

Convergence of collocation methods for delay differential equations

Jan Sieber

University of Exeter (UK)

joint work with

Alessia Andò (University of Udine, Italy)

Outline

- ▶ DDE-Biftool approach to distributed delays and renewal equations
- ▶ Convergence analysis problems for DDEs
- ▶ Convergence of discretization & Newton iteration

Distributed delays

Linear DDEs: Representation Theorem ensures r.h.s. has form

$$x'(t) = \sum A_j x(t - \tau_j) + \int_0^{\tau_{\max}} G(s)x(t-s)ds$$

Nonlinear DDEs:

$$x'(t) = f\left(x(t - \tau_j), \int_{s_1}^{s_2} g(s, x(t-s), p)ds, \dots\right) \quad ?$$

No interface for general nonlinear functional of $x_t = x(t + (\cdot))$

DDE-Biftool

uses formulation as differential-algebraic problem:

$$Mx'(t) = f(x(t - \tau_m), \dots, p)$$

$$0 = \int_0^{\tau_d} g_d(s, x_\ell(t-s), p_i) ds - x_k(t)$$

...

DDE-Biftool

uses formulation as differential-algebraic problem:

$$Mx'(t) = f(x(t - \tau_m), \dots, p)$$

$$0 = \int_0^{\tau_d} g_d(s, x_\ell(t-s), p_i) ds - x_k(t)$$

...

nonsquare,
can be singular

DDE-Biftool

uses formulation as differential-algebraic problem:

$$Mx'(t) = f(x(t - \tau_m), \dots, p)$$

$$0 = \int_0^{\tau_d} g_d(s, x_\ell(t-s), p_i) ds - x_k(t)$$

...

nonsquare,
can be singular

index provided by user

DDE-Biftool

uses formulation as differential-algebraic problem:

$$Mx'(t) = f(x(t - \tau_m), \dots, p)$$

$$0 = \int_0^{\tau_d} g_d(s, x_\ell(t-s), p_i) ds - x_k(t)$$

nonsquare,
can be singular

...

state or parameter (index provided by user)

function provided by user

DDE-Biftool

uses formulation as differential-algebraic problem:

$$Mx'(t) = f(x(t - \tau_m), \dots, p)$$

distributed delay, can be used with delay

$$0 = \int_0^{\tau_d} g_d(s, x_\ell(t-s), p_i) ds - x_k(t)$$

nonsquare, can be singular

...

function provided by user

state or parameter (index provided by user)

index provided by user

- ▶ permits multiple nested integrals as ℓ and k can overlap,
- ▶ x_k can be multidimensional,
- ▶ several distributed delays possible
- ▶ approximated by N discrete delays $\tau_j = s_j \tau_d$

$$\int \dots \approx \sum_{j=1}^N w_j \tau_d g(s_j \tau_d, x_\ell(t - s_j \tau_d), p_i)$$

DDE-Biftool example renewal equation (RE)

Breda *et al.* 2016

$$x(t) = \frac{\gamma}{2} \int_{\tau_2}^{\tau_1 + \tau_2} x(s)(1 - x(s)) ds$$

implemented as

$$0 = x(t) - \frac{\gamma}{2} y(t - \tau_2)$$

$$0 = \int_0^{\tau_1} x(s)(1 - x(s)) ds - y(t)$$

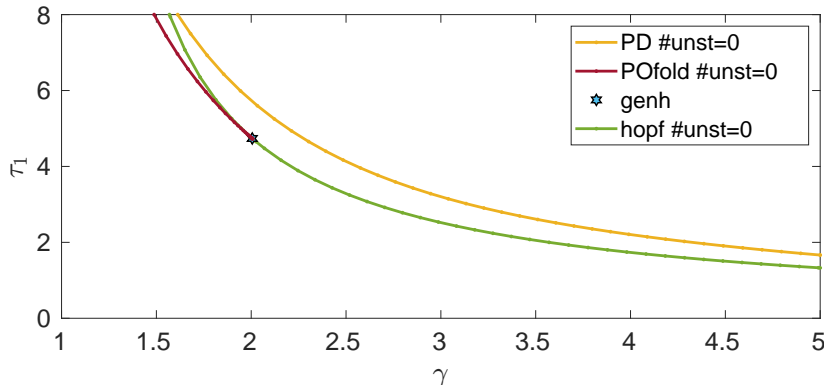
DDE-Biftool example renewal equation (RE)

Breda *et al.* 2016

$$0 = x(t) - \frac{\gamma}{2} y(t - \tau_2)$$

$$0 = \int_0^{\tau_1} x(s)(1 - x(s))ds - y(t)$$

$$p = (\gamma, \tau_1, \tau_2), \quad u = (x, y)$$



Complex example: size structured *Daphnia* population model

Diekmann *et al.* 2010, Andò 2020

resource: $\dot{r}(t) = f_0(r(t)) - f_c(r(t))p_{\text{eff}}(a_{\text{max}}, t),$

maturation threshold: $0 = f_{\text{thr}}(a_m(t), s(a_m(t), t)) - S_m,$

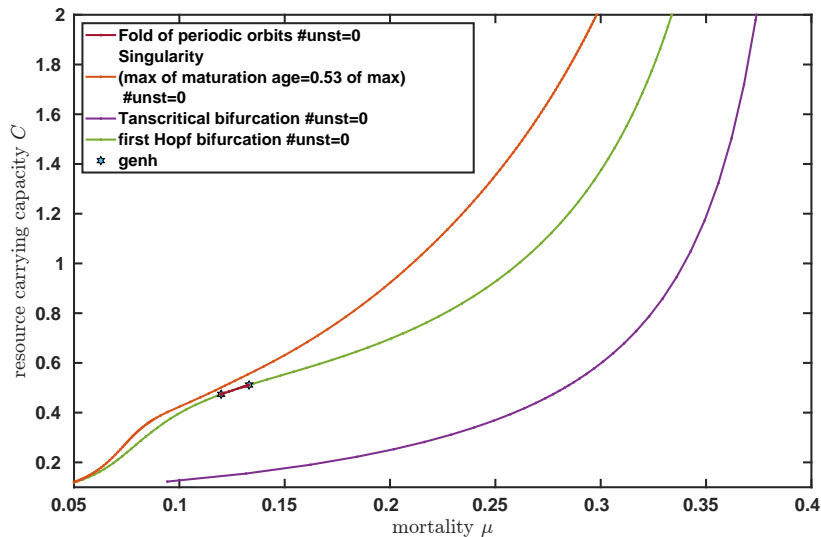
birth rate: $b(t) = p_{\text{eff}}(a_{\text{max}}, t) - p_{\text{eff}}(a_m(t), t),$

cohort size: $s(a, t) = g_0(a) + \int_0^a g_r(\alpha, f_r(r(t - \alpha)))d\alpha,$

effective population: $p_{\text{eff}}(a, t) = \int_0^a h(\alpha, s(\alpha, t), r(t))b(t - \alpha)d\alpha.$

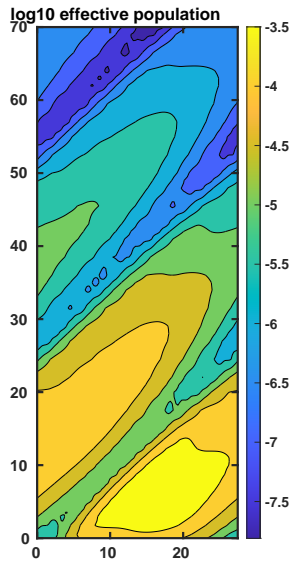
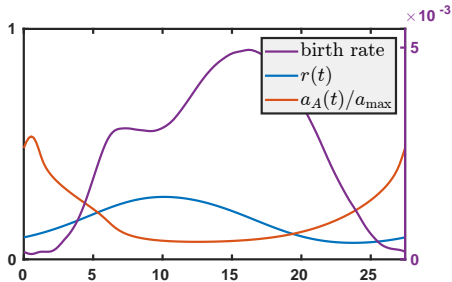
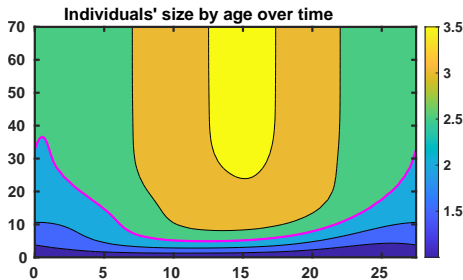
Complex example: size structured *Daphnia* population model

Diekmann *et al.* 2010, Andò 2020



Size structured *Daphnia* population model: shock

Diekmann *et al.* 2010, Andò 2020



DDE-Biftool distributed delays conclusion

- ▶ more complex, flexible mechanism for nested integrals
⇒ daphnia demo
- ▶ uses interface for state-dependent delays to avoid introducing N coupled parameters, $\tau(j, x, p) = s_j \tau_d$,
- ▶ discretized renewal equations \sim neutral equations:

$$x(t) = \sum_{j=1}^N w_j \tau_d g(s_j \tau_d, x(t - s_j \tau_d), p)$$

⇒ essential spectral radius of time-1 map > 1

⇒ high-frequency instability

⇒ ignore high-frequency eigenvalues of equilibria

⇒ ignore Floquet multipliers with highly oscillatory eigenfunctions.

- ▶ Renewal equations can be converted to equivalent DDEs
- ▶ vectorized g mandatory

Convergence of numerical discretization

DDE-Biftool:

$$\dot{x}(t) = f(x(t - \tau_m), p)$$

time rescaling $\Rightarrow x'(t_k) = Tf(x(t_k - \tau_m/T), p)$ at $L \times n_{\text{deg}}$ times t_k

+continuity & periodicity for piecewise continuous polynomial x with L pieces, degree n_{deg} .

Convergence proof for constant delay:

Engelborghs & Doedel'02: stability for linear DDEs

thought this implies convergence, but

$$F : (x, T) \mapsto Tf(x((\cdot) - \tau_m/T))$$

is not continuously differentiable w.r.t. unknown period T

(term $x'((\cdot) - \tau_m/T)\tau_m/T$ shows up) (solved by Andò 2020)

$F : C^k \rightarrow C^\ell$ is only $C^{k-\ell}$ if $k \geq \ell$

DDEs with state-dependent delays

$$F(x)(t) = f(x_t) \quad x_t(s) := x(t+s), \quad f: C \rightarrow \mathbb{R}^n \text{ functional}$$

is cont. diff. only if delays constant.

$$F(x)(t) = x(t+x(t)) \Rightarrow [\partial F(x)y](t) = y(t+x(t)) + x'(t+x(t))y(t)$$

Instead: **mild differentiability** concept (Hartung *et al.*'06)

$$[\partial^k F(x)(y)^k] \text{ depends on } x, x', \dots, x^{(k)}, y', \dots, y^{(k-1)},$$

(continuously), but **not** $y^{(k)}$.

Result: ($0 = \Phi(x^*)$, $0 = \Phi_L(x_L)$)

- ▶ $\|x_L - x^*\|_{0,1} \sim L^{-n_{\text{deg}}}$ if F is $\geq n_{\text{deg}}$ times mild. diff. & $\partial\Phi(x^*)$ is invertible
- ▶ Newton iteration convergence limited by $\|x_L - x^*\|_{0,1}$,
 \Rightarrow better convergence for higher-accuracy solutions

DDEs with state-dependent delays

Result: ($0 = \Phi(x^*)$, $0 = \Phi_L(x_L)$)

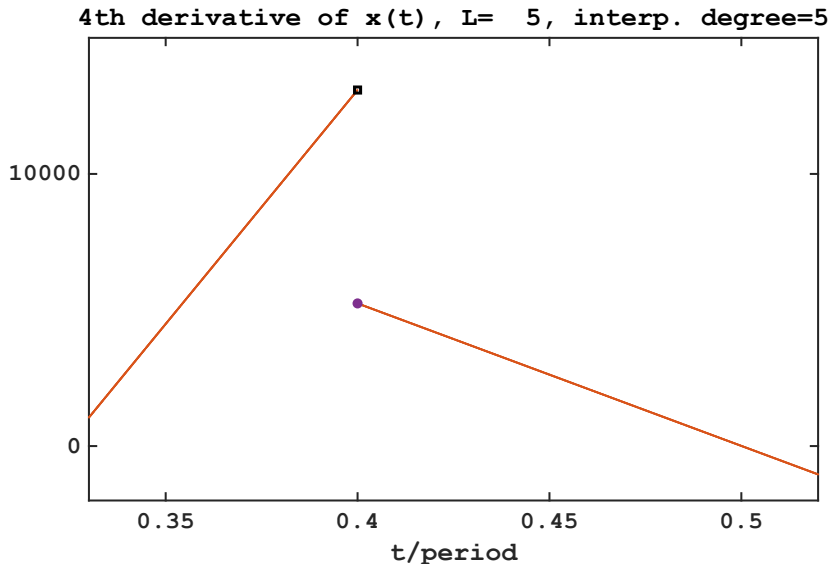
- ▶ $\|x_L - x^*\|_{0,1} \sim L^{-n_{\text{deg}}}$ if F is $\geq n_{\text{deg}}$ times mild. diff. & $\partial\Phi(x^*)$ is invertible
- ▶ Newton iteration convergence limited by $\|x_L - x^*\|_{0,1}$,
⇒ better convergence for higher-accuracy solutions

Issues:

- !! Φ_L only cont. diff. if x_L is cont. diff., but x'_L discontinuous
⇒ Jacobian $\partial\Phi_L(\cdot)$ is discontinuous on solution space,
violates standard assumptions for convergence of
discretization and Newton iteration

DDEs with state-dependent delays — issues

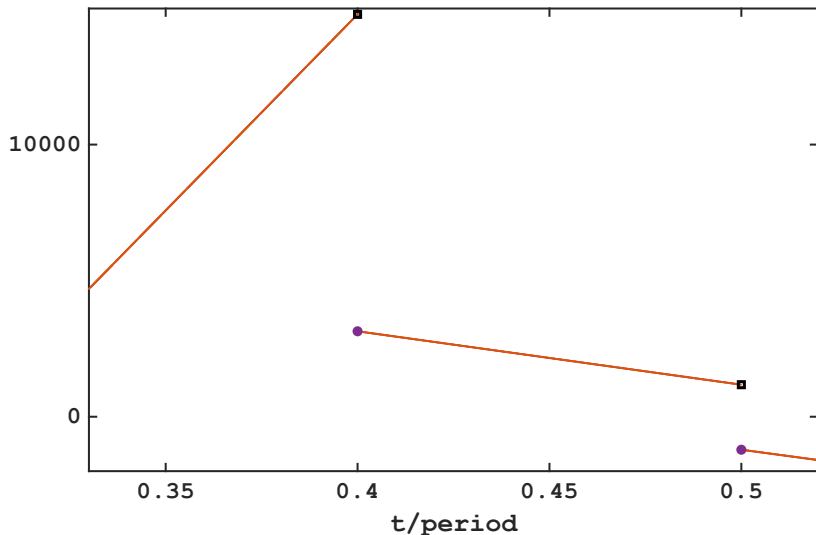
All derivatives discontinuous, but converge to true solution:



DDEs with state-dependent delays — issues

All derivatives discontinuous, but converge to true solution:

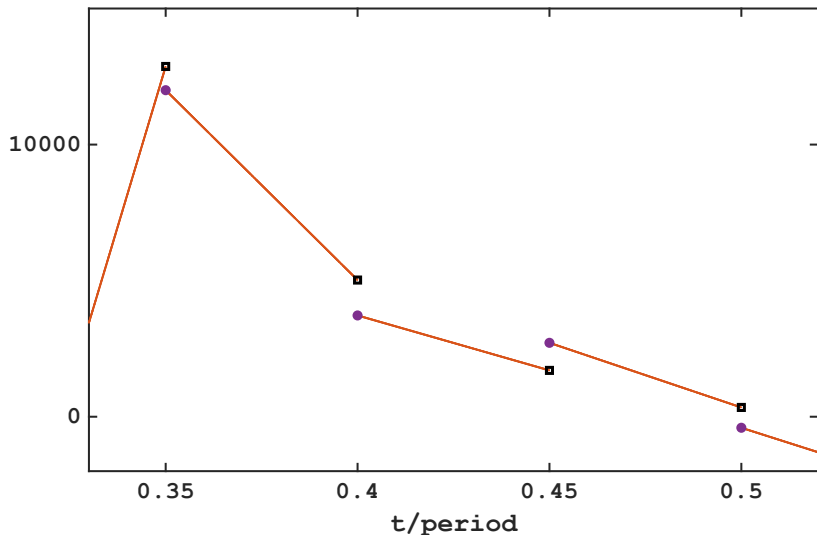
4th derivative of $x(t)$, $L=10$, interp. degree=5



DDEs with state-dependent delays — issues

All derivatives discontinuous, but converge to true solution:

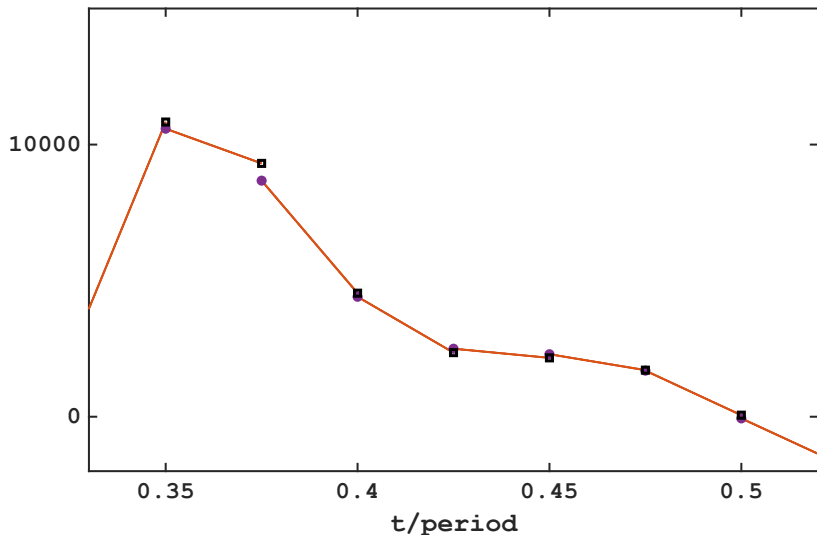
4th derivative of $x(t)$, $L=20$, interp. degree=5



DDEs with state-dependent delays — issues

All derivatives discontinuous, but converge to true solution:

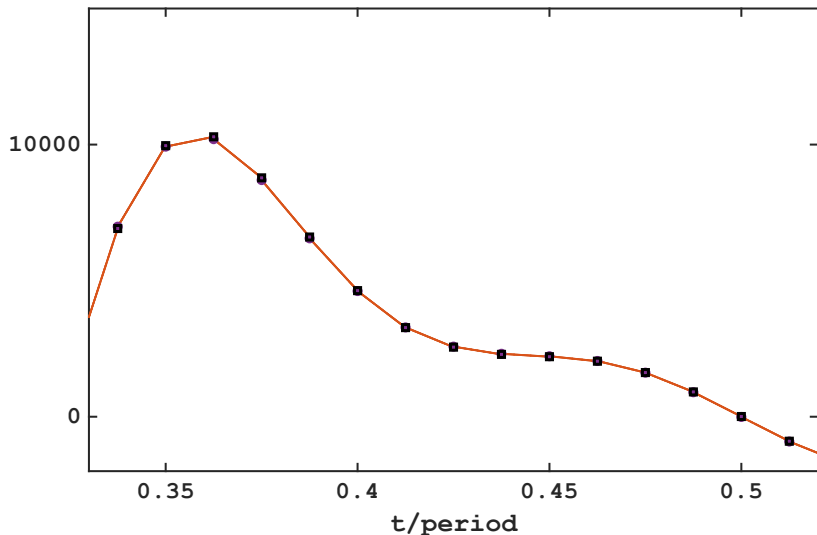
4th derivative of $x(t)$, $L=40$, interp. degree=5



DDEs with state-dependent delays — issues

All derivatives discontinuous, but converge to true solution:

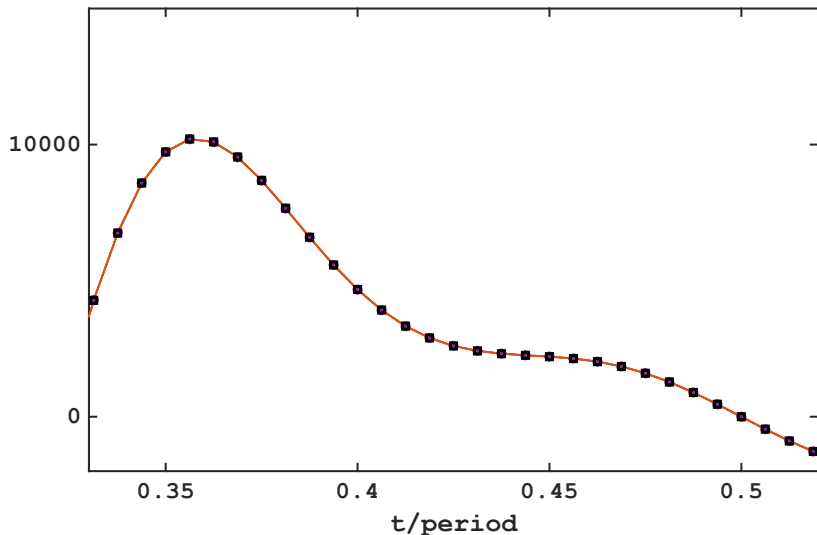
4th derivative of $x(t)$, $L=80$, interp. degree=5



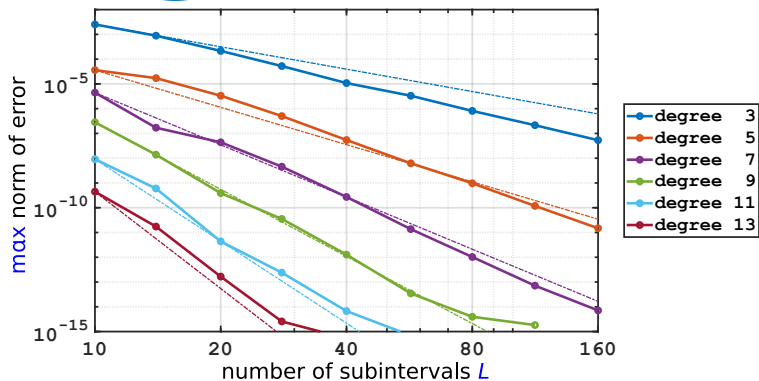
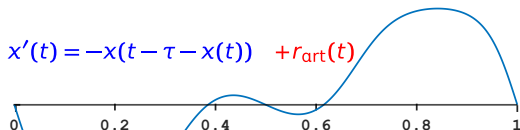
DDEs with state-dependent delays — issues

All derivatives discontinuous, but converge to true solution:

4th derivative of $x(t)$, $L=160$, interp. degree=5



DDEs with state-dependent delays — example error plot



Conclusion

sourceforge.net/p/ddebiftool/git/ci/master/tree/

- ▶ bifurcation analysis for DDEs with distributed delays and Renewal Equations feasible
- ▶ linear stability analysis for REs suffers instabilities
- ▶ expectation management for speed
- ▶ convergence proof of numerical method surprisingly recent for constant delays (Andò'20), current preprint for state-dependent delays
- ▶ difficulty: lack of continuous differentiability of r.h.s.