (2013). Multiobjective cuckoo search for design optimization. *Computers & Operations Research*, 40(6), 1616-1624.
[32] Yang, X.-S. (2010). Firefly algorithm, Levy flights and global optimization. *Research and development in intelligent systems XXVI*. Springer, 209-218.
[33] Murphy, P. and Aha, D. (1994). *UCI repository of machine learning*. University of California, Department of Information and Computer Science.
[34] Ozdamar, K. (2002). *Paket programlari ile istatistiksel veri analizi-1*. Kaan Kitabevi, Eskisehir.
[35] Tatlıdil, H. (1996). *Uygulamalı Çok Değişkenli İstatistiksel Analiz*. Cem Ofset Ltd. Şti, Ankara.
[36] Hair Jr, J., Anderson, R., and Tatham, R. (1998). *Multivariate data analysis*.NJ: PrenticeYHall Inc, Upper Saddle River.
[37] Lorr, M. (1983). *Cluster analysis for social scientists*.Jossey-Bass Incorporated Pub.
[38] Boushaki, S. I., Kamel, N., and Bendjeghaba, O. (2018). A new quantum chaotic cuckoo search algorithm for data clustering. *Expert Systems with Applications*, 96, 358-372.
[39] Frigui, H. and Krishnapuram, R. (1999). A robust competitive clustering algorithm with applications in computer vision. *Ieee transactions on pattern analysis and machine intelligence*, 21(5), 450-465.
[40] Karakoyun, M. (2015). *Kurbağa sıçrama algoritmasının kümeleme problemlerine uygulanması*. Master Thesis. Selçuk University.
[41] Sharma, S. (1996). Applied multivariate techniques.
[42] Tabak, J. (2014). *Geometry: the language of space and form*. Infobase Publishing.
[43] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*,281-297.
[44] Han, J. and Kamber, M. (2010). Data mining: concepts and techniques. [Nachdr.], *Amsterdam: Elsevier/Morgan Kaufmann*, 11, 6.
[45] Tan, P.-N., Steinbach, M., and Kumar, V. (2006). Classification: basic concepts, decision trees, and model evaluation. *Introduction to data mining*, 1, 145-205.
[46] Xu, R. and Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3), 645-678.
[47] Dodge, Y. (2012). *Statistical data analysis based on the L1-norm and related methods*. Birkhäuser.
[48] Dinçer, Ş. E. (2006). *Veri madenciliğinde K-means algoritması ve tıp alanında uygulanması*. Master Thesis. Kocaeli University.
[49] Karakoyun, M., Saglam, A., Baykan, N. A., and Altun, A. A. (2017). Non-locally color image segmentation for remote sensing images in different color spaces by using data-clustering methods. *5th International Conference on Advanced Technology & Sciences (ICAT'17)*, 6-12.
[50] Höppner, F., Klawonn, F., Kruse, R., and Runkler, T. (1999). *Fuzzy cluster analysis: methods for classification, data analysis and image recognition*. John Wiley & Sons.
[51] Moertini, V. (2002). Introduction to Five DataClustering Algorithms Clustering Algorithm. *Integral*, 7(2).
[52] Goldbogen, J. A., Friedlaender, A. S., Calambokidis, J., Mckenna, M. F., Simon, M., and Nowacek, D. P. (2013). Integrative approaches to the study of baleen whale diving behavior, feeding performance, and foraging ecology. *BioScience*, 63(2), 90-100.
[53] Tanyıldızı, E. and Cigalı, T. (2017). Kaotik Haritalı Balina Optimizasyon Algoritmaları. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 29(1), 307-317.
[54] Pavlyukevich, I. (2007). Lévy flights, non-local search and simulated annealing. *Journal of Computational Physics*, 226(2), 1830-1844.
[55] Reynolds, A. M. and Frye, M. A. (2007). Free-flight odor tracking in Drosophila is consistent with an optimal intermittent scale-free search. *PloS one*, 2(4), e354.

[56] Shlesinger, M. F. (2006). Search research. *Nature*, 443(7109), 281-282.
[57] Chechkin, A. V., Metzler, R., Klafter, J., and Gonchar, V. Y. (2008). Introduction to the theory of Lévy flights. *Anomalous transport*, 1, 129.
[58] Yang, X.-S. (2010). *Engineering optimization: an introduction with metaheuristic applications*. John Wiley & Sons.
[59] Lee, C.-Y. and Yao, X. (2001). Evolutionary algorithms with adaptive lévy mutations. *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, 568-575.

Ayşe Nagehan Mat *received the undergraduate degree from the Department of Computer Engineering of Selcuk University and still working for Master's degree. She is working as an International Officer at Selcuk University.*
https://orcid.org/0000-0003-4975-6418

Onur İnan *received the Ph.D. degree from the Department of Computer Engineering of Selcuk University. He is working as a Doctor lecturer at the Computer Engineering Department of Necmettin Erbakan University.*
https://orcid.org/0000-0003-4573-7025

Murat Karakoyun *received the Ph.D. degree from the Department of Computer Engineering of Konya Technical University. He is working as a Research Assistant at Computer Engineering Department of Necmettin Erbakan University.*
https://orcid.org/0000-0002-0677-9313

An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.11, No.2, pp.227-237 (2021)
http://doi.org/10.11121/ijocta.01.2021.001073

RESEARCH ARTICLE

Novel stability and passivity analysis for three types of nonlinear LRC circuits

Muzaffer Ateş[a*] and Nezir Kadah[b]

[a] *Departments of Electrical-Electronics Engineering, University of Van Yuzuncu Yil,Turkey*
[b] *Departments of Electrical-Electronics Engineering, Adana Alparslan Turkes Science and Technology University, Turkey*
mates@yyu.edu.tr, nkadah@atu.edu.tr

ARTICLE INFO

Article History:
Received 08 January 2021
Accepted 09 May 2021
Available 31 July 2021

Keywords:
Lyapunov stability
Nonlinear systems
Nonlinear LRC circuits
Passivity
Gronwall's inequality

AMS Classification 2010:
34D23; 34D20; 34C23

ABSTRACT

In this paper, the global asymptotic stability and strong passivity of three types of nonlinear *LRC* circuits are investigated by utilizing the Lyapunov's direct method. The stability conditions are obtained by constructing appropriate energy (or Lyapunov) function, which demonstrates the practical application of the Lyapunov theory with a clear perspective. Many specialists construct Lyapunov functions by using some properties of the functions with much trial and errors or for a system they choose candidate Lyapunov functions. So, for a given system the Lyapunov function is not unique. But we insist that the Lyapunov (energy) function is unique for a given physical system. Thus, this study clarifies Lyapunov stability with suitable tools and also improves some previous studies. Our approach is constructing energy function for a given nonlinear system that based on the power-energy relationship of the system. Hence for a dynamical system, the derivative of the Lyapunov function is equal to the negative value of the dissipative power in the system. These aspects have not been addressed in the literature. This paper is an attempt towards filling this gap. The provided results are central importance for the stability analysis of nonlinear systems. Some simulation results are also given successfully that verify the theoretical predictions.

1. Introduction

In history, modeling and stability analysis of nonlinear systems are the most important and popular problems in control theory. Since almost all systems are nonlinear in nature [1], a number of promising studies have been analyzed in the literature. Many researchers as Lagrange, Hamilton, Poincare and Lyapunov are focused on the modelling problem to analyze the dynamic behavior of systems [1, 2]. The aforementioned methods are based on the energy utilization of the related systems. However, since all systems are not in linear forms, certain mathematical solutions are not available to solve these issues. Furthermore, *closed-form expressions* for the solutions of the linear systems are not possible to solve nonlinear systems. Nevertheless, it is important to be able to make some assumptions about the conduct of a nonlinear system called *qualitative analysis*.

The stability of the equilibrium point was first examined by Lagrange; however, the Lagrange principle was only suitable for the Lagrange systems (conservative systems) [1], but engineering systems usually have damping [3]. Then, the stability theory of motion derived from the concepts of Lagrange's principle and Poincare's regular solution (Lyapunov stable motion) was developed by Lyapunov [2]. Hamiltonian and Lagrangian systems comply with conservative systems (exact differential equations), but Lyapunov stability theory can be applicable to arbitrary differential equations. Thus, the Lyapunov direct method is

*Corresponding Author

the most common and efficient tool for stability analysis [4–7]. In this context, many elegant studies on the qualitative behavior of systems can be recorded in the literature. Most experiments are carried out on the axiomatization of the stability principle. The problem of stability of the solar system attracted a great deal of early interest. Then, Lyapunov used his second (direct) method that there is no need to solve the differential equations explicitly to investigate the stability of the given systems. The Lyapunov's direct method is still recognized as an effective tool to study the stability theory of dynamical systems such as: the global asymptotic stability of the electrical RLC circuit [8], neural networks with time varying delays [9,10], power systems analysis [11], robot manipulators [12], dissipativity analysis of discrete-time neural networks [13], global robust passivity analysis [14], dissipativity and passivity analysis of neural networks [15]. This method is the best way to determine the asymptotic stability or asymptotic controllability of nonlinear systems. The central notion is that the energy of the system diminishes along suitably chosen paths, such that the system attains a minimal energy configuration at the invariant equilibrium. Here, this result has been presented both mathematically and through simulation.

Lyapunov theory is based on the Torricelli principle [16]. Therefore, the storage energy of the dynamic system decreases over time along the trajectories of the system. So, the direct method provides the opportunity to examine the stability of the equilibrium points with minimum energy. This meaning (diminishing of energy) tends us to the passivity of the systems. Passivity, which is the basic feature of the dynamic systems theory [17–20], can now be debated. LRC circuits, viscoelastic systems and thermodynamic systems are typical examples of dissipative systems with the external sources. The terminology of dissipativity is a generalization of the concept of passivity [21]. Apparently, a dissipative system is not a conservative system. The main point of passivity theory is that the systems are internally stable [22, 23]. Storage functions are bounded [21], and this result has been proved mathematically in the proof of Theorem 4. Thus, some new passivity results with Gronwall's inequality [3] can be shown to define the strict passivity or boundedness of the systems involved.

Natural (real) energy functions of the dynamic systems empower the Lyapunov's direct process implementations more than the Lyapunov candidate functions. Thus, each energy function used in this study is constructed from the physical meaning of the given system and its time derivative (directional) is equal to the negative value of the dissipated power in the system. For example, for any unforced dissipative system, the time derivative of the energy (Lyapunov) function $E(t)$ along the system orbits gives

$$V'(t) = -\sum_{i=1}^{n} R_i I_i^2, \tag{1}$$

where R_i is the damping term (or resistance) and I_i is the velocity (or current) of the *ith* component of the system.

The above arguments are not clear in the related literature. Hence, many specialists chose candidate Lyapunov functions or consider some Lyapunov functions or construct Lyapunov functions with much trial and error for their systems [1, 11, 24, 25] without any physical meaning. Generally, these tools make Lyapunov stability very complex (see [24, 25]). Because, still there is an idea in the literature, constructing Lyapunov functions for nonlinear systems is a difficult task [26, 27]. But, for the first and second order ordinary differential equations we highly simplified Lyapunov stability theory with LRC circuit systems. Hence, the interested knows how to construct the energy (Lyapunov) function and checks the result of the time derivative (directional) of the energy function with (1). This approach also improves some well-known studies. These improvements will take place in section 4. In addition, [6] does not involve the passivity analysis, the notion of power –energy relationship constructing Lyapunov functions, and equation (1) and its implications. The present work includes these and some improvements.

The rest of this paper is organized as follows. Section 2 presents some definitions and auxiliary results. Section 3 deals with the main results. Section 4 deals with discussion. Section 5 closes the paper with a short conclusion.

2. Preliminaries

A commonly used model for an autonomous nonlinear system is

$$x'(t) = f(x(t), u(t)), \quad x(0) = x_0, \forall t \geq 0, \tag{2}$$

where $t \in \Re_+ (\Re_+ = [0, \infty))$ denotes time, $x \in \Re^n$ denotes the state of the system, while $u \in \Re^m$ is called the input or the control function. However,

$f : \Re^n \times \Re^m \to \Re^n$ satisfies Lipschitz condition. The state vector $x(t) \in D$ in which $D \subseteqq \Re^n$ is a domain that contains the origin $x = 0$. We assume that (2) is well posed, that is, there exists a unique solution $x : [0, \infty) \to \Re^n$ for every initial data $x(0) = x_0 \in \Re^n$, and x depends continuously on x_0 according to the normed topology on $\Re^n$. Let $f(0, 0) = 0$, $f(x, 0) \neq 0$ for $x \neq 0$, and $||.||$ is the Euclidean norm on $\Re^n$. Further, assume that u is an admissible real valued input function so that

$$\sum_{i=1}^{m} \int_0^t u_i(t)dt \leq K < \infty, \quad \forall t > 0, \qquad (3)$$

where K is a positive constant. A state $\overline{x} \in \Re^n$ is an equilibrium of (2) if $f(\overline{x}, 0) = 0$. A system or machine attains its minimum of energy at the equilibrium points.

We shall now need some basic definitions on the properties of the Lyapunov functions.

Definition 1. *([1]) A function $\alpha(\Re^+, \Re^+)$ is of class κ if it is continuous on $[0, \infty)$, monotonically increasing, and $\alpha(0) = 0$. A class κ function $\alpha(r)$ belongs to class κ_∞ if $\alpha(r) \to \infty$ as $r \to \infty$.*

Definition 2. *([1]) A function $E(x) \in C^1(\Re^+ \times \Re^n, \Re^+)$ is said to be positive definite, decrescent and radially unbounded function if there exist functions α and β of class κ are such that*

(i) $\alpha(||x||) \leq E(x) \leq \beta(||x||), \quad \forall x \in \Re^n$,

(ii) $E'(x(t)) \leq 0$,

(iii) $\alpha(||x||) \to \infty$ as $||x|| \to \infty$,

(iv) Furthermore, assume that the set $S = \{x \in \Re^n : E'(x) = 0\}$, contains no invariant set other than the set $\{0\}$.

Now, to motivate the definitions of passivity we can use electrical circuits. Inflow power of a simple resistive system is always nonnegative with the voltage $u(t)$ as input and the current $y(t)$ as output, that is, if $uy \geq 0$ for all $u, y \in \Re$, and for a multiport network we have $u^T y = \sum_{i=1}^m u_i y_i \geq 0$. Therefore, for general nonlinear systems we can state some new passivity properties. These new properties are of interest in circuit and control theory [24] and has applications in mathematical control theory with Gronwall's inequality [3].

Definition 3. *([24]) System (2) is passive if there exists a positive definite function $E(t)$ is such that*

$$(i) \quad E'(t) \leq r(u, y).$$

Moreover, it is lossless if

$$(ii) \quad E'(t) = r(u, y),$$

and strictly passive if

$$(iii) \quad E'(t) + \Psi(x, y) \leq r(u, y),$$

for some positive definite function Ψ, where $y = x'$, and $r(t) = r(u(t), y(t)) = \sum_{i=1}^m u_i y_i$ is the supply rate function of (2) defined on $\Re^m \times \Re^n$, and satisfies

$$\int_0^t |r(s)|ds < \infty,$$

for all $t \geq 0$, with $r(0, y) = 0$.

Lemma 1. *If system (1) is passive with an energy-like function E, then the origin of $x' = f(x, 0)$ is stable.*

Proof. See [24]. □

3. Main results

The matter under discussion is the stability of the origin $(0, 0)$ and the passivity of the following nonlinear resistive, inductive, and capacitive *LRC* circuits for one input variable ($m = 1$) and two state variables ($n = 2$). In this work, the inputs of the systems (circuits) are bounded and admissible continuous functions in t. In addition, a nonlinear resistor can be both current controlled *(R(i))* and voltage controlled *(R(v))* element. A nonlinear inductor is current controlled *(L(i))* element while a nonlinear capacitor is said to be voltage controlled *(C(v))* element. The internal resistance of a current source is infinite while that of a voltage source is finite and especially chosen to be small.

3.1. Nonlinear resistor

In the following circuit [28] (Figure 1) there is a *nonlinear resistive element* which is specified by $i_2 = f(v_2)$ and the remaining components R_1, R_2, L, C are positive scalars.

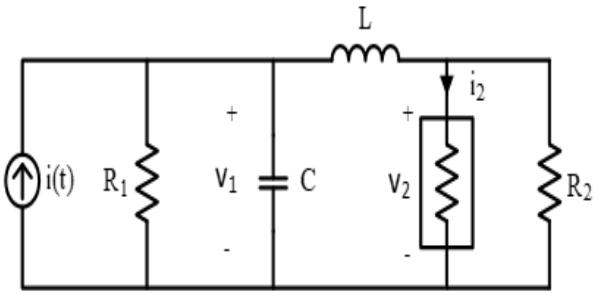


Figure 1. *LRC* circuit with nonlinear resistive element.

Theorem 1. *The nonlinear element of the time-invariant circuit shown in Figure 1 is specified by the relation* $i_2 = f(v_2)$. v_1 *and* v_2 *are the state variables of the circuit. Then, the solution* $v(t) = 0$ *to the system.*

$$\begin{cases} v_1' = \frac{1}{C}[i(t) - \frac{v_1}{R_1} - f(v_2) - \frac{v_2}{R_2}], \\ \\ v_2' = [f'(v_2) + \frac{1}{R_2}]^{-1}[\frac{v_1 - v_2}{L}] \end{cases} \tag{4}$$

with $i(t) = 0$*, is globally asymptotically stable and the circuit will be lossless at infinity if*

(i) $v_1 > v_2$,
(ii) $v_2 f(v_2) \geq 0$,
(iii) $f(0) = 0$.

Proof. First, let write down the state equations of the above circuit:

$$\begin{cases} i(t) = \frac{v_1}{R_1} + Cv_1' + f(v_2) + \frac{v_2}{R_2}, \\ \\ v_1 - v_2 = L\frac{d}{dt}[f(v_2) + \frac{v_2}{R_2}] = L[f'(v_2)v_2' + \frac{v_2'}{R_2}]. \end{cases}$$

Then after some arrangement we obtain system (4). The natural energy function $E_1(t) = E_1(v_1, v_2)$ from the storage elements (capacitor and inductor) of this circuit is

$$E_1(t) = \frac{1}{2}Cv_1^2 + \frac{1}{2}L[f(v_2) + \frac{v_2}{R_2}]^2.$$

The energy function ($E_1 : \Re^2 \to \Re^+$) satisfies

(i) $E_1(0) = 0$,
(ii) $E_1(v) > 0, \quad \forall v \in \Re^2 - \{0\}$.

E_1 is confirmed by the hypothesis (i) of Definition 2. Thus, E_1 is a positive definite function. Then, we write

$$E_1(t) \geq \frac{1}{2}Cv_1^2 \equiv \alpha(||v_1||). \tag{5}$$

The derivative of the Lyapunov function along the trajectories of system (4) gives

$$E_1'(t) = Cv_1v_1' + L[f(v_2) + \frac{v_2}{R_2}][f'(v_2) + \frac{1}{R_2}]v_2',$$

By using system (4), we have

$$E_1'(t) = i(t)v_1 - \frac{v_1^2}{R_1} - f(v_2)v_1 - \frac{v_1v_2}{R_2} + [f(v_2) + \frac{v_2}{R_2}](v_1 - v_2),$$

$$E_1'(t) = i(t)v_1 - \frac{v_1^2}{R_1} - f(v_2)v_1 - \frac{v_1v_2}{R_2} + f(v_2)v_1 + \frac{v_1v_2}{R_2} - f(v_2)v_2 - \frac{v_2^2}{R_2},$$

$$E_1'(t) = i(t)v_1 - \frac{v_1^2}{R_1} - f(v_2)v_2 - \frac{v_2^2}{R_2}.$$

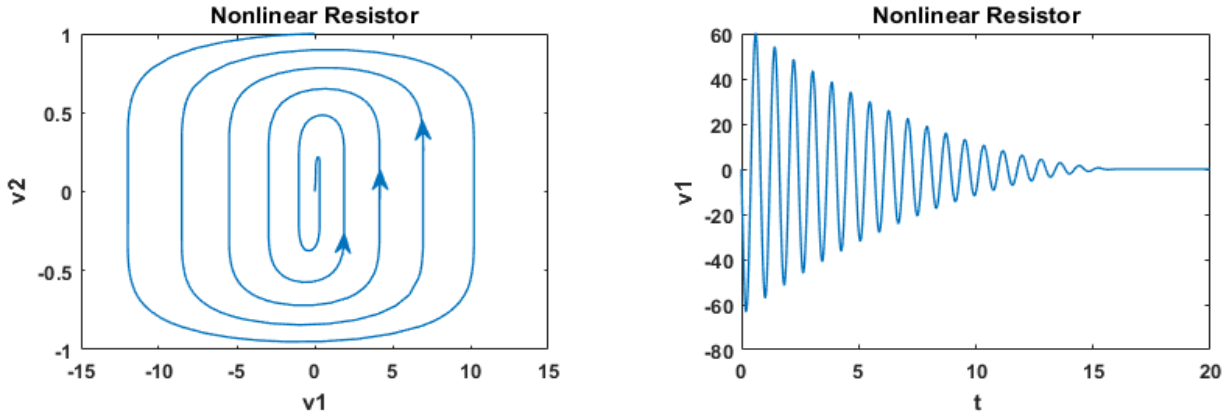
For $i(t) = 0$, it follows that

$$E_1'(t) = -\frac{v_1^2}{R_1} + \frac{v_2^2}{R_2} - v_2f(v_2) = -R_1I_{R_1}^2 - R_2I_{R_2}^2 - R_{NL}i_2^2,$$

where R_{NL} represents the resistance value of the nonlinear element. E_1' is verified by (1). The application of Theorem 1 shows that: $E_1' \leq 0$ on $\Re^2$, $E_1(\infty) = 0$ and $E_1(v) \to \infty$ as $||v|| \to \infty$. Hence, all the motions of (4) are bounded (as illustrated in Figure 2a, 2b). The set S where $E_1' = 0$ is $\{0, 0\}$. This implies that $\{0, 0\}$ is the only invariant subset of S, and the zero solution or equilibrium solution of (4) is globally asymptotically stable. It can be seen that (4) is a lossless system at infinity due to $v(t) = 0$. Hence, the system is zero-state observable. □

3.2. Nonlinear inductor

The circuit shown in Figure 3 [24] contains a *nonlinear inductive element* and is driven by a time-dependent current source $i_s(t)$. Suppose the nonlinear inductor is a Josephson junction described by $i_L = a\phi(t) + b\phi^3(t)$, where ϕ is the magnetic flux of the inductor, $a, b > 0$ are positive constants. The remaining elements R and C are linear and have positive real values.



(a) Phase plane plot of system (4) with $L = \frac{1}{15}H$ **(b)** Solution of system (4) with $L = \frac{4}{3}H$
$[R_1 = R_2 = 1k\Omega,\ C = 10mF,\ f(v_2) = 5v_2^3]$

Figure 2. The motions of system (4).

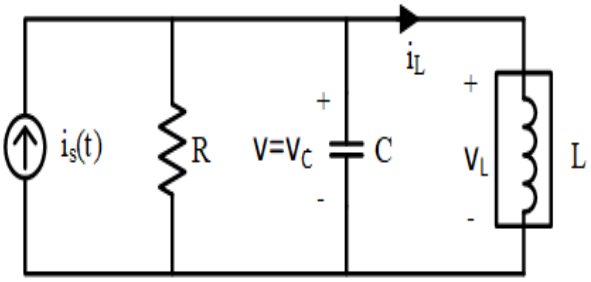


Figure 3. *LRC* circuit with nonlinear inductive element.

Theorem 2. *Let ϕ and $v(= v_L)$ be the state variables of the above circuit (Figure 3) with $i_L = a\phi(t) + b\phi^3(t)$. Then, the solution $(\phi(t), v(t)) = (0, 0)$ to the system*

$$\begin{cases} \phi' = \frac{v}{L(a+3b\phi^2)}, \\ v' = \frac{1}{C}[i_s(t) - \frac{v}{R} - a\phi - b\phi^3] \end{cases} \tag{6}$$

with $i_s(t) = 0$ is globally asymptotically stable or the circuit is lossless at infinity.

Proof. First, let write down the node and v equations of the above circuit:

$$\begin{cases} v = L\frac{d}{dt}(a\phi + b\phi^3), \\ i_s(t) = \frac{v}{R} + Cv' + a\phi + b\phi^3 \end{cases}$$

Then after some rearrangement we obtain system (6).
The native energy function for this circuit is

$$E_2(t) = E_2(\phi, v) = \frac{1}{2}L(a\phi + b\phi^3)^2 + \frac{1}{2}Cv^2.$$

The energy function $(E_2 : \Re^2 \to \Re^+)$ satisfies

(i) $E_2(0) = 0$,
(ii) $E_2(\phi, v) > 0, \forall(\phi, v) \in \Re^2 - \{\sqrt{\phi^2 + v^2} \neq 0\}$.

E_2 is confirmed by the hypothesis (i) of Definition 2. Thus, E_2 is a positive definite function. Then, we write

$$E_2(t) \geq \frac{1}{2}Cv^2 \equiv a(||v||). \tag{7}$$

The derivative of the energy function E_2 along the trajectories of system (6) gives

$$E_2'(t) = L[a\phi + b\phi^3][a + 3b\phi^2]\phi' + Cvv'.$$

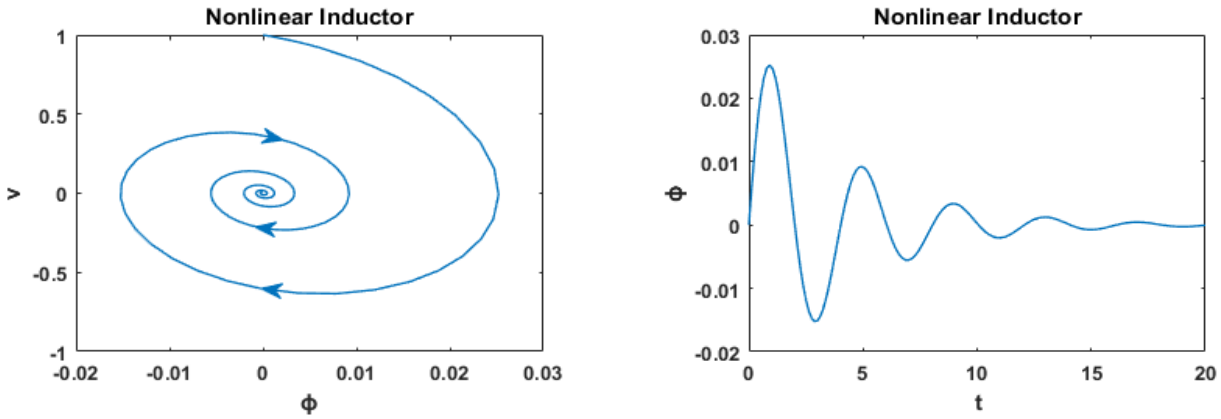
By using (6), we have

$$E_2'(t) = i_s(t)v - \frac{1}{2}v^2.$$

For $i_s(t) = 0$, it follows that

$$E_2'(t) = -\frac{1}{2}v^2 = -RI_R^2.$$

E_2 is verified by (1). The application of Theorem 2 shows that: $E_2' \leq 0$ on $\Re^2$, $E_2(\infty) = 0$ and $E_2(\phi, v) \to \infty$ as $\sqrt{\phi^2 + v^2} \to \infty$. Hence, all the motions of (6) are bounded (as illustrated in Figure 4a, 4b). The set S where $E_2' = 0$ is $\{\phi, 0\}$ and from (6) this implies that $\{0, 0\}$ is the only invariant subset of S, such that the zero solution or equilibrium solution of (6) is globally asymptotically stable. Thus, system (6) with its Lyapunov function satisfies all the assumptions of



(a) Phase plane plot of system (6) **(b)** Solution of system (6)

$[a = b = 1,\ R = 100\Omega,\ C = 20mF,\ L = 20H]$

Figure 4. The motions of system (6)

Theorem 2. Therefore, (6) or the related circuit is lossless due to the trivial solution which occurs at infinity. □

3.3. Nonlinear capacitor

A *nonlinear capacitive element* is present in the following circuit [29] and its voltage drop is no longer given by q/C, but is more accurately described by $\alpha q(t) + \beta q^3(t)$, where α and β are constants. The remaining elements L and R are positive scalars.

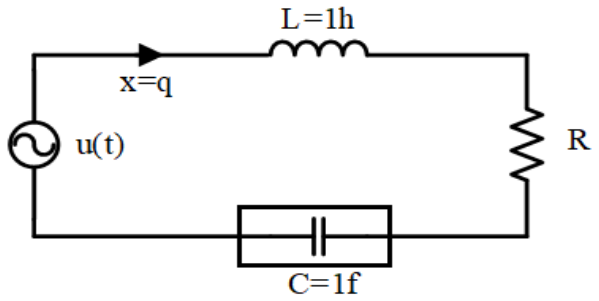


Figure 5. *LRC* circuit with nonlinear capacitive element.

The above dynamical system generates a differential equation of the form

$$q'' + Rq' + \alpha q + \beta q^3 = u(t).$$

Let, $q = x$ is the flow of the charge, α and β are real constants, and u is the applied voltage. Instead of the above equation

$$\begin{cases} x' = y, \\ y' = -Ry - \alpha x - \beta x^3 + u, \end{cases} \tag{8}$$

will be discussed.

Theorem 3. *The system* (8) *is stable if*

$$u = 0, \quad \alpha > 0, \quad \beta < 0$$

and it is globally asymptotically stable if $\beta < 0$ *is replaced by* $\beta > 0$.

Proof. Let $(x(t), y(t))$ be a solution of (8) for $t \geq 0$. In the case of $\beta < 0$: Let $\beta = -c$ ($c > 0$, α *constant*), the system has the equilibrium point $(0,0)$ and $(\pm\sqrt{\alpha c^{-1}}, 0)$. In the other case ($\beta > 0$), the system has $(0,0)$ and $(\pm\sqrt{\alpha\beta^{-1}}, 0)$. Therefore, $(0,0)$ is the only invariant equilibrium point of (8). For the first case we investigate the stability of (8). The storage energy function from power- energy relationship can be constructed in the neighborhood of the equilibrium point $(0,0)$ as

$$E_3(t) = E_3(x,y) = \frac{1}{2}y^2 + \int_0^x (\alpha x + \beta x^3)dx,$$

$$E_3(t) = E_3(x,y) = \frac{1}{2}y^2 + \frac{\alpha}{2}x^2 + \frac{\beta}{4}x^4.$$

The energy function ($E_3 : \Re^2 \to \Re^+$) satisfies

(i) $E_3(0) = 0$,
(ii) $E_3(x,y) > 0, \forall (x,y) \in \Re^2 - \{\sqrt{x^2+y^2} \neq 0\}$.

E_3 is not radially unbounded. But, in the neighborhood of $(0,0)$, E_3 is positive definite, and we have

$$E_3(t) \geq \frac{1}{2}y^2. \tag{9}$$

(9) obeys (i) and (ii) of Definition 2. The derivative of the energy function E_3 along the trajectories of system (8) gives

$$E_3' = yy' + \alpha xx' + \beta x^3 x'.$$

By using (8), we have

$$E_3' = -Ry^2 + uy.$$

For $u(t) = 0$, it follows that

$$E_3' = -Ry^2. \tag{10}$$

(10) is verified by (2) and system (8) is stable. . On the other hand, the integration of (10) from 0 to $t(\geq 0)$ gives

$$E_3(x(t), y(t)) \leq E_3(x(0), y(0)), \quad t \geq 0.$$

That is, E_3 is a decreasing function along the solution curve $(x(t), y(t))$, and $(0, 0)$ is a minimum point of E_3. The above inequality implies that the motion $(x(t), y(t))$ will stay in the neighborhood of the equilibrium point $(0, 0)$ for $t \geq 0$ provided that the initial point $(x(0), y(0)) = (x_0, y_0)$ is sufficiently near the point $(0, 0)$. Hence, the origin is stable. In addition, for $\alpha = 1$ and $\beta < 0$ we have the following inequalities.

$$E_3(x(t), y(t)) \leq x^2(t) + y^2(t)$$

and

$$E_3(x(0), y(0)) \leq x^2(0) + y^2(0).$$

Then, it follows that

$$x^2(t) \leq E_3(t) \leq x^2 + y^2,$$

$$x^2(t) + y^2(t) \leq x^2(0) + y^2(0), \quad t \geq 0.$$

Then, for any given $\epsilon > 0$, there is a $\delta > 0$. Thus if,

$$\sqrt{x^2(0) + y^2(0)} = \sqrt{x_0^2 + y_0^2}$$

and

$$\sqrt{x^2(t) + y^2(t)} < \epsilon.$$

Then

$$||x_0|| < \delta \quad \text{implies that} \quad ||x(t)|| < \epsilon.$$

This is precisely the most common definition of stability of a system which has an isolated equilibrium point $(0, 0)$.

In the case where $\beta > 0 : E_3'(x, y) \leq 0$ at all points $(x, y) \in \Re^2$. That is $E_3(t)$ is a decreasing function along any motion of (8), $E_3(\infty) = 0$ and $E_3(x) \to \infty$ as $||x|| \to \infty$.Hence, all the solutions of (8) are bounded.The set S, where $E_3 = 0$ is $(x, 0)$, and $(0, 0)$ is the only invariant subset of S. Thus, the application of Theorem 3 shows that the solution $x(t) = 0$ to (8) as $t \to \infty$. Therefore, the system is zero-state observable. This also implies that there is no energy dissipation in the circuit at infinity ($t = \infty$); that is, the circuit will be lossless at infinity. This explanation is compatible with Figure 6a, 6b. Besides, when the value of R increased, the motion goes to the equilibrium point immediately. □

The simulations are verifying our theoretical results. The trajectories in the phase spaces (Figure 2a, 4a, and 6a) go to the equilibrium solutions $(x(\infty), y(\infty)) = (0, 0)$. On the other hand, time series solutions (Figure 2b, 4b, and 6b) approach zero at infinity.

The three strong passivity results or the boundedness of the motions (strict passivity) of (4), (6) and (8) with their input functions are the following.

Theorem 4. *Suppose that all the conditions in Theorems 1, 2, and 3 are satisfied and also assume that* (3) *holds such that*

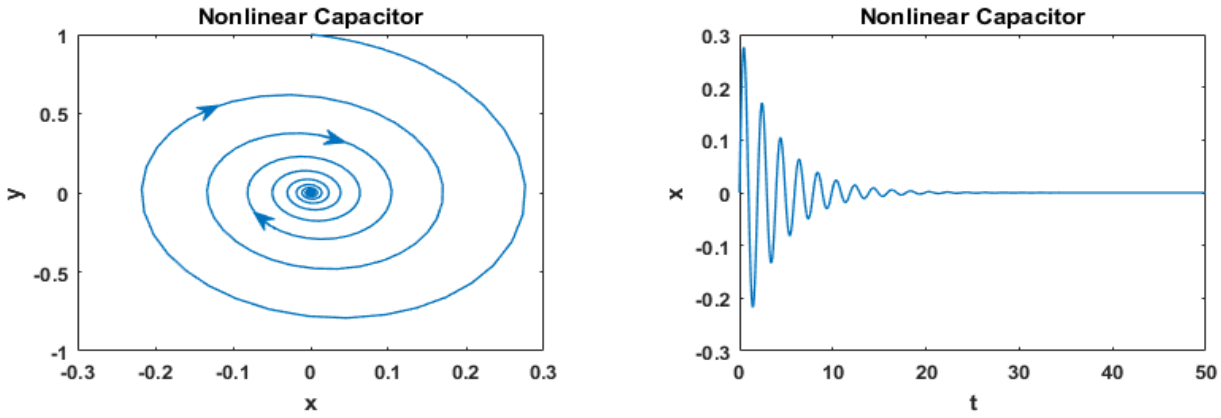
$$max \quad \{\int_0^t i(s)ds, \int_0^t i_s(s)ds, \int_0^t u(s)ds\} \leq K < \infty, \quad \forall t > 0,$$

where K is a positive constant. Then, all the motions of (4), (6) *and* (8) *with their forcing functions are bounded or the systems are strongly passive.*

Proof. From the proof of Theorem 1 we have

$$E_1'(t) = -\frac{v_1^2}{R_1} - \frac{v_2^2}{R_2} - v_2 f(v_2) + iv_1.$$

Then



(a) Phase plane plot of system (6) **(b)** Solution of system (6)

$[R = 0.5\Omega,\ \alpha = \beta = 10]$

Figure 6. The motions of system (6).

$$E_1'(t) + \frac{v_1^2}{R_1} + \frac{v_2^2}{R_2} \leq iv_1.$$

Hence, by Definition 3, the system (4) is strictly passive.

Furthermore

$$E_1'(t) \leq iv_1 \leq i(1 + v_1^2),$$

which also known as the dissipative inequality.

By (5), it follows that

$$E_1'(t) \leq i(t) + \frac{2}{C}E_1(t)i(t). \tag{11}$$

Integrating (11) from 0 to $t(> 0)$, and using Theorem 4, it follows that

$$E_1(t) \leq K + \frac{2}{C}\int_{t_0}^{t} E_1(s)i(s)ds.$$

Then, Gronwall's inequality [3] yields

$$E_1(t) \leq K\exp(\frac{2K}{C}). \tag{12}$$

Using the foregoing procedure, the following results are obtained which determine the upper bounds of E_2 and E_3, respectively.

$$E_2(t) \leq K\exp(\frac{2K}{C}), \tag{13}$$

and

$$E_3(t) \leq K\exp(2K), \tag{14}$$

Finally, the connection between (5) and (12), (7) and (13), and (9) and (14), respectively, give the following results:

$$\frac{1}{2}Cv_1^2 \leq E_1(t) \leq K\exp(\frac{2K}{C}),$$

$$\frac{1}{2}Cv^2 \leq E_2(t) \leq K\exp(\frac{2K}{C}),$$

and

$$\frac{1}{2}y^2 \leq E_3(t) \leq K\exp(2K).$$

Thus, the energy functions E_1, E_2, and E_3 are bounded. This also implies that all the motions of (4), (6), and (8) are bounded in magnitude. Hence, the related systems (or circuits) are strongly passive. □

4. Discussion

The properties of energy function and its Lie derivative determine the criteria of Lyapunov stability theory. Thus, our natural approach in this paper may be applicable to all physical systems whatever the orders of the systems. Here, we will only improve the stability of some second order systems that relevant to our study. In this connection, the stability of the following differential equations (with their arguments) has been investigated in [25] and [30], respectively,

$$x'' + a(t)f(x, x')x' + b(t)g(x) = 0 \tag{a1}$$

$$V_0 = \frac{1}{2}y^2 + b(t)\int_0^x g(\xi)d\xi + k, \quad (k > 0, \textit{constant}), \tag{a2}$$

$$V_0' = -a(t)f(x,y)y^2 + b'(t)\int_0^x g(\xi)d\xi; \tag{a3}$$

and

$$x'' + x' + p(t)g_1(x) + q(t)g_2(x) = 0, \tag{b1}$$

$$V(t,x,y) = \frac{1}{2}y^2 + p(t)G_1(x) + q(t)G_2(x), \tag{b2}$$

where $G_i(x) = \int_0^x g_i(\xi)d\xi \quad (i = 1, 2)$,

$$V'(t,x,y) = p'(t)G_1(x) - p(t)xg_1(x) + q'(t)G_2(x) - q(t)xg_2(x). \tag{b3}$$

Remark 1

There may be some objection regarding to the derivative of Lyapunov functions, but power-energy relationship shows that this objection is unfounded. For example, consider a series *LRC* circuit which has $b(t)g(q)$ voltage on a time varying nonlinear capacitor with $q(t)$ charge that flows in the circuit. Let P_C, W_C be the power and energy of the capacitor. Then, we have the followings:

(i) $P_C(t) = b(t)g(q)\frac{dq}{dt} = b(t)g(q)q'$,

(ii) $W_C(t) = \int_0^q b(t)g(s)ds, \quad W_C(0) = 0$

where $ds = q'(t)dt$. Then,

(iii) $\frac{d}{dt}W_C(t) = b(t)g(q)q'$.

Thus, for the construction of energy and power functions that associated with capacitors, we nicely apply the above loop. This may enable us to improve the stability of many systems, because the above algorithm eliminates to take the partial derivative of $W_C(t)$.

(A) The natural approach improves the result given in [25] such as:

$(a1)$ may represent a *LRC* circuit (dissipative) system with

$$x' = y, \quad y' = -a(t)f(x,y)y - b(t)g(x). \tag{a4}$$

The natural energy function for $(a4)$ must be

$$V(t,x,y) = \frac{1}{2}y^2 + \int_0^{x(t)} b(t)g(s)ds. \tag{a5}$$

Since, $(a5)$ has been constructed from the power-energy relationship of $(a4)$. The Lie derivative of $(a5)$ is

$$V'(t,x,y) = -\alpha(t)f(x,y)y^2. \tag{a6}$$

$(a6)$ is the dissipated power of $(a4)$ and verified by (1). The difference between $(a2)$ and $(a5)$, and between $(a3)$ and $(a6)$ state our improvement.

(B) The natural approach improves the results given in [30] such as:

The actual energy function for $(b1)$ is

$$V(t,x,y) = \frac{1}{2}y^2 + \int_0^{x(t)} [p(t)g_1(s) + q(t)g_2(s)]ds, \tag{b4}$$

where $ds = x'(t)dt$, $p > 0$ and $q > 0$ are continuous functions on $[0, \infty)$, g_1, g_2 are continuous functions on $\Re$, satisfying (A_1) of [30].

Then, the time derivative of (b4) along the solutions of (b1) gives

$$V'(t) = -y^2 < 0. \tag{b5}$$

In fact, the coefficient of x' in $(b1)$ is 1, and $(b5)$ is confirmed by (1) due to the suitable tool. The comparisons between $(b2)$ and $(b4)$, and between $(b3)$ and $(b5)$ give our improvement.

Further, the approach in this study also improves many results in the books [1] and [24] that based on Lyapunov approach. This list can be extended for the other related references that not cited here.

5. Conclusion

The energy of a system determines its behavior. In this context, this paper plays two important roles in the Lyapunov stability theory. First, it provides the construction of the energy function, which may obtain from the physical meaning of the given system. Second, it implies that the derivative of the energy function along the system trajectories is equal to the negative value of the

dissipated power in the system. These further clarify the Lyapunov stability. Hence, one can nicely check the derivative of the energy function of a given physical system with (1). The proposed approach can be applicable to higher order differential systems. From now on, everyone involved in the subject will be able to find the same stability results for a system under consideration. We hope this present work will open new doors in the stability analysis of differential systems.

References

[1] Vidyasagar, M. (2002). *Nonlinear System Analysis*. Society for Industrial and Applied Mathematics, USA.

[2] Tongren, D. (2007). *Approaches to the Qualitative Theory of Ordinary Differential Equations: Dynamical Systems and Nonlinear Oscillations*. World Scientific, Peking, China.

[3] Meiss, J.D. (2007). *Differential Dynamical Systems*. Revised Edition. Society for Industrial and Applied Mathematics, USA.

[4] Yang, C., Sun, J., Zhang, Q., & Ma, X. (2013). Lyapunov stability and strong passivity analysis for nonlinear descriptor systems. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 60(4), 1003–1012.

[5] Platonov, A.V. (2020). Stability analysis for nonlinear mechanical systems with nonstationary potential forces. *15th International Conference on Stability and Oscillations of Nonlinear Control Systems (Pyatnitskiy's Conference) (STAB)*, Moscow, Russia.

[6] Ateş, M., & Laribi, S. (2018). New results on the global asymptotic stability of certain nonlinear RLC circuits. *Turkish Journal of Electrical Engineering and Computer Science*, 26(1), 434–441.

[7] Zhang, L., & Yu, L. (2013). Global asymptotic stability of certain third-order nonlinear differential equations. *Mathematical Methods in the Applied Sciences*, 36(14), 1845–1850.

[8] Sen, N. (2020). Stability analysis of electrical RLC circuit described by the Caputo-Liouville generalized fractional derivative. *Alexandria Engineering Journal*, 59(4), 2083-2090.

[9] Nagamani, G., & Radhika, T. (2016). Dissipativity and passivity analysis of markovian jump neural networks with two additive time varying delays. *Neural Processing Letters*, 44(2), 571–592.

[10] Nagamani, G., Radhika, T., & Zhu, Q. (2017). An improved result on dissipativity and passivity analysis of markovian jump stochastic neural networks with two delay components. *IEEE Transactions on Neural Networks and Learning Systems*, 28(12), 3018–3031.

[11] Rao, M.R.M. (1981). *Ordinary Differential Equations Theory and Applications*. Edward Arnold, Inc., London, UK.

[12] Andreev, A., & Peregudova, O. (2020). The direct Lyapunov method in the motion stabilization problems of robot manipulators. *15th International Conference on Stability and Oscillations of Nonlinear Control Systems (Pyatnitskiy's Conference) (STAB)*, Moscow, Russia.

[13] Nagamani, G., Soundararajan, G., Subramaniam, R., & Azeem, M. (2020). Robust extended dissipativity analysis for Markovian jump discrete-time delayed stochastic singular neural networks. *Neural Computing and Applications*, 32(13), 9699–9712.

[14] Balasubramaniam P, & Nagamani, G. (2011). Global robust passivity analysis for stochastic interval neural networks with interval time-varying delays and markovian jumping parameters. *Journal of Optimization Theory and Applications*, 149(1), 197–215.

[15] Nagamani, G., Radhika T., & Balasubramaniam, P. (2015). A delay decomposition approach for robust dissipativity and passivity analysis of neutral-type neural networks with leakage time-varying delay. *Complexity*, 21(5), 248–264.

[16] Sastry, S. (1999). *Nonlinear Systems, Analysis, Stability, and Control*. Springer, New York, USA.

[17] Jeltsema, D., Ortega, R., & Scherpen, J.M.A. (2003). A novel passivity property of nonlinear RLC circuits. *Proceedings 4th Mathmod Sympositum (ARGESIM Report No. 24)*, Vienna, Austria, 845-853.

[18] Jeltsema, D., Ortega, R., & Scherpen, J.M.A. (2003). On passivity and power-balance inequalities of nonlinear RLC circuits. *IEEE Transactions on Circuits Systems I: Fundamental Theory and Applications*, 50(9), 1174–1179.

[19] Ramirez, H.S., & Lopez, E.M.N. (2000). On the passivity of general nonlinear systems. *Proceedings of 14th Symposium on Mathematical Theory of Networks and Systems*, Perpignan, France, 1–6.

[20] Zhu, F., Xia, M., & Antsaklis, P.J. (2014). Passivity Analysis and Passivation of Interconnected Event-Triggered Feedback Systems Using Passivity Indices. *Proceedings of the*

19th World Congress the International Federation of Automatic Control, Cape Town, South Africa, 24–29.

[21] Willems, J.C. (1972). Dissipative dynamical systems part I: General theory. *Archive for Rotational Mechanics and Analysis*, 45(5), 321–3519.

[22] Wang, J.L., Wu, H.N., Huang, T., Ren, S.Y., & Wu, J. (2018). Passivity and output synchronization of complex dynamical networks with fixed and adaptive coupling strength. *IEEE Transactions on Neural Networks and Learning Systems*, 29(2), 364–376.

[23] Wang, J.L., Wu, H.N., Huang, T., Ren, S.Y., & Wu, J. (2017). Passivity of directed and undirected complex dynamical networks with adaptive coupling weights. *IEEE Transactions on Neural Networks and Learning Systems*, 28(8), 1827–1839.

[24] Khalil, H.K. (2015). *Nonlinear Control*. Pearson Education, London, UK.

[25] Tunç, C., & Tunç, E. (2007). On the asymptotic behavior of solutions of second order differential equations. *Journal of The Franklin Institute*, 344(5), 391–398.

[26] Lee, T.C., Tan, Y., & Mareels, I. (2020). Detectability and uniform global asymptotic stability in switched nonlinear time-varying systems. *IEEE Transactions on Automatic Control*, 65(5), 2123–2138.

[27] Abate, A., Ahmed, D., Giacobbe, M., & Peruffo, A. (2021). Formal synthesis of Lyapunov neural networks. *IEEE Control Systems Letters*, 5(3), 773–778.

[28] Chua, L.O., Desoer, C.A., & Kuh, E.S. (1987). *Linear and Nonlinear Circuits*. McGraw-Hill, New York, USA.

[29] Dennis, D.Z., & Michael, R.C. (2000). *Advanced engineering mathematics*. Jones and Bartlett Publisher, London, UK.

[30] Sugie, J., & Amano, Y. (2004). Global asymptotic stability of nonautonomous systems of Lienard type. *Journal of Mathematical Analysis and Applications*, 289, 673-690.

Muzaffer Ateş *received his B.Sc. degree in Electrical and Electronics Engineering from METU, Turkey. Then he received his M.Sc. and PhD degrees in applied mathematics from Van Yuzuncu Yil University, Van, Turkey. Now, he is working at Electrical and Electronics Engineering Department in Van Yuzuncu Yil University. His research areas are nonlinear systems, mathematical control theory, circuit theory and Lyapunov stability theory.*

https://orcid.org/0000-0002-4394-0815

Nezir Kadah *received his B.Sc. degree from Mersin University, Mersin, Turkey, in 2012 and his M.Sc. degree from Van Yuzuncu Yil University, Van, Turkey, in 2019. He is currently a PhD student of Electrical and Electronics Engineering at Adana Alparslan Turkes Science and Technology University (ATU), since 2019. He is also working at the department of Information Technology of ATU as the system and network administrator. His research areas are nonlinear systems, system identification, and control theory.*

https://orcid.org/0000-0001-9320-1140

INSTRUCTIONS FOR AUTHORS

Aims and Scope

This journal shares the research carried out through different disciplines in regards to optimization, control and their applications.

The basic fields of this journal are linear, nonlinear, stochastic, parametric, discrete and dynamic programming; heuristic algorithms in optimization, control theory, game theory and their applications. Problems such as managerial decisions, time minimization, profit maximizations and other related topics are also shared in this journal.

Besides the research articles expository papers, which are hard to express or model, conference proceedings, book reviews and announcements are also welcome.

Journal Topics

- Applied Mathematics,
- Financial Mathematics,
- Control Theory,
- Game Theory,
- Fractional Calculus,
- Fractional Control,
- Modeling of Bio-systems for Optimization and Control,
- Linear Programming,
- Nonlinear Programming,
- Stochastic Programming,
- Parametric Programming,
- Conic Programming,
- Discrete Programming,
- Dynamic Programming,
- Optimization with Artificial Intelligence,
- Operational Research in Life and Human Sciences,
- Heuristic Algorithms in Optimization,
- Applications Related to Optimization on Engineering.

Submission of Manuscripts

New Submissions

Solicited and contributed manuscripts should be submitted to IJOCTA via the journal's online submission system. You need to make registration prior to submitting a new manuscript (please click here to register and do not forget to define yourself as an "Author" in doing so). You may then click on the "New Submission" link on your User Home.

IMPORTANT: If you already have an account, please click here to login. It is likely that you will have created an account if you have reviewed or authored for the journal in the past.

On the submission page, enter data and answer questions as prompted. Click on the “Next” button on each screen to save your work and advance to the next screen. The names and contact details of at least four internationally recognized experts who can review your manuscript should be entered in the "Comments for the Editor" box.

You will be prompted to upload your files: Click on the “Browse” button and locate the file on your computer. Select the description of the file in the drop down next to the Browse button. When you have selected all files you wish to upload, click the “Upload” button. Review your submission before sending to the Editors. Click the “Submit” button when you are done reviewing. Authors are responsible for verifying all files have uploaded correctly.

You may stop a submission at any phase and save it to submit later. Acknowledgment of receipt of the manuscript by IJOCTA Online Submission System will be sent to the corresponding author, including an assigned manuscript number that should be included in all subsequent correspondence. You can also log-

on to submission web page of IJOCTA any time to check the status of your manuscript. You will receive an e-mail once a decision has been made on your manuscript.

Each manuscript must be accompanied by a statement that it has not been published elsewhere and that it has not been submitted simultaneously for publication elsewhere.

Manuscripts can be prepared using LaTeX (.tex) or MSWord (.docx). However, manuscripts with heavy mathematical content will only be accepted as LaTeX files.

Preferred first submission format (for reviewing purpose only) is Portable Document File (.pdf). Please find below the templates for first submission.

Click here to download Word template for first submission (.docx)

Click here to download LaTeX template for first submission (.tex)

Revised Manuscripts

Revised manuscripts should be submitted via IJOCTA online system to ensure that they are linked to the original submission. It is also necessary to attach a separate file in which a point-by-point explanation is given to the specific points/questions raised by the referees and the corresponding changes made in the revised version.

To upload your revised manuscript, please go to your author page and click on the related manuscript title. Navigate to the "Review" link on the top left and scroll down the page. Click on the "Choose File" button under the "Editor Decision" title, choose the revised article (in pdf format) that you want to submit, and click on the "Upload" button to upload the author version. Repeat the same steps to upload the "Responses to Reviewers/Editor" file and make sure that you click the "Upload" button again.

To avoid any delay in making the article available freely online, the authors also need to upload the source files (Word or LaTex) when submitting revised manuscripts. Files can be compressed if necessary. The two-column final submission templates are as follows:

Click here to download Word template for final submission (.docx)

Click here to download LaTeX template for final submission (.tex)

Authors are responsible for obtaining permission to reproduce copyrighted material from other sources and are required to sign an agreement for the transfer of copyright to IJOCTA.

Article Processing Charges

There are no charges for submission and/or publication.

English Editing

Papers must be in English. Both British and American spelling is acceptable, provided usage is consistent within the manuscript. Manuscripts that are written in English that is ambiguous or incomprehensible, in the opinion of the Editor, will be returned to the authors with a request to resubmit once the language issues have been improved. This policy does not imply that all papers must be written in "perfect" English, whatever that may mean. Rather, the criteria require that the intended meaning of the authors must be clearly understandable, i.e., not obscured by language problems, by referees who have agreed to review the paper.

Presentation of Papers

Manuscript style

Use a standard font of the **11-point type: Times New Roman** is preferred. It is necessary to single line space your manuscript. Normally manuscripts are expected not to exceed 25 single-spaced pages including text, tables, figures and bibliography. All illustrations, figures, and tables are placed within the text at the appropriate points, rather than at the end.

During the submission process you must enter: (1) the full title, (2) names and affiliations of all authors and (3) the full address, including email, telephone and fax of the author who is to check the proofs. Supply a brief **biography** of each author at the end of the manuscript after references.

- Include the name(s) of any **sponsor(s)** of the research contained in the paper, along with **grant number(s).**
- Enter an **abstract** of no more than 250 words for all articles.

Keywords

Authors should prepare no more than 5 keywords for their manuscript.

Maximum five **AMS Classification number** (http://www.ams.org/mathscinet/msc/msc2010.html) of the study should be specified after keywords.

Writing Abstract

An abstract is a concise summary of the whole paper, not just the conclusions. The abstract should be no more than 250 words and convey the following:

1. An introduction to the work. This should be accessible by scientists in any field and express the necessity of the experiments executed.
2. Some scientific detail regarding the background to the problem.
3. A summary of the main result.
4. The implications of the result.
5. A broader perspective of the results, once again understandable across scientific disciplines.

It is crucial that the abstract conveys the importance of the work and be understandable without reference to the rest of the manuscript to a multidisciplinary audience. Abstracts should not contain any citation to other published works.

Reference Style

Reference citations in the text should be identified by numbers in square brackets "[ ]". All references must be complete and accurate. Please ensure that every reference cited in the text is also present in the reference list (and vice versa). Online citations should include date of access. References should be listed in the following style:

Journal article

Author, A.A., & Author, B. (Year). Title of article. Title of Journal, Vol(Issue), pages.

Castles, F.G., Curtin, J.C., & Vowles, J. (2006). Public policy in Australia and New Zealand: The new global context. Australian Journal of Political Science, 41(2), 131–143.

Book

Author, A. (Year). Title of book. Publisher, Place of Publication.

Mercer, P.A., & Smith, G. (1993). Private Viewdata in the UK. 2nd ed. Longman, London.

Chapter

Author, A. (Year). Title of chapter. In: A. Editor and B. Editor, eds. Title of book. Publisher, Place of publication, pages.

Bantz, C.R. (1995). Social dimensions of software development. In: J.A. Anderson, ed. Annual review of software management and development. CA: Sage, Newbury Park, 502–510.

Internet document

Author, A. (Year). Title of document [online]. Source. Available from: URL [Accessed (date)].

Holland, M. (2004). Guide to citing Internet sources [online]. Poole, Bournemouth University. Available from: http://www.bournemouth.ac.uk/library/using/guide_to_citing_internet_sourc.html [Accessed 4 November 2004].

Newspaper article

Author, A. (or Title of Newspaper) (Year). Title of article. Title of Newspaper, day Month, page, column.

Independent (1992). Picking up the bills. Independent, 4 June, p. 28a.

Thesis

Author, A. (Year). Title of thesis. Type of thesis (degree). Name of University.

Agutter, A.J. (1995). The linguistic significance of current British slang. PhD Thesis. Edinburgh University.

Illustrations

Illustrations submitted (line drawings, halftones, photos, photomicrographs, etc.) should be clean originals or digital files. Digital files are recommended for highest quality reproduction and should follow these guidelines:

- 300 dpi or higher
- Sized to fit on journal page
- TIFF or JPEG format only
- Embedded in text files and submitted as separate files (if required)

Tables and Figures

Tables and figures (illustrations) should be embedded in the text at the appropriate points, rather than at the end. A short descriptive title should appear above each table with a clear legend and any footnotes suitably identified below.

Proofs

Page proofs are sent to the designated author using IJOCTA EProof system. They must be carefully checked and returned within 48 hours of receipt.

Offprints/Reprints

Each corresponding author of an article will receive a PDF file of the article via email. This file is for personal use only and may not be copied and disseminated in any form without prior written permission from IJOCTA.

Submission Preparation Checklist

As part of the submission process, authors are required to check off their submission's compliance with all of the following items, and submissions may be returned to authors that do not adhere to these guidelines.

1. The submission has not been previously published, nor is it before another journal for consideration (or an explanation has been provided in Comments for the Editor).
2. The paper is in PDF format and prepared using the IJOCTA's two-column template.
3. All references cited in the manuscript have been listed in the References list (and vice-versa) following the referencing style of the journal.
4. There is no copyright material used in the manuscript (or all necessary permissions have been granted).
5. Details of all authors have been provided correctly.
6. ORCID profile numbers of "all" authors are mandatory, and they are provided at the end of the manuscript as in the template (visit https://orcid.org for more details on ORCID).
7. The text adheres to the stylistic and bibliographic requirements outlined in the Author Guidelines.
8. Maximum five AMS Classification number (http://www.ams.org/mathscinet/msc/msc2010.html) of the study have been provided after keywords.
9. The names and email addresses of at least FOUR (4) possible reviewers have been indicated in "Comments for the Editor" box in "Paper Submission Step 1-Start". Please note that at least two of the recommendations should be from different countries. Avoid suggesting reviewers you have a conflict of interest.

Peer Review Process

All contributions, prepared according to the author guidelines and submitted via IJOCTA online submission system are evaluated according to the criteria of originality and quality of their scientific

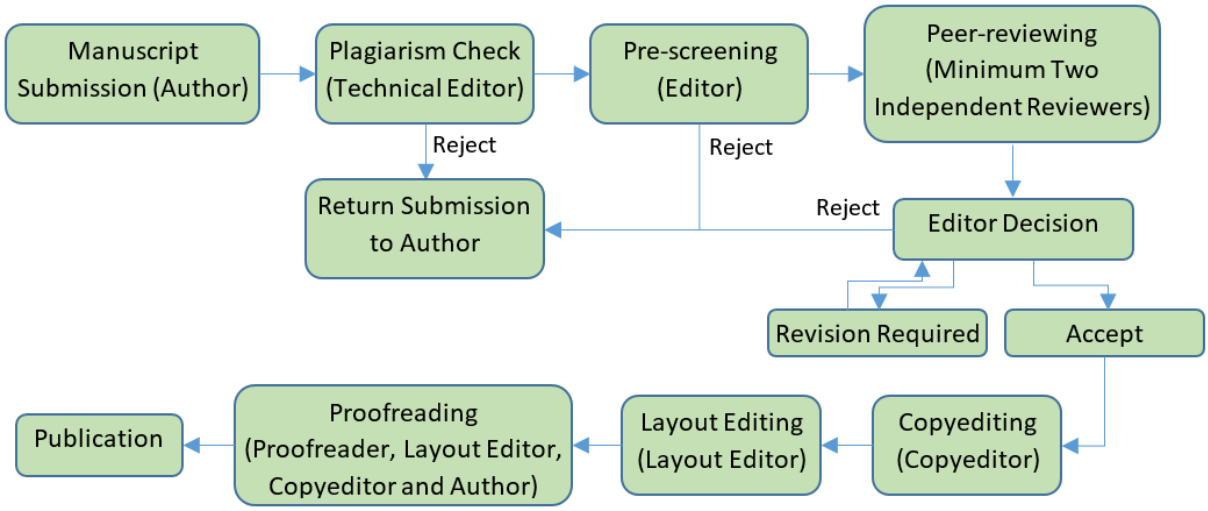
content. The corresponding author will receive a confirmation e-mail with a reference number assigned to the paper, which he/she is asked to quote in all subsequent correspondence.

All manuscripts are first checked by the Technical Editor using plagiarism detection software (iThenticate) to verify originality and ensure the quality of the written work. If the result is not satisfactory (i.e. exceeding the limit of 30% of overlapping), the submission is rejected and the author is notified.

After the plagiarism check, the manuscripts are evaluated by the Editor-in-Chief and can be rejected without reviewing if considered not of sufficient interest or novelty, too preliminary or out of the scope of the journal. If the manuscript is considered suitable for further evaluation, it is first sent to the Area Editor. Based on his/her opinion the paper is then sent to at least two independent reviewers. Each reviewer is allowed up to four weeks to return his/her feedback but this duration may be extended based on his/her availability. IJOCTA has instituted a blind peer review process where the reviewers' identities are not known to authors. When the reviews are received, the Area Editor gives a decision and lets the author know it together with the reviewer comments and any supplementary files.

Should the reviews be positive, the authors are expected to submit the revised version usually within two months the editor decision is sent (this period can be extended when the authors contact to the editor and let him/her know that they need extra time for resubmission). If a revised paper is not resubmitted within the deadline, it is considered as a new submission after all the changes requested by reviewers have been made. Authors are required to submit a new cover letter, a response to reviewers letter and the revised manuscript (which ideally shows the revisions made in a different color or highlighted). If a change in authorship (addition or removal of author) has occurred during the revision, authors are requested to clarify the reason for change, and all authors (including the removed/added ones) need to submit a written consent for the change. The revised version is evaluated by the Area editor and/or reviewers and the Editor-in-Chief brings a decision about final acceptance based on their suggestions. If necessary, further revision can be asked for to fulfil all the requirements of the reviewers.

When a manuscript is accepted for publication, an acceptance letter is sent to the corresponding author and the authors are asked to submit the source file of the manuscript conforming to the IJOCTA two-column final submission template. After that stage, changes of authors of the manuscript are not possible. The manuscript is sent to the Copyeditor and a linguistic, metrological and technical revision is made, at which stage the authors are asked to make the final corrections in no more than a week. The layout editor prepares the galleys and the authors receive the galley proof for final check before printing. The authors are expected to correct only typographical errors on the proofs and return the proofs within 48 hours. After the final check by the layout editor and the proofreader, the manuscript is assigned a DOI number, made publicly available and listed in the forthcoming journal issue. After printing the issue, the corresponding metadata and files published in this issue are sent to the databases for indexing.



Publication Ethics and Malpractice Statement

IJOCTA is committed to ensuring ethics in publication and quality of articles. Conforming to standards of expected ethical behavior is therefore necessary for all parties (the author, the editor(s), the peer reviewer) involved in the act of publishing.

International Standards for Editors

The editors of the IJOCTA are responsible for deciding which of the articles submitted to the journal should be published considering their intellectual content without regard to race, gender, sexual orientation, religious belief, ethnic origin, citizenship, or political philosophy of the authors. The editors may be guided by the policies of the journal's editorial board and constrained by such legal requirements as shall then be in force regarding libel, copyright infringement and plagiarism. The editors may confer with other editors or reviewers in making this decision. As guardians and stewards of the research record, editors should encourage authors to strive for, and adhere themselves to, the highest standards of publication ethics. Furthermore, editors are in a unique position to indirectly foster responsible conduct of research through their policies and processes.

To achieve the maximum effect within the research community, ideally all editors should adhere to universal standards and good practices.

- Editors are accountable and should take responsibility for everything they publish.
- Editors should make fair and unbiased decisions independent from commercial consideration and ensure a fair and appropriate peer review process.
- Editors should adopt editorial policies that encourage maximum transparency and complete, honest reporting.
- Editors should guard the integrity of the published record by issuing corrections and retractions when needed and pursuing suspected or alleged research and publication misconduct.
- Editors should pursue reviewer and editorial misconduct.
- Editors should critically assess the ethical conduct of studies in humans and animals.
- Peer reviewers and authors should be told what is expected of them.
- Editors should have appropriate policies in place for handling editorial conflicts of interest.

Reference:

Kleinert S & Wager E (2011). Responsible research publication: international standards for editors. A position statement developed at the 2nd World Conference on Research Integrity, Singapore, July 22-24, 2010. Chapter 51 in: Mayer T & Steneck N (eds) Promoting Research Integrity in a Global Environment. Imperial College Press / World Scientific Publishing, Singapore (pp 317-28). (ISBN 978-981-4340-97-7) [Link].

International Standards for Authors

Publication is the final stage of research and therefore a responsibility for all researchers. Scholarly publications are expected to provide a detailed and permanent record of research. Because publications form the basis for both new research and the application of findings, they can affect not only the research community but also, indirectly, society at large. Researchers therefore have a responsibility to ensure that their publications are honest, clear, accurate, complete and balanced, and should avoid misleading, selective or ambiguous reporting. Journal editors also have responsibilities for ensuring the integrity of the research literature and these are set out in companion guidelines.

- The research being reported should have been conducted in an ethical and responsible manner and should comply with all relevant legislation.
- Researchers should present their results clearly, honestly, and without fabrication, falsification or inappropriate data manipulation.
- Researchers should strive to describe their methods clearly and unambiguously so that their findings can be confirmed by others.
- Researchers should adhere to publication requirements that submitted work is original, is not plagiarised, and has not been published elsewhere.
- Authors should take collective responsibility for submitted and published work.
- The authorship of research publications should accurately reflect individuals' contributions to the work and its reporting.

- Funding sources and relevant conflicts of interest should be disclosed.
- When an author discovers a significant error or inaccuracy in his/her own published work, it is the author's obligation to promptly notify the journal's Editor-in-Chief and cooperate with them to either retract the paper or to publish an appropriate erratum.

Reference:

Wager E & Kleinert S (2011) Responsible research publication: international standards for authors. A position statement developed at the 2nd World Conference on Research Integrity, Singapore, July 22-24, 2010. Chapter 50 in: Mayer T & Steneck N (eds) Promoting Research Integrity in a Global Environment. Imperial College Press / World Scientific Publishing, Singapore (pp 309-16). (ISBN 978-981-4340-97-7) [Link].

Basic principles to which peer reviewers should adhere

Peer review in all its forms plays an important role in ensuring the integrity of the scholarly record. The process depends to a large extent on trust and requires that everyone involved behaves responsibly and ethically. Peer reviewers play a central and critical part in the peer-review process as the peer review assists the Editors in making editorial decisions and, through the editorial communication with the author, may also assist the author in improving the manuscript.

Peer reviewers should:

- respect the confidentiality of peer review and not reveal any details of a manuscript or its review, during or after the peer-review process, beyond those that are released by the journal;
- not use information obtained during the peer-review process for their own or any other person's or organization's advantage, or to disadvantage or discredit others;
- only agree to review manuscripts for which they have the subject expertise required to carry out a proper assessment and which they can assess within a reasonable time-frame;
- declare all potential conflicting interests, seeking advice from the journal if they are unsure whether something constitutes a relevant conflict;
- not allow their reviews to be influenced by the origins of a manuscript, by the nationality, religion, political beliefs, gender or other characteristics of the authors, or by commercial considerations;
- be objective and constructive in their reviews, refraining from being hostile or inflammatory and from making libellous or derogatory personal comments;
- acknowledge that peer review is largely a reciprocal endeavour and undertake to carry out their fair share of reviewing, in a timely manner;
- provide personal and professional information that is accurate and a true representation of their expertise when creating or updating journal accounts.

Reference:

Homes I (2013). COPE Ethical Guidelines for Peer Reviewers, March 2013, v1 [Link].

Copyright Notice

An International Journal of Optimization and Control: Theories & Applications

Volume: 11 Number: 2
July 2021

CONTENTS

www.ijocta.org
editor@ijocta.org

ISSN 2146-0957