World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:1, No:9, 2007

The Differential Transform Method for Advection-Diffusion Problems

M. F. Patrício, P. M. Rosa

***Abstract*—In this paper a class of numerical methods to solve linear and nonlinear PDEs and also systems of PDEs is developed. The Differential Transform method associated with the Method of Lines (MoL) is used. The theory for linear problems is extended to the nonlinear case, and a recurrence relation is established. This method can achieve an arbitrary high-order accuracy in time. A variable step-size algorithm and some numerical results are also presented.

***Keywords*—Method of Lines, Differential Transform Method.

I. Introduction

Problems involving diffusion-advection equations arise in many domains of Science. There are several methods for solving these equations.

The differential transform (DT) method has been used in different situations such as eigenvalue problems ([1]) and initial value problems ([3]). The technique concept was firstly introduced by J. Zhou ([6]) for electrical circuits. More recently, A. Kurnaz *et al.* used TD methods for approximating the solution of a system of ODEs, with good results for smooth profile solutions.

Using the method of lines (MoL), the DT method can be extended for solving systems of PDEs. For linear problems, this method is quite efficient, but for non linear equations with sharp gradient solutions, the results are not that good.

In section 2 the MoL approach for solving a PDE is presented, and the eigenvalues which depend on the parameters that appear in the PDE are studied. This is important for the stability analysis. Section 3 will be dedicated to the TD method and its application to both linear and nonlinear problems: a recurrence relation that allows to approximate the solution is established, and a variable step-size algorithm is presented. To end, in section 4 some numerical results that show the behavior of the method are obtained. One of these problems concerns a predator-prey model often used in Ecology.

II. The MoL Approach

Consider the one-dimensional diffusion-advection equation

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} - \beta \frac{\partial u}{\partial x} + s(x), \tag{1}$$

with $\alpha, \beta > 0,\ 0 < x < X,\ t > 0$, and initial and boundary conditions given by

$$u(x,0) = i(x), \quad 0 \le x \le X, \tag{2}$$

and

$$u(0,t) = v(t), \quad u(X,t) = g(t), \quad t > 0, \tag{3}$$

respectively. Assume a rectangular grid with step-sizes $\Delta x = \frac{X}{m}$ (in space) and $\Delta t = \frac{T}{n}$ (in time).

Replacing the spatial derivatives in (1) by the central-difference approximations

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{1}{\Delta x^2}[u(x-\Delta x,t) - 2u(x,t) + u(x+\Delta x,t)]$$

and

$$\frac{\partial u}{\partial x} \approx \frac{1}{2\Delta x}[u(x+\Delta x,t) - u(x-\Delta x,t)]$$

and applying (2) and (3) for $t = 0$ and at the boundaries, the first-order ordinary linear system is obtained.

$$\frac{dU}{dt}(t) = AU(t) + b(t), \quad t > 0, \tag{4}$$

where $U(t) = [u_1(t) \quad \dots \quad u_{m-1}(t)]^t$. Since $u_0(t) = v(t)$ and $u_m(t) = g(t)$, it is easy to establish that A is the tridiagonal matrix [1]

$$A = \frac{1}{\Delta x^2}\text{Tridiag}\left(\alpha + \frac{\beta\Delta x}{2}, -2\alpha, \alpha - \frac{\beta \Delta x}{2}\right).$$

and

$$b(t) = \begin{bmatrix} \left(\alpha + \frac{\beta\Delta x}{2}\right)\frac{v(t)}{\Delta x^2} + s(x_1) \\ s(x_2) \\ \vdots \\ s(x_{m-2}) \\ \left(\alpha - \frac{\beta\Delta x}{2}\right)\frac{g(t)}{\Delta x^2} + s(x_{m-1}) \end{bmatrix}.$$

In order to solve (4), it is useful to know the eigenvalues of matrix A. As it is known, these are given by

$$\lambda_j = \frac{2}{\Delta x^2}\left(-\alpha + \sqrt{\psi}\cos\frac{j\pi}{m}\right), \quad j = 1, 2, \dots, m-1, \tag{5}$$

where $\psi = \alpha^2 - \frac{\beta^2 \Delta x^2}{4}$.

From (5) it follows that

i. if $0 < \beta\Delta x \le 2\alpha$, then the eigenvalues are real numbers satisfying

$$-\frac{2}{\Delta x^2}(\alpha + \sqrt{\psi}) \le \lambda_j \le \frac{2}{\Delta x^2}(-\alpha + \sqrt{\psi}).$$

M. F. Patrício: Departamento de Matemática, Faculdade de Ciências e Tecnologia, Universidade de Coimbra, Apartado 3008, 3001-454 Coimbra, Portugal, email: mfsp@mat.uc.pt

P. M. Rosa: Departamento de Física e Matemática, Instituto Superior de Engenharia de Coimbra, Rua Pedro Nunes - Quinta da Nora, 3030-199 Coimbra, Portugal, email: pammr@isec.pt

[1] Tridiag $(a, b, c) = [T_{ij}]$ represents a tridiagonal matrix such that $T_{ii} = b$, $T_{i+1,i} = a$ and $T_{i,i+1} = c$.

ii. if $0 < 2\alpha < \beta\Delta x$, then the eigenvalues are complex numbers, $\lambda_j = -\frac{2\alpha}{\Delta x^2} + i\gamma_j$, where

$$-\frac{2}{\Delta x^2}\sqrt{-\psi} \leq \gamma_j \leq \frac{2}{\Delta x^2}\sqrt{-\psi}.$$

In the next section the linear and nonlinear cases are studied separately.

III. The Differential Transform Method

Following the approach presented in [4], if we define the k-th order DT of a function $U_i(t)$, at $t = t_j$, by

$$Y_i(k) = \frac{1}{k!}U_i^{(k)}(t_j),$$

then the inverse DT of $Y_i(k)$ is

$$U_i(t) = \sum_{k=0}^{+\infty} Y_i(k)(t-t_j)^k = \sum_{k=0}^{+\infty} \frac{1}{k!}U_i^{(k)}(t_j)(t-t_j)^k. \quad (6)$$

A. Linear Problem

In order to illustrate the DT method, consider again the linear system (4) of $m-1$ unknowns, for $t \in [0, t_n]$, with initial data $u^0(x) = \begin{bmatrix} u_1^0 & \dots & u_{m-1}^0 \end{bmatrix}^t$. Let $Y(k) = [Y_1(k) \quad \dots \quad Y_{m-1}(k)]^t$. Applying DT to (4) the recurrence relation

$$(k+1)Y(k+1) = AY(k) + \frac{1}{k!}U_i^{(k)}(t_j)$$

can be obtained, and therefore it is easy to establish that

$$Y(k+1) = \frac{1}{(k+1)!}\left[A^{k+1}Y(0) + \sum_{r=0}^{k} A^r b^{(k-r)}(t_j)\right], \quad (7)$$

with $Y(0) = U(0) = \begin{bmatrix} u_1^0 & \dots & u_{m-1}^0 \end{bmatrix}^t$.

Take the first $p+1$ terms of the inverse DT, an approximation to $U_i(t)$, $1 \leq i \leq m-1$, can be obtained.

An algorithm for solving problem (4) is, therefore, straightforward:

1) Consider a time discretization $t_j = t_{j-1} + \Delta t$, creating subintervals $I_j = [t_{j-1}, t_j]$, $j = 1, \dots, n$.
2) For $j = 0, 1, \dots, n-1$,
   a) Compute $Y(k)$ $(k = 0, 1, \dots p)$ as in (7), using $t = t_j$;
   b) Consider

$$U(t_{j+1}) \approx \sum_{k=0}^{p} Y(k)\Delta t^k.$$

Important remarks:

1) It is not necessary to impose a fixed time-step. In fact, we can change step one of the algorithm, for example, as follows: In the interval $I_{j+1} = [t_j, t_{j+1}]$ start by approximating $U(t_{j+1})$ using time-step Δt, and denote such approximation by $\tilde{U}$. Next, compute the approximation for the same value using time-step $\frac{\Delta t}{2}$, and denote it by $\bar{U}$.
   - **If** $||\tilde{U} - \bar{U}|| < \epsilon$ (ϵ =tolerance), **then** accept Δt and proceed with this time-step in the next interval ($[t_{j+1}, t_{j+2}]$);
   - **else** compare the approximation for $U\left(t_{j+\frac{1}{2}}\right)$ obtained with step-sizes $\frac{\Delta t}{2}$ and $\frac{\Delta t}{4}$. This process should be repeated until the approximation obtained with time-step $\frac{\Delta t}{2^k}$ is acceptable for some $k \geq 1$.

   It should be noticed that if, after some iterations, the same value of Δt is considered to be acceptable, then it may be reasonable to try to increase the time-step. However, it must always be tested if the solution obtained with a larger time-step is acceptable, and therefore constant monitoring (just like it has been done above) is advised.
2) As far as stability is concerned, it is easy to show that this method is stable provided $|\lambda\Delta t| < 1$, where λ are the eigenvalues of matrix A (see equation (4)).
3) Note that different values of p can be used in different time-intervals I_j, depending on the accuracy we want to achieve. Obviously, higher-order methods are more expensive.

B. Nonlinear Problem

Consider now the well-known Burger's Equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \lambda\frac{\partial^2 u}{\partial x^2}, \quad \lambda > 0, \quad 0 < x < 1, \quad t > 0, \quad (8)$$

with initial and boundary conditions given by

$$u(x,0) = \left[1 + \exp\left(\frac{x}{2\lambda}\right)\right]^{-1}$$

and

$$v(t) := u(0,t) = \left[1 + \exp\left(-\frac{t}{4\lambda}\right)\right]^{-1},$$

$$g(t) := u(1,t) = \left[1 + \exp\left(\frac{1}{2\lambda} - \frac{t}{4\lambda}\right)\right]^{-1},$$

respectively.

It is easy to show that the exact solution of this problem, for $\lambda \neq 0$, is given by

$$u(x,t) = \left[1 + \exp\left(\frac{x}{2\lambda} - \frac{t}{4\lambda}\right)\right]^{-1}.$$

Discretizing equation (8) using centered differences for the spatial derivatives, the (non-linear) differential system of $m-1$ unknowns is obtained:

$$\frac{dU}{dt}(t) = AU(t) + b(t) + F(t,U), \quad (9)$$

where $A = \frac{\lambda}{\Delta x^2}$Tridiag $(1, -2, 1)$,

$$b_i(t) = \begin{cases} \frac{\lambda}{\Delta x^2}v(t) & \text{, if } i = 1 \\ 0 & \text{, if } 2 \leq i \leq m-2 \\ \frac{\lambda}{\Delta x^2}g(t) & \text{, if } i = m-1 \end{cases},$$

and $F(t,U)$ contains the non-linear part of the system:

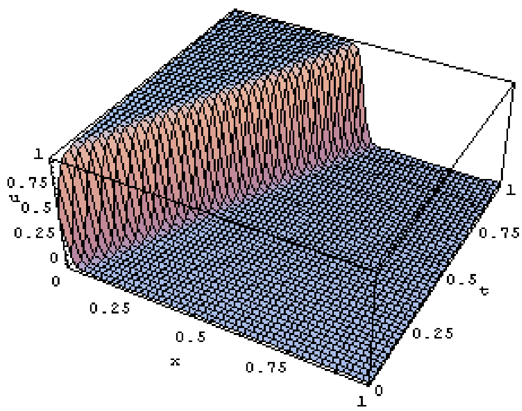


Fig. 1. Exact solution of problem (8).

$$F_i(t,U) = \begin{cases} \frac{1}{2\Delta x} U_1(t)(v(t) - U_2(t)) & \text{, if } i = 1 \\ \frac{1}{2\Delta x} U_i(t)(U_{i-1}(t) - U_{i+1}(t)) & \text{, if } 2 \le i \le m-2 \\ \frac{1}{2\Delta x} U_{m-1}(t)(U_{m-2}(t) - g(t)) & \text{, if } i = m-1 \end{cases}.$$

Applying the differential transform method presented before, we get the recurrence relation

$$(k+1)Y(k+1) = AY(k) + \frac{1}{k!} b^{(k)}(t_j) + G(k), \quad (10)$$

where $Y(0) = U(0) = [u(x_1, 0) \quad \ldots \quad u(x_{m-1}, 0)]^t$ and $G(k)$ is the differential transform of $F(t, U)$:

$$G_i(k) = \begin{cases} \frac{1}{2\Delta x}\left[Y_1(k) \otimes \left(\frac{1}{k!} v^{(k)}(t_j) - Y_2(k)\right)\right] & \text{, if } i = 1 \\ \frac{1}{2\Delta x}\left[Y_i(k) \otimes \left(Y_{i-1}(k) - Y_{i+1}(k)\right)\right] & \text{, if } 2 \le i \le m-2 \\ \frac{1}{2\Delta x}\left[Y_{m-1}(k) \otimes \left(Y_{m-2}(k) - \frac{1}{k!} g^{(k)}(t_j)\right)\right] & \text{, if } i = m-1 \end{cases}.$$

In the previous expression, operator $\otimes$ is defined by

$$R(k) \otimes S(k) := \sum_{r=0}^{k} R(r) S(k-r).$$

An algorithm for approximating $U(t_{j+1})$ is now straightforward, just like what has been done in the linear case.

Note that the procedure described for Burger's Equation can be extended to other nonlinear problems.

IV. Numerical Results

In this section some numerical results that illustrate the behavior of the method presented in section 3 are obtained. A linear and nonlinear single PDE, as well as a system of PDEs frequently used in Ecology, are studied.

A. Problem 1: Diffusion Equation

Consider the diffusion equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad (11)$$

with initial conditions

$$u(x, 0) = \sin \pi x, \quad 0 \le x \le 1,$$

and boundary conditions

$$u(0, t) = u(1, t) = 0, \quad t > 0.$$

It can be easily verified that the exact solution for this problem is $u(x, t) = e^{-\pi^2 t} \sin(\pi x)$ (see figure 2).

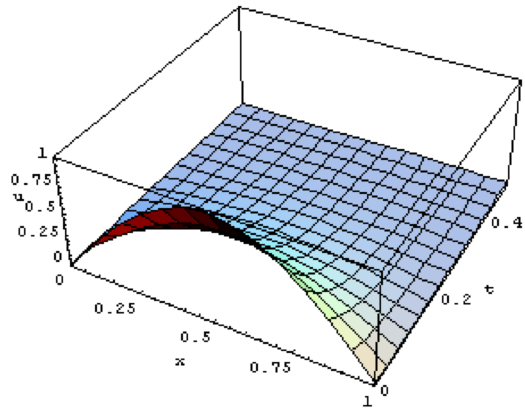


Fig. 2. Exact solutions for the diffusion equation.

Table I contains the EMAX [2] errors for different values of t obtained by using the TD method with $p = 1$ and $p = 3$.

TABLE I
EMAX ERRORS WITH $\Delta x = 10^{-1}$ AND $\Delta t = 0.5 \times 10^{-2}$.

t	$p = 1$	$p = 3$
0.1	0.61635×10^{-2}	0.302586×10^{-2}
0.3	0.25263×10^{-2}	0.127124×10^{-2}
0.5	0.575319×10^{-3}	0.296718×10^{-3}

B. Problem 2: Burger's Equation

As it was presented before, the TD method can be applied to non-linear PDEs. Figure 3 contains some plots of the results (exact and approximate solutions) obtained by this method with order $p = 1$, $\Delta x = 10^{-2}$ and $\Delta t = 10^{-4}$ for equation (8).

The corresponding error values are shown in table II.

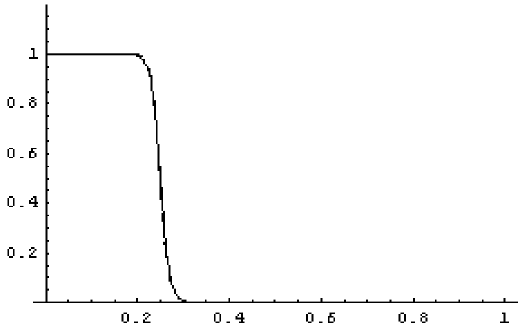
TABLE II
EMAX ERRORS FOR PROBLEM (8) WITH $\Delta x = 10^{-2}$ AND $\Delta t = 10^{-4}$.

t	EMAX error
0.2	0.522059×10^{-1}
0.4	0.547426×10^{-1}
0.6	0.548488×10^{-1}
0.8	0.548533×10^{-1}
1.0	0.548534×10^{-1}

C. Problem 3: A Predator-Prey Ecological Model

The following system of PDEs models the interaction of a certain species of predator (v) and preys (u). For simplicity, a one-dimensional spatial domain is considered.

[2] EMAX represents the maximum error: $\text{EMAX}(t) = \max_j |u(x_j, t) - U(x_j, t)|$.



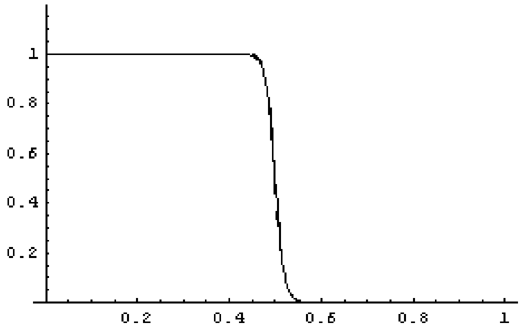


Fig. 3. Numerical (dashed line) and exact solution (solid line) of problem (8) with $\lambda = 0.005$ for $t = 0.5$ (top), and $t = 1.0$ (bottom).

$$\begin{cases} \frac{\partial u}{\partial t} = D_u \frac{\partial^2 u}{\partial x^2} + au - B(u,v) \\ \frac{\partial v}{\partial t} = D_v \frac{\partial^2 v}{\partial x^2} - cv + kB(u,v) \end{cases} . \qquad (12)$$

In this system, au is the growth rate of u, cv is the death rate of v, $k > 0$, and $B(u,v)$ is the interaction between u and v. The quantities D_u and D_v are the diffusion coefficients of u and v respectively. Typically, the Dirichlet boundary conditions (hostile exterior) are considered:

$$u(0,t) = u(1,t) = v(0,t) = v(1,t) = 0.$$

Letting $B(u,v) = d(u+v)$, it is possible to obtain a system of linear PDEs that can easily be solved using the algorithm presented in section 2. In fact, discretizing the spatial derivatives using central differences, and after some manipulation, the semi-discrete system (which is a particular case of system (4)) can be obtained.

$$\frac{dW}{dt} = AW(t).$$

In the previous equation, $W(t)$ is given by

$$W(t) = [u_1(t) \quad \ldots \quad u_{m-1}(t) \quad v_1(t) \quad v_{m-1}(t)]^t \in \mathbb{R}^{2m-2},$$

and A is the $(2m-2) \times (2m-2)$ block matrix

$$A = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array} \right],$$

where

$$A_{11} = \text{Tridiag}\left(\frac{D_u}{\Delta x^2}, a - d - 2\frac{D_u}{\Delta x^2}, \frac{D_u}{\Delta x^2} \right),$$

$$A_{12} = \text{Diag}(-d),$$

$$A_{21} = \text{Diag}(kd)$$

and

$$A_{22} = \text{Tridiag}\left(\frac{D_v}{\Delta x^2}, kd - c - 2\frac{D_v}{\Delta x^2}, \frac{D_v}{\Delta x^2} \right).$$

In the previous expressions, $\text{Diag}(\gamma)$ represents a diagonal matrix with diagonal elements equal to γ.

The presented algorithm can therefore be applied in a straightforward way. We used the initial conditions $u(x,0) = \sin(\pi x)$ and $v(x,0) = \frac{1}{2}\sin(\pi x)$. As for the values of the parameters present in (12), the choices were $D_u = D_v = a = c = 0.1$ and $d = k = 1$. Figure 4 contains the results of the TD method when applied to system (12), with order $p = 3$. For this choice of the parameters, the predator species increases which makes the number of preys to decrease ($t = 0.6$). However, when the preys vanish (see figure 4 at $t = 1.5$), this makes the number of predators to decrease (eventually to extinction).

Final Remarks:

- The TD method is particularly simple to program, has a rather low computational cost, and an arbitrarily high order in time.
- The authors have a fully automated *Mathematica* © code for solving all the problems presented in this paper.

References

[1] Chen, C. K. and Ho, S. H., *Applications of the differential transform to eigenvalue problems*, Applied Mathematics and Computation, 79, 173–188, (1996).

[2] Enright, W. H., Second derivative multistep methods for stiff ordinary differential equations, SIAM J. Num. Anal., 11, 321-331, (1974).

[3] Jang, M., Chen, C. and Liu, L., On solving the initial-value problem using the differential transform method, Applied Mathematics and Computation, 115, 145–160, (2000).

[4] Kurnaz, A. and Oturanç, G., *The differential transform approximation for the system of ordinary differential equations*, International Journal of Computer Mathematics, 88, 709–719, (2005).

[5] Oliveira, P. de, Patrício, M. F. and Santos, J., *Positive Solutions with multistep methods for convection-diffusion-reaction equations*, Report number 97–22 (1997), Departament of Matemathics, University Coimbra.

[6] Zhou, J., *Differential transformation and its applications for electrical circuits*, Wuhan, China: Huarjung University Press, (1986).

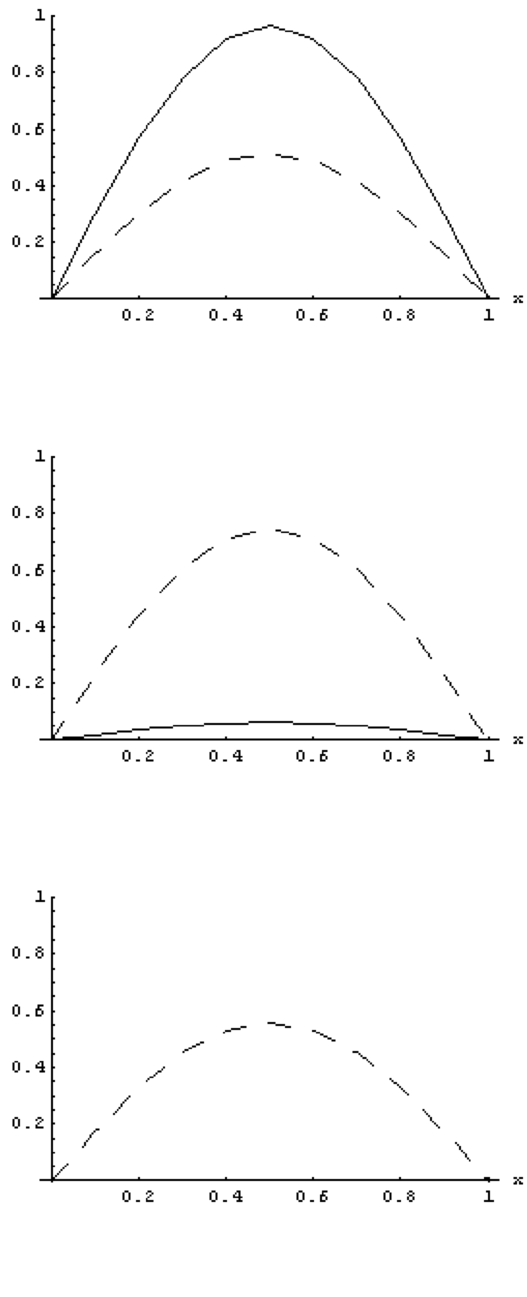


Fig. 4. Preys (solid line) and predators (dashed line) at $t = 0$ (top), $t = 0.6$ (center) and $t = 1.5$ (bottom).