Worst-Case Analysis of Digital Control Loops with Uncertain Input/Output Timing

(Benchmark Proposal)

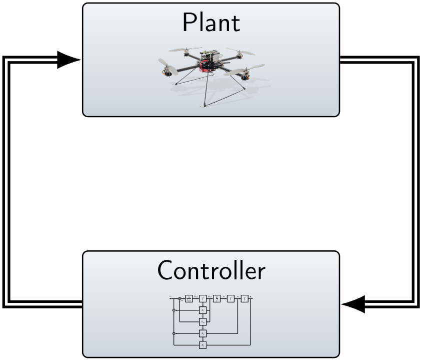
Maximilian Gaukler and Peter Ulbrich

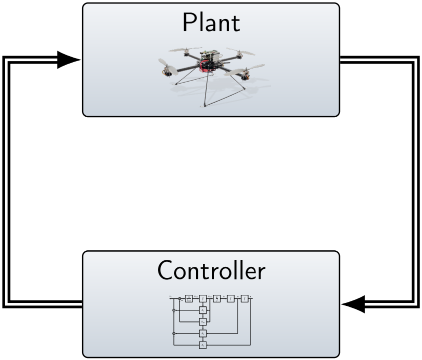
Friedrich-Alexander-Universität
Erlangen-Nürnberg (FAU)

ARCH'19, Montréal, Canada
April 15, 2019

Motivation

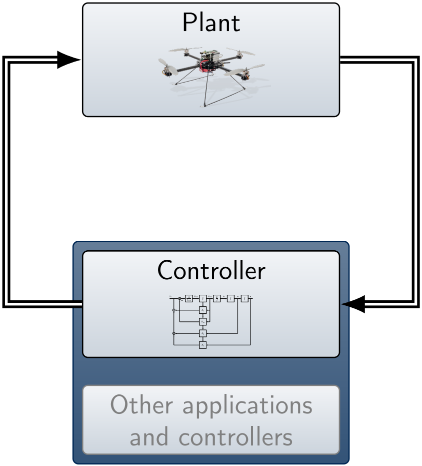
Motivation





Controller Design:

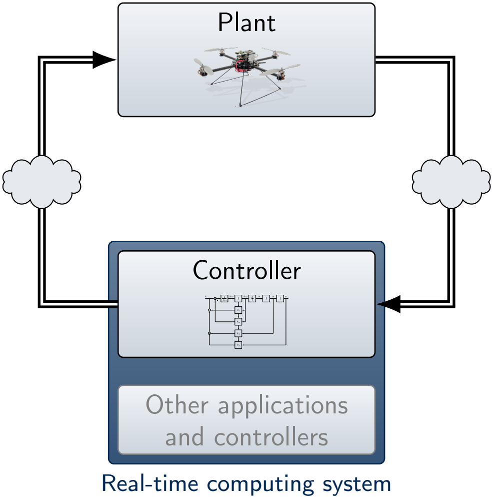
input/output assumed periodic



Real-time computing system

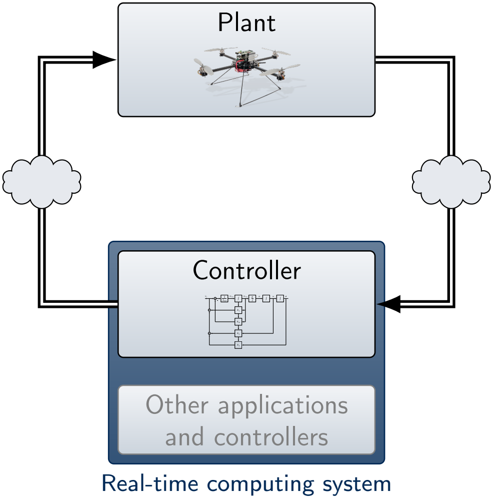
Controller Design:

input/output assumed periodic



Controller Design:

input/output assumed periodic

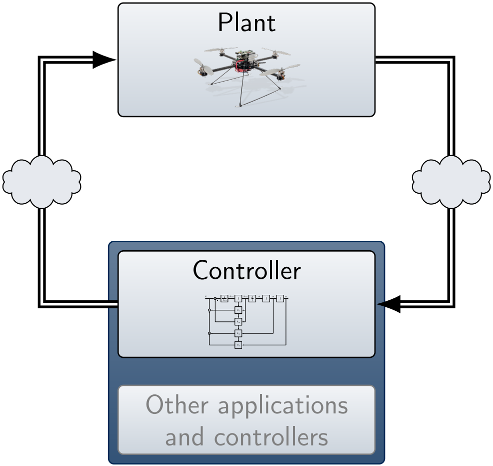


Controller Design:

input/output assumed periodic

Modern Real-Time Systems:

- Network / bus systems
- Smart sensors
- High complexity



Real-time computing system

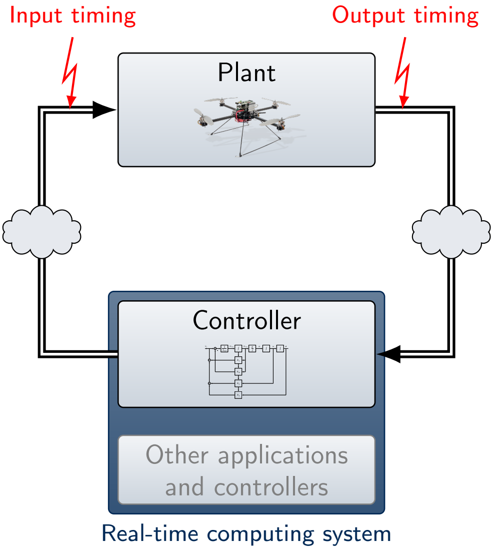
Controller Design:

input/output assumed periodic

Modern Real-Time Systems:

- Network / bus systems
- Smart sensors
- High complexity

$\rightsquigarrow$ **Strict timing is difficult!**



Controller Design:

input/output assumed periodic

Modern Real-Time Systems:

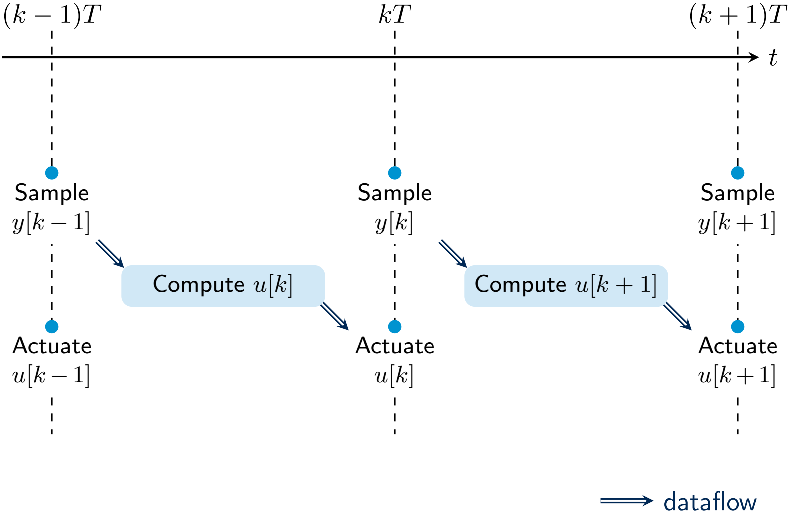
- Network / bus systems
- Smart sensors
- High complexity

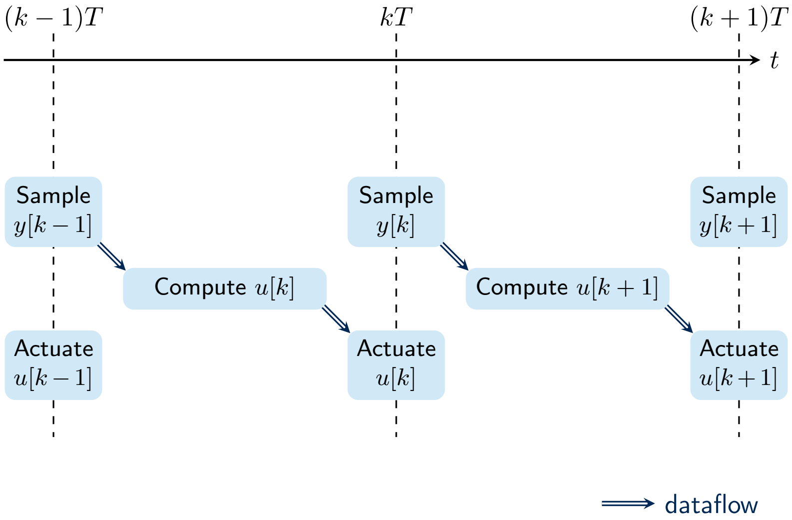
⇝ **Strict timing is difficult!**

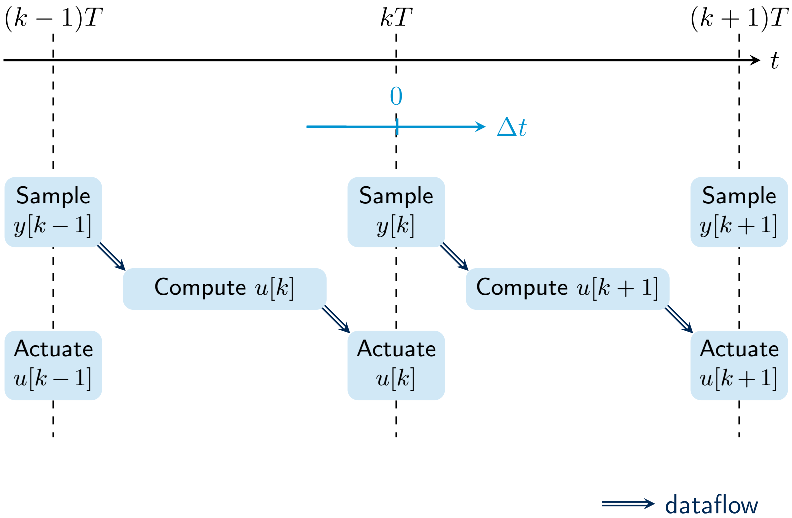
Desired Alternative:

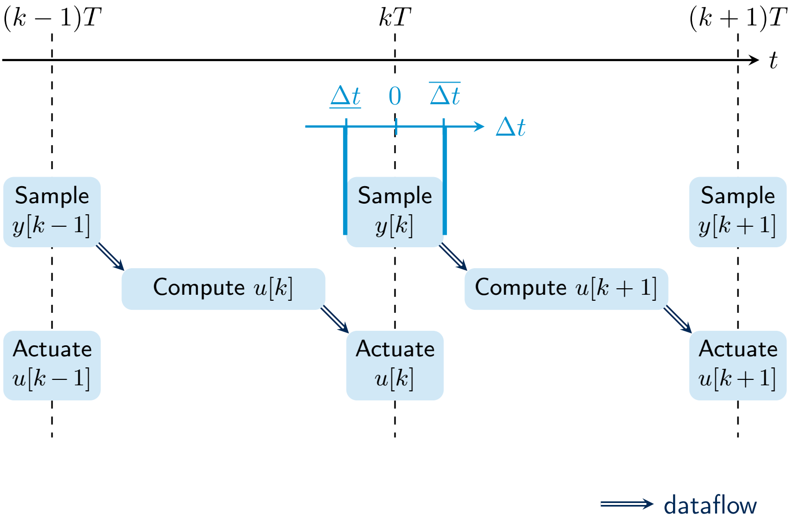
Allow some timing deviation

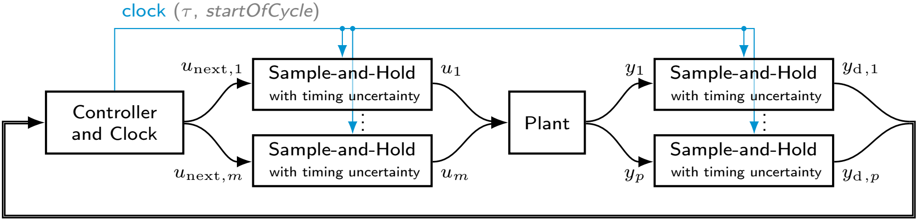
⇝ Formal safety guarantees?

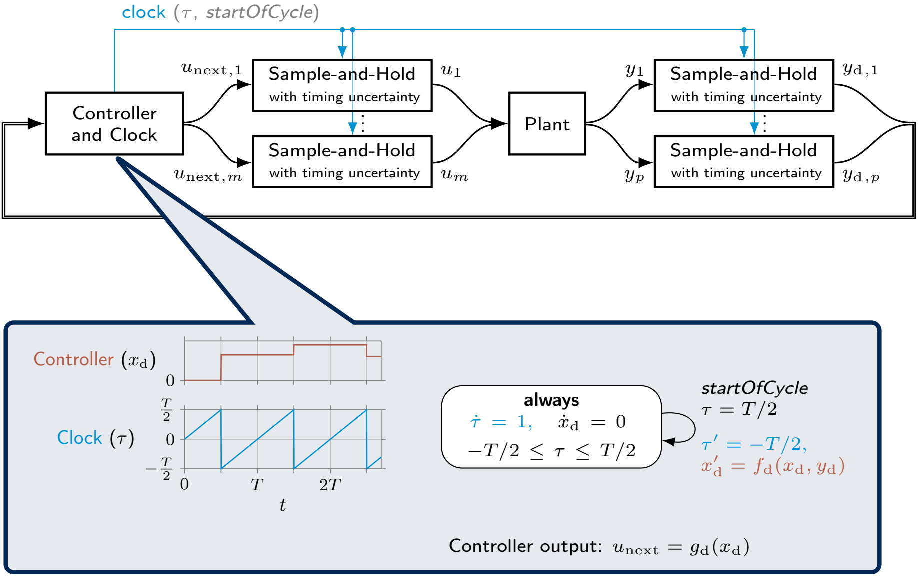

$(k-1)T$
kT
$(k+1)T$
t
Sample
$y[k-1]$
Sample
$y[k]$
Sample
$y[k+1]$
Compute $u[k]$
Compute $u[k+1]$
Actuate
$u[k-1]$
Actuate
$u[k]$
Actuate
$u[k+1]$
dataflow

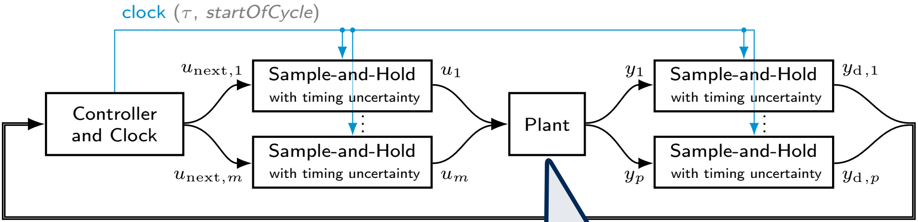

$(k-1)T$
kT
$(k+1)T$
t
Sample $y[k-1]$
Sample $y[k]$
Sample $y[k+1]$
Compute $u[k]$
Compute $u[k+1]$
Actuate $u[k-1]$
Actuate $u[k]$
Actuate $u[k+1]$
$\Longrightarrow$ dataflow


$(k-1)T$
kT
$(k+1)T$
t
0
Δt
Sample $y[k-1]$
Sample $y[k]$
Sample $y[k+1]$
Compute $u[k]$
Compute $u[k+1]$
Actuate $u[k-1]$
Actuate $u[k]$
Actuate $u[k+1]$
dataflow


$(k-1)T$
kT
$(k+1)T$
t
$\underline{\Delta t}$
0
$\overline{\Delta t}$
Δt
Sample $y[k-1]$
Sample $y[k]$
Sample $y[k+1]$
Compute $u[k]$
Compute $u[k+1]$
Actuate $u[k-1]$
Actuate $u[k]$
Actuate $u[k+1]$
dataflow


clock (τ, *startOfCycle*)
Controller and Clock
$u_{\mathrm{next},1}$
$u_{\mathrm{next},m}$
Sample-and-Hold with timing uncertainty
Sample-and-Hold with timing uncertainty
u_1
u_m
Plant
y_1
y_p
Sample-and-Hold with timing uncertainty
Sample-and-Hold with timing uncertainty
$y_{\mathrm{d},1}$
$y_{\mathrm{d},p}$

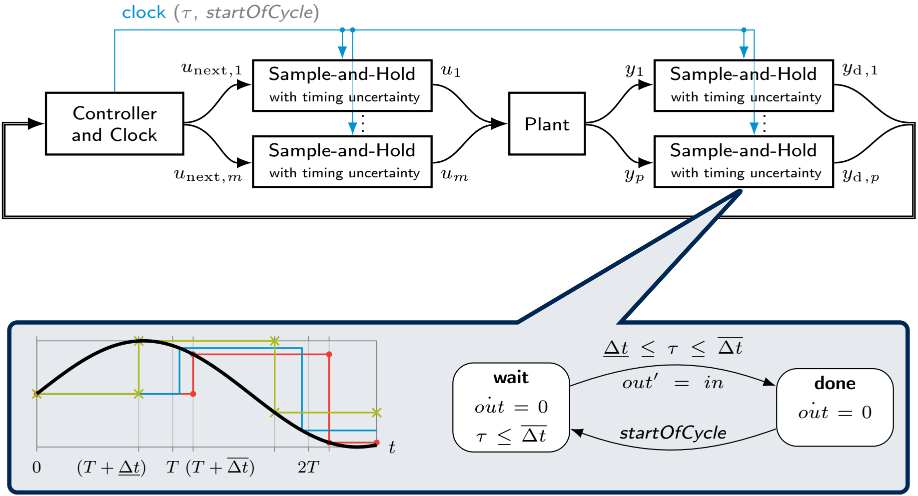

clock (τ, *startOfCycle*)
Controller and Clock
$u_{\text{next},1}$
$u_{\text{next},m}$
Sample-and-Hold with timing uncertainty
Sample-and-Hold with timing uncertainty
u_1
u_m
Plant
y_1
y_p
Sample-and-Hold with timing uncertainty
Sample-and-Hold with timing uncertainty
$y_{\text{d},1}$
$y_{\text{d},p}$
Controller (x_d)
Clock (τ)
0
$\frac{T}{2}$
$-\frac{T}{2}$
0
T
$2T$
t
always
$\dot{\tau} = 1, \quad \dot{x}_\text{d} = 0$
$-T/2 \leq \tau \leq T/2$
startOfCycle
$\tau = T/2$
$\tau' = -T/2,$
$x'_\text{d} = f_\text{d}(x_\text{d}, y_\text{d})$
Controller output: $u_\text{next} = g_\text{d}(x_\text{d})$



$$\dot{x}_{\mathrm{p}}(t) = f_{\mathrm{p}}(x_{\mathrm{p}}(t), u(t), d(t))$$
$$y(t) = g_{\mathrm{p}}(x_{\mathrm{p}}(t), d(t))$$
$$d(t) \in D$$

- Multiple inputs and outputs
- Bounded disturbance and measurement noise

Formalization: Hybrid Automata



1. System model: network of hybrid automata
   - Machine-readable, *unambiguous*

1. System model: network of hybrid automata
   - Machine-readable, *unambiguous*

2. Verification goal: tight worst-case bounds (infinite-time reachable set)
   - Metric: bloating factor

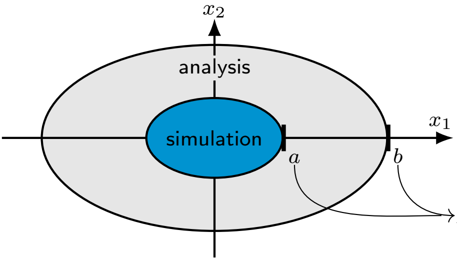
FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

❶ System model: network of hybrid automata

- Machine-readable, *unambiguous*

❷ Verification goal: tight worst-case bounds (infinite-time reachable set)

- Metric: bloating factor



$$K = \frac{b}{a} = \text{“}\frac{\text{upper bound from analysis}}{\text{worst observed in simulation}}\text{”}$$

1. System model: network of hybrid automata
   - Machine-readable, *unambiguous*
2. Verification goal: tight worst-case bounds (infinite-time reachable set)
   - Metric: bloating factor
3. Example systems
   - Linear, no disturbance
   - From 1D examples to a simplified 3-axis quadrocopter controller

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

❶ System model: network of hybrid automata

- Machine-readable, *unambiguous*

❷ Verification goal: tight worst-case bounds (infinite-time reachable set)

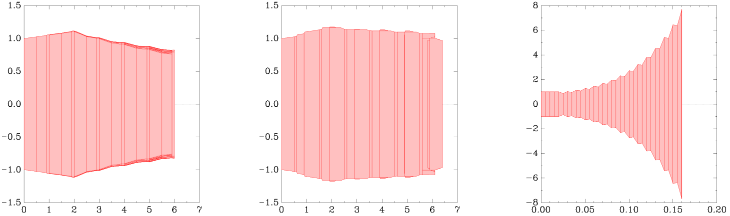
- Metric: bloating factor

❸ Example systems

- Linear, no disturbance
- From 1D examples to a simplified 3-axis quadrocopter controller

❹ Experiments with SpaceEx: Success only for trivial examples

Reachable set over time:



✓ 1D, small uncertainty ✓ 1D, large uncertainty × 3D, perfect timing (!)

- Problem: Timing uncertainties in digital control
  - Hard to avoid
  - Verification is challenging, but of high practical relevance

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

- Problem: Timing uncertainties in digital control
  - Hard to avoid
  - Verification is challenging, but of high practical relevance

- Is a pure hybrid-automata approach suitable here?

- Problem: Timing uncertainties in digital control
  - Hard to avoid
  - Verification is challenging, but of high practical relevance

- Is a pure hybrid-automata approach suitable here?

- $\rightsquigarrow$ Future work: "non-hybrid" alternatives
  - Continuous-time abstraction: continuization
  - Discrete-time: LMI-based robust stability

- Problem: Timing uncertainties in digital control
  - Hard to avoid
  - Verification is challenging, but of high practical relevance
- Is a pure hybrid-automata approach suitable here?
- $\rightsquigarrow$ Future work: “non-hybrid” alternatives
  - Continuous-time abstraction: continuization
  - Discrete-time: LMI-based robust stability

Can your tool solve the benchmark?

`http://qronos.de` $\rightarrow$ Files and code (GPLv3)

Appendix

	n_p	n_d	m	p	timing	SpaceEx	t_{SE}	K_{SE}
1D example								
A2	1	1	1	1	varying (negligible)	✓	1 s	1.001
A1	1	1	1	1	varying (small)	✓	1 s	1.010
A3	1	1	1	1	varying (medium)	✓	2 s	1.059
A4	1	1	1	1	varying (large)	× error (GLPK)	—	—
A5	2	2	2	2	varying (like A3)	× crash (GLPK)	—	—
3D, trivial (stable, negligible controller)								
B1	3	2	2	1	varying	✓	16 s	1.097
1-axis quadrotor angular rate control								
C1	1	2	1	1	constant	× timeout	—	—
C2	1	2	1	1	varying	× crash	—	—
3-axis quadrotor angular rate control								
D1	3	6	4	3	constant	× diverging	—	—
D2	3	6	4	3	varying	× crash	—	—

- Structural problem: Failures even with $\Delta t = 0$

$\rightsquigarrow$ Tools optimized for mostly-continuous systems?

$\neq$ here: many discrete transitions, "unstable" inbetween

- Dimension problematic: $2^{(\#\text{inputs}+\#\text{outputs})}$ discrete states