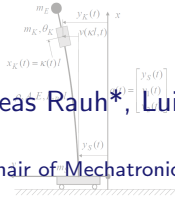


Verified Simulation of Dynamic Systems

Pre-Conference Workshop at ECC 2015
14th European Control Conference

Linz, Austria, July 14th, 2015



Andreas Rauh*, Luise Senkel*, Ekaterina Auer**

*Chair of Mechatronics, University of Rostock, Germany

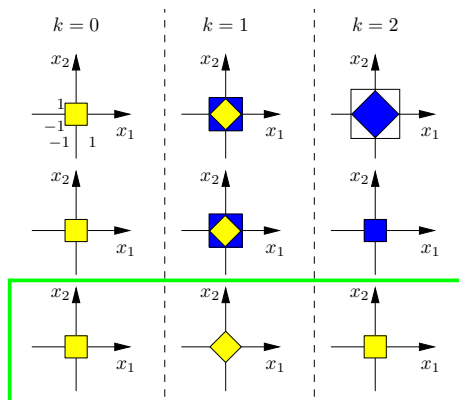
**Faculty of Engineering, University of Applied Sciences Wismar, Germany



Contents

- Discrete-time sets of difference equations
- Approaches for verified integration of ODEs in VALENCIA-IVP, alternatives to Taylor series or Taylor model-based techniques
- Fundamental iteration scheme
- Exponential enclosure technique based on real-valued interval arithmetic
- State-space transformation: Decoupling of state equations
- Use of complex-valued interval arithmetic
- Generalizations for systems with single and multiple eigenvalues and dominant (piecewise) linear dynamics
- Conclusions and outlook on future work

Wrapping Effect (1)



- Linear set of state equations

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k$$

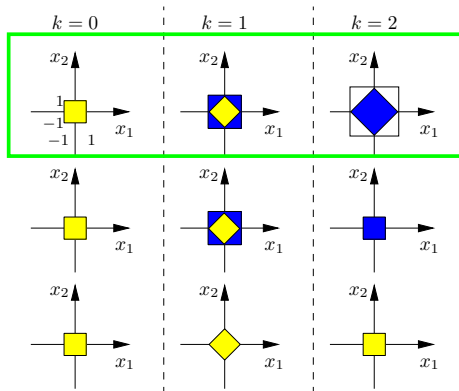
- Example:

$$[\mathbf{x}_0] = \begin{bmatrix} [-1; 1] \\ [-1; 1] \end{bmatrix}$$

$$\mathbf{A}_k = \mathbf{A} = \frac{1}{2}\sqrt{2} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$$

⇒ Rotation of 45°

Wrapping Effect (2)



- Traditional interval arithmetic evaluation:

$$\mathbf{x}_1 = \mathbf{A} \mathbf{x}_0$$

$$\mathbf{x}_2 = \mathbf{A} \mathbf{x}_1$$

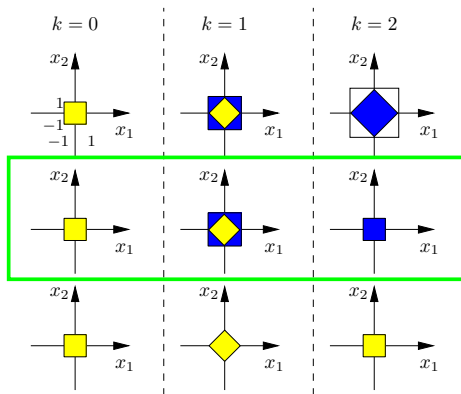
$$\vdots$$

$$\mathbf{x}_{k+1} = \mathbf{A} \mathbf{x}_k$$

⇒ Exponential growth of the enclosing interval boxes

⇒ Wrapping effect

Wrapping Effect (3)



- Modified system matrix:

$$\mathbf{x}_1 = \mathbf{A} \mathbf{x}_0 = \tilde{\mathbf{A}}_0 \mathbf{x}_0$$

$$\mathbf{x}_2 = \mathbf{A} \tilde{\mathbf{A}}_0 \mathbf{x}_0 = \tilde{\mathbf{A}}_1 \mathbf{x}_0$$

$$\vdots$$

$$\mathbf{x}_{k+1} = \mathbf{A} \tilde{\mathbf{A}}_{k-1} \mathbf{x}_k = \tilde{\mathbf{A}}_k \mathbf{x}_0$$

$$\tilde{\mathbf{A}}_k = \mathbf{A} \tilde{\mathbf{A}}_{k-1}$$

⇒ Significant reduction of the wrapping effect for linear systems

⇒ Elimination of the wrapping effect for point matrices \mathbf{A}

Discrete-Time Dynamical Systems with Time-Invariant Interval Parameters (1)

$$\begin{bmatrix} \mathbf{x}_{k+1} \\ \mathbf{p}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_k(\mathbf{x}_k, \mathbf{p}_k, \mathbf{u}_k, k) \\ \mathbf{p}_k \end{bmatrix}$$

\mathbf{u}_k : given open-/ closed-loop control

Problem

Determine state enclosures at each time step k for a given finite time horizon

Solution Approach 1

Recursive computation of state intervals

$$[\mathbf{x}_{k+1}] = \mathbf{g}_k([\mathbf{x}_k], [\mathbf{p}_k], \mathbf{u}_k, k) : \text{open-loop control}$$

$$[\mathbf{x}_{k+1}] = \mathbf{g}_k([\mathbf{x}_k], [\mathbf{p}_k], \mathbf{u}_k([\mathbf{x}_k], [\mathbf{p}_k]), k) : \text{closed-loop control}$$

Discrete-Time Dynamical Systems with Time-Invariant Interval Parameters (2)

$$\begin{bmatrix} \mathbf{x}_{k+1} \\ \mathbf{p}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_k(\mathbf{x}_k, \mathbf{p}_k, \mathbf{u}_k, k) \\ \mathbf{p}_k \end{bmatrix}$$

\mathbf{u}_k : given open-/ closed-loop control

Problem

Determine state enclosures at each time step k for a given finite time horizon

Solution Approach 2

Computation of state intervals $[\mathbf{x}_{k+1}]$ by coordinate transformations for reduction of overestimation caused by the wrapping effect, e.g. linear transformations

$$[\mathbf{x}_k] = \mathbf{T}_k \cdot [\tilde{\mathbf{x}}_k] \quad \Longrightarrow \quad [\tilde{\mathbf{x}}_{k+1}] = \mathbf{T}_{k+1}^{-1} \cdot \mathbf{g}_k(\mathbf{T}_k \cdot [\tilde{\mathbf{x}}_k], [\mathbf{p}_k], \dots)$$

Discrete-Time Dynamical Systems with Time-Invariant Parameter Uncertainties (3)

Solution Approach 3

Idea: Subdividing and merging of state intervals representing $[\mathbf{x}_{k+1}]$ for tight enclosures of complexly shaped regions

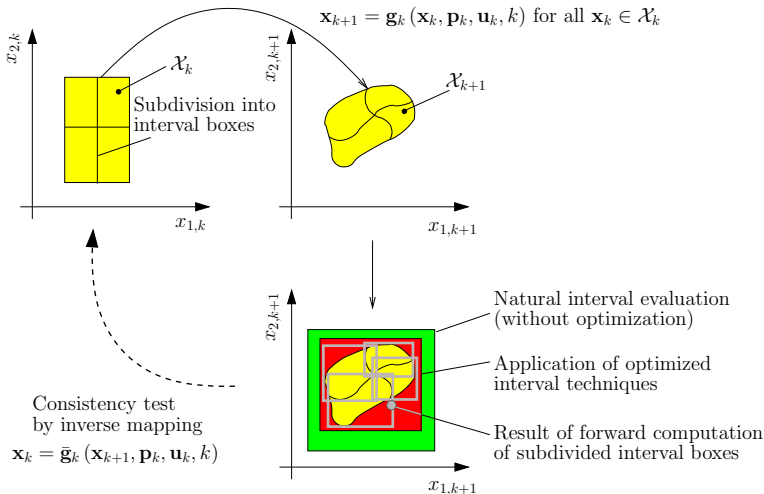
- 1 Consistency tests by inverse mapping of state equation

$$\mathbf{x}_k = \bar{\mathbf{g}}_k(\mathbf{x}_{k+1}, \mathbf{p}_k, \dots)$$

$[\mathbf{x}'_{k+1}] \subset [\mathbf{x}_{k+1}]$ subinterval of forward computation

- 2 Interval Newton methods for state equations where inverse mapping cannot be calculated analytically
- 3 Merging of subintervals in case of small overestimation of the union of the merged subintervals

Discrete-Time Dynamical Systems with Time-Invariant Parameter Uncertainties (4)



Discrete-Time Dynamical Systems with Time-Invariant Parameter Uncertainties (4)

Solution Approach 4

Computation of state variables $[\mathbf{x}_{k+1}]$ by explicit replacement of $[\mathbf{x}_k]$, $[\mathbf{x}_{k-1}]$, \dots , $[\mathbf{x}_2]$, $[\mathbf{x}_1]$ in terms of the initial state $[\mathbf{x}_0]$ and all parameter uncertainties $[\mathbf{x}_0]$, $[\mathbf{x}_1]$, \dots , $[\mathbf{x}_{k-1}]$, $[\mathbf{x}_k]$, i.e.,

$$[\mathbf{x}_{k+1}] = \mathbf{g}_k \left(\mathbf{g}_{k-1} \left(\dots \mathbf{g}_1 \left(\mathbf{g}_0 ([\mathbf{x}_0], [\mathbf{p}_0], \mathbf{u}_0, 0), [\mathbf{p}_1], \mathbf{u}_1, 1 \right) \dots \right) \right)$$

Evaluation by mean-value rule, monotonicity tests, and global optimization

Quantification of Overestimation (1)

- Basic criterion (\mathbf{u}_k not explicitly denoted)

$$\underbrace{\sum_{i=1}^n \alpha_i x_{k+1,i}}_{=: A(\mathbf{x}_{k+1})} = \underbrace{\sum_{i=1}^n \alpha_i g_i(\mathbf{x}_k, \mathbf{p}_k, k)}_{=: B(\mathbf{x}_k, \mathbf{p}_k, k)}$$

- Left-hand side evaluated purely with interval enclosures $\mathbf{x}_{k+1} \in [\mathbf{x}_{k+1}]$
- Right-hand side evaluated purely with interval enclosures $\mathbf{x}_k \in [\mathbf{x}_k]$
- Perform symbolic simplification of the right-hand side (or use of higher-order Taylor expansion techniques, mean value theorem)

Quantification of Overestimation (1)

- Basic criterion (\mathbf{u}_k not explicitly denoted)

$$\underbrace{\sum_{i=1}^n \alpha_i x_{k+1,i}}_{=: A(\mathbf{x}_{k+1})} = \underbrace{\sum_{i=1}^n \alpha_i g_i(\mathbf{x}_k, \mathbf{p}_k, k)}_{=: B(\mathbf{x}_k, \mathbf{p}_k, k)}$$

- Left-hand side evaluated purely with interval enclosures $\mathbf{x}_{k+1} \in [\mathbf{x}_{k+1}]$
- Right-hand side evaluated purely with interval enclosures $\mathbf{x}_k \in [\mathbf{x}_k]$
- Perform symbolic simplification of the right-hand side (or use of higher-order Taylor expansion techniques, mean value theorem)

The overestimation criterion often reflects physical conservation properties or is chosen to cancel out nonlinear terms with multiple dependencies on common interval variables

Quantification of Overestimation (2)

- Overestimation in simulations with L_k intervals

$$C_{\Sigma, k+1} = \frac{1}{L_k} \sum_{l=1}^{L_k} C_{k+1}^{\langle l \rangle} = \frac{1}{L_k} \sum_{l=1}^{L_k} \left\{ \text{diam} \left\{ A \left(\left[\mathbf{x}_{k+1}^{\langle l \rangle} \right] \right) \right\} \right. \\ \left. - \text{diam} \left\{ B \left(\left[\mathbf{x}_k^{\langle l \rangle} \right], \left[\mathbf{p}_k^{\langle l \rangle} \right], k \right) \right\} \right\}$$

with

$$C_{k+1} = \text{diam} \left\{ A \left(\left[\mathbf{x}_{k+1} \right] \right) \right\} - \text{diam} \left\{ B \left(\left[\mathbf{x}_k \right], \left[\mathbf{p}_k \right], k \right) \right\} \geq 0$$

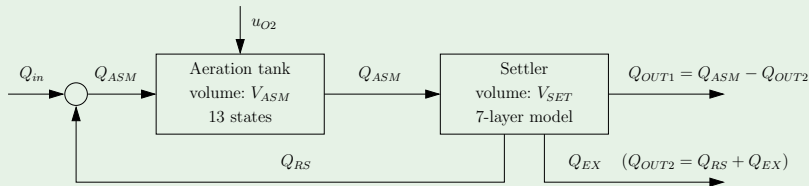
and

$$\text{diam} \left\{ A \left(\left[\mathbf{x}_{k+1} \right] \right) \right\} \geq \text{diam} \left\{ B \left(\left[\mathbf{x}_k \right], \left[\mathbf{p}_k \right], k \right) \right\}$$

- Subsequent subdivision of $\left[\mathbf{x}_{k+1} \right]$

Discretized Model of Biological Wastewater Treatment (1)

Schematic system representation



- Description of aeration tank by 13 ordinary differential equations
- Description of settler tank by 14 ordinary differential equations
- Discretization with respect to time

Discretized Model of Biological Wastewater Treatment (2)

Typical kinetics

- ① Aerobic growth of heterotrophic biomass

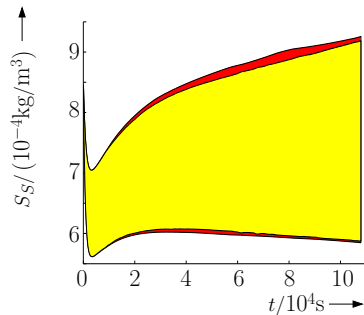
$$r_1 = \hat{\mu}_H \left(\frac{S_S}{K_S + S_S} \right) \left(\frac{S_O}{K_{OH} + S_O} \right) \left(\frac{S_{ALK}}{K_{ALKH} + S_{ALK}} \right) X_H$$

- ② Anoxic growth of heterotrophic biomass (denitrification)

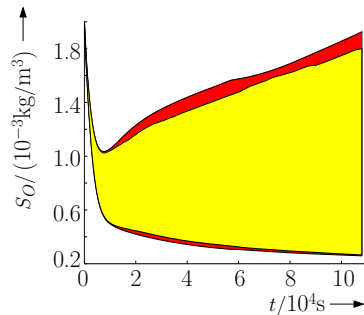
$$r_2 = \hat{\mu}_H \left(\frac{S_S}{K_S + S_S} \right) \left(\frac{S_{NO}}{K_{NO} + S_{NO}} \right) \cdot \left(\frac{K_{OH}}{K_{OH} + S_O} \right) \left(\frac{S_{ALK}}{K_{ALKH} + S_{ALK}} \right) \eta_G X_H$$

Discretized Model of Biological Wastewater Treatment (3)

Interval enclosure of S_S



Interval enclosure of S_O



Initial Value Problem with Interval Uncertainty

Definition of the initial value problem (IVP)

- Given set of ordinary differential equations (ODEs)

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

with smooth right-hand sides

- Uncertain initial conditions

$$\mathbf{x}(0) \in [\mathbf{x}_0] := [\mathbf{x}(0)] = [\underline{\mathbf{x}}(0); \bar{\mathbf{x}}(0)]$$

- Component-wise definition of interval vectors $[\mathbf{x}] = [[x_1] \ \dots \ [x_n]]^T$
with the vector entries $[x_i] = [\underline{x}_i; \bar{x}_i]$, $\underline{x}_i \leq x_i \leq \bar{x}_i$, $i = 1, \dots, n$

Initial Value Problem with Interval Uncertainty

Definition of the initial value problem (IVP)

- Given set of ordinary differential equations (ODEs)

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

with smooth right-hand sides

- Uncertain initial conditions

$$\mathbf{x}(0) \in [\mathbf{x}_0] := [\mathbf{x}(0)] = [\underline{\mathbf{x}}(0) ; \bar{\mathbf{x}}(0)]$$

- Uncertainty in parameters p_j , $j = 1, \dots, n_p$, and control (input) signals $\mathbf{u}(t)$ are assumed to be included in the expression for $\mathbf{f}(\mathbf{x}(t))$, e.g. $\dot{p}_j = 0$ or $\mathbf{u}(t) = \mathbf{u}(\mathbf{x}(t), \mathbf{p})$

Verified Integration of ODEs (1)

Solution Approach 1 (VNODE/ VNODE-LP by N. S. Nedialkov)

Computation of state enclosures of initial value problem IVP by series expansion **with respect to time**

$$[\mathbf{x}](t_{k+1}) = [\mathbf{x}](t_k) + h_k \cdot \phi([\mathbf{x}](t_k)) + [\mathbf{e}_k]$$

with

$$\phi(\cdot) := \sum_{i=1}^{\nu} \frac{h_k^{i-1}}{i!} \cdot \frac{d^{i-1} \mathbf{f}(\cdot)}{dt^{i-1}}$$

and the discretization error

$$[\mathbf{e}_k] := \frac{h_k^{\nu+1}}{(\nu+1)!} \cdot \frac{d^{\nu} \mathbf{f}(\cdot)}{dt^{\nu}} \Bigg|_{[\tau_k], [\mathbf{B}_k]}$$

evaluated for the time interval $[\tau_k] := [t_k ; t_{k+1}]$ and the bounding box of all states and parameters $[\mathbf{B}_k]$ for the time interval $[\tau_k]$ using Picard iteration

Verified Integration of ODEs (2)

Solution Approach 2

Computation of state enclosures of IVP by series expansion **with respect to time and initial states**, implemented in the Taylor-model-based solver COSY VI by M. Berz and K. Makino

Verified Integration of ODEs (2)

Solution Approach 2

Computation of state enclosures of IVP by series expansion **with respect to time and initial states**, implemented in the Taylor-model-based solver COSY VI by M. Berz and K. Makino

Solution Approach 3

Computation of two basic types of state enclosures

$$[\mathbf{x}_{\text{encl}}](t) := \tilde{\mathbf{x}}(t) + [\mathbf{R}](t)$$

$$[\mathbf{x}_{\text{encl}}](t) := \exp\left([\mathbf{\Lambda}](t) \cdot (t - t_0)\right) \cdot [\mathbf{x}_{\text{encl}}](t_0)$$

of IVP with a non-validated approximate solution $\tilde{\mathbf{x}}(t)$ and guaranteed error bounds $[\mathbf{R}](t)$, implemented in VALENCIA-IVP by A. Rauh and E. Auer for ODE systems given in state-space representation with extensions towards implicit ODE and DAE systems with arbitrary index

Picard Iteration: Theory and Illustrative Example

Picard iteration, integration horizon $t \in [\tau_k] := [t_k ; t_{k+1}]$

$$\Psi \left([\mathbf{B}]^{(\kappa)} \right) := [\mathbf{x}] (t_k) + [0 ; t_{k+1} - t_k] \cdot \mathbf{f} \left([\mathbf{B}]^{(\kappa)} \right) \quad \text{with} \quad [\mathbf{B}^{(0)}] = [\mathbf{x}] (t_k)$$

Different cases

- $\Psi \left([\mathbf{B}]^{(\kappa)} \right) \subseteq \left([\mathbf{B}]^{(\kappa)} \right)$: Continue with $[\mathbf{B}]^{(\kappa+1)} := \Psi \left([\mathbf{B}]^{(\kappa)} \right)$
- Else: Inflation of $[\mathbf{B}]^{(\kappa+1)}$ necessary
- Continue for maximum number of iterations or until no significant change can be noticed

Picard Iteration: Theory and Illustrative Example

Picard iteration, integration horizon $t \in [\tau_k] := [t_k ; t_{k+1}]$

$$\Psi \left([\mathbf{B}]^{(\kappa)} \right) := [\mathbf{x}] (t_k) + [0 ; t_{k+1} - t_k] \cdot \mathbf{f} \left([\mathbf{B}]^{(\kappa)} \right) \quad \text{with} \quad [\mathbf{B}^{(0)}] = [\mathbf{x}] (t_k)$$

Different cases

- $\Psi \left([\mathbf{B}]^{(\kappa)} \right) \subseteq \left([\mathbf{B}]^{(\kappa)} \right)$: Continue with $[\mathbf{B}]^{(\kappa+1)} := \Psi \left([\mathbf{B}]^{(\kappa)} \right)$
- Else: Inflation of $[\mathbf{B}]^{(\kappa+1)}$ necessary
- Continue for maximum number of iterations or until no significant change can be noticed

MATLAB examples

run_picard.m

Coupled Sets of ODEs: Wrapping Effect and Gershgorin Circle Theorem

Basic observation

Wrapping effect occurs in simulations of coupled sets of state equations

Closely related to overlapping discs in **Gershgorin Circle Theorem**

- Circle $\langle m_i, r_i \rangle$ in complex plane with midpoint $m_i = a_{ii}$, radius
$$r_i = \sum_{j=1, i \neq j}^n |a_{ji}|$$
- All eigenvalues are included in the union of all corresponding discs
- Disjoint regions: Number of eigenvalues in the disjoint subsets is equal to the number of overlapping discs in the subset
- Example: `gershgorin_ex.m`

Coupled Sets of ODEs: Wrapping Effect and Gershgorin Circle Theorem

Basic observation

Wrapping effect occurs in simulations of coupled sets of state equations

Note

- Decoupling of state equations by suitable coordinate transformation
- Use of subsequent exponential enclosure technique
- Exploitation of system property of cooperativity (monotonicity of the solution with respect to parameters and initial conditions): Multiple (verified) evaluations of the state equations for corner points of interval uncertainty

Fundamental Solution Approach

Definition of the state enclosure

$$\mathbf{x}^*(t) \in [\mathbf{x}](t) := \tilde{\mathbf{x}}(t) + [\mathbf{R}](t)$$

Constituents of the solution

- Approximate solution (non-verified) $\tilde{\mathbf{x}}(t)$
- Verified error bound $[\mathbf{R}](t)$
- Computation of the error bound by a suitable iteration scheme

Note

Without suitable counter-measures, solution enclosures may not converge, even for asymptotically stable systems

Fundamental Solution Approach

Definition of the state enclosure

$$\mathbf{x}^*(t) \in [\mathbf{x}](t) := \tilde{\mathbf{x}}(t) + [\mathbf{R}](t)$$

Constituents of the solution

- Approximate solution (non-verified) $\tilde{\mathbf{x}}(t)$
- Verified error bound $[\mathbf{R}](t)$
- Computation of the error bound by a suitable iteration scheme

Reference

Rauh, Andreas; Westphal, Ramona; Auer, Ekaterina; Aschemann, Harald: *Exponential Enclosure Techniques for the Computation of Guaranteed State Enclosures in VALENCIA-IVP*, Reliable Computing: Special volume devoted to material presented at SCAN 2012, Novosibirsk, Russia, Vol. 19, Issue 1, pp. 66-90, 2013.

Exponential Enclosure Technique

Definition of the state enclosure

- Representation of contracting state enclosures by using

$$\mathbf{x}^*(t) \in [\mathbf{x}_e](t) := \exp([\mathbf{\Lambda}] \cdot t) \cdot [\mathbf{x}_e](0)$$

with $0 \notin [x_{e,i}](0)$, $[\mathbf{x}_e](0) = [\mathbf{x}_0]$ for the diagonal matrix

$$[\mathbf{\Lambda}] := \text{diag} \{[\lambda_i]\} \quad , \quad i = 1, \dots, n$$

with element-wise negative real entries λ_i

- Definition of the interval matrix exponential

$$\exp([\mathbf{\Lambda}] \cdot t) := \text{diag} \{ \exp([\lambda_1] \cdot t), \dots, \exp([\lambda_n] \cdot t) \}$$

Exponential Enclosure Technique

Derivation of the iteration scheme

- Picard iteration

$$\mathbf{x}^*(t) \in [\mathbf{x}_e]^{(\kappa+1)}(t) := [\mathbf{x}_0] + \int_0^t \mathbf{f}([\mathbf{x}_e]^{(\kappa)}(s)) ds$$

- Reformulation by the time-dependent expression

$$\begin{aligned} \mathbf{x}^*(t) &\in \exp([\mathbf{\Lambda}]^{(\kappa+1)} \cdot t) \cdot [\mathbf{x}_e](0) = [\mathbf{x}_e]^{(\kappa+1)}(t) \\ &=: [\mathbf{x}_0] + \int_0^t \mathbf{f}(\exp([\mathbf{\Lambda}]^{(\kappa)} \cdot s) \cdot [\mathbf{x}_e](0)) ds \end{aligned}$$

Exponential Enclosure Technique

Derivation of the iteration scheme

- Picard iteration

$$\mathbf{x}^*(t) \in [\mathbf{x}_e]^{(\kappa+1)}(t) := [\mathbf{x}_0] + \int_0^t \mathbf{f}([\mathbf{x}_e]^{(\kappa)}(s)) ds$$

- Differentiation with respect to time and evaluation for $t \in [0; T]$

$$\begin{aligned} \dot{\mathbf{x}}^*([0; T]) &\in \text{diag}\{[\lambda_i]^{(\kappa+1)}\} \cdot \exp([\mathbf{\Lambda}]^{(\kappa+1)} \cdot [0; T]) \cdot [\mathbf{x}_e](0) \\ &\subseteq \mathbf{f}(\exp([\mathbf{\Lambda}]^{(\kappa)} \cdot [0; T]) \cdot [\mathbf{x}_e](0)) \end{aligned}$$

Exponential Enclosure Technique

Derivation of the iteration scheme

- Picard iteration

$$\mathbf{x}^*(t) \in [\mathbf{x}_e]^{(\kappa+1)}(t) := [\mathbf{x}_0] + \int_0^t \mathbf{f}([\mathbf{x}_e]^{(\kappa)}(s)) ds$$

- Convergence of the iteration process for

$$\exp([\mathbf{\Lambda}]^{(\kappa+1)} \cdot t) \cdot [\mathbf{x}_e](0) \subseteq \exp([\mathbf{\Lambda}]^{(\kappa)} \cdot t) \cdot [\mathbf{x}_e](0)$$

equivalent to

$$[\lambda_i]^{(\kappa+1)} \subseteq [\lambda_i]^{(\kappa)} \quad \text{and} \quad [\mathbf{\Lambda}]^{(\kappa+1)} \subseteq [\mathbf{\Lambda}]^{(\kappa)}$$

Exponential Enclosure Technique

Derivation of the iteration scheme

- Picard iteration

$$\mathbf{x}^*(t) \in [\mathbf{x}_e]^{(\kappa+1)}(t) := [\mathbf{x}_0] + \int_0^t \mathbf{f}([\mathbf{x}_e]^{(\kappa)}(s)) ds$$

- Resulting iteration formula

$$[\lambda_i]^{(\kappa+1)} := \frac{f_i \left(\exp([\mathbf{\Lambda}]^{(\kappa)} \cdot [0; T]) \cdot [\mathbf{x}_e](0) \right)}{\exp([\lambda_i]^{(\kappa)} \cdot [0; T]) \cdot [x_{e,i}](0)}, \quad i = 1, \dots, n$$

with the guaranteed state enclosure at the point $t = T$

$$\mathbf{x}^*(T) \in [\mathbf{x}_e](T) := \exp([\mathbf{\Lambda}] \cdot T) \cdot [\mathbf{x}_e](0)$$

Exponential Enclosure Technique: Special Case

Application to linear system models

- Simplified state equations

$$f_i(\mathbf{x}) = \sum_{j=1}^n a_{ij} \cdot x_j$$

- Simplification of the iteration formula

$$[\lambda_i]^{(\kappa+1)} := \sum_{j=1, i \neq j}^n \left\{ a_{ij} \cdot \exp \left(\left([\lambda_j]^{(\kappa)} - [\lambda_i]^{(\kappa)} \right) \cdot [0 ; T] \right) \cdot \frac{[x_{e,j}](0)}{[x_{e,i}](0)} \right\} \\ + a_{ii} \quad \text{with} \quad a_{ij} \in [a_{ij}]$$

Note

Free of overestimation if the equations are decoupled with $a_{ij} = 0, i \neq j$

Exponential Enclosure Technique: Special Case

Application to linear system models

- Simplified state equations

$$f_i(\mathbf{x}) = \sum_{j=1}^n a_{ij} \cdot x_j$$

- Simplification of the iteration formula

$$[\lambda_i]^{(\kappa+1)} := \sum_{j=1, i \neq j}^n \left\{ a_{ij} \cdot \exp \left(\left([\lambda_j]^{(\kappa)} - [\lambda_i]^{(\kappa)} \right) \cdot [0 ; T] \right) \cdot \frac{[x_{e,j}](0)}{[x_{e,i}](0)} \right\} \\ + a_{ii} \quad \text{with} \quad a_{ij} \in [a_{ij}]$$

Solution

Transformation of the state equations (decoupling) into real-valued Jordan canonical form (assumption of pairwise different eigenvalues)

Exponential Enclosure Technique: Special Case

Application to linear system models

- Simplified state equations

$$f_i(\mathbf{x}) = \sum_{j=1}^n a_{ij} \cdot x_j$$

- Simplification of the iteration formula

$$[\lambda_i]^{(\kappa+1)} := \sum_{j=1, i \neq j}^n \left\{ a_{ij} \cdot \exp \left(\left([\lambda_j]^{(\kappa)} - [\lambda_i]^{(\kappa)} \right) \cdot [0 ; T] \right) \cdot \frac{[x_{e,j}](0)}{[x_{e,i}](0)} \right\} \\ + a_{ii} \quad \text{with} \quad a_{ij} \in [a_{ij}]$$

Computation of state transformation matrices

Use of approximately computed (floating point) eigenvector matrix with a verified inverse and a verified transformation of the initial states

Exponential Enclosures: State-Space Transformation

Decoupling of the state equations

$$\dot{\mathbf{z}}(t) = \Sigma \cdot \mathbf{z}(t) \quad \text{with} \quad \Sigma = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{bmatrix} \quad \text{and} \quad \mathbf{z}(0) \in [\mathbf{z}(0)]$$

Applicability of the iteration scheme

The value zero must not be included in the true solution set
⇒ Oscillating systems with complex eigenvalues are problematic

$$\Sigma = \text{blkdiag}\{\dots, \bar{\Sigma}_i, \dots\}, \quad \bar{\Sigma}_i = \begin{bmatrix} \sigma_i & \omega_i \\ -\omega_i & \sigma_i \end{bmatrix}$$

Exponential Enclosures: State-Space Transformation

Decoupling of the state equations

$$\dot{\mathbf{z}}(t) = \Sigma \cdot \mathbf{z}(t) \quad \text{with} \quad \Sigma = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{bmatrix} \quad \text{and} \quad \mathbf{z}(0) \in [\mathbf{z}(0)]$$

Solution: Use of complex valued Jordan canonical form, see also `ODE_complex.m`

Complex-valued IVP $\dot{\mathbf{z}}(t) = \Sigma \cdot \mathbf{z}(t)$ with $\mathbf{z}(0) \in \mathbb{C}^n$, $\mathbf{z}(0) \in [\mathbf{z}(0)]$

$$\Sigma = \text{blkdiag}\{\dots, \Sigma_i, \dots\}, \quad \Sigma_i = \begin{bmatrix} \sigma_i + j\omega_i & 0 \\ 0 & \sigma_i - j\omega_i \end{bmatrix}$$

Exponential Enclosures: State-Space Transformation

Decoupling of the state equations

$$\dot{\mathbf{z}}(t) = \mathbf{\Sigma} \cdot \mathbf{z}(t) \quad \text{with} \quad \mathbf{\Sigma} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{bmatrix} \quad \text{and} \quad \mathbf{z}(0) \in [\mathbf{z}(0)]$$

Analysis of the applicability: Eigenvalues of the multiplicity $\delta_i = 1$

- Exact solution $z_i(t) = e^{(\sigma_i + j\omega_i) \cdot t} \cdot z_i(0)$, $z_{i+1}(t) = e^{(\sigma_i - j\omega_i) \cdot t} \cdot z_{i+1}(0)$
- Iteration procedure is always applicable for $z_i(0) \neq 0$ due to

$$|z_i(t)|^2 = \left(e^{(\sigma_i + j\omega_i) \cdot t} \cdot e^{(\sigma_i - j\omega_i) \cdot t} \right) \cdot |z_i(0)|^2 = e^{2\sigma_i t} \cdot |z_i(0)|^2 \neq 0$$

Exponential Enclosures: State-Space Transformation

Decoupling of the state equations

$$\dot{\mathbf{z}}(t) = \mathbf{\Sigma} \cdot \mathbf{z}(t) \quad \text{with} \quad \mathbf{\Sigma} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{bmatrix} \quad \text{and} \quad \mathbf{z}(0) \in [\mathbf{z}(0)]$$

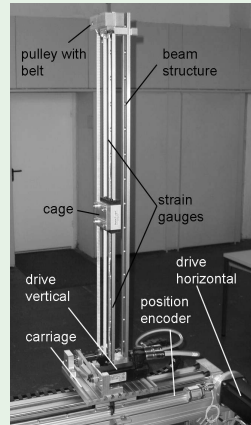
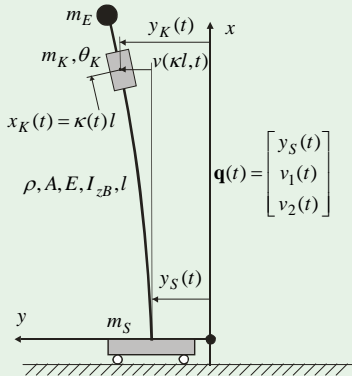
Analysis of the applicability: Eigenvalues of the multiplicity $\delta_i = 1$

- Solution remains asymptotically stable for decoupled (oscillatory) linear systems
- (Limited) Overestimation in the initial conditions
- **Multiple eigenvalues lead to a non-negligible wrapping effect**

Simulation of the Dynamics of a Controlled High-Speed Rack Feeder System

Test rig at the Chair of Mechatronics, University of Rostock

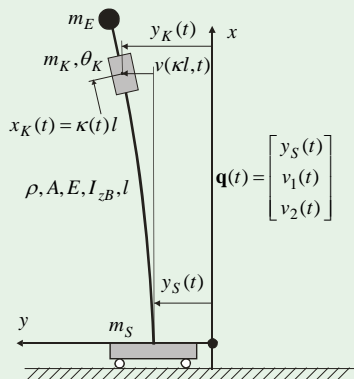
Elastic multibody model



Simulation of the Dynamics of a Controlled High-Speed Rack Feeder System

Test rig at the Chair of Mechatronics, University of Rostock

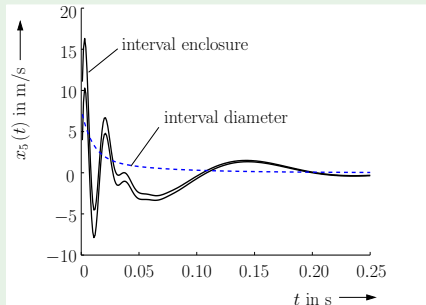
Elastic multibody model



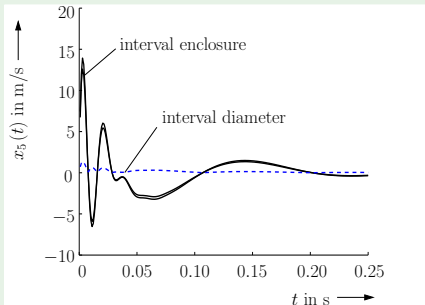
- Linear, time-invariant system model for $\kappa = \text{const}$
- Nonlinear (resp. linear, time-varying) model for $\kappa \neq \text{const}$
- System order 6 with real as well as complex eigenvalues of multiplicity 1
- Asymptotically stable after design of a suitable state feedback controller

Representative Simulation Results

Complex-valued exponential enclosure technique



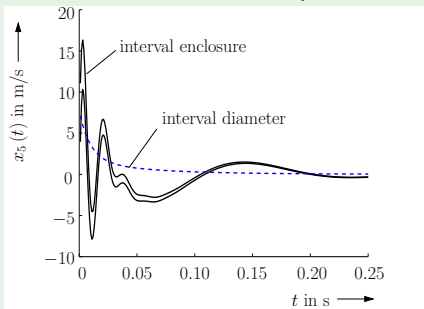
VNODE-LP



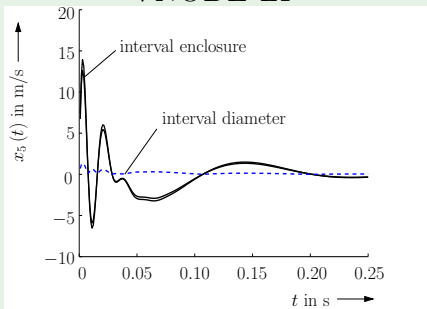
- Increased diameters of the exponential state enclosures due to the wrapping effect in initial conditions: complex intervals are represented in midpoint/ radius form provided by INTLAB

Representative Simulation Results

Complex-valued exponential enclosure technique



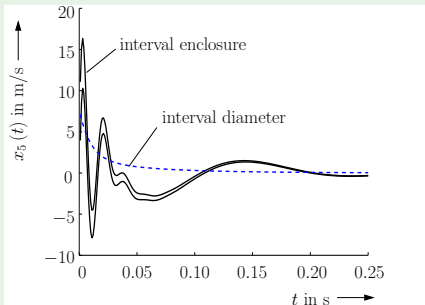
VNODE-LP



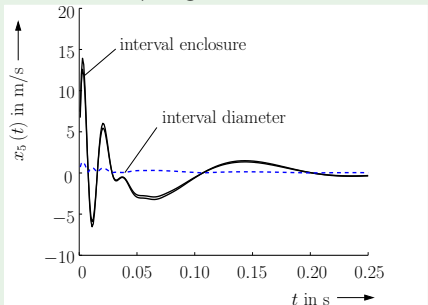
- Similar results can be obtained for the case of time-varying parameters κ with sufficiently small integration step sizes

Representative Simulation Results

Complex-valued exponential enclosure technique



VNODE-LP



- A. Rauh, R. Westphal, H. Aschemann, Harald: *Verified Simulation of Control Systems with Interval Parameters Using an Exponential State Enclosure Technique*, Proc. of IEEE Intl. Conf. on Methods and Models in Automation and Robotics MMAR 2013, Miedzyzdroje, Poland, 2013.

Extension to Systems with Multiple (Complex) Eigenvalues

Canonical form with real eigenvalues

$$\dot{\mathbf{z}}(t) = \Sigma \cdot \mathbf{z}(t) \quad \text{with} \quad \Sigma = \text{blkdiag}\{\lambda_1, \lambda_2, \dots, \Sigma_i, \dots, \lambda_n\},$$

$$\Sigma_i = \begin{bmatrix} \lambda_i & 1 & \dots & 0 \\ 0 & \lambda_i & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 0 & \dots & 0 & \lambda_i \end{bmatrix} \in \mathbb{R}^{\delta_i \times \delta_i} \quad \text{and} \quad \mathbf{z}(0) \in [\mathbf{z}(0)]$$

- Solve decoupled equations independently
- Jordan block with $\delta_i > 1 \implies$ solve “from bottom to top”
 \implies Further details: see Section *Outlook*

Uncertain Compartment Model of Granulopoiesis (1)

Granulopoiesis

- Seven compartments representing the proliferation stages of granulocytes (human blood cells)
- Stem cell compartment (S): cell growth until reaching circulating blood
- Function compartment (F): cells in blood
- Compartment bone marrow (CBM)
- Compartment blood (CBL)
- Precursor cells (P)
- Mature cells (M)
- Reserve cells (R)

Uncertain Compartment Model of Granulopoiesis (2)

State variables

$$x_1 = S$$

$$x_2 = CBM_1, \dots, x_{11} = CBM_{10}$$

$$x_{12} = CBL_1, \dots, x_{21} = CBL_{10}$$

$$x_{22} = P_1, \dots, x_{31} = P_{10}$$

$$x_{32} = M$$

$$x_{33} = R$$

$$x_{34} = F$$

$$x_{35} = RegI$$

$$x_{36} = RegII$$

Uncertain Compartment Model of Granulopoiesis (3)

Abbreviations

$$\begin{aligned}u_1 &= \gamma_1 \exp(-\nu_1 x_1) \\ &\quad + \gamma_2 \exp(-\nu_2 \cdot (x_2 + \dots + x_{11} + x_{22} + \dots + x_{33})) + \gamma_3 \\u_2 &= \gamma_4 - \gamma_5 \exp(-\nu_3 x_{35}) & u_6 &= \gamma_6 - \gamma_7 \exp(-\nu_4 x_{35}) \\u_3 &= 2(1 - \rho)u_1 x_1 & u_7 &= \lambda_p x_{31} \\u_4 &= \beta & u_8 &= \gamma_8 - \gamma_9 \exp(-\nu_5 x_{36}) \\u_5 &= \lambda_c x_{11} & u_9 &= u_8 x_{33} \\u_{10} &= \gamma_{10} \exp(-\nu_6 \cdot (g_1 x_1 + g_2 \cdot (x_2 + \dots + x_{11}) + g_3 \cdot (x_{22} + \dots + x_{34}))) \\u_{11} &= \gamma_{11} \exp(-\nu_7 x_{34})\end{aligned}$$

Uncertain Compartment Model of Granulopoiesis (4)

Nonlinear set of state equations (Fliedner and Steinbach)

$$\dot{x}_1 = (2\rho - 1)u_1x_1$$

$$\dot{x}_2 = u_3 - \lambda_c x_2 + u_2 x_2 - u_4 x_2 + \Phi x_{12}$$

$$\dot{x}_i = \lambda_c x_{i-1} - \lambda_c x_i + u_2 x_i - u_4 x_i + \Phi x_{i+10} \quad \text{for } i = 3, \dots, 11$$

$$\dot{x}_i = u_4 x_{i-10} - \Phi x_i \quad \text{for } i = 12, \dots, 21$$

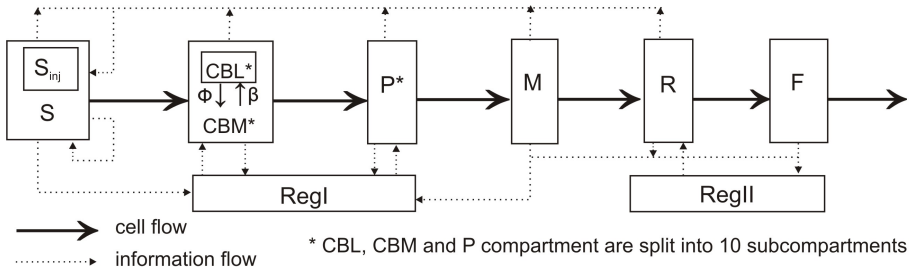
$$\dot{x}_{22} = u_5 + u_6 x_{22} - \lambda_p x_{22}$$

$$\dot{x}_i = \lambda_p x_{i-1} + u_6 x_i - \lambda_p x_i \quad \text{for } i = 23, \dots, 31$$

$$\dot{x}_{32} = u_7 - \lambda_M x_{32}, \quad \dot{x}_{33} = \lambda_M x_{32} - u_8 x_{33}, \quad \dot{x}_{34} = u_9 - \lambda_F x_{34}$$

$$\dot{x}_{35} = u_{10} - \lambda_{RegI} x_{35}, \quad \dot{x}_{36} = u_{11} - \lambda_{RegII} x_{36}$$

Uncertain Compartment Model of Granulopoiesis (5)



Uncertain Compartment Model of Granulopoiesis (6)

Constraint A: Directly evaluated using a guaranteed enclosure for $x = [x_1, \dots, x_{36}]^T$

$$H_{H,l} := x_{36+l} = x_{l+1} + x_{l+11} \quad \text{with } l = 1, \dots, 10$$

Substitution of the interval enclosures $[x_{l+1}]$, $[x_{l+11}]$ for the state variables

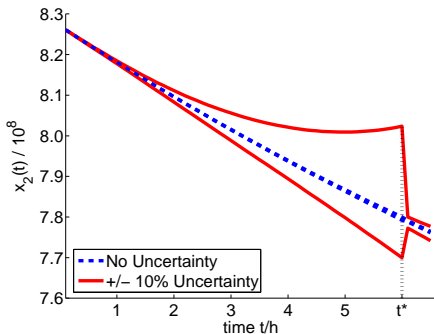
Constraint B: Integration of the corresponding time derivatives via additional ODEs

$$\dot{H}_{V,l} := \begin{cases} \dot{x}_{37} = u_3 - \lambda_c x_2 + u_2 x_2 & \text{for } l = 1 \\ \dot{x}_{36+l} = \lambda_c x_{l+1} - \lambda_c x_{l+2} + u_2 x_{l+2} & \text{for } l = 2, \dots, 10 \end{cases}$$

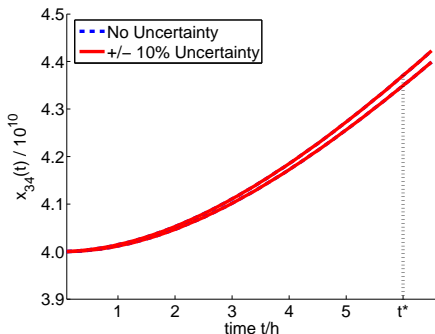
with initial conditions $H_{V,l}(0) = H_{H,l}(0)$ for $l = 1, \dots, 10$ evaluated for the enclosure $[\mathbf{x}(0)]$

Uncertain Compartment Model of Granulopoiesis (7)

Interval enclosure of $x_2(t)$, cell count of the CBM_1 compartment



Interval enclosure of $x_{34}(t)$, cell count of the F compartment



Branch and bound heuristics for (physical) conservation properties, similar use for mass and energy conservation laws (e.g. Hamiltonian systems)

Conclusions and Outlook on Future Work

- Computation of verified state enclosures for discrete-time and continuous-time dynamic systems
- Handling of uncertainty in initial conditions and parameters
- Possibilities for detection and elimination of overestimation (especially wrapping effect)
- Novel iteration scheme based on complex-valued interval arithmetic
- Extensions to systems with multiple conjugate complex eigenvalues

Conclusions and Outlook on Future Work

- Computation of verified state enclosures for discrete-time and continuous-time dynamic systems
- Handling of uncertainty in initial conditions and parameters
- Possibilities for detection and elimination of overestimation (especially wrapping effect)
- Novel iteration scheme based on complex-valued interval arithmetic
- Extensions to systems with multiple conjugate complex eigenvalues
- Verified, real-time capable safety analysis of control strategies:
Verification of compatibility with state constraints
- Online sensitivity analysis in predictive control frameworks
- Online sensitivity analysis for state and parameter estimation
- Analysis of feedback linearizing control procedures
- Verification of interval-based sliding mode techniques

Extension to Systems with Multiple (Complex) Eigenvalues

Canonical form with conjugate complex eigenvalues

$$\Sigma = \text{blkdiag}\{\dots, \Sigma_i^+, \Sigma_i^-, \dots\}$$

$$\text{with } \Sigma_i^+ = \begin{bmatrix} \sigma_i + j\omega_i & 1 & \dots & 0 \\ 0 & \sigma_i + j\omega_i & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 0 & \dots & 0 & \sigma_i + j\omega_i \end{bmatrix} \in \mathbb{C}^{\delta_i \times \delta_i}$$

$$\text{and } \Sigma_i^- = \begin{bmatrix} \sigma_i - j\omega_i & 1 & \dots & 0 \\ 0 & \sigma_i - j\omega_i & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 0 & \dots & 0 & \sigma_i - j\omega_i \end{bmatrix} \in \mathbb{C}^{\delta_i \times \delta_i}$$

for each eigenvalue pair $\sigma_i \pm j\omega_i$ with $\delta_i > 1$

Extension to Systems with Multiple (Complex) Eigenvalues

Analytic representation of the solutions $z_{i+j}(t)$, $j = 0, \dots, \delta_i - 1$

$$z_{i+j}^*(t) = e^{(\sigma_i + j\omega_i) \cdot t} \cdot \left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} \cdot z_{i+\zeta}(0) \right)$$

Applicability of the standard exponential enclosure technique

- Computation of the square of its absolute value

$$|z_{i+j}^*(t)|^2 = e^{2\sigma_i \cdot t} \cdot \left| \left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} \cdot z_{i+\zeta}(0) \right) \right|^2 = e^{2\sigma_i \cdot t} \cdot |\chi_j|^2$$

- Exponential enclosure technique of VALENCIA-IVP is applicable if

$$0 \notin \left\{ \chi_j \mid \chi_j = \Re\{\chi_j\} + j\Im\{\chi_j\}, z_{i+\zeta}(0) \in [z_{i+\zeta}(0)], t \in [0; T] \right\}$$

holds for any $j = 0, \dots, \delta_i - 1$

Extension to Systems with Multiple (Complex) Eigenvalues

Modification of the iteration scheme

- Definition of the enclosure and its time derivative

$$z_{i+j} = \left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{\lambda_{i+j}t}$$

$$\begin{aligned} \dot{z}_{i+j} &= \lambda_{i+j} \cdot \left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{\lambda_{i+j}t} \\ &\quad + \left(\sum_{\zeta=j+1}^{\delta_i-1} \frac{t^{\zeta-(j+1)}}{(\zeta-(j+1))!} z_{i+\zeta}(0) \right) \cdot e^{\lambda_{i+j}t} \end{aligned}$$

- Evaluation for $j = 0, \dots, \delta_i - 1$, $z_{\zeta}(0) \in [z_{\zeta}(0)]$, $t \in [0 ; T]$
- Compute enclosures $[\lambda_i], \dots, [\lambda_{i+\delta_i-1}]$

Extension to Systems with Multiple (Complex) Eigenvalues

Modification of the iteration scheme

- Subsystem model (eigenvalue λ_i^* with multiplicity $\delta_i > 1$)

$$\dot{z}_i = \lambda_i^* \cdot z_i + z_{i+1}$$

$$\dot{z}_{i+1} = \lambda_i^* \cdot z_{i+1} + z_{i+2}$$

$$\vdots$$

$$\dot{z}_{i+\delta_i-1} = \lambda_i^* \cdot z_{i+\delta_i-1}$$

- One-sided decoupling of equations
- Solutions can be computed in the order $z_{i+\delta_i-1}, z_{i+\delta_i-2}, \dots, z_{i+1}, z_i$

Extension to Systems with Multiple (Complex) Eigenvalues

Iteration scheme

- Iteration procedure

$$\begin{aligned}
 [\lambda_{i+j}]^{(\kappa+1)} := & \frac{\lambda_i^* \cdot \left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{[\lambda_{i+j}]^{(\kappa)} t}}{\left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{[\lambda_{i+j}]^{(\kappa)} t}} \\
 & + \frac{\left(\sum_{\zeta=j+1}^{\delta_i-1} \frac{t^{\zeta-(j+1)}}{(\zeta-(j+1))!} z_{i+\zeta}(0) \right) \cdot \left(e^{[\lambda_{i+j+1}] t} - e^{[\lambda_{i+j}]^{(\kappa)} t} \right)}{\left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{[\lambda_{i+j}]^{(\kappa)} t}}
 \end{aligned}$$

- One-sided decoupling: $[\lambda_{i+j}]$ depends on result for $[\lambda_{i+j+1}]$

Extension to Systems with Multiple (Complex) Eigenvalues

Iteration scheme

- Iteration procedure

$$[\lambda_{i+j}]^{(\kappa+1)} := \frac{\lambda_i^* \cdot \left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{[\lambda_{i+j}]^{(\kappa)} t}}{\left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{[\lambda_{i+j}]^{(\kappa)} t}} + \frac{\left(\sum_{\zeta=j+1}^{\delta_i-1} \frac{t^{\zeta-(j+1)}}{(\zeta-(j+1))!} z_{i+\zeta}(0) \right) \cdot \left(e^{[\lambda_{i+j+1}] t} - e^{[\lambda_{i+j}]^{(\kappa)} t} \right)}{\left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right) \cdot e^{[\lambda_{i+j}]^{(\kappa)} t}}$$

- Evaluation for $\lambda_i^* \in [\lambda_i^*]$, $z_\zeta(0) \in [z_\zeta(0)]$, $t \in [0 ; T]$

Extension to Systems with Multiple (Complex) Eigenvalues

Iteration scheme

- Iteration procedure – simplified

$$[\lambda_{i+j}]^{(\kappa+1)} := \lambda_i^* + \frac{\left(\sum_{\zeta=j+1}^{\delta_i-1} \frac{t^{\zeta-(j+1)}}{(\zeta-(j+1))!} z_{i+\zeta}(0) \right) \cdot \left(e^{([\lambda_{i+j+1}] - [\lambda_{i+j}]^{(\kappa)})t} - 1 \right)}{\left(\sum_{\zeta=j}^{\delta_i-1} \frac{t^{\zeta-j}}{(\zeta-j)!} z_{i+\zeta}(0) \right)}$$

- Evaluation for $\lambda_i^* \in [\lambda_i^*]$, $z_{\zeta}(0) \in [z_{\zeta}(0)]$, $t \in [0 ; T]$

Extension to Systems with Multiple (Complex) Eigenvalues

Practically important generalization

- If **additional terms** are included in the system model

$$\dot{z}_i = \lambda_i^* \cdot z_i + z_{i+1} + f_i(\mathbf{z})$$

$$\dot{z}_{i+1} = \lambda_i^* \cdot z_{i+1} + z_{i+2} + f_{i+1}(\mathbf{z})$$

$$\vdots$$

$$\dot{z}_{i+\delta_i-1} = \lambda_i^* \cdot z_{i+\delta_i-1} + f_{i+\delta_i-1}(\mathbf{z}) ,$$

a vector-valued iteration has to be performed

$$\begin{bmatrix} [\lambda_i]^{(\kappa+1)} \\ [\lambda_{i+1}]^{(\kappa+1)} \\ \vdots \\ [\lambda_{i+\delta_i-1}]^{(\kappa+1)} \end{bmatrix} \stackrel{!}{\subset} \begin{bmatrix} [\lambda_i]^{(\kappa)} \\ [\lambda_{i+1}]^{(\kappa)} \\ \vdots \\ [\lambda_{i+\delta_i-1}]^{(\kappa)} \end{bmatrix}$$

Extension to Systems with Multiple (Complex) Eigenvalues

Simplified enclosure

- Definition of the enclosure and its time derivative

$$z_{i+j} = (z_{i+j}(0) + t \cdot z_{i+j+1}(0)) \cdot e^{\lambda_{i+j}t}$$

$$\dot{z}_{i+j} = \lambda_{i+j} (z_{i+j}(0) + t \cdot z_{i+j+1}(0)) \cdot e^{\lambda_{i+j}t} + z_{i+j+1}(0) \cdot e^{\lambda_{i+j}t}$$

- Simplified iteration procedure

$$[\lambda_{i+j}]^{(\kappa+1)} := \frac{\lambda_i^* \cdot (z_{i+j}(0) + t \cdot z_{i+j+1}(0)) \cdot e^{[\lambda_{i+j}]^{(\kappa)}t}}{(z_{i+j}(0) + t \cdot z_{i+j+1}(0)) \cdot e^{[\lambda_{i+j}]^{(\kappa)}t}} + \frac{z_{i+j+1}(0) \cdot (e^{[\lambda_{i+j+1}]t} - e^{[\lambda_{i+j}]^{(\kappa)}t})}{(z_{i+j}(0) + t \cdot z_{i+j+1}(0)) \cdot e^{[\lambda_{i+j}]^{(\kappa)}t}}$$

Extension to Systems with Multiple (Complex) Eigenvalues

Simplified enclosure

- Definition of the enclosure and its time derivative

$$z_{i+j} = (z_{i+j}(0) + t \cdot z_{i+j+1}(0)) \cdot e^{\lambda_{i+j}t}$$

$$\dot{z}_{i+j} = \lambda_{i+j} (z_{i+j}(0) + t \cdot z_{i+j+1}(0)) \cdot e^{\lambda_{i+j}t} + z_{i+j+1}(0) \cdot e^{\lambda_{i+j}t}$$

- Simplified iteration procedure

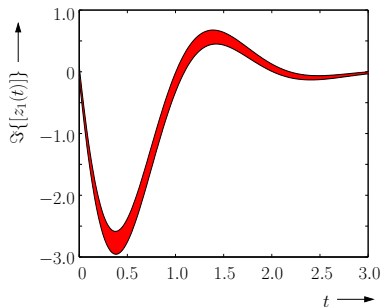
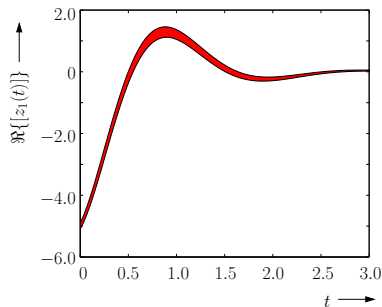
$$[\lambda_{i+j}]^{(\kappa+1)} := \lambda_i^* + \frac{e^{([\lambda_{i+j+1}] - [\lambda_{i+j}]^{(\kappa)})t} - 1}{\frac{z_{i+j}(0)}{z_{i+j+1}(0)} + t}$$

- Evaluation for $j = 0, \dots, \delta_i - 1$, $z_\zeta(0) \in [z_\zeta(0)]$, $t \in [0 ; T]$
- Decoupled/ coupled evaluation as before

Illustrative Example

Complex IVP with $\delta_i = 2$

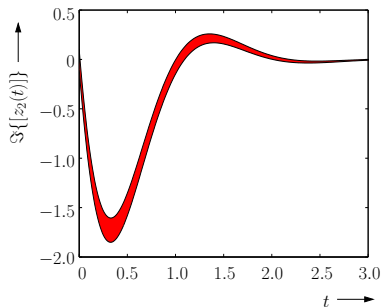
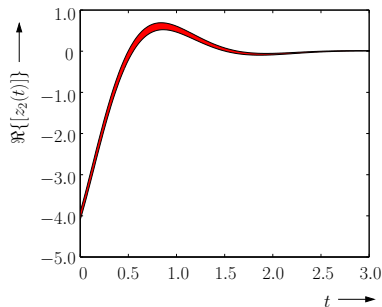
$$\mathbf{z}(0) \in \begin{bmatrix} \langle -5, 0.1 \rangle \\ \langle -4, 0.1 \rangle \end{bmatrix}, \quad \lambda^* \in \langle -2 + 3j, 0.1 \rangle$$



Illustrative Example

Complex IVP with $\delta_i = 2$

$$\mathbf{z}(0) \in \begin{bmatrix} \langle -5, 0.1 \rangle \\ \langle -4, 0.1 \rangle \end{bmatrix}, \quad \lambda^* \in \langle -2 + 3j, 0.1 \rangle$$



Dziękuję bardzo za uwagę!

Thank you for your attention!

Спасибо за Ваше внимание!

Merci beaucoup pour votre attention!

¡Muchas gracias por su atención!

Grazie mille per la vostra attenzione!

Vielen Dank für Ihre Aufmerksamkeit!

