Advances in Difference Equations
a SpringerOpen Journal

RESEARCH Open Access

Numerical treatment of nonlinear model of virus propagation in computer networks: an innovative evolutionary Padé approximation scheme

Javaid Ali[1*], Muhammad Saeed[1], Muhammad Rafiq[2] and Shaukat Iqbal[3]

*Correspondence:
javaidaliwaseer@gmail.com
[1]Department of Mathematics, University of Management and Technology, Lahore, Pakistan
Full list of author information is available at the end of the article

Abstract

This work proposes a novel mesh free evolutionary Padé approximation (EPA) framework for obtaining closed form numerical solution of a nonlinear dynamical continuous model of virus propagation in computer networks. The proposed computational architecture of EPA scheme assimilates a Padé approximation to transform the underlying nonlinear model to an equivalent optimization problem. Initial conditions, dynamical positivity and boundedness are dealt with as problem constraints and are handled through penalty function approach. Differential evolution is employed to obtain closed form numerical solution of the model by solving the developed optimization problem. The numerical results of EPA are compared with finite difference schemes like fourth order Runge–Kutta (RK-4), ODE45 and Euler methods. Contrary to these standard methods, the proposed EPA scheme is independent of the choice of step lengths and unconditionally converges to true steady state points. An error analysis based on residuals witnesses that the convergence speed of EPA is higher than a globally convergent non-standard finite difference (NSFD) scheme for smaller as well as larger time steps.

Keywords: Nonlinear model; Evolutionary computing; Padé approximation; Optimization

1 Introduction

A computer virus is a malicious code which executes harmful and unauthorized activities like erasing necessary files, accessing confidential data and personal information like passwords, account numbers, contact lists etc. Depending on the way of propagation, functioning and damaging the systems/users, malwares are classified into various categories. These include computational viruses, computer worms, Trojans, Rootkits, spyware, logic bombs and so on [1, 2]. Dissemination of computer viruses to other connected systems bears a high resemblance to the behavior of biological viruses [3–5]; therefore, various models of computer virus propagation have been proposed by using an epidemiological analog [6–9].

Transformation of the dynamics of computer virus propagation into mathematical language is an effective methodology to understand and analyze the spreading behavior of viruses. The mathematical prototypes related to the characteristics of variables, parameters and the functional relations governing the dynamics of the virus propagation classify the model as deterministic, stochastic, continuous, discrete, global or individual [2]. Over the years several compartmental models have been evolved. These involve susceptible-infected-susceptible (SIS) models [10, 11], susceptible-infected-recovered (SIR) [12–14], susceptible-infected-recovered-susceptible (SIRS) [15, 16], susceptible-exposed-infected-recovered (SEIR) [17, 18], SEIQS (quarantined class included) [19], SEIRS [20, 21], SEIQRS [22], SEIRS-V (including vaccinated subpopulation) [23] models and so on.

Mathematical tool-based numerical analysis of these epidemiological models is an essential part of investigations for acquiring better knowledge of their evolution, impact and the deriving mechanisms, especially when the analytical solution is not available. A profound understanding of the model helps in imposing precautionary measures and evaluating their effectiveness in preventing the networks from infections.

The studies conducted by Rafiq et al. [24] on a nonlinear model of virus propagation in a computer network proposed by Mei Peng [25] have exposed the divergence behaviors of RK-4 and Euler methods for certain step lengths. The similar behaviors of RK-4 and Euler schemes have also been highlighted in [26] and [27]. In these studies a globally convergent non-standard finite difference (NSFD) scheme proposed by Mickens [28] has successfully been applied to the said model. To the best of the author's knowledge, till date, the nonlinear model of virus propagation in a computer network has no analytical solution. The important factors which require further research are: (i) finding an analytical solution or constructing alternative efficient convergent numerical schemes, (ii) analysis of the convergence speeds, and (iii) the error analyses of numerical schemes, at least in the common cases when they all converge.

Over the recent years, many modern metaheuristics have been proposed to cope with the most sophisticated problems by transforming them to optimization problems. Metaheuristic algorithms are inspired by natural phenomena like evolution [29, 30], swarm behaviors [31, 32], food foraging behavior [33, 34], sport strategies [35], water dynamics [36, 37] etc. For more detailed studies one may consult the survey articles [35, 38]. Metaheuristics-based approaches of solving differential equations belong to the class of non-standard mesh free methods. The applications of these heuristics to differential equations can be found in [39–43], but their applications to epidemic models are very rare.

The present work presents an innovative numerical scheme by integrating evolutionary computing and Padé rational functions [44–46] for numerical treatment of the model. The proposed computational framework involves following novel aspects:

(i) Construction of an equivalent optimization problem by exploiting interpolation and extrapolation strengths of Padé approximation.
(ii) Preservation of positivity, boundedness and initial conditions agreement by defining problem constraints.
(iii) Construction of fitness/objective function, an essential requirement for evolutionary computing, by use of penalty function approach.
(iv) Implementation of differential evolution (DE) to optimize the constructed fitness function.
(v) Evolvement of unconditionally convergent closed form numerical solution of nonlinear model of virus propagation in computer networks.

The developed method is named the evolutionary Padé approximation (EPA) scheme.

The rest of the paper is organized as follows. In Sect. 2, relative basic concepts are revisited. Section 2 elaborates the proposed framework of EPA scheme for numerical treatment of nonlinear dynamical continuous model of virus propagation in computer networks. In Sect. 5, analyses of the results are presented. In the end, conclusion and some future directions are given.

2 Related concepts

2.1 Padé approximation

The idea of a Padé approximation was formulated at the end of the 19th century within the classical theory of continued fractions [44]. The Padé approximation of order (N, M) is a rational function of the form [45]

$$P_{N,M}(t) = \frac{\sum_{i=0}^{N} a_i t^i}{\sum_{j=0}^{M} b_j t^j}.$$

The polynomials $\sum_{i=0}^{N} a_i t^i$ and $\sum_{j=0}^{M} b_j t^j$ are called Padé approximants. Normalizing it by b_0 ($\neq 0$) the following form is obtained:

$$P_{N,M}(t) = \frac{\sum_{i=0}^{N} a_i t^i}{1 + \sum_{j=1}^{M} b_j t^j}.$$

It involves $(N + M + 1)$ unknown coefficients which are to be determined in such a way that the Maclaurin series expansions of $P_{N,M}(t)$ coincides with some target function as far as possible [46].

2.2 Differential evolution: the evolutionary algorithm

In evolutionary computing, differential evolution (DE) [30] is a competent population-based stochastic global search method. Initialization, mutation, recombination and selection are the main operations of a DE algorithm. In mutation, a trial solution $\boldsymbol{v}$ for improving $\boldsymbol{x}$ is generated with the help of the other two mutually different solutions $\boldsymbol{y}$ and $\boldsymbol{z}$:

$$v_m = x_m + F \times (y_m - z_m), \quad \text{for each } m = 1, 2, 3, \ldots, \text{Dimension}.$$

Here F is a positive real number acting as an algorithmic constant.

In recombination, the coordinates of $\boldsymbol{v}$ are combined with those of $\boldsymbol{x}$ probabilistically. A predefined positive real number CR is used as recombination probability and a random number $rand \in (0, 1)$ is generated randomly for each coordinate m. Recombination takes place according to

$$u_m = \begin{cases} v_m & \text{if } rand < CR, \\ x_m & \text{otherwise}. \end{cases}$$

In the selection step the solution $\boldsymbol{u}$ is assigned to $\boldsymbol{x}$ if it appears to be better than $\boldsymbol{x}$. The iterative process of a DE algorithm carries on trying to improve until some termination criteria are met. Upon termination, DE returns the best member of the population as an optimal solution.

2.3 Penalty function

Evolutionary algorithms like DE are generally designed for unconstrained optimization problems and there is very rare existence of unconstrained real world problems. Therefore, transforming the constrained optimization problem to an unconstrained through suitable approach is inevitable. Penalty functions belong to the most effective methods to handle constraints [47, 48]. A penalty function allows for accepting the feasible solution and penalizes an infeasible solution by adding a sufficiently large positive number to the objective function depending on degree of violation of constraints. For an objective function $\psi(\boldsymbol{x})$ and a solution $\boldsymbol{x}$, the penalty function $\mathcal{H}(\boldsymbol{x})$ defines an unconstrained penalized function $\varphi(\boldsymbol{x})$ by the following relation:

$$\varphi(\boldsymbol{x}) = \begin{cases} \psi(\boldsymbol{x}) + \mathcal{H}(\boldsymbol{x}) & \text{if } \boldsymbol{x} \text{ is infeasible,} \\ \psi(\boldsymbol{x}) & \text{if } \boldsymbol{x} \text{ is feasible.} \end{cases}$$

For a minimization problem, $\mathcal{H}(\boldsymbol{x}) \geq 0$ whereas for a maximization problem $\mathcal{H}(\boldsymbol{x}) \leq 0$.

3 Mathematical model of virus propagation in computer networks

The considered SEIR model of virus transmission in a computer network that was proposed by Mei Peng in [25] and referred to in [26] is described by Fig. 1.

At any time 't' the state variables of the model are defined by:

$S(t)$: susceptible computers; $E(t)$: exposed computers; $I(t)$: infected computers; $R(t)$: recovered computers.

The model parameters are:

N: The total population of computers in the network.
p: Rate at which antivirus recovers susceptible computers.
k: Rate at which antivirus recovers exposed computers.
a: Rate at which antivirus cannot cure exposed computers.
β_1: Contact rate of susceptible with infected computers.
β_2: Contact rate of susceptible with exposed computers.
μ: Removal rate of a computer from the network.
r: Recovery rate of infected computers that are cured.

The system of governing equations for the model is given as under:

$$\left.\begin{aligned} &S'(t) = (1-p)N - \beta_1 S(t)I(t) - \beta_2 S(t)E(t) - pS(t) - \mu S(t), \\ &E'(t) = \beta_1 S(t)I(t) + \beta_2 S(t)E(t) - kE(t) - \alpha E(t) - \mu E(t), \\ &I'(t) = \alpha E(t) - rI(t) - \mu I(t), \\ &R'(t) = pS(t) + kE(t) + rI(t). \end{aligned}\right\} \tag{1}$$

Here

$$S(t) + E(t) + I(t) + R(t) = N(t). \tag{2}$$

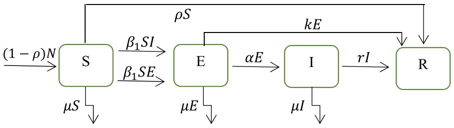

Figure 1 SEIR model for virus propagation in a computer network

Table 1 Values of the model parameters for two points of equilibriums

Equilibrium point	Model parameters							
	α	β_1	β_1	N	p	r	k	μ
VE	0.01	0.7	0.8	100	0.5	0.6	0.4	0.02
VFE	0.01	0.002	0.003	100	0.5	0.6	0.4	0.02

Suppose that

$$\begin{aligned}
&X_1(t) = A - \beta_1 S(t)I(t) - \beta_2 S(t)E(t) - aS(t),\\
&X_2(t) = \beta_1 S(t)I(t) + \beta_2 S(t)E(t) - bE(t),\\
&X_3(t) = \alpha E(t) + cI(t), \quad A = (1-p)N, a = p+\mu, b = k+\alpha+\mu, c = r+\mu.
\end{aligned}$$

Then, by (2) and the above suppositions, model (1) can be reduced to the following form:

$$\left.\begin{aligned}
S'(t) &= X_1(t),\\
E'(t) &= X_2(t),\\
I'(t) &= X_3(t).
\end{aligned}\right\} \tag{3}$$

The initial conditions are

$$S_0 = S(0) = 50; \qquad E_0 = E(0) = 40; \qquad I_0 = I(0) = 20.$$

The basic reproductive number defined as the average number of secondary infections produced by a primary infection can be identified:

$$R_o = \frac{A(\alpha\beta_1 + c\beta_2)}{abc}.$$

For $R_o > 1$ the virus equilibrium (VE) point is found:

$$\left(S^0, E^0, I^0\right) = \left(\frac{A}{aR_o}, \frac{A(R_o - 1)}{bR_o}, \frac{A\alpha(R_o - 1)}{bcR_o}\right).$$

For $R_o < 1$ the following virus free equilibrium (VFE) point is identified:

$$\left(S^*, E^*, I^*\right) = \left(\frac{A}{a}, 0, 0\right).$$

The model related parameters used in [24, 25] are exhibited in Table 1.

4 Evolutionary Padé approximation numerical (EPA) scheme

The architecture of the proposed evolutionary computing-based Padé approximation scheme involves the four main steps which are presented in the following.

4.1 Construct residual functional based on Padé approximation

Suppose that $S(t), E(t)$ and $I(t)$ are approximated by Padé rational functions according to

$$S(t) \approx \frac{\sum_{i=0}^{N} a_i t^i}{1 + \sum_{j=1}^{M} b_j t^j}; \qquad E(t) \approx \frac{\sum_{i=0}^{N} c_i t^i}{1 + \sum_{j=1}^{M} d_j t^j}; \qquad I(t) \approx \frac{\sum_{i=0}^{N} e_i t^i}{1 + \sum_{j=1}^{M} f_j t^j}.$$

Imposing initial conditions $S(t_0) = S_0$, $E(t_0) = E_0$, $I(t_0) = I_0$, we obtain

$$a_0 = S_0, \qquad c_0 = E_0, \qquad e_0 = I_0. \tag{4}$$

For each of the discrete time steps $t_q = t_0 + qh$; $q = 0, 1, 2, 3, \ldots, q_{\max}$, the above system (3) reduces to the following form:

$$\left.\begin{aligned} \varepsilon_1(t_q) = 0, \\ \varepsilon_2(t_q) = 0, \\ \varepsilon_3(t_q) = 0. \end{aligned}\right\} \tag{5}$$

Here ε_1, ε_2 and ε_3 are the residuals defined by

$$\begin{aligned} \varepsilon_1(t_q) = {} & \left(1 + \sum_{j=1}^{M} b_j t_q^j\right)\left(\sum_{i=0}^{N} i a_i t_q^{i-1}\right) - \left(\sum_{i=0}^{N} a_i t_q^i\right)\left(\sum_{j=1}^{M} j b_j t_q^{j-1}\right) \\ & - X_1(t_q)\left(1 + \sum_{j=1}^{M} b_j t_q^j\right)^2, \end{aligned} \tag{6}$$

$$\begin{aligned} \varepsilon_2(t_q) = {} & \left(1 + \sum_{j=1}^{M} d_j t_q^j\right)\left(\sum_{i=0}^{N} i c_i t_q^{i-1}\right) - \left(\sum_{i=0}^{N} c_i t_q^i\right)\left(\sum_{j=1}^{M} j d_j t_q^{j-1}\right) \\ & - X_2(t_q)\left(1 + \sum_{j=1}^{M} d_j t_q^j\right)^2, \end{aligned} \tag{7}$$

$$\begin{aligned} \varepsilon_3(t_q) = {} & \left(1 + \sum_{j=1}^{M} f_j t_q^j\right)\left(\sum_{i=0}^{N} i e_i t_q^{i-1}\right) - \left(\sum_{i=0}^{N} e_i t_q^i\right)\left(\sum_{j=1}^{M} j f_j t_q^{j-1}\right) \\ & - X_3(t_q)\left(1 + \sum_{j=1}^{M} f_j t_q^j\right)^2. \end{aligned} \tag{8}$$

The problem reduces to the problem of finding $3(M + N)$ coefficients of Padé approximants by solving system (5) having $3q_{\max}$ nonlinear simultaneous equations. System (5) is highly nonlinear and may possess many solutions in general. By assuming $\boldsymbol{x} = (a_1, a_2, \ldots, a_M, b_1, b_2, \ldots, b_N, c_1, c_2, \ldots, c_M, d_1, d_2, \ldots, d_N, e_1, e_2, \ldots, e_M, f_1, f_2, \ldots, f_N)^T \in \mathbb{R}^{3(M+N)}$, the system (5) is further converted to a minimization problem of the form:

$$\text{Minimize } \psi(\boldsymbol{x}) = \frac{1}{3} \sum_{\nu=1}^{3} \sum_{q=0}^{q_{\max}} \left[\varepsilon_\nu(t_q)\right]^2. \tag{9}$$

4.2 Formation of problem constraints

The initial conditions (4) of the model are considered as equality constraints:

$$h_1(t) = S(t) - S_0 = 0, \tag{10}$$

$$h_2(t) = E(t) - E_0 = 0, \tag{11}$$

$$h_3(t) = I(t) - I_0 = 0. \tag{12}$$

The inequality constraints (13) to (15) are related to positivity whereas (16) incorporates the boundedness of the numerical solution.

$$g_{1q} = \frac{\sum_{i=0}^{N} a_i t_q^i}{1 + \sum_{j=1}^{M} b_j t_q^j} \geq 0, \tag{13}$$

$$g_{2q} = \frac{\sum_{i=0}^{N} c_i t_q^i}{1 + \sum_{j=1}^{M} d_j t_q^j} \geq 0, \tag{14}$$

$$g_{3q} = \frac{\sum_{i=0}^{N} e_i t_q^i}{1 + \sum_{j=1}^{M} f_j t_q^j} \geq 0, \tag{15}$$

$$g_{1q} + g_{2q} + g_{3q} \leq \frac{A}{a}. \tag{16}$$

4.3 Employing penalty function approach

An equivalent unconstrained minimization model is obtained by using the following penalty function approach:

$$\mathcal{H}(\boldsymbol{x}) = \sum_{q=1}^{q_{\max}} L_q \times \max\left\{0, (h_1)^2, (h_2)^2, (h_3)^2, -g_{1q}, -g_{2q}, -g_{3q}, \sum_{s=1}^{3} g_{sq} - \frac{A}{a}\right\}.$$

The scalar $L_q$is a large positive real number acting as a penalty factor at qth discrete time step. The unconstrained objective function is

$$\text{Minimize } \varphi(\boldsymbol{x}) = \psi(\boldsymbol{x}) + \mathcal{H}(\boldsymbol{x}). \tag{17}$$

4.4 Use of differential evolution for optimization process

To optimize objective function (17) the iterative steps of DE are executed as follows:

1. Generate a population of K solutions ($\boldsymbol{x}_j \in \mathbb{R}^{3(M+N)}$; $1 \leq j \leq K$) randomly.
2. Evaluate the fitness $\varphi_j = \varphi(\boldsymbol{x}_j)$ of each solution. Preserve the best solution with the smallest objective function value. Set$T = 0$.
3. Set $T = T + 1$.
4. For each of $j = 1, 2, 3, \dots, K$, choose three distinct solutions $\boldsymbol{x}_A$, $\boldsymbol{x}_B$ and $\boldsymbol{x}_C$ from the population excluding $\boldsymbol{x}_j$. Set $\boldsymbol{y} = \boldsymbol{x}_j$.
5. For each of the dimensions $i = 1, 2, 3, \dots, 3(M + N)$, alter the ith coordinate according to

$$y_i = \begin{cases} x_{Ai} + F \times (x_{Bi} - x_{Ci}) & \text{if } rand < CR, \\ x_{ji} & \text{otherwise.} \end{cases}$$

6. If $\varphi(\boldsymbol{y}) < \varphi_j$ then $\boldsymbol{x}_j \leftarrow \boldsymbol{y}$, otherwise discard $\boldsymbol{y}$.
7. Update the best solution.
8. If $T >$ Number of Allowed Iterations, then terminate by preserving the best solution, otherwise start next iteration from step 3.

In step 5 the symbol *rand* denotes a random number in the interval $(0, 1)$, F is a differential constant and CR is crossover fraction.

Table 2 Optimized coefficients of Padé approximate solutions

	i	(sign(coefficient)) × log(|coefficient|)					
		a_i	b_i	c_i	d_i	e_i	f_i
VE	1	20.5658	20.3369	19.696	15.817	–19.0657	–19.3137
	2	21.019	20.7901	18.1203	14.2412	–18.6749	–18.923
VFE	1	36.4774	31.9115	–13.0156	–41.8388	–14.5164	–41.3819
	2	–38.6245	–34.0586	–6.66374	–43.6549	–5.88668	–44.8167

5 Numerical results

Four parameters of DE algorithm have been set: population size = 50; $CR = 0.9$; $F = 0.5$ and the maximum number of iterations = 2000. The order of the Padé approximation is set as $(N, M) = (2, 2)$. The parameter $q_{\max}$ is set to be 2000. The value of each penalty factor is set to be $L_q = 10^{10}$ for all q.

The optimized coefficients of Padé approximants returned by DE optimizer are given in Table 2.

5.1 Convergence analysis

This section presents the unconditional convergence of numerical solution found by the proposed method. Let the optimized coefficients of Padé approximation be denoted by a_i^0, b_i^0, c_i^0, d_i^0, e_i^0, f_i^0 for VE equilibrium point and a_i^*, b_i^*, c_i^*, d_i^*, e_i^*, f_i^* for VFE steady state point. Then for the VE point

$$\lim_{t\to\infty} S(t) \approx \lim_{n\to\infty} S_{N,M}(t) = \begin{cases} \frac{a_N^0}{b_M^0} & \text{if } N = M, \\ 0 & \text{if } N < M, \\ \infty & \text{if } N > M. \end{cases}$$

Setting $N = M = 2$ and applying the reverse operation on optimized coefficients, we get

$$\lim_{t\to\infty} S(t) \approx \frac{a_N^0}{b_M^0} = \frac{-4{,}731{,}817{,}317{,}514{,}237}{-3{,}763{,}605{,}488{,}155{,}694} \approx 1.2573 = S^0.$$

Similarly,

$$\lim_{t\to\infty} E(t) \approx \frac{c_N^0}{d_M^0} = \frac{-5{,}018{,}532{,}630{,}836{,}860}{-10{,}373{,}441{,}962{,}128.1} \approx 48.378653568783371 \approx E^0,$$

$$\lim_{t\to\infty} I(t) \approx \frac{e_N^0}{f_M^0} = \frac{-2{,}180{,}339{,}159{,}320{,}262}{-2{,}794{,}228{,}815{,}931{,}405} \approx 0.780300864012630 \approx I^0.$$

For the VFE point

$$\lim_{t\to\infty} S(t) \approx \frac{a_N^*}{b_M^*} = \frac{-3.457895266178034\text{E}16}{-359{,}621{,}107{,}682{,}515.6} \approx 96.153846153846132 \approx S^*.$$

Similarly,

$$\lim_{t\to\infty} E(t) \approx \frac{c_N^*}{d_M^*} \approx 2.2653\text{E}{-17} \approx E^*,$$

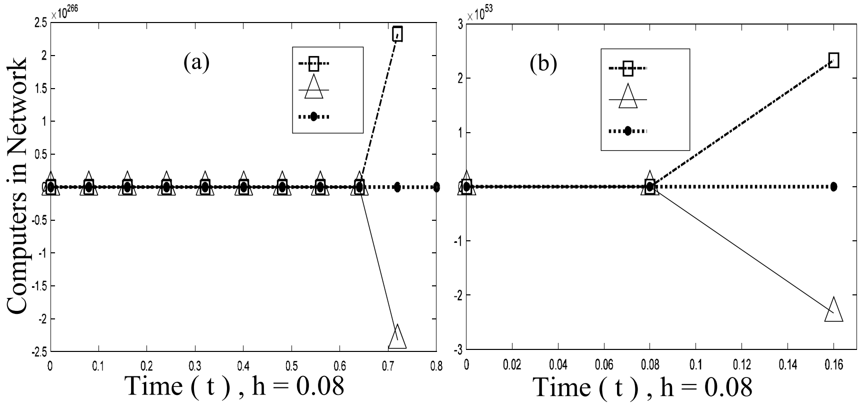

Figure 2 Divergence behaviors of (**a**) Euler method, (**b**) Rk-4 method for VE and $h = 0.08$

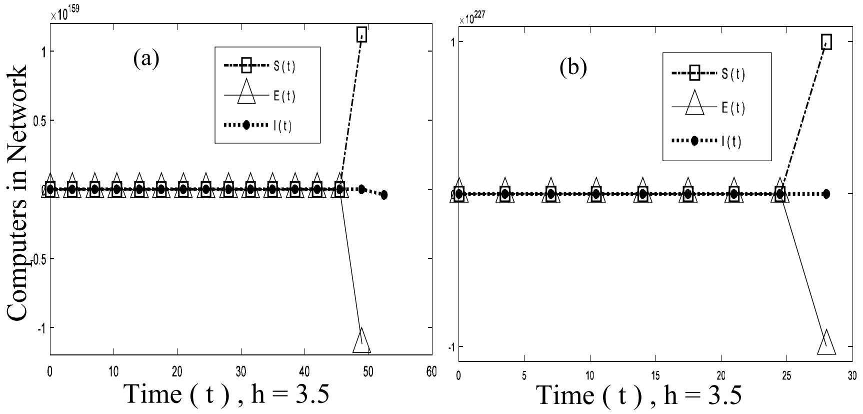

Figure 3 Divergence behaviors of (**a**) Euler method, (**b**) RK-4 method for VFE with $h = 3.5$

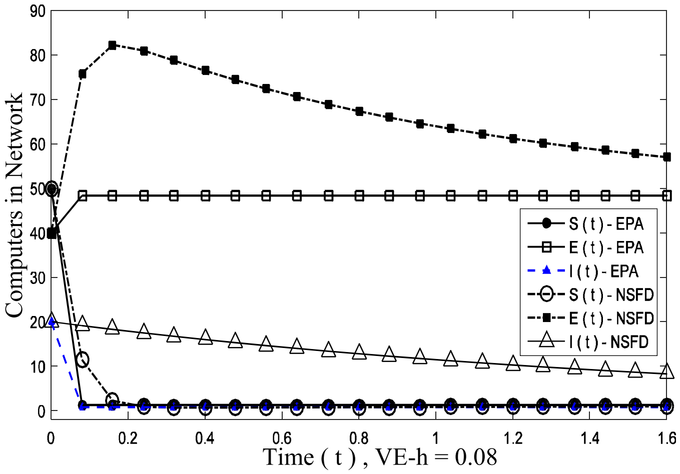

Figure 4 Convergence curves of EPA and NSFDE to VE point for $h = 0.08$

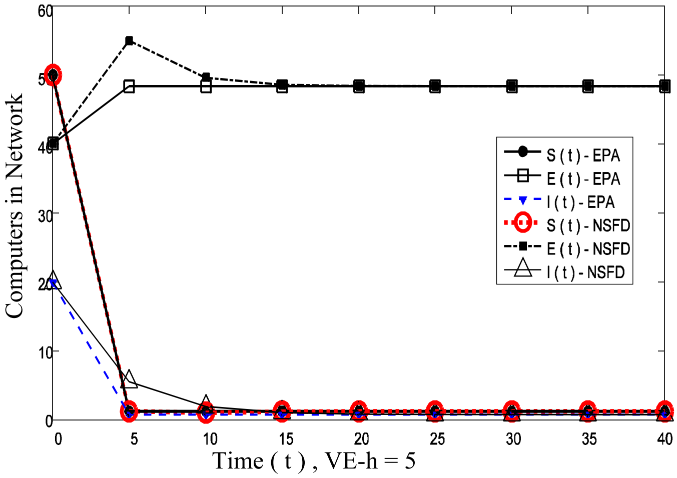

Figure 5 Convergence curves of EPA and NSFDE to VE point for $h = 5$

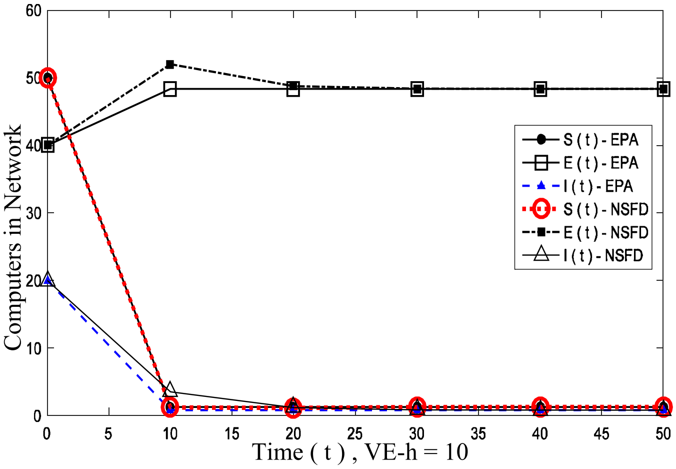

Figure 6 Convergence curves of EPA and NSFD to VE point for $h = 10$

$$\lim_{t\to\infty} I(t) \approx \frac{e_N^*}{f_M^*} \approx 1.01293\text{E}{-}17 \approx I^*.$$

This proves that the closed form numerical solution found by the proposed EPA approach unconditionally converges to the steady state points.

From Figs. 2 and 3 the divergence behaviors of EULER and RK-4 methods can be observed for $h = 0.08$ and $h = 3.5$.

On the other hand, from Figs. 4–10, it can observed that EPA is quickly convergent and is in good agreement with NSFD for both of the equilibrium points and the considered values of step length h.

5.2 Error analysis

To describe the dynamics of system (1) accurately, a necessary condition for a numerical solution is to satisfy the system (1) for all of the time steps. This section presents the error analysis by evaluating absolute residuals ((6) to (8)) of the numerical solutions found by

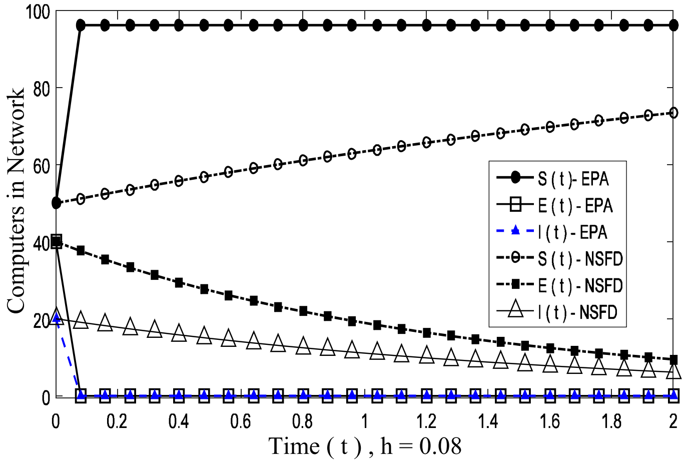

Figure 7 Convergence curves of EPA and NSFD to VFE with $h = 0.08$

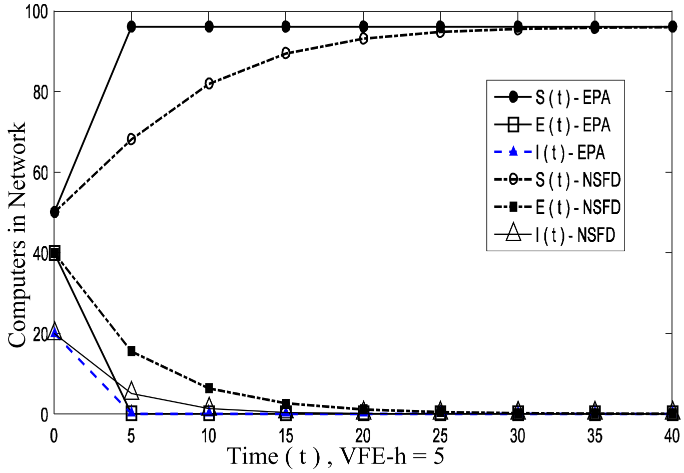

Figure 8 Convergence curves of EPA and NSFD to VFE point at $h = 5$

EPA, RK-4, Euler and NSFD. The derivatives of the EPA solution are calculated analytically whereas a forward difference scheme is used to approximate the derivatives of numerical solutions of EU, RK-4 and NSFD.

Table 3 and Table 4 present the absolute residuals of numerical solutions at 200 time steps for the $h = 0.01$ and $h = 0.1$, respectively.

It can be observed from Tables 3 and 4 that the numerical solutions found by EPA scheme for both steady state points satisfy the governing equations with high accuracies as compared to RK-4, Euler and NSFD schemes. The convergence of RK-4, Euler and NSFD schemes occurs at least after 200 time steps.

Since for higher values of step lengths ($h = 3.5, 5, 10, 36$) RK-4 and Euler methods diverge so the comparisons of absolute residuals for EPA and NSFD are demonstrated graphically in Figs. 10–13.

The proposed EPA scheme is also compared with an optimized version of Matlab ODE45 algorithm. Figures 14 and 15 present the convergence curves of ODE45 for $E(t)$

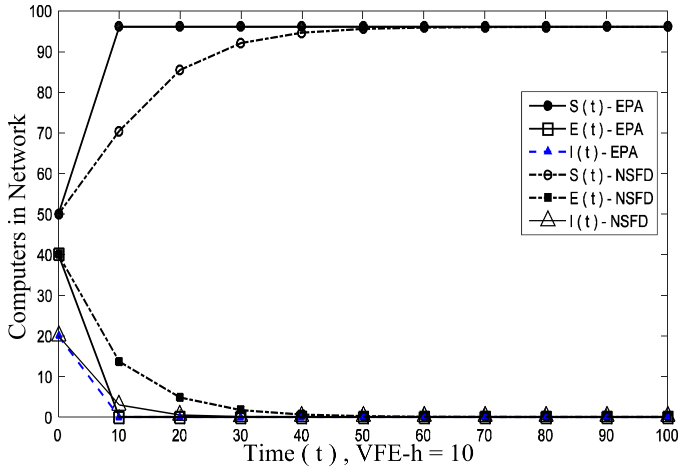

Figure 9 Convergence curves of EPA and NSFD to VFE point at $h = 10$

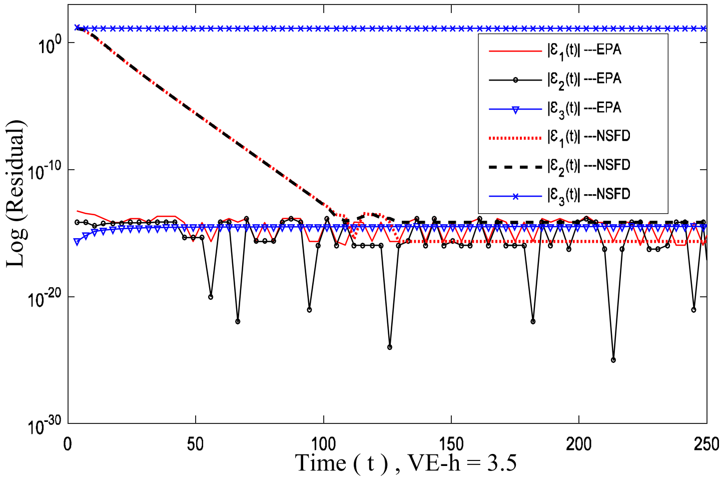

Figure 10 Absolute residuals of EPA and NSFD solutions for VE at $h = 3.5$

and $I(t)$ at VE and VFE points, respectively. Positivity of a numerical solution is an essential property of epidemiological dynamical models as negative values of state variables do not have any physical interpretation. It can be observed from these figures that ODE45 converges to the steady states but does not possess positivity of the numerical solutions.

Figure 16 presents the comparisons of absolute residuals of solutions of ODE45 and EPA method for $h = 0.008$ at VE. It demonstrates that the residuals of solutions found by EPA are very small as compared to those of ODE45 up to more than 3700 time steps.

Figure 17 exhibits the convergence curves of state variables for VE and VFE steady states. One can observe from Fig. 17 that the numerical solutions determined by ODE45 at VE steady state also converge wrongly to VFE equilibrium point.

6 Conclusion

This work proposed an evolutionary computing-based framework of Padé approximation of numerical solution of a nonlinear dynamical continuous model of virus propagation in

Table 3 Absolute residuals of numerical solutions for $h = 0.01$

Method	t	VE			VFE		
		$\lvert\varepsilon_1\rvert$	$\lvert\varepsilon_2\rvert$	$\lvert\varepsilon_3\rvert$	$\lvert\varepsilon_1\rvert$	$\lvert\varepsilon_2\rvert$	$\lvert\varepsilon_3\rvert$
RK-4	0.01	3.7E02	3.7E02	5.5E03	2.4E–02	1.5E–01	2.0E03
	0.40	1.5E–04	1.4E–01	6.0E03	2.1E–02	1.0E–01	1.3E03
	0.80	3.3E–04	9.5E–02	5.4E03	1.9E–02	6.9E–02	8.8E03
	1.20	4.3E–04	6.3E–02	5.0E03	1.7E–02	4.7E–02	5.9E02
	1.60	4.6E–04	4.2E–02	4.8E03	1.6E–02	3.3E–02	4.1E02
	1.99	4.3E–04	2.8E–04	4.7E03	1.6E–02	2.4E–02	2.9E02
Euler	0.01	2.3E–13	6.8E–13	6.0E03	3.2E–13	2.2E–13	2.0E03
	0.40	1.7E–15	1.4E–13	6.0E03	2.4E–13	1.1E–13	1.3E03
	0.80	3.6E–15	4.3E–13	5.4E03	2.9E–13	3.2E–14	8.8E02
	1.20	3.9E–15	3.3E–14	5.0E03	7.8E–14	3.6E–14	5.9E02
	1.60	1.9E–15	3.0E–13	4.8E03	4.5E–13	5.7E–14	4.0E02
	1.99	5.6E–16	5.0E–14	4.7E03	1.7E–13	7.9E–14	2.8E02
NSFD	0.01	7.3E02	7.3E02	4.8E03	1.1E–01	3.5E–01	1.9E03
	0.40	2.0E–01	4.8E–01	6.0E03	9.0E–02	2.5E–01	1.3E03
	0.80	1.6E–01	3.5E–01	5.4E03	7.7E–02	1.8E–01	8.9E02
	1.20	1.2E–01	2.5E–01	5.5E03	6.5E–02	1.3E–01	6.0E02
	1.60	9.6E–02	1.8E–01	4.8E03	5.6E–02	9.1E–02	4.1E02
	1.99	7.3E–02	1.3E–01	4.7E03	4.8E–02	6.7E–02	2.9E02
EPA	0.01	**2.2E–08**	**9.5E–08**	**1.3E–08**	**1.9E–09**	**1.2E–11**	**2.1E–10**
	0.40	**1.5E–12**	**1.2E–12**	**6.6E–14**	**9.5E–13**	**7.1E–14**	**9.3E–14**
	0.80	**2.3E–13**	**2.1E–14**	**2.8E–14**	**6.5E–13**	**1.3E–14**	**3.5E–13**
	1.20	**1.8E–14**	**5.7E–14**	**1.1E–14**	**7.8E–14**	**2.1E–15**	**1.9E–15**
	1.60	**4.6E–14**	**2.8E–14**	**4.4E–15**	**7.1E–15**	**2.2E–15**	**6.1E–17**
	1.99	**5.8E–14**	**4.3E–14**	**1.7E–15**	**1.4E–14**	**3.8E–15**	**1.1E–15**

Table 4 Absolute residuals for $h = 0.1$

Method	t	VE			VFE		
		$\lvert\varepsilon_1\rvert$	$\lvert\varepsilon_2\rvert$	$\lvert\varepsilon_3\rvert$	$\lvert\varepsilon_1\rvert$	$\lvert\varepsilon_2\rvert$	$\lvert\varepsilon_3\rvert$
RK–4	0.10	5.9E83	5.9E83	5.9E83	2.3E–01	1.3E00	1.6E02
	4.00	NaN	NaN	NaN	8.0E–02	4.8E–02	5.5E00
	8.00	NaN	NaN	NaN	1.7E–02	3.6E–03	5.0E–01
	12.0	NaN	NaN	NaN	2.8E–03	3.4E–04	5.3E–02
	16.0	NaN	NaN	NaN	4.1E–04	3.1E–05	5.2E–03
	19.9	NaN	NaN	NaN	5.9E–05	3.04E–06	5.3E–04
Euler	0.10	0.0E00	0.0E00	3.8E03	2.1E–14	3.6E–15	1.6E02
	4.00	NaN	NaN	NaN	2.8E–14	4.4E–16	4.5E00
	8.00	NaN	NaN	NaN	2.8E–14	6.9E–17	4.1E–01
	12.0	NaN	NaN	NaN	3.6E–14	0.0E00	4.1E–02
	16.0	NaN	NaN	NaN	2.1E–14	2.2E–19	3.8E–03
	19.9	NaN	NaN	NaN	7.1E–14	2.7E–20	3.6E–04
NSFD	0.10	6.0E02	6.0E02	6.4E02	9.8E–01	3.0E00	1.7E02
	4.00	1.8E–01	2.6E–01	4.7E02	2.2E–01	1.7E–01	7.9E00
	8.00	1.1E–02	1.2E–02	4.7E02	4.3E–02	1.5E–02	8.5E–02
	12.0	8.5E–04	8.6E–03	4.8E02	7.1E–03	1.6E–03	9.8E–02
	16.0	7.4E–05	7.3E–05	4.8E02	1.1E–03	1.6E–04	1.1E–02
	19.9	7.1E–06	6.9E–06	4.8E02	1.7E–04	1.7E–05	1.7E–05
EPA	0.10	**1.5E–11**	**1.3E–10**	**8.5E–12**	**1.9E–11**	**2.7E–13**	**2.0E–12**
	4.00	**4.9E–14**	**1.4E–14**	**1.7E–16**	**2.1E–14**	**4.3E–15**	**1.8E–15**
	8.00	**2.8E–14**	**7.1E–15**	**8.6E–16**	**1.8E–14**	**2.9E–15**	**1.2E–15**
	12.0	**2.8E–14**	**3.6E–15**	**1.6E–15**	**1.1E–14**	**2.1E–15**	**8.6E–16**
	16.0	**6.8E–15**	**5.3E–15**	**1.9E–15**	**1.8E–14**	**1.6E–15**	**6.7E–16**
	19.9	**2.8E–14**	**8.0E–15**	**2.3E–15**	**7.1E–15**	**1.3E–15**	**5.5E–16**

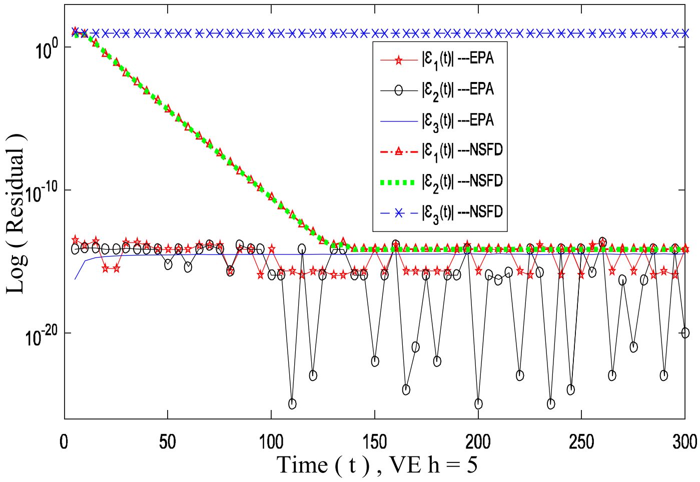

Figure 11 Absolute residuals of EPA and NSFD solutions at VE for *h* = 5

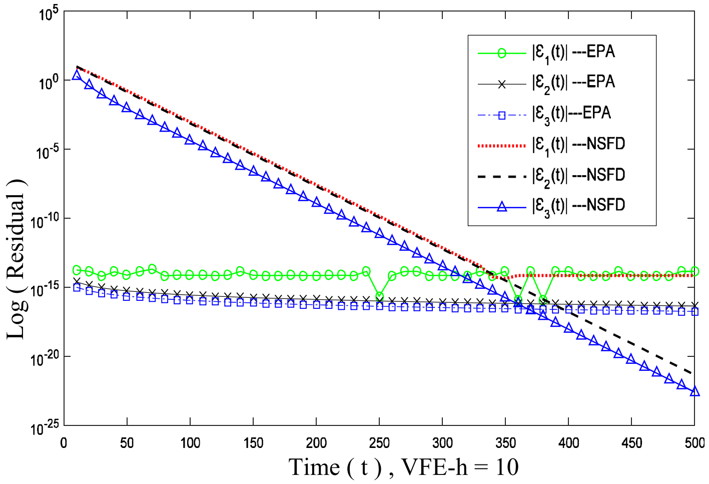

Figure 12 Absolute residuals of EPA and NSFD solutions at VFE for *h* = 10

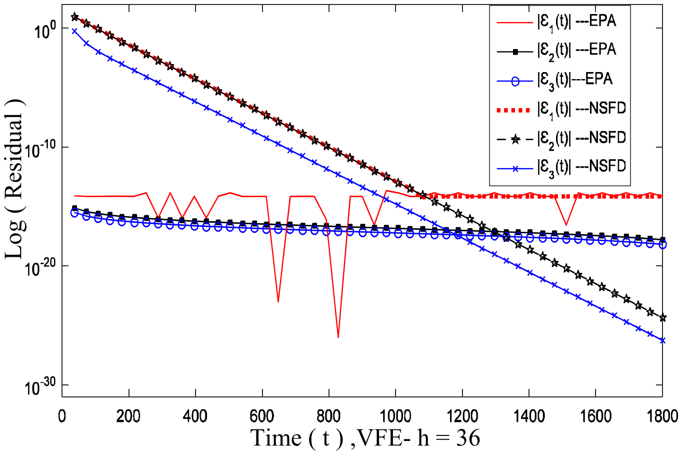

Figure 13 Absolute residuals of EPA and NSFD solutions at VFE for *h* = 36

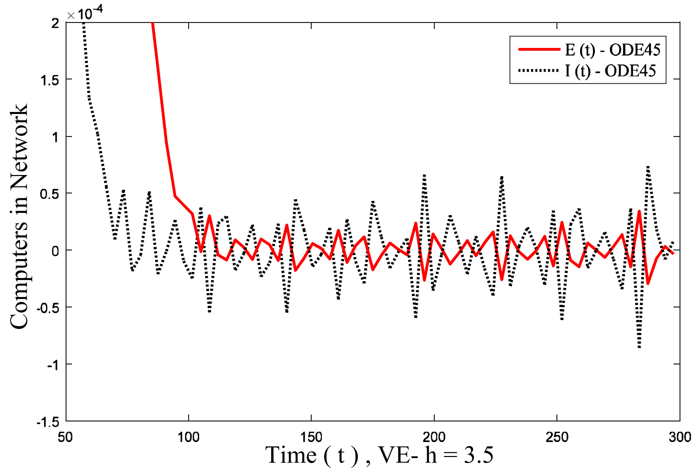

Figure 14 Convergence curves of ODE45 to VE point at $h = 3.5$

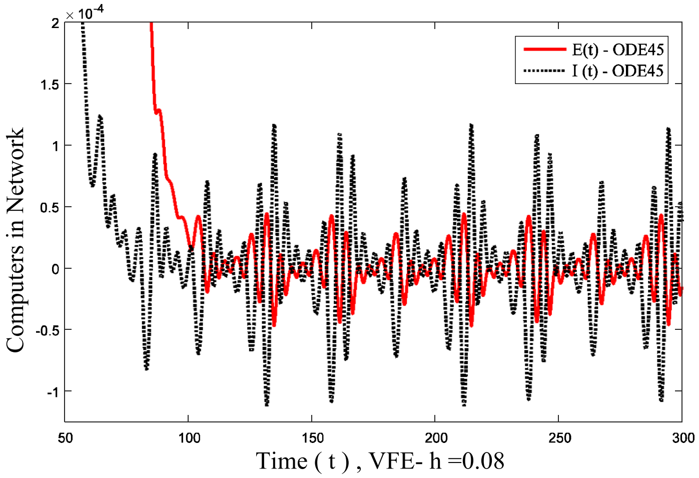

Figure 15 Convergence curves of ODE45 to VFE point at $h = 0.08$

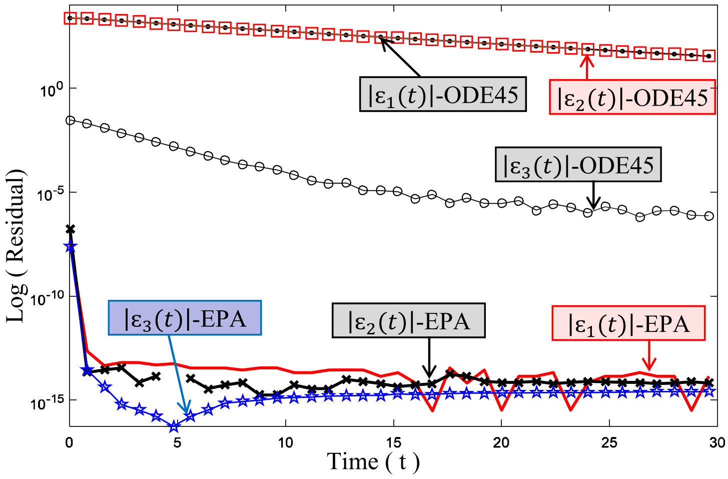

Figure 16 Absolute residuals of EPA and ODE45 solutions at VE for $h = 0.008$

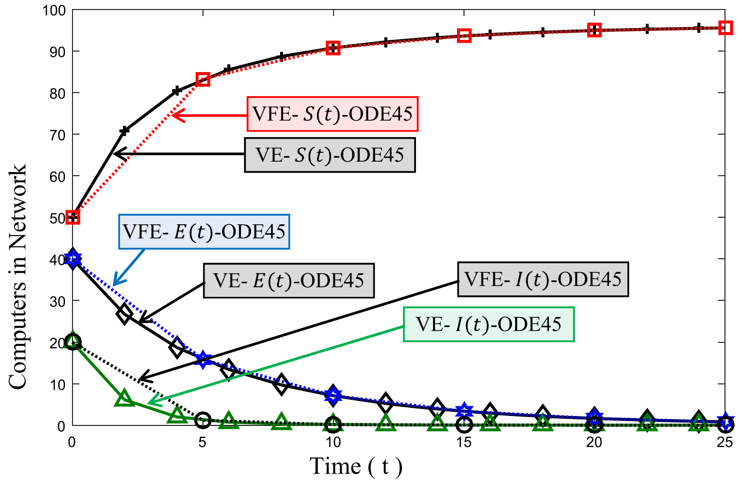

Figure 17 Convergence curves of ODE45 for VE ($h = 2$) and VFE ($h = 5$)

computer networks. A new consolidation of two distinguished techniques, Padé approximation and differential evolution, was evolved for numerical treatment of the computer virus propagation model. From the analyses of the related facts and figures it is concluded that:

- The evolutionary Padé approximation technique is successfully developed and implemented to the model of virus propagation in computer network.
- EPA yielded good approximations of state variables which satisfy the governing equations with high accuracy.
- The initial conditions and preservation of positivity and bonded-ness of the solution were efficiently handled through constraints and the penalty function approach.
- The EPA produced a closed form numerical solution of the model having no analytical solution.
- The obtained solution possesses very fast convergence, surpassing NSFD.
- An advantageous aspect of the proposed framework is that the solution found by EPA is valid for several values of step lengths and needs no re-simulation for changed step length. It is analogous to consumption of less computational efforts.
- The comparison of the tables and figures demonstrates that the solutions obtained from EPA are found to be in good agreement with NSFD particular.
- The error analysis shows that residual errors of EPA solution remain very low at each time step as compared to RK-4, Euler and NSFD schemes.
- It is also observed that the Euler, RK4 and ODE45 type finite difference schemes are not equipped with specific tools to preserve essential properties like positivity, boundedness and dynamical consistency of real world physical models. On the other hand EPA preserves all of these vital properties of the underlying model.

The efficiency of EPA is independent of choice of step length and unconditionally converges to steady state equilibriums more consistently. The proposed framework is suitable for many other nonlinear models that can be formulated as optimization problems. It is worth mentioning that the Padé approximation of order $(2, 2)$ was used in the present work. The accuracy of the numerical solution can be enhanced by using its higher order and more robust optimization approach. As a future work, we intend to apply the proposed framework to stiff nonlinear ODEs and more complicated dynamical models with integer and/or fractional orders.

Acknowledgements
We would like to express sincere thanks to the reviewers for their highly insightful and valuable suggestions concerning our paper.

Funding
Not applicable.

Availability of data and materials
All of the necessary data and the implementation details have been included in the manuscript.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
The authors have achieved equal contributions. All authors read and approved the manuscript.

Author details
[1]Department of Mathematics, University of Management and Technology, Lahore, Pakistan. [2]Faculty of Engineering, University of Central Punjab, Lahore, Pakistan. [3]Department of Informatics and Systems, School of Systems and Technology, University of Management and Technology, Lahore, Pakistan.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 17 January 2018 Accepted: 6 June 2018 Published online: 20 June 2018

References

1. Tipton, H.F., Krause, M.: Information Security Management Handbook. Auerbach Publications, Boca Raton (2010)
2. Martín del Rey, A.: Mathematical modelling of the propagation of malware: a review. Secur. Commun. Netw. **8**(15), 2561–2579 (2015)
3. Sun, C., Hsieh, Y.H.: Global analysis of an SEIR model with varying population size and vaccination. Appl. Math. Model. **34**(10), 2685–2697 (2010)
4. Anderson, R.M., May, R.M.: Infectious Diseases of Humans: Dynamics and Control. Oxford University Press, Oxford (1992)
5. Brauer, F., Chavez, C.C.: Mathematical Models in Population Biology and Epidemiology. Springer, New York (2001)
6. Song, L.P., Jin, Z., Sun, G.Q.: Modelling and analysing of botnet interactions. Physica A **390**(2), 347–358 (2011)
7. Ren, J., Yang, X., Yang, L.X., Xu, Y., Yang, F.: A delayed computer virus propagation model and its dynamics. Chaos Solitons Fractals **45**(1), 74–79 (2012)
8. Meisel, M., Pappas, V., Zhang, L.A.: Taxonomy of biologically inspired research in computer networking. Comput. Netw. **54**, 901–916 (2010)
9. Murray, W.H.: The application of epidemiology to computer viruses. Comput. Secur. **7**(2), 139–145 (1988)
10. Amador, J., Artalejo, J.R.: Modelling computer virus with the BSDE approach. Comput. Netw. **57**, 302–316 (2012)
11. Wang, Y., Cao, J., Jin, Z., Zhang, H., Sun, G.Q.: Impact of media coverage on epidemic spreading in complex networks. Physica A **23**, 5824–5835 (2013)
12. Mishra, B.K., Saini, D.: Mathematical models on computer viruses. Appl. Math. Comput. **187**(2), 929–936 (2007)
13. Shukla, J.B., Singh, G., Shukla, P., Tripathi, A.: Modeling and analysis of the effects of antivirus software on an infected computer network. Appl. Math. Comput. **227**, 11–18 (2014)
14. Kermack, W.O., McKendrick, A.G.: Contributions of mathematical theory to epidemics. Proc. R. Soc. Lond. Ser. A **115**, 700–721 (1927)
15. Feng, L., Liao, X., Han, Q., Li, H.: Dynamical analysis and control strategies on malware propagation model. Appl. Math. Model. **16–17**, 8225–8236 (2013)
16. Amador, J., Artalejo, J.R.: Stochastic modelling of computer virus spreading with warning signals. J. Franklin Inst. **50**, 1112–1138 (2013)
17. Mishra, B.K., Saini, D.K.: SEIRS epidemic model with delay for transmission of malicious objects in computer network. Appl. Math. Comput. **188**(2), 1476–1482 (2007)
18. Wang, F., Zhang, Y., Wang, C., Ma, J.: Stability analysis of an e-SEIAR model with point-to-group worm propagation. Commun. Nonlinear Sci. Numer. Simul. **20**(3), 897–904 (2015)
19. Martín del Rey, A., Sánchez, R.G.: A discrete mathematical model to simulate malware spreading. Int. J. Mod. Phys. C **23**, 1250064 (2012)
20. Yang, Y.: A note on global stability of VEISV propagation modelling for network worm attack. Appl. Math. Model. **39**(2), 776–780 (2015)
21. Mishra, B.K., Pandey, S.K.: Dynamic model of worms with vertical transmission in computer network. Appl. Math. Comput. **217**, 8438–8446 (2011)
22. Mishra, B.K., Jha, N.: SEIQS model for the transmission of malicious objects in computer network. Appl. Math. Model. **34**, 710–715 (2010)

23. Mishra, B.K., Keshri, N.: Mathematical model on the transmission of worms in wireless sensor network. Appl. Math. Model. **37**, 4103–4111 (2013)
24. Rafiq, M., Raza, A., Anayat, A.: Numerical modelling of virus transmission in a computer network. In: Proceedings of 14th International Bhurban Conference on Applied Sciences and Technology (IBCAST-2017), pp. 414–419 (2017)
25. Peng, M., He, X., Huang, J., Dong, T.: Modelling computer virus and its dynamics. Math. Probl. Eng. **2013**(5), Article ID 842614 (2013)
26. Rafiq, M., Raza, A., Rafia: Numerical modelling of transmission dynamics of vector-born plant pathogen. In: Proceedings of 14th International Bhurban Conference on Applied Sciences and Technology (IBCAST-2017), pp. 214–219 (2017)
27. Zafar, Z., Rehan, K., Mushtaq, M., Rafiq, M.: Numerical treatment for nonlinear Brusselator chemical model. J. Differ. Equ. Appl. **23**(3), 521–538 (2017)
28. Mickens, R.E.: Advances in Applications of Non-standard Finite Difference Schemes. World Scientific, Singapore (2000)
29. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Pearson Education, Upper Saddle River (1989)
30. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**, 341–359 (1997)
31. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: IEEE International Conference on Neural Networks, 1942–1948 (1995)
32. Wang, H., Wang, W., Sun, H., Rahnamayan, S.: Firefly algorithm with random attraction. Int. J. Bio-Inspir. Comput. **8**(1), 33–41 (2016)
33. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J. Glob. Optim. **39**, 459–471 (2007)
34. Passino, K.M.: Bio mimicry of bacterial foraging for distribution optimization and control. IEEE Control Syst. **22**(3), 52–67 (2002)
35. Alatas, B.: Sports inspired computational intelligence algorithms for global optimization. Artif. Intell. Rev. (2017). https://doi.org/10.1007/s10462-017-9587-x
36. Ali, J., Saeed, M., Chaudhry, N.A., Luqman, M., Tabassum, M.F.: Artificial showering algorithm: a new meta-heuristic for unconstrained optimization. Sci. Int. **27**(6), 4939–4942 (2015)
37. Sadollah, A., Eskander, H., Bahreinejad, A., Kim, J.H.: Water cycle algorithm with evaporation rate for solving constrained and unconstrained optimization problems. Appl. Soft Comput. **30**, 58–71 (2015)
38. Alexandros, T., Georgios, D.: Nature inspired optimization algorithms related to physical phenomena and laws of science: a survey. Int. J. Artif. Intell. Tools **26**, 750022 (2017). https://doi.org/10.1142/s0218213017500221
39. Babaei, M.: A general approach to approximate solutions of nonlinear differential equations using particle swarm optimization. Appl. Soft Comput. **13**(7), 3354–3365 (2013)
40. Ara, A., et al.: Wavelets optimization method for evaluation of fractional partial differential equations: an application to financial modelling. Adv. Differ. Equ. **2018**, 8 (2018). https://doi.org/10.1186/s13662-017-1461-2
41. Panagant, N., Bureerat, S.: Solving partial differential equations using a new differential evolution algorithm. Math. Probl. Eng. **2014**, 747490 (2014)
42. Lee, Z.Y.: Method of bilaterally bounded to solution Blasius equation using particle swarm optimization. Appl. Math. Comput. **179**, 779–786 (2006)
43. Karr, C.L., Wilson, E.: A self-tuning evolutionary algorithm applied to an inverse partial differential equation. Appl. Intell. **19**, 147–155 (2003)
44. Padé, H.: Sur la répresentation approchée d'une fonction par des fractions rationelles. Ann. Sci. Éc. Norm. Supér. **9**(suppl.), 1–93 (1892)
45. Vijta, M.: Some remarks on the Padé-approximations. In: Proceedings of the 3rd TEMPUS-INTCOM Symposium, pp. 1–6 (2000)
46. Bojdi, Z.K., Ahmadi-Asl, S., Aminataei, A.: A new extended Padé approximation and its application. Adv. Numer. Anal. **2013**, Article ID 263467 (2013). https://doi.org/10.1155/2013/263467
47. Chaudhary, N.A., Ahmad, M.O., Ali, J.: Constraint handling in genetic algorithms by a 2-parameter-exponential penalty function approach. Pak. J. Sci. **61**(3), 122–129 (2009)
48. Coello, C.A.C., Montes, E.M.: Constraint handling in genetic algorithms through the use of dominance-based tournament selection. Adv. Eng. Inform. **16**, 193–203 (2002)