On the Extension of Adams–Bashforth–Moulton Methods for Numerical Integration of Delay Differential Equations and Application to the Moon's Orbit

Dan Aksim and Dmitry Pavlov

Laboratory of Ephemeris Astronomy
Institute of Applied Astronomy of the Russian Academy of Sciences
St. Petersburg, Russia

May 11, 2018

Types of differential equations

Ordinary differential equation (ODE):

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t))$$

(Retarded) delay differential equation (DDE):

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(\varphi(t)), ...),\ \varphi(t) < t$$

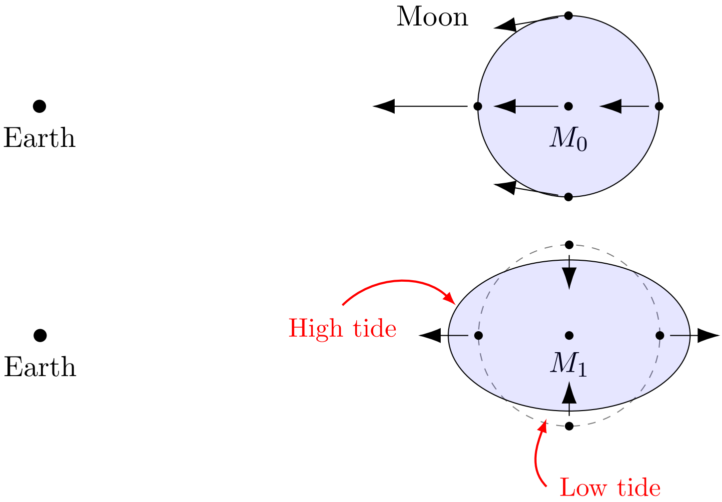
Advanced differential equation:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(\psi(t)), ...),\ \psi(t) > t$$

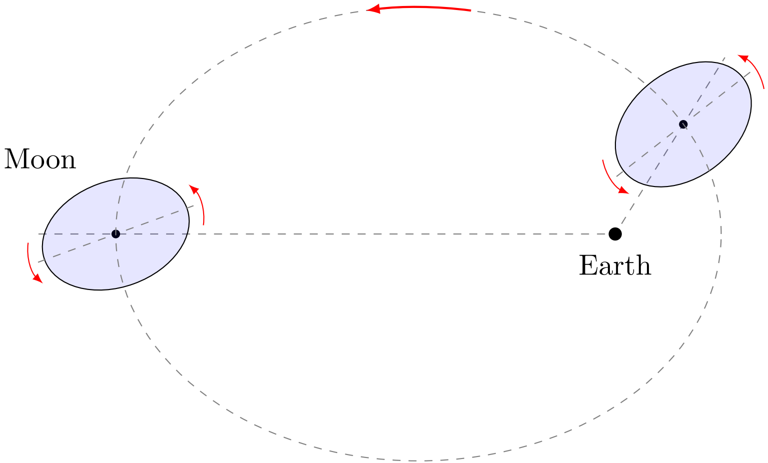
DDE of neutral type:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \dot{\mathbf{x}}(\xi(t)), ...),\ \xi(t) \neq t$$

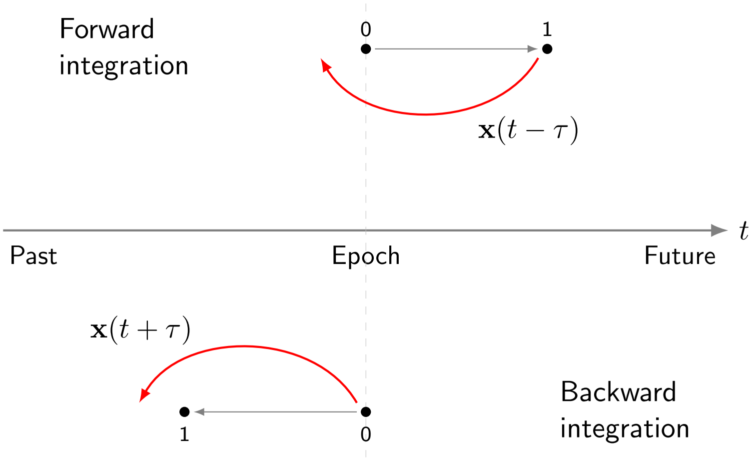
Tidal forces (I)



Tidal forces (II)



From retarded to advanced equations



Rotational equation of the Moon (general form)

Forward: retarded DDE of neutral type with constant delays

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t-\tau), \dot{\mathbf{x}}(t-\tau))$$

Backward: advanced DDE of neutral type with constant delays

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t+\tau), \dot{\mathbf{x}}(t+\tau))$$

Initial condition at the epoch:

$$\mathbf{x}(t_0) = \mathbf{x}_0$$

Rotational equation of the Moon (actual form)

Euler's equation for a rotating reference frame:

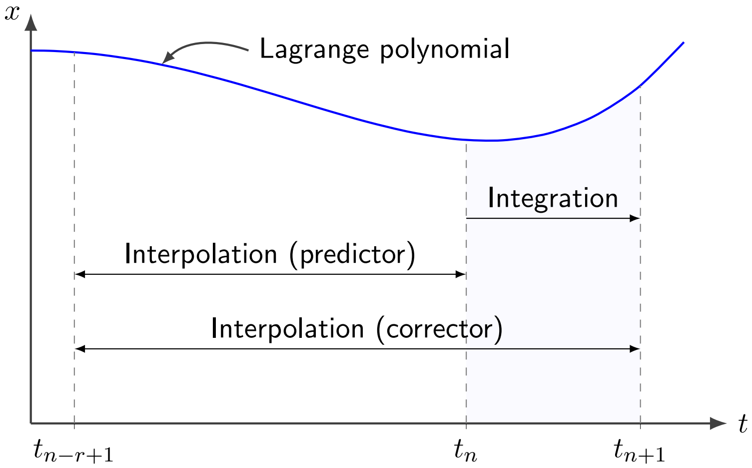
$$\dot{\boldsymbol{\omega}} = \left(\frac{I}{m}\right)^{-1} \left[\frac{\mathbf{N}}{m} - \frac{\dot{I}}{m}\boldsymbol{\omega} - \boldsymbol{\omega} \times \left(\frac{I}{m}\boldsymbol{\omega}\right)\right],$$

$\boldsymbol{\omega}$ — angular velocity,
$\mathbf{N}(t)$ — torque,
I/m — inertia tensor

$$\frac{I}{m} = \frac{I_0}{m} - \frac{I_c}{m} - k_2 \frac{\mu_E}{\mu_M}\left(\frac{R_M}{r}\right)^5 \begin{bmatrix} x^2 - \frac{1}{3}r^2 & xy & xz \\ xy & y^2 - \frac{1}{3}r^2 & yz \\ xz & yz & z^2 - \frac{1}{3}r^2 \end{bmatrix}$$
$$+ k_2 \frac{R_M^5}{3\mu_M} \begin{bmatrix} \omega_x^2 - \frac{1}{3}(\omega^2 - n^2) & \omega_x\omega_y & \omega_x\omega_z \\ \omega_x\omega_y & \omega_y^2 - \frac{1}{3}(\omega^2 - n^2) & \omega_y\omega_z \\ \omega_x\omega_z & \omega_y\omega_z & \omega_z^2 - \frac{1}{3}(\omega^2 + 2n^2) \end{bmatrix},$$

where $\mathbf{r} = (x, y, z)^{\mathrm{T}} = \mathbf{r}(t-\tau)$, $\boldsymbol{\omega} = \boldsymbol{\omega}(t-\tau)$, $\tau = 0.096\ \mathrm{d}$

Adams–Bashforth–Moulton methods (I)



Adams–Bashforth–Moulton methods (II)

1. Predictor — Adams–Bashforth (order 2):

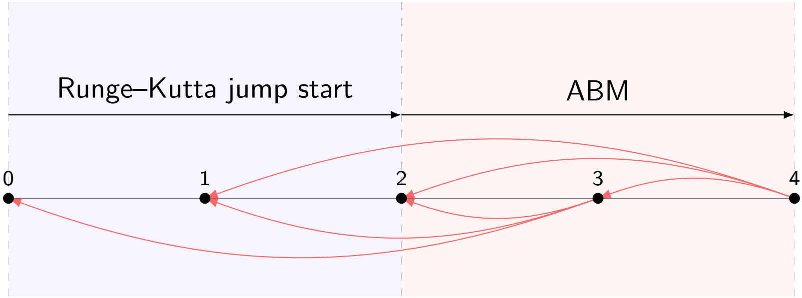
$$\mathbf{x}_{n+2} = \mathbf{x}_{n+1} + h\left(\frac{3}{2}\mathbf{f}_{n+1} - \frac{1}{2}\mathbf{f}_n\right)$$

2. Evaluation of $\mathbf{f}_{n+2} = \mathbf{f}(t_{n+2}, \mathbf{x}_{n+2})$
3. Corrector — Adams–Moulton (order 3):

$$\mathbf{x}_{n+2} = \mathbf{x}_{n+1} + h\left(\frac{5}{12}\mathbf{f}_{n+2} + \frac{2}{3}\mathbf{f}_{n+1} - \frac{1}{12}\mathbf{f}_n\right)$$

4. (Optional). PECE, PECEC, PECECE

Adams–Bashforth–Moulton methods (III)



First $(r-1)$ steps must be performed by a single-step method.

The 'nested RK4' method for DDEs

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t \pm \tau), \dot{\mathbf{x}}(t \pm \tau))$$

$$\mathbf{x}(t_0) = \mathbf{x}_0$$

1. Introduce a new function

$$\mathbf{g}(t) = \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t), \dot{\mathbf{x}}(t))$$

2. Retrieve delayed states by integrating $\mathbf{g}(t)$ with RK4

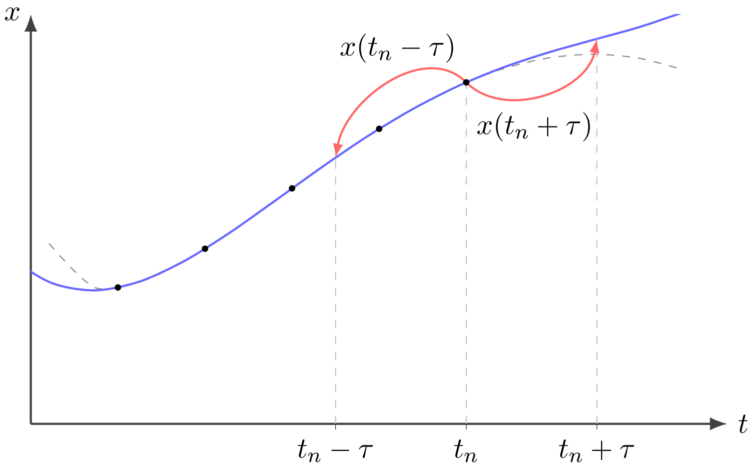
$$\mathbf{x}(t \pm \tau) = {}^{\mathrm{RK4}}\mathcal{I}_{t \to t \pm \tau}\mathbf{g}(t)$$

$$\dot{\mathbf{x}}(t \pm \tau) = \mathbf{g}(t \pm \tau)$$

Drawbacks

- Calculation of each delayed state requires 4 RHS calls
- No previous knowledge of $\mathbf{x}(t)$ is being used

The interpolation method for DDEs (I)



The interpolation method for DDEs (II)

The algorithm

1. Jump start by the 'nested RK4' algorithm (with 8th order Dormand–Prince)
2. P and C stages are simple ABM
3. Each E stage constructs a Lagrange interpolating polynomial (any order), which is used to find $\mathbf{x}(t \pm \tau)$ and $\dot{\mathbf{x}}(t \pm \tau)$

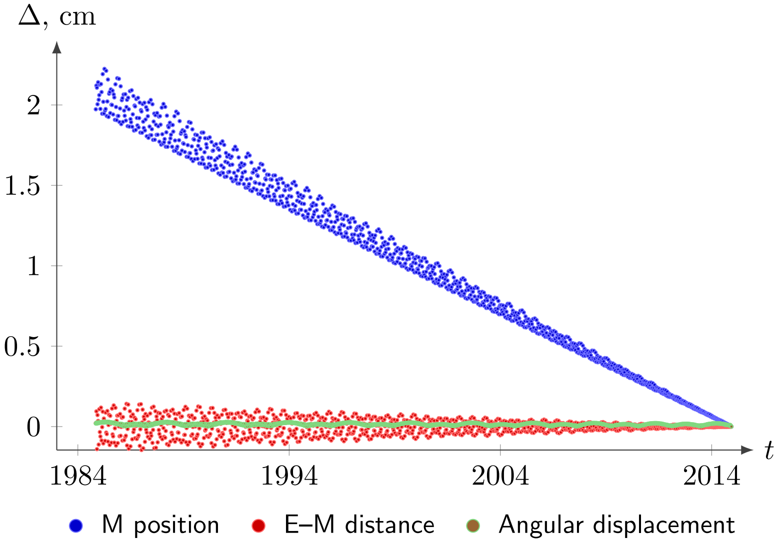
Advantages

- Much cheaper delayed states
- No simplifying assumptions about the function

Other methods

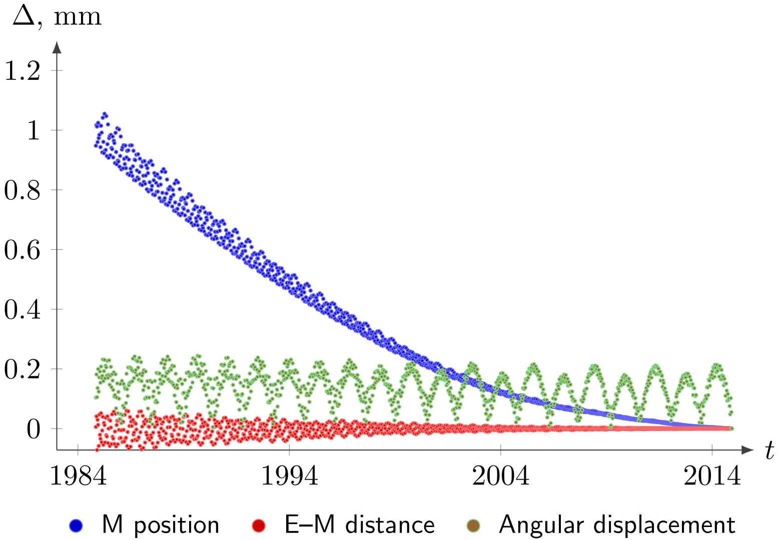
1. **Pavlov, IAA RAS**
   Nested Runge–Kutta method (described above)
2. **Hofmann, Institut für Erdmessung**
   Quadratic Taylor expansion using $\mathbf{x}$, $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$
3. **Williams, NASA JPL**
   Approximation with pre-fit polynomials (exact algorithm unknown)

Results. The forward–backward test



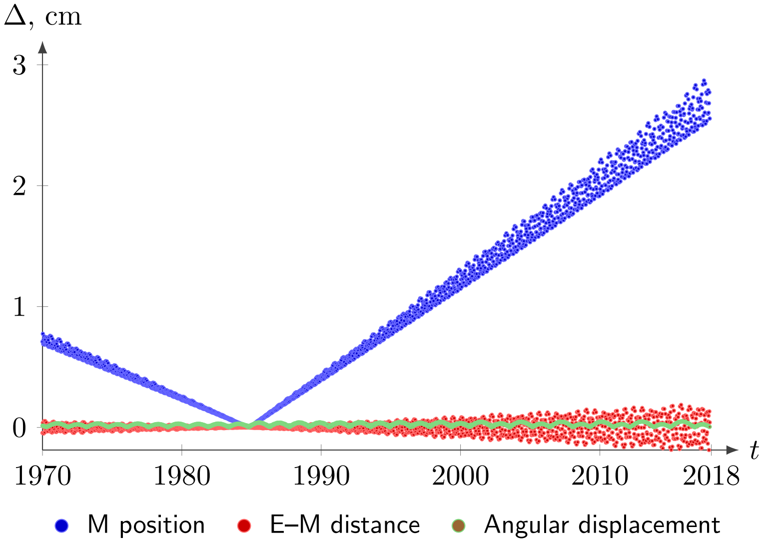
Orders: ABM — 13, interpolation and extrapolation — 9

Results. The forward–backward test (no EGM)



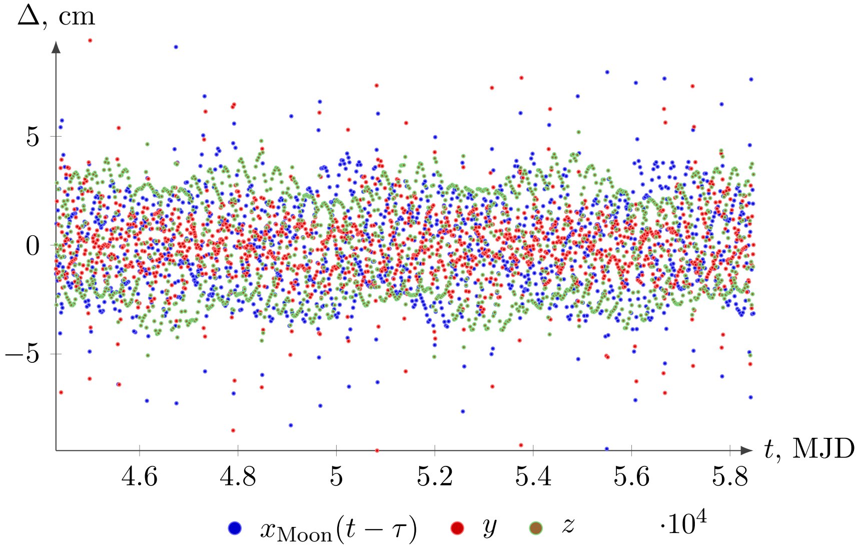
Orders: ABM — 13, interpolation and extrapolation — 9

Results. The old–new test

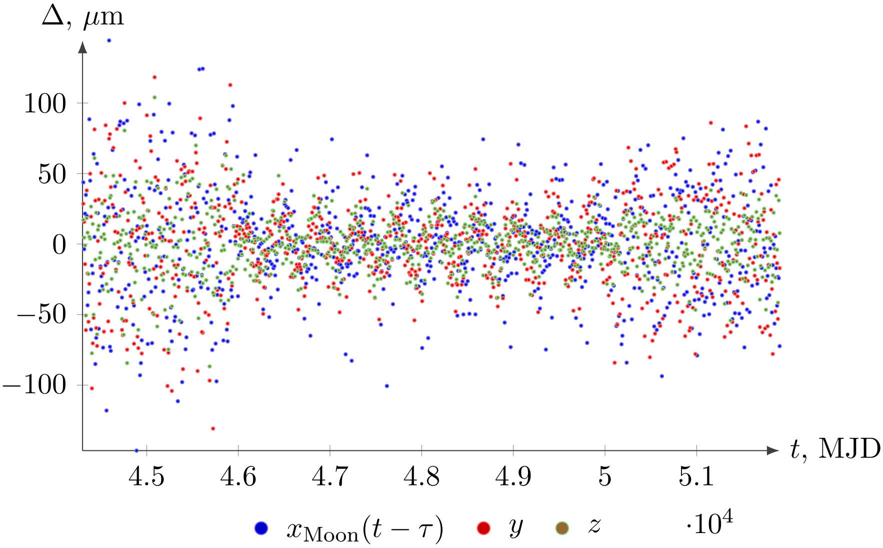


Orders: ABM — 13, interpolation and extrapolation — 9

Results. Approximation accuracy (RK4)



Results. Approximation accuracy (Lagrange)



Results. O−C

Station	Timespan	NPs	One-way wrms, cm
McDonald	1969–1985	3552/52	19.9
MLRS1	1983–1988	588/43	11.2
MLRS2	1988–2015	3216/454	3.4
Haleakala	1984–1990	748/22	5.7
OCA (Ruby)	1984–1986	1109/79	17.0
OCA (YAG)	1987–2005	8203/121	2.0
OCA (MeO)	2009–2017	1814/22	1.42
OCA (IR)	2015–2017	1814/20	1.27 (old), 1.26 (new)
APOLLO	2006–2016	2609/39	1.37

Backup slides

Runge–Kutta methods

General form for the Runge–Kutta family of methods:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h \sum_{i=1}^{s} b_i \mathbf{k}_i$$

$$\mathbf{k}_s = \mathbf{f}(t_n + c_s h, \mathbf{x}_n + h \sum_{j=1}^{s-1} a_{s,j} \mathbf{k}_j)$$

Drawbacks

- Butcher barriers:
  - $p \geq 5$: no RK method exists of order p with $s = p$ stages
  - $p \geq 7$: no RK method exists of order p with $s = p + 1$ stages
  - $p \geq 8$: no RK method exists of order p with $s = p + 2$ stages
- Higher orders are problematic