RVFuzzer: Finding Input Validation Bugs in Robotic Vehicles through Control-Guided Testing

Taegyu Kim, Chung Hwan Kim, Junghwan Rhee,
Fan Fei, Zhan Tu, Gregory Walkup,
Xiangyu Zhang, Xinyan Deng, Dongyan Xu

PURDUE UNIVERSITY
CERIAS

Robotic Vehicles?

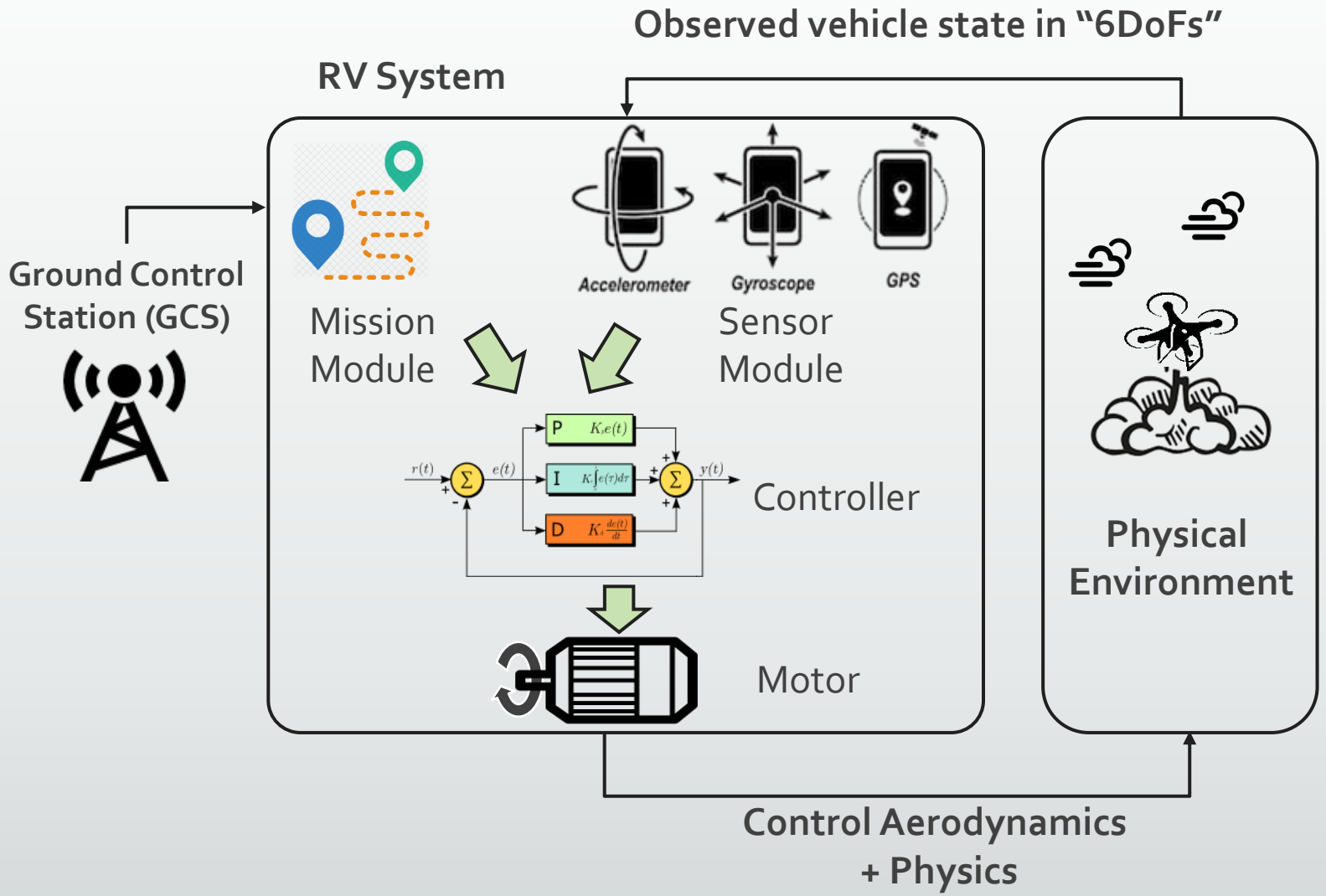
December 8, 2014 / News / Photo Galleries / Sheepshead Bay

Drone strike! Our photographer injured by TGI Friday's mistletoe copter

BY VANESSA OGLE

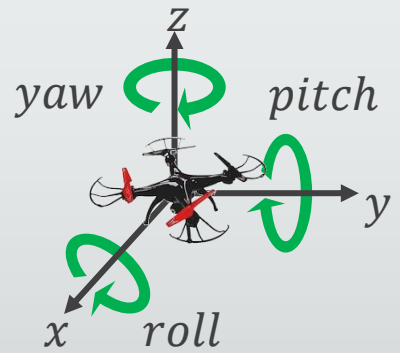
Published Aug 26, 2013 at 10:12 AM | Updated at 1:14 PM EDT on Aug 26, 2013

How Do Robotic Vehicles Work?

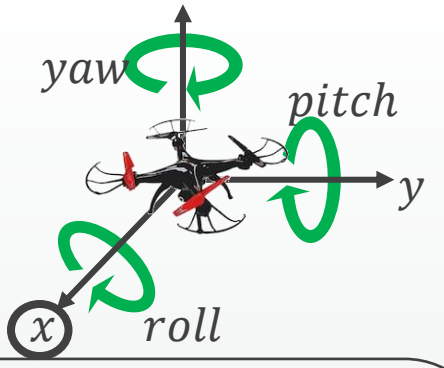


- Execute GCS commands
- Stabilize physical operations

6 degrees of freedom (6DoF)



Complexity of Robotic Vehicle Control Software



- **Hundreds of parameters**
- **Dynamically configurable!**

Ground Control Station (GCS)

Controller Param.

Mission Param.

x-axis Cascading Controller

POS Controller

VEL Controller

ACCEL Controller

k_x r_x x_x

$\dot{k}_x$ $\dot{r}_x$ $\dot{x}_x$

$\ddot{k}_x$ $\ddot{r}_x$ $\ddot{x}_x$

o_x

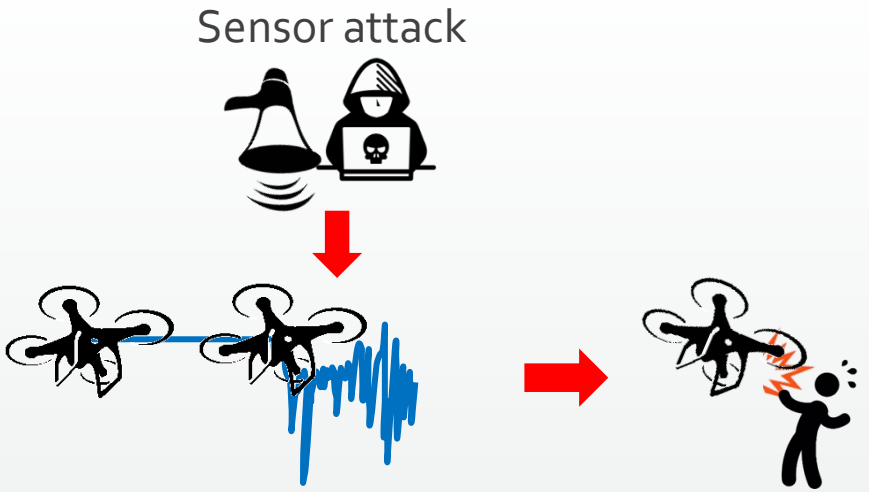
Physical Operations

Sensor + Sensor Param.

Feedback Input

Landscape of RV Attacks

- Physical attacks [Security'15, EuroS&P'17..]
  - e.g., sensor spoofing
  - Defense: control-based detection and filter



- Software "syntactic" bug exploitation [NDSS'18]
  - e.g., buffer overflow
  - Defense: program fuzzing and hardening

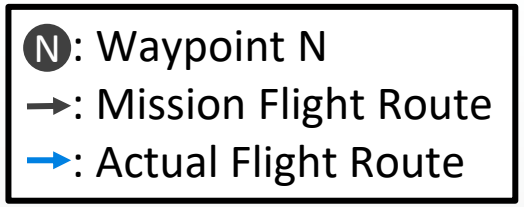
- **Control-"semantic" bug exploitation**
  - Less explored yet
  - Not defendable with above approaches

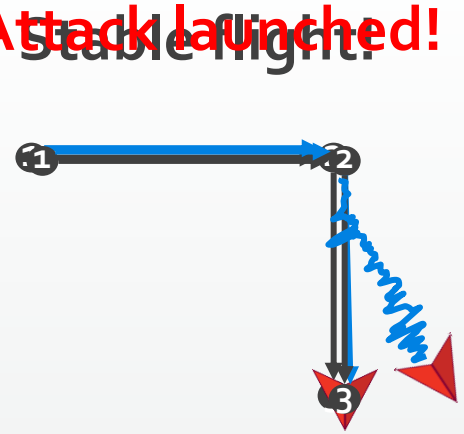
Control-Semantic Bug Exploitation

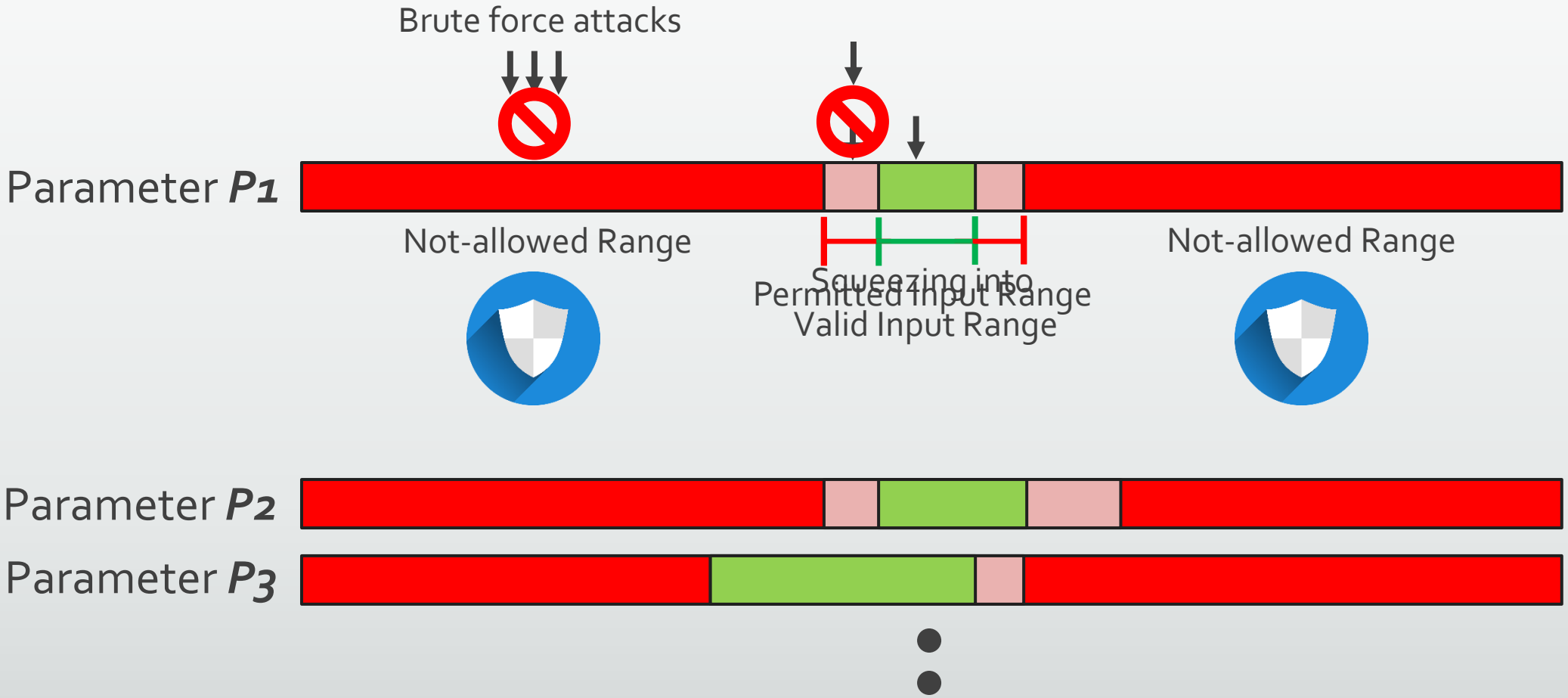
- Malicious parameter-change command
  - GCS-Vehicle communication is not secure [BlackHat'16, NOMS'16]
    - e.g., MAVLink
  - Cause *at least* one controller to malfunction

- Why is this meaningful to attackers?
  - (Remotely) triggered by ***single*** malicious control parameter-change command
    - Leave minimum footprint
  - No need for sensor spoofing, code injection, trojaned exploits
  - Launched even after program is hardened against traditional exploits

Nature of Control-Semantic Bug

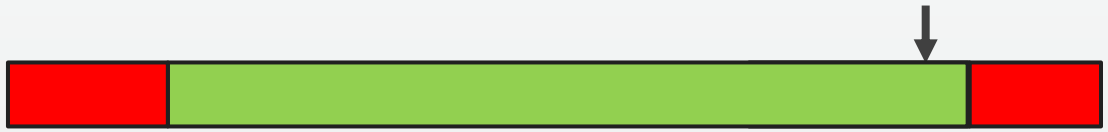






Wind Effect

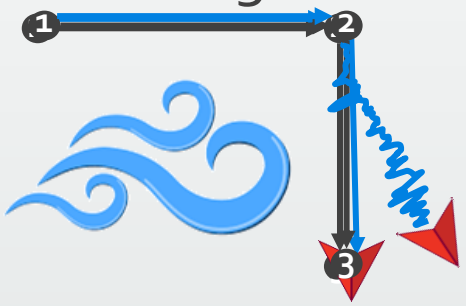
Parameter ***P***



N: Waypoint N
→: Mission Flight Route
→: Actual Flight Route

Stable flight!
w/o strong wind

Attack
w/ strong wind



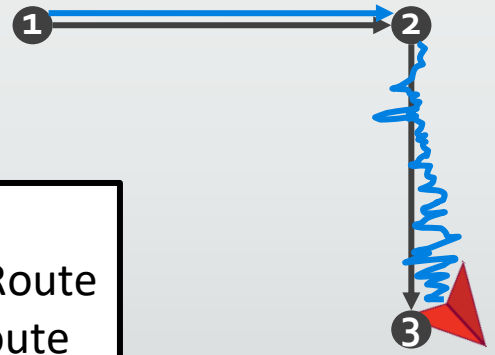
Finding the Bugs: Challenge and Solution

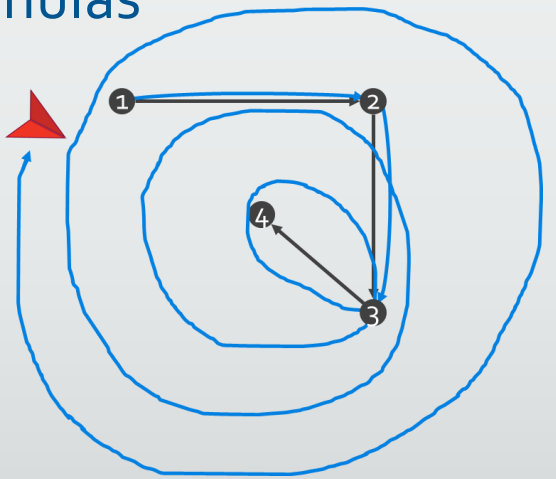
Challenge

- How to detect a bad program run?
  - Bad traditional program run?
    - e.g., program crash
    - NOT applicable to control programs
  - Bad control program run?
    - e.g., physical control instability
    - NOT involve in program crash

Solution

- Define *control instability condition*
  - Non-transient divergence between
    - Reference state and observed state
    - Reference state and mission
  - Detectable with the standard control properties and formulas





N: Waypoint N
→: Mission Flight Route
→: Actual Flight Route

Finding the Bugs: Challenge and Solution

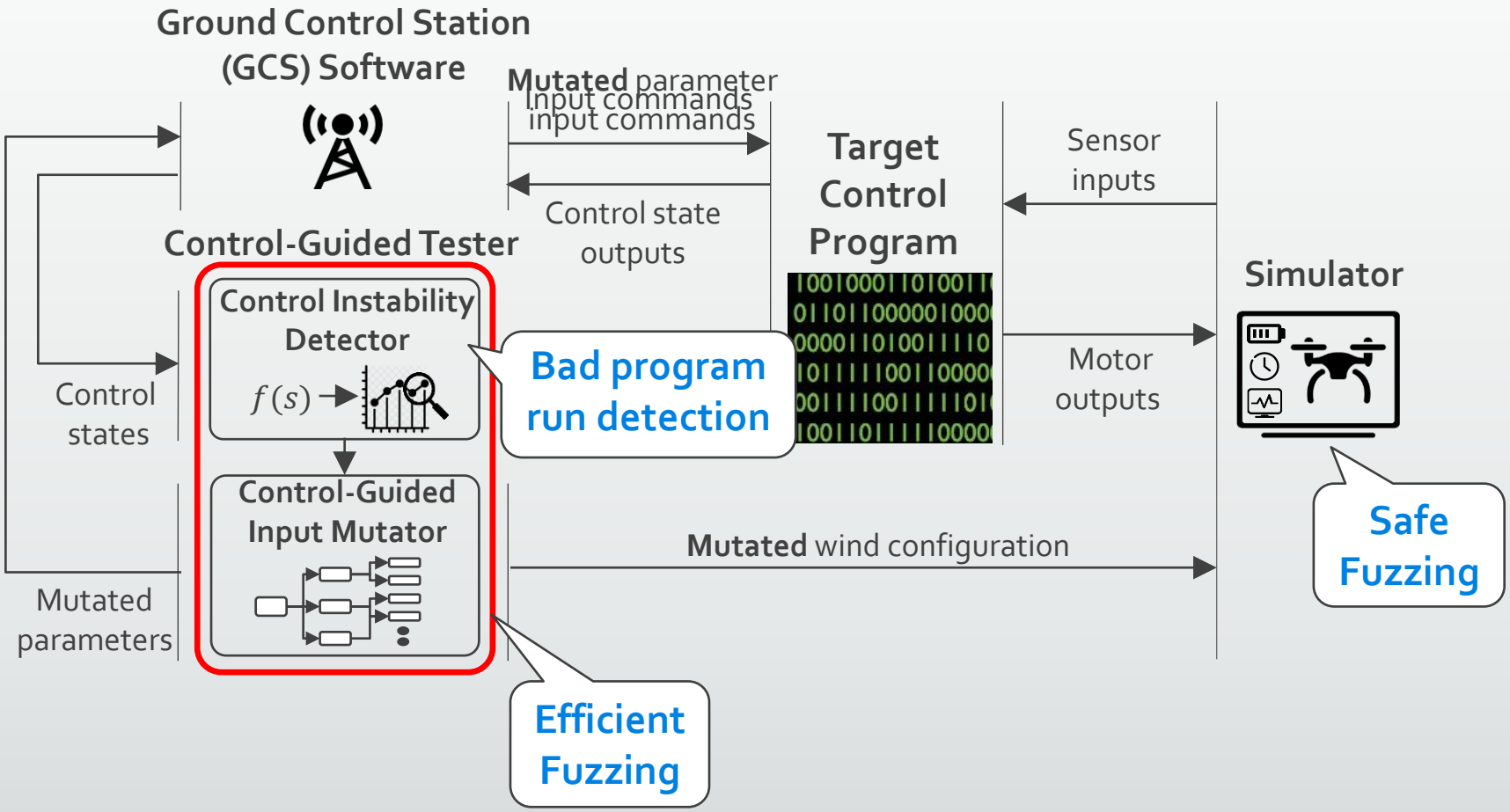
Challenge

- How to fuzz control loops?
  - Safety
    - Real vehicle crashes are dangerous
  - Efficiency
    - Hundreds of parameters
    - Large value ranges of parameters
    - Wind effect

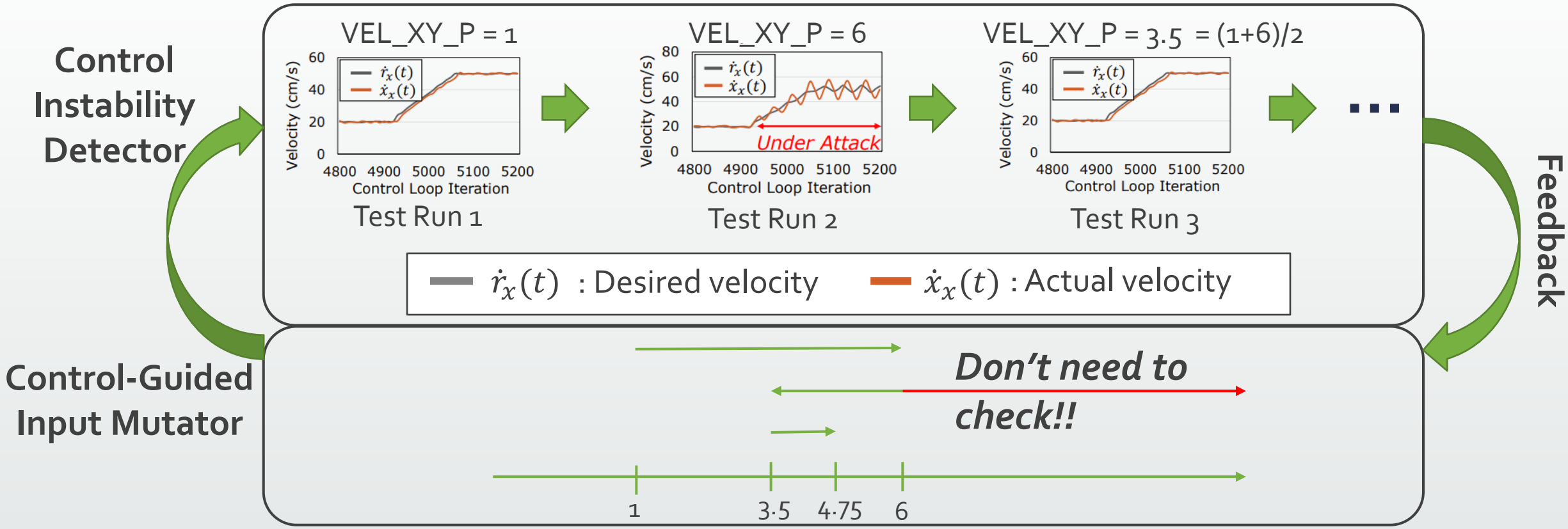
Solution

- Use a high-fidelity simulator
  - Provide a ***virtual physical*** world
  - Fuzz control loops safely
- Control-Guided, Feedback-Directed

Overview of RVFuzzer



Control-Guided Parameter Mutation



- Based on the monotonic control property
  - Increasing (decreasing) the value of a control parameter
  - → Maintain or intensify the control instability [IROS'99, AIAA'05, ...]

Evaluation with ArduPilot and PX4: 89 Bugs Found

Module	Sub-module	ArduPilot		PX4	
		RIB	RSB	RIB	RSB
Controller	x, y-axis position	1	0	1	1
	z-axis velocity	2	1	1	1
	x, y-axis position	1	0	1	1
	z-axis velocity	1	0	1	0
	z-axis acceleration	3	0	0	0
	Roll angle	1	0	1	1
	Roll angular rate	5	0	3	3
	Pitch angle	1	0	1	1
	Pitch angular rate	5	0	3	3
	Yaw angle	1	0	2	2
	Yaw angular rate	6	0	3	3
	Motor	0	0	3	3
Sensor	Inertia sensor	3	3	0	0
Mission	x, y-axis velocity	1	1	2	0
	z-axis velocity	2	0	4	0
	z-axis acceleration	2	0	0	0
	Roll, pitch	1	1	1	1
Total	-	36	6	27	20

- 8-days testing
- 89 bugs are found
- 8 confirmed by developers
- 7 patched by developers

RIB: Range Implementation Bug

RSB: Range Specification Bug

Evaluation: Vulnerable Parameters of ArduPilot

Control Program Module	Parameter	Physical Impacts			
		C	D	U	S
Controller	PSC_POSXY_P	✓			✓
	PSC_VELXY_P	✓	✓	✓	
	PSC_VELXY_I		✓	✓	
	PSC_POSZ_P				✓
	PSC_VELZ_P	✓			
	PSC_ACCZ_P	✓			✓
	PSC_ACCZ_I	✓	✓	✓	
	PSC_ACCZ_D	✓	✓	✓	
	ATC_ANG_RLL_P	✓			
	ATC_RAT_RLL_I	✓			
	ATC_RAT_RLL_IMAX	✓			✓
	ATC_RAT_RLL_D	✓			
	ATC_RAT_RLL_P	✓		✓	
	ATC_RAT_RLL_FF	✓		✓	
	ATC_ANG_PIT_P	✓			
	ATC_RAT_PIT_P	✓		✓	
	ATC_RAT_PIT_I	✓			
	ATC_RAT_PIT_IMAX	✓			

Control Program Module	Parameter	Physical Impacts			
		C	D	U	S
Controller	ATC_RAT_PIT_D	✓			✓
	ATC_RAT_PIT_FF	✓		✓	✓
	ATC_ANG_YAW_P	✓			
	ATC_SLEW_YAW			✓	
	ATC_RAT_YAW_P			✓	
	ATC_RAT_YAW_I			✓	
	ATC_RAT_YAW_IMAX				✓
	ATC_RAT_YAW_D	✓			✓
	ATC_RAT_YAW_FF	✓		✓	
Sensor	INS_POS1_Z	✓		✓	
	INS_POS2_Z	✓		✓	
	INS_POS3_Z	✓		✓	
Mission	WPNAV_SPEED				✓
	WPNAV_SPEED_UP				✓
	WPNAV_SPEED_DN				✓
	WPNAV_ACCEL	✓			✓
	WPNAV_ACCEL_Z	✓			✓
	ANGLE_MAX	✓			✓

C: Crash

D: Deviation from trajectory

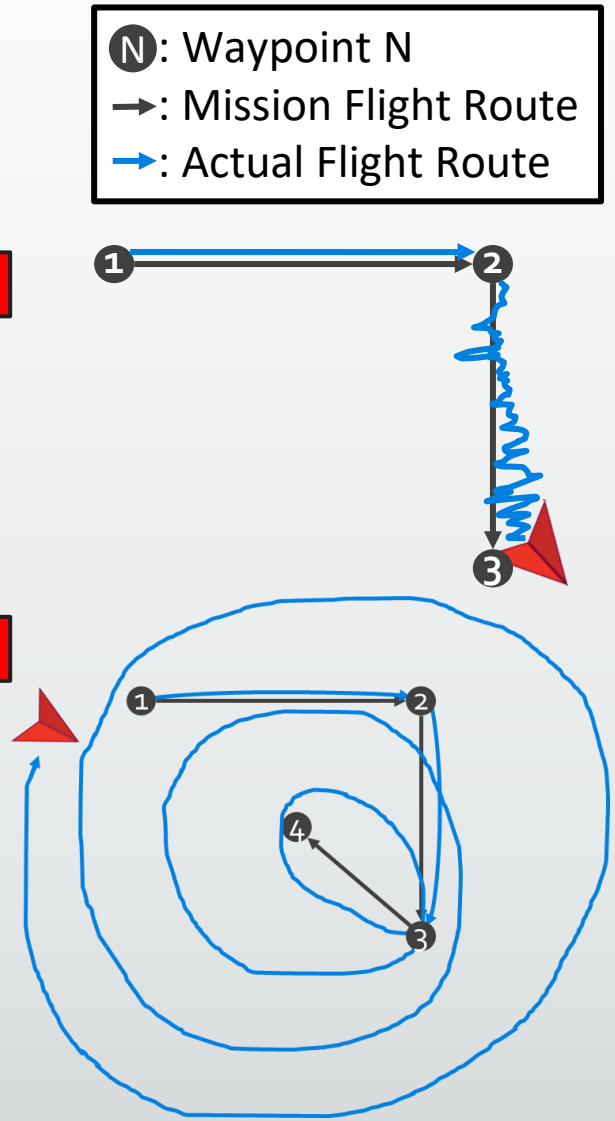
U: Unstable movement

S: Stuck in a certain location

Case Studies: Two Control-Semantic Bug Exploitation

= Maximum motor power

= Roll angular control gain



PURDUE UNIVERSITY

CERIAS

Summary

- Introduce a new type of control-semantic bugs
  - Malicious parameter-change commands
- RVFuzzer, a cyber-physical system fuzzing tool
  - Control-guided detection of bad ***control*** program run
    - By detecting generic control instability properties
  - Safe, efficient control loop fuzzing
    - By leveraging a high-fidelity simulator and control properties
- 89 bugs found in ArduPilot and PX4

Thank you!
Questions?

tgkim@purdue.edu

PURDUE UNIVERSITY

CERIAS