Commun Nonlinear Sci Numer Simulat 30 (2016) 139–150

Contents lists available at ScienceDirect

Commun Nonlinear Sci Numer Simulat

journal homepage: www.elsevier.com/locate/cnsns

A windowing waveform relaxation method for time-fractional differential equations☆

Xiao-Li Ding [a,*], Yao-Lin Jiang [b]

[a] Department of Mathematics, Xi'an Polytechnic University, Xi'an, Shaanxi 710048, China
[b] Department of Mathematical Sciences, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China

ARTICLE INFO

Article history:
Received 3 October 2014
Revised 12 May 2015
Accepted 17 June 2015
Available online 26 June 2015

Keywords:
Time-fractional differential equations
Waveform relaxation
Windowing technique

ABSTRACT

This paper presents a new windowing waveform relaxation method for time-fractional differential equations. Unlike the classical case, the proposed windowing method uses the history part of the solution at each window. Second, it is the first time that a multi-domain finite difference scheme together with a windowing method has been used for the time-fractional differential equations, which makes the numerical scheme very efficient. Third, the paper provides an effective estimation on window length. Numerical results are given to further illustrate the theoretical analysis.

1. Introduction

Fractional calculus has been used widely to deal with some problems in fluid and continuum mechanics [1,2], viscoelastic and viscoplastic flow [3] and anomalous diffusion [4–6]. The main advantage of fractional derivatives lies in that they are more suitable for describing memory and hereditary properties of various materials and process in comparison with classical integer-order derivative. In these years, numerical solutions to fractional differential equations have been investigated (for example, see [3,7–11]).

Waveform relaxation (WR) method is an iterative method for solving ordinary differential equations. It was originally proposed to simulate large circuits in [12]. The method has two main advantages. On the one hand, it can decouple a given large differential system into a set of weakly coupled small subsystems. On the other hand, it can also decouple a complicated differential system into a set of simplified subsystems. Some authors have successfully applied the WR method into solving ordinary differential equations [12–15], differential-algebraic equations [16], functional differential equations [17] and fractional differential equations [18–20].

Although the WR method of fractional differential equations is eventually convergent, the convergence rate is very slow during the iterations on finite long intervals. One of the reasons of slow convergence is that the solution of fractional differential equation at a certain time depends on the whole history of the solution at previous times. Considering the design of numerical schemes, an immediate consequence is that the entire past trajectory of the numerical solution must be carried forward and used in the current time step. This impacts both the storage and the cost of the numerical method, both of which may substantially increase over time. Therefore, it is natural to accelerate the WR method in terms of both storage and computation.

☆ This work was supported by the Science and Technology Planning Project (2014JQ1041) of Shaanxi Province and the Scientific Research Program Funded by Shaanxi Provincial Education Department (14JK1300). The work was partially supported also by the Natural Science Foundation of China (11371287).

* Corresponding author. Tel.: +86 15829680246.
E-mail address: dingding0605@126.com (X.-L. Ding).

http://dx.doi.org/10.1016/j.cnsns.2015.06.017
1007-5704/

In this paper, we will develop the windowing technique following the idea of the windowing WR method proposed by Leimkuhler and Ruehli in RC circuit simulation [15]. Basic idea of the windowing WR is that the interval of integration is split into a series of subintervals, called windows, with iteration taking place on each window successively. During the last years, there has been an explosion of research work on the analysis of the windowing WR method (for example, see [21,22]). However, there is currently no development of windowing WR method for fractional differential equations.

This paper is an effort to meet the high computational demand of the time fractional differential equations (TFDEs). The original windowing WR method for classic ODEs must be modified to take into account the memory effect of the TFDEs and we propose to include the history part of the solution in the iteration step. On the one hand, the proposed windowing WR method maintains the nice property of the original windowing WR method, i.e., the speed of convergence of the WR method can be accelerated using suitable window. On the other hand, it has a particular advantage for the time fractional differential equations in that the numerical solution on each subinterval is updated by using the history value after k iterations, thus the memory restriction is relaxed. In this paper, we provide a convergence analysis of the new method, and give an effective estimation on window length.

To implement the windowing method, one also needs to be cognizant of the challenge in the serial time stepping scheme for the TFDEs. In this paper, we adopt a multi-domain finite difference method. The whole time domain will be decomposed into subintervals, on which finite difference scheme is established.

The paper is organized as follows. In the next section, a brief background on the windowing WR method and the TFDEs will be provided. In Section 3, we describe the new windowing WR method for the TFDEs and derive an effective estimation on window length. Numerical results will be shown in Section 4.

2. Preliminaries

In this section, we provide background knowledge on the windowing WR method for classic ODEs and introduce some concepts about fractional calculus.

2.1. The windowing WR method

Consider the following scalar ODE:

$$\begin{cases} \dfrac{dx(t)}{dt} = f(x(t), t), t \in \Omega = [0, T], \\ x(0) = x_0, \end{cases} \tag{2.1}$$

where $f(x(t), t) : \mathbb{R}^n \times [0, T] \to \mathbb{R}^n$ is a given function, x_0 is a known initial value, and $x(t)$ is to be computed. In the following, we describe the basic settings of the original windowing WR method (cf. [21]), aiming to solve (2.1).

First, the time domain is decomposed into N elements:

$$0 = T_0 < T_1 < \cdots < T_N = T, \quad \Omega_i = [T_i, T_{i+1}],$$

where each subinterval $\Omega_i = [T_i, T_{i+1}]$ is called a window, and the lengths of windows are $H_i = T_{i+1} - T_i$ for $i = 0, 1, \ldots, N-1$.

Next, we introduce a splitting function $F(\cdot, \cdot, t) : \mathbb{R}^n \times \mathbb{R}^n \times [0, T] \to \mathbb{R}^n$ satisfies

$$F(x(t), x(t), t) = f(x(t), t),$$

for any $x(t) \in \mathbb{R}^n$. The splitting function is chosen to attempt to decouple the original system into easily solvable independent subsystems. As an example, one can use the classic Jacobi or Gauss–Seidel scheme for F. The computation of WR solutions starts at T_0 and goes on for one window to another window until $T_N = T$ is reached. That is,

$$\begin{cases} \dfrac{dx_i^{(k+1)}(t)}{dt} = F(x_i^{(k+1)}(t), x_i^{(k)}(t), t), \quad t \in [T_i, T_{i+1}], \\ x_i^{(k+1)}(T_i) = x_{i-1}^{(k_{i-1})}(T_i), \quad k = 0, 1, \ldots, k_i - 1, \\ i = 0, 1, \ldots, N-1, \end{cases} \tag{2.2}$$

where $x_{-1}^{(k_{-1})}(0) = x_0$ and the initial guesses $x_i^{(0)}(t) \equiv x_{i-1}^{(k_{i-1})}(T_i)$ on $[T_i, T_{i+1}]$ for all i. By the existing convergence analysis (cf. [21]), it has $\lim_{k_i \to +\infty} x_i^{(k_i)}(t) = x_i(t)$ for all i, where $x_i(t)$ denotes the exact solution of Eq. (2.1) on the subinterval Ω_i. And the optimal window length is of practice importance for the implementation of the windowing WR method.

2.2. Fractional differential equations

Let $t \in (a, b)$. The fractional integral of order α of a given function $x(t)$ is defined as

$$({}_aI_t^{\alpha} x)(t) = \frac{1}{\Gamma(\alpha)} \int_a^t (t-\tau)^{\alpha-1} x(\tau) d\tau, \quad a < t \le b.$$