Reachability Analysis for High-Index Linear Differential Algebraic Equations (DAEs)

https://github.com/verivital/daev/

17th International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS'19), August 27, 2019

Hoang-Dung Tran, Luan Viet Nguyen, Nathaniel Hamilton, Weiming Xiang & **Taylor T. Johnson**

VeriVITAL-The Verification and Validation for Intelligent and Trustworthy Autonomy Laboratory (http://www.verivital.com)

Electrical Engineering and Computer Science (EECS)

Motivation: Mass Dampers

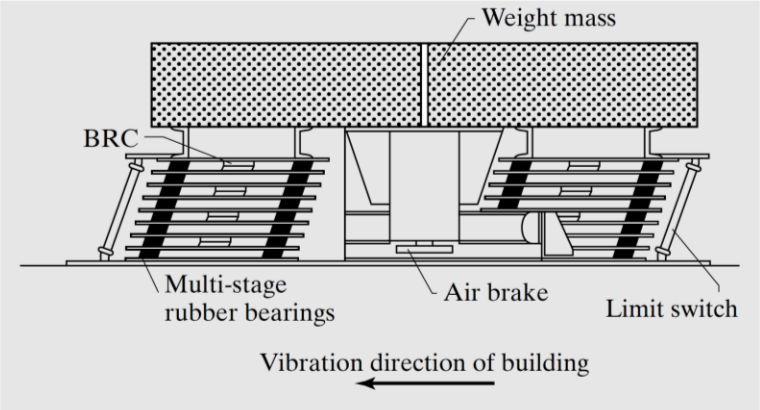


FIGURE 4.4: Tuned mass damper with spring and damper assemblage.

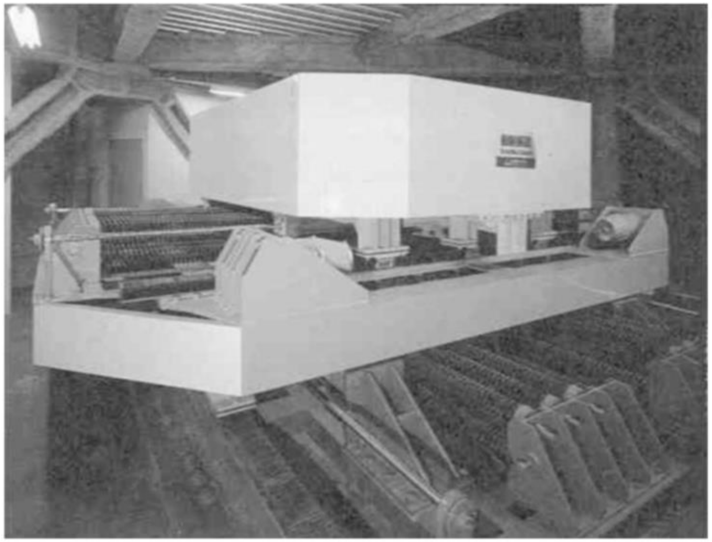


FIGURE 4.3: Tuned mass damper for Chiba-Port Tower. (Courtesy of J. Connor.)

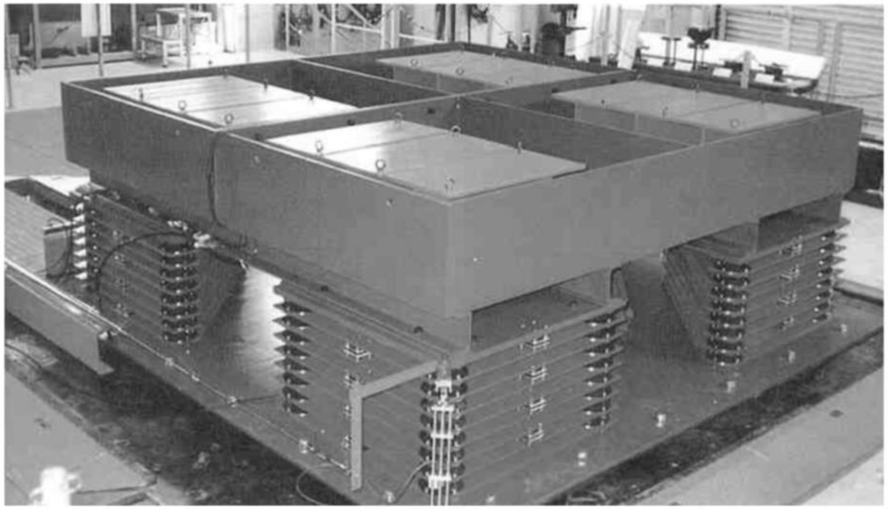
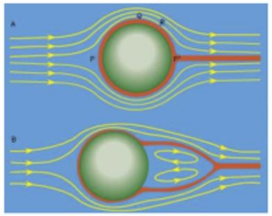


FIGURE 4.6: Tuned mass damper—Huis Ten Bosch Tower, Nagasaki. (Courtesy of J. Connor.)

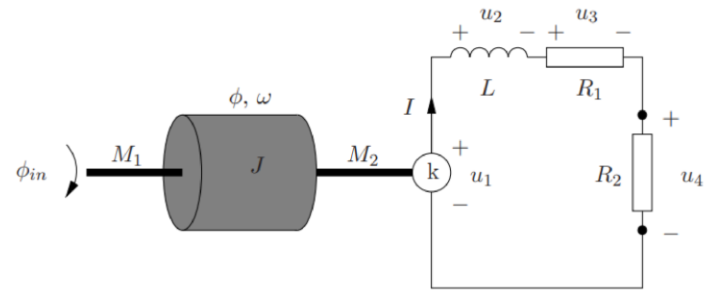
[Intro to Structural Motion Control, Connor 2003]

Motivation

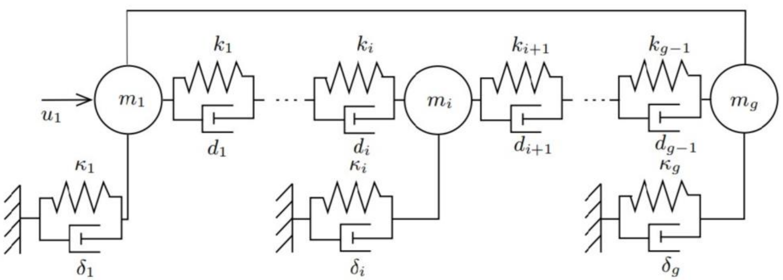


$$\frac{\partial v}{\partial t} = \Delta v - \nabla \rho + f, \text{ in } \Omega \times (0,T)$$
$$\nabla v = 0, \text{ in } \Omega \times (0,T),$$

Index-2 semi-discretized Stoke System (fluids)



Index-3 DAE system electrical generator (power)



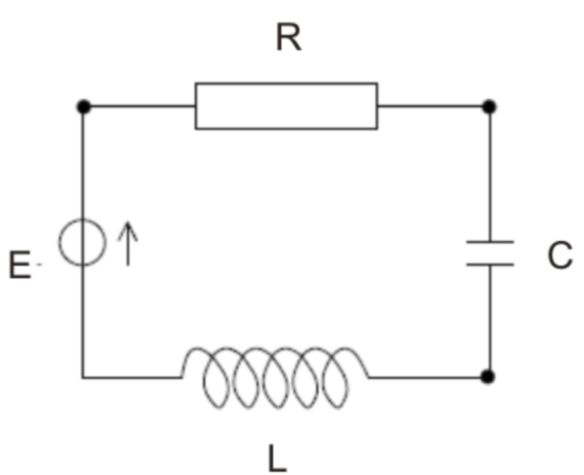
Index-3 damped mass-spring system (earthquake)

Index-2 interconnected rotating masses (IRM) system (automotive)

- Most existing cyber-physical systems (CPS) verification techniques ***only focus on physical behaviors as ordinary differential equations (ODEs), or hybrid variants thereof (hybrid automata, etc.)***
- Many CPS domains naturally model systems as DAEs instead of ODEs
  - Mechatronics, robotics, electrical circuits, earthquake engineering, water distribution networks / fluid dynamics (certain problems), process/chemical engineering, …

DAE Modeling Intuition

- Consider an RLC (resistor, inductor, capacitor) circuit
- Kirchhoff's current law (KCL) and voltage law (KVL) => algebraic constraints + ODEs for transient behavior
  - KCL: conservation of current: $i_E = i_R = i_C = i_L$
  - KVL: conservation of energy: $V_R + V_C + V_L + V_E = 0$
  - Ohm's laws:



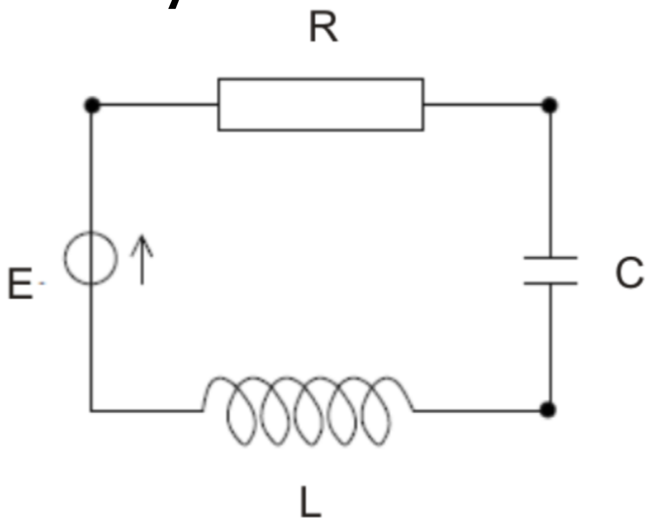
$$\mathrm{C}\dot{V}_C = i_c$$

$$\mathrm{L}\dot{V}_L = i_L$$

$$V_R = R\, i_R$$

DAE Modeling Intuition

- Replace equal currents (i_R to i_E, i_C to i_L), don't have to, but reduces dimensionality for fewer state variables



$$\dot{V}_C = \frac{1}{C} i_L$$

$$\dot{V}_L = \frac{1}{L} i_E$$

$$0 = V_R + R i_E$$

$$0 = V_E + V_R + V_C + V_L$$

$$0 = i_L - i_E$$

- Now a DAE system with:

$$x(t) = \begin{bmatrix} V_C(t) \\ V_L(t) \\ V_R(t) \\ i_L(t) \\ i_E(t) \end{bmatrix}$$

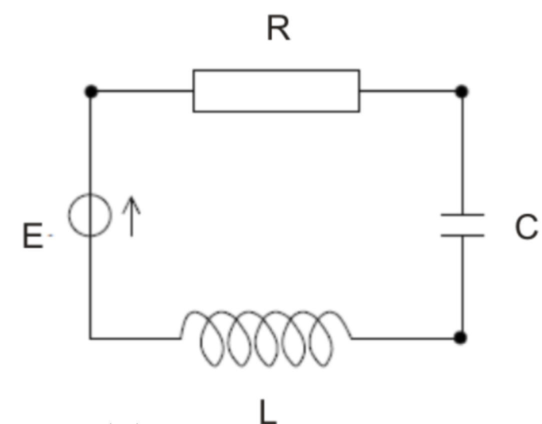
DAE Modeling Intuition

- Linear DAE system:

$$A = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{C} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{L} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\frac{dx}{dt} = \dot{x} = Ax$$
$$0 = Bx + Dz$$

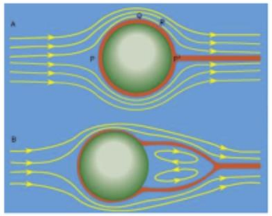
$$\dot{V}_C = \frac{1}{C} i_L$$
$$\dot{V}_L = \frac{1}{L} i_E$$
$$0 = V_R + R i_E$$
$$0 = V_E + V_R + V_C + V_L$$
$$0 = i_L - i_E$$



$$B = \begin{bmatrix} 0 & 0 & 1 & 0 & R \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix} \qquad D = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

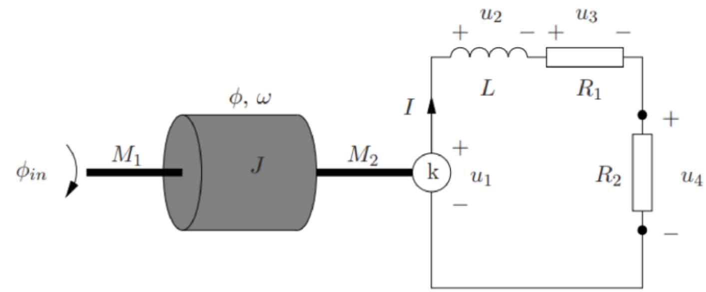
$$x(t) = \begin{bmatrix} V_C(t) \\ V_L(t) \\ V_R(t) \\ i_L(t) \\ i_E(t) \end{bmatrix}, \qquad z(t) = V_E(t)$$

Motivation

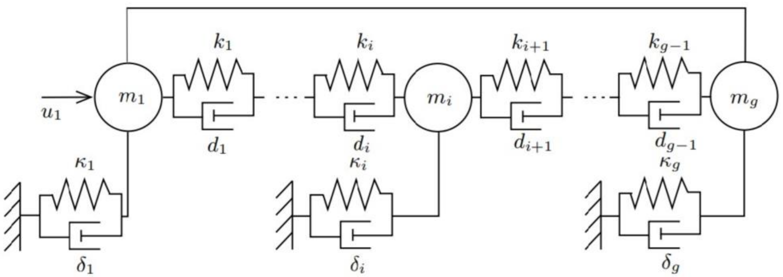


$$\frac{\partial v}{\partial t} = \Delta v - \nabla \rho + f, \text{ in } \Omega \times (0,T)$$
$$\nabla v = 0, \text{ in } \Omega \times (0,T),$$

Index-2 semi-discretized Stoke System (fluids)



Index-3 DAE system electrical generator (power)



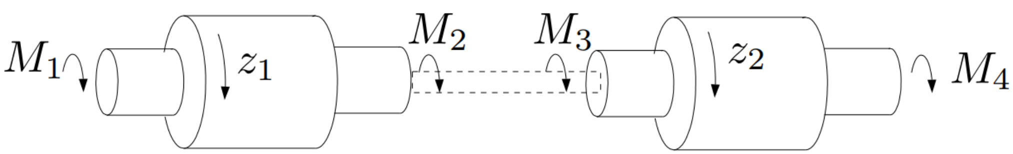
Index-3 damped mass-spring system (earthquake)

Index-2 interconnected rotating masses (IRM) system (automotive)

- Most existing cyber-physical systems (CPS) verification techniques ***only focus on ODE dynamics, or hybrid variants thereof (hybrid automata, etc.)***
- ***Verifying DAE systems is more complex than ODE systems***
- No existing works (to our knowledge) on ***verifying high-index (>1) DAEs***
- ***Scalability: state-space explosion / "curse of dimensionality"***
- **How to verify safety of systems with DAE dynamics?**

Linear DAE Systems

- Linear DAE System: $\boldsymbol{E\dot{x}(t) = Ax(t) + Bu(t)}$
  - $x(t) \in \mathrm{R}^n$ is the state vector
  - $u(t) \in \mathrm{R}^m$ is the s input vector
  - $E, A \in \mathrm{R}^{n\times n}$ and $B \in \mathrm{R}^{n\times m}$ are the DAEs matrices, where E is *singular (non-invertible)*
  - ***Index of a DAE***: typically (can depend on initial conditions) the minimum number of times to differentiate DAEs wrt t to get ODEs ("**index reduction**"), where ODEs are called index-0, can typically evaluate rank(E) to check
- Example: Index-2 interconnected rotating masses (IRM) system

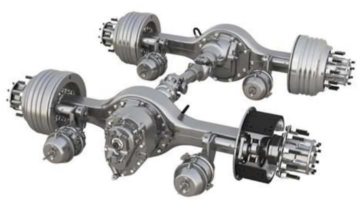


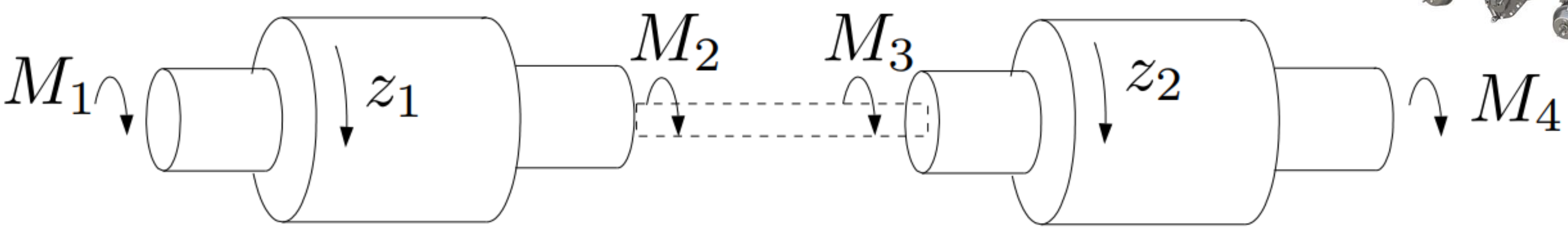
$$\begin{bmatrix} J_1 & 0 & 0 & 0 \\ 0 & J_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{z}_1(t) \\ \dot{z}_2(t) \\ \dot{M}_2(t) \\ \dot{M}_3(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & -1 \\ -1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ M_2(t) \\ M_3(t) \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} M_1(t) \\ M_4(t) \end{bmatrix}$$

Where $J_1 = 1, J_2 = 2, M_2(t) + M_3(t) = 0, z_1(t) = z_2(t)$

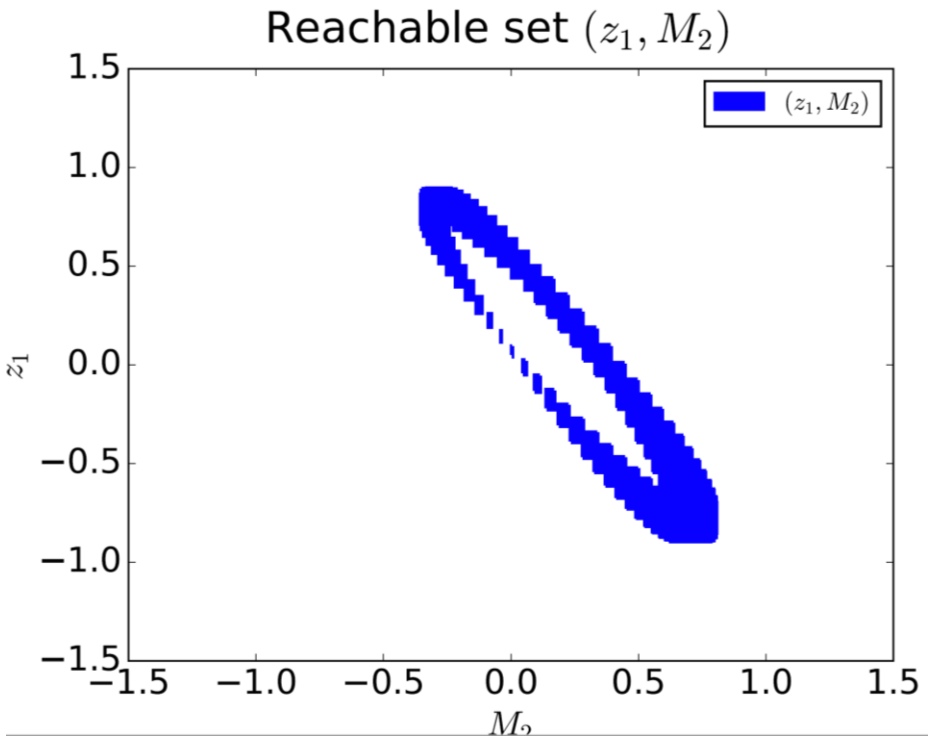
Linear DAE Systems

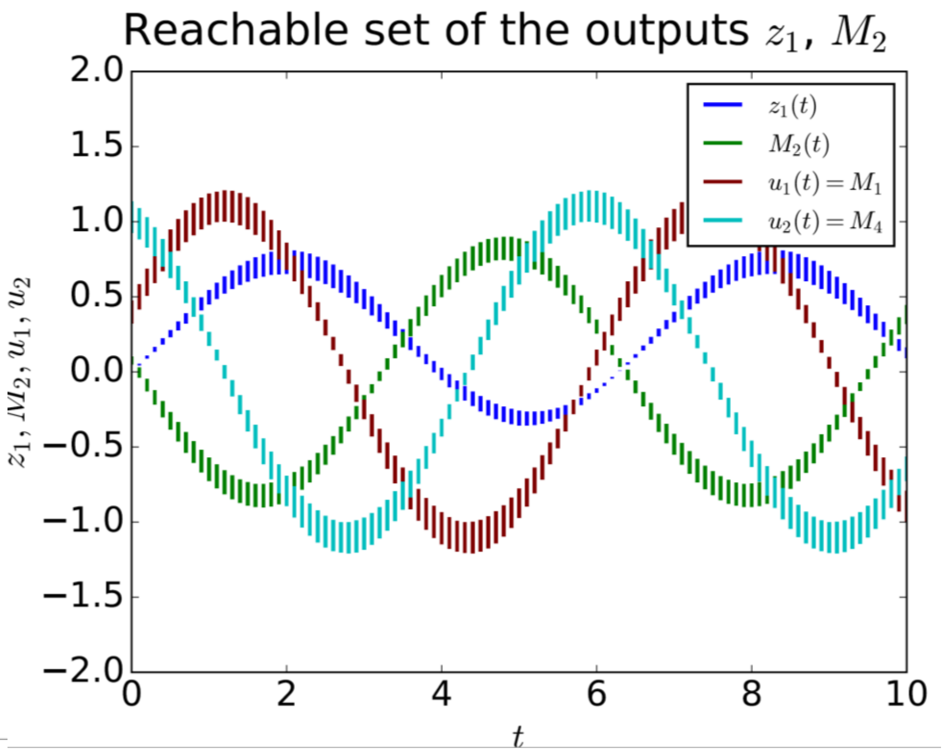
- Index-2 interconnected rotating masses (IRM) system





Reachable sets computed using daev: https://github.com/verivital/daev





Our Approach

1. Decoupling

DAEs	Marz Decoupling	ODEs		AC: Algebraic Constraints
$E\,\dot{x} = Ax + Bu$	$=$	$\dot{x}_1 = N_1 x_1 + Bu$	$+$	$\dot{x}_i = N_i x_i + M_i u$

2. Consistency Checking
   - Define a ***consistent space*** for the initial state and input
   - Guarantee a solution for the DAE system

3. Construct reachable set for the decoupled system
   - Using ***Star-sets*** and Simulation

4. Construct reachable set for original DAE system

5. Perform safety verification & falsification using computed reachable set

Index-1 Decoupling

- **Definition (Tractability index).** Assume that the DAE system $E\dot{x}(t) = Ax(t) + Bu(t)$ is **solvable**, i.e., the matrix pair (E, A) is **regular**. A **matrix chain** is defined by:

  $$E_0 = E, A_0 = A$$

  $$E_{j+1} = E_j - A_jQ_j, A_{j+1} = A_jO_j, j \geq 0, \text{ where } E_jQ_j = 0, Q_j^2 = Q_j, P_j = I_n - Q_j$$

  Where ∃ index μ s.t. E_μ is non-singular and $\forall j \in [0, \mu - 1)$, E_j is singular

  μ is called the **tractability index**

  A matrix pair (E, A) is **regular** if $\det(sE - A) \neq 0$

- **Lemma 1 (Index-1 DAE decoupling).** An index-1 DAE system can be decoupled using the matrix chain defined as follows:

  $$\Delta_1\text{: } \dot{x_1}(t) = N_1x_1(t) + M_1u(t), \text{ ODE subsystems}$$

  $$\Delta_2\text{: } \dot{x_2}(t) = N_2x_1(t) + M_2u(t), \text{ AC subsystems}$$

  $$x(t) = x_1(t) + x_2(t)$$

  $$x_1(t) = P_0x(t), N_1 = P_0E_1^{-1}A_0, M_1 = P_0E_1^{-1}B$$

  $$x_2(t) = Q_0x(t), N_2 = Q_0E_1^{-1}A_0, M_2 = Q_0E_1^{-1}B$$

Index-2 Decoupling

- **Lemma 2 (Index-2 DAE decoupling).** An index-2 DAE system can be decoupled using the matrix chain defined as follows:

$$\Delta_1\colon \dot{x_1}(t) = N_1 x_1(t) + M_1 u(t), \text{ ODE subsystems}$$

$$\Delta_2\colon \dot{x_2}(t) = N_2 x_1(t) + M_2 u(t), \text{ AC subsystems 1}$$

$$\Delta_3\colon \dot{x_3}(t) = N_3 x_1(t) + M_3 u(t) + L_3 \dot{x_2}(t), \text{ AC subsystems 2}$$

$$x(t) = x_1(t) + x_2(t) + x_3(t)$$

$$x_1(t) = P_0 P_1 x(t), N_1 = P_0 P_1 E_2^{-1} A_2, M_1 = P_0 P_1 E_2^{-1} B$$

$$x_2(t) = P_0 Q_1 x(t), N_2 = P_0 Q_1 E_2^{-1} A_2, M_2 = P_0 Q_1 E_2^{-1} B$$

$$x_3(t) = Q_0 x(t), N_3 = Q_0 P_1 E_2^{-1} A_2, M_3 = Q_0 P_1 E_2^{-1} B, L_3 = Q_0 Q_1$$

- Intuition: basically taking derivatives wrt t of the algebraic constraint subsystems to get ODEs
- Scalability issue: increasing dimensionality, more state variables being introduced

Index-3 Decoupling

- **Lemma 3 (Index-3 DAE decoupling).** An index-3 DAE system can be decoupled using the matrix chain defined as follows:

Δ_1: $\dot{x_1}(t) = N_1 x_1(t) + M_1 u(t)$, ODE subsystems

Δ_2: $\dot{x_2}(t) = N_2 x_1(t) + M_2 u(t)$, AC subsystems 1

Δ_3: $\dot{x_3}(t) = N_3 x_1(t) + M_3 u(t) + L_3 \dot{x_2}(t)$, AC subsystems 2

Δ_4: $\dot{x_4}(t) = N_4 x_1(t) + M_4 u(t) + L_4 \dot{x_3}(t) + Z_4 \dot{x_2}(t)$, AC subsystems 3

$$x(t) = x_1(t) + x_2(t) + x_3(t) + x_4(t)$$

$$x_1(t) = P_0 P_1 P_2 x(t), N_1 = P_0 P_1 P_2 E_3^{-1} A_3, M_1 = P_0 P_1 P_2 E_3^{-1} B$$

$$x_2(t) = P_0 P_1 Q_2 x(t), N_2 = P_0 P_1 Q_2 E_3^{-1} A_3, M_2 = P_0 P_1 Q_2 E_3^{-1} B$$

$$x_3(t) = P_0 Q_1 x(t), N_3 = P_0 Q_1 P_2 E_3^{-1} A_3, M_3 = P_0 Q_1 P_2 E_3^{-1} B, L_3 = P_0 Q_1 Q_2$$

$$x_4(t) = Q_0 x(t), N_3 = Q_0 P_1 P_2 E_3^{-1} A_3, M_4 = Q_0 P_1 P_2 E_3^{-1} B, L_4 = Q_0 Q_1, Z_4 = Q_0\ P_1 Q_2$$

Admissible Projectors

- Why is it needed?

Algorithm 3.1 Admissible Projectors Construction

Input: (E, A) % matrices of a DAE system
Output: admissible projectors

procedure INITIALIZATION
  projectors = [ ] % a list of projectors
  $E_0 = E,\ A_0 = A$ and $n = \textit{number of state variables}$
procedure CONSTRUCTION OF ADMISSIBLE PROJECTORS
  if $rank(E_0) == n$:
    exit() % E is nonsingular, thus, the DAE is equivalent to an ODE.
  else:
    $Q_0 = orthogonal\_projector\_on\_Ker(E_0),\ P_0 = I_n - Q_0,\ E_1 = E_0 - A_0Q_0$
    if $rank(E_1) == n$:
      projectors $\leftarrow Q_0$ % the DAE has index-1
    else:
      $Q_1 = orthogonal\_projector\_on\_Ker(E_1),\ P_1 = I_n - Q_1$
      $A_1 = A_0P_0,\ E_2 = E_1 - A_1Q_1$
      if $rank(E_2) == n$:
        $Q_1^* = -Q_1E_2^{-1}A_1$
        projectors $\leftarrow (Q_0,\ Q_1^*)$ % the DAE has index-2
      else:
        $Q_2 = orthogonal\_projector\_on\_Ker(E_2),\ P_2 = I_n - Q_2$
        $A_2 = A_1P_1,\ E_3 = E_2 - A_2Q_2$
        if $rank(E_3) == n$:
          $Q_2' = Q_2E_3^{-1}A_2,\ \ P_2' = I_n - Q_2',\ \ Q_1' = Q_1P_2'E_3^{-1}A_1$
          $E_2' = E_1 - A_1Q_1',\ P_1' = I_n - Q_1',\ \ A_2' = A_1P_1'$
          $Q_2'' = orthogonal\_projector\_on\_Ker(E_2'),\ P_2'' = I_n - Q_2''$
          $E_3'' = E_2' - A_2'Q_2'',\ Q_2^* = -Q_2''(E_3'')^{-1}A_2'$
          projectors $\leftarrow (Q_0,\ Q_1', Q_2^*)$ % the DAE has index-3
        else:
          exit() % the DAE has index lager than 3
  return projectors

Example: Decoupling for IRM System

- Consistent initial set of states

$$Q_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad Q_1 = \begin{bmatrix} \frac{2}{3} & \frac{-2}{3} & 0 & 0 & 0 & 0 \\ \frac{-1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 \\ \frac{2}{3} & \frac{-1}{3} & 0 & 0 & 0 & 0 \\ \frac{-2}{3} & \frac{2}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- IRM can be decoupled into one ODE and two AC subsystems

$$N_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}, \quad N_2 = 0, \quad N_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-2}{3} & \frac{1}{3} \\ 0 & 0 & 0 & 0 & \frac{2}{3} & \frac{-1}{3} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad L_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3} & \frac{-2}{3} & 0 & 0 & 0 & 0 \\ \frac{-2}{3} & \frac{2}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Consistency Checking

- To guarantee a solution for the DAE system, the initial states and inputs must satisfy the following conditions

Index-1 DAE: $x_2(0) = N_2 x_1(0) + M_2 u(0)$

Index-2 DAE: $x_2(0) = N_2 x_1(0) + M_2 u(0)$

$$x_3(0) = N_3 x_1(0) + M_3 u(0) + L_3\, \dot{x}_2(0)$$

Index-3 DAE: $x_2(0) = N_2 x_1(0) + M_2 u(0)$

$$x_3(0) = N_3 x_1(0) + M_3 u(0) + L_3 \dot{x}_2(0)$$

$$x_4(0) = N_4 x_1(0) + M_4 u(0) + L_4 \dot{x}_3(0) + Z_4 \dot{x}_2(0)$$

- Where input $u(t)$ is **smooth** such that: $\dot{u}(t) = A_u u(t), u(0) = u_0 \in \mathrm{U}_0$
  - $A_u \in \mathrm{R}^{m \times n}$: user-defined input matrix
  - U_0: the set of initial inputs

Consistency Checking

- **Definition (Consistent space).** Consider the DAE system Δ: $E\dot{x}(t) = Ax(t) + Bu(t)$, by letting $u(t) = 0$, we define a **consistent matrix** Γ as:

$$\text{Index-1 } \Delta: \Gamma = Q_0 - N_2 P$$

$$\text{Index-2 } \Delta: \begin{bmatrix} P_0 Q_1 - N_2 P_0 P_1 \\ Q_0 - (N_3 + L_3 N_2 N_1) P_0 P_1 \end{bmatrix}$$

$$\text{Index-2 } \Delta: \begin{bmatrix} P_0 P_1 Q_2 - N_2 P_0 P_1 P_2 \\ P_0 Q_1 - (N_3 + L_3 N_2 N_1) P_0 P_1 P_2 \\ Q_0 - \left[N_4 + L_4\left(N_3 N_1 + L_3 N_2 N_1^2\right) + Z_4 N_2 N_1\right] P_0 P_1 P_2 \end{bmatrix}$$

Then, $Ker(\Gamma)$ is the **consistent space** of the system Δ, also denotes null space of the matrix Γ

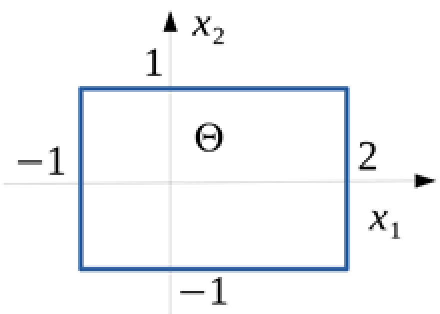
- An initial state x_0 is **consistent** if it is in the consistent space, i.e., $\Gamma x_0 = 0$

Reachability Analysis

- **Definition (Modified Star-Set).** A **modified star set** Θ is a tuple $\langle V, P \rangle$, where $V = [v_1, v_2, \ldots, v_k] \in \mathbb{R}^{n\times k}$ is a **star basis matrix** and P is a **linear predicate**. The set of states represented by the star is given by:

$$[\![\Theta]\!] = \{x \mid x = \Sigma_{i=1}^{k}(\alpha_i v_i) = V \times \alpha, P(\alpha) \triangleq C\alpha \le d\}$$

where, $\alpha = [\alpha_1 = 1, \alpha_2, \ldots, \alpha_k]^T$, $C \in \mathbb{R}^{p\times k}$, $P \in \mathbb{R}^p$, and p is the number of linear constraints.



$$V = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \qquad d = \begin{bmatrix} 1 \\ -1 \\ 2 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

[Stanley Bak, Hoang-Dung Tran, Taylor T. Johnson, "Numerical Verification of Affine Systems with Up to a Billion Dimensions“, HSCC’19]

[Hoang-Dung Tran, Patrick Musau, Diego Manzanas Lopez, Xiaodong Yang, Luan Viet Nguyen, Weiming Xiang, Taylor T. Johnson, "Star-Based Reachability Analysis for Deep Neural Networks", FM’19]

Reachability Analysis

- **Lemma 4 (Reachable Set Construction).** Given an autonomous DAE system $E\dot{x}(t) = Ax(t) + Bu(t)$ where $u(t) = 0$ and a consistent initial set of states $\Theta(0) = \langle V(0), P\rangle$, let Θ_1(t) be the **reachable set** at time t of the corresponding ODE subsystem after decoupling. Then, the **reachable set at time** $\boldsymbol{t}$ of the system is given by $\Theta(\mathrm{t}) = \langle V(t) = \Psi V_1(t), P\rangle$, where Ψ is a reachable set projector defined as

$$\text{Index-1: } \Psi = I_n + N_2$$

$$\text{Index-2: } \Psi = I_n + N_2 + N_3 + L_3 N_2 N_1$$

$$\text{Index-3: } \Psi = I_n + N_2 + N_3 + L_3 N_2 N_1 + L_4 N_3 N_1 + L_4 L_3 N_2 N_1^2 + Z_4 N_2 N_1$$

- Recall N_i, L_j, Z_k are from Marz decoupling discussed earlier

Reachability Analysis

Algorithm 5.1 Reachable set computation

Inputs: Matrices of an autonomous DAE system (E, A), initial set of states $\Theta(0) = \langle V(0), P \rangle$, time step h, number of steps N.

Output: Reachable set % A list of stars

procedure INITIALIZATION
  $ListOfStars = [\,]$
  Decoupling the system
  Obtain consistent space $Ker(\Gamma)$
  If $V(0) \notin Ker(\Gamma)$: exit() % inconsistent initial set of states
  Else: Obtain initial set of states for ODE subsystem:
    $\Theta_1(0) = \langle V_1(0), P \rangle$, $V_1(0) = [v_1^1(0) \ \cdots \ v_k^1(0)]$

procedure REACHABLE SET CONSTRUCTION
  for $j = 0, 1, 2, \cdots, N$:
    for $i = 1, 2, \cdots, k$:
      Compute $v_i^1(jh) = e^{N_1 jh} v_i^1(0)$ % using ODE solvers
    Construct $V_1(jh) = [v_1^1(jh) \ v_2^1(jh) \ \cdots \ v_k^1(jh)]$
    Compute $V(jh)$ from $V_1(jh)$
    Construct $\Theta(jh) = \langle V(jh), P \rangle$
    $ListOfStars \leftarrow \Theta(jh)$
  return $ListOfStars$

Bounded-time safety verification/falsification

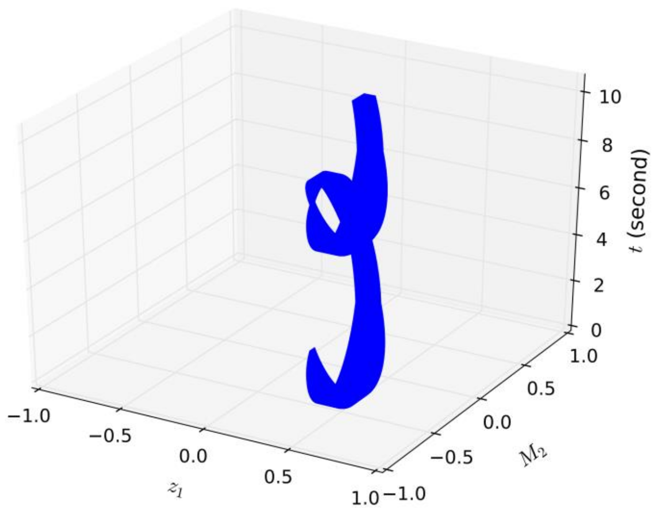
Algorithm 5.2 Bounded-time safety verification/falsification

Inputs: $Reachable\_Set$ % a list of stars; $Unsafe(\Delta) \triangleq Gx \leq f$ % the unsafe set
Output: $Safe/Unsafe$ and $Unsafe\_Trace$

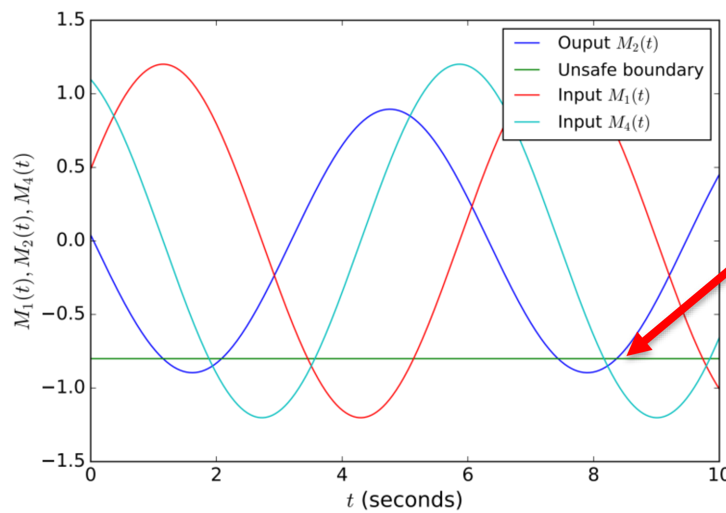
procedure INITIALIZATION
    N = number of stars in the reachable set
    $Status = Safe$
    $Unsafe\_Trace = [\,]$
procedure VERIFICATION/FALSIFICATION
    for $j = 1, 2, \cdots, N$:
        $\Theta_j = Reachable\_Set[j] = \langle V_j, P \rangle$, $P \triangleq C\alpha \leq d$
        Construct $\bar{P} \triangleq \begin{bmatrix} GV_j \\ C \end{bmatrix} \alpha \leq \begin{bmatrix} f \\ d \end{bmatrix}$
        If $\bar{P}$ is feasible:
            $Status = Unsafe$, get $\alpha_{feasible}$, exit()
    If $Status = Unsafe$:
        for $j = 1, 2, \cdots, N$:
            Compute $x_j = V_j \alpha_{feasible}$
            $Unsafe\_Trace \leftarrow x_j$
    return $Status$, $Unsafe\_Trace$

Reachability Analysis for IRM System

- Sinusoid input $\begin{bmatrix}\dot{M}_1(t)\\ \dot{M}_4(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}\begin{bmatrix}M_1(t)\\ M_4(t)\end{bmatrix}, u(0) = \begin{bmatrix}M_1(0)\\ M_4(0)\end{bmatrix} \in U$
- A consistent initial set of states $V(0) = \begin{bmatrix}0 & 0\\ 0 & 0\\ 0.513 & 0\\ -0.513 & 0\\ -0.616 & 0.447\\ 0.308 & 0.894\end{bmatrix}, P(\alpha) \triangleq C\alpha \le d,\ C = \begin{bmatrix}1 & 0\\ -1 & 0\\ 0 & 1\\ 0 & -1\end{bmatrix}, d = \begin{bmatrix}0.2\\ -0.1\\ 1.2\\ -1.0\end{bmatrix}$
- Safety verification w.r.t unsafe specification $M_2(t) \le -0.8$



Reachable set



Violation

An unsafe trace

Scalability Performance

Table 1. Verification results for all benchmarks using Daev.

Benchmarks	n	Index	Unsafe Set	Result	V-T(s)
RL network [24]	3	2	$x_1 \leq -0.2 \wedge x_2 \leq -0.1$	unsafe	0.184
			$x_1 \geq 0.2$	safe	0.44
RLC circuit [12]	4	1	$x_1 + x_3 \geq 0.2$	unsafe	0.224
			$x_4 \leq -0.3$	safe	1.37
Interconnected rotating mass [30]	4	2	$x_3 \leq -0.9$	unsafe	0.37
			$x_4 \leq -1.0$	safe	0.114
Generator [20]	9	3	$x_9 \geq 0.01$	unsafe	0.4
			$x_1 \geq 1.0$	safe	0.684
Damped-mass spring [27]	11	3	$x_3 \leq 1 \wedge x_8 \leq 1.5$	safe	1.06
			$x_8 \leq -0.2$	unsafe	1.08
PEEC [9]	480	2	$x_{478} \geq 0.05$	safe	28.84
			$x_{478} \geq 0.01$	unsafe	28.25
MNA-1 [9]	578	2	$x_1 \geq -0.001$	safe	192.7
			$x_1 \geq -0.0015$	unsafe	202.6
MNA-4 [9]	980	3	$x_2 \geq 0.0005$	safe	1858.4
			$x_2 \geq 0.0002$	unsafe	1836.04
Stokes-equation [27]	4880	2	$v_x^c + v_y^c \leq -0.04$	unsafe	3502.3
			$v_x^c \geq 0.2$	safe	3532.3

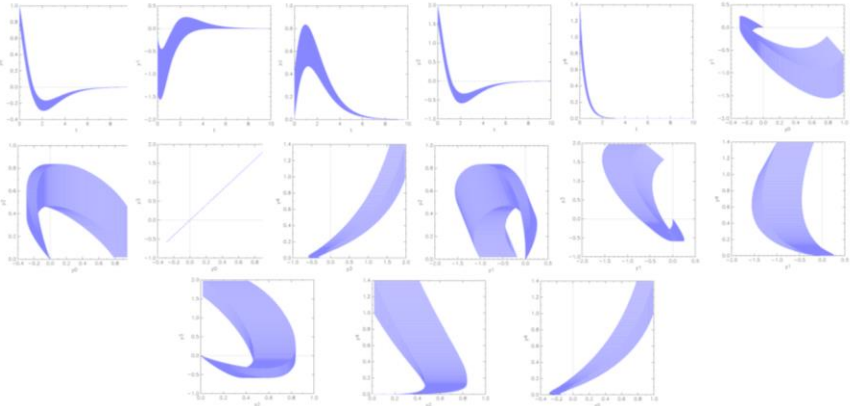
Benchmark details: ARCH'18 paper, "Linear Differential-Algebraic Equations"

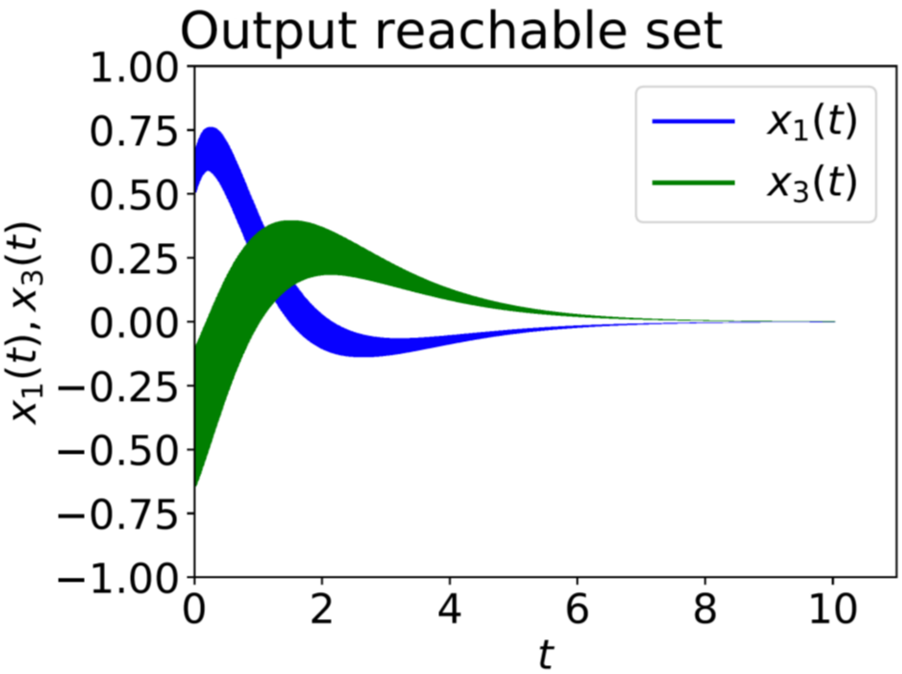
Takeaways:

- Daev is scalable in verifying large DAE systems (≥ 1K state variables) where other over-approximation approaches not applicable
- Daev can produce unsafe traces
- Available: https://github.com/verivital/daev

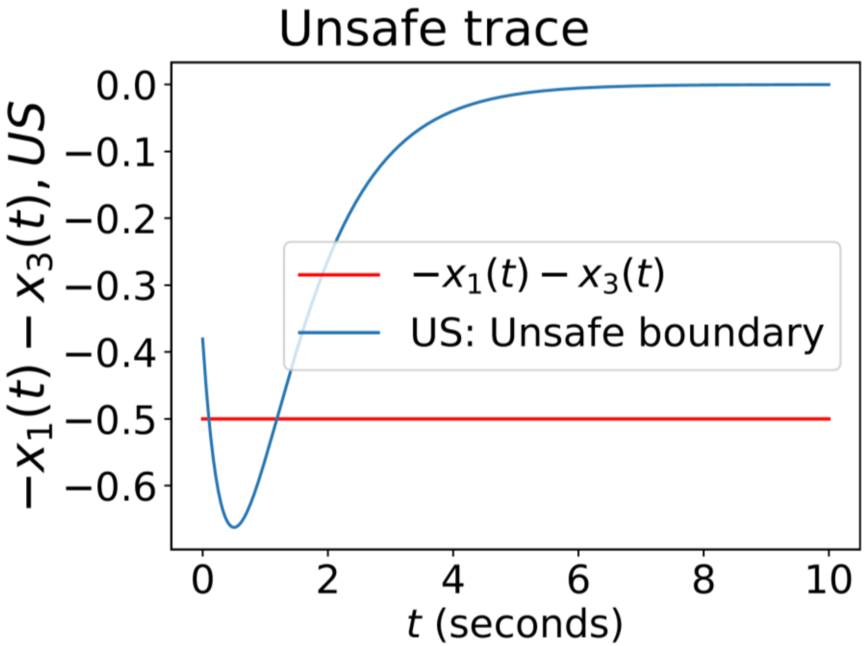
  https://github.com/verivital/daev/releases/tag/formats2019

RLC Circuit



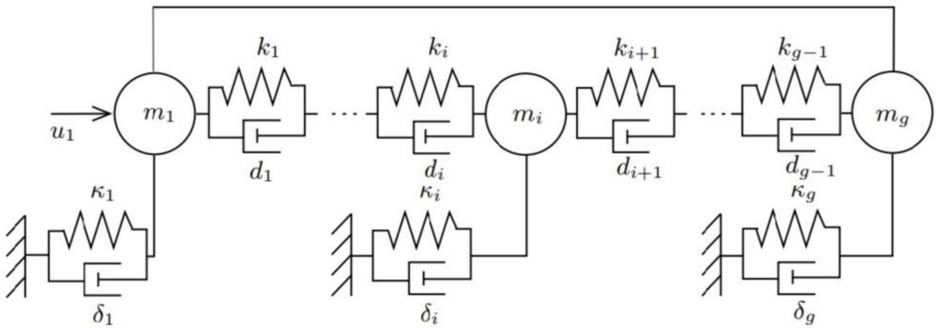


Output reachable set
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10
t
$x_1(t), x_3(t)$
$x_1(t)$
$x_3(t)$



Unsafe trace
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
0
2
4
6
8
10
t (seconds)
$-x_1(t) - x_3(t)$, US
$-x_1(t) - x_3(t)$
US: Unsafe boundary

Damped Mass Spring



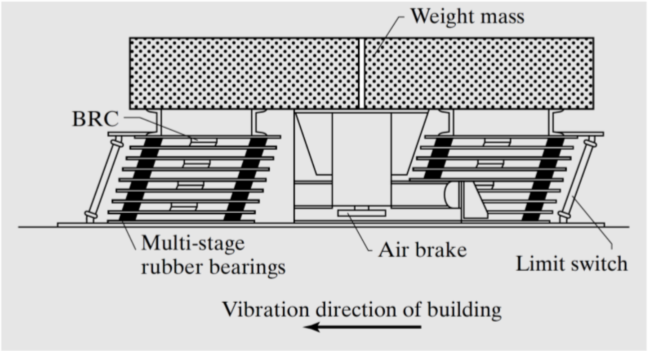
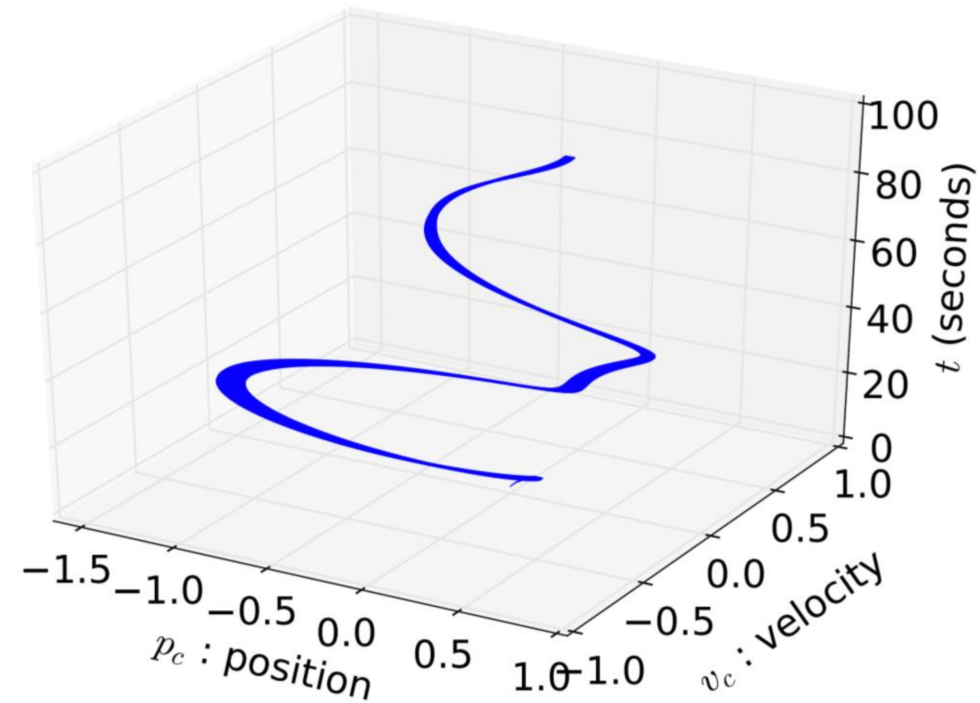
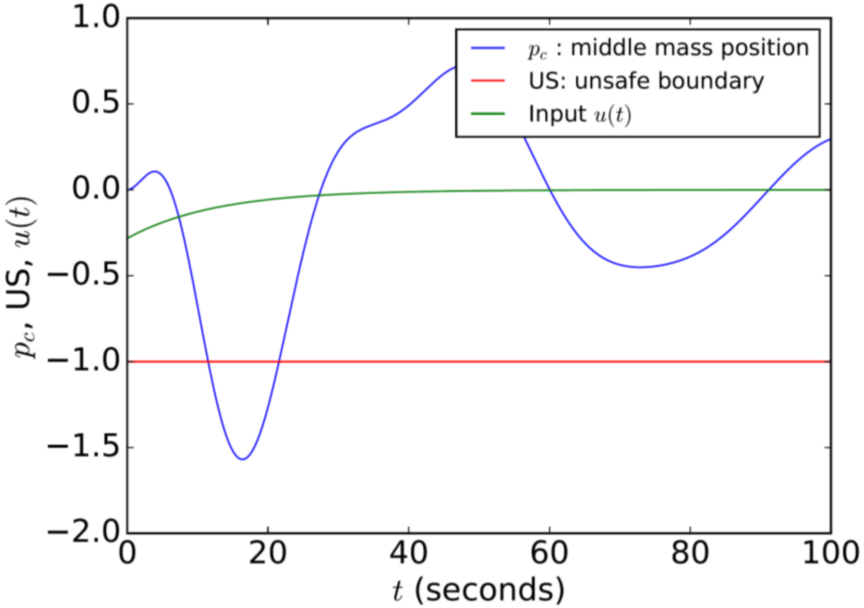


FIGURE 4.4: Tuned mass damper with spring and damper assemblage.

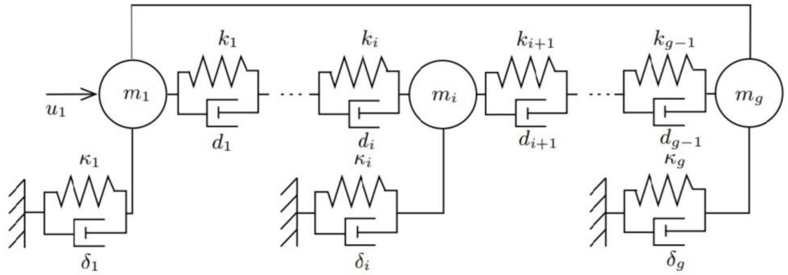
Reachable Set (p_c, v_c) vs. time t



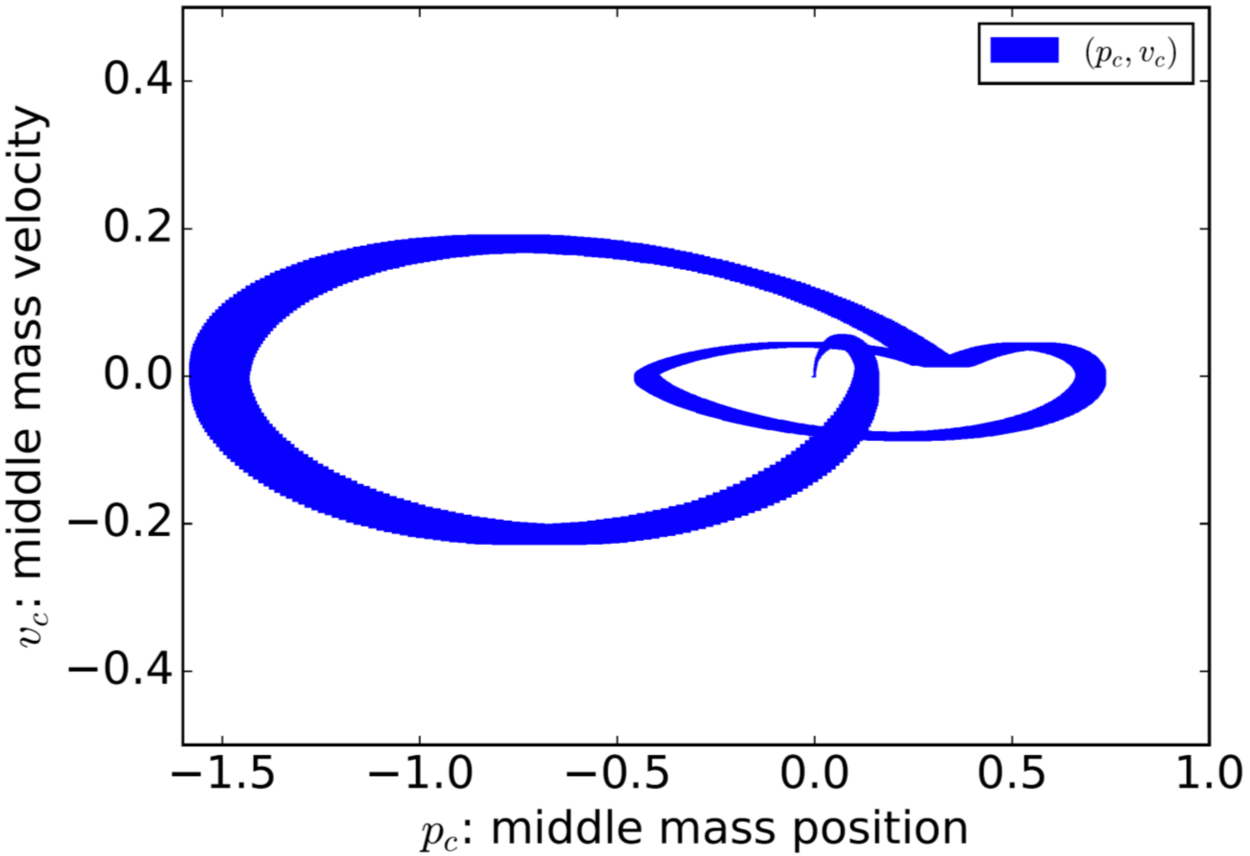
Unsafe trace



Damped Mass Spring

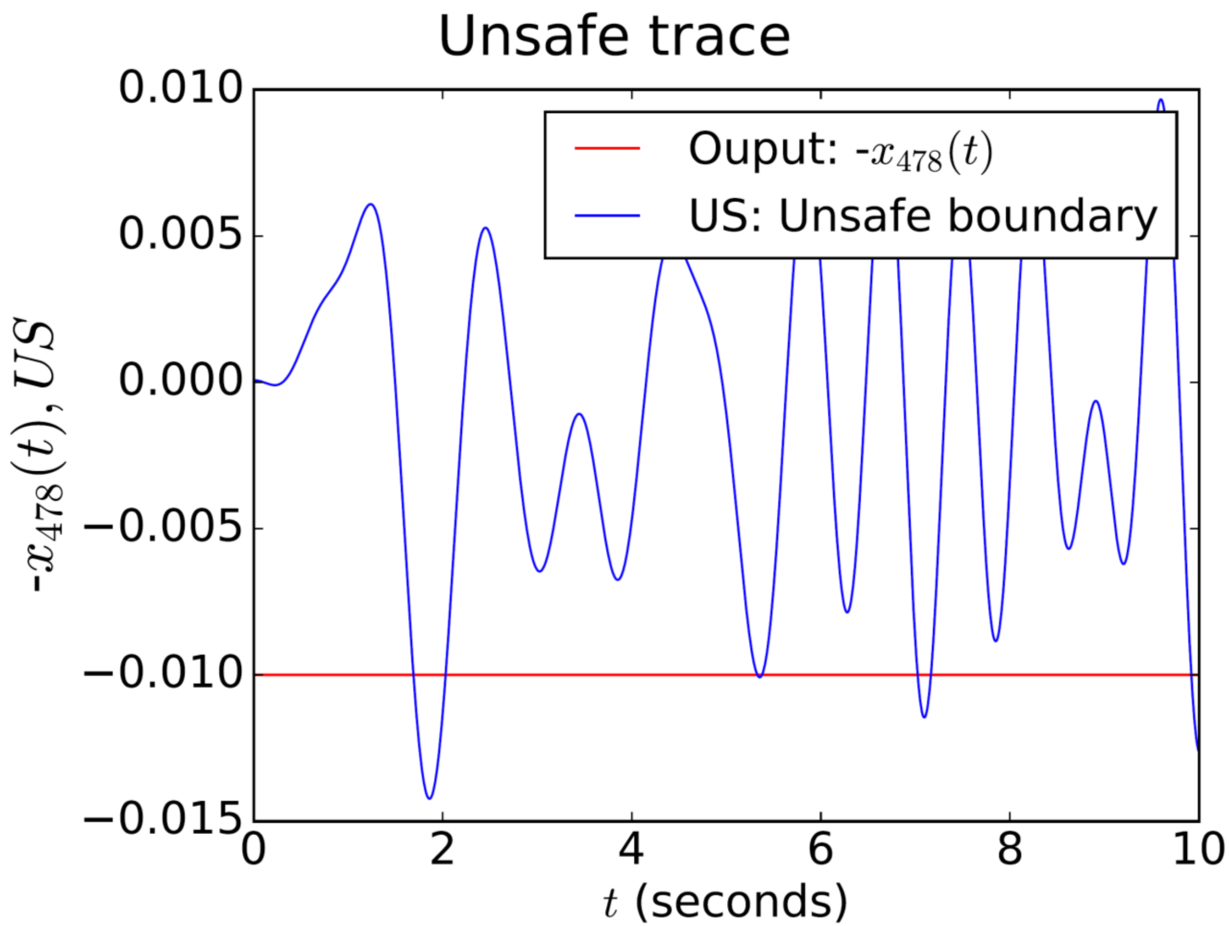


Reachable set (p_c, v_c) in $[0, 100]$ seconds

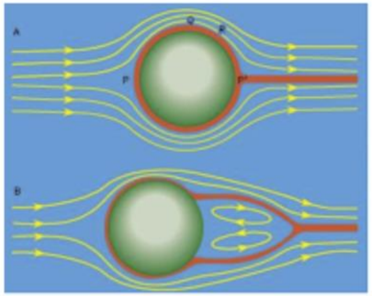


Partial Element Equivalent Circuit (PEEC)

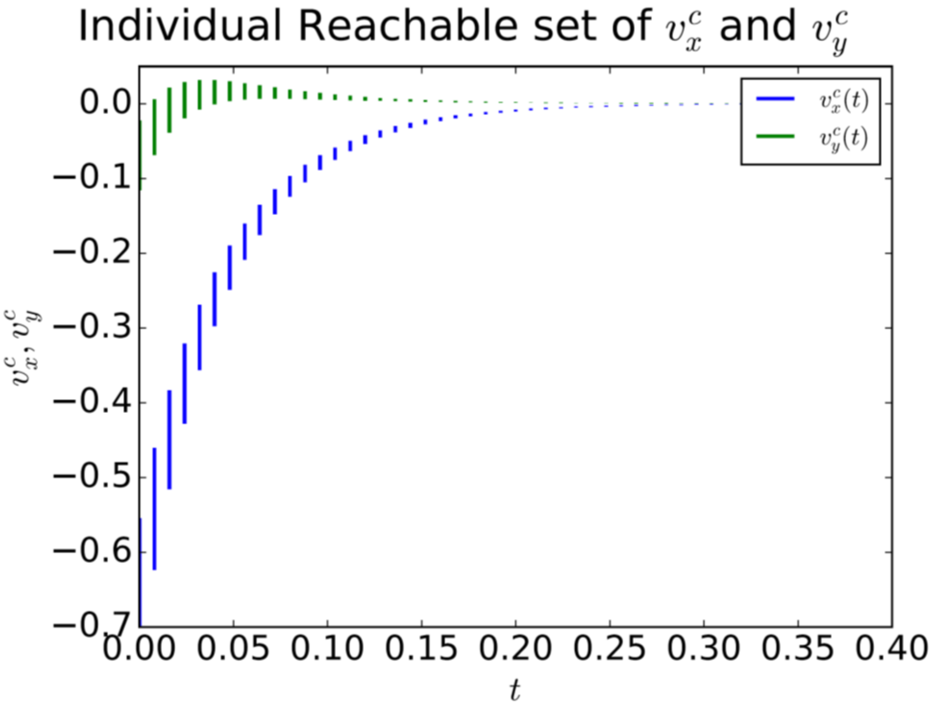
- Electromagnetics application: RF engineering

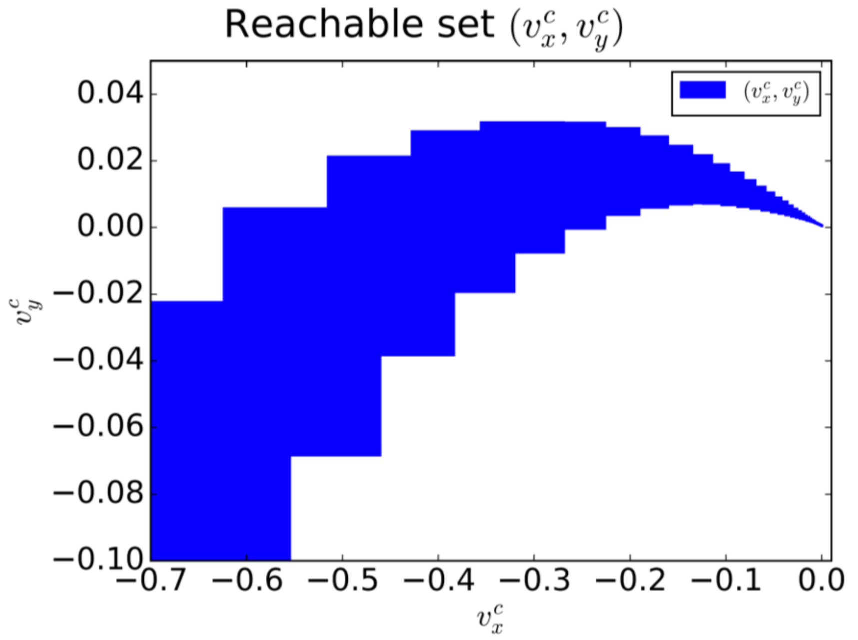


Stokes

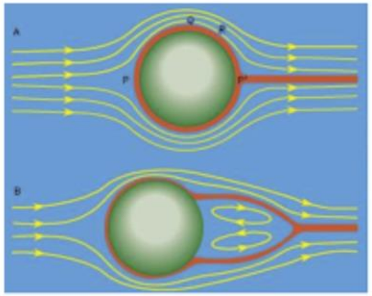


$$\frac{\partial v}{\partial t} = \Delta v - \nabla \rho + f, \text{ in } \Omega \times (0, T)$$
$$\nabla v = 0, \text{ in } \Omega \times (0, T),$$

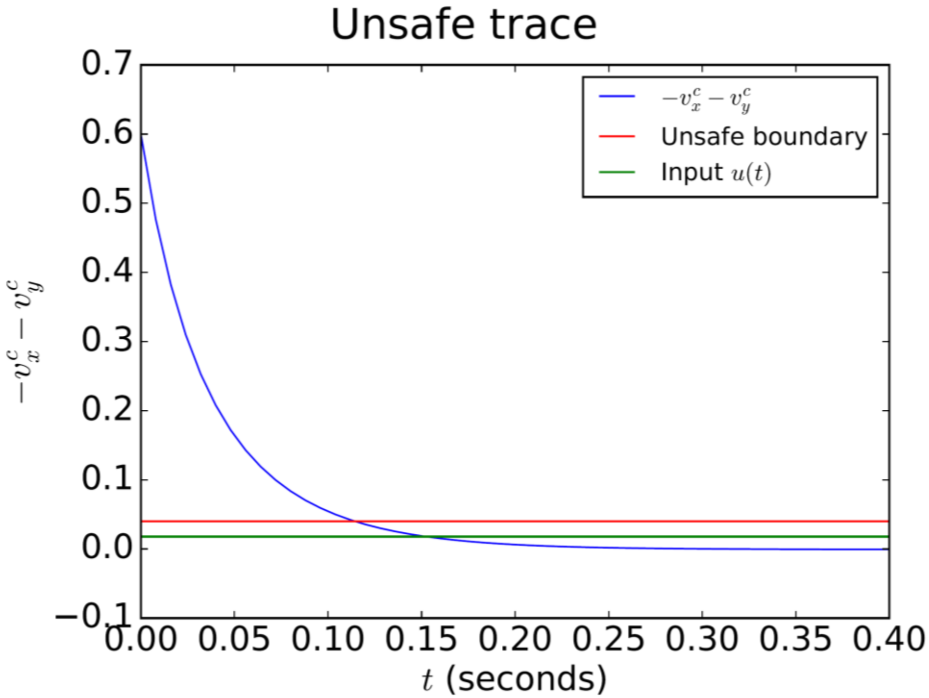


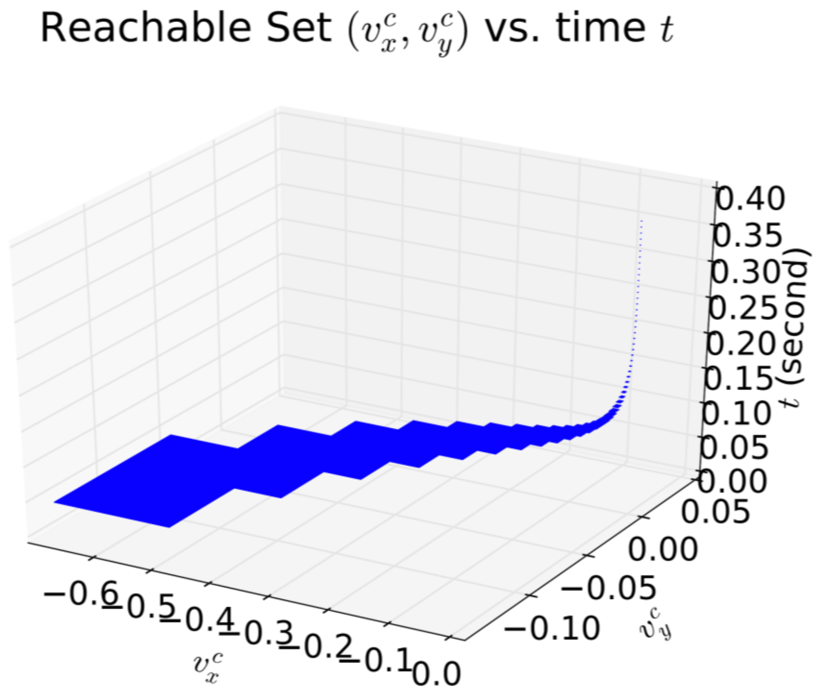


Stokes



$$\frac{\partial v}{\partial t} = \Delta v - \nabla \rho + f, \text{ in } \Omega \times (0, T)$$
$$\nabla v = 0, \text{ in } \Omega \times (0, T),$$





Scalability Analysis

- Stokes-equation PDE

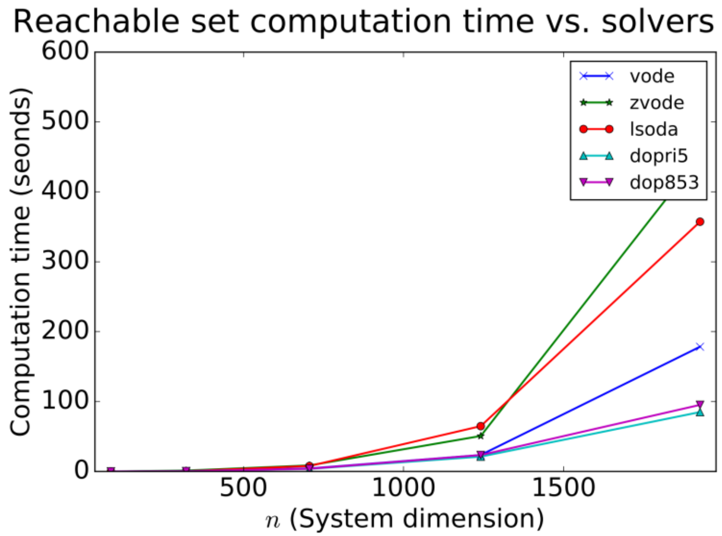
$$\frac{\partial v}{\partial t} = \Delta v - \nabla \rho + f, \quad \text{in } \Omega \times (0,T),$$
$$\nabla v = 0, \quad \text{in } \Omega \times (0,T),$$

Boundary conditions => algebraic constraints (Finite-difference method based on marker-and-cell [MAC])

Table 2. Verification time of Stokes-equation with different dimensions n.

n	86	321	706	1241	1926	2761
D-T	$0.012s$	$0.63s$	$6.32s$	$40.38s$	$155.32s$	$466.38s$
RSC-T	$0.019s$	$0.37s$	$2.98s$	$19.29s$	$68.15s$	$200.89s$
CS-T	$0.0017s$	$0.0014s$	$0.0015s$	$0.0017s$	$0.0018s$	$0.002s$
V-T	$0.0327s$	$1.0014s$	$9.3015s$	$59.6717s$	$223.4718s$	$667.272s$

D-T: decoupling time,
RSC-T: reachable set computation time
CS-T: checking safety time
V-T: verification time (overall total time sum)



Takeaway:

- Decoupling and reachable set computation times dominate the time for verification process
- Time for checking safety is almost unchanged and very small
- *vode*, *dopri5*, and *dop853* solvers should be used for large DAE systems

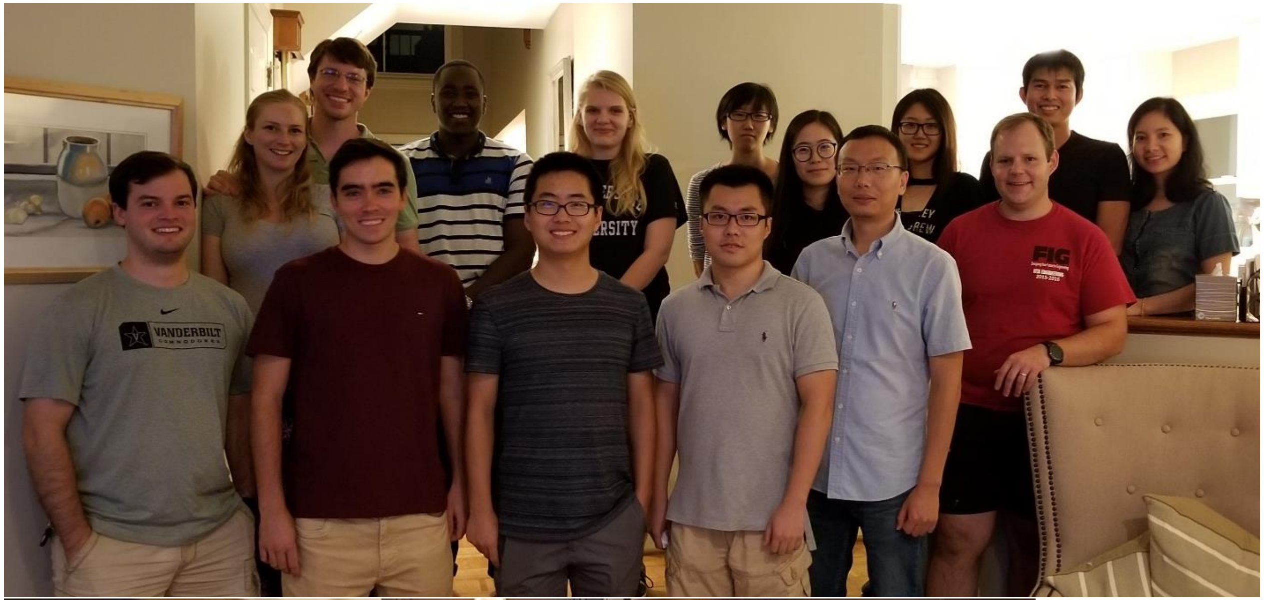
Conclusion and future works

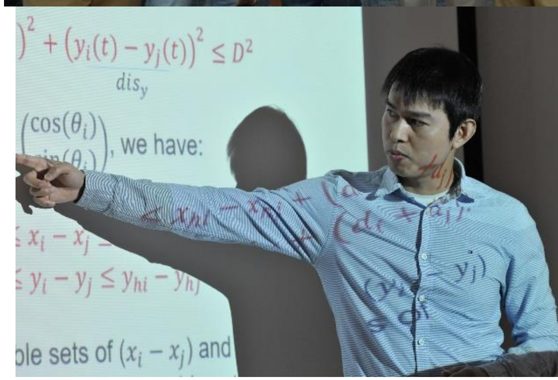
- **Conclusion**
  - ✓ A simulation-based reachability analysis for high-index, linear DAE systems
  - ✓ Based on the effective combination of a **decoupling method** and a reachable set computation using **star-sets**
  - ✓ Design and implementation of the approach in a Python toolbox, called **Daev:** https://github.com/verivital/daev/
  - ✓ Applied to verify/falsify high-index linear DAE systems
  - ✓ Approach can deal with DAE systems with up to thousands of state variables

- **Future Work**
  - ✓ Enhance the time performance and the scalability of our approach
  - ✓ Apply to verify million-dimensional DAE systems
  - ✓ DAEs with hybrid/switching behavior (time or state-dependent)

Thank You


Controlling Groups of Swarm Robots

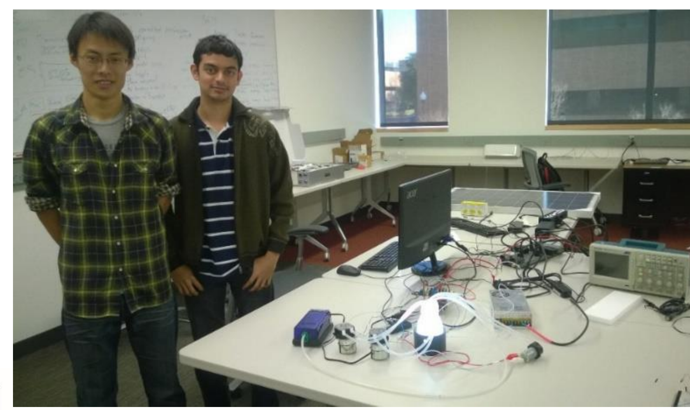

VANDERBILT


$+ (y_i(t) - y_j(t))^2 \le D^2$
dis_y
we have:
$\le y_i - y_j \le y_{hi} - y_{hj}$
ble sets of $(x_i - x_j)$ and


Montreal
CPS-IoT Week 2019
HSCC ICCPS IPSN RTAS IoTDI
April 15-18, 2019, Montreal, Canada
Sponsors
IEEE
NSF
SIGBED
intel
arm
BOSCH
IBM
MathWorks
DENSO
CSS
SeRo
TOYOTA

Thank You! Questions?

- **Students**
  - **VU EECS:** Hoang-Dung Tran (PhD), Nate Hamilton (PhD), Ayana Wild (PhD), Patrick Musau (PhD), Xiaodong Yang (PhD), Ran Hao (PhD), Tianshu Bao (PhD), Diego Manzanas (PhD), Weiming Xiang (Postdoc), Joel Rosenfeld (Postdoc)
  - **UTA CSE:**Shafiul Chowdhury (PhD)
  - **UTA Alumni**: Luan Viet Nguyen (PhD), Omar Beg (PhD), Nathan Hervey (MS), Ruoshi Zhang (MS), Shweta Hardas (MS), Randy Long (MS), Rahul (MS), Amol (MS)
- **Recent Collaborators**
  - Vanderbilt: Gabor Karsai, Xenofon Koutsoukos, Janos Sztipanovits, …
  - UTA: Ali Davoudi, Christoph Csallner, Matt Wright, Steve Mattingly, Colleen Casey
  - Illinois: Sayan Mitra, Marco Caccamo Lui Sha, Amy LaViers
  - AFRL: Stanley Bak and Steven Drager
  - Toyota: Jim Kapinski, Xiaoqing Jin, Jyo Deshmukh, Ken Butts, Issac Ito
  - Waterloo: Sebastian Fischmeister
  - Toronto: Andreas Veneris
  - ANU: Sergiy Bogomolov
  - UTSW: Ian White, Victor Salinas, Rama Ranganathan



Taylor T. Johnson
http://www.TaylorTJohnson.com
Taylor.Johnson@vanderbilt.edu
http://www.verivital.com

VANDERBILT UNIVERSITY®