Fast predictor-corrector methods for solving nonlinear time-fractional differential equations

Yongtao Zhou
A cooperative work with Cui Li and Martin Stynes

Beijing Computational Science Research Center
Henan University of Economics and Law

12th August 2022

Outline

Review

Consider the nonlinear fractional initial-value problem (FIVP)

$$ {}^{C}D_0^{\alpha} y(t) = g(t, y(t)) \quad \text{for } t \in (0, T], \quad \text{with } y(0) = y_0, \tag{1.1} $$

where ${}^{C}D_0^{\alpha}$ is the Caputo fractional derivative of order $\alpha \in (0, 1)$, the nonlinear function g is assumed to be continuous and satisfies a Lipschitz condition in its second variable on a suitable set D:

$$ |g(t, y) - g(t, \hat{y})| \leq \hat{L}|y - \hat{y}| \quad \text{for } t \in [0, T]. \tag{1.2} $$

- Direct method: directly approximate the fractional derivative operator in (1.1);
- Indirect method: transform (1.1) into the corresponding integral form, and then use the numerical methods to solve the integral equation.

Direct method

- L1 type method

  Keith Oldham & Jerome Spanier (Book, 1974);
  Zhizhong Sun, & Xiaonan Wu (Appl. Numer. Math., 2006);
  Yumin Lin & Chuanju Xu (J. Comput. Phys., 2007);
  Martin Stynes et al. (SIAM J. Numer. Anal., 2017);
  Honglin Liao et al. (SIAM J. Numer. Anal., 2018);
  ...

- L2 type method

  Guanghua Gao et al. (J. Comput. Phys., 2014);
  Anatoly A. Alikhanov (J. Comput. Phys., 2015);
  Anatoly A. Alikhanov, Chengming Huang (Appl. Math. Comput., 2021);
  ...

- Convolution quadrature

  Christian Lubich (SIAM J. Math. Anal., 1986);
  Bangti Jin et al. (SIAM J. Sci. Comput., 2017);
  ...

- Spectral method

  ...

- Discontinuous Galerkin method

  ...

Indirect method

FIVP (1.1) is equivalent to the weakly singular Volterra integral equation

$$y(t) = y_0 + \frac{1}{\Gamma(\alpha)} \int_{s=0}^{t} (t-s)^{\alpha-1} g(s, y(s))\, ds. \tag{1.3}$$

- Predictor-corrector method

  Kai Diethelm et al., (Numer. Algorithms, 2004);
  Weihua Deng (J. Comput. Appl. Math., 2007);
  Changpin Li et al., (J. Comput. Phys., 2016);
  Thien Binh Nguyen & Bongsoo Jang (Fract. Calc. Appl. Anal., 2017);
  ...

- Block-by-block method

  Junying Cao & Chuanju Xu (J. Comput. Phys., 2013);
  ...

- Corrected methods

  Wanrong Cao et al., (SIAM J. Sci. Comput., 2016);
  ...

Graded meshes

- A complication is that typical solutions of this class of problem lack regularity at the initial time. The maximum errors of most schemes are $O(N^{-\alpha})$ (see Zhou & Stynes (East Asian J. Appl. Math., 2022)):

  L1 method;
  $\overline{\text{L1}}$ method;
  L2-1_σ method;
  ...

- To address this difficulty in their numerical solution, it is by now well known that nonuniform meshes — specifically, graded meshes — are an efficient way of handling the initial weak singularity appearing in solutions;

  Martin Stynes et al. (SIAM J. Numer. Anal., 2017);
  Jinye Shen et al. (Submitted, 2022);
  Hu Chen & Martin Stynes (J. Sci. Comput., 2019);
  ...

- Divide the interval $[0, T]$ into N sub-intervals by the mesh points $t_n := T(n/N)^r$ for $0 \leq n \leq N$. Here $r \geq 1$ is a user-chosen *mesh-grading parameter*.

The solution y of (1.1)

- Existence of a solution of (1.1) is studied in [Kai Diethelm, Book, 2010, Theorem 6.1].
- Uniqueness of this solution is guaranteed by [Kai Diethelm, Book, 2010, Theorem 6.5].

Furthermore, from Kai Diethelm et al., (Numer. Algorithms, 2004, Theorem 2.1) one sees that y has the structure described in the following lemma.

Lemma 2.1

(Structure of solution y) Assume that the set D where the Lipschitz condition holds includes $\{(t, y(t)) : 0 \le t \le T\}$, where $y \in C[0, T]$ is the solution of (1.1).

(a) *Suppose that $g \in C^2(D)$. Then there exist a function $\psi \in C^1[0, T]$ and $c_1, c_2, \ldots, c_{\hat{v}} \in \mathbb{R}$ such that*

$$y(t) = \psi(t) + \sum_{v=1}^{\hat{v}} c_v t^{v\alpha} \quad \textit{where } \hat{v} := \lceil 1/\alpha \rceil - 1.$$

(b) *Suppose that $g \in C^3(D)$. Then there exist a function $\psi \in C^2[0, T]$ and $c_1, c_2, \ldots, c_{\hat{v}}, d_1, d_2, \ldots, d_{\tilde{v}} \in \mathbb{R}$ such that*

$$y(t) = \psi(t) + \sum_{v=1}^{\hat{v}} c_v t^{v\alpha} + \sum_{v=1}^{\tilde{v}} d_v t^{1+v\alpha} \quad \textit{with } \hat{v} := \lceil 2/\alpha \rceil - 1, \ \tilde{v} := \lceil 1/\alpha \rceil - 1.$$

The solution y of (1.1)

- y can be written as a sum of singular and regular parts. In particular $y(t) \sim y_0 + c_1 t^\alpha$ near $t=0$, so y has a weak singularity at $t=0$.
- From Lemma 2.1 it follows that

$$y \in C[0,T] \cap C^2(0,T) \text{ with } |y^{(k)}(t)| \le C(1+t^{\alpha-k}) \text{ for } k=0,1,2,\ t\in(0,T]. \tag{2.1}$$

- For $z := {}^C D_0^\alpha y$ one has

$$z \in C[0,T] \cap C^2(0,T) \text{ with } |z^{(k)}(t)| \le C(1+t^{\alpha-k}) \text{ for } k=0,1,2,\ t\in(0,T]. \tag{2.2}$$

Predictor-corrector method (from Kai Diethelm et al., (Numer. Algorithms, 2004))

At each mesh point $t = t_{n+1}$, one can rewrite (1.3) as

$$y(t_{n+1}) = y_0 + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{n-1} \int_{s=t_j}^{t_{j+1}} (t_{n+1} - s)^{\alpha-1} g(s, y(s))\, ds$$
$$+ \frac{1}{\Gamma(\alpha)} \int_{s=t_n}^{t_{n+1}} (t_{n+1} - s)^{\alpha-1} g(s, y(s))\, ds. \tag{2.3}$$

- The prediction stage: on each interval $I_j = [t_j, t_{j+1}]$, $0 \le j \le n$, replace $g(s, y(s))$ by $g_j = g(t_j, y_j)$;

$$y^P_{n+1} = y_0 + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{n} b_{j,n+1} g_j. \tag{2.4}$$

- The correction stage: on each interval $I_j = [t_j, t_{j+1}]$, $0 \le j \le n$, replace $g(s, y(s))$ by the linear interpolation of g_j and g_{j+1};

$$y_{n+1} = y_0 + \frac{1}{\Gamma(\alpha)} \left(\sum_{j=0}^{n} a_{j,n+1} g_j + a_{n+1,n+1} g(t_{n+1}, y^P_{n+1}) \right). \tag{2.5}$$

- Different approximations of the history part are used in the prediction and correction stages; Too expensive; Order of convergence $N^{-\min\{2\alpha, 1+\alpha\}}$ (see Yanzhi Liu et al. Numer. Algorithms, 2018, Theorem 1.5);

New predictor-corrector method

At each mesh point $t = t_{n+1}$, one can rewrite (1.3) as

$$y(t_{n+1}) = y_0 + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{n-1} \int_{s=t_j}^{t_{j+1}} (t_{n+1} - s)^{\alpha-1} g(s, y(s))\, ds + \frac{1}{\Gamma(\alpha)} \int_{s=t_n}^{t_{n+1}} (t_{n+1} - s)^{\alpha-1} g(s, y(s))\, ds. \quad (2.6)$$

- The prediction stage: for $n = 0$, replace $g_y := g(s, y(s))$ by g_0; for $n \geq 1$, on each interval I_j for $j = 0, 1, \ldots, n-1$, replace g_y by the linear interpolation of g_j and g_{j+1}, but on the final interval I_n, replace g_y by the linear function that interpolates to g_{n-1} and g_n;

$$y_{n+1}^P = y_0 + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{n-1} \left(a_{j,0}^{n+1} g_j + a_{j,1}^{n+1} g_{j+1} \right) + \frac{1}{\Gamma(\alpha)} \left(b_{n,0}^{n+1} g_{n-1} + b_{n,1}^{n+1} g_n \right). \quad (2.7)$$

- The correction stage: on each interval I_j for $j = 0, 1, \ldots, n$, replace g_y by the linear interpolation of g_j and g_{j+1};

$$y_{n+1} = y_0 + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{n-1} \left(a_{j,0}^{n+1} g_j + a_{j,1}^{n+1} g_{j+1} \right) + \frac{1}{\Gamma(\alpha)} \left[a_{n,0}^{n+1} g_n + a_{n,1}^{n+1} g(t_{n+1}, y_{n+1}^P) \right]. \quad (2.8)$$

- Introduced by Nguyen & Jang (Fract. Calc. Appl. Anal., 2017) on uniform meshes, but we implement it on our graded mesh.

Error analysis of the second-order predictor-corrector method

For this error analysis, we give first some preparatory results.

Lemma 2.2

[Alfio Quarteroni and Alberto Valli, Book, 1994, Lemma 1.4.2] (Classical discrete Gronwall inequality) Let $\{k_j\}_{j=0}^N$ and $\{q_j\}_{j=0}^N$ be nonnegative sequences. Let $d_0 \geq 0$. Assume that the sequence $\{\phi_n\}_{n=0}^N$ satisfies

$$\begin{cases} \phi_0 \leq d_0, \\ \phi_n \leq d_0 + \sum\limits_{j=0}^{n-1} q_j + \sum\limits_{j=0}^{n-1} k_j \phi_j \quad \text{for } 1 \leq n \leq N. \end{cases}$$

Then one has

$$\phi_n \leq \left(d_0 + \sum_{j=0}^{n-1} q_j \right) \exp \left(\sum_{j=0}^{n-1} k_j \right) \quad \text{for } 1 \leq n \leq N.$$

Error analysis of the second-order predictor-corrector method

Lemma 2.3

Let $w \in C[0,T] \cap C^2(0,T]$. *Assume that* $|w^{(k)}(t)| \le C(1+t^{\alpha-k})$ *for* $k=0,1,2$ *and all* $t \in (0,T]$. *For* $n = 0,1,\dots,N-1$, *set*

$$I_1^{n+1} = \left| \sum_{j=0}^{n} \int_{s=t_j}^{t_{j+1}} (t_{n+1}-s)^{\alpha-1} (w - \Pi_{1,j} w)(s)\, ds \right|,$$

$$I_2^{n+1} = \left| \sum_{j=0}^{n-1} \int_{s=t_j}^{t_{j+1}} (t_{n+1}-s)^{\alpha-1} (w - \Pi_{1,j} w)(s)\, ds + \int_{s=t_n}^{t_{n+1}} (t_{n+1}-s)^{\alpha-1} (w - \Pi_{1,n-1} w)(s)\, ds \right|.$$

Then

$$I_1^{n+1} + I_2^{n+1} \le C \begin{cases} N^{-2r\alpha} & \text{for } r\alpha < 1, \\ N^{-2} \ln N & \text{for } r\alpha = 1, \\ N^{-2} & \text{for } r\alpha > 1. \end{cases} \quad 0 \le n \le N-1.$$

Error analysis of the second-order predictor-corrector method

We can now prove an error bound for our predictor-corrector solution.

Theorem 2.4

Recall that $y(t)$ is the solution of (2.6) and $\{y_j\}_{j=0}^{N}$ is the solution of the second-order predictor-corrector method (2.7) and (2.8). One has

$$|y(t_j) - y_j| \le C \begin{cases} N^{-2r\alpha} & \text{for } r\alpha < 1, \\ N^{-2}\ln N & \text{for } r\alpha = 1, \\ N^{-2} & \text{for } r\alpha > 1. \end{cases} \quad 1 \le j \le N.$$

Remark 2.1

Our predictor-corrector method and its error analysis can easily be generalised to the FIVP (1.1) with $\alpha \in (1,2)$, with two initial conditions specifying $y(0)$ and $y'(0)$. In this setting the error for $1 \le j \le N$ satisfies the bound

$$|y(t_j) - y_j| \le C \begin{cases} N^{-2}\ln N & \text{for } r\alpha = 1, \\ N^{-2} & \text{otherwise}, \end{cases}$$

on our graded mesh.

Fast evaluation of the second-order predictor-corrector method

- To reduce the computational cost and storage requirements of our predictor-corrector method (2.7) and (2.8), we use the sum-of-exponentials (SOE) technique of [Shidong Jiang et al. Commun. Comput. Phys., 2017].
- For the history part, we replace the kernel $t^{\alpha-1}$ by its SOE approximation. Then

$$\bar{y}_{n+1}^P = y_0 + \frac{1}{\Gamma(\alpha)}\left(\sum_{i=1}^{N_{exp}} \varpi_i P_i^n + b_{n,0}^{n+1}\bar{g}_{n-1} + b_{n,1}^{n+1}\bar{g}_n\right), \tag{2.9}$$

$$\bar{y}_{n+1} = y_0 + \frac{1}{\Gamma(\alpha)}\left(\sum_{i=1}^{N_{exp}} \varpi_i P_i^n + a_{n,0}^{n+1}\bar{g}_n + a_{n,1}^{n+1} g(t_{n+1}, \bar{y}_{n+1}^P)\right). \tag{2.10}$$

where P_i^n can be get by using a recursive relation

$$P_i^n = e^{-s_i \tau_{n+1}} P_i^{n-1} + A_{i,0}^{n+1}\bar{g}_{n-1} + A_{i,1}^{n+1}\bar{g}_n \quad \text{for } 1 \le i \le N_{exp},\ 1 \le n \le N-1.$$

Fast evaluation of the second-order predictor-corrector method

Theorem 2.5

Recall that $y(t)$ *is the solution of* (2.6) *and* $\{\bar{y}_j\}_{j=0}^{N}$ *is the solution of the fast second-order predictor-corrector method* (2.9) *and* (2.10). *Then*

$$|y(t_j) - \bar{y}_j| \le C\epsilon + C \begin{cases} N^{-2r\alpha} & \text{for } r\alpha < 1, \\ N^{-2} \ln N & \text{for } r\alpha = 1, \\ N^{-2} & \text{for } r\alpha > 1. \end{cases} \quad 1 \le j \le N.$$

An FIVP numerical example

Example 2.6

Consider the FIVP ${}^{C}D_0^{\alpha} y(t) = y(t) - t^3 + \frac{6t^{3-\alpha}}{\Gamma(4-\alpha)}$ for $t \in (0,1]$, subject to $y(0) = 1$. Its exact solution is $y(t) = E_\alpha(t^\alpha) + t^3$, where

$$E_\alpha(s) = \sum_{k=0}^{\infty} \frac{s^k}{\Gamma(\alpha k + 1)}$$

is the standard Mittag-Leffler function. Thus,

$${}^{C}D_0^{\alpha} y(t) = \frac{6t^{3-\alpha}}{\Gamma(4-\alpha)} + 1 + \frac{t^\alpha}{\Gamma(\alpha+1)} + \frac{(t^\alpha)^2}{\Gamma(2\alpha+1)} + \dots,$$

that is, ${}^{C}D_0^{\alpha} y(t) \sim 1 + t^\alpha/\Gamma(\alpha+1)$ near $t = 0$.

An FIVP numerical example

Table: Maximum nodal errors, convergence orders and CPU times of PCM, fPCM and PCLiu for Example 2.6 with $\alpha = 0.4$.

		PCM			fPCM			PCLiu		
N	r	err_N	p_N	CPU	err_N^f	p_N	CPU	err_N^{Liu}	p_N	CPU
512	1	1.6747e–3	–	0.77	1.6747e–3	–	0.11	2.5478e–3	–	1.14
1024		8.8251e–4	0.9242	2.79	8.8251e–4	0.9242	0.21	9.6958e–4	1.3938	4.23
2048		4.7328e–4	0.8989	11.10	4.7328e–4	0.8989	0.39	4.7328e–4	1.0347	16.83
4096		2.5752e–4	0.8781	47.14	2.5752e–4	0.8781	0.80	2.5752e–4	0.8781	67.16
EOC			0.8			0.8			0.8	
512	$\frac{1+\alpha}{2\alpha}$	2.8666e–5	–	0.72	2.8666e–5	–	0.14	3.3740e–3	–	1.08
1024		1.0588e–5	1.4369	2.75	1.0588e–5	1.4369	0.26	1.2830e–3	1.3950	4.21
2048		3.9484e–6	1.4231	10.98	3.9484e–6	1.4231	0.54	4.8719e–4	1.3969	16.69
4096		1.4814e–6	1.4143	43.83	1.4814e–6	1.4143	1.12	1.8485e–4	1.3981	67.29
EOC			1.4			1.4			1.4	
512	$\frac{2}{2\alpha}$	3.3281e–5	–	0.73	3.3240e–5	–	0.17	5.0311e–3	–	1.08
1024		9.1429e–6	1.8640	2.77	9.1102e–6	1.8674	0.33	1.9151e–3	1.3934	4.22
2048		2.4145e–6	1.9209	10.91	2.3823e–6	1.9352	0.68	7.2759e–4	1.3962	16.85
4096		6.2386e–7	1.9525	43.55	5.3970e–7	2.1421	1.45	2.7613e–4	1.3978	67.13
EOC			2			2			1.4	
512	$\frac{2.5}{2\alpha}$	4.8397e–5	–	0.71	4.8377e–5	–	0.19	6.5953e–3	–	1.10
1024		1.3551e–5	1.8366	2.77	1.3751e–5	1.8148	0.39	2.5126e–3	1.3923	4.22
2048		3.6150e–6	1.9063	10.97	2.6800e–6	2.3592	0.80	9.5493e–4	1.3957	16.87
4096		9.3930e–7	1.9443	43.74	6.8790e–7	1.9619	1.69	3.6247e–4	1.3975	66.89
EOC			2			2			1.4	

Second-order predictor-corrector method for a nonlinear time-fractional Benjamin-Bona-Mahony-Burgers equation

Consider the following nonlinear time-fractional Benjamin-Bona-Mahony-Burgers (BBMB) initial-boundary value problem:

$$\begin{cases} {}^C D_0^\alpha (u - u_{xx}) + \gamma\, u u_x - \lambda\, u_{xx} = f(x,t,u) \quad \text{for } \ (x,t) \in \Omega \times (0,T], \\ u(x,0) = \varphi(x) \quad \text{for } x \in \bar{\Omega} \\ u(x,t) = 0 \quad \text{for } (x,t) \in \partial\Omega \times (0,T], \end{cases} \tag{2.11}$$

where $\lambda\ (\geq 0)$ and γ are constants, f satisfies the Lipschitz condition

$$|f(x,t,u) - f(x,t,v)| \leq L|u-v| \ \text{ for } L > 0,\ u,v \in \mathbb{R},\ (x,t) \in \bar{\Omega} \times [0,T]. \tag{2.12}$$

We assume that (2.11) has a unique solution which satisfies the following bounds for $(x,t) \in \bar{\Omega} \times (0,T]$:

$$\left|\frac{\partial^l u}{\partial x^l}(x,t)\right| \leq C \quad \text{for } l = 3,4; \tag{2.13a}$$

$$u(\cdot,t) \in C[0,T] \cap C^2(0,T], \quad \left|\frac{\partial^{l+k} u}{\partial x^l \partial t^k}(x,t)\right| \leq C(1 + t^{\alpha-k}) \text{for } l = 0,1,2,\ k = 0,1,2. \tag{2.13b}$$

Remarks

- For notational simplicity we assume that $\Omega = (x_L, x_R) \subset \mathbb{R}^1$; our method and its analysis can be extended without difficulty to rectangular domains $\Omega \subset \mathbb{R}^d$ with $d > 1$ whose sides are parallel to the coordinate planes.
- The BBMB equation is a mathematical model of the propagation of small amplitude long waves in certain nonlinear dispersive media system that improves the Korteweg-de Vries (KdV) equation.
- For efficient numerical methods for BBMB solution, see [Benjamin et al., Philos. Trans. Roy. Soc. London Ser. A, 1972; Dehghan Mehdi et al., Comput. Math. Appl., 2014; Qifeng Zhang and Lingling Liu, J. Sci. Comput., 2021] and their references.

Useful notations

- BBMB equation can be rewritten as the following integro-differential equation:

$$u(x,t) - u_{xx}(x,t) = G(x,t) + \frac{1}{\Gamma(\alpha)} \int_{s=0}^{t} (t-s)^{\alpha-1} F(x,s,u)\, ds$$
$$\text{for } (x,t) \in (x_L, x_R) \times (0,T], \tag{2.14}$$

where

$$G(x,t) := u(x,0) - u_{xx}(x,0) = \varphi(x) - (\varphi_{xx})(x),$$
$$F(x,t,u) := \lambda\, u_{xx}(x,t) - \gamma\,(uu_x)(x,t) + f(x,t,u(x,t)).$$

- Introduce the following notations:

$$h = (x_R - x_L)/M, \quad x_i = x_L + ih \ \text{ for } 0 \le i \le M,$$

$$\delta_x v_i = \frac{v_i - v_{i-1}}{h}, \quad \Delta_x v_i = \frac{v_{i+1} - v_{i-1}}{2h},$$

$$\delta_x^2 v_i = \frac{v_{i+1} - 2v_i + v_{i-1}}{h^2}, \quad J(v_i, w_i) = v_i \Delta_x w_i.$$

Second-order predictor-corrector method

Predictor-corrector method for (2.11):

$$\begin{cases} (u^P)_i^{n+1} - \delta_x^2 (u^P)_i^{n+1} = G(x_i, t_{n+1}) + \dfrac{1}{\Gamma(\alpha)} \Bigg[\displaystyle\sum_{j=0}^{n-1} \left(a_{j,0}^{n+1} F_i^j + a_{j,1}^{n+1} F_i^{j+1} \right) \\ \qquad\qquad + b_{n,0}^{n+1} F_i^{n-1} + b_{n,1}^{n+1} F_i^n \Bigg], \\ u_i^{n+1} - \delta_x^2 u_i^{n+1} = G(x_i, t_{n+1}) + \dfrac{1}{\Gamma(\alpha)} \Bigg[\displaystyle\sum_{j=0}^{n-1} \left(a_{j,0}^{n+1} F_i^j + a_{j,1}^{n+1} F_i^{j+1} \right) \\ \qquad\qquad + a_{n,0}^{n+1} F_i^n + a_{n,1}^{n+1} (F^P)_i^{n+1} \Bigg], \\ u_i^0 = \varphi(x_i) \quad \text{for } 0 \le i \le M, \qquad u_0^j = u_M^j = 0 \quad \text{for } 0 \le j \le N, \end{cases} \tag{2.15}$$

and

$$F_i^j := \lambda\, \delta_x^2 u_i^j - \gamma\, J\left(u_i^j, u_i^j \right) + f(x_i, t_j, u_i^j) \quad \text{for } 1 \le j \le n,$$

$$\left(F^P \right)_i^{n+1} := \lambda\, \delta_x^2 (u^P)_i^{n+1} - \gamma\, J\left((u^P)_i^{n+1}, (u^P)_i^{n+1} \right) + f\left(x_i, t_{n+1}, (u^P)_i^{n+1} \right).$$

Fast second-order predictor-corrector method

Fast predictor-corrector method for (2.11):

$$\begin{cases} (\bar{u}^P)_i^{n+1} - \delta_x^2 (\bar{u}^P)_i^{n+1} = G(x_i, t_{n+1}) + \dfrac{1}{\Gamma(\alpha)} \left(\displaystyle\sum_{i=1}^{N_{exp}} \varpi_i Q_i^n + b_{n,0}^{n+1} \bar{F}_i^{n-1} + b_{n,1}^{n+1} \bar{F}_i^n \right), \\ \bar{u}_i^{n+1} - \delta_x^2 \bar{u}_i^{n+1} = G(x_i, t_{n+1}) + \dfrac{1}{\Gamma(\alpha)} \left(\displaystyle\sum_{i=1}^{N_{exp}} \varpi_i Q_i^n + a_{n,0}^{n+1} \bar{F}_i^n + a_{n,1}^{n+1} (\bar{F}^P)_i^{n+1} \right), \\ Q_i^0 = 0, \quad Q_i^n = e^{-s_i \tau_{n+1}} Q_i^{n-1} + A_{i,0}^{n+1} \bar{F}_i^{n-1} + A_{i,1}^{n+1} \bar{F}_i^n \\ \quad \text{for } 1 \le i \le N_{exp}, \ 1 \le n \le N-1, \\ \bar{u}_i^0 = \varphi(x_i) \quad \text{for } 0 \le i \le M, \qquad \bar{u}_0^j = \bar{u}_M^j = 0 \quad \text{for } 0 \le j \le N, \end{cases} \tag{2.16}$$

where

$$\bar{F}_i^j := \lambda\, \delta_x^2 \bar{u}_i^j - \gamma\, J\left(\bar{u}_i^j, \bar{u}_i^j\right) + f(x_i, t_j, \bar{u}_i^j) \quad \text{for } 1 \le j \le n,$$

$$\left(\bar{F}^P\right)_i^{n+1} := \lambda\, \delta_x^2 (\bar{u}^P)_i^{n+1} - \gamma\, J\left((\bar{u}^P)_i^{n+1}, (\bar{u}^P)_i^{n+1}\right) + f\left(x_i, t_{n+1}, (\bar{u}^P)_i^{n+1}\right).$$

Error analysis of the predictor-corrector method

Lemma 2.7

Let the sequences $\{k_{\theta,j}\}_{j=0}^{N}$ *for* $\theta = 1, 2, \ldots, 6$ *satisfy* $k_{\theta,j} \geq 0$ *for all* θ *and* j. *Let* $d_0 \geq 0$. *Assume that the sequences* $\{\phi_n\}_{n=0}^{N}$ *and* $\left\{\phi_n^P\right\}_{n=0}^{N}$ *satisfy*

$$\begin{cases} 0 \leq \phi_0 \leq d_0, \\ 0 \leq \phi_{n+1}^P \leq d_0 + \sum_{j=0}^{n} \left(k_{1,j}\phi_j + k_{2,j}\phi_j^2\right) \quad \text{for } 0 \leq n \leq N-1, \\ 0 \leq \phi_{n+1} \leq d_0 + \sum_{j=0}^{n} \left(k_{3,j}\phi_j + k_{4,j}\phi_j^2\right) + k_{5,n+1}\phi_{n+1}^P + k_{6,n+1}(\phi_{n+1}^P)^2 \\ \quad \text{for } 0 \leq n \leq N-1. \end{cases} \tag{2.17}$$

For $n = 0, 1, \ldots, N-1$, *define*

$$C_0^n := \max\Bigg\{1 + k_{3,0} + k_{4,0}d_0 + k_{5,1}(1 + k_{1,0} + k_{2,0}d_0) + k_{6,1}d_0(1 + k_{1,0} + k_{2,0}d_0)^2,$$
$$(1 + k_{5,n+1} + k_{6,n+1}) \exp\left[\sum_{j=0}^{n} (k_{3,j} + k_{4,j}) + (k_{5,n+1} + k_{6,n+1}) \sum_{j=0}^{n} (k_{1,j} + k_{2,j})\right]\Bigg\},$$
$$\hat{C}_0^n := 1 + \left(\max_{0 \leq i \leq n-1} C_0^i\right) \sum_{j=0}^{n} (k_{1,j} + k_{2,j}).$$

Assume that $C_0^n d_0 \leq 1$ *and* $\hat{C}_0^n d_0 \leq 1$ *for* $0 \leq n \leq N-1$. *Then one has*

$$\phi_{n+1} \leq C_0^n d_0 \quad \text{for } 0 \leq n \leq N-1. \tag{2.18}$$

Error analysis of the predictor-corrector method

Theorem 2.8

Recall that u *and* $\{u_i^n\}$ *are the solutions of* (2.11) *and* (2.15), *respectively. Assume that* (2.12) *and* (2.13) *hold true. Then for all sufficiently large* N *and all sufficiently small* h

$$\left\| u(\cdot, t_j) - u^j \right\|_{H^1} \le Ch^2 + C \begin{cases} N^{-2r\alpha} & \text{for } r\alpha < 1, \\ N^{-2} \ln N & \text{for } r\alpha = 1, \\ N^{-2} & \text{for } r\alpha > 1. \end{cases} \quad 1 \le j \le N.$$

Theorem 2.9

Recall that u *and* $\{\bar{u}_i^n\}$ *are the solutions of* (2.11) *and* (2.16), *respectively. Assume that* (2.12) *and* (2.13) *hold true. Then for all sufficiently large* N *and all sufficiently small* h

$$\left\| u(\cdot, t_j) - \bar{u}^j \right\|_{H^1} \le C\epsilon + Ch^2 + C \begin{cases} N^{-2r\alpha} & \text{for } r\alpha < 1, \\ N^{-2} \ln N & \text{for } r\alpha = 1, \\ N^{-2} & \text{for } r\alpha > 1. \end{cases} \quad 1 \le j \le N.$$

Remark:

$$\langle v, w \rangle = h \sum_{i=1}^{M-1} v_i w_i, \quad \|v\| = \sqrt{\langle v, v \rangle}, \quad |v|_1 = \sqrt{\langle \delta_x v, \delta_x v \rangle}, \quad \|v\|_{H^1} = \sqrt{\|v\|^2 + |v|_1^2}.$$

Numerical example

Example 2.10

Consider the following nonlinear time-fractional BBMB equation

$$ {}^{C}D_0^{\alpha}(u - u_{xx}) + uu_x - u_{xx} - u(1-u) = f(x,t) \quad \text{for } x = (x_1, x_2) \in \Omega,\ t \in (0,1], $$

where $\Omega = (0,\pi) \times (0,\pi)$; the function f and the initial-boundary value conditions are determined by the analytical solution $u(x_1, x_2, t) = \frac{1}{2}(t^{\alpha} + t^{2\alpha})(\sin x_1)(\sin x_2)$, which has the weakly singular behaviour assumed in (2.13). One can check that

$$ {}^{C}D_0^{\alpha}(u - u_{xx})(x,t) = \frac{3}{2}\left[\Gamma(\alpha+1) + \frac{2\alpha\Gamma(2\alpha)}{\Gamma(\alpha+1)} t^{\alpha}\right](\sin x_1)(\sin x_2); $$

that is, ${}^{C}D_0^{\alpha}(u - u_{xx})(x,t)$ behaves like $C(1 + t^{\alpha})$ near $t = 0$ for each fixed x.

Numerical example

Table: Global errors, convergence orders and CPU times of PCM, fPCM and PCLiu for Example 2.10 with $\alpha = 0.4$ and $M = N$.

		PCM			fPCM			PCLiu		
N	r	$E(M,N)$	p_t	CPU	$E(M,N)^f$	p_t	CPU	$E(M,N)^{Liu}$	p_t	CPU
12	1	2.4709e–2	–	0.04	2.4709e–2	–	0.04	1.8042e–2	–	0.04
24		1.3372e–2	0.8859	0.45	1.3372e–2	0.8859	0.39	6.0026e–3	1.5877	0.45
48		7.0795e–3	0.9175	25.72	7.0795e–3	0.9175	25.33	5.2572e–3	0.1913	25.95
96		3.7852e–3	0.9033	1351.18	3.7852e–3	0.9033	1384.25	3.7096e–3	0.5030	1391.36
EOC			0.8			0.8			0.8	
12	$\frac{1+\alpha}{2\alpha}$	9.8060e–3	–	0.03	9.8057e–3	–	0.03	2.6408e–2	–	0.03
24		2.6098e–3	1.9097	0.41	2.6098e–3	1.9097	0.40	8.0697e–3	1.7104	0.40
48		8.3609e–4	1.6422	22.58	8.3609e–4	1.6422	22.38	2.5415e–3	1.6668	22.13
96		3.2204e–4	1.3764	1376.02	3.2204e–4	1.3764	1345.20	8.2799e–4	1.6180	1385.82
EOC			1.4			1.4			1.4	
12	$\frac{2}{2\alpha}$	8.1472e–3	–	0.02	8.1469e–3	–	0.03	3.6417e–2	–	0.02
24		2.3802e–3	1.7752	0.41	2.3798e–3	1.7754	0.40	1.1454e–2	1.6687	0.40
48		6.3484e–4	1.9066	22.15	6.3441e–4	1.9074	21.75	3.6840e–3	1.6365	21.59
96		1.6359e–4	1.9564	1282.64	1.6313e–4	1.9594	1272.54	1.2202e–3	1.5942	1351.20
EOC			2			2			1.4	
12	$\frac{2.5}{2\alpha}$	5.8548e–3	–	0.03	5.8545e–3	–	0.03	4.5929e–2	–	0.03
24		2.0454e–3	1.5172	0.42	2.0451e–3	1.5174	0.40	1.4753e–2	1.6384	0.40
48		5.8051e–4	1.8170	22.61	5.8008e–4	1.8178	22.28	4.7998e–3	1.6199	22.31
96		1.5361e–4	1.9180	1388.66	1.5316e–4	1.9212	1381.92	1.6015e–3	1.5835	1395.45
EOC			2			2			1.4	

Numerical example

Table: Global errors, convergence orders and CPU times of PCM, fPCM and PCLiu for Example 2.10 with $\alpha = 0.8$, $r = 2/(2\alpha)$ and $N = 10000$.

	PCM			fPCM			PCLiu		
M	$E(M,N)$	p_x	CPU	$E(M,N)^f$	p_x	CPU	$E(M,N)^{Liu}$	p_x	CPU
6	9.0018e–2	–	101.41	9.0017e–2	–	5.65	9.0018e–2	–	159.01
8	2.4275e–2	1.8907	114.83	2.4274e–2	1.8908	12.00	2.4275e–2	1.8907	178.08
16	6.3393e–3	1.9371	177.68	6.3380e–3	1.9373	47.99	6.3393e–3	1.9371	259.29
32	1.6196e–3	1.9686	788.40	1.6183e–3	1.9695	548.89	1.6197e–3	1.9686	862.69
EOC		2			2			2	

Third-order predictor-corrector method

- The proposed method is analyzed under the following regularity assumptions on the solution:

$$y \in C[0,T] \cap C^3(0,T] \text{ with } |y^{(k)}(t)| \leq C(1+t^{\alpha-k}) \text{ for } k = 0,1,2,3,\ t \in (0,T]. \tag{3.1}$$

- Then, $z := {}^C D_0^\alpha y$ satisfies that

$$z \in C[0,T] \cap C^3(0,T], \quad |z^{(k)}(t)| \leq C(1+t^{\alpha-k}) \quad \text{for } k = 0,1,2,3,\ t \in (0,T]. \tag{3.2}$$

Third-order predictor-corrector method

At each mesh point $t = t_{n+1}$, one can rewrite (1.3) as

$$\begin{aligned} y(t_{n+1}) = y_0 &+ \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{n-1} \int_{s=t_j}^{t_{j+1}} (t_{n+1} - s)^{\alpha-1} g(s, y(s))\, ds \\ &+ \frac{1}{\Gamma(\alpha)} \int_{s=t_n}^{t_{n+1}} (t_{n+1} - s)^{\alpha-1} g(s, y(s))\, ds. \end{aligned} \tag{3.3}$$

The prediction stage:

- When $n = 0$, we use $g_y(t_0)$ to approximate $g_y(t)$ on the interval $[t_0, t_1]$;
- When $n = 1$, we use $\Pi_{1,0} g_y(t)$ to approximate $g_y(t)$ on the intervals $[t_0, t_1]$ and $[t_1, t_2]$;
- When $n \geq 2$, we use $\Pi_{1,0} g_y(t)$ to approximate $g_y(t)$ on the first small interval $[t_0, t_1]$, $\Pi_{2,j} g_y(t)$ to approximate $f_y(t)$ on each interval $[t_j, t_{j+1}]$ $(j = 1, 2, \ldots, n-1)$ and $\Pi_{2,n-1} g_y(t)$ to approximate $g_y(t)$ on the last small interval $[t_n, t_{n+1}]$.

$$y_{n+1}^P = y_0 + \frac{1}{\Gamma(\alpha)} \left(\sum_{j=0}^{n} d_j^{n+1} g_j + c_{n,-1}^{n+1} g_{n-2} + c_{n,0}^{n+1} g_{n-1} + c_{n,1}^{n+1} g_n \right). \tag{3.4}$$

The correction stage: we use $\Pi_{1,0} g_y(t)$ to approximate $g_y(t)$ on the first small interval $[t_0, t_1]$ and $\Pi_{2,j} g_y(t)$ to approximate $g_y(t)$ on the intervals $[t_j, t_{j+1}]$ $(j = 1, 2, \ldots, n)$.

$$y_{n+1} = y_0 + \frac{1}{\Gamma(\alpha)} \left(\sum_{j=0}^{n} d_j^{n+1} g_j + b_{n,-1}^{n+1} g_{n-1} + b_{n,0}^{n+1} g_n + b_{n,1}^{n+1} g_{n+1}^P \right). \tag{3.5}$$

Error analysis of the third-order predictor-corrector method

Lemma 3.1

Let $w \in C[0,T] \cap C^3(0,T]$. *Suppose that* $|w^{(k)}(t)| \le C(1+t^{\alpha-k})$ *for* $k=0,1,2,3,\ t \in (0,T]$. *For* $n \ge 0$, *we define*

$$I_3^{n+1} = \left| \int_{s=t_0}^{t_1} (t_{n+1}-s)^{\alpha-1}(w-\Pi_{1,0}w)(s)\,ds + \sum_{j=1}^{n} \int_{s=t_j}^{t_{j+1}} (t_{n+1}-s)^{\alpha-1}(w-\Pi_{2,j}w)(s)\,ds \right|,$$

$$\begin{aligned} I_4^{n+1} = \Bigg| & \int_{s=t_0}^{t_1} (t_{n+1}-s)^{\alpha-1}(w-\Pi_{1,0}w)(s)\,ds + \sum_{j=1}^{n-1} \int_{s=t_j}^{t_{j+1}} (t_{n+1}-s)^{\alpha-1}(w-\Pi_{2,j}w)(s)\,ds \\ & + \int_{s=t_n}^{t_{n+1}} (t_{n+1}-s)^{\alpha-1}(w-\Pi_{2,n-1}w)(s)\,ds \Bigg|. \end{aligned}$$

Then

$$I_3^{n+1} + I_4^{n+1} \le C \begin{cases} N^{-2r\alpha} & \text{for } 1 \le r < \frac{3}{2\alpha}, \\ N^{-3}\ln N & \text{for } r = \frac{3}{2\alpha}, \\ N^{-3} & \text{for } r > \frac{3}{2\alpha}. \end{cases} \qquad 0 \le n \le N-1.$$

Error analysis of the third-order predictor-corrector method

We can now prove an error bound for our predictor-corrector solution.

Theorem 3.2

Assume that $y(t_j)$ *and* $\{y_j\}_{j=0}^{N}$ *are the solutions of* (2.6) *and the third-order predictor-corrector method* (3.4), (3.5), *respectively. Assume also that* (3.1) *holds true. Then*

$$|y(t_j) - y_j| \leq C \begin{cases} N^{-2r\alpha} & \text{for } 1 \leq r < \frac{3}{2\alpha}, \\ N^{-3}\ln N & \text{for } r = \frac{3}{2\alpha}, \\ N^{-3} & \text{for } r > \frac{3}{2\alpha}. \end{cases} \qquad 0 \leq n \leq N-1.$$

Fast evaluation of the third-order predictor-corrector method

For the history part, we replace the kernel $t^{\alpha-1}$ by its SOE approximation, Then, we get the fast predictor-corrector method

$$\begin{cases} \bar{y}_{n+1}^P = y_0 + \dfrac{1}{\Gamma(\alpha)} \left(\sum\limits_{i=1}^{N_{exp}} \varpi_i \bar{p}_i^n + c_{n,-1}^{n+1} \bar{g}_{n-2} + c_{n,0}^{n+1} \bar{g}_{n-1} + c_{n,1}^{n+1} \bar{g}_n \right), \\ \bar{y}_{n+1} = y_0 + \dfrac{1}{\Gamma(\alpha)} \left(\sum\limits_{i=1}^{N_{exp}} \varpi_i \bar{p}_i^n + b_{n,-1}^{n+1} \bar{g}_{n-1} + b_{n,0}^{n+1} \bar{g}_n + b_{n,1}^{n+1} \bar{g}_{n+1}^P \right), \\ \bar{p}_i^0 = 0, \quad \bar{p}_i^1 = \int_{s=t_0}^{t_1} e^{-s_i(t_{n+1}-s)} (L_{0,0}\bar{g}_0 + L_{0,1}\bar{g}_1)\, ds \quad \text{for } i = 1, 2, \ldots, N_{exp}, \\ \bar{p}_i^n = e^{-s_i \tau_{n+1}} \bar{p}_i^{n-1} + A_{i,-1}^{n+1} \bar{g}_{n-2} + A_{i,0}^{n+1} \bar{g}_{n-1} + A_{i,1}^{n+1} \bar{g}_n \quad \text{for } i = 1, 2, \ldots, N_{exp}, \ n = 2, 3, \ldots, N. \end{cases} \tag{3.6}$$

Fast evaluation of the third-order predictor-corrector method

Theorem 3.3

Assume that $y(t_j)$ *and* $\{\bar{y}_j\}_{j=0}^N$ *are the solutions of* (2.6) *and the fast third-order predictor-corrector method* (3.6), *respectively. Assume also that* (3.1) *holds true. Then*

$$|y(t_j) - \bar{y}_j| \leq C\epsilon + C \begin{cases} N^{-2r\alpha} & \text{for } 1 \leq r < \frac{3}{2\alpha}, \\ N^{-3} \ln N & \text{for } r = \frac{3}{2\alpha}, \\ N^{-3} & \text{for } r > \frac{3}{2\alpha}. \end{cases} \qquad 1 \leq j \leq N.$$

Numerical example

Example 3.4

Consider the following FDEs with $\alpha \in (0,1)$:

$$ {}^{C}D_0^{\alpha} y(t) = -y(t), \quad t \in (0,1]; \qquad y(0) = 1. \tag{3.7}$$

The exact solution of (3.7) is $y(t) = E_\alpha(-t^\alpha)$, where

$$E_\alpha(s) = \sum_{k=0}^{\infty} \frac{s^k}{\Gamma(\alpha k + 1)}$$

is the Mittag-Leffler function. Since

$${}^{C}D_0^{\alpha} y(t) = -1 - \frac{-t^\alpha}{\Gamma(\alpha+1)} - \frac{(-t^\alpha)^2}{\Gamma(2\alpha+1)} - \dots,$$

that is, ${}^{C}D_0^{\alpha} y(t)$ behaves as $C(1 + t^\alpha)$.

Numerical example

Table: Global errors, convergence orders and CPU times of PCM and fPCM for problem (3.7) with $\alpha = 0.5$.

		PCM			fPCM		
N	r	err_N	p	CPU	err_N^f	p	CPU
64	1	1.1732e–3	–	2.58	1.1732e–3	–	3.34
128		6.9056e–4	0.7646	9.51	6.9056e–4	0.7646	6.95
256		4.1422e–4	0.7374	39.85	4.1422e–4	0.7374	14.99
512		2.3219e–4	0.8351	162.84	2.3219e–4	0.8351	32.26
EOC			1			1	
64	$r = \frac{2}{2\alpha}$	1.0150e–4	–	2.41	1.0150e–4	–	4.68
128		1.8584e–5	2.4493	9.61	1.8584e–5	2.4493	9.97
256		4.2737e–6	2.1205	39.23	4.2737e–6	2.1205	22.57
512		1.0898e–6	1.9715	157.12	1.0898e–6	1.9715	48.61
EOC			2			2	
64	$r = \frac{3}{2\alpha}$	8.3324e–6	–	2.44	8.3324e–6	–	5.71
128		8.1803e–7	3.3485	9.53	8.1803e–7	3.3485	13.35
256		9.6599e–8	3.0821	39.48	9.6599e–8	3.0821	29.22
512		1.2096e–8	2.9975	159.67	1.2096e–8	2.9975	65.48
EOC			3			3	
64	$r = \frac{4}{2\alpha}$	3.5974e–6	–	2.41	3.5974e–6	–	6.95
128		3.6817e–7	3.2885	8.71	3.6817e–7	3.2885	15.04
256		4.1714e–8	3.1418	35.14	4.1714e–8	3.1418	33.36
512		4.9751e–9	3.0677	142.50	4.9751e–9	3.0677	73.69
EOC			3			3	

Numerical example

Example 3.5

Consider the following Benjamin-Bona-Mahony-Burgers equation

$$
{}^{C}D_0^{\alpha}(u - u_{xx}) + uu_x - u_{xx} = f(x,t) \quad \text{for } (x,t) \in (0,1) \times (0,1], \tag{3.8a}
$$

$$
u(x,0) = \sin(\pi x) \quad \text{for } x \in [0,1], \qquad u(0,t) = u(1,t) = 0 \quad \text{for } t \in (0,1], \tag{3.8b}
$$

the function f, the initial-boundary value conditions are determined by the exact solution $u(x,t) = (1 + t^{\alpha} + t^{2\alpha}) \sin(\pi x)$. One can check that

$$
{}^{C}D_0^{\alpha}(u - u_{xx})(x,t) = \left[\Gamma(\alpha+1) + \frac{2\alpha\Gamma(2\alpha)}{\Gamma(1+\alpha)} t^{\alpha}\right] (1+\pi^2) \sin(\pi x)
$$

behaves as $C(1 + t^{\alpha})$.

Numerical example

Table: Global errors and convergence orders of PCM and fPCM for problem (3.8) with $r = 3/(2\alpha)$ and $M = 8000$.

Scheme	N	$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
		$E_{M,N}$	p_t	$E_{M,N}$	p_t	$E_{M,N}$	p_t
PCM	12	6.4472e–2	–	3.8631e–3	–	4.8683e–4	–
	24	3.1108e–3	4.3733	2.5987e–4	3.8939	4.4781e–5	3.4425
	48	1.7218e–4	4.1753	2.2876e–5	3.5058	5.7787e–6	2.9541
	96	1.1986e–5	3.8445	2.4634e–6	3.2151	7.9723e–7	2.8577
EOC			3		3		3
fPCM	12	6.4472e–2	–	3.8631e–3	–	4.8683e–4	–
	24	3.1108e–3	4.3733	2.5987e–4	3.8939	4.4781e–5	3.4425
	48	1.7218e–4	4.1753	2.2876e–5	3.5058	5.7787e–6	2.9541
	96	1.1986e–5	3.8445	2.4634e–6	3.2151	7.9723e–7	2.8577
EOC			3		3		3

Numerical example

Table: Global errors, convergence orders and CPU times of PCM and fPCM for problem (3.8) with $\alpha = 0.8$, $r = 3/(2\alpha)$ and $N = 2000$.

	PCM			fPCM		
M	$E(M,N)$	p_x	CPU	$E(M,N)^f$	p_x	CPU
8	8.9024e–2	1.9377	2141.30	8.9024e–2	1.9377	134.57
16	2.2690e–2	1.9722	2131.37	2.2690e–2	1.9722	133.85
32	5.7260e–3	1.9864	2164.59	5.7260e–3	1.9864	132.92
64	1.4382e–3	1.9932	2158.93	1.4382e–3	1.9933	137.41
EOC		2			2	

Concluding remarks

- The second-order and third-order predictor-corrector methods of Nguyen and Jang [Fract. Calc. Appl. Anal., 2017, 447–476] are generalised to graded meshes to solve nonlinear fractional initial-value problems whose typical solutions have a weak singularity at the initial time.
- In comparison with existing predictor-corrector methods in the literature, this new methods significantly improve the numerical accuracy while reducing the computational cost.
- To increase its computational efficiency still further, the corresponding fast algorithms based on the sum-of-exponentials approximation to the kernel of the scheme are described.
- The methods (and its fast variant) are then extended to solve the nonlinear time-fractional Benjamin-Bona-Mahony-Burgers (BBMB) initial-boundary value problem, combined with a standard discretisation of the spatial derivatives on a uniform mesh.
- Several numerical experiments show the sharpness of our theoretical error bounds for both problems.

Thank you for your attention!

Yongtao Zhou

yongtaozhou@csrc.ac.cn