Generalized differential transform method to differential-difference equation

Li Zou\textsuperscript{a,}\textsuperscript{*}, Zhen Wang\textsuperscript{b,c}, Zhi Zong\textsuperscript{d}

\textbf{Article history:}
Received 22 May 2009
Received in revised form 13 September 2009
Accepted 15 September 2009
Available online 19 September 2009
Communicated by A.R. Bishop

\textbf{Abstract}
In this Letter, we generalize the differential transform method to solve differential-difference equation for the first time. Two simple but typical examples are applied to illustrate the validity and the great potential of the generalized differential transform method in solving differential-difference equation. A Padé technique is also introduced and combined with GDTM in aim of extending the convergence area of presented series solutions. Comparisons are made between the results of the proposed method and exact solutions. Then we apply the differential transform method to the discrete KdV equation and the discrete mKdV equation, and successfully obtain solitary wave solutions. The results reveal that the proposed method is very effective and simple. We should point out that generalized differential transform method is also easy to be applied to other nonlinear differential-difference equations.

\textbf{Keywords:}
Differential-difference equation
Differential transform method
Generalized differential transform method
Differential transform-Padé approximation

\section{1. Introduction}

The concept of differential transform was first introduced by Pukhov [1], who solved linear and nonlinear initial value problems in electric circuit analysis. Zhou [2] had also separately studied differential transform method at the same time with Pukhov. The differential transform method obtains an analytical solution in the form of a polynomial. It is different from the traditional high order Taylor’s series method, which requires symbolic competition of the necessary derivatives of the data functions. The Taylor series method is computationally taken long time for large orders. Chen and Ho have developed this method for PDEs and obtained closed form series solutions for linear and nonlinear initial value problems [3]. Ayaz developed differential transform method to two-dimensional problem for PDEs initial value problems [4,5]. Kurnaz et al. generalized DTM to \( n \)-dimensional case in order to solve PDEs [6]. Besides the differential transform method is independent on whether or not there exist small parameters in the considered equation. Therefore, the differential transform method can overcome the foregoing restrictions and limitations of perturbation techniques so that it provides us with a possibility to analyze strongly nonlinear problems. This method has been successfully applied to solve many types of nonlinear problems [7–10]. However, the application of DTM only circumscribes integral-differential equation. Here, we generalized the method to nonlinear difference-differential equations.

To cope with the rapid development of science and technology, there is an increasing need to solve complicated nonlinear differential-difference equations (DDEs). The DDEs play an important role in modelling complicated physical phenomena such as particle vibrations in lattices, currents flow in electrical networks, and pulses in biological chains. The solutions of these DDEs can provide numerical simulations.
of nonlinear partial differential equations, queuing problems, and discretizations in solid state and quantum physics. Since the works of Fermi, Pasta, and Ulam in the 1950s [11], there were quite a number of research works developed during the last decades on DDEs. For instance, Levi and his co-workers analyzed the condition for existence of higher symmetries for a class of DDEs [12,13], Yamilov and his co-workers [14,15] made outstanding contribution to the classification of DDEs, integrability tests and connections between integrable PDEs and DDEs [16,17] and a lot of works were developed to analyze the properties of solutions of DDEs [18,19].

The aim of this Letter is to directly extend the DTM to consider the numerical solution of the differential-difference equations. The Letter has been organized as follows. In Section 2, we extended the differential transform method to solving nonlinear differential-difference equation.

## 2. Generalized DT method for DDEs

### 2.1. Basic idea

In the Letter, we apply the differential transform method to the discussed problem. We extend the basic idea of DTM to the differential-difference equation.

Let us consider the most general form of algebraic difference-differential equation

\[ \mathcal{N}(u_n(t), u_{n+1}(t), u_{n-1}(t), u_{n+2}(t), u_{n-2}(t), \ldots) = 0 \]

where \( \mathcal{N} \) is a nonlinear differential operator, \( n \) and \( t \) denote independent variables, \( u_n(t) \) is unknown functions in vector form. For simplicity, we ignore all boundary or initial conditions, which can be treated in the same way. Based on the basic theorems of the one-dimensional differential transform method, the differential transform of the kth derivative of a function \( u_n(t) \) is defined as follows:

\[ U_n(k) = \left. \frac{d^ku_n(t)}{dt^k} \right|_{t=0} \]

where \( u_n(t) \) is the original function and \( U_n(k) \) is the transformed function.

The differential inverse transform of \( U_n(k) \) is defined as

\[ u_n(t) = \sum_{k=0}^{\infty} U_n(k)(t-t_0)^k = \sum_{k=0}^{\infty} U(n,k)(t-t_0)^k. \]  

In a real application, when \( t_0 \) is taken as 0, then the function \( u_n(t) \) can be expressed by a finite series and with the aid of Eq. (3), \( u_n(t) \) can be written as

\[ u_n(t) = \sum_{k=0}^{M} U(n,k)t^k = \sum_{k=0}^{M} \left. \frac{d^k u_n(t)}{dt^k} \right|_{t=0} t^k. \]

The fundamental mathematical operations performed by generalized differential transform method are listed in the following Table 1.

<table>
<thead>
<tr>
<th>Original function</th>
<th>Transformed function</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f(n,t) = g(n,t) + h(n,t) )</td>
<td>( F(n,k) = G(n,k) + H(n,k) )</td>
</tr>
<tr>
<td>( f(n,t) = \alpha g(n,t) )</td>
<td>( F(n,k) = \alpha G(n,k) )</td>
</tr>
<tr>
<td>( f(n,t) = \frac{d^n g(n,t)}{d^t^n} )</td>
<td>( F(n,k) = (k+1)G(n,k+1) )</td>
</tr>
<tr>
<td>( f(n,t) = g(n,t)h(n,t) )</td>
<td>( F(n,k) = \sum_{r=0}^{k} G(n,r)h(n,k-r) )</td>
</tr>
<tr>
<td>( f(n,t) = d^n g(n,t) )</td>
<td>( F(n,k) = (k+m)G(n,k+m) )</td>
</tr>
<tr>
<td>( f(n,t) = g(n + s, t) )</td>
<td>( F(n,k) = G(n + s, k) )</td>
</tr>
</tbody>
</table>

### 2.2. Differential transform-Padé technique

The accuracy and convergence of the solution given by series Eq. (5) can be further enhanced by the differential transform-Padé technique. The basic idea of summation theory is to represent \( f(x) \), the function in question, by a convergent expression. In Euler summation this expression is the limit of the convergent series, while in Borel summation this expression is the limit of a convergent integral. The difficulty with Euler and Borel summation is that all of the terms of the divergent series must be known exactly before the sum can be...
found even approximately. But in real computation, only a few terms of a series can be calculated before a state of exhaustion is reached. Therefore, a summation algorithm is needed which requires as input only a finite number of terms of divergent series. Then as each new term is given, we can give a new and improved estimate of exact sum of the divergent series. Padé approximation is a well-known summation method which having this property.

As a method of enhancing accuracy and convergence of the series, Padé approximation is widely applied [20–22]. The idea of Padé summation is to replace a power series

\[ f(x) = \sum_{n=0}^{+\infty} c_n x^n \]

by a sequence of rational functions which is a ratio of two polynomials

\[ p_N(x) = \frac{\sum_{k=0}^{N} a_k x^k}{\sum_{k=0}^{M} b_k x^k} \]

where we choose \( b_0 = 1 \) without loss of generality. We choose the remaining \((M + N + 1)\) coefficients \( a_0, a_1, \ldots, a_N, b_1, b_2, \ldots, b_M\), so that the first \((M + N + 1)\) terms in the power series expansion of \( p_N(x) \) match the first \((M + N + 1)\) terms of the power series \( f(x) = \sum_{n=0}^{+\infty} c_n x^n \). The resulting rational function \( p_N(x) \) is called a Padé approximant. We will see that constructing \( p_N(x) \) is very useful. If \( \sum c_n x^n \) is a power series representation of the function \( f(x) \), then in many instances \( p_N(x) \) \( f(x) \) as \( N, M \to \infty \), even if \( \sum c_n x^n \) is a divergent series. Usually we consider only the convergence of the Padé sequences \( f_0^J, f_1^J, f_2^J, \ldots \) having \( N = M + J \) with \( J \) fixed and \( M \to \infty \). If \( J = 0 \) then this sequence is called diagonal sequence.

It often works quite well, even beyond their proven range of applicability. We combine the differential transform with Padé technique, and call this method differential transform-Padé approximation.

3. Applications

3.1. The Volterra equation

To verify the validity and the potential of GDTM in solving differential-difference equations, we will consider the following Volterra equation [23,24]:

\[ u_n' = u_n(u_{n+1} - u_{n-1}) \]

with the initial condition

\[ u_n(0) = n \]

whose exact solution can be expressed as

\[ u_n(t) = \frac{n}{1 - 2t} \]

According to the inspiration of Table 1, the transformed version of Eq. (8) can be written in the following recurrence formula

\[ (k + 1)U(n,k + 1) = \sum_{s=0}^{k} U(n,k-s)U(n+1,s) + \sum_{s=0}^{k} U(n,k-s)U(n-1,s) \]

The related initial conditions should be also transformed as follows

\[ U(n, 0) = n \]

\[ U(n + 1, 0) = n + 1 \]

\[ U(n - 1, 0) = n - 1 \]

Now, we should determine all of the residual kth derivations of function \( u_n(t) \), i.e. the coefficients of series solution Eq. (4) respect to \( t^k \). Eq. (13) has provided the initial value of the sequence of \( U(n,k) \) for the recurrence formula equation (11). So we can determine all of the kth derivatives of function \( U(n,k) \) one by one according to Eq. (11). Substituting these obtained \( U(n,k) \) into (4), we obtained the closed form solution as

\[ u_n(t) = \sum_{k=0}^{\infty} U_n(k)t^k = \sum_{k=0}^{\infty} U(n,k)t^k = n + 2nt + 4nt^2 + 8nt^3 + 16nt^4 + 32nt^5 + 64nt^6 + 128nt^7 + 256nt^8 + \cdots \]

As this is an infinite series, we truncate it with finite terms for considering of practice. The 6th-order approximation is given by

\[ u_n(t) = \sum_{k=0}^{5} U_n(k)t^k = \sum_{k=0}^{5} U(n,k)t^k = n + 2nt + 4nt^2 + 8nt^3 + 16nt^4 + 32nt^5 + 64nt^6. \]
If we apply differential transform-Padé technique to our 6th-order approximation (15), we get \([2, 3]\) generalized differential-difference Padé approximation:

\[
U_n(t) = \frac{n}{1 - 2t}.
\]  

The result can be easily obtained, especially by means of symbolic software such as Mathematica, Maple, MatLab, and so on. Obviously, this result coincides with exact solution Eq. (10) to Volterra equation subject to initial condition \(u_n(0) = n\).

Fig. 1 presents the comparison of exact solution and the 6th approximation solution obtained by generalized differential transform method. Good agreement is provided near the original point \(t = 0\), but difference between exact solution and approximation solution is larger along with the longer distance to original point \(t = 0\). Fig. 2 shows the comparison of exact solutions and \([2, 3]\) Padé approximation based on the 6th approximation solution. This figure shows that Padé technique has provided a global agreement with exact solution and numerical solution.

3.2. The Lotka–Volterra equation

Another example called the Lotka–Volterra equation is also considered \([23, 24]\):

\[
\dot{u}_n = u_n(u_{n-1} - u_{n+1}) + u_n(u_{n-2} - u_{n+2})
\]

with the initial condition

\[
u_n(0) = n,
\]

whose exact solution can be expressed as

\[
u_n(t) = \frac{n}{1 + 6t}.
\]
The transformed version of Eq. (17) is in the following form
\[
(k + 1)U(n, k + 1) = \sum_{s=0}^{k} U(n, k - s)U(n - 1, s) - \sum_{s=0}^{k} U(n, k - s)U(n + 1, s) + \sum_{s=0}^{k} U(n, k - s)U(n - 2, s) \\
- \sum_{s=0}^{k} U(n, k - s)U(n + 2, s).
\] 
(20)

The transformed initial condition is
\[
U(n, 0) = n.
\] 
(21)

Similarly, the following expression values are deduced
\[
U(n + 1, 0) = n + 1, \quad U(n - 1, 0) = n - 1,
\]
\[
U(n + 2, 0) = n + 2, \quad U(n - 2, 0) = n - 2.
\] 
(22)

Now we obtained the closed form solution as
\[
u_n(t) = \sum_{k=0}^{\infty} U_n(k) t^k = \sum_{k=0}^{\infty} U(n, k) t^k = n - 6nt + 36nt^2 - 216nt^3 + 1296nt^4 - 7776nt^5 \\
+ 46656nt^6 - 279936nt^7 + 1679616nt^8 + \cdots.
\]
(23)

The 6th-order approximation is given by
\[
u_n(t) = \sum_{k=0}^{6} U_n(k) t^k = \sum_{k=0}^{6} U(n, k) t^k = n - 6nt + 36nt^2 - 216nt^3 + 1296nt^4 - 7776nt^5 + 46656nt^6.
\]
(24)

The we apply differential transform-Padé technique to our 6th-order approximation (24), we get [2, 3] D-P approximation:
\[
U_n(t) = \frac{n}{1 + 6t}.
\] 
(25)

Obviously, this is an exact solution of Volterra equation with the initial condition \(u_n(0) = n\).

Differential transform-Padé approximation is an effective method to accelerate the convergence of the result and enlarge the convergence field. Thus differential transform-Padé approximation achieves a high convergence rate over a considerably large convergence region.

Fig. 3 gives the comparison of approximation solution by using generalized differential transform method and exact solution. It can be found again that approximation solution agrees well with exact solution near the area of original \(t = 0\). Fig. 4 presents the comparison of exact solutions and the Padé approximation solution based on the asymptotic series given by generalized differential transform method.

3.3. The discrete KdV equation

To verify the validity and the potential of generalized differential transform method (GDTM), we apply it to the discrete KdV equation, which describes the motions of waves in nonlinear optics, plasma or fluids. Analytic solitary solutions for this equation are obtained in use of the proposed method. The validity and effectiveness of the GDTM in solving the nonlinear solitary wave problems are shown.
The discrete KdV equation
\[ \frac{\partial u_n}{\partial t} = u_n^2(u_{n+1} - u_{n-1}) \] (26)
with the initial condition
\[ u_n(0) = 1 + \frac{\cosh d - 1}{1 + \cosh(dn - 2)}. \] (27)

The exact solution is found to be
\[ u_n(t) = 1 + \frac{\cosh d - 1}{1 + \cosh(dn + 2 \sinh(d)t - 2)}. \] (28)

The transformed version of Eq. (26) can be written in the following recurrence formula
\[ (k + 1)U(n, k + 1) = \sum_{s=0}^{k} \sum_{t=0}^{s} U(n, t)U(n, s-t)U(n + 1, k - s) \]
\[ - \sum_{s=0}^{k} \sum_{t=0}^{s} U(n, t)U(n, s-t)U(n - 1, k - s). \] (29)

The related initial conditions should be also transformed as follows
\[ U(n, 0) = 1 + \frac{\cosh d - 1}{1 + \cosh(dn - 2)}. \] (30)

Easily, the following implicit initial conditions are given through the final row relation between original functions and translated functions in Table 1
\[ U(n + 1, 0) = 1 + \frac{\cosh d - 1}{1 + \cosh(d(n + 1) - 2)}. \]
\[ U(n - 1, 0) = 1 + \frac{\cosh d - 1}{1 + \cosh(d(n - 1) - 2)}. \] (31)

Now, we should determine all of the residual kth derivations of function \( u_n(t) \), i.e. the coefficients of series solution Eq. (4) respect to \( t^k \). Eqs. (30) and (31) have provided the initial value of the sequence of \( U(n, k) \) for the recurrence formula equation (29). So we can determine all of the kth derivatives of function \( U(n, k) \) one by one according to Eq. (29). Substituting these obtained \( U(n, k) \) into (4), we obtained the closed form solution as
\[ u_n(t) = \sum_{k=0}^{\infty} U_n(k)t^k = \sum_{k=0}^{\infty} U(n, k)t^k. \] (32)

As this is an infinite series, we truncate it with finite terms for considering of practice. The 10th-order approximation is given by
\[ u_n(t) = \sum_{k=0}^{10} U_n(k)t^k = \sum_{k=0}^{10} U(n, k)t^k. \] (33)
For \( d = 0.3, n = 7 \), we get the 10th-order approximation

\[
\phi_{7,10} = 1.022612678 - 0.0006880286717t - 0.002081231824t^2 + 0.00008475177332t^3 \\
+ 0.0001272154494t^4 - 0.000006646511445t^5 - 0.000006582455750t^6 \\
+ 0.0000004254158208t^7 + 0.000000311803868t^8 - 0.0000002423640975t^9 \\
- 0.00000001397926130t^{10}.
\]  

(34)

If we apply differential transform-Padé technique to our 10th-order approximation (34), we get [4, 4] D-P approximation:

\[
P_{\phi_{7,10}} = \frac{1.022612678 + 0.02497188206t + 0.07628827024t^2 + 0.009735037972t^3 + 0.001484768910t^4}{0.9999999999 + 0.02509250206t + 0.07665342715t^2 + 0.009717414175t^3 + 0.001482114389t^4}.
\]  

(35)

The result can be easily obtained, especially by means of symbolic software such as Mathematica, Maple, MatLab, and so on. For \( d = 0.5, n = 7 \), we can obtain

\[
\phi_{7,10} = 1.038069920 - 0.02520024157t + 0.002173386415t^2 + 0.003602811035t^3 \\
- 0.001506515113t^4 - 0.00001468953519t^5 + 0.0001935930937t^6 \\
- 0.00005373672612t^7 - 0.0000623560459t^8 + 0.00007991701771t^9 \\
- 0.00001584872772t^{10}.
\]  

(36)

Employing differential transform-Padé approximation, we get a good approximation to exact solution

\[
P_{\phi_{7,10}} = \frac{1.038069920 + 0.4753482144t + 0.2370913251t^2 + 0.04302566443t^3 + 0.008291785543t^4}{0.9999999999 + 0.4821914655t + 0.2380083223t^2 + 0.0427454176t^3 + 0.008304803512t^4}.
\]  

(37)

Comparisons are made between the exact solution, the 10th-order approximation and the differential transform-Padé approximation obtained by generalized differential transform method, when \( d = 0.3, n = 7 \), as is shown in Fig. 5. Fig. 6 shows the exact solution, the 10th-order approximation and the differential transform Padé approximation, when \( d = 0.5, n = 7 \). The so-called differential transform-Padé technique is employed, which greatly accelerates the convergence.

3.4. The discrete mKdV equation

We consider the discrete mKdV equation

\[
\frac{\partial u_n}{\partial t} = (\alpha - u_n^2)(u_{n+1} - u_{n-1})
\]  

(38)

with the initial condition

\[
u_n(0) = \sqrt{-\alpha} \tanh \left( \frac{d}{2} \right) (1 + \cosh d) \text{sech}(dn - 2),
\]  

(39)

whose exact solution can be written as

\[
u_n(t) = \sqrt{-\alpha} \tanh \left( \frac{d}{2} \right) (1 + \cosh d) \text{sech}(dn - 2\alpha \tanh \left( \frac{d}{2} \right) (1 + \cosh d)t - 2).
\]  

(40)
According to the inspiration of Table 1, the transformed version of Eq. (38) is

\[(k + 1)U(n, k + 1) = \alpha(U(n - 1, k) - U(n + 1, k)) = \sum_{s=0}^{k} \sum_{t=0}^{s} U(n, t)U(n, s - t)[U(n + 1, k - s) - U(n - 1, k - s)].\]  
\[(41)\]

The transformed initial condition is

\[U(n, 0) = \sqrt{-\alpha \tanh \left(\frac{d}{2}\right)}(1 + \cosh d) \, \text{sech}(dn - 2).\]  
\[(42)\]

Easily, the following implicit initial conditions are given through the final row relation between original functions and translated functions in Table 1

\[U(n + 1, 0) = \sqrt{-\alpha \tanh \left(\frac{d}{2}\right)}(1 + \cosh d) \, \text{sech}(dn + 1 - 2).\]  
\[U(n - 1, 0) = \sqrt{-\alpha \tanh \left(\frac{d}{2}\right)}(1 + \cosh d) \, \text{sech}(dn - 1 - 2).\]  
\[(43)\]

Similarly, we obtained the closed form solution as

\[u_n(t) = \sum_{k=0}^{\infty} U_n(k) t^k = \sum_{k=0}^{\infty} U(n, k) t^k.\]  
\[(44)\]

As this is an infinite series, we truncate it with finite terms for considering of practice. The 6th-order approximation is given by

\[u_n(t) = \sum_{k=0}^{6} U_n(k) t^k = \sum_{k=0}^{6} U(n, k) t^k.\]  
\[(45)\]

When \(d = 0.1\) and \(n = 1\)

\[u_1(t) = 0.05861592646 + 0.04491536735t + 0.015597397000t^2 + 0.002337821840t^3 - 0.0005400926288t^4 - 0.00050268932465t^5 - 0.0001913367047t^6.\]  
\[(46)\]

Then we apply differential transform-Padé technique to our 6th-order approximation (46), we get [2, 4] D-P approximation. For \(d = 0.1, n = 1, \alpha = -4\), the differential transform-Padé approximation is obtained as

\[P_{\phi_{1.6}} = \frac{0.05861592646 + 0.01115354009t + 0.0004546823181t^2}{1.0 - 0.5759838551t + 0.1830187187t^2 - 0.0268532957t^3 + 0.004066750228t^4}.\]  
\[(47)\]

For \(d = 0.1, n = 35, \alpha = -4\), we can obtain the differential transform-Padé approximation

\[P_{\phi_{35.6}} = \frac{0.08516097651 - 0.01195764671t - 0.000876749797t^2}{1.0 + 0.5849137922t + 0.2181928040t^2 + 0.03179769643t^3 + 0.005950541105t^4}.\]  
\[(48)\]

For \(d = 0.3, n = 7, \alpha = -4\), we can obtain

\[P_{\phi_{7.6}} = \frac{0.6060080207 - 0.009888130925t - 0.1194061105t^2}{1.0 + 0.2264905849t + 2.7664444692t^2 + 0.1439110599t^3 + 0.8790046023t^4}.\]  
\[(49)\]

Differential transform-Padé approximation is an effective method to accelerate the convergence of the result and enlarge the convergence field. Thus differential transform-Padé approximation achieves a high convergence rate over a considerably large convergence region.
Fig. 7. Comparison of the exact solution with 10th-order approximation and \([2, 4]\) D-Padé approximation, when \(d = 0.1, n = 1, \alpha = -4\).

Fig. 8. Comparison of the exact solution with 10th-order approximation and \([2, 4]\) D-Padé approximation, when \(d = 0.1, n = 35, \alpha = -4\).

Fig. 9. Comparison of the exact solution with 10th-order approximation and \([2, 4]\) D-Padé approximation, when \(d = 0.3, n = 7, \alpha = -4\).

Figs. 7–9 give the comparison of approximation solution by using generalized differential transform method, \([2, 4]\) differential transform-Padé approximation and exact solution, respectively, when \(d = 0.1, n = 1, \alpha = -4, d = 0.1, n = 35, \alpha = -4, d = 0.3, n = 7, \alpha = -4\). When we gain the 6th-order approximation and employ differential transform-Padé approximation, the result agrees well with the exact solution. This shows that DTM-Padé method has an accuracy convergence in large region.

4. Discussions and conclusions

In this Letter, we successfully overcome the discrete variable \(n\) in nonlinear differential-difference equation and extend the differential transform method to solve the nonlinear differential-difference equation. This generalized differential transform method has great potential application to a wide variety of nonlinear problems in science and engineering. The present study has confirmed that the differential transform method offers great advantages of straightforward applicability, computational efficiency and high accuracy.
Acknowledgements

This work is partially supported by Natural Sciences Foundation of China under the grant 50909017 and Natural Sciences Foundation of China under the grant 50921001, the State Key Laboratory of Structure Analysis for Industrial Equipment under the grant S08018, Liaoning province Sciences & Technology under the grant 20062175 and The Key programme of DLUT for mathematica and X under the grant 842320.

References