Delay Differential Equations

Tony Humphries

McGill

tony.humphries@mcgill.ca

at

2016 NZMRI Summer School
Continuation Methods in Dynamical Systems
Raglan, New Zealand

12th, 14th & 15th January 2016
Acknowledgements

Collaborators
Daniel Bernucci (McGill/GA Tech), Renato Calleja (McGill/UNAMexico), Orianna DeMasi (McGill/Berkeley), Luca Guglielmi (L’Aquila), Namdar Homayounfar (McGill/Toronto), Jayme De Luca (UFSCar Brazil), Morgan Craig (U. de Montréal), Alexey Eremin (McGill/St Petersburg Russia), Bernd Krauskopf (Auckland), Mike Mackey (McGill), Felicia Magpantay (McGill/Manitoba), Michael Snarski (McGill/Brown), Finn Upham (McGill/NYU).

Disclaimer
Any typos, errors, misleading assertions, falsehoods, bad jokes or whatever else, contained in these lectures are my fault alone.

Funding:
NZMRI, University of Auckland, Bernd Krauskopf
### Lecture 1: Delay Differential Equations
- Introduction to Delay Differential Equations
- DDE IVPs
- DDEs as Dynamical Systems
- Linearization
- Numerical solution of DDE IVPs

### Lecture 2: Continuation for Constant Delay DDEs

### Lecture 3: Continuation of DDEs with State-Dependent Delays
A Delay Differential Equation (DDE) is a differential equation where the state variable appears with delayed argument. This can manifest itself in many ways. Simplest scenario is

**Constant Delay DDE**

\[ \dot{u}(t) = f(t, u(t), u(t - \tau)), \quad u(t) \in \mathbb{R}^d \]

where delay \( \tau > 0 \) is constant.

**Example: Mackey-Glass Equation**

\[ \dot{u}(t) = -\gamma u(t) + \beta \frac{u(t - \tau)}{1 + u(t - \tau)^n}, \quad u(t) \in \mathbb{R} \]

very early model of circulating white blood cell numbers

A scalar equation with chaotic dynamics
Variable Delays

DDEs with Time-Dependent Delay

\[ \dot{u}(t) = f(t, u(t), u(t - \tau(t))), \quad u(t) \in \mathbb{R}^d \]

where delay \( \tau(t) \geq 0 \) is a given function.

Example: Pantograph Equation

\[ \dot{u}(t) = au(t) + bu(kt), \quad u(t) \in \mathbb{R} \]

where \( a, b \) and \( k \) are parameters with \( k \in (0, 1) \).

Here \( kt = t - \tau(t) \) with \( \tau(t) = (1 - k)t \).

Originates from modelling pantographs!

Relatively complete theory/numerics developed for constant and time-dependent delays.
State-Dependent Delays

DDEs with State-Dependent Delay

\[ \dot{u}(t) = f(t, u(t), u(t - \tau(t, u(t)))) \], \quad u(t) \in \mathbb{R}^d 

where delay \( \tau(t, u(t)) \geq 0 \) depends on solution.

Example: Sawtooth Equation

\[ \varepsilon \dot{u}(t) = -\gamma u(t) - \kappa u(t - a - cu(t)) \], \quad u(t) \in \mathbb{R} 

with \( \varepsilon > 0, a > 0, c > 0 \) and \( \gamma + \kappa > 0 \).

Model problem introduced by Mallet-Paret and Nussbaum, gets its name from stable period solutions seen in \( \varepsilon \to 0 \) limit.

\( \tau(t, u(t)) = a + cu(t) \) is a delay provided \( u(t) \geq -a/c \).

Fortunately if \( u(t) = -a/c \) then \( \dot{u}(t) = -(\gamma + \kappa)u(t) > 0 \).

State-Dependent Delays subject of much current research
Delay Equations we won’t solve

∃ many other types of delay equations

Neutral Equations

Equation is neutral if derivatives of delay terms appear

\[ \dot{u}(t) = f(t, u(t), u(t - \tau_1), \dot{u}(t - \tau_2)), \quad u(t) \in \mathbb{R}^d \]

These are nasty: ongoing research area

The \( \tau \) and \( \tau_i \) above are called discrete delays.

Distributed Delay Examples

Finite distributed delay: \( \dot{u}(t) = \int_{t-\tau}^{t} f(s, u(s)) \, ds \)

Infinite delay:

\[ \dot{u}(t) = f \left( \int_{-\infty}^{t} u(s) g_a^n(t - s) \, ds \right), \quad g_a^n(x) = \frac{1}{\Gamma(n)} a^n x^{n-1} e^{-ax} \]
Threshold Conditions

Delays can be defined implicitly:
\[ \int_{t-\tau}^{t} V(u(s)) ds = a \]

\( a \) is given constant, \( V(\cdot) \) a given function; \( \tau \) must be determined.

[Leibniz can help or hinder here]

Implicit delays also appear in electrodynamics:

Wheeler-Feynman Electrodynamics

proton \( p(t) \) and electron \( e(t) \) interact through light cones in space-time.

\[ \tau_{p}^{\pm} = \frac{1}{c} \| p(t) - e(t \pm \tau_{p}^{\pm}) \|, \]

Neutral equation, with advanced and retarded terms, and implicit delays that stumped Feynman.
**DDE Initial Value Problems (IVPs)**

### Constant Delay DDE IVP

\[
\dot{u}(t) = f(t, u(t), u(t - \tau)), \quad u(t) \in \mathbb{R}^d, \ t \geq t_0
\]

For unique IVP solution for \( t \geq t_0 \)

- it is *not* sufficient to specify \( u(t_0) \)
- To evaluate RHS at \( t_0 \) require \( u(t_0 - \tau) \)
- \( \forall s \in [t_0 - \tau, t_0] \) require a value of \( u(s) \) to evaluate RHS of DDE at \( t = s + \tau \in [t_0, t_0 + \tau] \).

For uniqueness of IVP solution need an initial function

\[
u(t) = \varphi(t), \quad \forall t \in [t_0 - \tau, t_0]
\]

Provided \( \varphi \) is Lipschitz and \( f = f(t, u, v) \) is Lipschitz in its arguments this is sufficient for local existence and uniqueness even in state-dependent case.
**Lecture 1: Delay Differential Equations**

### Breaking Points and Smoothing

#### Breaking Point at $t_0$

Usually

$$\dot{\varphi}(t_0) \neq f(t_0, \varphi(t_0), \varphi(t_0 - \tau))$$

so $\dot{u}(t^-_0) \neq \dot{u}(t^+_0)$. This is a **breaking point**.

#### Breaking Points at $t_0 + k\tau$

$$\ddot{u}(t) = f_t(t, u(t), u(t - \tau)) + \dot{u}(t) f_u(t, u(t), u(t - \tau))$$

$$+ \dot{u}(t - \tau) f_v(t, u(t), u(t - \tau)).$$

So $\ddot{u}$ generically discontinuous at $t_0 + \tau$ and similarly, $u^{(k+1)}(t)$ discontinuous at $t = t_0 + k\tau$ for $k \geq 0$.

- **Smoothing:** $u(t) \in C^{k+1}$ for $t \geq t_0 + k\tau$
- No such smoothing for neutral problems
DDEs as Dynamical Systems

Phase space of DS is set of (initial) states of system:

\[ \{ u_t : u_t(\theta) = u(t + \theta), \ \theta \in [-\tau, 0] \} \]

But for \( t \in (t_0, t_0 + \tau) \) \( \exists \ \theta \in (-\tau, 0) \) s.t. \( t + \theta = t_0 \).
\( u_t(\theta) \) is not differentiable at this \( \theta \).

Phase Space of continuous functions

\[ \{ \varphi : \varphi \in C([-\tau, 0], \mathbb{R}^d) \} \]

Phase space is infinite dimensional even for scalar \( d = 1 \) problems

Retarded Functional Differential Equations

\[ \dot{u}(t) = F(t, u_t), \quad F : \mathbb{R} \times C \to \mathbb{R}^d \]

- Lack of differentiability is a serious hindrance to theory
Scalar Example

Suppose \( f(u, v) \) satisfies \( f(0, 0) = 0 \) so \( u = 0 \) is a steady state then

\[
\dot{u}(t) = f(u(t), u(t - \tau)) = f_u(0, 0)u(t) + f_v(0, 0)u(t - \tau) + h.o.t
\]

and linearization is

\[
\dot{u}(t) = f_u(0, 0)u(t) + f_v(0, 0)u(t - \tau) = \mu u(t) + \sigma u(t - \tau)
\]

Positing \( u(t) = e^{\lambda t} \) gives transcendental characteristic equation

\[
\lambda - \mu - \sigma e^{-\tau\lambda} = 0.
\]

Let \( \lambda = x + iy \) and take real and imaginary parts:

\[
x - \mu - \sigma e^{-\tau x} \cos(y\tau) = y + \sigma e^{-\tau x} \sin(y\tau) = 0
\]

Infinitely many roots, all lie on curve

\[
y = \pm \sqrt{\sigma^2 e^{-2\tau x} - (x - \mu)^2}
\]

- Laplace transforms show all solutions are exponentials
- Finitely many roots to right of any vertical line in \( \mathbb{C} \);
- All characteristic roots satisfy \( x < |\mu| + |\sigma| \)
- Stable manifolds is infinite dimensional
Linerization for DDEs in $\mathbb{R}^d$

\[
\dot{u}(t) = f(u(t), u(t - \tau_1), \ldots, u(t - \tau_m))
\]

Let $f(u, v_1, \ldots, v_m): \mathbb{R}^d \times \mathbb{R}^{md} \to \mathbb{R}^d$ satisfy $f(0, 0, \ldots, 0) = 0$, so $u = 0$ is a steady state.

Linearization is variational equation

\[
\dot{u}(t) = A_0 u(t) + \sum_{j=1}^m A_j u(t - \tau_j),
\]

where $A_0 = f_u$ and $A_j = f_{v_j}$ are $d \times d$ matrices evaluated at the steady-state (essentially a Jacobian matrix for each ’delay’).

There is nontrivial solution $u(t) = e^{\lambda t}v \in \mathbb{R}^d$ with $\Delta(\lambda)v = 0$ if

\[
0 = \det(\Delta(\lambda)), \quad \Delta(\lambda) = \lambda I_d - A_0 - \sum_{j=1}^m A_j e^{-\lambda \tau_j}.
\]

- Characteristic equation has infinitely many roots
- Variational equation soln: $u(t) = \sum_i \alpha_i e^{\lambda_i t} v_i$
- Finitely many $\lambda_i$ with $Re(\lambda_i) > \beta$ for any $\beta \in \mathbb{R}$.
- State-dependent DDEs are linearized by freezing the delays
Simple Numerical Methods

Method of steps

For $t \in [t_0, t_0 + \tau]$:
\[ \dot{u}(t) = f(t, u(t), u(t - \tau)) = f(t, u(t), \varphi(t - \tau)) \]

Solve as nonautonomous ODE for $t \in [t_0, t_0 + \tau]$. Repeat.

- Tedious if there is a small delay, and fails if a delay vanishes
  (recall pantograph equation)

Sub-multiple step-sizes

Most methods for ODEs generate sequence $u_n$ where $u_n \approx u(t_n)$ and $t_n = t_0 + n\Delta t$ or with variable step-size $t_{n+1} = t_n + \Delta t_{n+1}$.

For constant delay DDE with single delay $\tau$ could try favourite numerical method with constant step-size $\Delta t = \tau/m$ for some integer $m$. Then if $u_n \approx u(t_n)$ we have $u_{n-m} \approx u(t_n - \tau)$

- Fails if delays are variable or state-dependent, if variable step-size is desired for accuracy, if there are multiple non-commensurate constant delays.
Continuous Runge-Kutta Methods

RK Methods for ODEs
Let \( u_n \approx u(t_n) \) where \( \dot{u}(t) = f(t, u(t)), \quad u(t_0) = u_0,\)

Standard RK method defines \( s \) intermediate stages per step

\[
Y_i = u_n + h \sum_{j=1}^{s} a_{ij} K_j, \quad K_i = f(t_n + c_i h, Y_i), \quad i = 1, \ldots, s
\]

\[
u_{n+1} = u_n + h \sum_{i=1}^{s} b_i K_i,
\]
(think of \( Y_i \approx u(t_n + c_i h) \), typically \( c_i \in [0, 1] \))

Continuous RK Methods
Define functions \( b_i(\theta) \) & let

\[
\eta(t_n + \theta h) = u_n + h \sum_{i=1}^{s} b_i(\theta) K_i, \quad \theta \in [0, 1]
\]
defines continuous extension of solution for \( t \in [t_n, t_{n+1}] \).
Proceed step by step.

- If \( a_{ij} = 0 \) for \( j \geq i \) method is explicit
- For (implicit) collocation RK methods \( b_i(\theta) \) defined naturally
Continuous Runge-Kutta Methods (CRKs) for DDEs

\[ \dot{u}(t) = f(t, u(t), u(t - \tau)), \quad u(t) = \varphi(t), \ t \in [t_0 - \tau, t_0]. \]

Continuous Runge-Kutta (CRK) method is defined by

\[
Y_i^{(n)} = u_n + h \sum_{j=1}^{s} a_{ij} f(t_n + c_j h, Y_j^{(n)}, \tilde{Y}_j^{(n)}), \quad \tilde{Y}_j^{(n)} = \eta(t_n + c_j h - \tau),
\]

\[ u_{n+1} = u_n + h \sum_{j=1}^{s} b_j(1) f(t_n + c_j h, Y_j^{(n)}, \tilde{Y}_j^{(n)}) \]

\( \tilde{Y}_j^{(n)} \) called spurious stages, \( \eta(t) \) is the continuous extension of numerical solution:

\[ \eta(t_m + \theta h) = u_m + h \sum_{j=1}^{s} b_j(\theta) f(t_m + c_j h, Y_j^{(m)}, \tilde{Y}_j^{(m)}), \quad m \leq n, \ \theta \in [0, 1] \]

- defines numerical solution as a continuous function for \( t \in [t_0, t_0 + T] \).
- if \( a_{ij} = 0 \) for \( j \geq i \) & \( h \leq \tau \) method is explicit
- \( b_i(\theta) \) polynomial (of degree equal to order of method)
- \( h > \tau \) called overlapping. Generalised methods can be explicit
- Spurious stages ruin super-convergence of quadrature methods