Numerical optimal control for distributed delay differential equations: A simultaneous approach based on linearization of the delayed variables

Tobias K. S. Ritschel

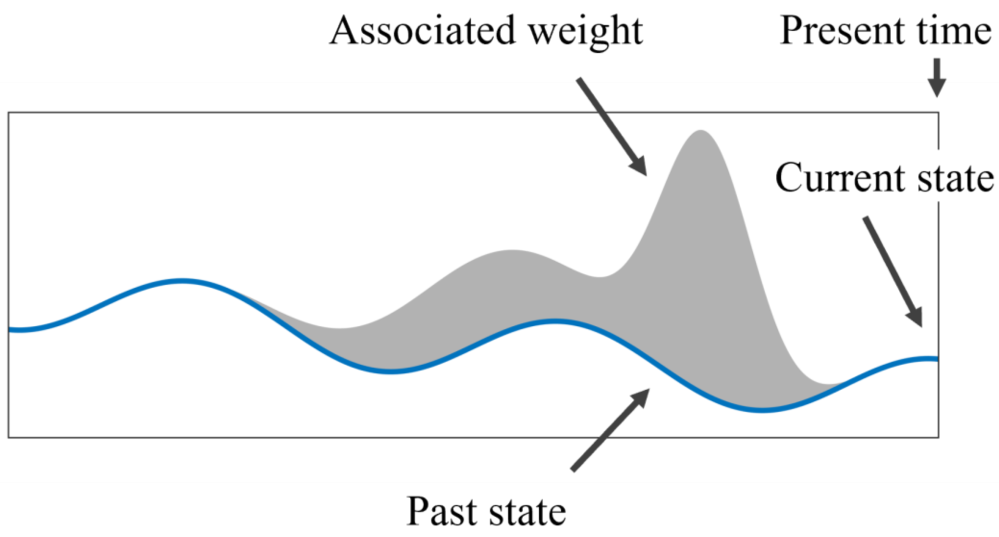
Department of Applied Mathematics and Computer Science,
Technical University of Denmark

16.30 – 16.50, WeC1.1, June 25, 2025
European Control Conference, Thessaloniki, Greece

Research question

Delay differential equation with <u>distributed</u> time delay

$$\dot{x}(t) = f\left(x(t), \int_{-\infty}^{t} \alpha(t-s)x(s)\mathrm{d}s\right)$$



Linearize delayed state

$$x(s) \approx x(t) + \dot{x}(t)(s-t)$$

<u>Question</u>: Can we develop numerical optimal control methods based on this approximation?

Optimal control problem

$$\min_{\{u_k\}_{k=0}^{N-1}} \quad \phi = \phi_x + \phi_{\Delta u}$$

$$\phi_x = \int_{t_0}^{t_f} \Phi(x(t), u(t), d(t), p)\,\mathrm{d}t,$$

Tracking/economic objective

$$\phi_{\Delta u} = \frac{1}{2}\sum_{k=0}^{N-1} \frac{\Delta u_k^T W_k \Delta u_k}{\Delta t}$$

Rate-of-movement penalization

subject to

$$x(t) = x_0(t), \quad t \in (-\infty, t_0],$$

Initial state function

$$\dot{x}(t) = f(x(t), z(t), u(t), d(t), p),$$

Delay differential equations

$$z(t) = \begin{bmatrix} (\alpha_1 * r_1)(t) \\ \vdots \\ (\alpha_m * r_m)(t) \end{bmatrix},$$

Memory states – given by the convolution

$$(\alpha_i * r_i)(t) = \int_{-\infty}^{t} \alpha_i(t-s, u(t)) r_i(s)\,\mathrm{d}s$$

$$r_i(t) = h_i(x(t), p),$$

Delayed variables

$$u(t) = u_k, \quad t \in [t_k, t_{k+1}[,$$

$$d(t) = d_k, \quad t \in [t_k, t_{k+1}[,$$

Zero-order-hold (ZOH) parametrization

$$x_{\min} \leq x(t) \leq x_{\max},$$

State bounds

$$u_{\min} \leq u_k \leq u_{\max},$$

Input bounds

Linearize delayed variables

Linearize delayed state

$$r_i(s) \approx r_i(t) + \dot{r}_i(t)(s-t)$$

Convolution

$$(\alpha_i * r_i)(t) \approx r_i(t) - \dot{r}_i(t) \int_0^\infty \alpha_i(\tau, u(t))\tau \,\mathrm{d}\tau.$$

Kernel "mean"

$$\gamma_i(u(t)) = \int_0^\infty \alpha_i(\tau, u(t))\tau \,\mathrm{d}\tau$$

Discretization w. Euler's implicit method

Approximate system

$$\dot{x}(t) = f(x(t), z(t), u(t), d(t), p),$$

$$z(t) = \left.\begin{bmatrix} r_1(t) - \dot{r}_1(t)\gamma_1(u(t)) \\ \vdots \\ r_m(t) - \dot{r}_m(t)\gamma_m(u(t)) \end{bmatrix}\right\} \textbf{Substitute linearized variables}$$

$$r_i(t) = h_i(x(t), p), \quad i = 1, \ldots, m$$

Euler's implicit method

$$x_{k,n+1} - x_{k,n} - f(x_{k,n+1}, z_{k,n+1}, u_k, d_k, p)\Delta t_{k,n} = 0$$

Approximate linearized delayed variable

$$v_{i,k,n+1} = r_{i,k,n+1} - \frac{r_{i,k,n+1} - r_{i,k,n}}{\Delta t_{k,n}}\gamma_i(u_k)$$

Simultaneous approach

Discretize the objective function (right rectangle rule)

$$\psi_x = \sum_{k=0}^{N-1} \sum_{n=0}^{M-1} \Phi(x_{k,n+1}, u_k, d_k, p) \Delta t_{k,n}$$

Transcribed optimal control problem

$$\min_{\{\{x_{k,n+1}\}_{n=0}^{M-1}, u_k\}_{k=0}^{N-1}} \quad \psi = \psi_x + \phi_{\Delta u},$$

subject to

$$\begin{aligned} &R_{k,n}(x_{k,n+1}, x_{k,n}, u_k, d_k, p) = 0, \\ &x_{\min} \leq x_{k,n+1} \leq x_{\max}, \\ &u_{\min} \leq u_k \leq u_{\max}, \end{aligned}$$

Distributed time delays from nonuniform velocity profiles

Hagen-Poiseuille flow

$$v(r) = a(R^2 - r^2), \quad a = \frac{\Delta P}{4\mu L}$$

Kernel (tau >= tau0)

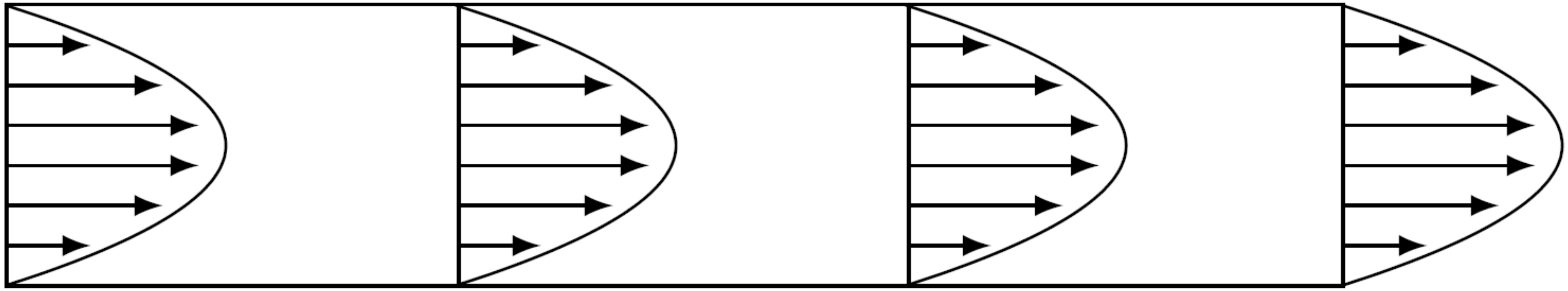
$$\alpha(\tau) = 2\frac{\tau_0^2}{\tau^3}$$

Minimum travel time

$$\tau_0 = \frac{L}{aR^2},$$

Kernel “mean”

$$\gamma = 2\tau_0$$



Footnote: **We also derive a formula for general monotonic velocity profiles*

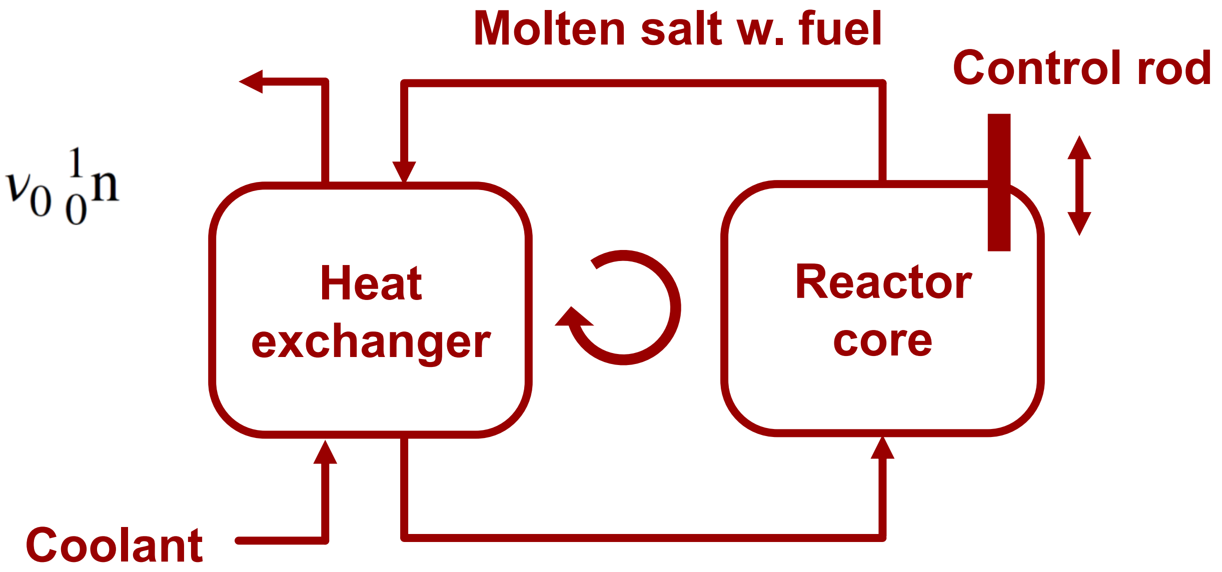
Molten salt nuclear reactor

Fission reaction

$$ {}^{1}_{0}\mathrm{n} + {}^{235}_{92}\mathrm{U} \xrightarrow{\text{neutron absorp.}} {}^{236}_{92}\mathrm{U} \xrightarrow{\text{fission}} \sum_{i=1}^{N_p} \nu_i \, {}^{a_i}_{z_i}\mathrm{E}_i + \nu_0 \, {}^{1}_{0}\mathrm{n} $$

Delayed release of a neutron

$$ {}^{a_i}_{z_i}\mathrm{E}_i \xrightarrow{\text{neg. beta decay}} {}^{a_i}_{z_i+1}\mathrm{E}^*_i \xrightarrow{\text{neutron emission}} {}^{a_i-1}_{z_i+1}\mathrm{E}^*_i + {}^{1}_{0}\mathrm{n} $$



Molten salt nuclear reactor model

Mass balance equations

$$\dot{C}_i(t) = (C_{i,in}(t) - C_i(t))D(t) + R_i(t), \quad i = 1, \ldots, N_g$$

$$\dot{C}_n(t) = R_n(t)$$

$$\dot{\rho}_{th}(t) = -\kappa \dot{T}_r(t)$$

Delayed variable (Hagen-Poiseuille for a length of L)

$$C_{i,in}(t) = e^{-\lambda_i \gamma} \int_{-\infty}^{t} \alpha_f(t-s) C_i(s) \, \mathrm{d}s$$

Energy balances

$$\dot{T}_r(t) = \frac{f_r(t)}{m_r}(T_{r,in}(t) - T_r(t)) + \frac{Q_g(t)}{m_r c_P}$$

$$\dot{T}_{hx}(t) = \frac{f_{hx}(t)}{m_{hx}}(T_{hx,in}(t) - T_{hx}(t)) - \frac{k_{hx}}{m_{hx} c_P}(T_{hx}(t) - T_c)$$

Other variables

$$\rho(t) = \rho_{th}(t) + \boxed{\rho_{ext}(t)}$$

$$Q_g(t) = Q_{g,0} \frac{C_n(t)}{C_{n,0}}$$

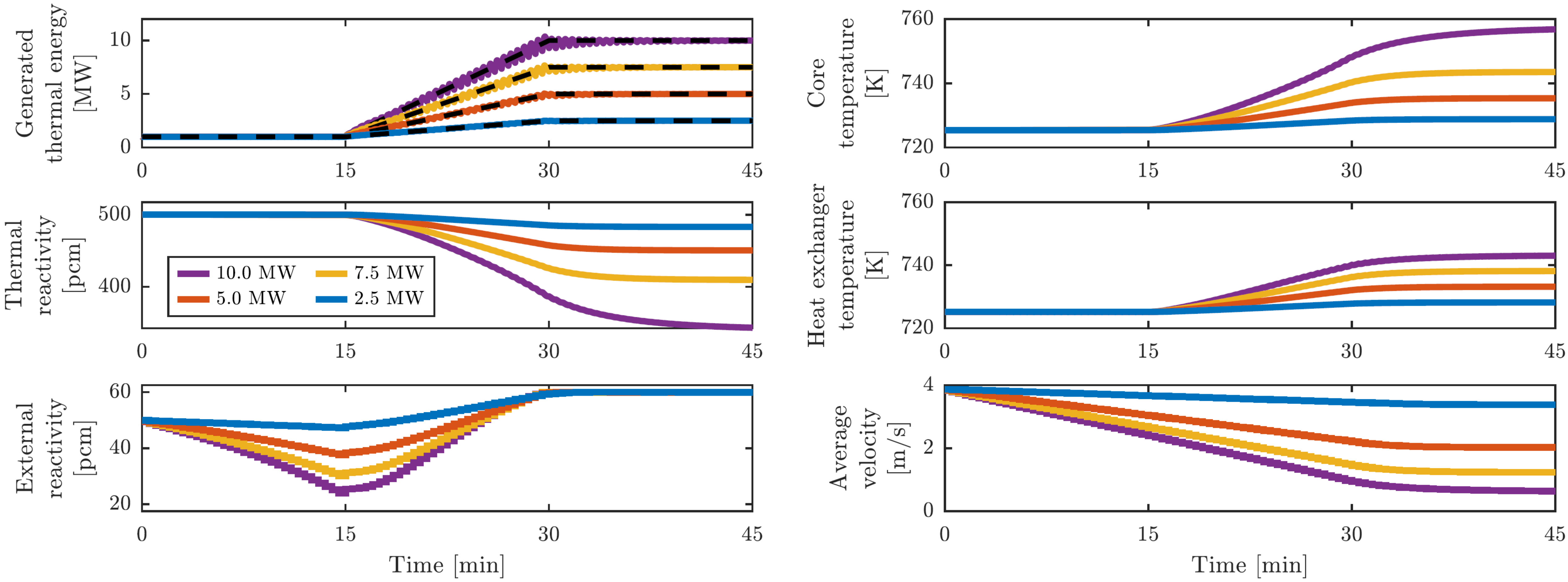
We control $\boxed{\Delta P}$ which affects α_f, α_h, and γ

Delayed variables (Hagen-Poiseuille for L/2)

$$T_{r,in}(t) = \int_{-\infty}^{t} \alpha_h(t-s) T_{hx}(s) \, \mathrm{d}s,$$

$$T_{hx,in}(t) = \int_{-\infty}^{t} \alpha_h(t-s) T_r(s) \, \mathrm{d}s,$$

Numerical example: Optimal ramping



Summary and future work

Numerical optimal control w.
input-dependent distributed delays

Linearize delayed variables

Simultaneous approach

Kernel approximations
for distributed time delays

$$\hat{\alpha}(t) = \sum_{m=0}^{M} c_m \ell_m(t)$$

Ritschel, T.K.S., Wyller, J., 2025. An Algorithm for Distributed Time Delay Identification without A Priori Knowledge of the Kernel. Automatica 178, pp. 112382. DOI: 10.1016/j.automatica.2025.112382.

Ritschel, T.K.S., 2025. On Erlang Mixture Approximations for Differential Equations with Distributed Time Delays. arXiv: [http://arxiv.org/abs/2502.12984 2502.12984].

Thank you

Appendix: Molten salt nuclear reactor model

$$D(t) = F(t)/V$$

$$F(t) = Av(t)$$

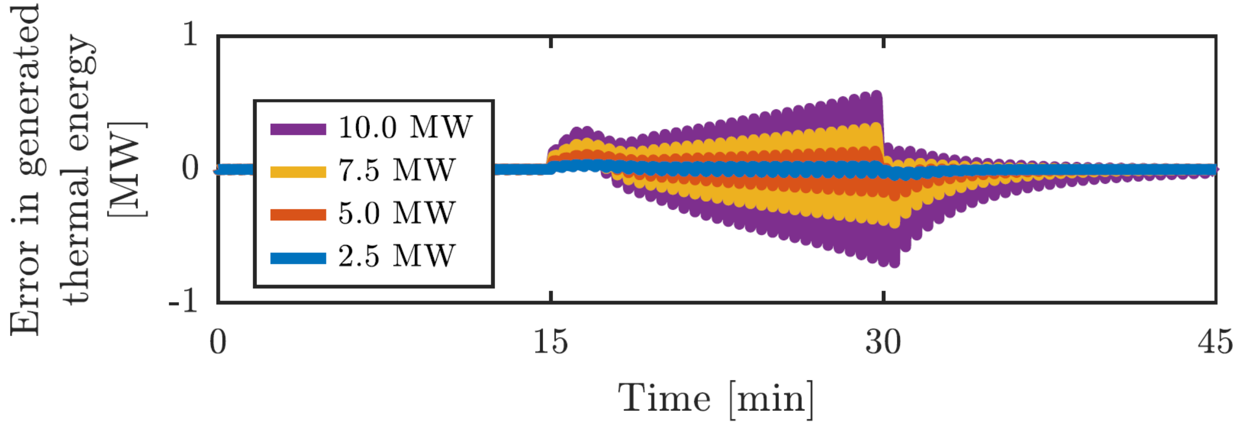
$$f_r(t) = f_{hx}(t) = \rho_s Av(t)$$

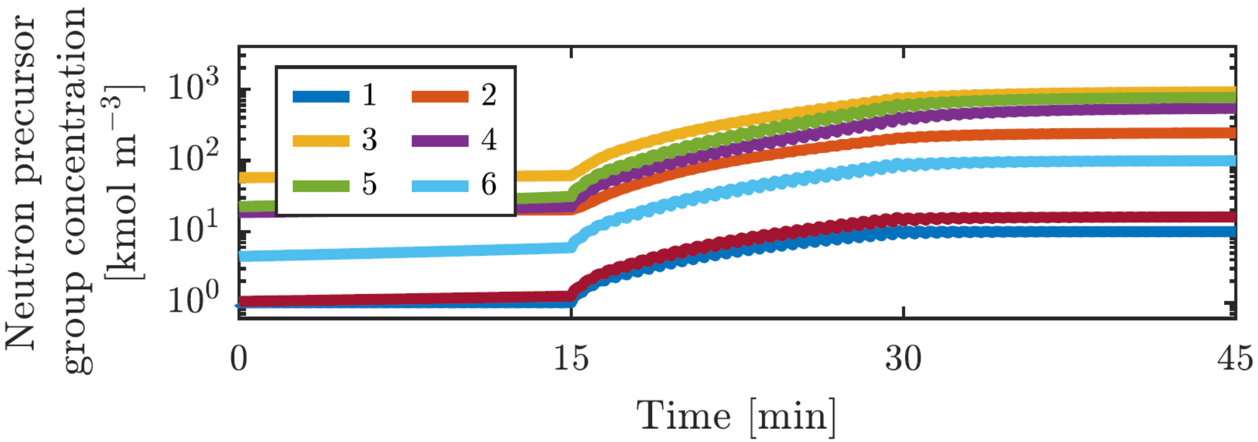
$$R(t) = S^T(t)r(t)$$

$$S(t) = \begin{bmatrix} -1 & & & 1 \\ & \ddots & & \vdots \\ & & -1 & 1 \\ \beta_1 & \cdots & \beta_{N_g} & \rho(t) - \beta \end{bmatrix}, \; r(t) = \begin{bmatrix} \lambda_1 C_1(t) \\ \vdots \\ \lambda_{N_g} C_{N_g}(t) \\ C_n(t)/\Lambda \end{bmatrix}$$

Appendix: Numerical example

Distributed time delays





Absolute time delays