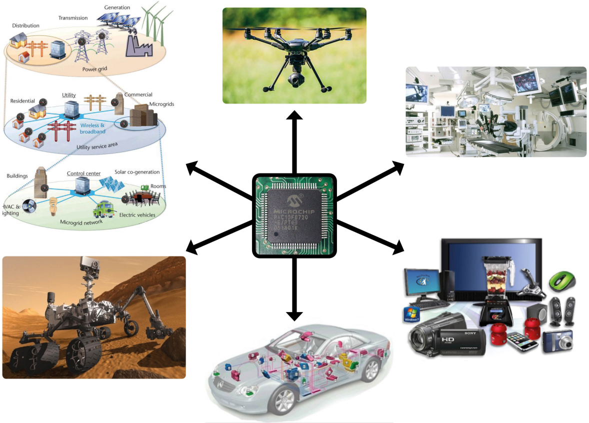
Automated Formal Synthesis of Digital Controllers for State-Space Physical Plants
CAV 2017

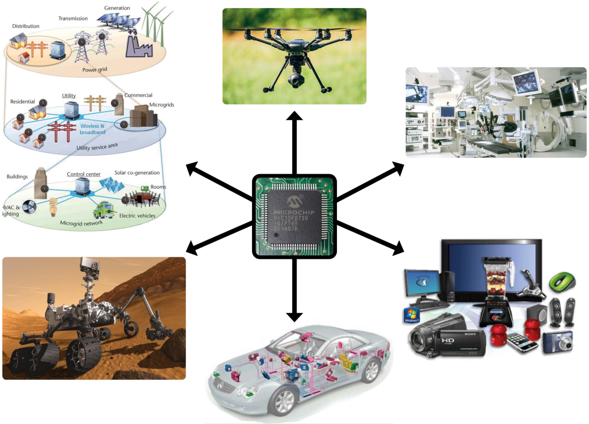
Alessandro Abate, Iury Bessa, Dario Cattaruzza,
Lucas Cordeiro, **Cristina David**, Pascal Kesseli,
Daniel Kroening and Elizabeth Polgreen

Diffblue Ltd.,
University of Oxford,
Federal University of Amazonas

Motivation

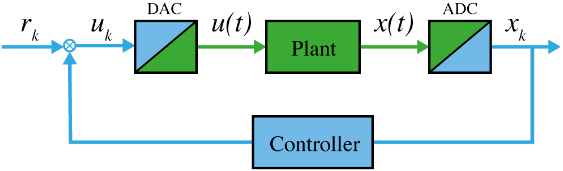

Generation
Transmission
Distribution
Power grid
Residential
Utility
Commercial
Microgrids
Wireless & broadband
Utility service area
Buildings
Control center
Solar co-generation
Rooms
Electric vehicles
Microgrid network
MICROCHIP

Motivation



Automatically synthesise feedback digital controllers that ensure stability and safety

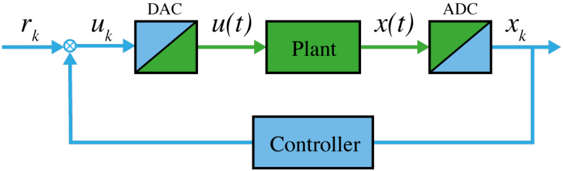
State-feedback architecture



Continuous-discrete system

- Plant: $\dot{x}(t) = Ax(t) + Bu(t), \quad t \in \mathbb{R}_0^+, \; x(0) = initial \; state$
- State-feedback controller: $u_k = r_k - Cx_k$

State-feedback architecture



Continuous-discrete system

- Plant: $\dot{x}(t) = Ax(t) + Bu(t), \quad t \in \mathbb{R}_0^+, \; x(0) = \textit{initial state}$
- State-feedback controller: $u_k = -Cx_k$

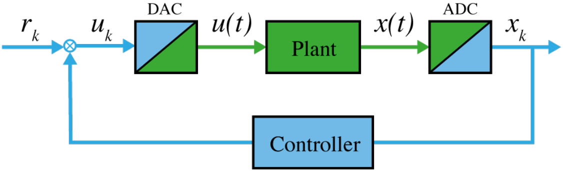
Our approach

1. Translate to a single digital domain

2. Evaluate sources of numerical error

3. Synthesise a controller C that makes the system stable and safe

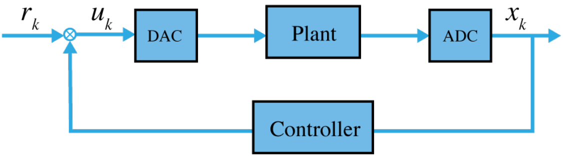
State-feedback architecture



Continuous-discrete system

- Plant: $\dot{x}(t) = Ax(t) + Bu(t), \quad t \in \mathbb{R}_0^+, \; x(0) = \textit{initial state}$
- State-feedback controller: $u_k = -Cx_k$

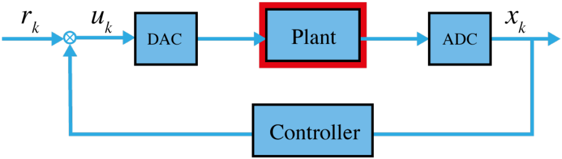
State-feedback architecture



Time and value domain discretization

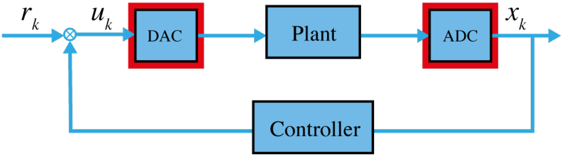
- Plant: $x_{k+1} = Ax_k + Bu_k, \quad k \in N, \ x_0 = initial \ state$
- State-feedback controller: $u_k = -Cx_k$
- Finite precision

Numerical errors



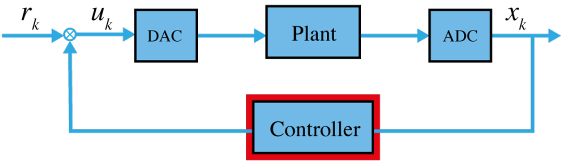
- **Truncation and rounding on the plant**

Numerical errors



- Truncation and rounding on the plant
- Truncation on the converters

Numerical errors



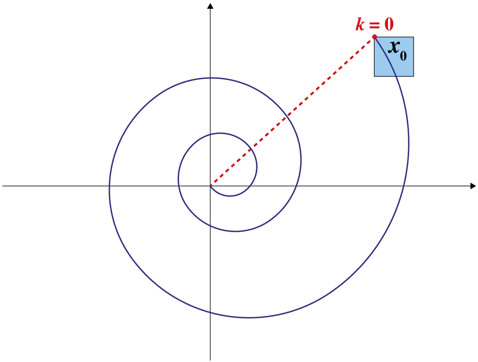
- **Truncation and rounding on the plant**
- **Truncation on the converters**
- **Rounding on the controller**

Stability

A system $x_{k+1} = Ax_k + Bu_k, u_k = -Cx_k$ is asymptotically stable if its executions converge to an equilibrium point.

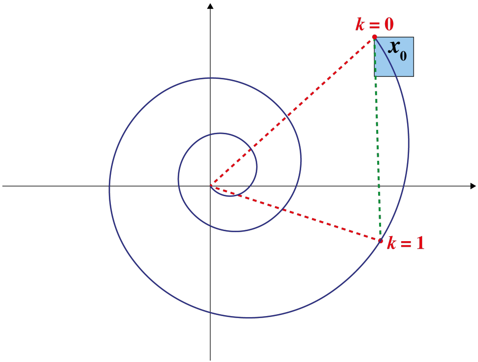
Stability

A system $x_{k+1} = Ax_k + Bu_k, u_k = -Cx_k$ is asymptotically stable if its executions converge to an equilibrium point.



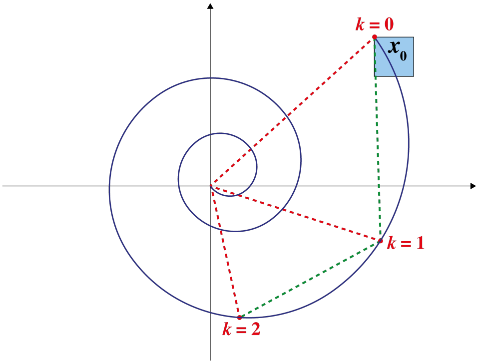
Stability

A system $x_{k+1} = Ax_k + Bu_k, u_k = -Cx_k$ is asymptotically stable if its executions converge to an equilibrium point.



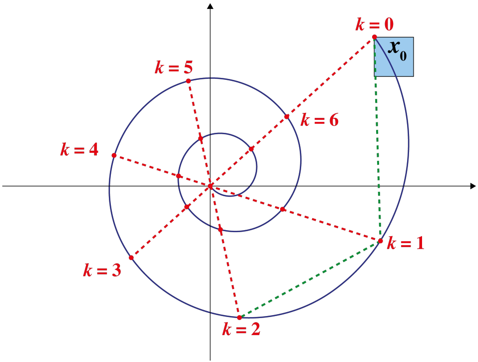
Stability

A system $x_{k+1} = Ax_k + Bu_k, u_k = -Cx_k$ **is asymptotically stable if its executions converge to an equilibrium point.**



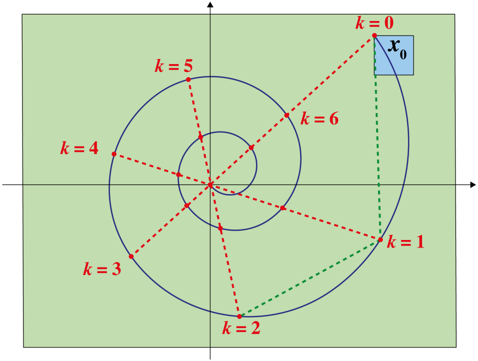
Stability

A system $x_{k+1} = Ax_k + Bu_k, u_k = -Cx_k$ is asymptotically stable if its executions converge to an equilibrium point.

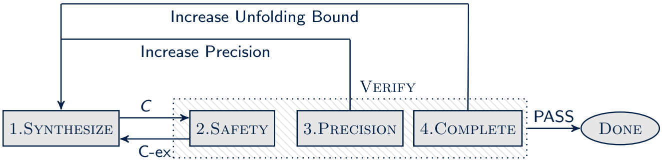


Stability and safety

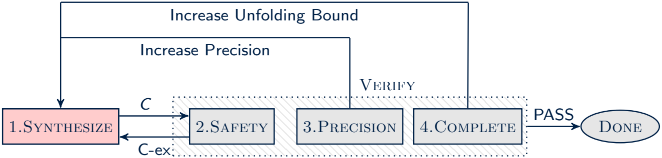
A system $x_{k+1} = Ax_k + Bu_k, u_k = -Cx_k$ **is asymptotically stable if its executions converge to an equilibrium point.**



Controller synthesis



Controller synthesis

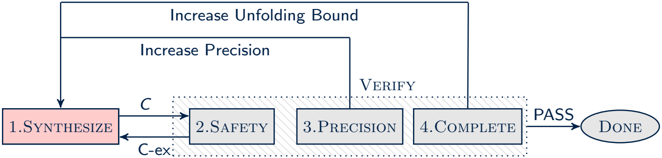


Find a controller for given $\{x_0\}$ and $k{=}6$ such that the system is stable and safe

SYNTHESIZE

Input: x_0, k.
Output: C.
C=nondet;
assume(**STABLE**(A,B,C));
assume(**SAFE**(x_0));
$i = 0$;
while $i < k$ **do**
  $x_{i+1} = x_i(A - BC)$
  assume(**SAFE**(x_{i+1}));
  $i = i + 1$;
end while
assert(false);

Controller synthesis

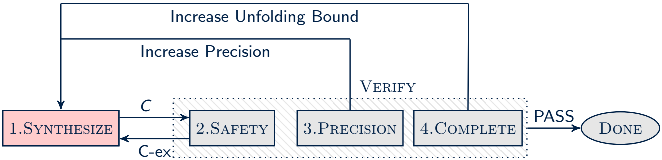


Find a controller for given $\{x_0\}$ and k=6 such that the system is stable and safe

SYNTHESIZE

Input: x_0, k.
Output: C.
C=nondet;
assume(**STABLE**(A,B,C));
assume(**SAFE**(x_0));
$i = 0$;
while $i < k$ **do**
  $x_{i+1} = x_i(A - BC)$
  assume(**SAFE**(x_{i+1}));
  $i = i + 1$;
end while
assert(false);

Controller synthesis

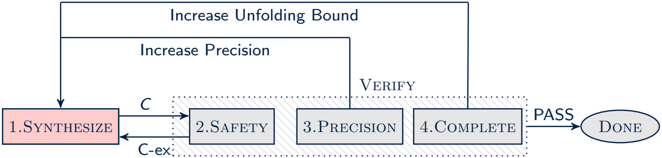


Find a controller for given $\{x_0\}$ and k=6 such that the system is stable and safe

SYNTHESIZE

Input: x_0, k.
Output: C.
C=nondet;
assume(**STABLE**(A,B,C));
assume(**SAFE**(x_0));
$i = 0$;
while $i < k$ **do**
    $x_{i+1} = x_i(A - BC)$
    assume(**SAFE**(x_{i+1}));
    $i = i + 1$;
end while
assert(false);

Controller synthesis

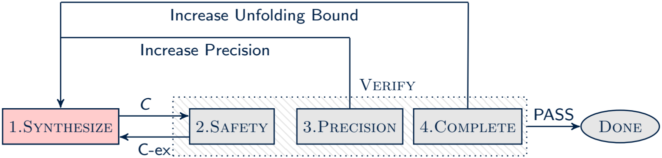


Find a controller for given $\{x_0\}$ and k=6 such that the system is stable and safe

SYNTHESIZE

Input: x_0, k.
Output: C.
C=nondet;
assume(**STABLE**(A,B,C));
assume(**SAFE**(x_0));
$i = 0$;
while $i < k$ **do**
  $x_{i+1} = x_i(A - BC)$
  assume(**SAFE**(x_{i+1}));
  $i = i + 1$;
end while
assert(false);

Controller synthesis

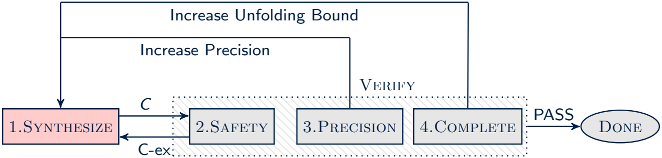


Find a controller for given $\{x_0\}$ and k=6 such that the system is stable and safe

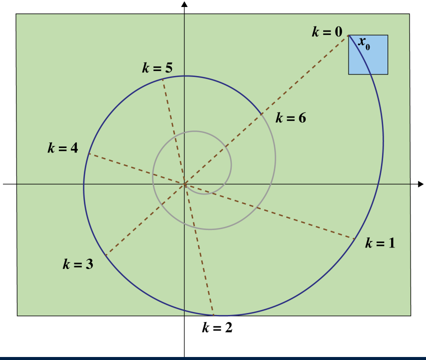
SYNTHESIZE

Input: x_0, k.
Output: C.
C=nondet;
assume(**STABLE**(A,B,C));
assume(**SAFE**(x_0));
$i = 0$;
while $i < k$ **do**
  $x_{i+1} = x_i(A - BC)$
  assume(**SAFE**(x_{i+1}));
  $i = i + 1$;
end while
assert(false);

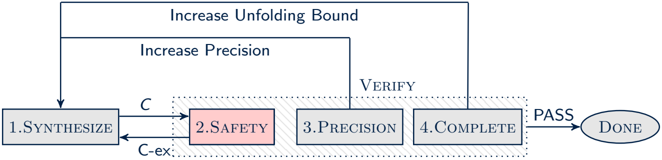
Controller synthesis

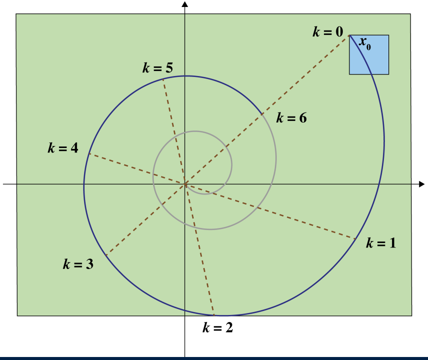


Find a controller for given $\{x_0\}$ and $k=6$ such that the system is stable and safe



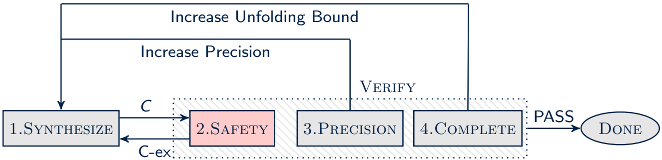
Controller synthesis

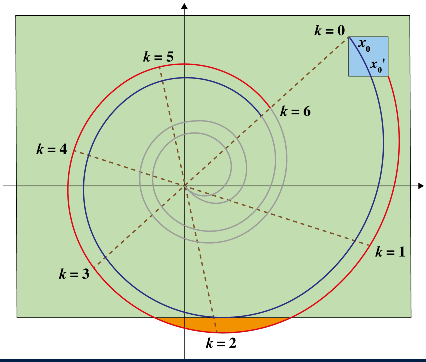




Find an initial state for which the system is unsafe

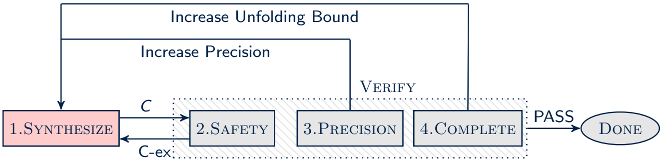
Controller synthesis

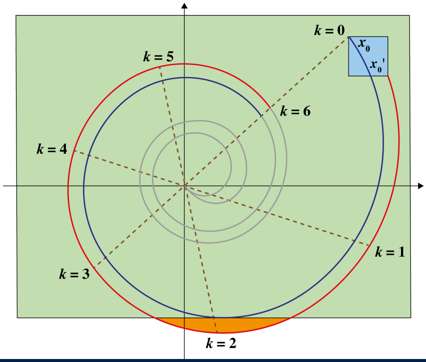




Find an initial state for which the system is unsafe

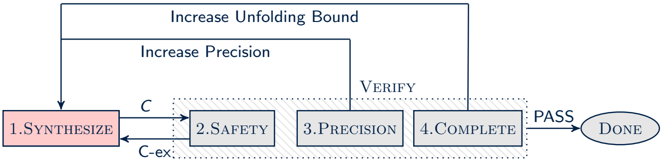
Controller synthesis

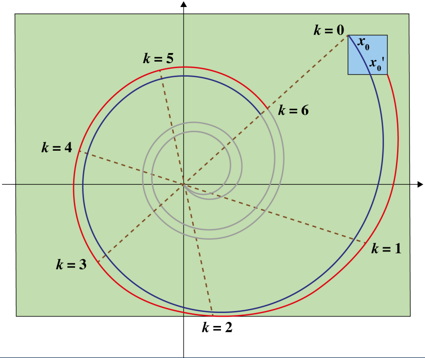




Find a controller for $\{x_0, x_0'\}$ and k=6 such that the system is stable and safe

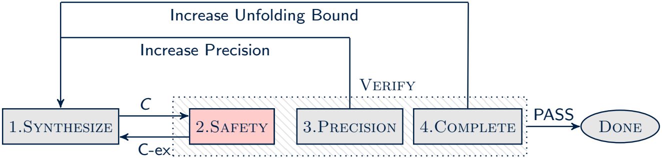
Controller synthesis

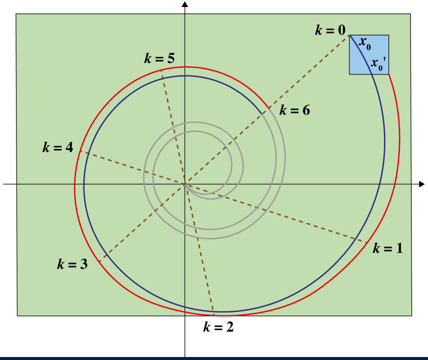




Find a controller for $\{x_0, x_0'\}$ and $k{=}6$ such that the system is stable and safe

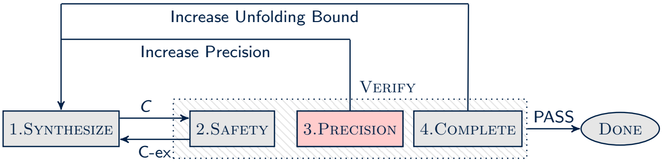
Controller synthesis



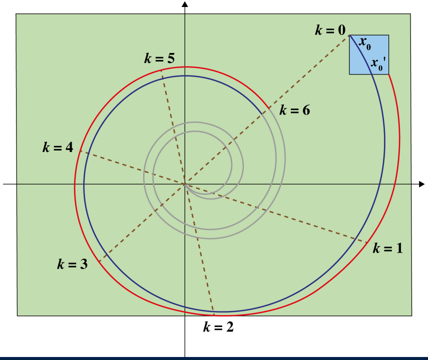


Find an initial state for which the system is unsafe

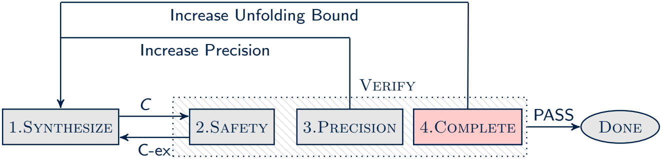
Controller synthesis



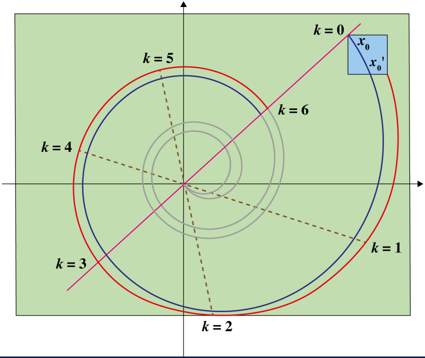
Check that the plant precision is sufficient



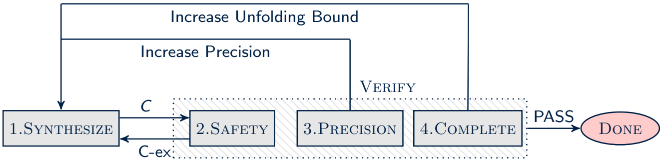
Controller synthesis



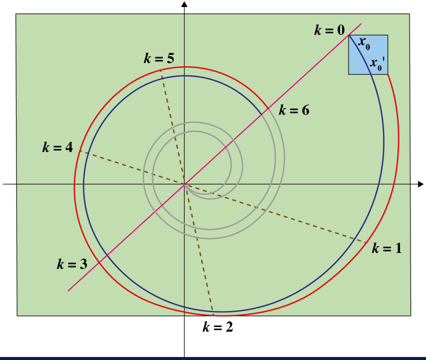
Check that k is sufficient



Controller synthesis



Controller found



Experimental results

#	Benchmark	Dimension	Completeness threshold	
			$\langle I_p, F_p \rangle$	Time
1	**Cruise Control**	1	8,16	7.44 s
2	**DC Motor**	2	8,16	7.76 s
3	**Helicopter**	3	8,16	12.13 s
4	**Inverted Pendulum**	4	8,16	8.82 s
5	**Magnetic Pointer**	2	8,16	10.31 s
6	**Magnetic Suspension**	2	12,20	21.55 s
7	**Pendulum**	2	8,16	9.08 s
8	**Suspension**	2	8,16	17.18 s
9	**Tape Driver**	3	8,16	8.05 s
10	**Satellite**	2	8,16	8.76 s

Synthesis phases: SYNTHESIZE 52%, VERIFY 48%

Experimental results

#	Benchmark	Dimension	Completeness threshold	
			$\langle I_p, F_p \rangle$	Time
1	**Cruise Control**	1	8,16	7.44 s
2	**DC Motor**	2	8,16	7.76 s
3	**Helicopter**	3	8,16	12.13 s
4	**Inverted Pendulum**	4	8,16	8.82 s
5	**Magnetic Pointer**	2	8,16	10.31 s
6	**Magnetic Suspension**	2	12,20	21.55 s
7	**Pendulum**	2	8,16	9.08 s
8	**Suspension**	2	8,16	17.18 s
9	**Tape Driver**	3	8,16	8.05 s
10	**Satellite**	2	8,16	8.76 s

Synthesis phases: SYNTHESIZE 52%, VERIFY 48%


Automated Formal Synthesis of Digital Controllers for State-Space Physical Plants

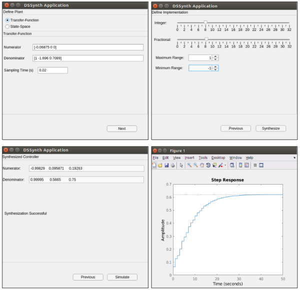
Check abstract acceleration based approach in the paper!

Conclusions

- **Automated synthesizer for digital state-feedback controllers that ensures stability and safety**
- **Evaluated the errors due to the controller's implementation and plant's modeling**

Conclusions

- **Automated synthesizer for digital state-feedback controllers that ensures stability and safety**
- **Evaluated the errors due to the controller's implementation and plant's modeling**



DSSynth Matlab toolbox:
www.cprover.org/DSSynth/dssynth-toolbox-1.0.0.zip