Advances in Difference Equations
a SpringerOpen Journal

RESEARCH

Open Access

A numerical solution by alternative Legendre polynomials on a model for novel coronavirus (COVID-19)

Elham Hashemizadeh[1*] and Mohammad Ali Ebadi[2]

*Correspondence:
hashemizadeh@kiau.ac.ir
[1] Department of Mathematics, Karaj
Branch, Islamic Azad University,
Karaj, Iran
Full list of author information is
available at the end of the article

Abstract

Coronavirus disease (COVID-19) is an infectious disease caused by a newly discovered coronavirus. This paper provides a numerical solution for the mathematical model of the novel coronavirus by the application of alternative Legendre polynomials to find the transmissibility of COVID-19. The mathematical model of the present problem is a system of differential equations. The goal is to convert this system to an algebraic system by use of the useful property of alternative Legendre polynomials and collocation method that can be solved easily. We compare the results of this method with those of the Runge–Kutta method to show the efficiency of the proposed method.

Keywords: Coronavirus; COVID-19; Operational matrix of derivative; Alternative Legendre polynomials

1 Introduction

An outbreak of the 2019 novel coronavirus disease (COVID-19) in Wuhan, China has spread quickly nationwide. The COVID-19 epidemic has spread very quickly from China to all the world [1, 2]. Countries continue to battle the novel coronavirus as it has infected more than 28 million around the world [3].

In [4] the COVID-19 mathematical model has been derived as follows, where $S_p(t)$ is susceptible people, $E_p(t)$ is exposed people, $I_p(t)$ is symptomatic infected people, $A_p(t)$ is asymptomatic infected people, $R_p(t)$ is recovered and dead people, and $W(t)$ is COVID-19 in reservoir in time t. The parameters needed are defined in Table 1 and $\Lambda_p = n_p \times N_p$, where N_p refers to the total number of people:

$$
\begin{cases}
\frac{dS_p}{dt} = \Lambda_p - m_p S_p - \beta_p S_p (I_p + kA_p) - \beta_W S_p W, \\
\frac{dE_p}{dt} = \beta_p S_p (I_p + kA_p) + \beta_W S_p W - (1 - \delta_p)\omega_p E_p - \delta_p \omega'_p E_p - m_p E_p, \\
\frac{dI_p}{dt} = (1 - \delta_p)\omega_p E_p - (\gamma_p + m_p)I_p, \\
\frac{dA_p}{dt} = \delta_p \omega'_p E_p - (\gamma'_p + m_p)A_p, \\
\frac{dR_p}{dt} = \gamma_p I_p + \gamma'_p A_p - m_p R_p, \\
\frac{dW}{dt} = \mu_p I_p + \mu'_p A_p - \varepsilon W.
\end{cases}
\tag{1}
$$

Springer

Table 1 Definition of the parameters in the COVID-19 model

Variables and parameters	Descriptions
n_p	The birth rate of people
m_p	The death rate of people
$\frac{1}{\omega_p}$	The incubation period of people
$\frac{1}{\omega_p'}$	The latent period of people
$\frac{1}{\gamma_p}$	The infectious period of symptomatic infection in people
$\frac{1}{\gamma_p'}$	The infectious period of asymptomatic infection in people
μ_p	The shedding coefficients from I_p to W
μ_p'	The shedding coefficients from A_p to W
δ_p	The proportion of asymptomatic infection rate of people
β_p	The transmission rate from I_p to S_p
β_W	The transmission rate from W to S_p
k	The multiple of the transmissibility of A_p to that of I_p
$\frac{1}{\varepsilon}$	The lifetime of the virus in W
c	The relative shedding rate of A_p compared to I_p

This paper aims to find the transmissibility of the COVID-19 by finding the unknowns S_p, E_p, I_p, A_p, R_p, and W. In medical sciences, the computation of these variables is vital to measure the progression of disease and to get a better cure.

In this paper, for finding these variables, we use alternative Legendre polynomials and their operational matrix of derivative. The proposed method results are compared to those of Runge–Kutta method, which shows the reliability of the proposed method.

There exist some related papers on this topic that have solved the coronavirus model or some differential equation system that appears in the disease model, so we refer the readers to them to see some similar methods on this topic [5–8].

The remainder of the article is organized as follows. In Sect. 2, we review the properties of alternative Legendre polynomials and approximation of a function with them. Then we present the operational matrix of derivatives of these polynomials. In Sect. 3, we implement the alternative Legendre polynomials method on the coronavirus model. Section 4 shows the applicability of the proposed method through a test problem, also the results are compared with Runge–Kutta method results that confirm the reliability of the proposed method. Then Sect. 5 concludes the paper.

## 2 Some basic concepts of alternative Legendre polynomials (ALPs)
2.1 Properties of ALPs
The set $P_n = \{P_{nk} : k = 0, 1, \ldots, n\}$ of alternative Legendre polynomials of degree n is defined by an explicit formula on the interval $[0, 1]$ (see [9]) as follows:

$$P_{nk}(t) = \sum_{j=0}^{n-k} (-1)^j \binom{n-k}{j} \binom{n+k+j+1}{n-k} t^{k+j}, \quad k = 0, 1, \ldots, n. \tag{2}$$

They are orthogonal on the interval $[0, 1]$ with the weight function $w(t) = 1$. The ALPs satisfy the orthogonality relationships

$$\int_0^1 P_{nk}(t)P_{nl}(t)\,dt = \begin{cases} \frac{1}{k+l+1}, & k = l, \\ 0, & k \neq l, \end{cases} \quad k, l = 0, 1, \ldots, n. \tag{3}$$

Figure 1 Plot of the ALPs with $n = 3$ on the interval $[0, 1]$
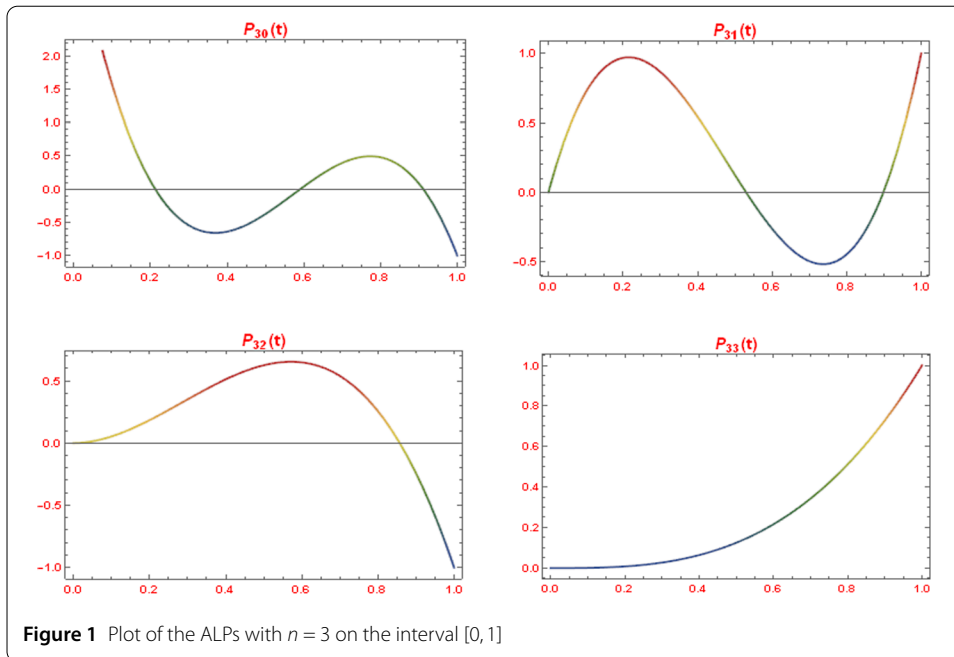
We can reproduce Eq. (2) with Rodrigues's type as follows:

$$P_{nk}(t) = \frac{1}{(n-k)!} \frac{1}{t^{k+1}} \frac{d^{n-k}}{dt^{n-k}} \left(t^{n+k+1}(1-t)^{n-k} \right),$$

$$k = 0, 1, \ldots, n.$$

(4)

So, we have

$$\int_0^1 P_{nk}(t)\,dt = \int_0^1 t^n\,dt = \frac{1}{n+1}, \quad k = 0, 1, \ldots, n.$$

(5)

Here, we note that each element of the set $P_n = \{P_{nk}\}_{k=0}^n$ is the polynomial of other n. For example, in the following we introduce the alternative Legendre polynomials $P_3 = \{P_{nk}\}_{k=0}^3$ ($n = 3$).

$$P_{30}(t) = 4 - 30t + 60t^2 - 35t^3, \qquad P_{31}(t) = 10t - 30t^2 + 21t^3,$$

$$P_{32}(t) = 6t^2 - 7t^3, \qquad P_{33}(t) = t^3.$$

In Fig. 1, we display the 4 set of ALPs with $n = 3$ over the interval $[0, 1]$.

2.2 Function approximation

Consider $P_n = \{P_{nk}\}_{k=0}^n \subset H = L^2[0, 1]$ to be a set of ALPs and suppose that $Y = \mathrm{Span}\{P_{nk}(t) : k = 0, 1, \ldots, n\}$. So, Y is a finite dimensional subspace of H. Suppose, f to be an arbitrary function in H. Therefore, based on the Weierstrass theorem, every continuous function $f(t)$ on the interval $[a, b]$ can be uniformly approximated by a polynomial function [9]. So, f has a unique best approximation in Y that we call $f^*(t)$. We have

$$\left\| f(t) - f^*(t) \right\|_2 \le \left\| f(t) - y(t) \right\|_2 : \quad \forall y(t) \in Y.$$

(6)

Then this implies that

$$\langle y, f - f^* \rangle = 0 : \quad \forall y(t) \in Y, \tag{7}$$

where $\langle \cdot, \cdot \rangle$ denotes an inner product. Therefore, any arbitrary function $f \in H = L^2[0,1]$ may be approximated in terms of ALPs. So, there exists a set of unique coefficients $\{c_k : k = 0, 1, \ldots, n\}$ such that

$$f(t) \approx f^*(t) = \sum_{k=0}^{n} c_k P_{nk}(t), \tag{8}$$

coefficient c_k can be obtained in the following form:

$$c_k = \frac{\langle f, P_{nk} \rangle}{\langle P_{nk}, P_{nk} \rangle} = (2k+1)\langle f, P_{nk} \rangle, \quad k = 0, 1, \ldots, n, \tag{9}$$

and

$$\langle f, f \rangle = \int_0^1 f^2(t)\, dt. \tag{10}$$

Also, Eq. (8) can be written in a matrix form as follows:

$$f(t) \simeq \sum_{k=0}^{n} c_k P_{nk}(t) = C^T \Phi(t), \tag{11}$$

where

$$C = [c_0, c_1, \ldots, c_n] \tag{12}$$

and

$$\Phi(t) = \left[P_{n0}(t), P_{n1}(t), \ldots, P_{nn}(t) \right]^T. \tag{13}$$

Let $a_{kj}^{(n)} = (-1)^j \binom{n-k}{j} \binom{n+k+j+1}{n-k}$, then Eq. (2) can be written as

$$P_{nk}(t) = \sum_{j=0}^{n-k} a_{kj}^{(n)} t^{k+j}, \quad k = 0, 1, \ldots, n. \tag{14}$$

By using Eq. (14), for $k = 0, 1, \ldots, n$ now, we can write

$$
\Phi(t) = \begin{bmatrix} P_{n0}(t) \\ P_{n1}(t) \\ \vdots \\ P_{nn}(t) \end{bmatrix} = \begin{bmatrix} \sum_{j=0}^{n} a_{0j}^{(n)} t^j \\ \sum_{j=0}^{n-1} a_{1j}^{(n)} t^{j+1} \\ \vdots \\ \sum_{j=0}^{n-n} a_{nj}^{(n)} t^{j+n} \end{bmatrix} = \begin{bmatrix} a_{00}^{(n)} + a_{01}^{(n)} t + a_{02}^{(n)} t^2 + \cdots + a_{0n}^{(n)} t^n \\ a_{01}^{(n)} t + a_{02}^{(n)} t^2 + \cdots + a_{0(n-1)}^{(n)} t^n \\ \vdots \\ 0 \end{bmatrix}
$$

$$
= \begin{bmatrix} a_{00}^{(n)} & a_{01}^{(n)} & \cdots & a_{0n}^{(n)} \\ 0 & a_{01}^{(n)} & \cdots & a_{0(n-1)}^{(n)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn}^{(n)} \end{bmatrix} \begin{bmatrix} 1 \\ t \\ \vdots \\ t^n \end{bmatrix} = \Phi X_t. \tag{15}
$$

Therefore Eq. (13) can be written in the following form:

$$
\Phi(t) = \Phi X_t, \tag{16}
$$

where

$$
X_t = \left[1, t, t^2, \ldots, t^n \right]^T, \tag{17}
$$

and Φ is the upper triangular matrix defined by [10]

$$
\Phi = [q_{kj}], \quad k, j = 0, 1, \ldots, n,
$$

$$
q_{kj} = \begin{cases} 0, & 0 \leq j < k, \\ (-1)^{j-k} \binom{n-k}{j-k} \binom{n+j+1}{n-k}, & k \leq j \leq n. \end{cases} \tag{18}
$$

Definition The tensor product of two vectors $f_{\hat{m}} = [f_i]$ and $g_{\hat{m}} = [g_i]$ is defined as

$$
f \otimes g = (f_i \times g_i)_{\hat{m}}. \tag{19}
$$

Similarly, for two matrices $A = [a_{i,j}]$ and $B = [b_{i,j}]$ of $\hat{m} \times \hat{m}$,

$$
A \otimes B = (a_{i,j} \times b_{i,j})_{\hat{m} \times \hat{m}}. \tag{20}
$$

The lemma below will be needed in Sect. 3.

Lemma *Let the functions $f(t), g(t) \in L^2[0,1]$ be expanded into ALPs, that is, $f(t) = f^T \Phi(t)$ and $g(t) = g^T \Phi(t)$. Then*

$$
f(t)g(t) = \left(f^T \otimes g^T \right) \Phi(t). \tag{21}
$$

Proof

$$
f(t)g(t) = f^T \Phi(t) \Phi^T(t) g = f_1 g_1 p_{n0}(t) + f_2 g_2 p_{n1}(t) + \cdots + f_n g_n p_{nn}(t) = \left(f^T \otimes g^T \right) \Phi(t). \quad \square
$$

2.3 Operational matrix of derivative

In this section, we derive the operational matrix of derivative of the ALPs that plays an important role in simplifying a system of differential equations and implementation of the proposed method.

To compute this operational matrix, we need to introduce the following properties of ALPs that can easily be deduced from the given definitions. Let $P_{ni}(t) = \sum_{r=0}^{n} p_r^{(i)} t^r$, $P_{nj}(t) = \sum_{r=0}^{n} p_r^{(j)} t^r$, and $P_{nk}(t) = \sum_{r=0}^{n} p_r^{(k)} t^r$ be ith, jth, and kth of ALPs, respectively. Therefore, we have

- $$\int_0^1 t^r P_{nk}(t)\, dt = \sum_{l=0}^{n-k} \frac{(-1)^l \binom{n-k}{l}\binom{n+k+l+1}{n-k}}{k+l+r+1}, \quad k = 0, 1, \ldots, n. \tag{22}$$

- $$P_{nk}(t) P_{nj}(t) = \sum_{r=0}^{2n} q_r^{(k,j)} t^r, \qquad q_r^{(k,j)} = \begin{cases} \sum_{l=0}^{r} p_l^{(k)} p_{r-l}^{(j)}, & r \leq n, \\ \sum_{l=r-n}^{n} p_l^{(k)} p_{r-l}^{(j)}, & r > n, \end{cases} \tag{23}$$

- $$\int_0^1 P_{ni}(t) P_{nj}(t) P_{nk}(t)\, dt = \sum_{r=0}^{2n} q_r^{(k,j)} \sum_{l=0}^{n-i} \frac{(-1)^l \binom{n-i}{l}\binom{n+i+l+1}{n-i}}{i+l+r+1}. \tag{24}$$

The derivative of the vector $\Phi(t)$ can be expressed by

$$\frac{d\Phi(t)}{dt} = D^{(1)}\Phi(t). \tag{25}$$

Here, $D^{(1)}$ is the $(n+1) \times (n+1)$ operational matrix of derivative.

So, by applying the differential operator with respect to t, we can write $D_t = \frac{d}{dt}$ (see[11]). By applying the polynomial $P_{nk}(t)$, we obtain

$$\begin{aligned}
D_t p_{nk}(t) &= D_t \sum_{j=0}^{n-k} (-1)^j \binom{n-k}{j}\binom{n+k+j+1}{n-k} t^{k+j} \\
&= \sum_{j=0}^{n-k} (-1)^j \binom{n-k}{j}\binom{n+k+j+1}{n-k} D_t t^{k+j} \\
&= \sum_{j=0}^{n-k} (-1)^j (k+j) \binom{n-k}{j}\binom{n+k+j+1}{n-k} t^{k+j-1}, \\
& k = 0, 1, \ldots, n.
\end{aligned} \tag{26}$$

Here, by using Eq. (11), one can approximate t^{k+j-1} in terms of ALPs as follows:

$$t^{k+j-1} \simeq \sum_{r=0}^{n} b_r^{(k,j)} p_{nr}(t). \tag{27}$$

The approximation coefficients $b_r^{(k,j)}$ are obtained using Eq. (9) as follows:

$$
\begin{aligned}
b_r^{(k,j)} &= (2r+1) \int_0^1 y(t) P_{nr} \, dt \\
&= (2r+1) \int_0^1 t^{k+j-1} \sum_{l=0}^{n-r} (-1)^l \binom{n-r}{l} \binom{n+r+l+1}{n-r} t^{r+l} \, dt \\
&= (2r+1) \sum_{l=0}^{n-r} \frac{(-1)^l \binom{n-r}{l}\binom{n+r+l+1}{n-r}}{k+j+r+l}, \quad r = 0, 1, \dots, n.
\end{aligned}
\tag{28}
$$

Substituting (27) into (26), we have

$$
D_t p_{nk}(t) = \sum_{j=0}^{n-k} (-1)^j \binom{n-k}{j} \binom{n+k+j+1}{n-k} \frac{\Gamma(k+j+1)}{\Gamma(k+j+\alpha+1)} \sum_{r=0}^n b_r^{(k,j)} p_{nr}(t),
\tag{29}
$$

$$
k = 0, 1, \dots, n.
$$

Then, using Eqs. (26) and (27), we have

$$
\begin{aligned}
&D_t p_{nk}(t) \\
&= \sum_{r=0}^n (2r+1) \\
&\quad \times \left[\sum_{j=0}^{n-k} (-1)^j (k+j) \binom{n-k}{j} \binom{n+k+j+1}{n-k} \sum_{l=0}^{n-r} \frac{(-1)^l \binom{n-r}{l}\binom{n+r+l+1}{n-r}}{k+j+r+l} \right] p_{nr}(t) \\
&= \sum_{r=0}^n \theta_{kr}^{(1)} p_{nr}(t)
\end{aligned}
\tag{30}
$$

hence

$$
\theta_{kr}^{(1)} = (2r+1) \left[\sum_{j=0}^{n-k} (-1)^j (k+j) \binom{n-k}{j} \binom{n+k+j+1}{n-k} \sum_{l=0}^{n-r} \frac{(-1)^l \binom{n-r}{l}\binom{n+r+l+1}{n-r}}{k+j+r+l} \right].
\tag{31}
$$

Therefore, for the vector $\Phi(t)$ defined by (13), we get

$$
\frac{d\Phi(t)}{dt} = D_t \Phi(t) = D^{(1)} \Phi(t),
\tag{32}
$$

where $D^{(1)}$ is the $(n+1) \times (n+1)$ operational matrix of derivative based on the ALPs as follows:

$$
D^{(1)} = \left[\theta_{kr}^{(1)} \right], \quad k, r = 0, 1, \dots, n.
\tag{33}
$$

3 Implementation of an alternative Legendre polynomials method on the novel coronavirus (COVID-19) problem

Firstly, note that the variable of system (1) becomes normalized as follows [4]:

$$s_p = \frac{S_p}{N_p}, \qquad e_p = \frac{E_p}{N_p}, \qquad i_p = \frac{I_p}{N_p}, \qquad a_p = \frac{A_p}{N_p}, \qquad r_p = \frac{R_p}{N_p}, \qquad w = \frac{\varepsilon W}{\mu_p N_p},$$

$$\mu_p' = c\mu_p, \qquad b_p = \beta_p N_p, \qquad b_W = \frac{\mu_p \beta_W N_p}{\varepsilon}.$$

So, the normalized model is changed as follows:

$$
\begin{cases}
\frac{ds_p}{dt} = n_p - m_p s_p - b_p s_p(i_p + ka_p) - b_W s_p w, \\
\frac{de_p}{dt} = b_p s_p(i_p + ka_p) + b_W s_p w - (1 - \delta_p)\omega_p e_p - \delta_p \omega_p' e_p - m_p e_p, \\
\frac{di_p}{dt} = (1 - \delta_p)\omega_p e_p - (\gamma_p + m_p)i_p, \\
\frac{da_p}{dt} = \delta_p \omega_p' e_p - (\gamma_p' + m_p)a_p, \\
\frac{dr_p}{dt} = \gamma_p i_p + \gamma_p' a_p - m_p r_p, \\
\frac{dw}{dt} = \varepsilon(i_p + ca_p - w)
\end{cases}
\tag{34}
$$

with the initial conditions

$$s_p(0) = s_0, \qquad e_p(0) = e_0, \qquad i_p(0) = i_0, \qquad a_p(0) = a_0, \qquad r_p(0) = r_0, \qquad w(0) = w_0.$$

The main objective of this paper is to implement ALPs approach on the system of differential Eqs. (34) with the above initial conditions to find the numerical solution of this system. From Eq. (11), we can approximate our unknown functions as follows:

$$
\begin{aligned}
s_p &= C_1^T \Phi(t), & e_p &= C_2^T \Phi(t), & i_p &= C_3^T \Phi(t), \\
a_p &= C_4^T \Phi(t), & r_p &= C_5^T \Phi(t), & w &= C_6^T \Phi(t),
\end{aligned}
\tag{35}
$$

where coefficient vectors $C_i : i = 1, \dots, 6$ that were defined in Eq. (29) are as follows:

$$
\begin{aligned}
C_1 &= [c_0, \dots, c_n]^T, & C_2 &= [c_{n+1}, \dots, c_{2n+1}]^T, & C_3 &= [c_{2n+2}, \dots, c_{3n+2}]^T, \\
C_4 &= [c_{3n+3}, \dots, c_{4n+3}]^T, & C_5 &= [c_{4n+4}, \dots, c_{5n+4}]^T, & C_6 &= [c_{5n+5}, \dots, c_{6n+5}]^T.
\end{aligned}
\tag{36}
$$

By using Eqs. (34) and (32), we have

$$
\begin{cases}
\frac{ds_p}{dt} = C_1^T \Phi'(t) = C_1^T D^{(1)} \Phi(t), \\
\frac{de_p}{dt} = C_2^T \Phi'(t) = C_2^T D^{(1)} \Phi(t), \\
\frac{di_p}{dt} = C_3^T \Phi'(t) = C_3^T D^{(1)} \Phi(t), \\
\frac{da_p}{dt} = C_4^T \Phi'(t) = C_4^T D^{(1)} \Phi(t), \\
\frac{dr_p}{dt} = C_5^T \Phi'(t) = C_5^T D^{(1)} \Phi(t), \\
\frac{dw}{dt} = C_6^T \Phi'(t) = C_6^T D^{(1)} \Phi(t).
\end{cases}
\tag{37}
$$

By substituting Eqs. (37) and (35) into the system of differential Eqs. (1), we have

$$
\begin{cases}
C_1^T D^{(1)} \Phi(t) = n_p - m_p C_1^T \Phi(t) - b_p C_1^T \otimes C_3^T \Phi(t) \\
\qquad\qquad - k b_p C_1^T \otimes C_4^T \Phi(t) - b_W C_1^T \otimes C_6^T \Phi(t), \\
C_2^T D^{(1)} \Phi(t) = b_p C_1^T \otimes C_3^T \Phi(t) + k b_p C_1^T \otimes C_4^T \Phi(t) + b_W C_1^T \otimes C_6^T \Phi(t) \\
\qquad\qquad - (1 - \delta_p)\omega_p C_2^T \Phi(t) - \delta_p \omega_p' C_2^T \Phi(t) - m_p C_2^T \Phi(t), \\
C_3^T D^{(1)} \Phi(t) = (1 - \delta_p)\omega_p C_2^T \Phi(t) - (\gamma_p + m_p) C_3^T \Phi(t), \\
C_4^T D^{(1)} \Phi(t) = \delta_p \omega_p' C_2^T \Phi(t) - (\gamma_p' + m_p) C_4^T \Phi(t), \\
C_5^T D^{(1)} \Phi(t) = \gamma_p C_3^T \Phi(t) + \gamma_p' C_4^T \Phi(t) - m_p C_5^T \Phi(t), \\
C_6^T D^{(1)} \Phi(t) = \varepsilon(C_3^T \Phi(t) + c C_4^T \Phi(t) - C_6^T \Phi(t)).
\end{cases}
\tag{38}
$$

Also, by considering the initial conditions for main problem (1) and Eq. (35), we have

$$
\begin{aligned}
& C_1^T \Phi(0) = s_0, \qquad C_2^T \Phi(0) = e_0, \qquad C_3^T \Phi(0) = i_0, \\
& C_4^T \Phi(0) = a_0, \qquad C_5^T \Phi(0) = r_0, \qquad C_6^T \Phi(0) = w_0.
\end{aligned}
\tag{39}
$$

Equation (39) gives six linear equations.

Since the total unknowns for vectors $C_i : i = 1, \ldots, 6$ are $(6n + 5)$, we collocate Eq. (38) in the set of $(6n - 1)$ nodal points t_l of the Guass–Chelyshkov [10] as follows:

$$
Q_n = \left\{ t_l \,|\, P_{n+1,0}(t_l) = 0, l = 0, 1, \ldots, n \right\}.
\tag{40}
$$

Now, by replacing the nodes t_l in Eq. (38),

$$
\begin{cases}
C_1^T D^{(1)} \Phi(t_i) = n_p - m_p C_1^T \Phi(t_i) - b_p C_1^T \otimes C_3^T \Phi(t_i) \\
\qquad\qquad - k b_p C_1^T \otimes C_4^{T \Phi(t_i)} - b_W C_1^T \otimes C_6^T \Phi(t_i), \\
C_2^T D^{(1)} \Phi(t_i) = b_p C_1^T \otimes C_3^T \Phi(t_i) + k b_p C_1^T \otimes C_4^T \Phi(t_i + b_W C_1^T \otimes C_6^T \Phi(t_i) \\
\qquad\qquad - (1 - \delta_p)\omega_p C_2^T \Phi(t_i) - \delta_p \omega_p' C_2^T \Phi(t_i) - m_p C_2^T \Phi(t_i), \\
C_3^T D^{(1)} \Phi(t_i) = (1 - \delta_p)\omega_p C_2^T \Phi(t_i) - (\gamma_p + m_p) C_3^T \Phi(t_i), \\
C_4^T D^{(1)} \Phi(t_i) = \delta_p \omega_p' C_2^T \Phi(t_i) - (\gamma_p' + m_p) C_4^T \Phi(t_i), \\
C_5^T D^{(1)} \Phi(t_i) = \gamma_p C_3^T \Phi(t_i) + \gamma_p' C_4^T \Phi(t_i) - m_p C_5^T \Phi(t_i), \\
C_6^T D^{(1)} \Phi(t_i) = \varepsilon(C_3^T \Phi(t_i) + c C_4^T \Phi(t_i) - C_6^T \Phi(t_i))
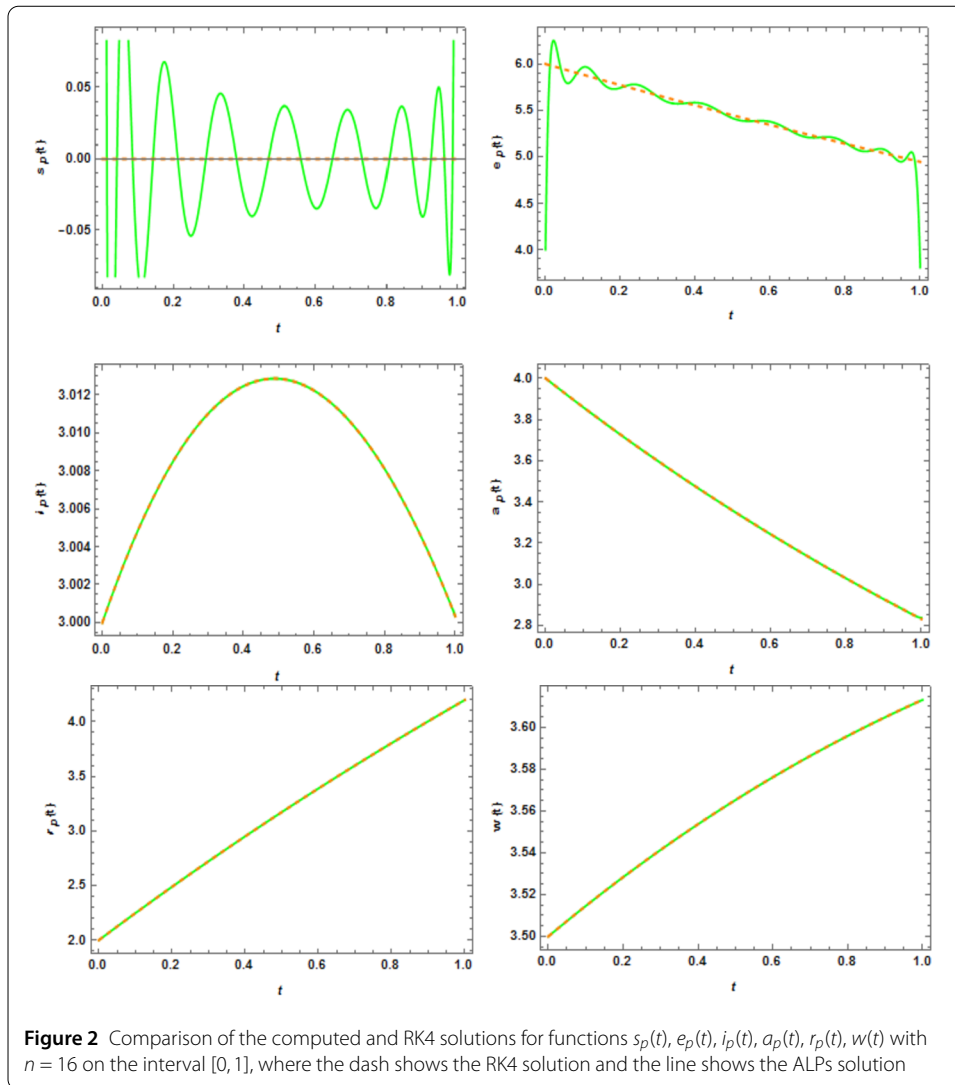\end{cases}
\tag{41}
$$

for $i = 0, \ldots, 6n - 1$, we can solve this system of $6n$ equations that resulted from Eqs. (39) and (41) by using Newton's iteration scheme [12–16] for calculating the unknown vectors $C_i : i = 1, \ldots, 6$.

For existence and stability of the proposed method with ALPs, we can refer to paper [9].

In our implementation, the calculations are done in *Mathematica 11* software, on a personal computer with Core-i5 processor, 2.67 GHZ frequency, and 4 GB memory.

4 Numerical example

In this section a test problem of the coronavirus model is solved by our proposed method.

Figure 2 Comparison of the computed and RK4 solutions for functions $s_p(t)$, $e_p(t)$, $i_p(t)$, $a_p(t)$, $r_p(t)$, $w(t)$ with $n = 16$ on the interval $[0, 1]$, where the dash shows the RK4 solution and the line shows the ALPs solution

The values of the initial conditions and parameters are given as [4]:

$$N_p = 1{,}000{,}000{,}000, \qquad \delta_p = k = \beta_p = \beta_w = \mu_p = \gamma'_p = c = 0.5, \qquad n_p = m_p = 0.0018,$$

$$\varepsilon = 0.1, \qquad w_p = w'_p = 0.1923, \qquad \gamma_p = 0.1724.$$

Also, we get the initial values of unknown parameters as follows:

$$s_p(0) = 2, \qquad e_p(0) = 4, \qquad i_p(0) = 3, \qquad a_p(0) = 4, \qquad r_p(0) = 2, \qquad w(0) = 3.5.$$

We solve this problem by $n = 16$ in ALPs. The results of proposed method are compared with the results of Runge–Kutta method. Figure 2 and Tables 2–7 show the comparison between them.

5 Conclusion

The World Health Organization declared the coronavirus (COVID-19) a pandemic on March 11, 2020. This virus spread quickly in more than 200 countries, and up to now

Table 2 Numerical comparison for $s_p(t)$ with $n = 16$ on the interval $[0, 1]$

t	RK4	Present method
0.0	2.000000000	2.00000000
0.2	1.59905E−13	0.03002862
0.4	1.59871E−13	−0.02923894
0.6	1.59898E−13	−0.03456393
0.8	1.59982E−13	−0.01139075
1	1.60124E−13	1.12482514

Table 3 Numerical comparison for $e_p(t)$ with $n = 16$ on the interval $[0, 1]$

t	RK4	Present method
0.0	4	4
0.2	5.77189614477	5.74277771538
0.4	5.55247760052	5.58266126424
0.6	5.34141367556	5.37685723359
0.8	5.13838625205	5.15034353745
1	4.94308933537	3.80689507404

Table 4 Numerical comparison for $i_p(t)$ with $n = 16$ on the interval $[0, 1]$

t	RK4	Present method
0.0	3	3
0.2	3.00849107875	3.00849108363
0.4	3.01246360734	3.01246360563
0.6	3.01223326389	3.01223326126
0.8	3.00809881420	3.00809880689
1	3.00034288194	3.00034287358

Table 5 Numerical comparison for $a_p(t)$ with $n = 16$ on the interval $[0, 1]$

t	RK4	Present method
0.0	4	4
0.2	3.72569099766	3.725275447302
0.4	3.47348293584	3.473083082484
0.6	3.24142145708	3.241082166196
0.8	3.02773254616	3.027574786909
1	2.83080613397	2.836696723317

Table 6 Numerical comparison for $r_p(t)$ with $n = 16$ on the interval $[0, 1]$

t	RK4	Present method
0.0	2	2
0.2	2.48888268589	2.48882899953
0.4	2.95149948421	2.95140029238
0.6	3.38981976531	3.38968102039
0.8	3.80563689580	3.80546221628
1	4.20058431510	4.20029857270

Table 7 Numerical comparison for $w(t)$ with $n = 16$ on the interval $[0, 1]$

t	RK4	Present method
0.0	3.5	3.5
0.2	3.52841252047	3.52840049359
0.4	3.55378078473	3.55375851466
0.6	3.57628769500	3.57625646333
0.8	3.59610020735	3.59606081175
1	3.61337096445	3.61330663649

more than 28 million around the world have been infected. This paper aims to solve the mathematical model of coronavirus that can show the transmissibility of this virus that is vital to measure the progression of the disease and to get a better cure. By use of alternative Legendre polynomials and their operational matrix of derivative, we convert the system of coronavirus model to an algebraic model. We compare the results of the present method with those of the Runge–Kutta method, which confirmed the reliability of the proposed method results.

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

Authors' contributions
The authors have equal contributions to each part of this paper. All the authors read and approved the final manuscript.

Author details
[1]Department of Mathematics, Karaj Branch, Islamic Azad University, Karaj, Iran. [2]Young Researchers and Elite Club, Karaj Branch, Islamic Azad University, Karaj, Iran.

Publisher's Note

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

References
1. World Health Organization. Coronavirus. World Health Organization, Available at https://www.who.int/health-topics/coronavirus
2. Novel, C.P.: The epidemiological characteristics of an outbreak of 2019 novel coronavirus diseases (COVID-19) in China. Zhonghua liu xing bing xue za zhi= Zhonghua liuxingbingxue zazhi **41**(2), 145 (2020)
3. This site shows Coronavirus cases online, Available at https://www.worldometers.info/coronavirus/
4. Chen, T.M., Rui, J., Wang, Q.P., Zhao, Z.Y., Cui, J.A., Yin, L.: A mathematical model for simulating the phase-based transmissibility of a novel coronavirus. Infect. Dis. Poverty **9**(1), 1–8 (2020)
5. Ullah, S., Khan, M.A.: Modeling the impact of non-pharmaceutical interventions on the dynamics of novel coronavirus with optimal control analysis with a case study. Chaos Solitons Fractals **139**, 110075 (2020)
6. Khan, M.A., Atangana, A.: Modeling the dynamics of novel coronavirus (2019-nCov) with fractional derivative. Alex. Eng. J. **59**(4), 2379–2389 (2020)
7. Atangana, A.: Modelling the spread of COVID-19 with new fractal-fractional operators: can the lockdown save mankind before vaccination? Chaos Solitons Fractals **136**, 109860 (2020)
8. Atangana, A., Araz, S.İ.: New numerical method for ordinary differential equations: Newton polynomial. J. Comput. Appl. Math. **372**, 112622 (2020)
9. Chelyshkov, V.S.: Alternative orthogonal polynomials and quadratures. Electron. Trans. Numer. Anal. **25**(7), 17–26 (2006)
10. Meng, Z., Yi, M., Huang, J., Song, L.: Numerical solutions of nonlinear fractional differential equations by alternative Legendre polynomials. Appl. Math. Comput. **336**, 454–464 (2018)
11. Ebadi, M.A., Hashemizadeh, E.: A new approach based on the Zernike radial polynomials for numerical solution of the fractional diffusion-wave and fractional Klein–Gordon equations. Phys. Scr. **93**(12), 125202 (2018)
12. Maleknejad, K., Hashemizadeh, E.: A numerical approach for Hammerstein integral equations of mixed type using operational matrices of hybrid functions. Sci. Bull. "Politeh." Univ. Buchar., Ser. A, Appl. Math. Phys., **73**, 95–104 (2011)
13. Maleknejad, K., Hashemizadeh, E., Basirat, B.: Computational method based on Bernstein operational matrices for nonlinear Volterra–Fredholm–Hammerstein integral equations. Commun. Nonlinear Sci. Numer. Simul. **17**(1), 52–61 (2012)
14. Maleknejad, K., Hashemizadeh, E.: Numerical solution of nonlinear singular ordinary differential equations arising in biology via operational matrix of shifted Legendre polynomials. Am. J. Comput. Appl. Math. **1**(1), 15–19 (2011)
15. Ebadi, M.A., Hashemizadeh, E.S., Refahi Sheikhani, A.H.: Zernike radial polynomials method for solving nonlinear singular boundary value problems arising in physiology. J. New Res. Math. **5**(19), 139–150 (2020)
16. Maleknejad, K., Hashemizadeh, E.: Numerical solution of the dynamic model of a chemical reactor by hybrid functions. Proc. Comput. Sci. **3**, 908–912 (2011)