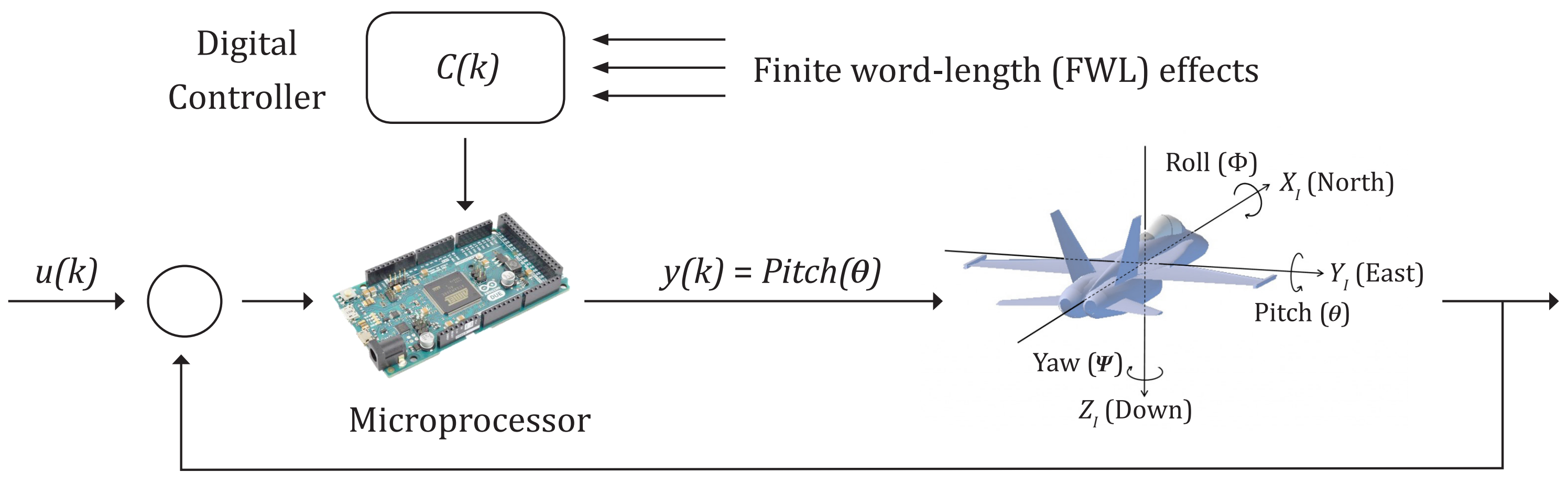
DSSynth: An Automated Digital Controller Synthesis Tool for Physical Plants

Alessandro Abate[1], Iury Bessa[2], Dario Cattaruzza[1], Lennon Chaves[2], Lucas Cordeiro[1,2], Cristina David[1], Pascal Kesseli[1], Daniel Kroening[1] and Elizabeth Polgreen[1]

[1] University of Oxford, Oxford, United Kingdom [2] Federal University of Amazonas, Manaus, Brazi

alessandro.abate@cs.ox.ac.uk, iurybessa@ufam.edu.br, dario.cattaruzza@cs.ox.ac.uk, lennon.correach@gmail.com, lucas.cordeiro@cs.ox.ac.uk, cristina.david@gmail.com, kesseli.pascal@gmail.com, kroening@cs.ox.ac.uk, elizabeth.polgreen@linacre.ox.ac.uk

I Motivation


Digital Controller
$C(k)$
Finite word-length (FWL) effects
$u(k)$
$y(k) = Pitch(\theta)$
Microprocessor

"...guaranteeing the correctness of cyber-physical systems (CPS) remains an outstanding challenge"

Xi Zheng et al., 2014

"Simulation alone is not sufficient to support verification and validation of CPS"

Sayan Mitra et al., 2013

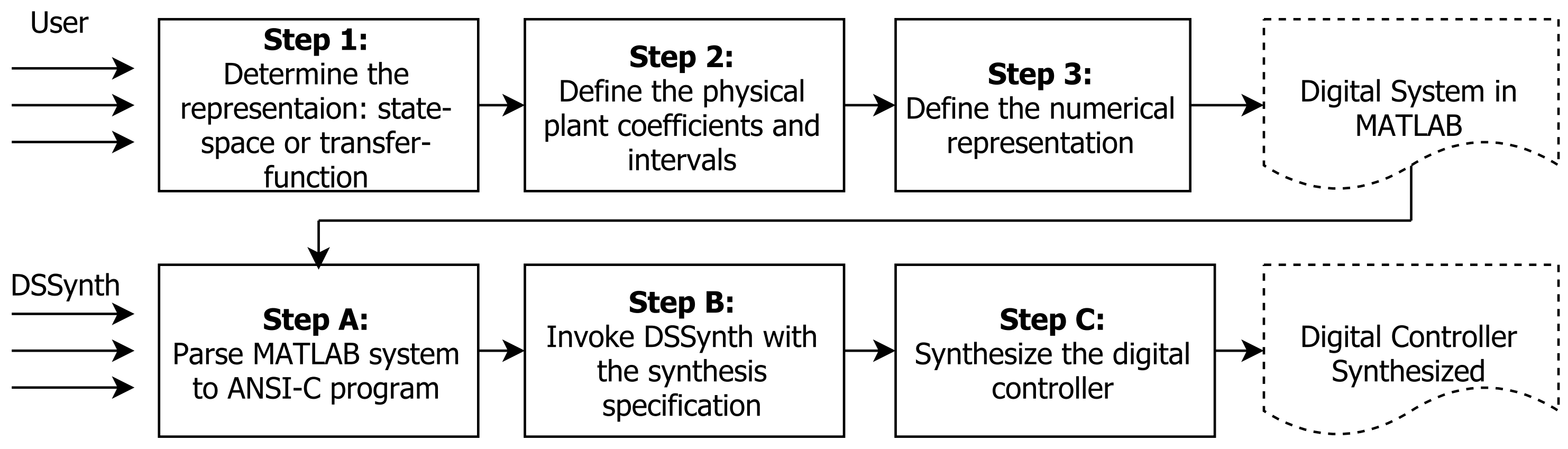
$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$ ← **State-space model**

$$H(z) = \frac{b_0 + b_1 z^{-1} + \ldots + b_m z^{-m}}{a_0 + a_1 z^{-1} + \ldots + a_n z^{-n}}$$ ← **Transfer-function model**

II Approach and Uniqueness

Counter-Example Guided Inductive Synthesis (CEGIS)

Generate sound digital controllers for stability and safety specifications with a very high degree of automation


User

Step 1: Determine the representaion: state-space or transfer-function

Step 2: Define the physical plant coefficients and intervals

Step 3: Define the numerical representation

Digital System in MATLAB

DSSynth

Step A: Parse MATLAB system to ANSI-C program

Step B: Invoke DSSynth with the synthesis specification

Step C: Synthesize the digital controller

Digital Controller Synthesized

Step 1

Determine the representation

- State-space model
- Transfer-function model

Step 2

Define the physical plant coefficients and intervals

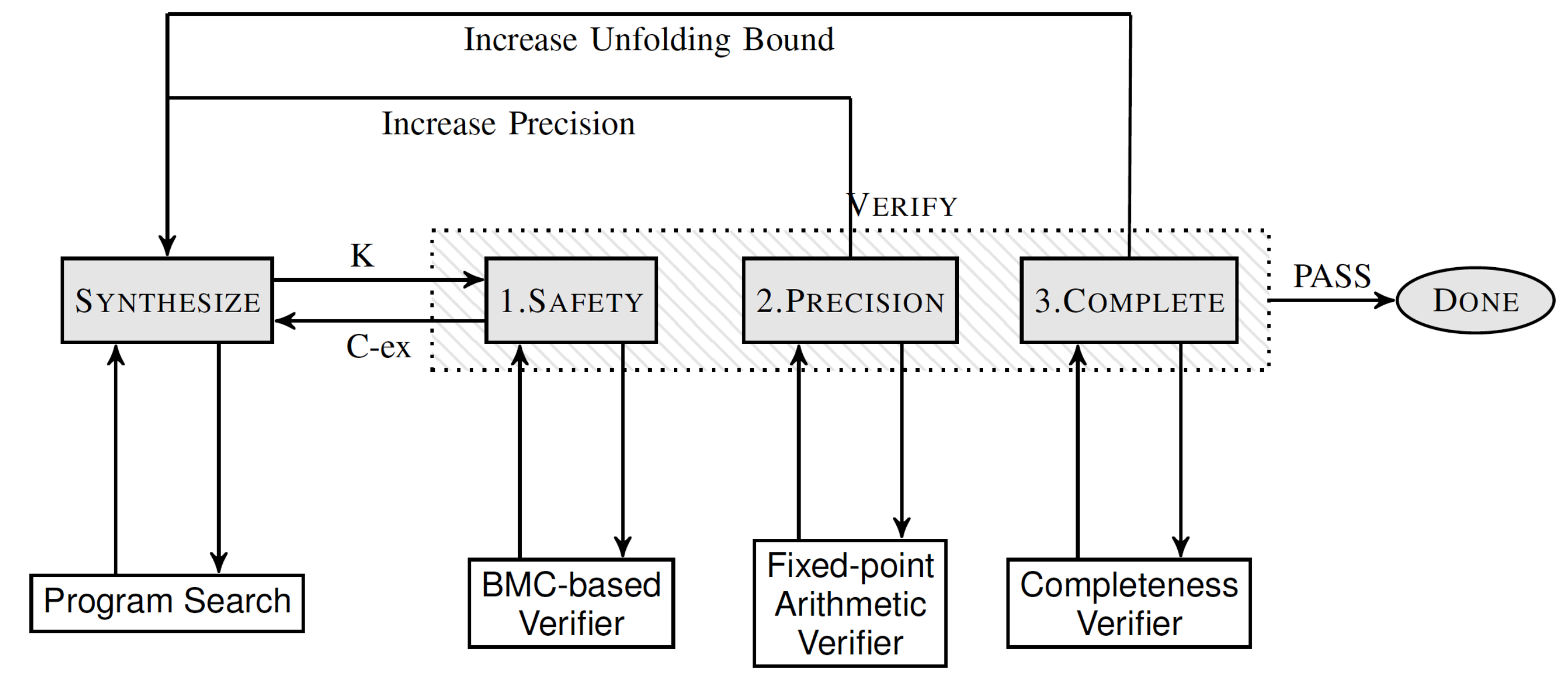
- state-space: matrices A, B, C and D
- transfer-function: coefficients $b_0, b_1, \ldots, b_m$ and $a_0, a_1, \ldots, a_n$
- uncertainty over the numerator and denominator coefficients

Step 3

Define the numerical representation

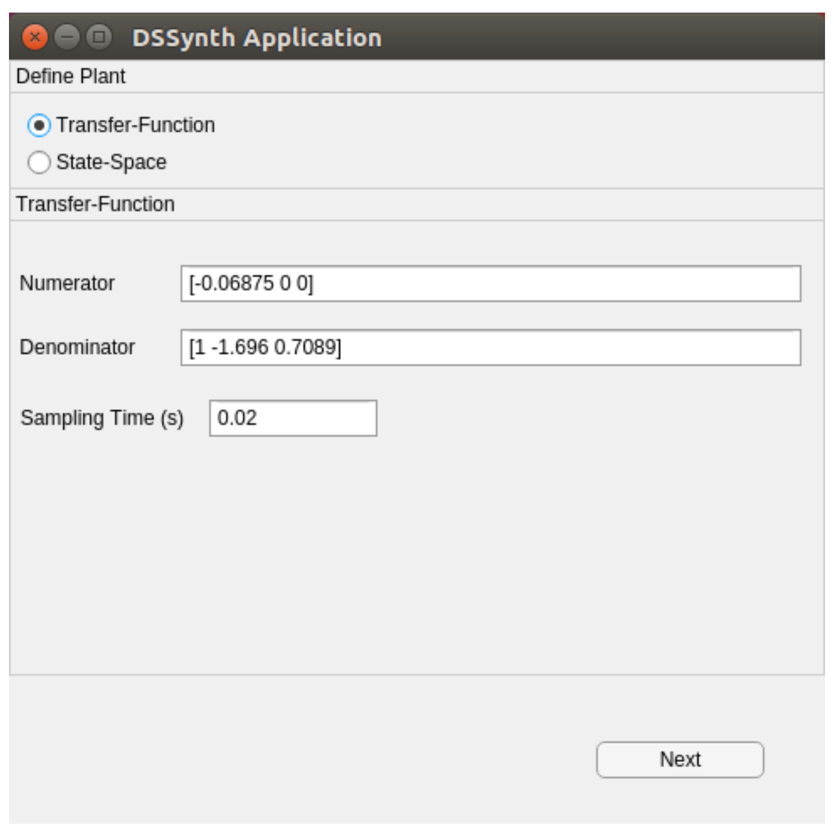
- ***I*** is the integer part
- ***F*** is the fractional part
- dynamical range

III CEGIS for Control Systems

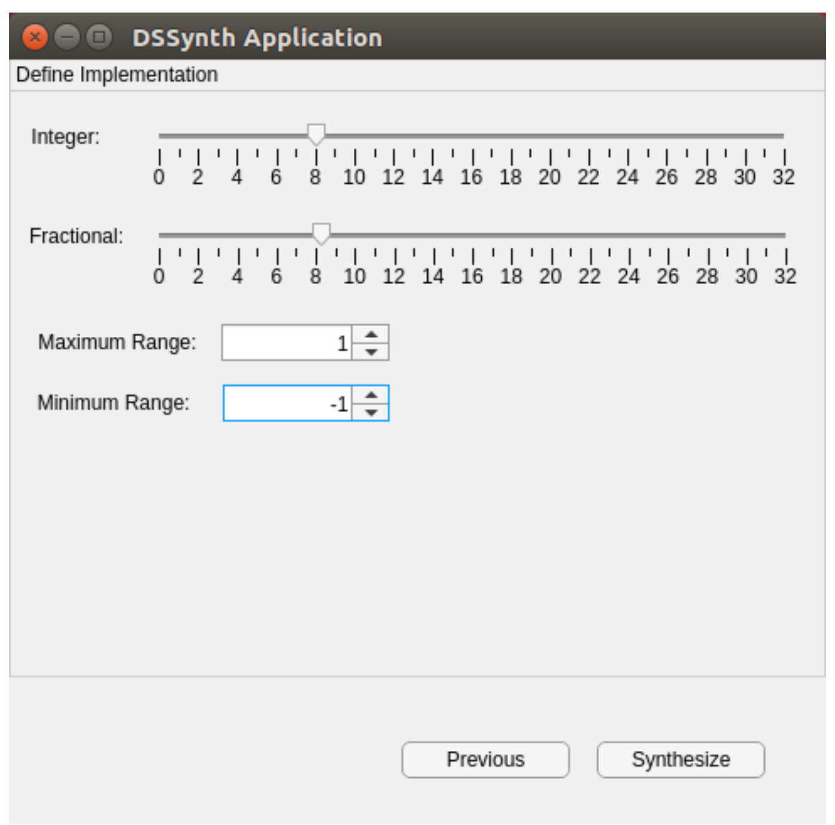

Increase Unfolding Bound
Increase Precision
VERIFY
SYNTHESIZE
K
C-ex
1.SAFETY
2.PRECISION
3.COMPLETE
PASS
DONE
Program Search
BMC-based Verifier
Fixed-point Arithmetic Verifier
Completeness Verifier

IV DSSynth Toolbox

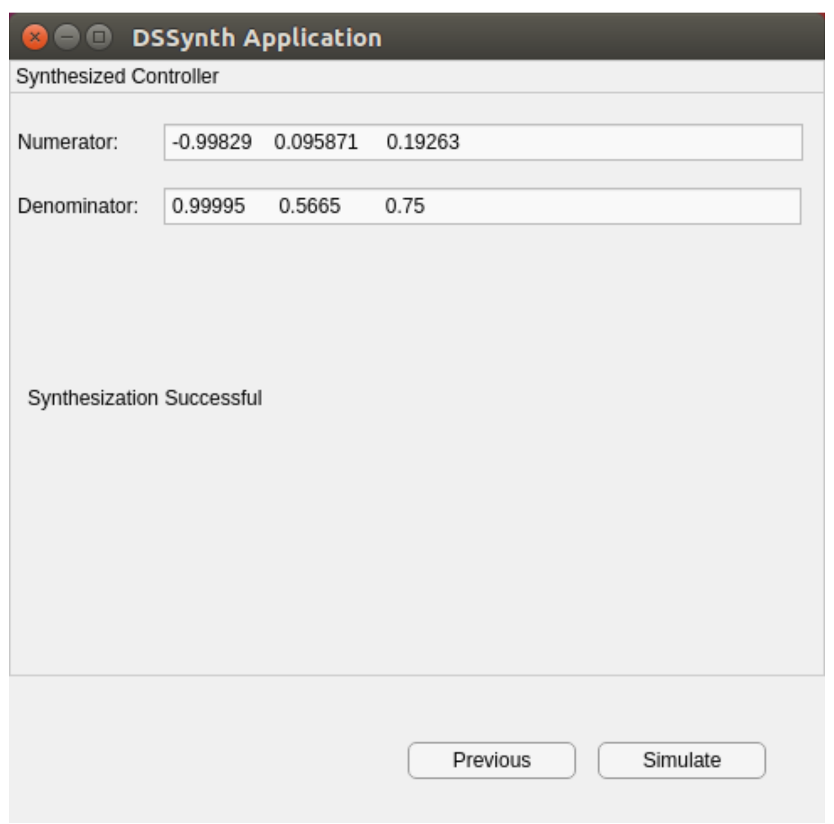
Steps 1 and 2


DSSynth Application
Define Plant
● Transfer-Function
○ State-Space
Transfer-Function
Numerator [-0.06875 0 0]
Denominator [1 -1.696 0.7089]
Sampling Time (s) 0.02

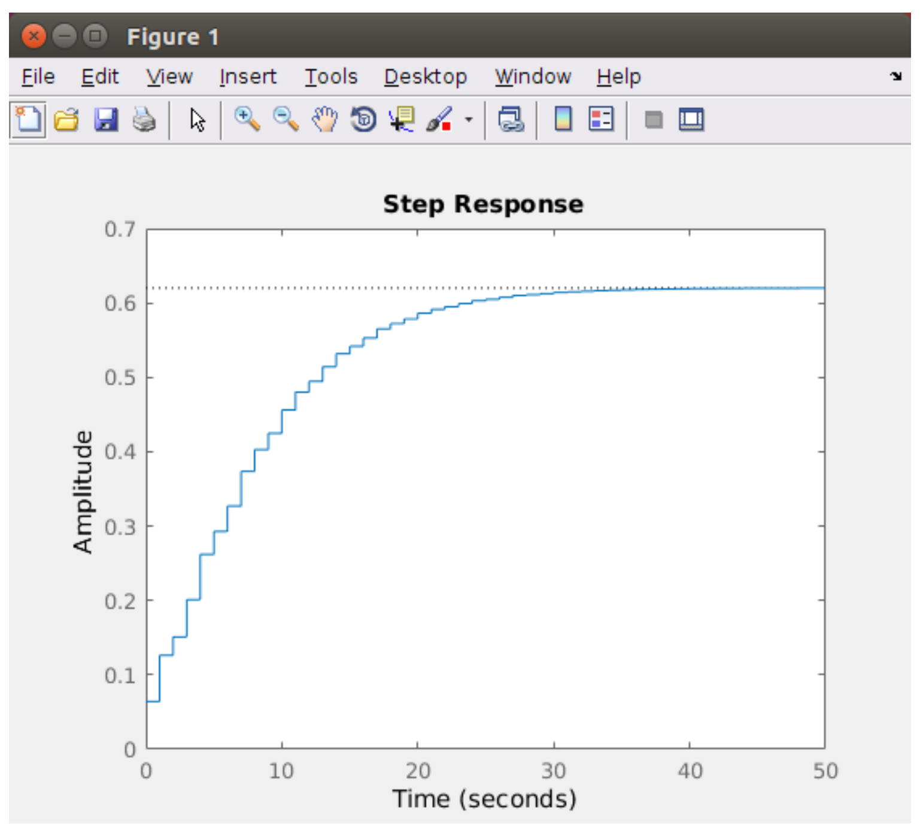
Step 3


DSSynth Application
Define Implementation
Integer:
Fractional:
Maximum Range: 1
Minimum Range: -1

Steps A, B and C


DSSynth Application
Synthesized Controller
Numerator: -0.99829 0.095871 0.19263
Denominator: 0.99995 0.5665 0.75
Synthesization Successful

Simulation in MATLAB



V Contributions

i. support for transfer-function and state-space representations in closed-loop form;

ii. synthesize different numerical representations and realization forms of the controller using CEGIS;

iii. provide a MATLAB toolbox to synthesize digital controllers while taking into account FWL effects.

As future work:

- DSSynth Toolbox will perform synthesis considering performance requirements;
- we will also pursue the application of CEGIS to further software engineering problems.

Sponsors: UNIVERSITY OF OXFORD · UNIVERSIDADE FEDERAL DO AMAZONAS · IN UNIVERSA SCIENTIA VERITAS

For further information, publications, and downloads, see: http://dsverifier.org/dsverifier/dssynth-toolbox/