RESEARCH

Open Access

CrossMark

Modeling the spread of computer virus via Caputo fractional derivative and the beta-derivative

Ebenezer Bonyah[1*], Abdon Atangana[2] and Muhammad Altaf Khan[3]

*Correspondence:
ebbonya@yahoo.com
[1] Department
of Mathematics and Statistics,
Kumasi Technical University,
Kumasi, Ghana
Full list of author information
is available at the end of the
article

Abstract

The concept of information science is inevitable in the human development as science and technology has become the driving force of all economics. The connection of one human being during epidemics is vital and can be studied using mathematical principles. In this study, a well-recognized model of computer virus by Piqueira et al. (J Comput Sci 1:31—34, 2005) and Piqueira and Araujo (Appl Math Comput 2(213):355—360, 2009) is investigated through the Caputo and beta-derivatives. A less detail of stability analysis was discussed on the extended model. The analytical solution of the extended model was solved via the Laplace perturbation method and the homotopy decomposition technique. The sequential summary of each of iteration method for the extend model was presented. Using the parameters in Piqueira and Araujo (Appl Math Comput 2(213):355—360, 2009), some numerical simulation results are presented.

Keywords: Computer virus, Caputo fractional derivative, Beta-derivative, Antidotal, Stability

Background

The idea of computer virus came into being around 1980 and has continued threatening the society. During these early stages, the threat of this virus was minimal [1]. Modern civilized societies are being automated with computer applications making life easy in the areas such as education, health, transportation, agriculture and many more. Following recent development in complex computer systems, the trend has shifted to sophisticate dynamic of computer virus which is difficult to deal with. In 2001, for example, the cost associated with computer virus was estimated to be 10.7 United State dollars for only the first quarter [1]. Consequently, a comprehensive understanding of computer virus dynamics has become inevitable to researchers considering the role played by this wonderful device. To ensure the safety and reliability of computers, this virus burden can be tackled in twofold: microscopic and macroscopic [2—6].

The microscopic level has been investigated by [3], who developed anti-virus program that removes virus from the computer system when detected. The program is capable of upgrading itself to ensure that new virus can be dealt with when attacks computer. The characteristics of this program are similar to that of vaccination against a disease. They are not able to guarantee safety in computer network system and also difficult to make

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 2 of 15

good future predictions. The macroscopic aspect of computer has seen tremendous attention in the area of modeling the spread of this virus and determining the long-term behavior of the virus in the network system since 1980 [4]. The concept of epidemiological modeling of disease has been applied in the study of the spread of computer virus in macroscopic scale [6−8].

Possibly reality of nature could be well understood via fractional calculus perspective. A considerable attention has been devoted to fractional differential equations by the fact that fractional-order system is capable to converge to the integer-order system timely. Fractional-order differential equations' applications in modeling processes have the merit of nonlocal property [9–11]. The model proposed in [10] is a deterministic one and fails to have hereditary and memory effect and therefore, cannot adequately describe the processes very well.

In this paper, we present the fractional-order derivative and obtain analytic numeric solution of the model presented in [10]. The rationale behind the application of fractional derivatives can also be ascertained from some of the current papers published on mathematical modeling [12–16]. In addition to this, the practical implication of fractional derivative can be established in [17].

Model formulation

In this study, we take into account the model proposed by [10]. In their study, the total population of this model is denoted by T which is subdivided into four groups. S denotes the non-infected computers capable of being infected after making contact with infected computer. A is the kind of computers non-infected equipped with antivirus. I denotes infected computers capable of infecting non-infected computers and R deals with removed ones due to infection or not. The recruitment rate of computers into the non-infected computers' class is denoted by N and μ is the proportion coefficient for the mortality rate which is not attributable to the virus. β is the rate of proportion of infection as a result of product of SI. The conversion of susceptible computer into antidotal is the product of SI denoted by α_{SA}. The proportion of converting infected computers into antidotal ones in the network is the product of SA denoted by α_{IA}. The rate of removed computers being converted into susceptible class is represented by σ and δ denotes the rate at which the virus rate computers useless and remove from the system.

The mathematical model under consideration here is given as:

$$\begin{cases} \frac{dS}{dt} = N - \alpha_{AS}SA - \beta SI - \mu S + \sigma R, \\ \frac{dI}{dt} = \beta SI - \alpha_{AI}AI - \delta I - \mu I, \\ \frac{dR}{dt} = \delta I - \sigma R - \mu R, \\ \frac{dA}{dt} = \alpha_{AS}SA + \alpha_{AS}AI - \mu A. \end{cases} \tag{1}$$

Here, the recruitment rate is taken to be $N = 0$, indicating that there is no new computer entering into the system during the propagation of the virus. This is because in reality the transfer of virus is faster than adding new computers into the system. The same reason can assign to the choice of $\mu = 0$, taking into account the fact that the computer outmodedness time is longer than the time of the virus action being manifested. Accordingly, equation system (1) is reformulated as follows:

$$
\begin{cases}
\frac{dS}{dt} = -\alpha_{AS}SA - \beta SI + \sigma R, \\
\frac{dI}{dt} = \beta SI - \alpha_{AI}AI - \delta I, \\
\frac{dR}{dt} = \delta I - \sigma R, \\
\frac{dA}{dt} = \alpha_{AS}SA + \alpha_{AS}AI.
\end{cases}
\tag{2}
$$

In this paper, we shall fully explore the concept of fractional derivatives and other current proposed derivatives, and in this study, we shall examine this model in the context of fractional derivatives as well as the beta-derivatives. Consequently, Eq. (2) can be transformed into the following:

$$
\begin{cases}
{}_0^A D_t^\alpha S(t) = -\alpha_{AS}SA - \beta SI + \sigma R, \\
{}_0^A D_t^\alpha I(t) = \beta SI - \alpha_{AI}AI - \delta I, \\
{}_0^A D_t^\alpha R(t) = \delta I - \sigma R, \\
{}_0^A D_t^\alpha A(t) = \alpha_{AS}SA + \alpha_{AS}AI.
\end{cases}
\tag{3}
$$

where ${}_0^A D_t^\alpha$ represents the Caputo derivative or the new derivative called beta-derivative. In the next section, some background on the use of fractional and beta-derivatives will be presented. The basic aim of this study is to explore both fractional and beta-derivatives for modeling epidemiological problem in computers. The fractional derivatives are noted for non-local problems and maybe appropriate for epidemiological issues. The fractional order, however, is an indispensable tool for numerical simulations, and therefore, a local derivative with fractional order is presented in this study to model the propagation of computer virus in a network.

This provides the invariance of Ω as to be determined. We conclude from this theorem, that it is sufficient to deal with the dynamics of (1) in Ω. In this respect, the model can be assumed as being epidemiologically and mathematically well-posed [18].

Basic concept about the beta-derivative and Caputo derivative

Definition 1   Let f be a function, such that, $f : [a, \infty) \to \mathbb{R}$. Then, the beta-derivative is expressed as follows:

$$
{}_0^A D_x^\beta \left(f(x) \right) = \lim_{\varepsilon \to 0} \frac{\left(x + \varepsilon \left(x + \frac{1}{\Gamma(\beta)} \right)^{1-\beta} \right) - f(x)}{\varepsilon}
\tag{4}
$$

for all $x \geqslant a, \beta \in (0, 1]$. Then if the limit of the above exists, it is said to be beta-differentiable. It can be noted that the above definition does not depend on the interval stated. If the function is differentiable, then definition given at a point zero is different from zero.

Theorem 1   *Assuming that $g \neq 0$ and there are two functions beta-differentiable with $\beta \in (0, 1]$; then, the following relations can be presented:*

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 4 of 15

1. ${}_0^A D_x^\alpha \big(af(x) + bg(x)\big) = a_0^A D_x^\alpha \big(f(x)\big) + b_0^A D_x^\alpha \big(f(x)\big)$ *for all a and b being real numbers,*

2. ${}_0^A D_x^\alpha (c) = 0$ *for c any given constant,*

3. ${}_0^A D_x^\alpha \big(f(x)g(x)\big) = g(x) {}_0^A D_x^\alpha \big(f(x)\big) + f(x) {}_0^A D_x^\alpha \big(g(x)\big)$

4. ${}_0^A D_x^\alpha \left(\dfrac{f(x)}{g(x)}\right) = \dfrac{g(x) {}_0^A D_x^\alpha (f(x)) - f(x) D_x^\alpha (g(x))}{g^2 x}$

The proofs of the above relations are identical to the one in [19].

Definition 2   Assuming that $f : [a, \infty) \to \mathbb{R}$ be a function in a way that f is differentiable and also alpha-differentiable. Assume g be a function defined in the range of f and also differentiable, then we obtain the following rule:

$$
{}_0^A D_x^\alpha I_x^\beta \big(f(x)\big) = \int_a^x \left(t + \frac{1}{\Gamma(\beta)}\right)^{\beta - 1} f(t) \, \mathrm{d}t. \tag{5}
$$

The above operator is referred to as the inverse operator of the proposed fractional derivative. We shall present the principle behind this statement using the following theorem.

Theorem 2 ${}_0^A D_x^\alpha \left[{}_0^A I_x^\beta f(x)\right] = f(x)$ *for all $x \geqslant a$ with f a given continuous and differentiable function.*

Definition 3   The Caputo fractional derivative of a differentiable function is expressed as:

$$
D_x^\alpha \big(f(x)\big) = \frac{1}{\Gamma(n - \alpha)} \int_0^x (x - t)^{n - \alpha - 1} \left(\frac{\mathrm{d}}{\mathrm{d}t}\right)^n f(t) \, \mathrm{d}t, \quad n - 1 < \alpha \leqslant n. \tag{6}
$$

The properties behind the use of the Caputo derivative can be established in [14–16, 19]. Given all the information discussed, we shall introduce in the subsequent section the analysis of this model.

Analysis of the mathematical model

This section is devoted to discuss the stability analysis of the model. The disease-free equilibrium and endemic equilibrium points are established. To determine the equilibria points, we make an asumption that the system does not depend on time; hence,

$$
{}_0^A D_t^\alpha S(t) = {}_0^A D_t^\alpha I(t) = {}_0^A D_t^\alpha R(t) = {}_0^A D_t^\alpha A(t) = 0. \tag{7}
$$

$$
\begin{cases}
0 = -\alpha_{AS} S^* A^* - \beta S^* I^* + \sigma R^* \\
0 = \beta S^* I^* - \alpha_{AI} A^* I^* - \delta I^* \\
0 = \delta I^* - \sigma R^* \\
0 = \alpha_{AS} S^* A + \alpha_{AS} A^* I^*
\end{cases}
. \tag{8}
$$

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 5 of 15

Solving the above system, we obtain

$$S^* = \frac{\delta}{\beta}; \quad I^* = \frac{T - \delta/\beta}{1 + \delta/\sigma}; \quad R^* = \frac{T - \delta/\beta}{1 + \sigma/\delta}; \quad A = 0. \tag{9}$$

Hence, the disease-free equilibria are given as:

$$E_1 = (S, I, R, A) = (0, 0, 0, T); \tag{10}$$

$$E_2 = (S, I, R, A) = (T, 0, 0, 0). \tag{11}$$

The stability analysis of this model has been extensively dealt with in [10]. The next section will be concentrated on an approximate solution based on the two analytical techniques for each situation.

Analysis of approximate solutions

One of the most challenging tasks in non-linear fractional differential equation systems is probably how to obtain exact analytical solutions. This accounts for the reasons why in recent times, a lot of attention has been devoted in the quest for obtaining techniques that can ensure asymptotic solutions in such situations. We shall make reference to some of the recent techniques on this subject which are efficient and effective and have been widely used; for instance, the decomposition method [12], Sumudu homotopy perturbation method [20], the Adomian Decomposition method [11, 21], homotopy perturbation method [19, 22, 23], the homotopy Laplace perturbation method [24], and the homotopy. In this study however, we shall make use of two of these stated techniques, specifically the Laplace homotopy perturbation method and the homotopy decomposition method. The homotopy decomposition method will be employed to provide solution to the model with the beta-derivative, followed by the Laplace homotopy perturbation method which will be used to solve the system with Caputo derivative.

Solution with the Laplace homotopy perturbation method

This form was initially proposed in [18], and has been also employed in various scientific researches. We shall explore the methodology for obtaining solution to system (3) with the Caputo fractional derivative in this section:

$$\begin{cases} {}_0^C D_t^\alpha S(t) = -\alpha_{AS} SA - \beta SI + \sigma R, \\ {}_0^C D_t^\alpha I(t) = \beta SI - \alpha_{AI} AI - \delta I, \\ {}_0^C D_t^\alpha R(t) = \delta I - \sigma R, \\ {}_0^C D_t^\alpha A(t) = \alpha_{AS} SA + \alpha_{AI} AI. \end{cases} \tag{12}$$

Applying the Laplace transform operator on both sides of the above system, we obtain the following:

$$\begin{cases} S(\tau) = \frac{1}{\tau^\alpha} S(0) + \frac{1}{\tau^\alpha} \ell(-\alpha_{AS} SA - \beta SI + \sigma R), \\ I(\tau) = \frac{1}{\tau^\alpha} I(0) + \frac{1}{\tau^\alpha} \ell(\beta SI - \alpha_{AI} AI - \delta I), \\ R(\tau) = \frac{1}{\tau^\alpha} R(0) + \frac{1}{\tau^\alpha} \ell(\delta I - \sigma R), \\ A(\tau) = \frac{1}{\tau^\alpha} A(0) + \frac{1}{\tau^\alpha} \ell(\alpha_{AS} SA + \alpha_{AI} AI). \end{cases} \tag{13}$$

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 6 of 15

The Laplace variable is denoted by τs. Further, the inverse Laplace transform is applied on both sides of the system to yield

$$
\begin{cases}
S(t) = S(0) + \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(-\alpha_{AS}SA - \beta SI + \sigma R)\right\}, \\
I(t) = I(0) + \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(\beta SI - \alpha_{AI}AI - \delta I)\right\}, \\
R(t) = R(0) + \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(\delta I - \sigma R)\right\}, \\
A(t) = A(0) + \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(\alpha_{AS}SA + \alpha_{AI}AI)\right\}.
\end{cases}
\tag{14}
$$

Following the system above, we shall make an assumption that a solution can be obtained in the form of series as follows:

$$
S(t) = \sum_{n=0}^{\infty} S_n(t), \quad I(t) = \sum_{n=0}^{\infty} I_n(t), \quad R(t) = \sum_{n=0}^{\infty} R_n(t), \quad A(t) = \sum_{n=0}^{\infty} A_n(t).
\tag{15}
$$

Conversely, substituting the above solution in system (11) and adding the embedding parameter $p \in (0, 1]$, bringing all the terms of the same power of the embedding parameter p together, we have

$$
p^0 : \begin{cases}
S_0(t) = S(0) \\
I_0(t) = I(0) \\
R_0(t) = R(0) \\
A_0(t) = A(0)
\end{cases},
\tag{16}
$$

$$
p^1 : \begin{cases}
S_1(t) = \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(-\alpha_{AS}S_0A_0 - \beta S_0I_0 + \sigma R_0)\right\} \\
I_1(t) = \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(\beta S_0I_0 - \alpha_{AI}A_0I_0 - \delta I_0)\right\} \\
R_1(t) = \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(\delta I_0 - \sigma R_0)\right\} \\
A_1(t) = \ell^{-1}\left\{\frac{1}{\tau^\alpha}\ell(\alpha_{AS}S_0A_0 + \alpha_{AI}A_0I_0)\right\}
\end{cases}
\tag{17}
$$

In broad-spectrum, we shall obtain the following system of iteration formulas (15):

$$
p^n : \begin{cases}
S_n(t) = \ell^{-1}\left\{\frac{1}{s^\alpha}\ell\left(-\alpha_{AS}\sum_{j=0}^{n-1} A_{(n-j-1)}S_jA_j - \beta\sum_{j=0}^{n-1} I_{(n-j-1)}S_j + \sigma R_{(n-1)}\right)\right\} \\
I_n(t) = \ell^{-1}\left\{\frac{1}{s^\alpha}\ell\left(\beta\sum_{j=0}^{n-1} I_{(n-j-1)}S_j - \alpha_{AI}\sum_{j=0}^{n-1} I_{(n-j-1)}A_j - \delta I_{(n-1)}\right)\right\} \\
R_n(t) = \ell^{-1}\left\{\frac{1}{s^\alpha}\ell\left(\delta I_{(n-1)} - \sigma R_{(n-1)}\right)\right\} \\
A_n(t) = \ell^{-1}\left\{\frac{1}{s^\alpha}\ell\left(\alpha_{AS}\alpha_{AS}\sum_{j=0}^{n-1} A_{(n-j-1)}S_jA_j + \alpha_{AI}\sum_{j=0}^{n-1} I_{(n-j-1)}A_j\right)\right\}
\end{cases}.
\tag{18}
$$

The above general rule can be simplified in the following algorithm steps:

$$
p^0 : \begin{cases}
S_0(t) = S(0) \\
I_0(t) = I(0) \\
R_0(t) = R(0) \\
A_0(t) = A(0)
\end{cases}
\tag{19}
$$

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 7 of 15

Algorithm 1. This technique can be employed to obtain a special solution to system (2) via a Caputo fractional derivative

Input p^0 : $\begin{cases} S_0(t) = S(0) \\ I_0(t) = I(0) \\ R_0(t) = R(0) \\ A_0(t) = A(0) \end{cases}$ as initial input,

j-number terms in the rough computation,

Output: $\begin{cases} S_{\text{appr}}(t) = S(0) \\ I_{\text{appr}}(t) = I(0) \\ R_{\text{appr}}(t) = R(0) \\ A_{\text{appr}}(t) = A(0), \end{cases}$ the estimated solution.

$$\text{Step1 : Put} \begin{cases} S_0(t) = S(0) \\ I_0(t) = I(0) \\ R_0(t) = R(0) \\ A_0(t) = A(0) \end{cases} \text{and} \begin{cases} S_{\text{appr}}(t) = S(0) \\ I_{\text{appr}}(t) = I(0) \\ R_{\text{appr}}(t) = R(0) \\ A_{\text{appr}}(t) = A(0) \end{cases} = \begin{cases} S_0(t) = S(t) \\ I_0(t) = I(t) \\ R_0(t) = R(t) \\ A_0(t) = A(t), \end{cases} \quad (20)$$

Step 2: For $j = 1$ to $n - 1$ do step 3, step 4 and step 5:

$$\begin{cases} S_1(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell(-\alpha_{AS}S_0A_0 - \beta S_0I_0 + \sigma R_0) \right\} \\ I_1(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell(\beta S_0I_0 - \alpha_{AI}A_0I_0 - \delta I_0) \right\} \\ R_1(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell(\delta I_0 - \sigma R_0) \right\} \\ A_1(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell(\alpha_{AS}S_0A_0 + \alpha_{AI}A_0I_0) \right\} \end{cases} \quad (21)$$

Step 3: Compute

$$\begin{cases} S_n(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell\left(-\alpha_{AS}\sum_{j=0}^{n-1} A_{(n-j-1)}S_jA_j - \beta \sum_{j=0}^{n-1} I_{(n-j-1)}S_j + \sigma R_{(n-1)} \right) \right\} \\[3mm] I_n(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell\left(\beta \sum_{j=0}^{n-1} I_{(n-j-1)}S_j - \alpha_{AI}\sum_{j=0}^{n-1} I_{(n-j-1)}A_j - \delta I_{(n-1)} \right) \right\} \\[3mm] R_n(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell\left(\delta I_{(n-1)} - \sigma R_{(n-1)} \right) \right\} \\[3mm] A_n(t) = \ell^{-1}\left\{ \frac{1}{\tau^\alpha}\ell\left(\alpha_{AS}\alpha_{AS}\sum_{j=0}^{n-1} A_{(n-j-1)}S_jA_j + \alpha_{AI}\sum_{j=0}^{n-1} I_{(n-j-1)}A_j \right) \right\} \end{cases} \quad (22)$$

Step 4: Compute

$$\begin{cases} S_{(m+1)}(t) = B_m(t) + S_{(\text{appr})}(t) \\ I_{(m+1)}(t) = B_m(t) + I_{(\text{appr})}(t) \\ R_{(m+1)}(t) = K_m(t) + R_{(\text{appr})}(t) \\ A_{(m+1)}(t) = K_m(t) + A_{(\text{appr})}(t) \end{cases} \quad (23)$$

Step 5: Compute

$$\begin{cases} S_{(\text{appr})}(t) = S_{\text{appr}}(t) + S_{(m+1)}(t) \\ I_{(\text{appr})}(t) = I_{\text{appr}}(t) + I_{(m+1)}(t) \\ R_{(\text{appr})}(t) = R_{\text{appr}}(t) + R_{(m+1)}(t) \\ A_{(\text{appr})}(t) = A_{\text{appr}}(t) + A_{(m+1)}(t) \end{cases} \quad (24)$$

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 8 of 15

Stop.

The above algorithm shall be applied to obtain the unique solution of system (3) via the Caputo derivative. We shall explore the situation where the beta-derivative is used and this will be discussed in "Basic concept about the beta-derivative and Caputo derivative" section.

Solution with the homotopy decomposition method

This method is explored here to obtain a special solution to system (3) with the beta-derivative. The reason is that using the Laplace transform on this derivative does not guarantee desirable results. The inverse operator of this derivative, however, can be applied which is termed as the beta-integral, and this helps to convert the ordinary differential equation into an integral equation in a way that the concept of homotopy can be applied. Consequently, taking the inverse operator on both sides of system (2), we have the following system of integral equations:

$$\begin{cases} S(\tau) = S(0) + {}_0^A I_t^\alpha (-\alpha_{AS} SA - \beta SI + \sigma R) \\ I(\tau) = I(0) + {}_0^A I_t^\alpha (\beta SI - \alpha_{AI} AI - \delta I) \\ R(\tau) = R(0) + {}_0^A I_t^\alpha (\delta I - \sigma R) \\ A(\tau) = A(0) + {}_0^A I_t^\alpha (\alpha_{AS} SA + \alpha_{AI} AI) \end{cases} \tag{25}$$

where

$$
{}_0^A I_t^\beta \left(f(x) \right) = \int_0^t \left(\tau + \frac{1}{\Gamma(\beta)} \right)^{1-\beta} f(\tau) \mathrm{d}\tau. \tag{26}
$$

Additionally, we shall make an assumption that a solution of the above can be derived in a form of series as follows:

$$
S(t) = \sum_{n=0}^\infty S_n(t), \quad I(t) = \sum_{n=0}^\infty I_n(t), \quad R(t) = \sum_{n=0}^\infty R_n(t), \quad A(t) = \sum_{n=0}^\infty A_n(t). \tag{27}
$$

Conversely, substituting the above solution in system (15) and adding the embedding parameter $p \in (0, 1]$, bringing all the terms of the same power of the embedding parameter p together, we have

$$
p^0 : \begin{cases} S_0(t) = S(0) \\ I_0(t) = I(0) \\ R_0(t) = R(0) \\ A_0(t) = A(0) \end{cases} \tag{28}
$$

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 9 of 15

$$p^1 : \begin{cases} S_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{-\alpha_{AS}S_0A_0 - \beta S_0I_0 + \sigma R_0\}d\tau \\[2mm] I_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{\beta S_0I_0 - \alpha_{AI}A_0I_0 - \delta I_0\}d\tau \\[2mm] R_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{\delta I_0 - \sigma R_0\}d\tau \\[2mm] A_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{\alpha_{AS}S_0A_0 + \alpha_{AI}A_0I_0\}d\tau. \end{cases} \tag{29}$$

In broad sense, we shall have the ensuing system of iteration formulas:

$$p^n : \begin{cases} S_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{-\alpha_{AS} \sum\limits_{j=0}^{n-1} A_{(n-j-1)}S_jA_j - \beta \sum\limits_{j=0}^{n-1} I_{(n-j-1)}S_j + \sigma R_{(n-1)}\right\}d\tau \\[3mm] I_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{\beta \sum\limits_{j=0}^{n-1} I_{(n-j-1)}S_j - \alpha_{AI} \sum\limits_{j=0}^{n-1} I_{(n-j-1)}A_j - \delta I_{(n-1)}\right\}d\tau \\[3mm] R_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{\delta I_{(n-1)} - \sigma R_{(n-1)}\right\}d\tau \\[3mm] A_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{\alpha_{AS}\alpha_{AS} \sum\limits_{j=0}^{n-1} A_{(n-j-1)}S_jA_j + \alpha_{AI} \sum\limits_{j=0}^{n-1} I_{(n-j-1)}A_j\right\}d\tau. \end{cases} \tag{30}$$

The above embellishment can be taken up in the succeeding procedure.

$$\text{Input} \begin{cases} S_0(t) = S(0) \\ I_0(t) = I(0) \\ R_0(t) = R(0) \\ A_0(t) = A(0) \end{cases} \quad \text{as initial input,}$$

j-number terms in the rough computation,

$$\text{Output:} \begin{cases} S_{\text{appr}}(t) \\ I_{\text{appr}}(t) \\ R_{\text{appr}}(t) \\ A_{\text{appr}}(t) \end{cases} \text{the estimated solution.}$$

Step 1:

$$\text{put} \begin{cases} S_0(t) = S(t) \\ I_0(t) = I(0) \\ R_0(t) = R(0) \\ A_0(t) = A(0) \end{cases} \quad \text{and} \quad \begin{cases} S_{\text{appr}}(t) \\ I_{\text{appr}}(t) \\ R_{\text{appr}}(t) \\ A_{\text{appr}}(t) \end{cases} = \begin{cases} S_0(t) \\ I_0(t) \\ R_0(t) \\ A_0(t) \end{cases} \tag{31}$$

Step 2: For $J - 1$ to $n - 1$ do step 3, step 4 and step 5:

$$\begin{cases} S_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{-\alpha_{AS}S_0A_0 - \beta S_0I_0 + \sigma R_0\}d\tau \\[2mm] I_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{\beta S_0I_0 - \alpha_{AI}A_0I_0 - \delta I_0\}d\tau \\[2mm] R_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{\delta I_0 - \sigma R_0\}d\tau \\[2mm] A_1(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \{\alpha_{AS}S_0A_0 + \alpha_{AI}A_0I_0\}d\tau \end{cases} \tag{32}$$

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 10 of 15

Step 3: Compute:

$$
\begin{cases}
B_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{-\alpha_{AS} \sum\limits_{j=0}^{n-1} A_{(n-j-1)} S_j A_j - \beta \sum\limits_{j=0}^{n-1} I_{(n-j-1)} S_j + \sigma R_{(n-1)}\right\} d\tau \\[2mm]
B_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{\beta \sum\limits_{j=0}^{n-1} I_{(n-j-1)} S_j - \alpha_{AI} \sum\limits_{j=0}^{n-1} I_{(n-j-1)} A_j - \delta I_{(n-1)}\right\} d\tau \\[2mm]
K_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{\delta I_{(n-1)} - \sigma R_{(n-1)}\right\} d\tau \\[2mm]
K_n(t) = \int\limits_0^t \left(\tau + \frac{1}{\Gamma(\beta)}\right)^{1-\beta} \left\{\alpha_{AS}\alpha_{AS} \sum\limits_{j=0}^{n-1} A_{(n-j-1)} S_j A_j + \alpha_{AI} \sum\limits_{j=0}^{n-1} I_{(n-j-1)} A_j\right\} d\tau
\end{cases}
$$

$$(33)$$

Step 4: Compute:

$$
\begin{cases}
S_{(m+1)}(t) = B_m(t) + S_{(appr)}(t) \\
I_{(m+1)}(t) = B_m(t) + I_{(appr)}(t) \\
R_{(m+1)}(t) = K_m(t) + R_{(appr)}(t) \\
A_{(m+1)}(t) = K_m(t) + A_{(appr)}(t)
\end{cases}
$$

$$(34)$$

Step 5: Compute:

$$
\begin{cases}
S_{(appr)}(t) = S_{(appr)}(t) + S_{(m+1)}(t) \\
I_{(appr)}(t) = I_{(appr)}(t) + I_{(m+1)}(t) \\
R_{(appr)}(t) = R_{(appr)}(t) + R_{(m+1)}(t) \\
A_{(appr)}(t) = A_{(appr)}(t) + A_{(m+1)}(t)
\end{cases}
$$

$$(35)$$

Numerical results

We shall employ both Algorithms 1 and 2 to obtain an approximate solution of system (3) via the Caputo fractional derivative and the beta-derivative methods.

With the Caputo fractional derivative

In this part, Algorithm 1 is applied to obtain an approximation solution of system (3):

$$
\begin{cases}
S_0(t) = e \\
I_0(t) = f \\
R_0(t) = g \\
A_0(t) = h
\end{cases}
$$

$$(36)$$

$$S_1(t) = \frac{t^\mu(-\alpha_{AS}eh - \beta ef + \sigma g)}{\Gamma[1+\mu]}$$

$$I_1(t) = \frac{t^\mu(\beta ef - \alpha_{AI}hf - \delta f)}{\Gamma[1+\mu]}$$

$$R_1(t) = \frac{t^\mu(\delta f - \sigma g)}{\Gamma[1+\mu]}$$

$$A_1(t) = \frac{t^\mu(\alpha_{AS}eh + \alpha_{AI}hf)}{\Gamma[1+\mu]}$$

$$(37)$$

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 11 of 15

With the beta-derivative
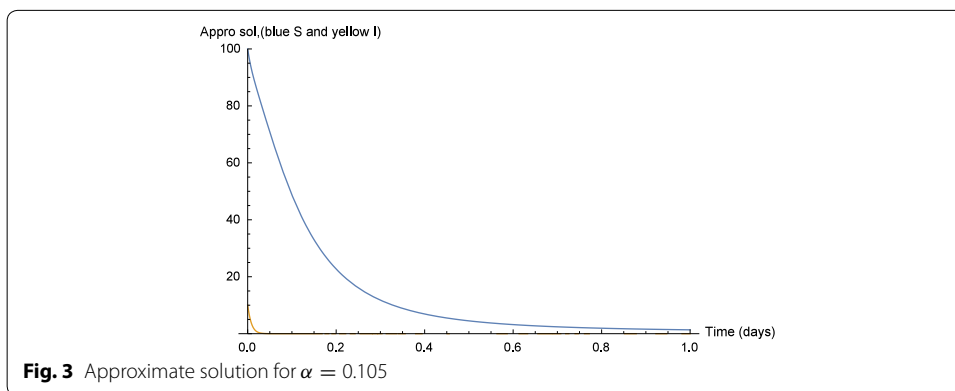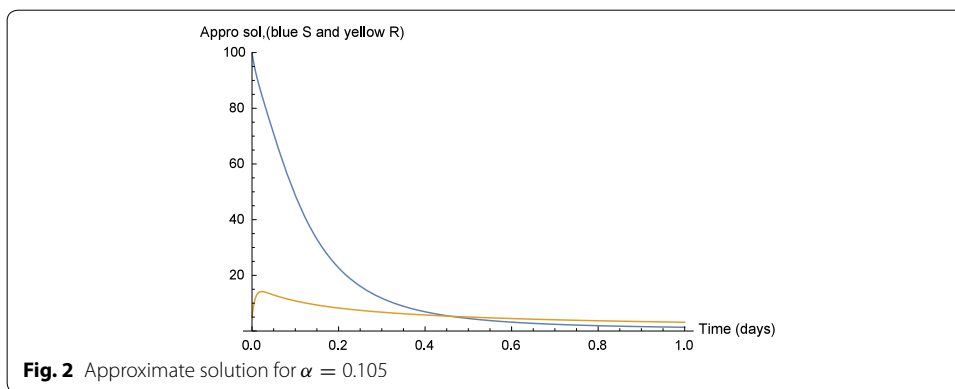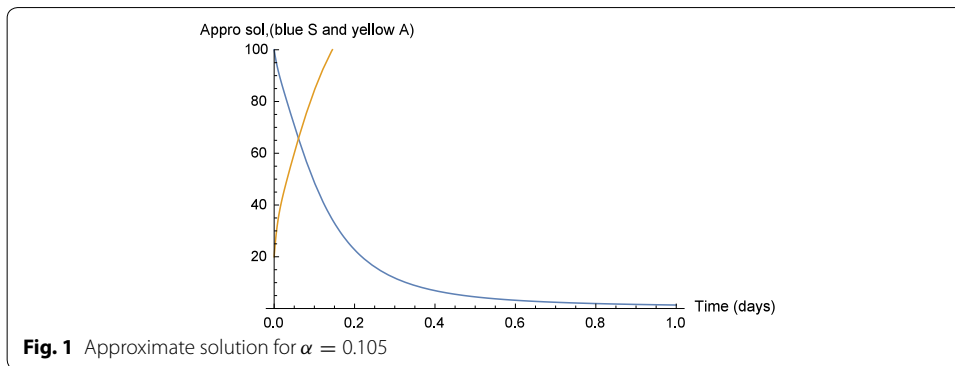
Algorithm 2 is used to derive the following series solutions:

$$S_1(t) = \frac{\left(\left(\frac{1}{\Gamma[\mu]}\right)^{-\mu} - \left(t + \frac{1}{\Gamma[\mu]}\right)^{-\mu}(1 + t\Gamma[\mu])^2\right)(-\alpha_{AS}eh - \beta ef + \sigma g)}{(-2 + \mu)\Gamma[\mu]^2} \tag{38}$$

$$I_1(t) = \frac{\left(\left(\frac{1}{\Gamma[\mu]}\right)^{-\mu} - \left(t + \frac{1}{\Gamma[\mu]}\right)^{-\mu}(1 + t\Gamma[\mu])^2\right)(\beta ef - \alpha_{AI}hf - \delta f)}{(-2 + \mu)\Gamma[\mu]^2} \tag{39}$$

$$R_1(t) = \frac{\left(\left(\frac{1}{\Gamma[\mu]}\right)^{-\mu} - \left(t + \frac{1}{\Gamma[\mu]}\right)^{-\mu}(1 + t\Gamma[\mu])^2\right)(\delta f - \sigma g)}{(-2 + \mu)\Gamma[\mu]^2} \tag{40}$$

$$A_1(t) = \frac{\left(\left(\frac{1}{\Gamma[\mu]}\right)^{-\mu} - \left(t + \frac{1}{\Gamma[\mu]}\right)^{-\mu}(1 + t\Gamma[\mu])^2\right)(\alpha_{AS}eh + \alpha_{AI}hf)}{(-2 + \mu)\Gamma[\mu]^2} \tag{41}$$
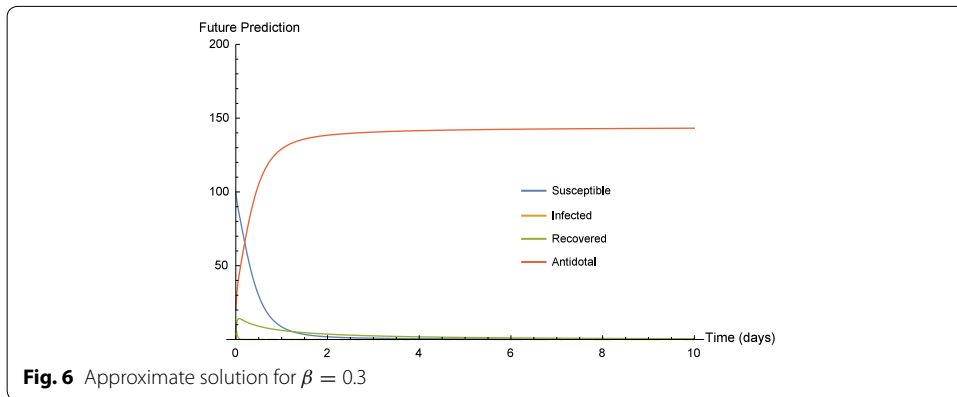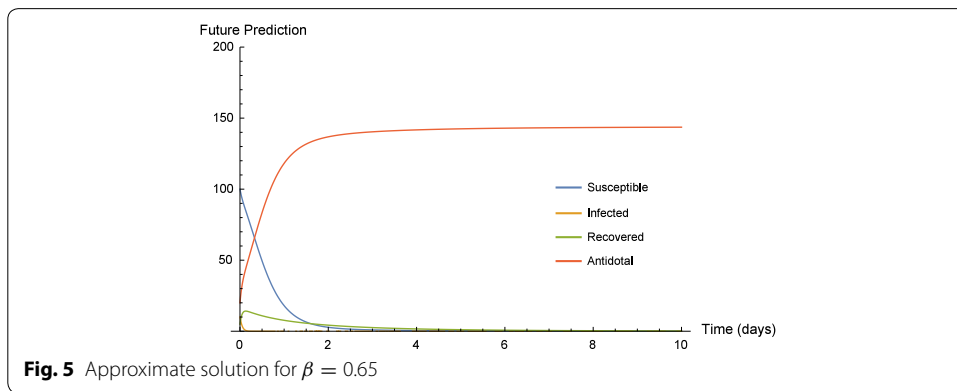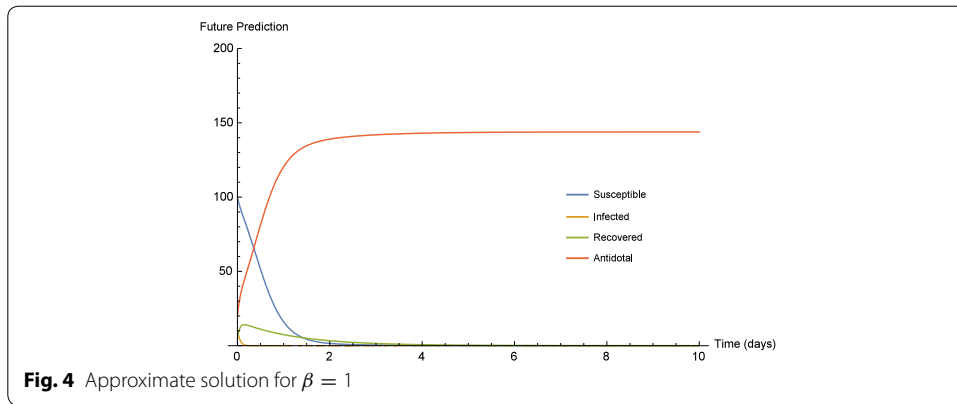
Numerical simulations

In this section, we use parameters in [10], to obtain numerical simulation on illustration of the approximate solutions based on function of time as well as alpha. The parameter values used are $\alpha_{AS} = 0.6$, $\alpha_{IA} = 0.4$, $\beta = 0.2$, $\sigma = 0.85$, $\delta = 0.40$ and with the following initial conditions $S(0) = 100$, $I(0) = 10$, $R(0) = 5$ and $A(0) = 20$.

Figures 1, 2, 3, 4, 5, 6, 7 and 8 depict the graphical representations of system (3). It is clearly seen from the graph that susceptible class which also represents the total population get quickly infected due to how fast the virus spread with the system. It is obvious in the above figures that the total number of infected computers decreased as the total number of antidotal increased which is a strategy to reduce the number of susceptible computers. The numerical predictions (Figs. 1, 2, 3, 4, 5, 6, 7, 8) are also attributable to fractional order of beta. Realistically, when the beta is above 0.5, the non-realistic prediction is attained. As the beta value increases above 0.5, it appears that anodotal computers exceed the entire initial population. Thus, there exists unrealistic prediction. The best option to secure more computers is put anti-virus on both infected and susceptible computers to avoid disaster and huge cost. This is achieved when the beta value is <0.5 as observed in (Figs. 4, 5, 6, 7, 8). It is not surprising, given the potential danger of anti virus in a computer system because the entire initial population ends up in that compartment. It is remarkable to note that when beta is 1 we obtain the ordinary derivative which implies the model ordinary derivative and hardly give good predictions. The newly introduced beta-calculus, however, has the potential of vividly describing a given physical problem. The obtained Figs. 1, 2, 3, 4, 5, 6, 7 and 8 is far better and accurate as compared to those obtained in [17]. This work has given better predictions as shown in Figs. 1, 2, 3, 4, 5, 6, 7 and 8 which is not the case of integer order in [17].
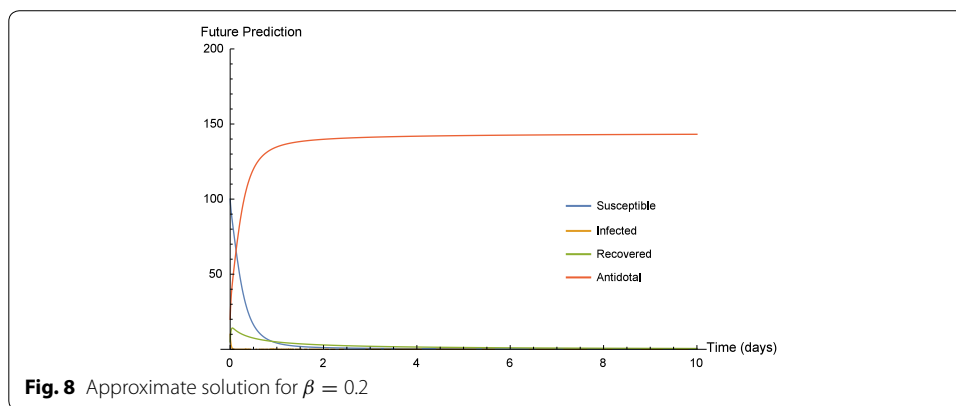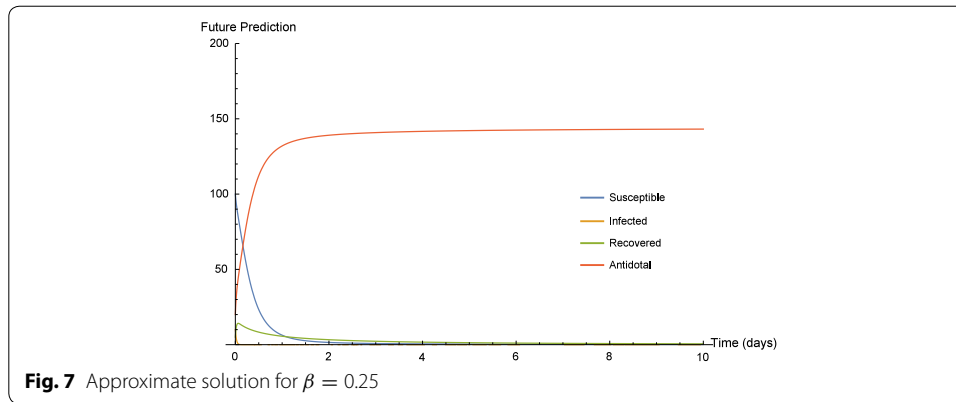
Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 12 of 15

Fig. 1 Approximate solution for $\alpha = 0.105$

Fig. 2 Approximate solution for $\alpha = 0.105$

Fig. 3 Approximate solution for $\alpha = 0.105$

Conclusions

The concept of beta-derivative and Caputo fractional derivative has assisted in investigating the spread of computer virus in a system. This computer virus has been found all over the world where computers are available and causing major financial losses to many establishments. It is worthy to note that the definition of fractional derivative is associated with the convolution of the derivative of a given function with its function power. Convolution is applied to many branches of engineering including image processing as a filter. Fractional derivative, however, in epidemiology serves a memory capable of tracing the spread from beginning to the infected individual. For beta-derivative which

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 13 of 15

Fig. 4 Approximate solution for $\beta = 1$

Fig. 5 Approximate solution for $\beta = 0.65$

Fig. 6 Approximate solution for $\beta = 0.3$

ranges between fractional order and local derivative, the spread of computer virus at local level is identified with a given fractional order. In this study, two distinct concepts of derivatives are employed to investigate the spread of computer virus. The proposed model based on the methodology used was solved iteratively. The numerical simulation results depict that the prediction is based on the fractional order of beta. Simply when beta is close to 1, we obtain non-realistic prediction which is not the case in [10] and when beta is ≤0.5, a good prediction is attained. Since it is the desire of any institution to have their computers with virus free, the initial computers at the end of the simulation moved into Anodotal section when beta is <0.5 as observed in Figs. 4, 5, 6, 7 and 8.

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 14 of 15

Fig. 7 Approximate solution for $\beta = 0.25$

Fig. 8 Approximate solution for $\beta = 0.2$

It worthy to notice that when beta is 1, we have ordinary differential derivative cases for both derivatives which do not provide a good prediction. Thus, with the newly introduced, beta-calculus has the potential of providing a vivid account of physical problem more precisely.

Authors' contributions
All the authors have contributed equally for the production of this study. All authors read and approved the final manuscript.

Author details
[1] Department of Mathematics and Statistics, Kumasi Technical University, Kumasi, Ghana. [2] Institute for Groundwater Studies, University of the Free State, Bloemfontein 9301, South Africa. [3] Department of Mathematics Abdul Wali Khan, University Mardan, Mardan 23200, Pakistan.

References
1. Denning PJ (1990) Computers under attack. Addison-Wesley, New York. http://www.washtech.com/news/netarch/12267-1.html
2. Forrest S, Hofmayer SA, Somayaj A (1997) Computer immunology. Commun ACM 40(10):88–96

Bonyah *et al. Asia Pac. J. Comput. Engin.* (2017) 4:1

Page 15 of 15

3. Kephart JO, Hogg T, Huberman BA (1989) Dynamics of computational ecosystems. Phys Rev A 40(1):404–421
4. Kephart JO, White SR, Chess DM (1993) Computers and epidemiology. IEEE Spectr 30(5):20–26
5. Kephart JO, Sorkin GB, Swimmer M (1997) An immune system for cyberspace. In: Proceedings of the IEEE international conference on systems, men, and cybernetics. IEEE, Orlando, p 879–884
6. Mishra BK, Saini D (2007) Mathematical models on computer viruses. Appl Math Comput 187(2):929–936
7. Mishra BK, Jha N (2007) Fixed period of temporary immunity after run of the anti-malicious software on computer nodes. Appl Math Comput 190:1207–1212
8. Draief M, Ganesh A, Massouili L (2008) Thresholds for virus spread on networks. Ann Appl Probab 18(2):359–378
9. Piqueira JRC, Navarro BF, Monteiro LHA (2005) Epidemiological models applied to viruses in computer networks. J Comput Sci 1:31–34
10. Piqueira JRC, Araujo VO (2009) A modified epidemiological model for computer viruses. Appl Math Comput 2(213):355–360
11. Wazwaz AM (2005) Adomian decomposition method for a reliable treatment of the Bratu-type equations. Appl Math Comput 166:652–663
12. Atangana A, Bildik N (2013) The use of fractional order derivative to predict the groundwater flow. Math Probl Eng 2013:9
13. Cloot A, Botha JF (2005) A generalised groundwater flow equation using the concept of non-integer order derivatives. Water SA 32(1):1–7
14. Atangana A, Vermeulen PD (2014) Analytical solutions of a space-time fractional derivative of groundwater flow equation. Abstr Appl Anal 2014:11
15. Su W, Baleanu D, Yang X, Jafari H (2013) Damped wave equation and dissipative wave equation in fractal strings within the local fractional variational iteration method. Fixed Point Theory Appl 2013(1):1
16. Miller KS, Ross B (1993) An introduction to the fractional calculus and fractional differential equations. Wiley, New York
17. Podlubny I (2002) Geometric and physical interpretation of fractional integration and fractional differentiation. Fract Calc Appl Anal 5:367–386
18. Hethcote HW (2000) The mathematics of infectious diseases. SIAM Rev 42(4):599–653
19. Ganji DD, Sadighi A (2007) Application of homotopy-perturbation and variational iteration methods to nonlinear heat transfer and porous media equations. J Comput Appl Math 207:24–34
20. Singh J, Devendra Kumar S (2011) Homotopy perturbation Sumudu transform method for nonlinear equations. Adv Appl Mech 4:165–175
21. Adomian G, Rach R (1993) Analytic solution of nonlinear boundary value problems in several dimensions by decomposition. J Math Anal Appl 174:118–137
22. Atangana A, Bildik N (2013) Approximate solution of Tuberculosis disease population dynamics model. Appl Anal 2013:1–8
23. Jafari H, Momani S (2007) Solving fractional diffusion and wave equations by modified homotopy perturbation method. Phys Lett A 370:388–396
24. Khan Y (2009) An effective modification of the Laplace decomposition method for nonlinear equations. Int J Nonlinear Sci Num Simul 10:1373–1376