Numerical methods for dynamical systems

Alexandre Chapoutot

ENSTA Paris
master CPS IP Paris

2020-2021
Part VIII

Numerical methods for IVP-DDE
Initial Value Problem of Ordinary Differential Equations

Consider an IVP for ODE, over the time interval \([0, t_{\text{end}}]\)

\[
\dot{y} = f(t, y) \quad \text{with} \quad y(0) = y_0
\]

IVP has a unique solution \(y(t; y_0)\) if \(f : \mathbb{R}^n \rightarrow \mathbb{R}^n\) is Lipschitz in \(y\)

\[
\forall t, \forall y_1, y_2 \in \mathbb{R}^n, \exists L > 0, \quad \| f(t, y_1) - f(t, y_2) \| \leq L \| y_1 - y_2 \|.
\]

Goal of numerical integration

- Compute a sequence of time instants: \(t_0 = 0 < t_1 < \cdots < t_n = t_{\text{end}}\)
- Compute a sequence of values: \(y_0, y_1, \ldots, y_n\) such that

\[
\forall \ell \in [0, n], \quad y_\ell \approx y(t_\ell; y_0).
\]

- s.t. \(y_{\ell+1} \approx y(t_\ell + h; y_\ell)\) with an error \(O(h^{p+1})\) where
  - \(h\) is the integration step-size
  - \(p\) is the order of the method
New problems to solve

System of Delay Differential Equations (DDE)

\[
\dot{y}(t) = f(t, y(t), y(t - \tau)) \quad t_0 \leq t \leq t_{end}
\]
\[
y(t) = \phi(t) \quad t \leq t_0
\]

System of Neutral Delay Differential Equations (NDDE)

\[
\dot{y}(t) = f(t, y(t), y(t - \tau), \dot{y}(t - \sigma)) \quad t_0 \leq t \leq t_{end}
\]
\[
y(t) = \phi(t) \quad t \leq t_0
\]

Remark

For \( t \geq t_0 \) it can be that \( t - \tau < t_0 \) so an \textit{initial function} \( \phi(t) \) (history) is needed.

We focus on DDE in this lecture.
Various kinds of delays

Constant delay $\tau$ and $\sigma$ are non-negative values
Variable or time dependant delay $\tau(t)$ and $\sigma(t)$
State dependant delay $\tau(t, y(t))$ and $\sigma(t, y(t))$

Remark: constant and time dependant delay are well studied in the literature. State variable is still an open problem.

Remark: We will focus on constant delays
Existence and uniqueness of the solution

It is related to Lipschitz property, in particular for

\[
\dot{y}(t) = f(t, y(t), y(t - \tau(t, y(t))), \dot{y}(t - \sigma(t, y(t)))) \quad t_0 \leq t \leq t_{\text{end}} \\
y(t) = \phi(t) \quad t \leq t_0
\]

if

\[
\inf_{[t_0, t_f] \times \mathbb{R}^d} \tau(t, x) = \tau_0 > 0 \quad \text{and} \quad \inf_{[t_0, t_f] \times \mathbb{R}^d} \sigma(t, x) = \sigma_0 > 0
\]

then problem reduces to IVP-ODE, on interval \([t_0, t_0 + H]\) with \(H = \min(\tau_0, \sigma_0)\), such that

\[
\dot{y}(t) = f(t, y(t), \phi(t - \tau(t, y(t))), \dot{\phi}(t - \sigma(t, y(t)))) \leq t_{\text{end}} \\
y(t) = \phi(t_0)
\]

This is named \textit{method of steps} but it does not always work in particular when delays \(\tau\) or \(\sigma\) vanishes around \(t^*\)
DDE is not an ODE

Consider the system in 1D

\[
\begin{align*}
\dot{y}(t) &= -y(t - 1), \quad t \geq 0 \\
y(t) &= 1, \quad t \leq 0
\end{align*}
\]

As \(\dot{y}(0)^- = 0\) and \(\dot{y}(0)^+ = -y(t - 1) = -1\) the derivative function has a jump at \(t = 0\).

And the second derivative \(\ddot{y}(t)\) is given by

\[
\ddot{y} = -\dot{y}(t - 1)
\]

and so it has a jump at \(t = 1\).

The third derivative \(y'''(t)\) has a jump at \(t = 2\) ans so forth.

\(\tau(t)\) and \(\sigma(t)\) are assumed continuous

- a jump discontinuity in \(\dot{y}\) is named 1-level primary discontinuity
- a jump discontinuity in \(y'''\) is named 2-level primary discontinuity
- \(\ldots\)

Remark that the solution becomes smoother and smoother as the primary discontinuity level increases.
Figure 1: Solutions of (3.5).
Continuous Runge-Kutta methods

It is important to have a continuous solution of the ODE in order to get values of the solution at time \( t - \tau \) for instance.

As previously seen (cf lecture on discontinuous simulation), we can build a polynomial approximation using \( y_n, f(y_n), y_{n+1} \) and \( f(y_{n+1}) \). The accuracy of the approach is interesting with only order 2 RK.

For DDE, we have to find high accurate continuous extension of the solution a.k.a. CERK (Continuous extension Runge-Kutta methods)
Recall that an explicit Runge-Kutta method is defined by:

\[ k_i = f \left( t_n + c_i h_n, y_n + h \sum_{j=1}^{i-1} a_{ij} k_j \right) \]

\[ y_{n+1} = y_n + h \sum_{i=1}^{s} b_i k_i \]  \hspace{1cm} (1)

They are built from order condition which relates the Taylor expansion of the true solution and the Taylor expansion of the numerical solution.

What we want is

\[ \eta(t_n + \theta h_n) = y_n + h \sum_{i=1}^{s} b_i(\theta) k_i \quad 0 \leq \theta \leq 1 \]

\( b_i(\theta) \) are polynomials of a suitable degree such that

\[ b_i(0) = 0 \quad \text{and} \quad b_i(1) = b_i \quad i = 1, \ldots, s \]

Interpolants of the first class are defined by only using intermediate steps \( k_i \) used to define the RK method.

Interpolants of the second class are defined by adding extra stages.
Interpolants of the first class

Theorem: interpolants of the first class

Every RK methods (explicit and implicit) of order $p \geq 1$ has a continuous extension $\eta(t)$ of order (and degree) $q = 1, \ldots, \lfloor p + 1 \rfloor$

Midpoint rule

\[
\begin{array}{c|c|c}
0.5 & 0.5 & 1 \\
\hline
b_1(\theta) = \theta
\end{array}
\]

Gauss’s method

\[
k_1 = f\left(t_n + \left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right)h_n, \quad y_n + h \left(\frac{1}{4}k_1 + \left(\frac{1}{4} - \frac{\sqrt{3}}{6}\right)k_2\right)\right)
\]

\[
k_2 = f\left(t_n + \left(\frac{1}{2} + \frac{\sqrt{3}}{6}\right)h_n, \quad y_n + h \left(\left(\frac{1}{4} + \frac{\sqrt{3}}{6}\right)k_1 + \frac{1}{4}k_2\right)\right)
\]

\[
y_{n+1} = y_n + h \left(\frac{1}{2}k_1 + \frac{1}{2}k_2\right)
\]

\[
b_1(\theta) = -\frac{\sqrt{3}}{2} \theta \left(\theta - 1 - \frac{\sqrt{3}}{3}\right) \text{ and}
\]

\[
b_2(\theta) = \frac{\sqrt{3}}{2} \theta \left(\theta - 1 + \frac{\sqrt{3}}{3}\right)
\]
Interpolants of the second class

Why adding new stages? To reach

\[
\max_{0 \leq \theta \leq 1} | y_{n+1}(t_n + \theta h_n) - \eta(t_n + \theta h_n) | = O(h_n^{p+1})
\]

We will consider CERK methods which will have the FSAL property but adding new stages to reach a given order of the numerical approximation and a given order of the continuous approximation will have some limitations.

<table>
<thead>
<tr>
<th>order</th>
<th>stages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>6</td>
<td>11</td>
</tr>
</tbody>
</table>

Example of order 3

\[
\begin{array}{c|ccc}
0 & \frac{12}{23} & \frac{368}{375} & \frac{125}{384} \\
\frac{12}{23} & \frac{4}{5} & \frac{-68}{375} & \frac{529}{1152} \\
\frac{4}{5} & \frac{-68}{375} & \frac{375}{375} \\
\frac{31}{144} & \frac{529}{1152} & \frac{125}{384} \\
\frac{1}{1} & b_1(\theta) & b_2(\theta) & b_3(\theta) & b_4(\theta) \\
\end{array}
\]

- \( b_1(\theta) = \frac{41}{72} \theta^3 - \frac{65}{48} \theta^2 + \theta \)
- \( b_2(\theta) = -\frac{529}{576} \theta^3 - \frac{529}{344} \theta^2 \)
- \( b_3(\theta) = -\frac{125}{192} \theta^3 - \frac{125}{128} \theta^2 \)
- \( b_4(\theta) = \theta^3 - \theta^2 \)