Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Evolutionary Optimized Padé Approximation Scheme for Analysis of Covid-19 Model with Crowding Effect

Javaid Ali, Ali Raza, Nauman Ahmed, Ali Ahmadian, Muhammad Rafiq, Massimiliano Ferrara

PII: S2214-7160(21)00024-5
DOI: https://doi.org/10.1016/j.orp.2021.100207
Reference: ORP 100207

To appear in: Operations Research Perspectives

Received date: 29 June 2021
Revised date: 1 November 2021
Accepted date: 8 November 2021

Please cite this article as: Javaid Ali, Ali Raza, Nauman Ahmed, Ali Ahmadian, Muhammad Rafiq, Massimiliano Ferrara, Evolutionary Optimized Padé Approximation Scheme for Analysis of Covid-19 Model with Crowding Effect, Operations Research Perspectives (2021), doi: https://doi.org/10.1016/j.orp.2021.100207

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2021 Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Evolutionary Optimized Padé Approximation Scheme for Analysis of Covid-19 Model with Crowding Effect

Javaid Ali¹, Ali Raza²,³, Nauman Ahmed⁴, Ali Ahmadian⁵,⁶,⁸*, Muhammad Rafiq⁷, Massimiliano Ferrara⁸

¹Department of Mathematics, GGCT, Punjab Higher Education Department, Lahore, Pakistan
²Department of Mathematics, National College of Business Administration and Economics Lahore, Pakistan.
³Department of Mathematics, Govt. Maulana Zafar Ali Khan Graduate College Wazirabad, Punjab Higher Education Department, Lahore, Pakistan.
⁴Department of Mathematics and Statistics, The University of Lahore, Pakistan.
⁵Institute of IR 4.0, The National University of Malaysia, 43600 UKM, Selangor, Malaysia.
⁶Department of Mathematics, Near East University, Nicosia, TRNC, Turkey
⁷Department of Mathematics, Faculty of Sciences, University of Central Punjab, Lahore, Pakistan
⁸Department of Law, Economics and Human Sciences & Decisions Lab, University Mediterranea of Reggio Calabria, 89125, Reggio Calabria Italy
Highlights:

- A novel evolutionary computation based Padé approximation (EPA) scheme for Covid-19 disease with crowding effect
- The crowding effect driven system is transformed to an equivalent nonlinear global optimization problem by assimilating Padé rational functions
- A hybrid of differential evolution (DE) and a convergent variant of Nelder-Mead Simplex algorithm is also proposed
- EPA-based solutions of Covid-19 model with crowding effect are in good agreement with those of a well-practiced non-standard finite difference (NSFD) scheme
Abstract
This work presents a novel evolutionary computation-based Padé approximation (EPA) scheme for constructing a closed-form approximate solution of a nonlinear dynamical model of Covid-19 disease with a crowding effect that is a growing trend in epidemiological modeling. In the proposed framework of the EPA scheme, the crowding effect-driven system is transformed to an equivalent nonlinear global optimization problem by assimilating Padé rational functions. The initial conditions, boundedness, and positivity of the solution are dealt with as problem constraints. Keeping in view the complexity of formulated optimization problem, a hybrid of differential evolution (DE) and a convergent variant of the Nelder-Mead Simplex algorithm is also proposed to obtain a reliable, optimal solution. The comparison of the EPA scheme results reveals that optimization results of all formulated optimization problems for the Covid-19 model with crowding effect are better than those of several modern metaheuristics. EPA-based solutions of the Covid-19 model with crowding effect are in good agreement with those of a well-practiced nonstandard finite difference (NSFD) scheme. The proposed EPA scheme is less sensitive to step lengths and converges to true equilibrium points unconditionally.
1. Literature survey

Bats were the source of Covid-19, which SARS-nCoV2 caused. SARS-CoV1 and MERS-CoV1 are similar viruses. Its genetic material, RNA, is encased in a lipid-layered surface with spike proteins that resemble the shape of a crown, hence the name corona. In December 2019, it was first seen in Wuhan, China. In February 2020, China, Italy, Iran, and South Korea were present. Then, by the middle of March 2020, it had spread all over the planet. Since then, it has appeared in waves in different parts of the world due to significantly modified forms. The elderly, immune-compromised, diabetics, cancer patients, and heart patients are the most vulnerable. It takes 4-14 days to incubate. The majorities of cases are mild, with symptoms such as fever, illness, and cough. A cytokine storm can induce shortness of breath, body aches, nausea, pneumonia, respiratory failure, and death in severe circumstances. It spreads through the inhalation of deadly virus respiratory droplets produced by infected people and breathed by adjacent people. To detect suspected patients, we employ PCR tests, and to see suspected recovered patients; we use antibody tests. Distancing, masking, hand washing, lockdown, and mass vaccination are still being used to manage the disease. Since its inception, vaccination has shown promising effects. Nonetheless, it is proving to be the only reliable way out of the problem.

Daniel Bernoulli developed the first mathematical epidemic model in 1760 to assess the efficacy of the smallpox vaccine. Modeling nowadays incorporates continual advancements in computational tools and illness data. Understanding disease dynamics, evaluating viable control techniques, and predicting future outbreaks are all aided by this information. They help to flatten the curve of the Covid-19 pandemic in the country by predicting disease patterns. The mathematical modeling of biological processes allows epidemiological assumptions to be revised. As a result, we can put our grasp of disease epidemiology to the test by comparing model findings to observed trends.

In 2020, Zeb et al. proposed a SEIQR mathematical model to control Covid-19 propagation via optimal control techniques [1]. Recently, in 2021, several complex epidemiological models have been submitted. For example, Riyapan et al. presented a SEIQRD mathematical model of Covid-19 in Bangkok, which explains the dynamical analysis of the proposed model [2]. Oud et al. suggested a SEIAQHRM mathematical model for Covid-19 to study the quarantine effect,
which describes how curable the vaccination for the dreadful virus or which strategy may be adopted during the vaccination [3]. Shaikh et al. presented a SEIARQ model of the coronavirus disease to check its outbreak in India to explain the strain of the virus through mathematical modeling [4]. Ahmed et al. presented a SEQIR model of coronavirus to study the relationship between symptomatic and asymptomatic classes [5]. In 2020, Ullah et al. introduced a SEIIQIIR model to comprehend the use of different drugs as vaccinations [6]. Peter et al. (in 2021) presented a SEIQR model to estimate the damages caused by a coronavirus in Pakistan [7]. Recently, Nazir et al. studied a SEIARW model to numerically analyze the effectiveness of the Euler method [8]. Kyrychko et al. presented a SEIAR model to calculate the propagation and role of agents in spreading coronavirus in Ukraine [9]. Khoshnaw suggested that medical awareness and keeping social distances were more effective in controlling the spread of coronaviruses [10]. Sasmita et al. presented a model that considers different subpopulations interactions for control of Covid-19 [11]. In 2020, Tiwar et al. predicted the importance of lockdown for controlling the spread of Covid-19 disease in India through the SEIRD model [12]. Wang et al., in 2021, proposed a SEIRD model which incorporates the different ways for the spread of disease [13]. Baek et al. proposed the SEIR model for controlling the spread of Covid-19 via quarantine and hospitalization [14]. Ardila et al. suggested that isolation and staying away are the key factors preventing the spread of Covid-19 [15]. Peter et al. presented the susceptible (S), Exposed (E), infected (I), Quarantine (Q), carriers (C), and recovered (R) model for controlling the disease in Nigeria [16]. In 2020, Moyles et al. proposed that social distancing is highly effective in preventing the spread of infection [17]. Ko et al. suggested the correlation of SEIR-HCD and SEIR-D model for the control of disease in Moscow [18]. Recently, Harjule has indicated the importance of modeling in controlling the spread of Covid-19 [19]. In 2020, Kim et al. proposed that staying home, keeping distance and early detection are the core points in controlling the spread of Covid-19 [20]. Some well-known models related to Covid-19 with different mathematical strategies are presented in the literature referred to in this study [21-29].

Modeling differential equations as equivalent optimization problems [30-38] is one of the most modern techniques to obtain reliable solutions, especially when the traditional method lacks handling one or more aspects of the physical phenomena. The related examples of such modeling include solving nonlinear ordinary differential equations, inverse partial differential equations, and epidemiological models. Ali et al. [37, 38] proposed a novel evolutionary Padé
approximation (EPA) scheme for solving epidemiological models and nonlinear partial differential equations. Padé rational approximation functions are a vital research area due to their tremendous applications in several scientific areas [39, 40]. Henri Eugene Padé [41] established suitable Padé approximations for series expansion of functions even where the Taylor series failed to converge. In past studies, Padé approximations were employed to enhance the precision of the solution only obtained by other master methods based on power series [42], differential transform [43], homotopy analysis [44], and Adomian decomposition [45].

The Padé approximation-based formulated optimization problem from differential equations may involve complex non-convexity, multimodality, and non-differentiability. Traditional mathematical programming approaches are not capable of solving such complex problems. On the other hand, evolutionary algorithms are capable of solving almost all kinds of optimization problems. They belong to a general class of optimization techniques known as metaheuristics. Metaheuristics are designed by imitating some natural procedures like flashing behavior of fireflies [46], student-teacher learning procedure [47], swarming [48], evolution [49, 50], water dynamics [51-53], sport strategies [54], food foraging behavior [55], animals’ hunting [56], etc. For more detailed studies, one may consult the survey article by Alexandros and Georgios [57].

Unfortunately, the theoretical authentication of metaheuristics is still deficient. Past studies [46, 58] have highlighted some severe challenges in the applications of metaheuristics. Despite their successful applications to very diverse problems, there are No Free Lunch (NFL) theorems [58] which reveal that one cannot guarantee that a method effective for one set of trials will be equally efficient for the other collection of problems. Keeping the complexity of the underlying problem in view, hybridization is an effective technique to improve the performance of the selected global search optimizer. As far as the present study is concerned, we have chosen a widely used classic version [59] of the worldwide search Differential Evolution (DE) algorithm [49] in combination with a convergent variant [60] of the Nelder-Mead Simplex (NMS) method [61] to obtain better results.

This work presents the first EPA scheme-based formulation and solution of the Covid-19 disease dynamical model with crowding effect as an optimization problem to the best of our knowledge. In the original proposal [37, 38] of the EPA scheme, the focus on optimization was minimal, and the underlying model was comparatively less complex. But the present study adds some new aspects to the theory of the EPA scheme for solving epidemiological models with crowding
effect. The architecture of the EPA scheme consists of the following novel features that were lacking in the previous studies.

(i) The impact of variation in the order of Padé approximants for state variables on resulting cost function has been studied.

(ii) A comparative study of the performance of several modern metaheuristics on formulated equivalent unconstrained optimization problems has been conducted, and the essence of the proposed hybrid evolutionary optimizer has been elaborated.

(iii) The strength of mathematical programming and metaheuristics for solving complex epidemiological models has been highlighted.

(iv) The EPA scheme preserves all vital properties like positivity, boundedness, feasibility, convergence to equilibrium points, etc., by enforcing the solution to satisfy related problem constraints.

(v) The EPA scheme's convergence speed is high compared to a well-practiced Nonstandard Finite Difference (NSFD) [62] structure-preserving method.

The remainder of the manuscript is organized as follows: Section 2 presents the formulation of the Covid-19 model with crowding effect along with stability analysis of equilibrium points. Section 3 consists of a detailed procedure of all components of the EPA scheme for solving the Covid-19 model with the crowding effect. Section 4 presents a rigorous analysis of the results of the EPA scheme in many aspects of reliability, efficiency, and structure-preserving ability. The concluding remarks are also presented in the end.

2. Model Formulation

With the help of the theory of population dynamics, we consider the whole population is denoted by \( N \), and subpopulations are defined as Susceptible humans \( S(t) \), Infected humans \( I(t) \), and Recovered humans \( R(t) \). Furthermore, the fixed values of the model are defined as: \( \mu \): (recruitment rate/mortality rate of humans), \( \beta \): (the force of infection), \( \frac{1}{1+ \alpha t} \): (represents the crowding effect in a population), and \( \gamma \): (the rate of recovery due to immunity or vaccination). The dreadful situation will occur when the "\( \alpha \)" increases continuously. Eventually, the rate of active cases is elevated. Hence, there is a direct relationship between the crowding factor and active patients of coronavirus. Here arises the inevitability of controlling the parameter "\( \alpha \)" so
that the dynamics of coronavirus disease may be governed. From the mathematical modeling point of view, we achieve the following coupled nonlinear dynamical system:

\[
\begin{align*}
\frac{dS}{dt} &= \mu N - \frac{\beta SI}{1 + \alpha I} - \mu S \\
\frac{dI}{dt} &= \frac{\beta SI}{1 + \alpha I} - (\gamma + \mu)I \\
\frac{dR}{dt} &= \gamma I - \mu R
\end{align*}
\]  

From the mathematical modeling point of view, we achieve the following coupled nonlinear dynamical system:

\[
\begin{align*}
\frac{dS}{dt} &= \mu N - \frac{\beta SI}{1 + \alpha I} - \mu S \\
\frac{dI}{dt} &= \frac{\beta SI}{1 + \alpha I} - (\gamma + \mu)I \\
\frac{dR}{dt} &= \gamma I - \mu R
\end{align*}
\]  

Fig. 1. Flow diagram of Covid-19 model with crowding effect [29].

2.1. Analysis of Model

2.1.1. Positivity and Boundedness

The feasible region of the system (1-3) as follows:

\[
H_0 = \{(S, I, R) \in \mathbb{R}_+^3: S + I + R = N, S \geq 0, I \geq 0, R \geq 0\}.
\]

For this purpose, we will investigate the following results.

**Theorem 1:** The solutions \((S, I, R) \in \mathbb{R}_+^3\) of the system (1-3) are positive at any time \(t \geq 0\), with given non-negative initial conditions.

Proof: It is clear from the system (1-3) as follows:

\[
\left. \frac{dS}{dt} \right|_{S=0} = \mu N \geq 0, \left. \frac{dI}{dt} \right|_{I=0} = 0 \geq 0, \left. \frac{dR}{dt} \right|_{R=0} = \gamma I \geq 0.
\]
Theorem 2: The solutions \((S, I, R) \in \mathbb{R}_+^3\) of the system (1-3) are bounded.

Proof: Let us consider the population function as follows:

\[
N(t) = S + I + R
\]

\[
\frac{dN}{dt} = \frac{dS}{dt} + \frac{dI}{dt} + \frac{dR}{dt}
\]

\[
\frac{dN}{dt} = \mu N - \mu (S + I + R), \quad S + I + R = N
\]

\[
\frac{dN}{dt} = \mu N - \mu N = 0.
\]

Since the total population is fixed, the system’s solution (1-3) is bounded and lies in the feasible region \(I_b\).

2.1.2. Model Equilibria

The system (1-3) admits two types of equilibria as follows:

Disease free equilibrium (DFE) \(E_0 = (N, 0, 0)\).

Endemic equilibrium (EE) \(E_* = (S_*, I_*, R_*)\).

\[
S^* = a_1 + a_2 I_*, \quad I_* = \frac{-A_1 + \sqrt{A_2^2 - 4A_1A_3}}{2A_1}, \quad R^* = \frac{\nu I_*}{\mu}
\]

where, \(a_1 = \frac{\gamma + \mu}{\beta}, a_2 = \frac{\alpha (\gamma + \mu)}{\beta}, A_2 = + (\beta a_1 + \mu (a_1 + a_2) - \mu N \alpha), A_1 = a_2 (\beta + \alpha), A_3 = (\mu a_2 - \mu N)\).

Note that, the reproduction number of the model \(R_0 = \rho (FV^{-1}) = \frac{\beta N}{\gamma + \mu}\).

2.2. Local Stability Analysis of the Model

Theorem 3: The system (1-3) is locally stable related to the virus-free equilibrium point \(E_0\), if \(R_0 < 1\) and unstable if \(R_0 > 1\).

Proof: For local stability, the Jacobian of the system (1-3) is as follows:
At $E_0$, the Jacobian becomes

$$j(E_0) = \begin{pmatrix} -\mu & -\beta N & 0 \\ 0 & \beta N - (\gamma + \mu) & 0 \\ 0 & \gamma & -\mu \end{pmatrix}.$$ 

These are the following eigen values as

$$\lambda_1 = -\mu < 0, \lambda_2 = \beta N - (\gamma + \mu) < 0, \text{ if } R_0 < 1, \lambda_3 = -\mu < 0.$$ 

Thus, point $E_0$ is locally asymptotically stable.

**Theorem 4:** For $R_0 > 1$, the system (1-3) at the positive endemic equilibrium $E_*$ of the system (1-3) is locally stable.

**Proof:** The Jacobian matrix of system (1-3) is as follows:

$$j = \begin{pmatrix} -\mu - \frac{\beta_1}{1 + a l_s} & -\frac{\beta s}{(1 + a l_s)^2} & 0 \\ \frac{\beta_1}{1 + a l_s} & \frac{\beta s}{(1 + a l_s)^2} - (\mu + \gamma) & 0 \\ 0 & \gamma & -\mu \end{pmatrix}.$$ 

At $E_*$, the Jacobian becomes

$$j(E_*) = \begin{pmatrix} -\mu - \frac{\beta_1}{1 + a l_s} & -\frac{\beta s_*}{(1 + a l_s)^2} & 0 \\ \frac{\beta_1}{1 + a l_s} & \frac{\beta s_*}{(1 + a l_s)^2} - (\mu + \gamma) & 0 \\ 0 & \gamma & -\mu \end{pmatrix}.$$ 

Which yields one eigenvalue $\lambda = -\mu < 0$ and the characteristics equation

$$\lambda^2 + (\mu + \frac{\beta_1}{1 + a l_s} - \frac{\beta s_*}{(1 + a l_s)^2} + (\mu + \gamma)) \lambda + (\mu + \frac{\beta_1}{1 + a l_s})(\frac{\beta s_*}{(1 + a l_s)^2} + (\mu + \gamma)) + (\frac{\beta s_*}{(1 + a l_s)^2})(\frac{\beta_1}{1 + a l_s}) = 0.$$
It is clear, for $R_0 > 1$, and by using Routh Hurwitz criteria for 2\textsuperscript{nd} order
\[
(\mu + \frac{\beta I_e}{1+\alpha I_e} - \frac{\beta S_e}{(1+\alpha I_e)^2} + (\mu + \gamma)) = (\mu + \frac{\beta I_e}{1+\alpha I_e} - \frac{(\mu+\gamma)}{(1+\alpha I_e)^2} + (\mu + \gamma)) > 0.
\]
And
\[
(\mu + \frac{\beta I_e}{1+\alpha I_e})(\frac{\beta S_e}{(1+\alpha I_e)^2} + (\mu + \gamma)) + (\frac{\beta S_e}{(1+\alpha I_e)^2})(\frac{\beta I_e}{1+\alpha I_e}) > 0.
\]

Hence, the system (1-3) is locally stable at $E_*$ for $R_0 > 1$. The proof is complete.

3. Evolutionary Padé Approximation (EPA) Scheme for Covid-19 Model

This section is dedicated to the architecture of the EPA scheme for solving the approximate model. The first component of the proposed scheme constructs an equivalent optimization problem by modeling the coupled system of governing equations described in Eqs. (1-3) using the Padé approximation. The second component executes the optimization task by implementing a classical DE algorithm [59]. To obtain global optimum solutions of the formulated problem, DE is further hybridized with a non-stagnated Nelder-Mead Simplex (NS-NMS) algorithm [60] that is a convergent variant of a simplex search based Nelder-Mead method [61]. The unconstrained minimization objective function $\psi(x)$ is highly nonlinear and may contain discontinuities and several local optima.

Moreover, as the orders ($M_p, N_p$) of Padé approximations increase, the curse of dimensionality involves. In the presence of such challenges, the use of an efficient and reliable optimization algorithm is complementary. We use the DE algorithm, which has earned very high ranks in the optimization community due to its tremendous success in almost all optimization problems.

3.1. Formulation of Equivalent Optimization Problem.

Formulation of equivalent optimization problem consists of four key factors that are (i) approximating the state variables by Padé approximations (ii) conversion of each governing equation to a residual function (iii) handling the initial conditions as problem constraints (iv) use of penalty functions for obtaining unconstrained global optimization problem. The global optimum solution of the resulting unconstrained optimization problem is analogous to the solution of the Covid-19 model with a crowding effect.
(I) Approximation by Padé rational functions

First, we approximate the state variables $S(t)$, $I(t)$, and $R(t)$ by Padé rational functions of appropriate orders. For any $v \in \{S, I, R\}$, let $A_v(t)$ and $B_v(t)$ denote polynomial functions of variable $t$ having degrees $M_v$ and $N_v$, respectively. Then Padé rational approximation functions of orders $(M_v, N_v) \in \mathbb{W}^2$ [37, 38] for state variables are:

$$S(t) \approx p_S(t) = \frac{A_S(t)}{B_S(t)}, I(t) \approx p_I(t) = \frac{A_I(t)}{B_I(t)}, R(t) \approx p_R(t) = \frac{A_R(t)}{B_R(t)}.$$

Here $A_v(t)$ and $B_v(t)$ are polynomial functions with real coefficients $a_{vj}, b_{vj}$ for each $j \in \mathbb{W}$ defined by:

$$A_v(t) = \sum_{j=0}^{M_v} a_{vj} t^j; B_v(t) = \sum_{j=0}^{N_v} b_{vj} t^j.$$

The derivative of $p_v(t)$ with respect to $t$ can be expressed as under:

$$p'_v(t) = \frac{A'_v(t) - p_v(t)B'_v(t)}{B_v(t)} \quad \forall \ v \in \{S, I, R\}.$$

(II) Formation of residual functions

The domain of $t$ for the Covid-19 model is $[0, \infty)$. For some positive integer $K$, let us consider a monotone strictly increasing sequence $\{t_0, t_1, t_2, t_3, ..., t_K\}$ as a subset of the domain $[0, \infty)$. The proposed scheme does not depend on step length to create such a sequence, but any appropriate step length can be used for ease of implementation. Let us present the unknown coefficients of Padé approximation functions for each variable $v \in \{S, I, R\}$ by a point $x_v = (a_{v0}, a_{v1}, ..., a_{vM_v}, b_{v0}, b_{v1}, b_{v2}, ..., b_{vN_v}) \in \mathbb{R}^{M_v+N_v+2}$, then there is a total number of $n = M_S + N_S + M_I + N_I + M_R + N_R + 6$ of unknown coefficients. Representing the corresponding approximate solutions and their derivatives by $p_{vr}, p'_{vr}$ at $t_r \in \{t_0, t_1, t_2, t_3, ..., t_K\}$ and then substituting in Eq. 1-3 we get a system of $3 \times (K + 1)$ nonlinear algebraic equations. Let us define real-valued residual functions $\varepsilon_S(x, t), \varepsilon_I(x, t)$ and $\varepsilon_R(x, t)$ of vector $x = (x_S, x_I, x_R) \in \mathbb{R}^n$ of unknown Padé approximation coefficients at any time $t \in [0, \infty)$. From (1-3), at each $t_r \in \{t_0, t_1, t_2, t_3, ..., t_K\}$ we get the following residuals from:
Handling initial conditions as problem constraints

The initial conditions and the physical restrictions give rise to the following equality and non-negativity constraints functions.

\[ H_S(x) = p_S(t_0) - S_0 = 0 \]  
\[ H_I(x) = p_I(t_0) - I_0 = 0 \]  
\[ H_R(x) = p_R(t_0) - R_0 = 0 \]  
\[ g_{vr}(x) = p_v(t_r) \geq 0, \forall v \in \{S, I, R\} \text{ and } t_r \in \{t_0, t_1, t_2, ..., t_K\}. \]

Formulation of unconstrained global optimization problem

The Padé approximation-based mathematical modeling of the underlying problem aims to find optimized decision vector \( x = (x_S, x_I, x_R) \in \mathbb{R}^n \) of unknown coefficients so that the residuals at each discrete time step are minimized by meeting all the problem constraints. It can be observed from Eqs. (7-10) that the residual functions are highly nonlinear and hence lead to a highly nonlinear optimization problem. Due to the high complexity of the resulting optimization problem, an efficient optimizer is needed to be used. Evolutionary algorithms are modern methods that are believed to be less sensitive to the nature of the underlying problem, but they require the problem to be unconstrained. For this purpose, we use the penalty function approach to embed problem constraints into the objective function. Let \( L \) be a large positive real number and \( P(x) \) be the function of the following relation.

\[ P(x) = \max \{0, |H_S(x)|, |H_I(x)|, |H_R(x)|, -g_S(x), -g_I(x), -g_R(x)\}. \]

Then the following unconstrained minimization problem is obtained

\[ \text{Minimize } \psi(x) = \frac{1}{3(K+1)} \sum_{r=0}^{K} (\sum_{v \in \{S, I, R\}} |e_v(x, t_r)|^2) + L \times P(x). \]
3.2. Differential Evolution (DE) Global Search Optimizer

DE algorithm [49] is a population-based random search algorithm that starts with a randomly generated set of several solutions known as population. DE algorithm tries to improve the current population by developing new solutions through mutation and crossover evolutionary operations. The individual of the population with the smallest objective function value is termed as the best solution. The iteration process of the DE algorithm continues until a preset criterion is satisfied. The standard DE algorithm uses only three parameters to work that are (i) the population size $\kappa \in \mathbb{N}$ (ii) crossover rate $c_r \in [0,1]$ and (iii) differential weight $\zeta > 0$. Geometric view of DE operations is shown in Fig. 2 (a). Several studies describe the classic DE algorithm, also referred to as DE/rand/1/bin, in precise and smarter forms. We have implemented the traditional DE algorithm as Price, Storn and Lampinen explained in a detailed study [59] of the DE algorithm. To minimize a function of the form $\psi : \mathbb{R}^n \rightarrow \mathbb{R}$, the sequence of operations of classic DE algorithm [59] is given below:

1. **Initialization**
   - The parameters $\kappa$, $c_r$ and $\zeta$ are set. A positive integer $T_{\text{max}}$ as a maximum number of iterations is also fixed. The iteration counter $\tau$ is initiated as $\tau = 0$.
   - A population of $\kappa$ solutions $\{x_j = (x_j^1, x_j^2, x_j^3, \ldots, x_j^n) \in \mathbb{R}^n; 1 \leq j \leq \kappa\}$ is generated randomly.
   - The $\kappa$ solutions are evaluated i.e. $\psi_j = \psi(x_j) \forall j = 1,2,3,\ldots,\kappa$. The best solution $x_{\text{best}} = \arg\{\min\{\psi_j: 1 \leq j \leq \kappa\}\}$ is preserved.

2. **Iterative process**
   - Set $\tau = \tau + 1$.
   - For each target solution $x_j, \ j \in \{1,2,3,\ldots,\kappa\}$, choose mutually distinct indices $j_1,j_2,j_3 \in \{1,2,3,\ldots,\kappa\}\setminus\{j\}$.
   - Construct mutant vector $u$ using differential mutation according as $u = x_{j_1} + \zeta \times (x_{j_2} - x_{j_3})$.
   - Apply uniform/discrete crossover as:
For each dimension \( i = 1,2,3,\ldots,n \), choose a random integer \( j_{\text{rand}} \in \{1,2,3,\ldots,n\} \) and generate a uniformly distributed random number \( i_{\text{rand}} \in (0,1) \) to find a trial solution \( \mathbf{y} = (y^1, y^2, y^3, \ldots, y^n) \in \mathbb{R}^n \) with coordinates:

\[
y^i = \begin{cases} 
  u^i, & \text{if } i_{\text{rand}} \leq c_r \text{ or } i = j_{\text{rand}} \\
  x^i, & \text{otherwise}
\end{cases}
\]

If \( \psi(\mathbf{y}) > \psi_j \) discard \( \mathbf{y} \) otherwise \( x_j \leftarrow \mathbf{y} \) and \( \psi_j \leftarrow \psi(\mathbf{y}) \).

- Update the best solution as: If \( \psi_j \leq \psi(\mathbf{x}^{\text{best}}) \) then \( \mathbf{x}^{\text{best}} \leftarrow \mathbf{x}_j \).

3. **Termination criterion**

   If \( \tau < T_{\text{max}} \) then repeat step 2; otherwise, display the best solution and stop.

4. **Optimum results**

   Display \( \mathbf{x}^{\text{best}} \) and \( \psi(\mathbf{x}^{\text{best}}) \) as the optimum solution to the problem.

### 3.3. Nelder-Mead Simplex (NMS) Local Optimizer

Nelder-Mead Simplex (NMS) algorithm [61] is a well-practiced gradient-free optimization method. For an \( n \)-dimensional problem, the NMS method starts with an initial \( n \)-simplex having non-zero volume. An \( n \)-simplex is a convex hull of points \( \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n+1} \) in \( \mathbb{R}^n \). The vertices are ordered from the best to the worst according to as \( \psi(\mathbf{x}_1) \leq \psi(\mathbf{x}_2) \leq \psi(\mathbf{x}_3) \leq \cdots \leq \psi(\mathbf{x}_{n+1}) \). At each iteration, the centroid \( \mathbf{G} = \frac{1}{n} \sum_{j=1}^{n} \mathbf{x}_j \) of all the non-worst vertices is computed. Then the operations of reflection, expansion, and contractions about \( \mathbf{G} \) are used to replace the worst vertex. In case of failure, the simplex is squeezed towards the best vertex using shrink operation.

The equations of four NMS operations are given below.

1. **Reflection**: \( \mathbf{y} = \mathbf{x}_{n+1} + \alpha \times (\mathbf{G} - \mathbf{x}_{n+1}) \).
2. **Expansion**: \( \mathbf{y} = \mathbf{x}_{n+1} + \beta \times (\mathbf{G} - \mathbf{x}_{n+1}) \).
3. **Contraction outside**: \( \mathbf{y} = \mathbf{x}_{n+1} + \gamma \times (\mathbf{G} - \mathbf{x}_{n+1}) \).
4. **Contraction inside**: \( \mathbf{y} = \mathbf{x}_{n+1} - \gamma \times (\mathbf{G} - \mathbf{x}_{n+1}) \).

   - (iv) **Shrink**: \( \mathbf{x}_j = \mathbf{x}_j + \delta \times (\mathbf{x}_1 - \mathbf{x}_j) \ \forall \ j = 2,3,\ldots,n+1 \).

The typical values of these parameters in the original study [61] are \( \alpha = 2, \beta = 3, \gamma = \delta = 0.5 \).

We use a non-stagnated convergent variant of the NMS method proposed by Ali et al. [60]. The order of operations of the NMS method is outlined as a flowchart presented in Fig. 2 (b). The
complete schematic framework of the proposed EPA scheme for the underlying model is presented in Fig. 3.

Fig. 2. (a) Geometry of DE algorithm in $\mathbb{R}^2$ (b) Schematic flow chart of NMS method.
Fig. 3. Schematic flow chart of the proposed EPA scheme.
4. Results and Discussion

This section consists of essential components of the current work. The principal objective of the present study is to obtain the semi-analytical solution of the considered model with high accuracy and preserve vital properties of the dynamics of the model. It is imperative to investigate the EPA scheme's accuracy, efficiency, and reliability when applied to the considered model. Therefore, in the following subsections, we conduct a twofold study of the solution obtained by the EPA scheme. Firstly, we analyze the optimization results of our proposed EPA scheme. Secondly, we compare the solutions obtained by the EPA scheme with a well-practiced nonstandard finite difference (NSFD) scheme to highlight the prominent aspects of the proposed method.

4.1 Analysis of Optimization Results

We observe that the global minimum value of the objective function $\psi(x)$ defined by Eq. (11) is zero, which corresponds to a global optimal point $x^\dagger$. Keeping in mind the positivity of residual functions $e_s(x,t), e_I(x,t), e_R(x,t)$ and the penalty function $P(x)$ for all $t$ and $x$, we have that:

$$\psi(x^\dagger) = 0 \iff e_v(x^\dagger, t) = 0 \quad \forall \ v \in \{S, I, R\} \text{ and } P(x^\dagger) = 0.$$

The equation $e_v(x^\dagger, t) = 0$ implies that $x^\dagger$ is the exact solution of the governing equation of $v \in \{S, I, R\}$, whereas $P(x^\dagger) = 0$ guarantees that all initial conditions and the mentioned structure-preserving properties of the model are accurately satisfied.

The proposed EPA scheme tries to find the approximate solution, say $x^{**}$, of the model with acceptable accuracy. Hence the objective of the optimization process is to minimize $\psi(x)$ as much as possible by ensuring that all problems constraints are satisfied. To optimize the formulated problem by NMS-DE, the parameters used in the proposed EPA scheme have been presented in Table 1.

The optimal values of the Padé approximation coefficients have been found under various options of orders Padé approximations at endemic and disease-free equilibrium states. To validate the performance of the proposed hybrid optimizer reasonably and explore the interpolating strength of Padé approximants, we have considered Padé rational functions of orders $(4, 4)$, $(5, 5)$ and $(6, 6)$ to approximate each state variable for endemic as well as disease-
free equilibrium. Table 2 presents the statistical measures (best, mean, and standard deviation (SD)) of optimization results of $\psi(x)$ produced by the proposed hybrid optimizer in 20 independent runs. Mean convergence curves of 2000 iterations of optimization progress using (4, 4), (5, 5), and (6, 6) order approximants have been exhibited in Fig. 4.

### Table 1 EPA parameters.

| Parameter | Description | Value | Reference |
|-----------|-------------|-------|-----------|
| $\kappa$  | Population size for DE algorithm | 50    | [49, 59]  |
| $c_r$     | The crossover acceptance rate of DE algorithm | 0.9   | [49, 59]  |
| $\zeta$   | Differential weight of DE algorithm | 0.5   | [49, 59]  |
| $T_{\text{max}}$ | Maximum number of iterations for DE algorithm to produce the optimum solution | 2000  | Assumed   |
| $(M_v, N_v)$ | Order of Padé approximation for state variable $v$ | (4,4), (5,5), (6,6) |       |
| $n$       | Problem dimension denoting the number of unknown Padé approximation coefficients | $\sum_{v \in [I,J,K]} (M_v + N_v)$ |       |
| $L$       | The penalty factor of penalty function | $10^9$ |           |
| $\mu$     | Death rate | 0.1   | [29]      |
| $\beta$   | Bilinear incidence rate | 0.2464 (EE) 0.1464 (DFE) | [29]    |
| $\gamma$  | Recovered rate | 0.07  | [29]      |
| $\alpha$  | Infection rate | 0.01  | [29]      |
| $V_{\text{threshold}}$ | The threshold value for convergence speed | $10^{-04}$ | Assumed |

### Table 2 Optimization results for two equilibrium states.

| Order | DFE | EE |
|-------|-----|----|
|       | Best | Mean | SD | Best | Mean | SD |
| (4,4) | 2.5e-07 | **8.1e-06** | 4.9e-04 | 8.9e-08 | 8.9e-06 | 3.1e-05 |
| (5,5) | **1.7e-11** | 3.6e-05 | 1.1e-05 | 4.2e-09 | **1.2e-05** | 1.6e-05 |
| (6,6) | 8.4e-07 | 9.4e-06 | **5.3e-06** | **3.7e-08** | 5.7e-05 | **7.1e-06** |

Tables 3-4 represent the optimum values of unknown coefficients, which belong to the smallest optimal values of the objective function over 20 runs.
### Table 3 Optimal values of coefficients at DFE state.

| Order | i   | $a_{s_i}$ | $b_{s_i}$ | $a_{l_i}$ | $b_{l_i}$ | $a_{R_i}$ | $b_{R_i}$ |
|-------|-----|-----------|-----------|-----------|-----------|-----------|-----------|
| (4,5) | 0   | 0.5       | 1         | 0.2       | 1         | 0.3       | 1         |
|       | 1   | 11.62412  | 23.19136  | -0.441054 | -1.99006  | 4.518759  | 15.1704   |
|       | 2   | 17.96686  | 27.99012  | 21.62841  | 129.7301  | 41.81043  | 151.2964  |
|       | 3   | -4.589881 | -4.792515 | -0.5847947| -40.31095 | -0.7930848| -13.7038  |
|       | 4   | 3.214352  | 3.215415  | 0.0045273 | 71.09928  | 0.0038315 | 14.43962  |
| (5,5) | 0   | 0.5       | 1         | 0.2       | 1         | 0.3       | 1         |
|       | 1   | 1.3390699 | 2.6037853 | 0.7072810 | 3.7679691 | 2.1501625 | 7.2888593 |
|       | 2   | 788.49054 | 1570.245  | -0.8496248| -5.7637786| 1.4227432 | 4.0616136 |
|       | 3   | 10.919599 | 36.620152 | 9.3823315 | 56.037333 | 7.9462428 | 30.799049 |
|       | 4   | 4.7504681 | 3.9790506 | -0.1915813| -7.508662 | -0.152888 | -2.5337034|
|       | 5   | 4.7330172 | 4.7324437 | 0.0001154 | 5.4266051 | -0.001095 | 0.7767076 |
| (6,6) | 0   | 0.5       | 1         | 0.2       | 1         | 0.3       | 1         |
|       | 1   | 1.08109   | 2.0329    | 1.2794    | 6.50122   | 0.452094  | 1.55232   |
|       | 2   | 3.2641    | 8.91808   | 8.57519   | 47.9268   | 1.48791   | 5.34671   |
|       | 3   | 10.8111   | 10.5012   | 6.6515    | 31.3825   | 6.66342   | 21.5409   |
|       | 4   | 8.40516   | 26.7405   | 0.304318  | 20.9876   | 5.69837   | 24.053    |
|       | 5   | 2.10008   | 1.75478   | -0.0081925| -1.98912  | -0.119462 | -1.33771  |
|       | 6   | 1.41142   | 1.41119   | 9.945e-05 | 1.2269    | 6.887e-05 | 0.603988  |

### Table 4 Optimal values of coefficients at EE state.

| Order | i   | $a_{s_i}$ | $b_{s_i}$ | $a_{l_i}$ | $b_{l_i}$ | $a_{R_i}$ | $b_{R_i}$ |
|-------|-----|-----------|-----------|-----------|-----------|-----------|-----------|
| (4,4) | 0   | 0.5       | 1         | 0.2       | 1         | 0.3       | 1         |
|       | 1   | 1.962976  | 3.873777  | 10.50587  | 52.57363  | 4.722648  | 115.74965 |
|       | 2   | 5.103082  | 9.836035  | 16.82871  | 96.37293  | 32.34391  | 111.6259  |
|       | 3   | -0.2254903| -0.3935344| -14.07242 | -77.26163 | -0.5267137| -1.305307 |
|       | 4   | 0.561096  | 0.8118427 | 3.553253  | 19.55911  | 0.082824  | 8.514212  |
| (5,5) | 0   | 0.5       | 1         | 0.2       | 1         | 0.3       | 1         |
|       | 1   | -0.2527975| -0.5398225| 4.208253  | 21.04604  | 2.752185  | 9.216065  |
|       | 2   | 21.55926  | 43.33501  | 1.465227  | 11.2728   | 17.88936  | 64.84665  |
|       | 3   | 71.79745  | 139.7148  | -7.792861 | -44.28272 | 16.88882  | 54.20429  |
|       | 4   | 3.595083  | 4.53876   | 2.867689  | 15.78287  | 1.212035  | 11.0835   |
|       | 5   | 8.729702  | 12.6298   | 1.85344   | 10.20357  | 0.6390833 | 5.025223  |
| (6,6) | 0   | 0.5       | 1         | 0.2       | 1         | 0.3       | 1         |
|       | 1   | -2.70156  | -5.4178   | 4.88544   | 24.4964   | 7.69363   | 25.8147   |
|       | 2   | -6.08778  | -12.5332  | 0.197156  | 8.6703    | 11.8406   | 39.6912   |
|       | 3   | -4.29643  | -4.72722  | 9.74047   | 28.7846   | 9.08754   | 50.9515   |
|       | 4   | -8.35969  | -17.1982  | 8.09084   | 66.408    | 7.75748   | 15.1099   |
|       | 5   | 3.24166   | -4.79335  | -0.899575 | -5.46805  | -0.0815278| 0.269303  |
|       | 6   | -2.99372  | -4.33166  | 1.66903   | 9.18781   | 1.07708   | 8.47145   |
Fig. 4. Mean convergence curves of DE-NMS at (a) DFE (b) EE state.

Fig. 5. Squared residuals of governing equation of class $S$ at (a) EE (b) DFE state.

Fig. 6. Squared residuals of governing equation of class $I$ at (a) EE (b) DFE state.
In the following subsections, the optimization results are presented and analyzed as:

(i) The accuracy of the solution is assessed by the fact that how small $\psi(x^{**})$ is found.

(ii) Consistency of NMS-DE algorithm demonstrated by computing best, mean and standard deviations of final optimal values over several optimizer attempts.

(iii) The convergence speed of NMS-DE is assessed by many iterations used to achieve a threshold value denoted by the threshold.

(iv) The effectiveness of the proposed method is justified by comparing the obtained solutions with multiple modern metaheuristics.

### 4.1.1 Accuracy

From Table 2, we observe that the best minimum values of $\psi(x^{**})$ ranges from 2.5e-07 to 1.7e-11 at DFE point and lie in the range [3.7e-08, 4.2e-09] at EE point for various orders of Padé approximants. Figs. 5-7 present the squared residuals of governing equations of state variables over a time of up to 150 days. One can observe from figures that the squared residuals can be minimized up to sufficiently small values 10e-15. Moreover, Fig. 5 shows that the solutions of governing equation of susceptible population achieve accuracies below 10e-6. In contrast, Fig. 6 shows that the accuracy of an infected population is below 10e-10 at both the equilibrium points and for all orders of the Padé approximants. Similarly, the accuracy of solutions of governing equation of susceptible population falls around 10e-10 and 10e-06 at DFE and EE points,
respectively. These results ensure that the solutions found by the EPA scheme can achieve approximate solutions of the model with good accuracies.

4.1.2 Consistency and Reliability

To assess the consistency of NMS-DE, we do not rely on the solution achieved by a single run of the optimizer. To investigate the reliability of the results, we perform a statistical analysis of marks obtained in multiple runs of the NMSE-DE algorithm. The results presented in Table 2 describe that the mean of the optimal value of $\psi(x^{*})$ remains within the range [8.1e-06, 5.7e-05] for all considered orders and at both equilibrium points with corresponding range [5.3e-06, 4.9e-04] for standard deviations. These facts and figures demonstrate that the performance of NMS-DE in optimization tasks is consistent, and results are reliable with good accuracy.

4.1.3 Convergence Speed

Fig. 4 presents the mean convergence curves of the optimization process at two equilibrium points for all orders of Padé approximants. We can observe from Fig. 4 (a) that, at DFE point, the threshold value $V_{\text{threshold}}$ is achieved in less than 500, 600, and 900 iterations on average for (4, 4), (5, 5), and (6,6) order solutions respectively. Similarly, $V_{\text{threshold}}$ is achieved in around 400, 500, and 750 iterations (please see Fig. 4 (b)) for (4, 4) in corresponding solutions at EE point. These observations highlight that the order of Padé approximants is directly proportional to computational cost and speed of convergence.

4.1.4 Performance Comparison with Other Metaheuristics

In this subsection, we present a comparison of the optimization results of our proposed method with original DE and NMS algorithms and with a few modern metaheuristics under the same settings of population size, the maximum number of iterations, and the threshold value. The other built-in parameters of all competing algorithms are kept the same as recommended in their original propositions. The metaheuristics used for comparison are Teaching-Learning Based Optimization (TLBO) [47], Particle Swarm Optimization (PSO) [48], DE [49], Controlled Showering Optimization (CSO) [52], Water Cycle Algorithm (WCA) [53], Artificial Bee Colony (ABC) [55] and Grey Wolf Optimizer (GWO) [56]. Tables 5 and 6 exhibit the best minimum value, percentage ($P\%$) of successful runs to achieve $V_{\text{threshold}}$, and an average number of
iterations ($T_a$) in successful runs to achieve $V_{\text{threshold}}$ over 20 independent runs of each algorithm that are used as performance measures to compare the results of all competing metaheuristics.

### Table 5 Optimization results of competing metaheuristics at DFE point.

| Algorithm | (4, 4) order solution | (5, 5) order solution | (6, 6) order solution |
|-----------|-----------------------|-----------------------|-----------------------|
|           | Best $P\%$ $T_a$      | Best $P\%$ $T_a$      | Best $P\%$ $T_a$      |
| TLBO      | 1.4e-04  58 1078     | 5.5e-05  74 1142     | 1.5e-06  68 1212     |
| PSO       | 3.8e-06  64 1685     | 1.0e-06  31 1715     | 7.1e-05  48 1729     |
| DE        | 3.3e-06  71 1011     | 9.2e-07  70 1501     | 8.7e-06  81 1811     |
| NMS       | 7.0e+06  0 -         | 3.5e+06  0 -         | 8.1e+06  0 -         |
| CSO       | 7.1e-03  53 1623     | 4.6e-06  63 1493     | 5.6e-05  57 1677     |
| WCA       | 5.1e-04  0 -         | 8.0e-04  0 -         | 1.4e-04  0 -         |
| ABC       | 3.3e-01  0 -         | 2.2e00  0 -          | 9.1e-01  0 -         |
| GWO       | 6.2e-02  0 -         | 2.8e-04  0 -         | 7.9e-03  0 -         |
| NMS-DE    | 2.5e-07  100 413     | 1.7e-11  100 668     | 8.4e-07  100 891     |

### Table 6 Optimization results of competing metaheuristics at EE point.

| Algorithm | (4, 4) order solution | (5, 5) order solution | (6, 6) order solution |
|-----------|-----------------------|-----------------------|-----------------------|
|           | Best $P\%$ $T_a$      | Best $P\%$ $T_a$      | Best $P\%$ $T_a$      |
| TLBO      | 2.3e-05  60 1053     | 1.7e-05  70 1083     | 1.5e-05  71 976      |
| PSO       | 4.6e-06  60 1723     | 2.1e-05  20 1690     | 1.5e-05  70 1729     |
| DE        | 6.1e-07  64 1303     | 1.2e-06  73 1429     | 5.8e-06  68 1691     |
| NMS       | 3.4e+08  0 -         | 5.9e+06  0 -         | 7.7e+06  0 -         |
| CSO       | 5.1e-05  50 1225     | 3.7e-05  60 1551     | 5.6e-05  60 1677     |
| WCA       | 2.9e-04  0 -         | 4.0e-04  0 -         | 9.1e-04  0 -         |
| ABC       | 4.0e00   0 -         | 1.8e00   0 -         | 2.4e01   0 -         |
| GWO       | 3.7e-02  0 -         | 9.1e-03  0 -         | 1.5e-02  0 -         |
| NMS-DE    | 8.0e-08  100 973     | 4.2e-09  100 1023    | 3.7e-08  100 1264    |

The features of results presented in Tables 5 and 6 are comprised of six test instances of solutions of various Padé approximants’ orders at two equilibrium points. We observe that due to significant dependency on provided initial guesses and the complexity of the underlying problem, the performance of the NMS algorithm is nowhere comparable to all other global search algorithms. Hence we exclude NMS results from further comparative analysis. One can observe from these results that the metaheuristics WCA, ABC, and GWO were unsuccessful in achieving the threshold value $V_{\text{threshold}}$ in all attempts and returned poor approximations of optimum values. Their mutual comparison is based on the best objective function values only.
WCA was able to minimize the objective function better than ABC and GWO. Therefore, further competition occurs between TLBO, PSO, DE, and CSO, and NMS-DE algorithms.

Comparing the results of TLBO, PSO, DE, CSO, and NMS-DE, the following vital facts can be observed:

1. The DE algorithm computed the best minimum objective function values are better than TLBO, PSO, and CSO for all test instances. The best minimum values found by NMS-DE are far better than those of all of its competitors. The optimal values found by NMS-DE are 92.42%, 99.998%, and 90.34% better than those of the DE algorithm for respective cases of DFE point. At EE point, NMSE found 85.41%, 99.65%, and 99.36% better values than the DE algorithm for three respective cases of endemic equilibrium.

2. NMS-DE algorithm outperformed other metaheuristics in the sense of success rates (100%) of achieving pre-set accuracy $V_{threshold}$ in all scenarios. On the other hand, none of the competing metaheuristics achieved such a consistent performance while solving the formulated optimization problem of the CovidCE model. For the 5th order solution at DFE point and 6th order solution at EE point, only TLBO showed success rates of 74% and 71%, respectively, better than those of DE, but DE outperformed TLBO, PSO, and CSO on the rest of the cases. This observation justifies the selection of DE as a global solver and the need for further hybridization to achieve a more efficient hybrid solver (NMS-DE).

3. The final notable point is the speed of convergence of the proposed NMS-DE is faster than those of TLBO, PSO, DE, and CSO. The computational cost is measured by the number of iterations required to achieve the threshold value $V_{threshold}$. At DFE point, NMS-DE consumed around 413 (20.65%), 668 (33.4%), and 891 (44.55%) iterations for 4th, 5th, and 6th order solution, respectively. At EE point these amounts are 973 (48.65%), 1023 (51.15%) and 1274 (63.2%) respectively. Fig. 8 presents the competitive percentage measure of the faster or slower convergence speed of NMS-DE over each successful metaheuristic. The positive and negative percentages reflect that the convergence speed of NMS-DE is faster or slower than a competing algorithm. The convergence speed of NMS-DE is better than all of the competitors except for the 6th order solution at EE point, where TLBO was better than NMS-DE.
4.2 Model Simulations

The graphical views of evolutions of subpopulations of the underlying model are presented in Fig. 9-11. One can observe that the optimized solutions found in be EPA scheme for all considered orders of Padé approximants at both equilibrium points converge exactly to steady-state points. This fact reflects that solution by EPA scheme is in good agreement with that of the NSFD method in the vicinity of equilibrium points [62]. It is worth mentioning that the convergence speed of EPA-based solutions is significantly faster than those of NSFD; we may also observe that the model’s physically vital properties like positivity, boundedness, dynamical consistency, and converges to actual steady states (disease-free and endemic equilibria) are also preserved by EPA scheme. Furthermore, the proposed scheme is also independent of the choice of step lengths and unconditionally converges to both states of the model.
Fig. 9. Dynamics of susceptible population at (a) EE (b) DFE states.

Fig. 10. Dynamics of Infected population at (a) EE (b) DFE states.
5. Conclusions

The context of the current study is the evolvement of an evolutionary solver for a complex Covid-19 with a crowding effect (CovidCE) model. The main focus of this research was the formulation and solution of the underlying CovidCE model as an equivalent global optimization problem. This work is a blend of mathematical programming, epidemiology, and metaheuristics. The first contribution relates to the theory of optimization that includes two prominent innovations (i) formulation of a new general order Padé approximation based equivalent global optimization problem preserving vital properties (positivity, boundedness, and stability) of the CovidCE model. This generalization allows varying or adjusting the orders of Padé approximants for approximating the solution more accurately (ii) The formulated optimization problem's complexity barred not only the use of deterministic optimizers but also modern metaheuristics. Therefore, the second contribution of this work was to tune a new hybrid method, named NMS-DE, to solve the formulated problem. This step was accompanied by a statistical analysis of the results of various orders of solutions. The comparative analysis witnessed that the proposed hybrid method handled the optimization phase of the EPA scheme efficiently and outperformed other metaheuristics in terms of accuracy, consistency, and speed of convergence. The Final contribution relates to the characteristics of the obtained solution of the CovidCE model. To the best of our knowledge, the exact analytical solution of such epidemiological models with all physical properties has not been reported so far. The current work enables us to observe an approximate closed-form solution of the CovidCE type model. The graphics of model simulations also reveal that the solution found by the EPA scheme is in good agreement with NSFD in convergence to equilibrium points and possesses a notably faster convergence speed.
CRediT author statement

Javaid Ali: Conceptualization, Methodology, Design Ali Raza and Nauman Ahmed: Data curation, Writing- Original draft preparation. Ali Ahmadian: Supervision and Validation of Data- Reviewing Original draft Muhammad Rafiq and Massimiliano Ferrara: Reviewing and Editing the final version and validation of data.
References

[1] Zeb A, Alzahrani E, Erturk VS, Zaman G. Mathematical model for coronavirus disease 2019 (Covid-19) containing isolation class. BioMed research international 2020; 10:1-7.

[2] Riyapan P, Shuaib SE, Intarasit A. A mathematical model of Covid-19 pandemic: a case study of Bangkok, Thailand. Computational and Mathematical Methods in Medicine 2021; 2021:1-11.

[3] Oud MAA, Ali A, Alrabaiah H, Ullah S, Khan MA, Islam S. A fractional-order mathematical model for Covid-19 dynamics with quarantine, isolation, and environmental viral load. Advances in Difference Equations, 2021; 2021: 1-19.

[4] Shaikh AS, Shaikh IN, Nisar KS. A mathematical model of Covid-19 using fractional derivative: outbreak in India with dynamics of transmission and control. Advances in Difference Equations, 2021; 2020: 1-19.

[5] Ahmed I, Modu GU, Yusuf A, Kumam P, Yusuf I. A mathematical model of coronavirus disease (Covid-19) containing asymptomatic and symptomatic classes. Results in Physics, 2021:1-15, 2021.

[6] Ullah S, Khan MA. Modelling the impact of non-pharmaceutical interventions on the dynamics of novel coronavirus with optimal control analysis with a case study. Chaos, Solitons & Fractals, 2020:1-32, 2020.

[7] Peter OI, Qureshi S, Yusuf A, Al-Shomrani M, Idowu AA. A new mathematical model of Covid-19 using real data from Pakistan. Results in Physics, 2021: 1-10, 2021.

[8] Nazir G, Zeb A, Shah K, Saeed T, Khan RA, Khan SIU. Study of Covid-19 mathematical model of fractional order via modified Euler method. Alexandria Engineering Journal, 60: 5287-5296, 2021.

[9] Kyrychko YN, Blyuss KB, Brovchenko I. Mathematical modelling of the dynamics and containment of Covid-19 in Ukraine. Scientific reports, 10: 1-17, 2020.

[10] Khoshnaw SH, Salih RH. Sulaimany S. Mathematical modelling for coronavirus disease (Covid-19) in predicting future behaviours and sensitivity analysis. Mathematical Modelling of Natural Phenomena, 10:1-13, 2020.
[11] Sasmita NR, Ikhwan M, Suyanto S, Chongsuvivatwong V. Optimal control on a mathematical model to pattern the progression of coronavirus disease 2019 (Covid-19) in Indonesia. *Global Health Research and Policy*, 5: 1-12, 2020.

[12] Tiwari V, Bisht N, Deyal N. Mathematical modelling based study and prediction of Covid-19 epidemic dissemination under the impact of lockdown in India. *medRxiv*, 6:1-16, 2020

[13] Wang S, Tang W, Xiong L, Fang M, Zhang B, Chiu CY, Fan R. Mathematical modelling of transmission dynamics of Covid-19. *Big Data and Information Analytics*, 6:12-25, 2021.

[14] Baek YJ, Lee T, Cho Y, Hyun JH, Kim MH, Sohn Y. Choi JY. A mathematical model of Covid-19 transmission in a tertiary hospital and assessment of the effects of different intervention strategies. *PloS one*, 15:1-10, 2020.

[15] Santaella-Tenorio J, Guerrero R, Bravo L. Mathematical model and Covid-19 modelos matemáticos y el covid-19. *Colombia Medica* (Cali, Colombia), 51: 1-12, 2020

[16] Peter OJ, Shaikh AS, Ibrahim MO, Nisar KS, Baleanu D, Khan I, Abioye AI. Analysis and dynamics of fractional-order mathematical model of Covid-19 in Nigeria using atangana-baleanu operator. *Computers, Materials and Continua*, 66: 1-20, 2020

[17] Moyles IR, Heffernan JM, Kong JD. Cost and social distancing dynamics in a mathematical model of Covid-19 with application to Ontario, Canada. *medRxiv*. 15:1-10, 2020

[18] O. I. Krivorot’ko, S. I. Kabanikhin, N. Y. Zyat’kov, A. Y. Prikhod’ko, N. M. Prokhoshin, M. A. Shishlenin. Mathematical modelling and forecasting of Covid-19 in Moscow and Novosibirsk region. *Numerical Analysis and Applications*, 13: 332-348, 2020.

[19] Harjule P, Tiwari V, Kumar A. Mathematical models to predict Covid-19 outbreak: An interim review. *Journal of Interdisciplinary Mathematics*, 15:1-26, 2021.

[20] Kim BN, Kim E, Lee S, Oh C. Mathematical model of Covid-19 transmission dynamics in South Korea: the impacts of travel restrictions, social distancing, and early detection. *Processes*, 8: 1-10, 2020.

[21] Diaz JEM, Raza A, Ahmed N, Rafiq M. Analysis of a nonstandard computer method to simulate a nonlinear stochastic epidemiological model of coronavirus-like diseases. *Computer Methods and Programs in Biomedicine*, vol. 204, pp. 1-10, 2021.
[22] Raza A, Ahmadian A, Rafiq M, Salahshour S, Laganà IR. An analysis of a nonlinear susceptible-exposed-infected-quarantine-recovered pandemic model of a novel coronavirus with delay effect. *Results in Physics*, vol. 21, 01-07, 2021.

[23] Muhammad Zamir, Kamal Shah, Fawad Nadeem, Mohd Yazid Bajuri, Ali Ahmadian, Soheil Salahshour, Massimiliano Ferrara: Threshold conditions for global stability of disease free state of COVID-19. *Results in Physics*, vol. 21, pp. 103784, (2021).

[24] Vedika Gupta, Nikita Jain, Piyush Katariya, Adarsh Kumar, Senthilkumar Mohan, Ali Ahmadian, Massimiliano Ferrara: An emotion care model using multimodal textual analysis on COVID-19. *Chaos, Solitons & Fractals*, vol. 144, pp. 110708, (2021).

[25] Pritam Saha, Debadyutı Mukherjee, Pawan Kumar Singh, Ali Ahmadian, Massimiliano Ferrara, Ram Sarkar: GraphCovidNet: A graph neural network based model for detecting COVID-19 from CT scans and X-rays of chest. *Scientific Reports*, vol. 11, no. 1, pp. 1-16, (2021).

[26] S Ahmad, A Ullah, K Shah, S Salahshour, A Ahmadian, T Ciano, Fuzzy fractional-order model of the novel coronavirus. *Advances in difference equations*, vol. 2020, pp. 1-17, (2020).

[27] Ahmad N, Elsonbaty A, Raza A, Rafiq M, Adel W. Numerical simulation and stability analysis of a novel reaction-diffusion Covid-19 model. *Nonlinear Dynamics*, 23, pp. 01-18, (2021).

[28] Neha Ghorui, Arijit Ghosh, Sankar Prasad Mondal, Mohd Yazid Bajuri, Ali Ahmadian, Soheil Salahshour, Massimiliano Ferrara, Identification of dominant risk factor involved in spread of COVID-19 using hesitant fuzzy MCDM methodology. *Results in physics*. 21, pp. 103811, (2021).

[29] Mohd Shariq, Karan Singh, Mohd Yazid Bajuri, Athanasios A Pantelous, Ali Ahmadian, Mehdi Salimi. A Secure and reliable RFID authentication protocol using digital schnorr
cryptosystem for IoT-enabled healthcare in COVID-19 scenario. *Sustainable Cities and Society*, 75 (2021). 103354.

[30] Mateescu GD, On the application of genetic algorithms to differential equations. *Romanian Journal of Economic Forecasting*, 2(2006), 5-9.

[31] Lee ZY, Method of bilaterally bounded to solution Blasius equation using particle swarm optimization. *Applied Mathematics and Computation*, 179 (2006) 779–786.

[32] Babaie M, A general approach to approximate solutions of nonlinear differential equations using particle swarm optimization, *Applied Soft Computing*, 13(7) (2013) 3354 – 3365.

[33] Karr CL, Wilson E, A self-tuning evolutionary algorithm applied to an inverse partial differential equation. *Applied Intelligence*, 19 (2003) 147–155.

[34] Cao H, Kang L, Chen Y, Evolutionary modelling of systems of ordinary differential equations with genetic programming. *Genetic Programming and Evolvable Machines*, 1(4) (2000) 309–337.

[35] Mastorakis, Nikos E, Unstable ordinary differential equations: solution via genetic algorithms and the method of Nelder-Mead. *WSEAS TRANSACTIONS on MATHEMATICS*, 5(12) (2006) 1276–1281.

[36] Pageant N, Bureerat S, Solving partial differential equations using a new differential evolution algorithm. *Mathematical Problems in Engineering*, 2014 (2014) No. 747490, p. 10.

[37] Ali J et al., Numerical treatment of nonlinear model of virus propagation in computer networks: an innovative evolutionary Padé approximation scheme. *Advances in Difference Equations*, 2018(1), 214 (2018).

[38] Nisar KS, Ali J, Mahmood MK, Ahmed D, Ali S, Hybrid evolutionary Padé approximation approach for numerical treatment of nonlinear partial differential equations, *Alexandria Engineering Journal*, Volume 60, Issue 5, 2021, Pages 4411-4421.

[39] Baker GA, Essentials of Padé approximants, Academic Press, New York, NY, USA, 1975.

[40] Baker GA, Morris PG, Padé approximants, Addison-Wesley, 1981.

[41] Padè HE, Sur la representation approchée d’une fonction par des fractions rationnelles. *Annales Scientifiques De l'École Normale Supérieure*, 9 (1892) 1-93.
[42] Leal HV, Guerrero F, Application of series method with Padé and Laplace-Padé re-summation methods to solve a model for the evolution of smoking habit in Spain. *Computational & Applied Mathematics*, 33 (1) (2013), pp. 1-12.

[43] Rashidi MM, Keimanesh M, Using differential transform method and Padé approximant for solving MHD flow in a laminar liquid film from a horizontal stretching surface, *Mathematical Problems in Engineering*, 2010 (2010), Article ID 491319, doi.org/10.1155/2010/491319.

[44] Guerrero F, Santonja F, Villanueva R, Solving a model for the evolution of smoking habit in Spain with homotopy analysis method. *Nonlinear Analysis: Real-World Application*, 14(1) (2013) 549–558.

[45] Wang Z, Zou L, Zong Z, Adomian decomposition and Padé approximate for solving differential-difference equation, *Applied Mathematics and Computation*, 218(4) (2011) 1371–1378.

[46] Yang XS. Nature-Inspired Metaheuristic Algorithms. Luniver Press. (2008) ISBN 978-1-905986-10-1.

[47] Rao RV, Savsani VJ, Vakharia DP, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Computer-Aided Design*, Volume 43, Issue 3, March 2011, Pages 303-315.

[48] Kennedy J, Eberhart R, Particle swarm optimization. In: IEEE International Conference on Neural Networks, 1942–1948 (1995).

[49] Storn R, Price K, Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11, 341–359 (1997).

[50] Goldberg DE, Genetic algorithms in search, optimization, and machine learning. Pearson Education, Upper Saddle River (1989).

[51] Luqman M, Saeed M, Ali J, Tabassam MF, Mahmood T, Targeted Showering Optimization: training irrigation tools to solve crop planning problems. *Pakistan Journal of Agricultural Science*, 56 (1), (2019) 225-235.

[52] Ali J, Saeed M, Tabassam MF, Iqbal S, Controlled showering optimization algorithm: an intelligent tool for decision making in global optimization. *Computational and Mathematical Organization Theory*, 25, 132–164 (2019). https://doi.org/10.1007/s10588-019-09293-6.
[53] Eskandar H, Sadollah A, Bahreininejad A, Hamdi M, Water cycle algorithm – a novel metaheuristic optimization method for solving constrained engineering optimization problems. *Computers & Structures*, Volumes 110–111, November 2012, Pages 151-166.

[54] Alatas B, Sports inspired computational intelligence algorithms for global optimization. *Artificial Intelligence Review*, (2017). https://doi.org/10.1007/s10462-017-9587-x

[55] Karaboga D, Basturk B, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39, 459–471 (2007).

[56] Mirjalili S, Mirjalili SM, Lewis A, Grey wolf optimizer. *Advances in Engineering Software*, Volume 69, March 2014, Pages 46-61.

[57] Alexandros T, Georgios D, Nature-inspired optimization algorithms related to physical phenomena and laws of science: a survey. *International Journal of Artificial Intelligence Tools*, 26, 750022 (2017).

[58] Wolpert DH, Macready WG, No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, Vol. 1, Issue 1, (1997), pp. 67-82.

[59] Price KV, Storn RM, Lampinen JA (eds.), *Differential Evolution a Practical Approach to Global Optimization*, Natural Computing Series, pp. 479–498. Springer, Berlin (2005).

[60] Ali J, Saeed M, Chaudhry NA, Tabassum MF, Luqman M, Low cost-efficient remedial strategy for stagnated Nelder-Mead simplex method. *Pakistan Journal of Science*, 69(1) (2017) 119–126.

[61] Nelder JA, Mead R, A simple method for function minimization. *The Computer Journal*, 7 (4) (1965) 308–313.

[62] Mickens RE, A fundamental principle for constructing nonstandard finite difference schemes for differential equations. *Journal of Difference Equations and Applications*, 11(7), (2005) 645-653.