THE UNIVERSITY OF BRITISH COLUMBIA

Paper# 283

PID-Piper: Recovering Robotic Vehicles from Physical Attacks

Pritam Dash, [+]Guanpeng Li, Zitao Chen, Mehdi Karimi, and Karthik Pattabiraman

University of British Columbia, [+]University of Iowa

DSN 2021

The 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks

Taipei, Taiwan, June 21-24, 2021

Autonomous Robotic Vehicles: An Overview

Robotic Vehicles (RV) are becoming popular in many industrial sectors.

Safeguard RVs, Safe missions.

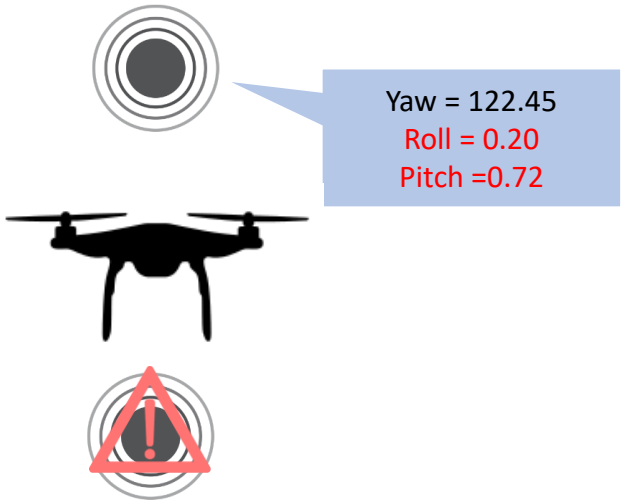
Physical Attacks Against Robotic Vehicles (RV)

GPS Spoofing.
Transmit malicious GPS Signals

Signal Injection.
Optical, Magnetic or Acoustic noise

Spoofed Position

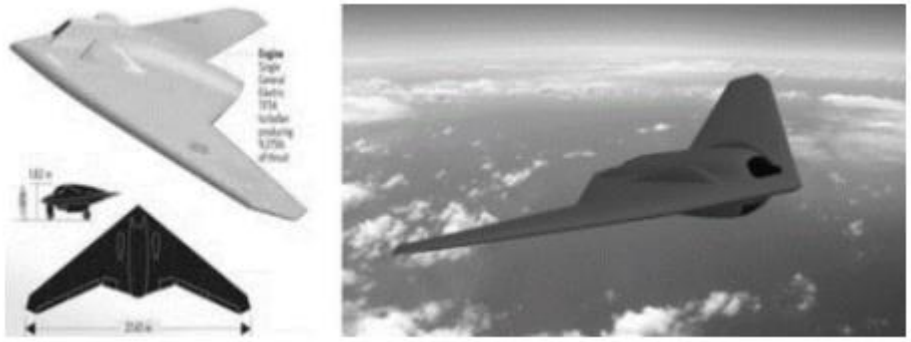
Actual Position

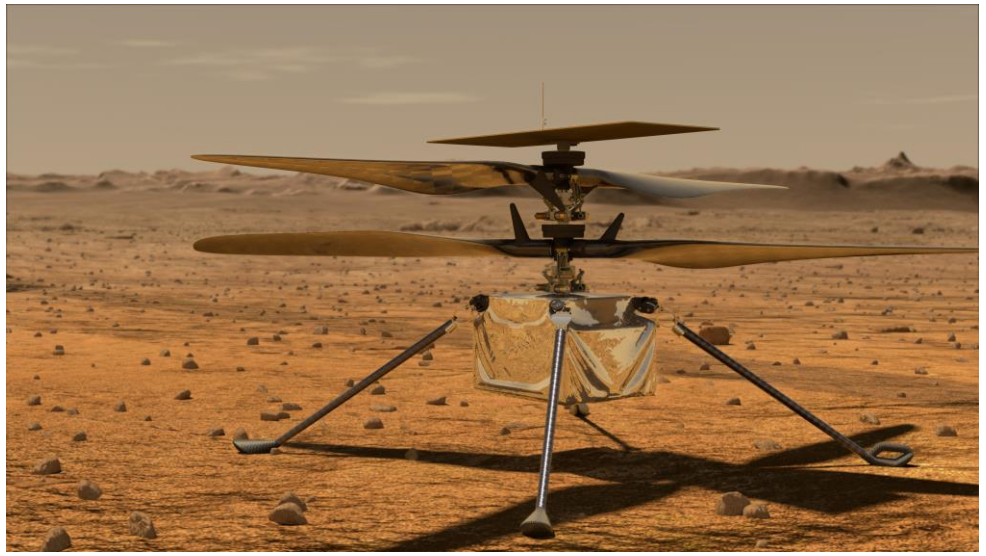


Tippenhauer et. al. On the requirements for successful GPS spoofing attacks. CCS'11
Son et. al. Rocking Drones with Intentional Sound Noise on Gyroscopic Sensors. Usenix Security'2015

Physical Attacks and Consequences

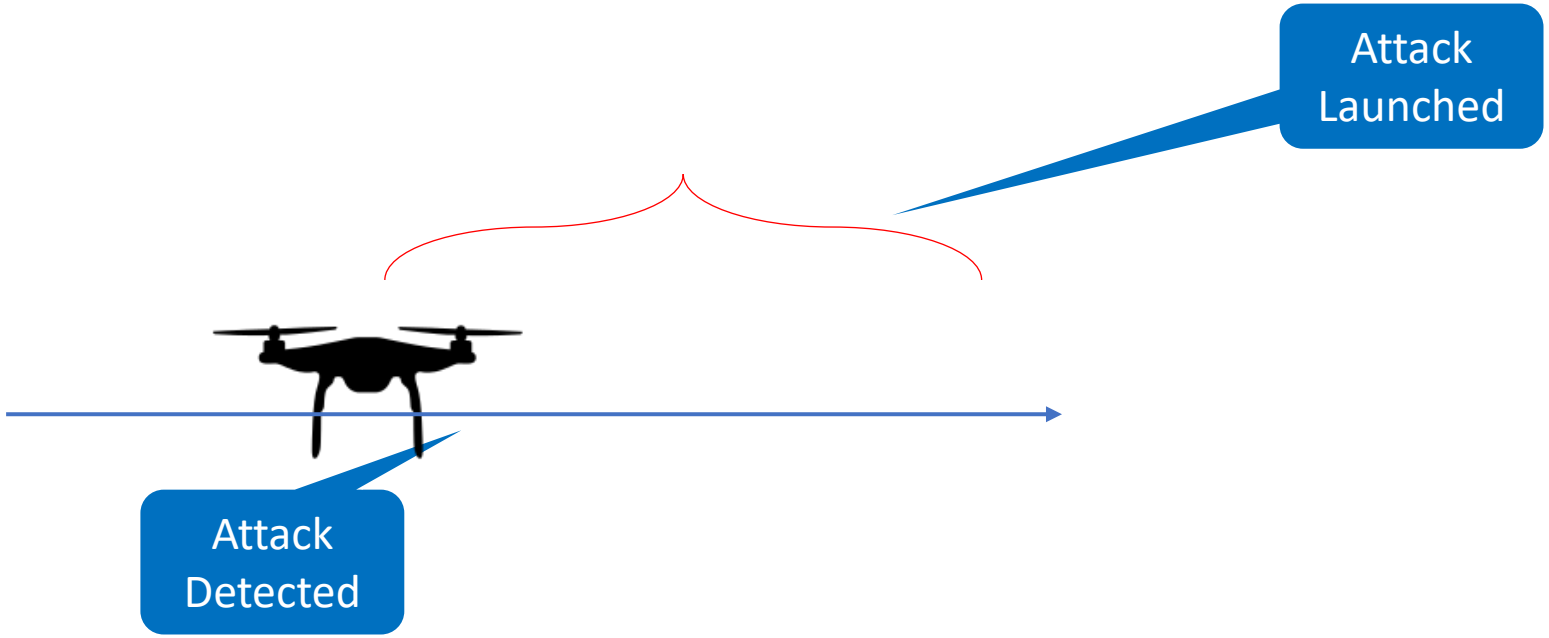
Iran–U.S. RQ-170 incident





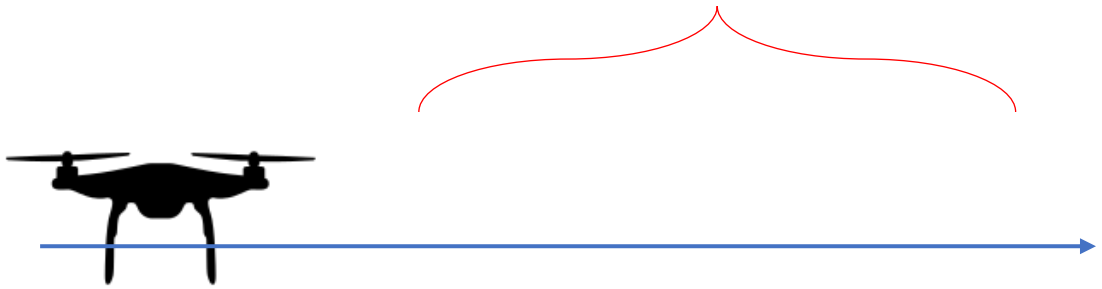
Iran–U.S. RQ-170 incident - https://en.wikipedia.org/wiki/Iran–U.S._RQ-170_incident
Ingenuity Flight Anomaly - https://www.space.com/mars-helicopter-ingenuity-sixth-flight-anomaly

Detecting Attacks Against RVs



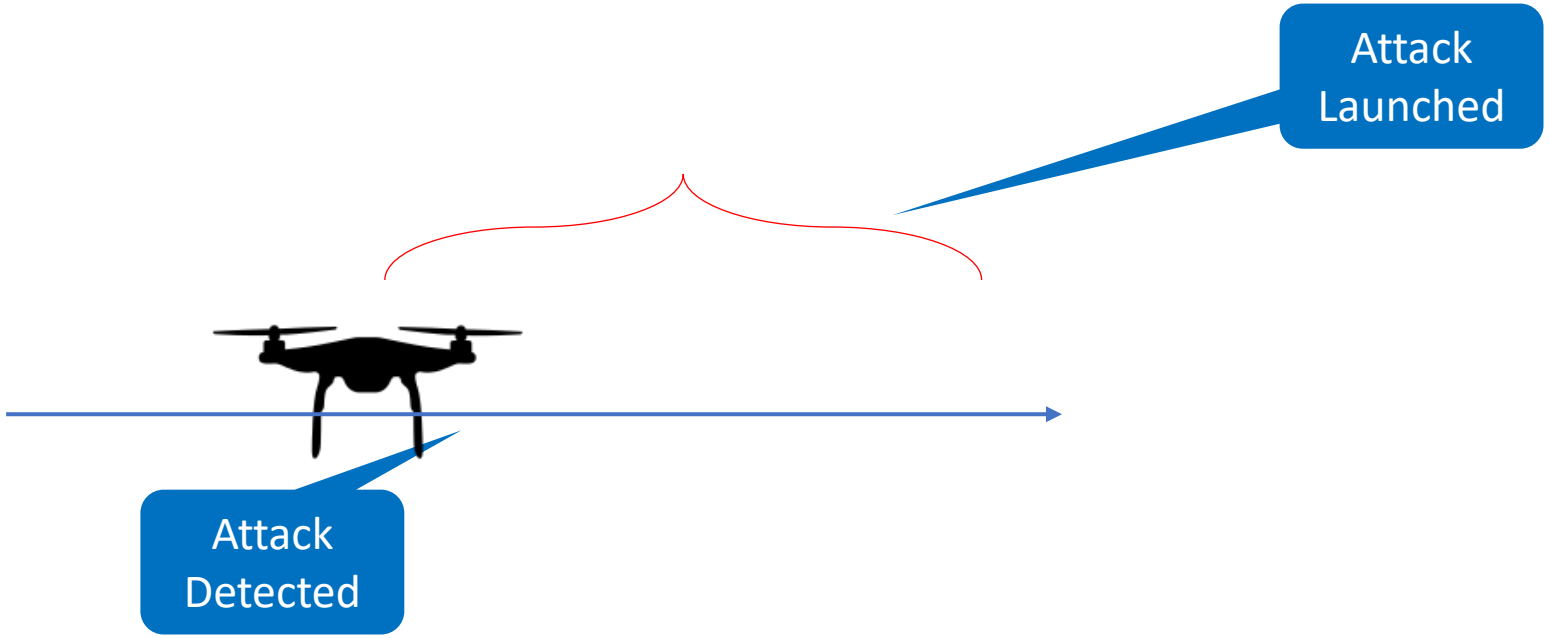
Choi et. al., Detecting Attacks against Robotic Vehicles: a Control Invariant Approach, CCS'18
Quinonez et. al., SAVIOR: Securing Autonomous Vehicles with Robust Physical Invariants, Usenix Security'20

Detection is not enough…



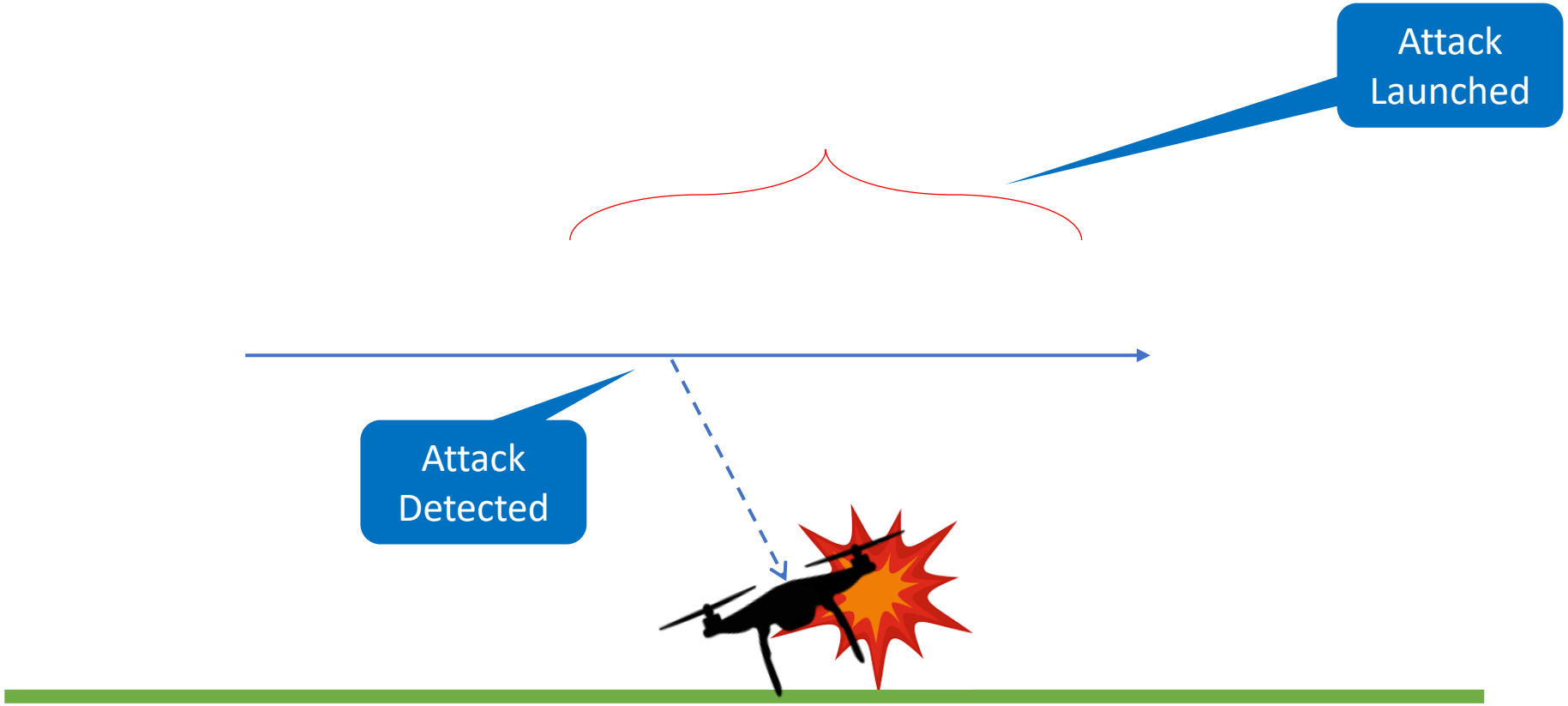
Choi et. al., Detecting Attacks against Robotic Vehicles: a Control Invariant Approach, CCS'18
Quinonez et. al., SAVIOR: Securing Autonomous Vehicles with Robust Physical Invariants, Usenix Security'20

Detection is not enough…



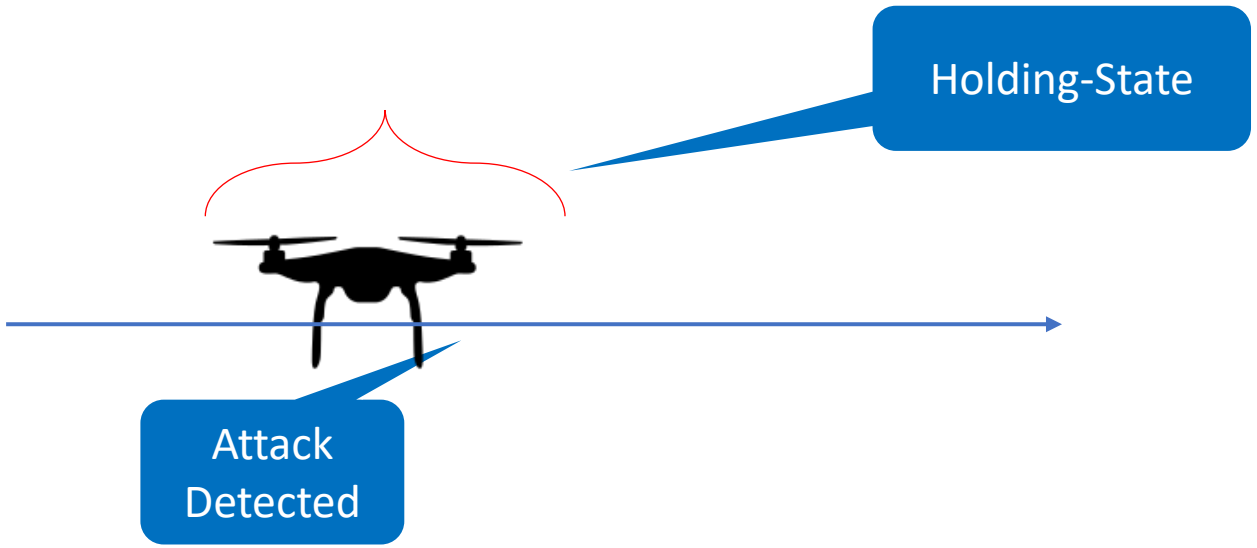
Choi et. al., Detecting Attacks against Robotic Vehicles: a Control Invariant Approach, CCS'18
Quinonez et. al., SAVIOR: Securing Autonomous Vehicles with Robust Physical Invariants, Usenix Security'20

Detection is not enough…



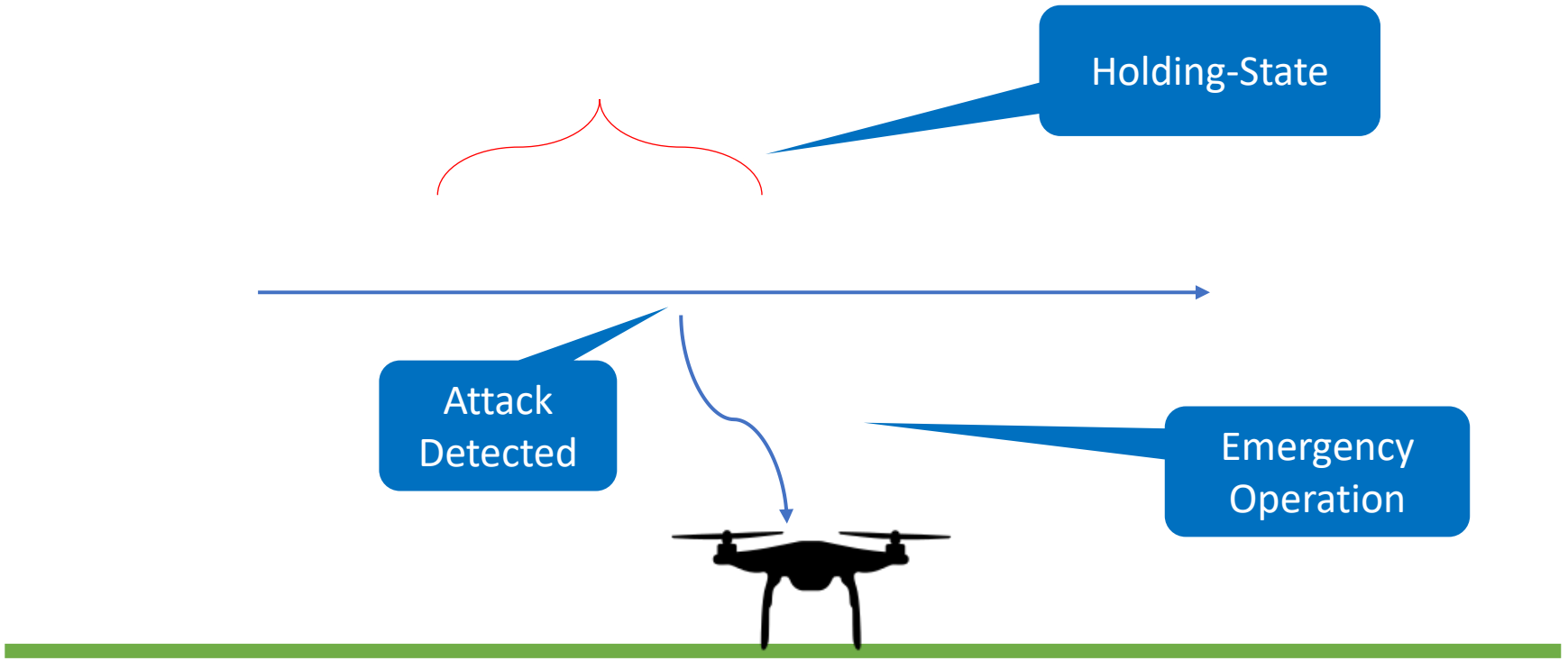
Choi et. al., Detecting Attacks against Robotic Vehicles: a Control Invariant Approach, CCS'18
Quinonez et. al., SAVIOR: Securing Autonomous Vehicles with Robust Physical Invariants, Usenix Security'20

SRR[RAID'20]: Recovery Approach



Choi et. al., Software Sensor based Real-Time Recovery from Sensor Attacks on Robotic Vehicles. RAID'2020

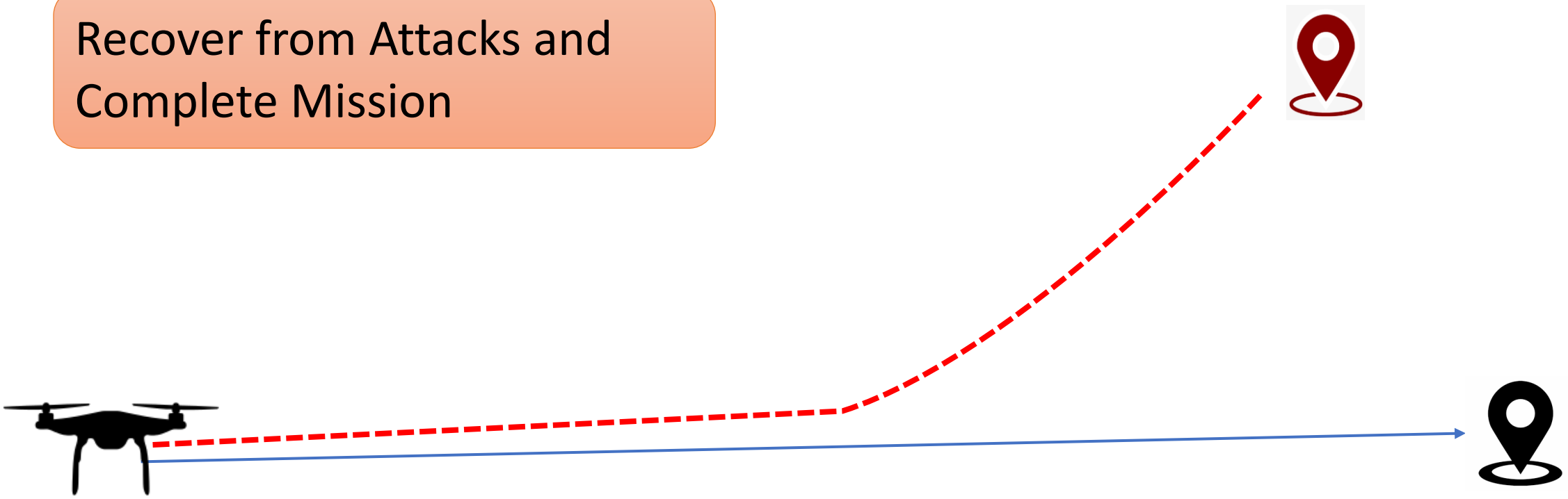
Remediation is not enough…



Choi et. al., Software Sensor based Real-Time Recovery from Sensor Attacks on Robotic Vehicles. RAID'2020

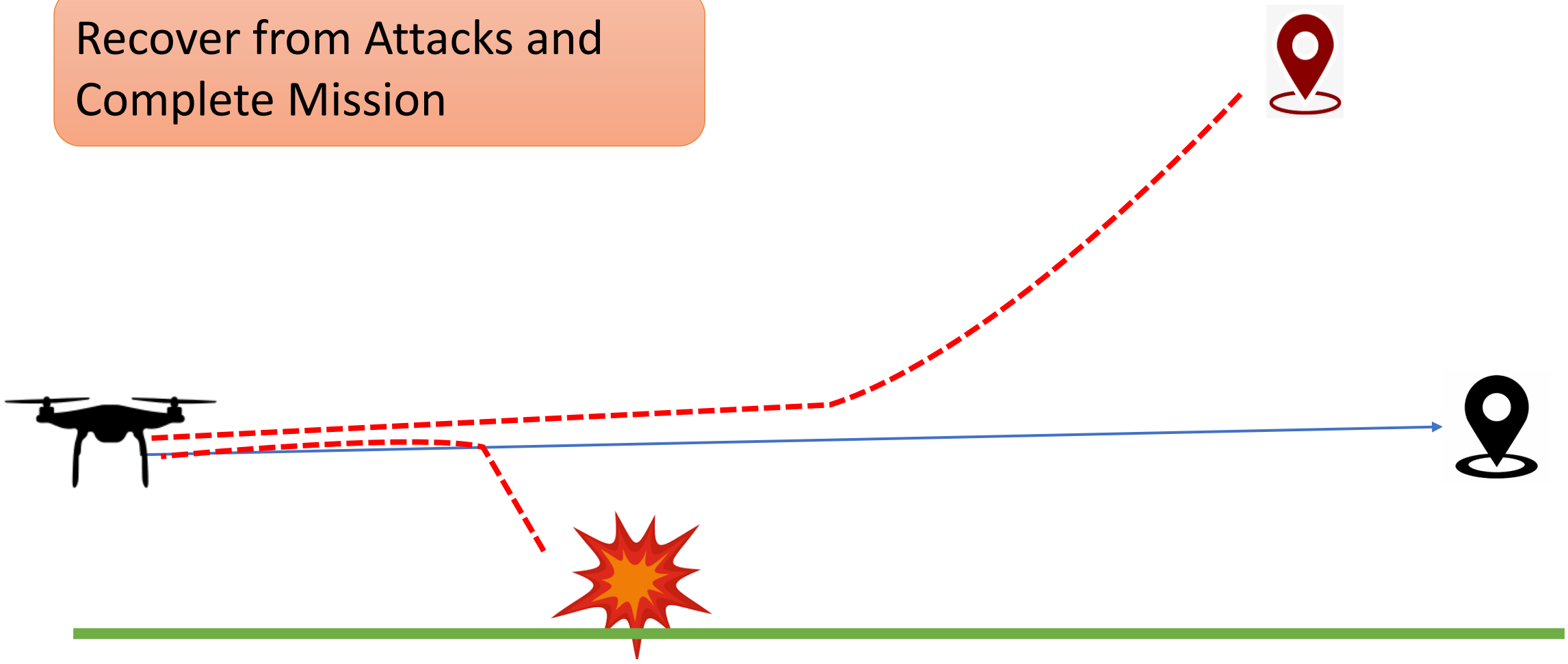
Goal

Recover from Attacks and Complete Mission



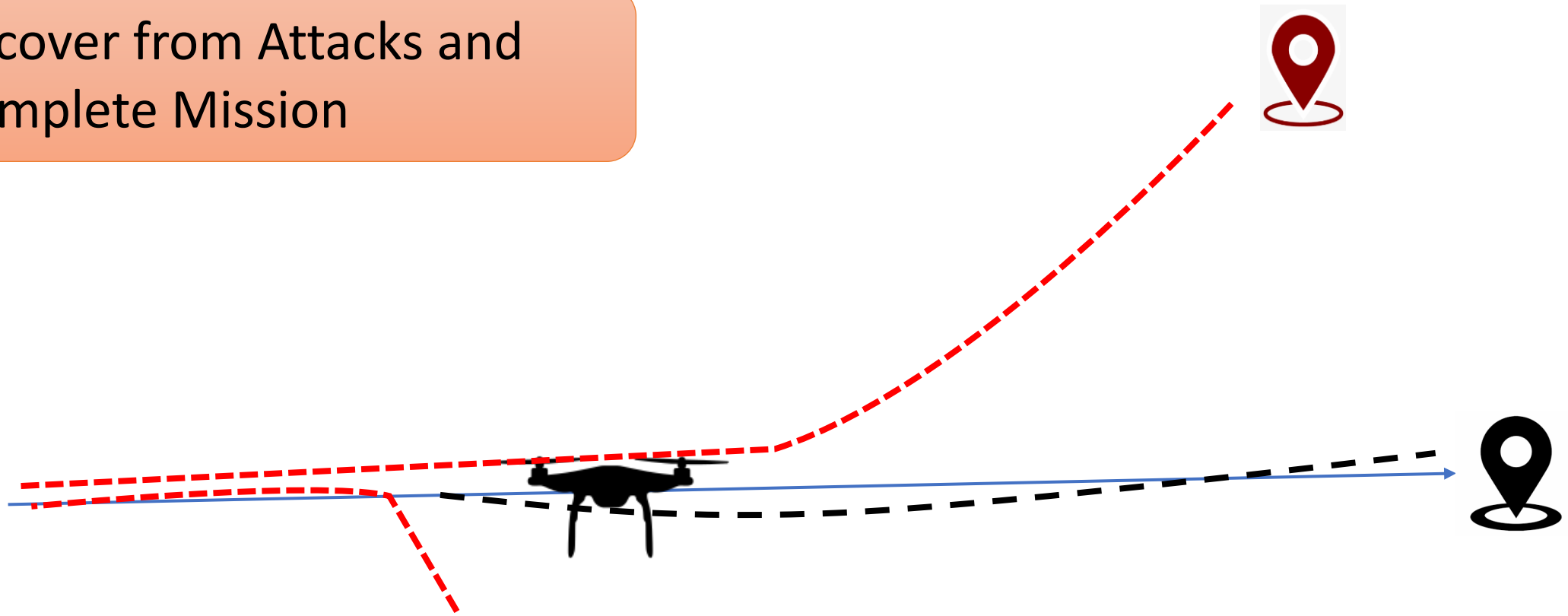
Goal

Recover from Attacks and Complete Mission



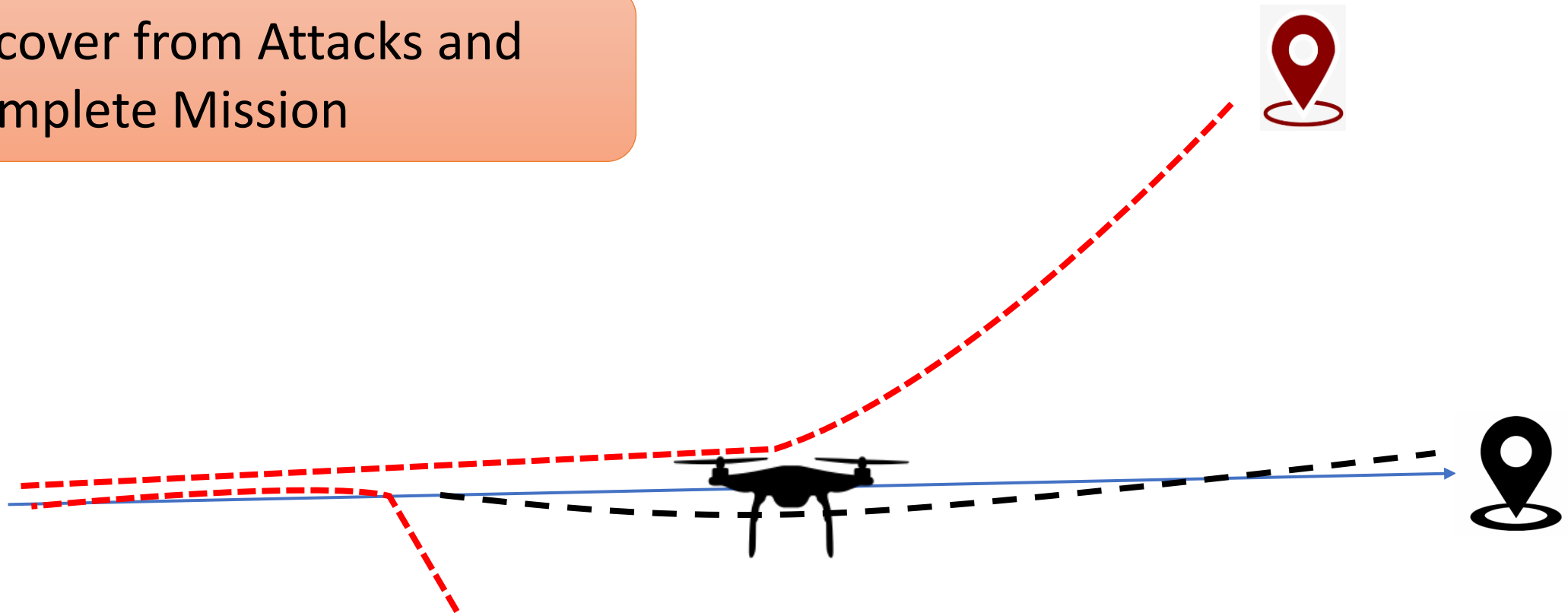
Goal

Recover from Attacks and Complete Mission



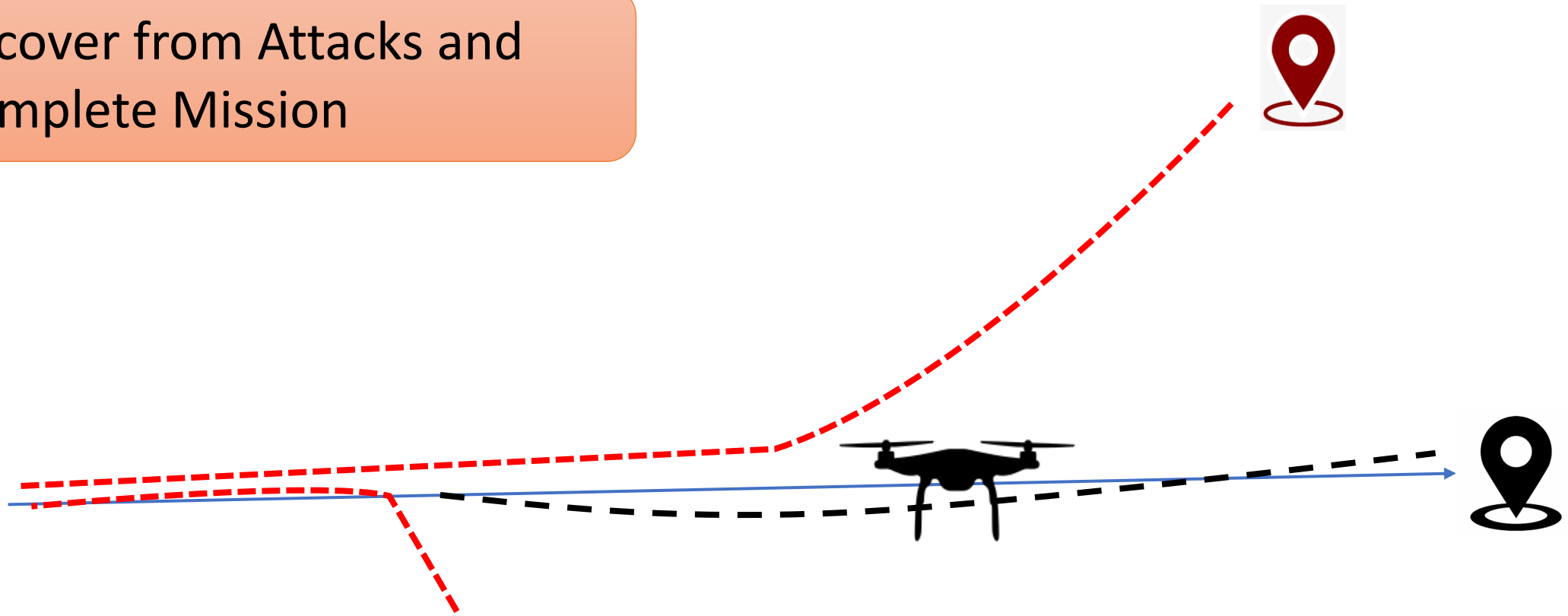
Goal

Recover from Attacks and Complete Mission



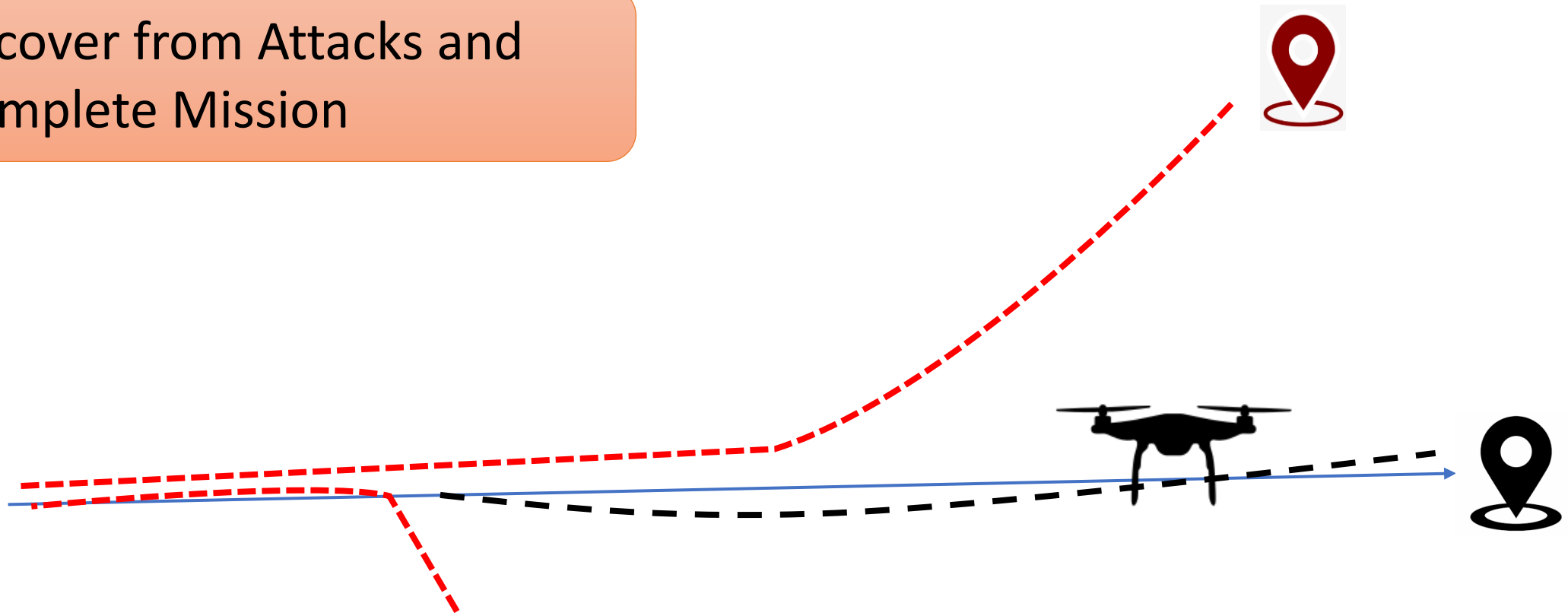
Goal

Recover from Attacks and Complete Mission



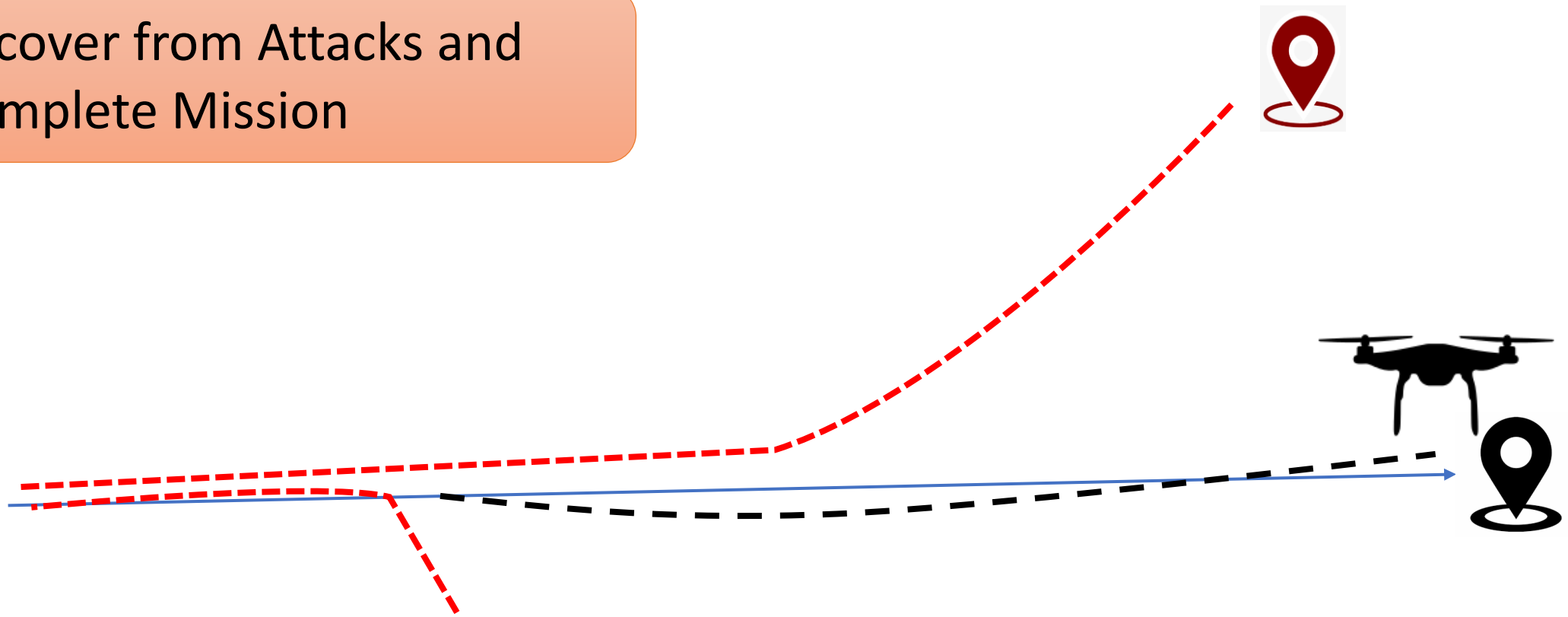
Goal

Recover from Attacks and Complete Mission



Goal

Recover from Attacks and Complete Mission



Goals

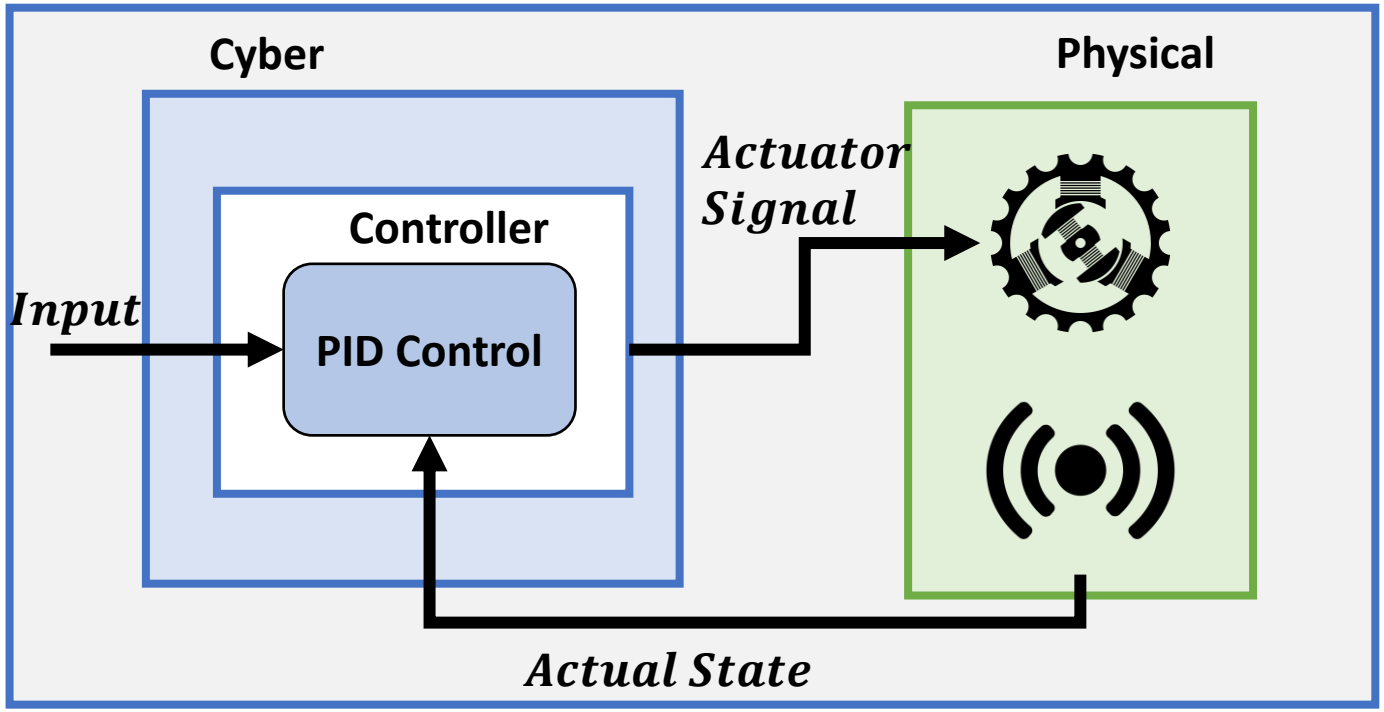
Recover from Attacks and Complete Mission

Limit impacts of Stealthy Attacks

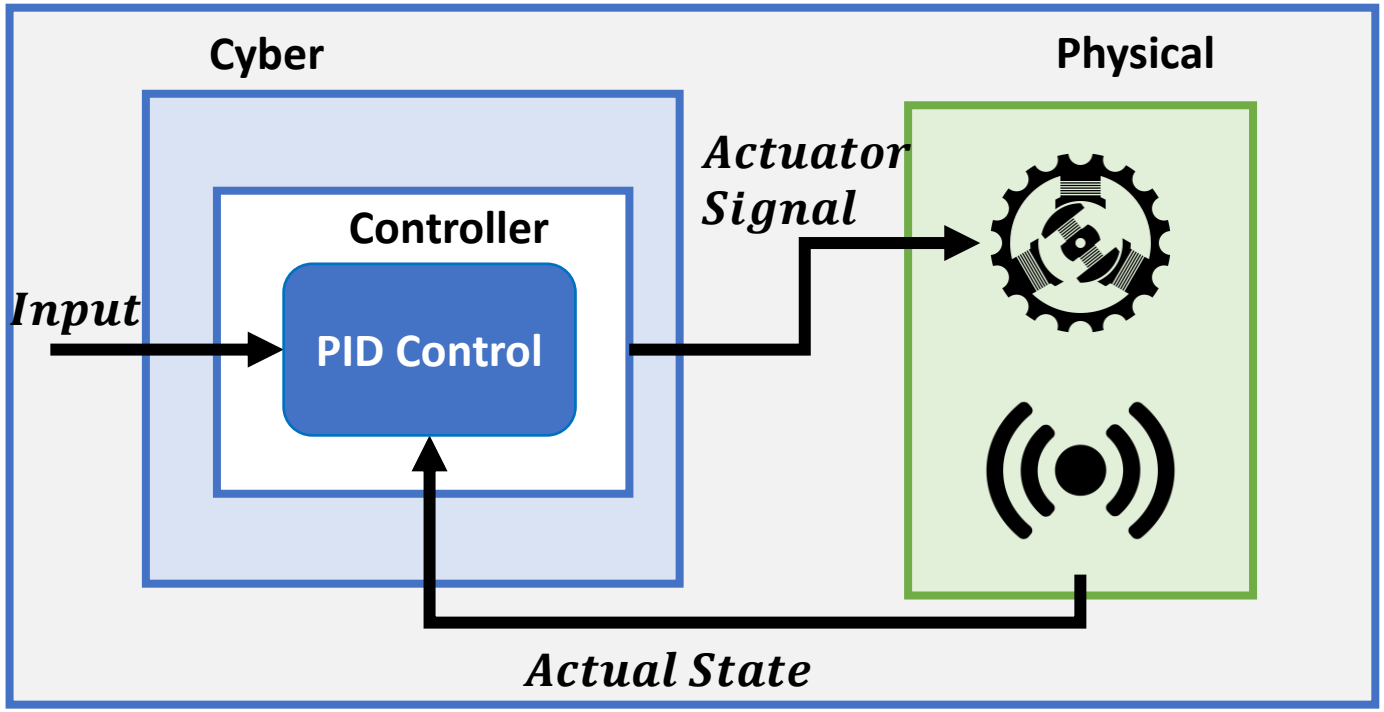
PID-Piper

Feed-Forward Control
Long Short-Term Memory (LSTM)

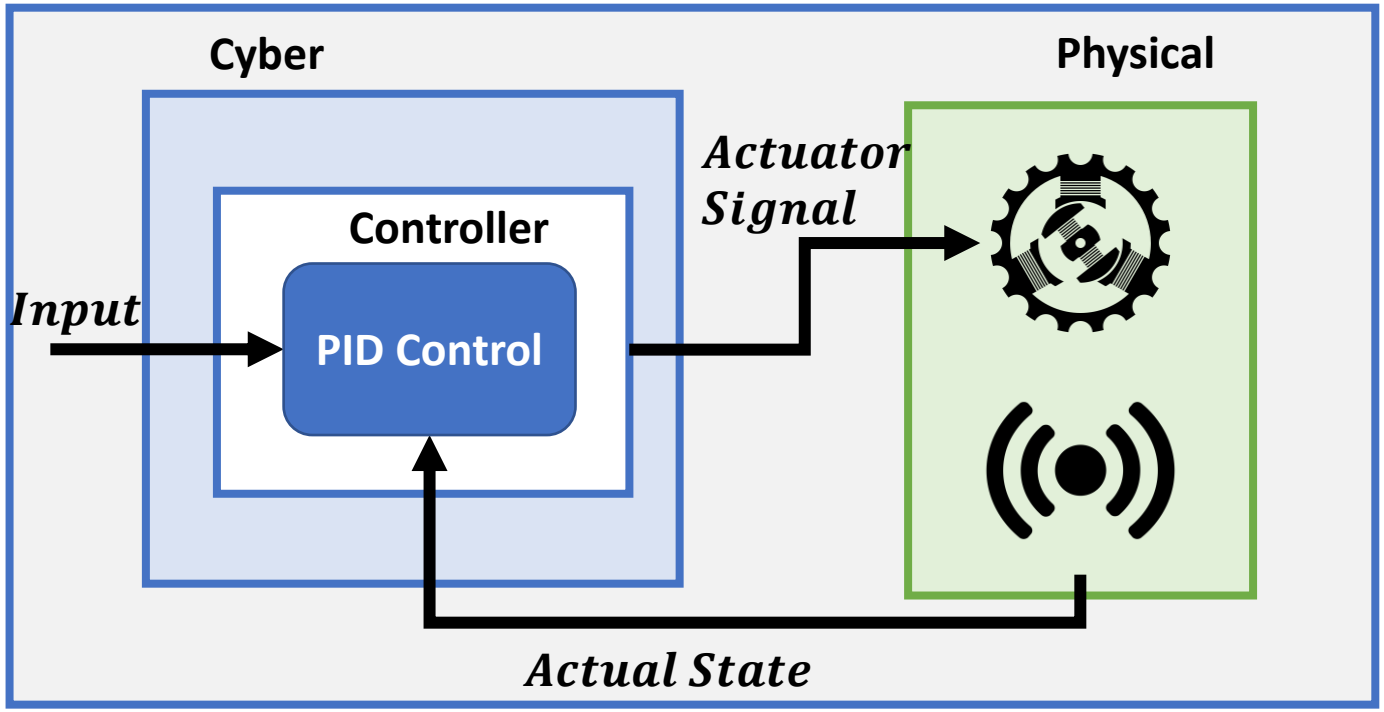
Sensor → PID Control → Actuator Signal



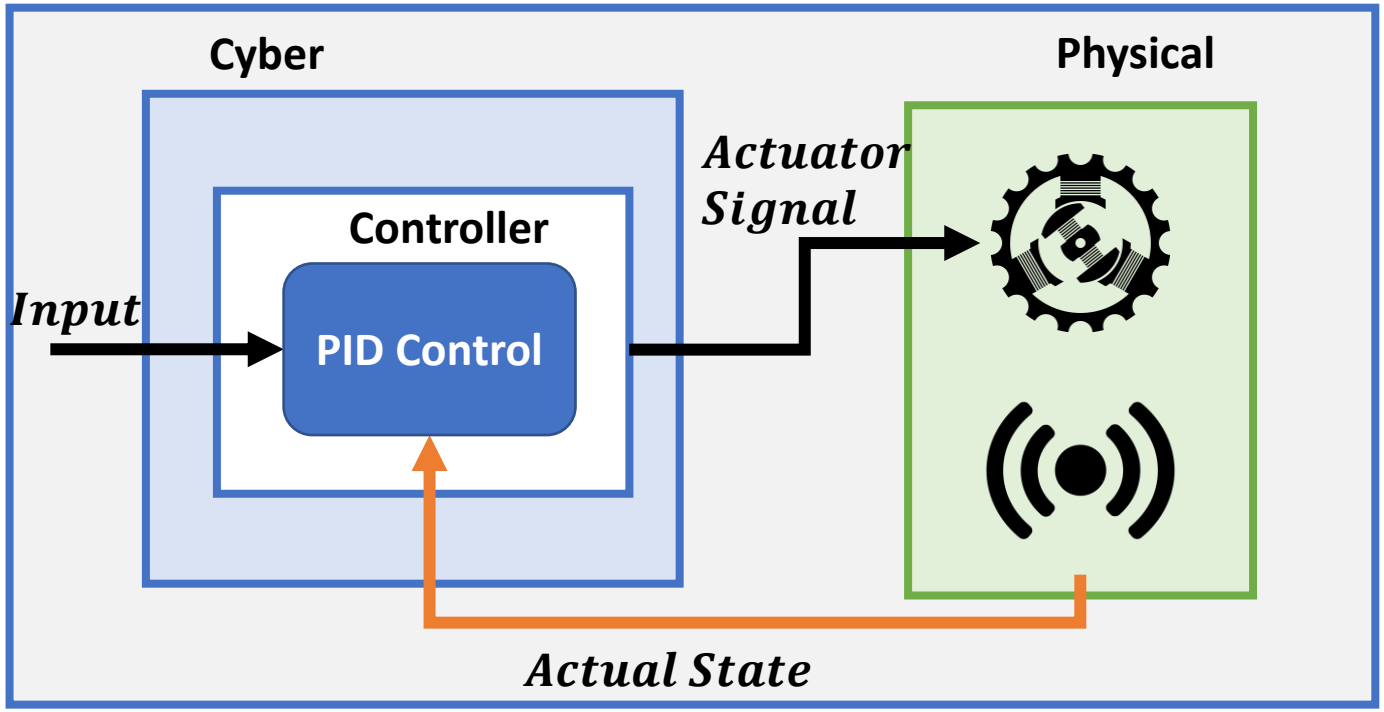
Sensor → PID Control → Actuator Signal

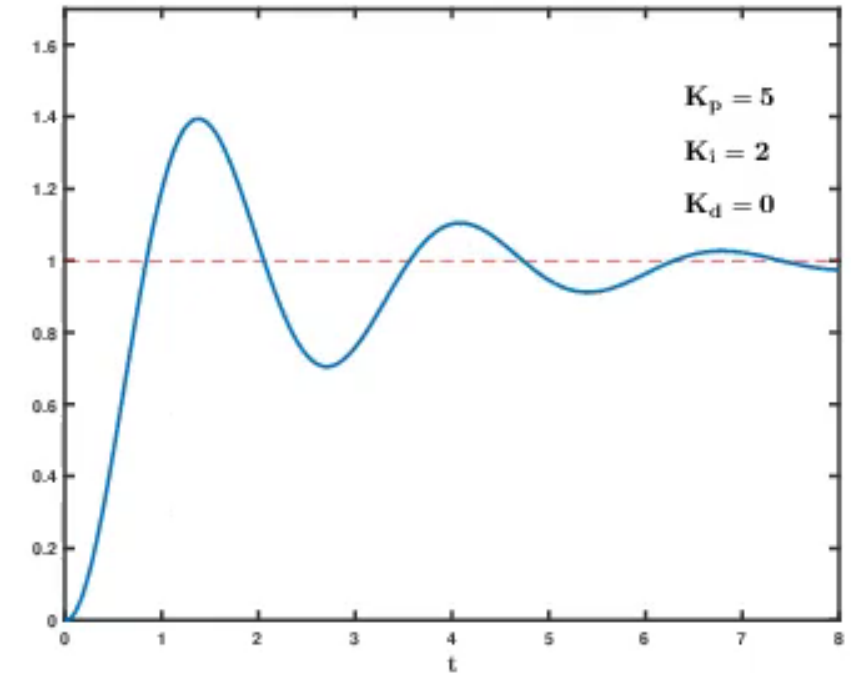


PID Compensation

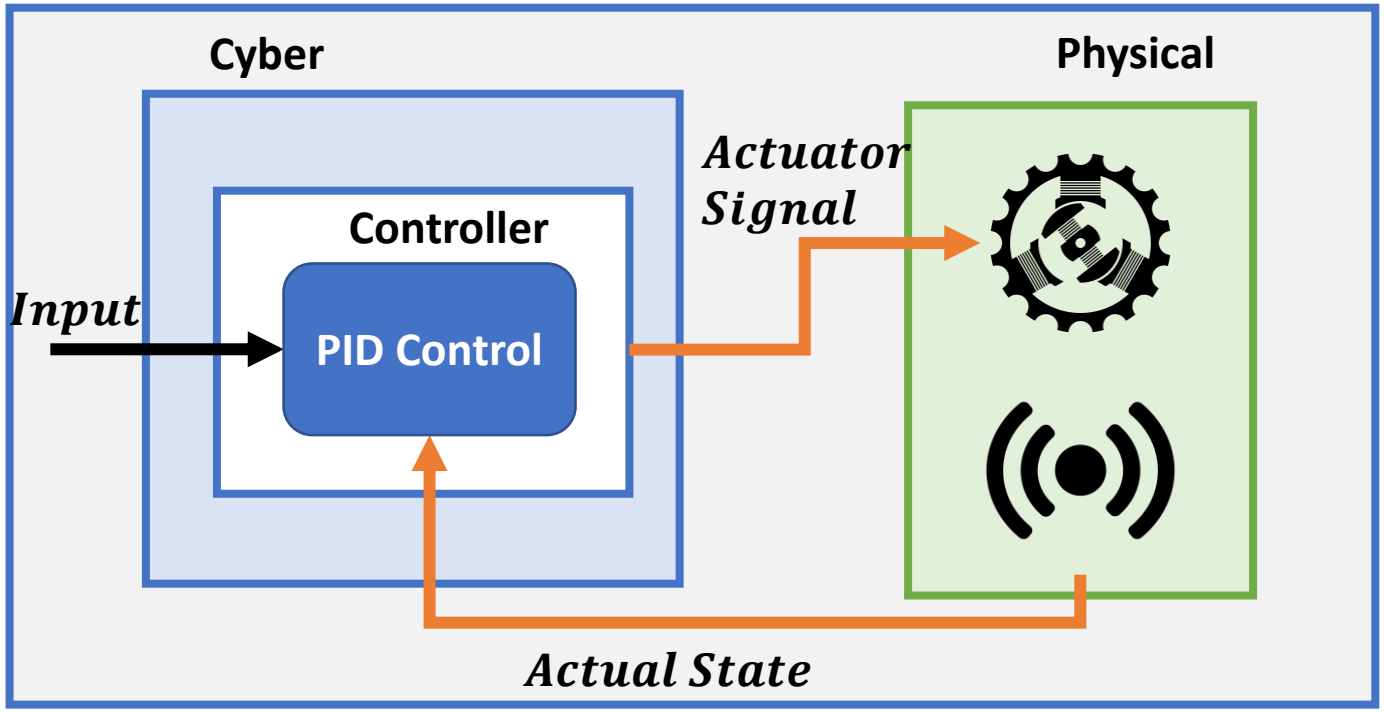


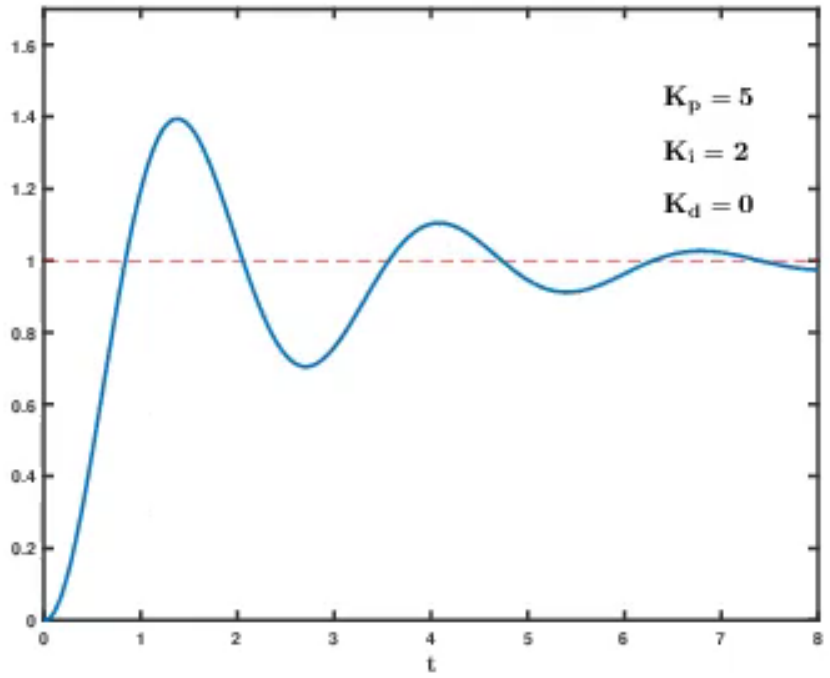
PID Compensation



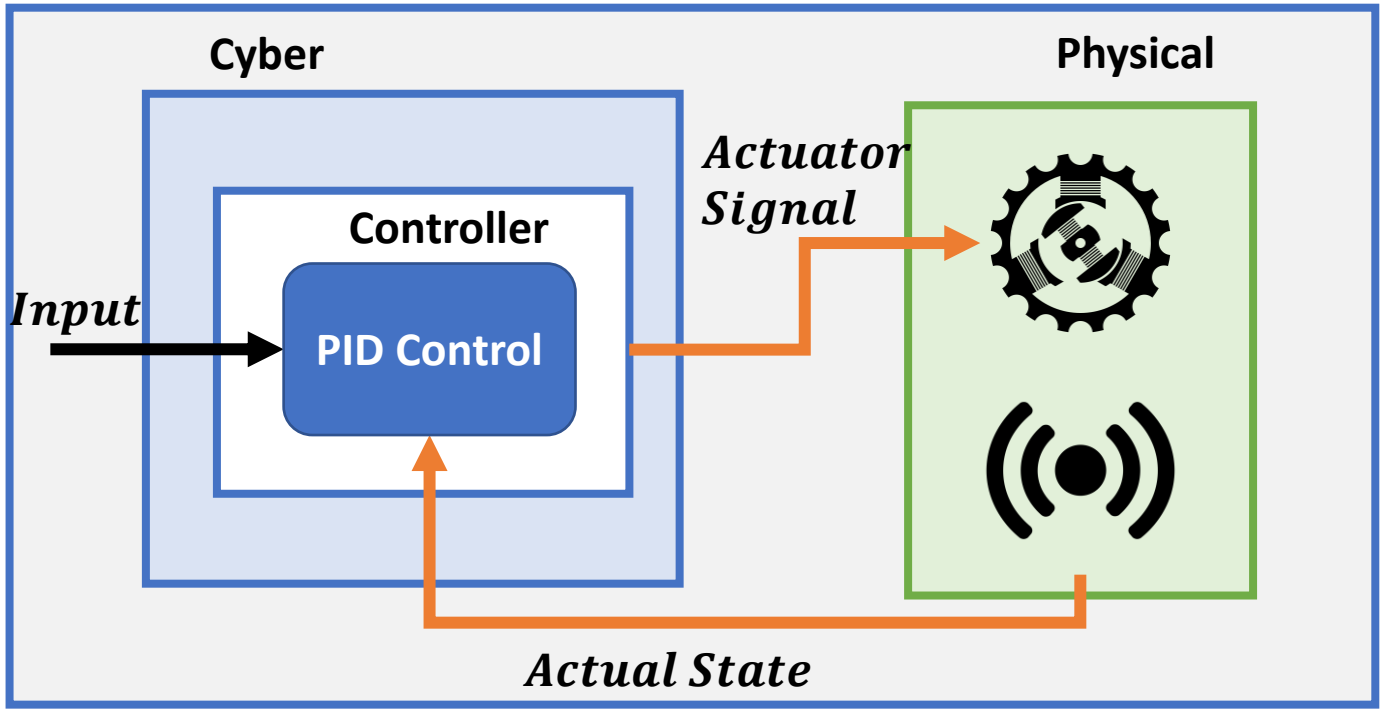


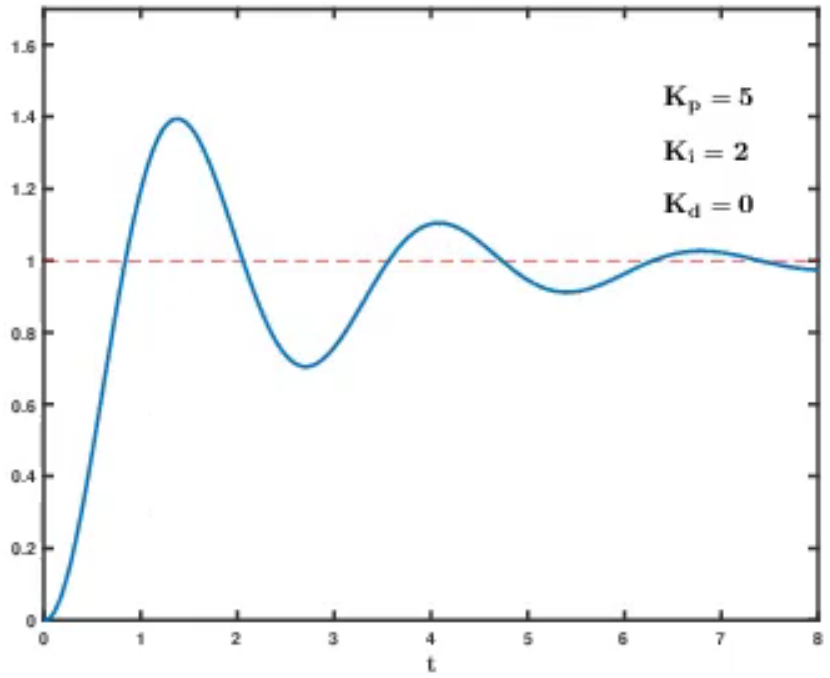
PID Compensation



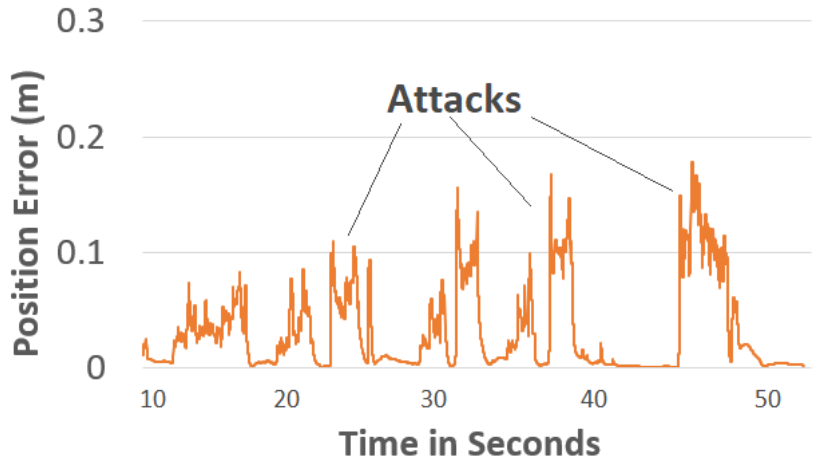


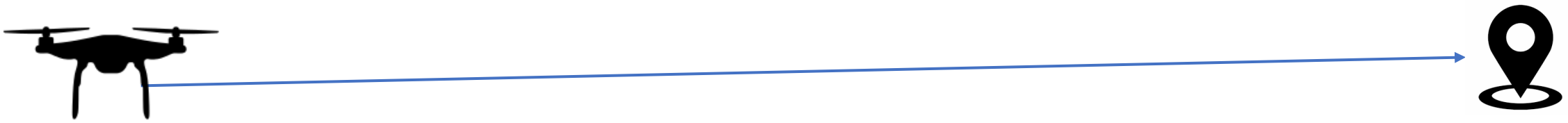
PID Compensation



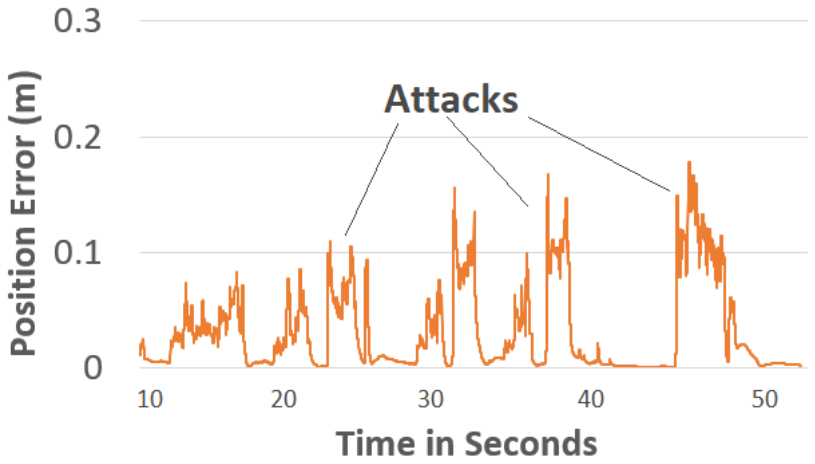


PID Compensation under Attacks

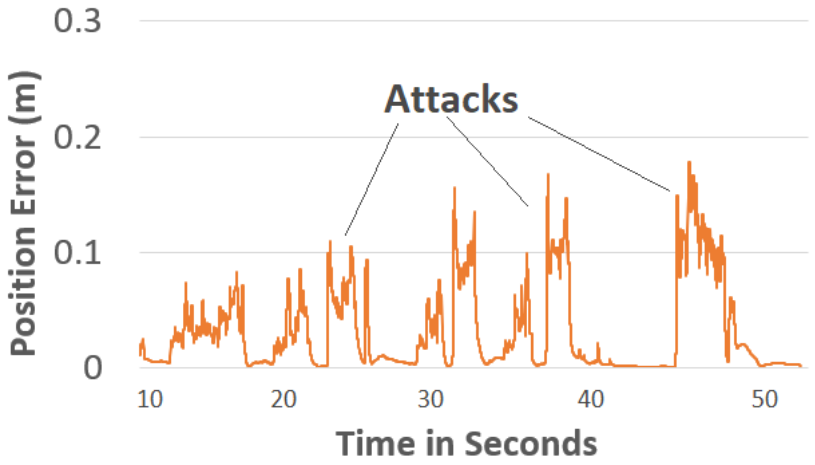


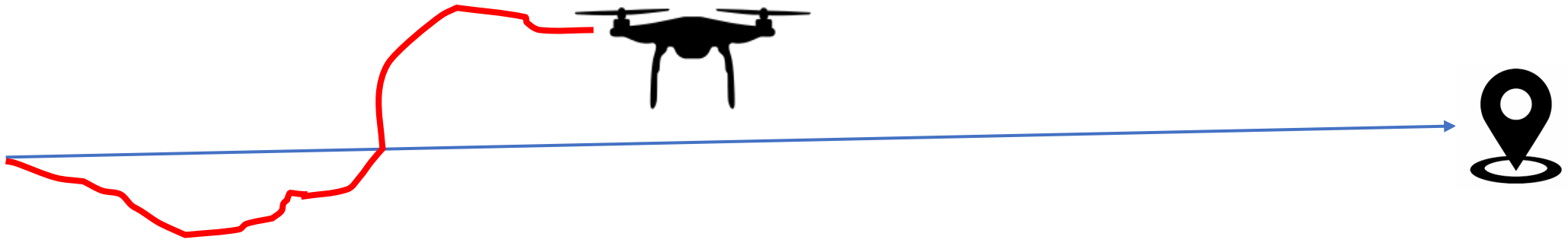


PID Over-Compensates under Attacks

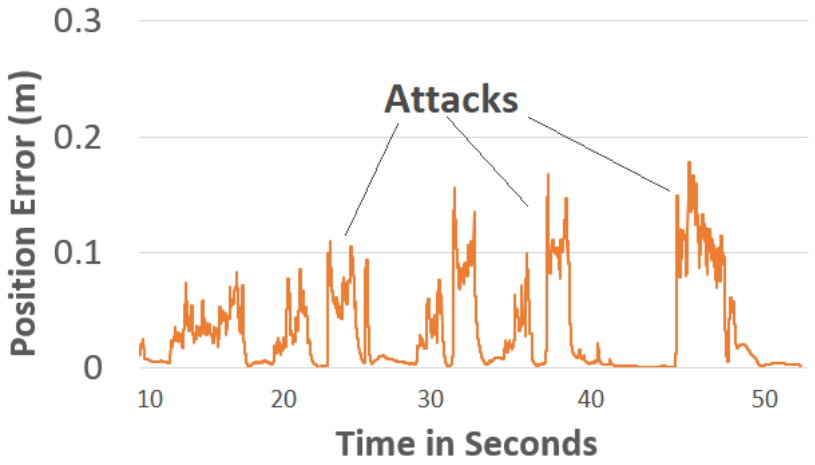


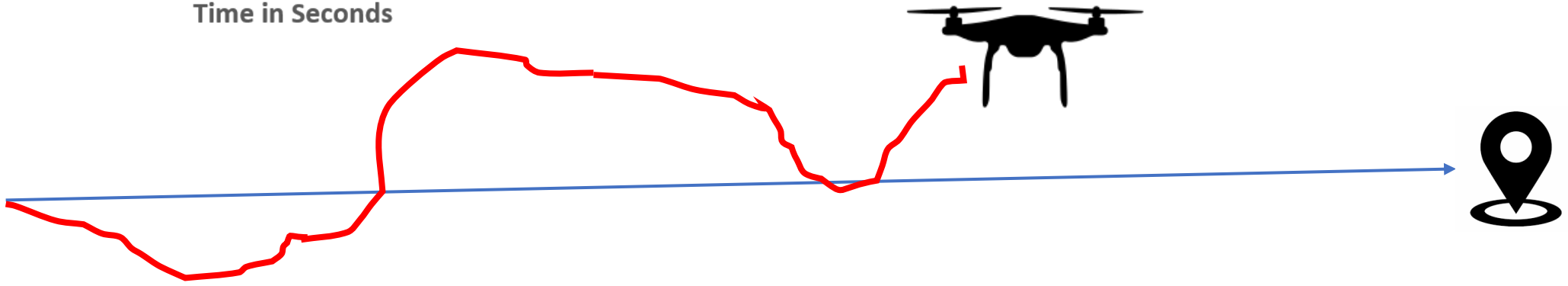
PID Over-Compensates under Attacks



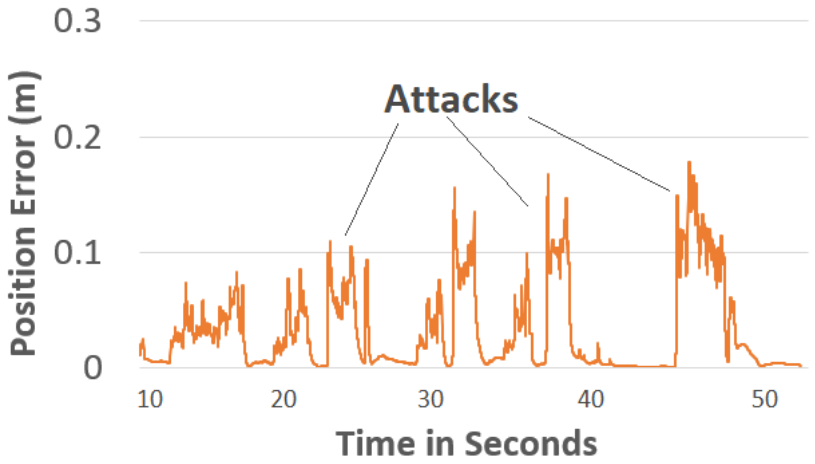


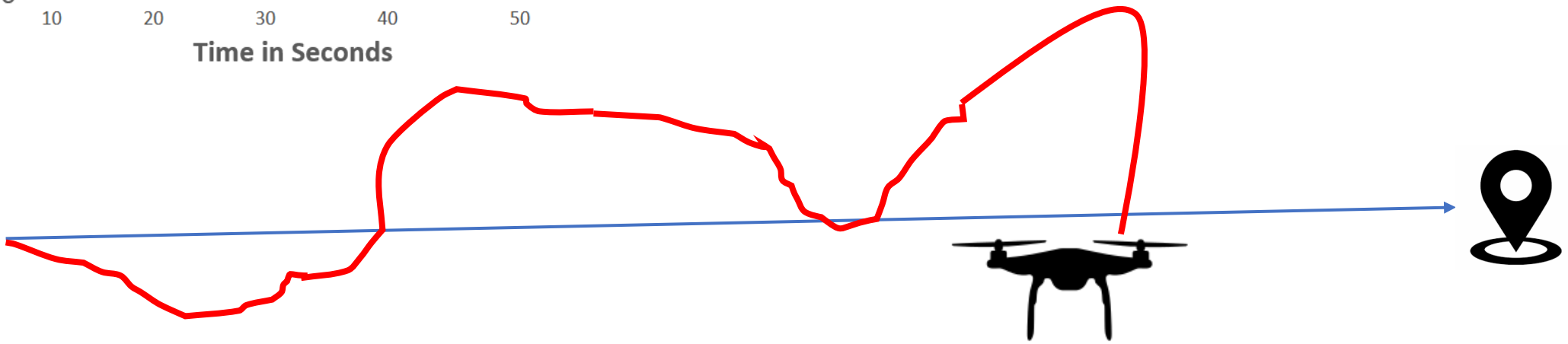
PID Over-Compensates under Attacks



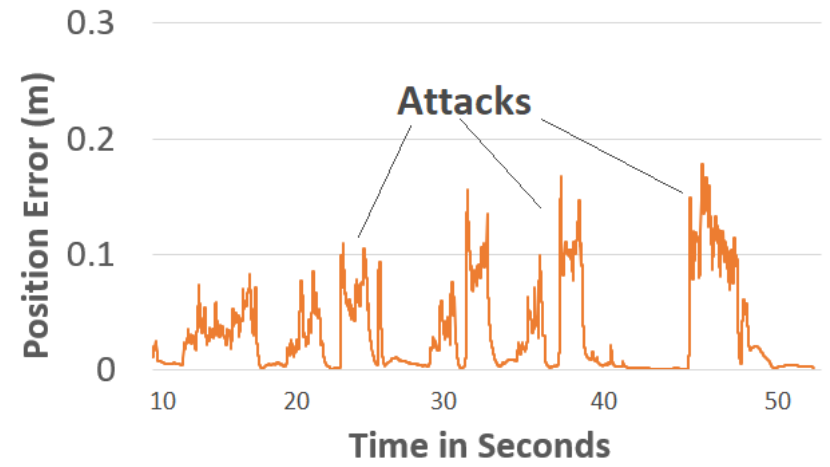


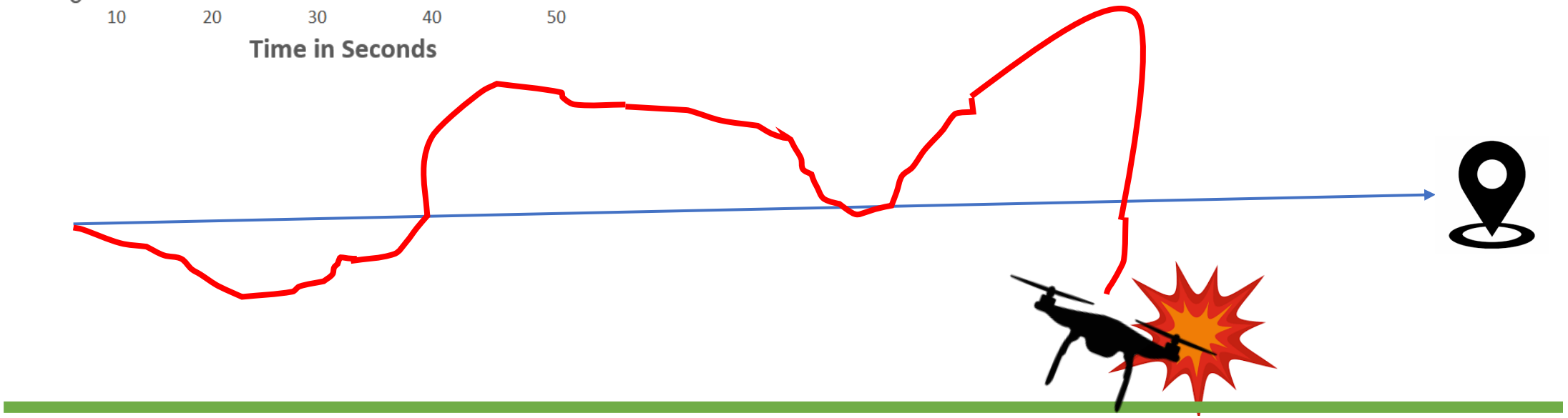
PID Over-Compensates under Attacks



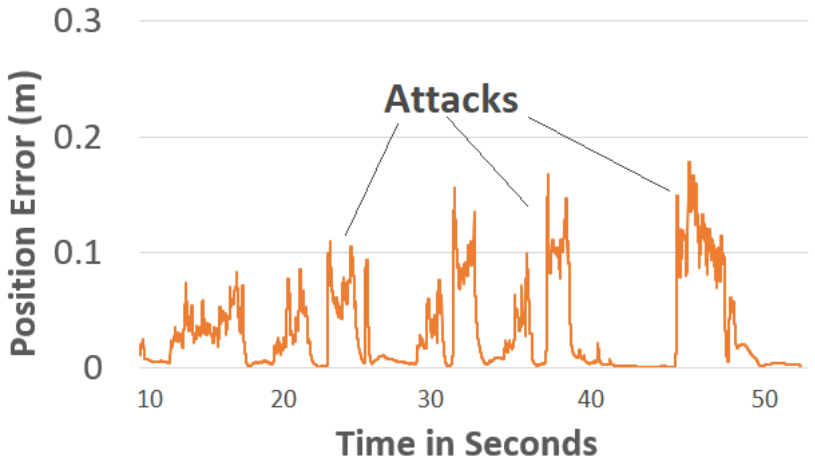


PID Over-Compensates under Attacks





PID Over-Compensates under Attacks

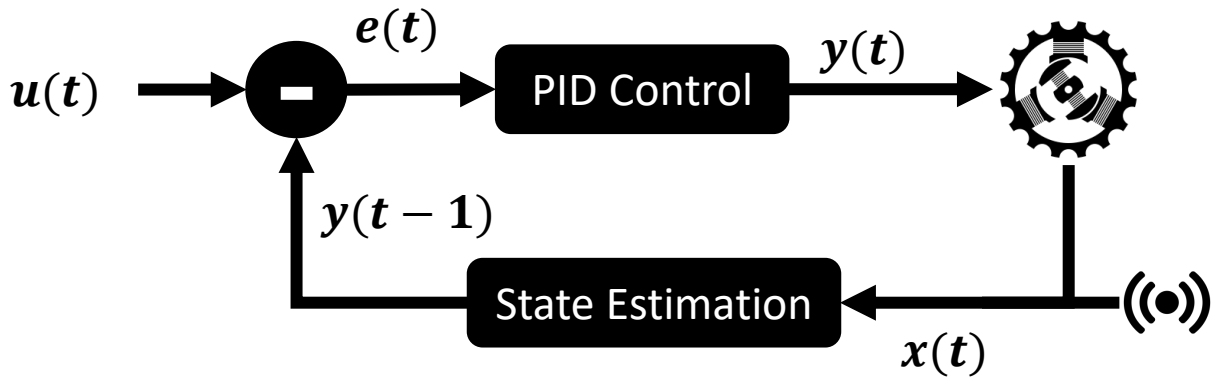


PID compensation → handling faults ✓

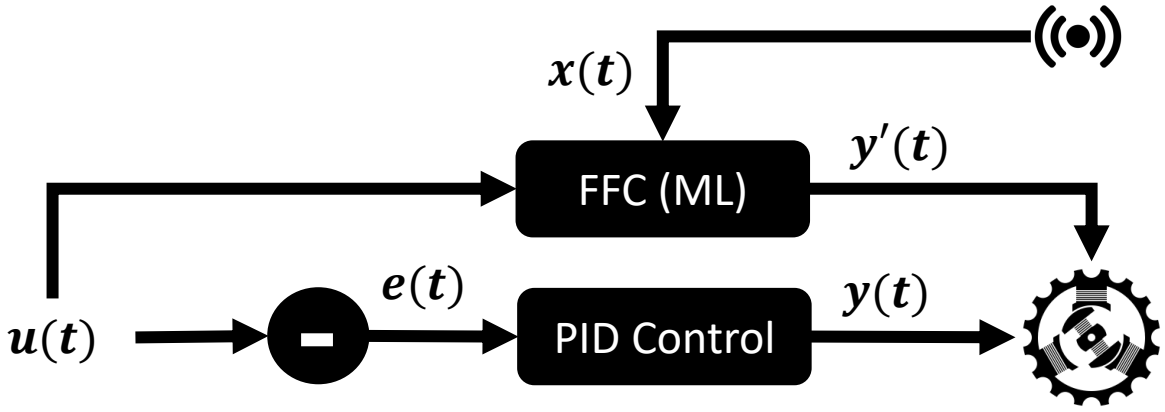
under attacks ✗

Recovering RVs from Attacks

PID Control

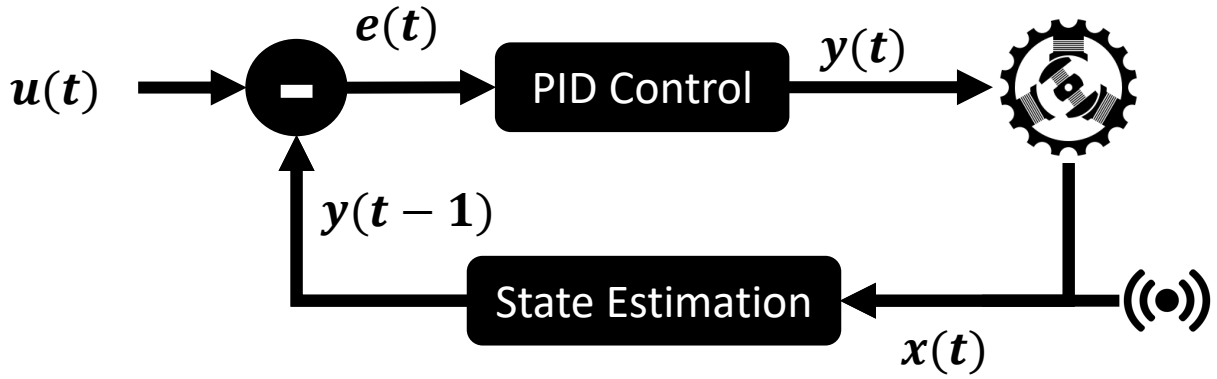


Feedforward Control

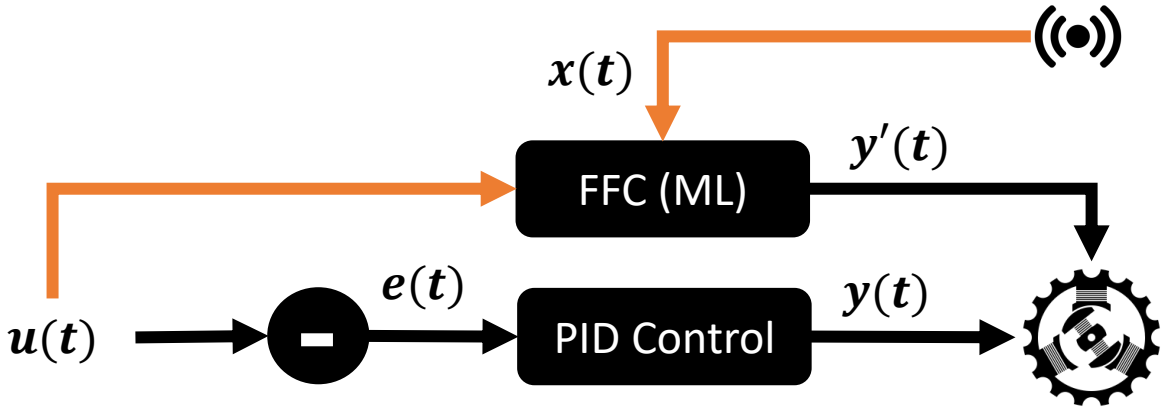


PID-Piper Recovery Approach

PID Control

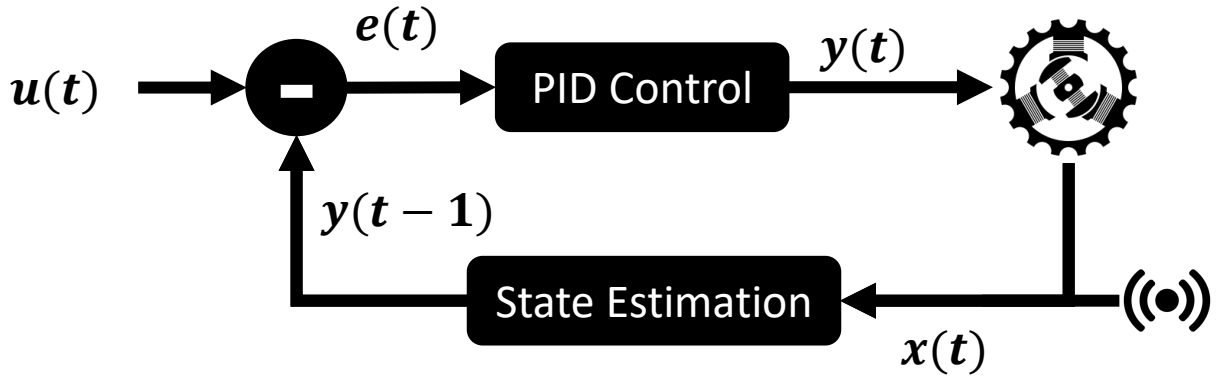


Feedforward Control

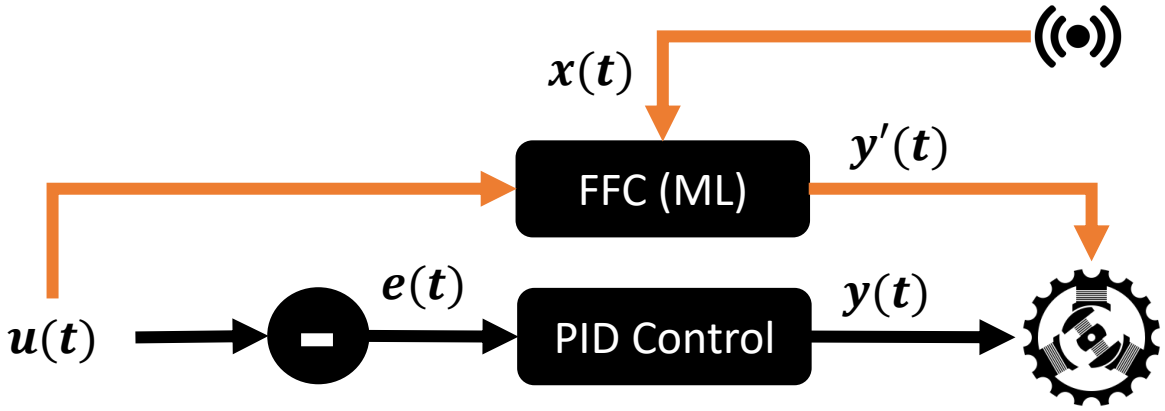


PID-Piper Recovery Approach

PID Control

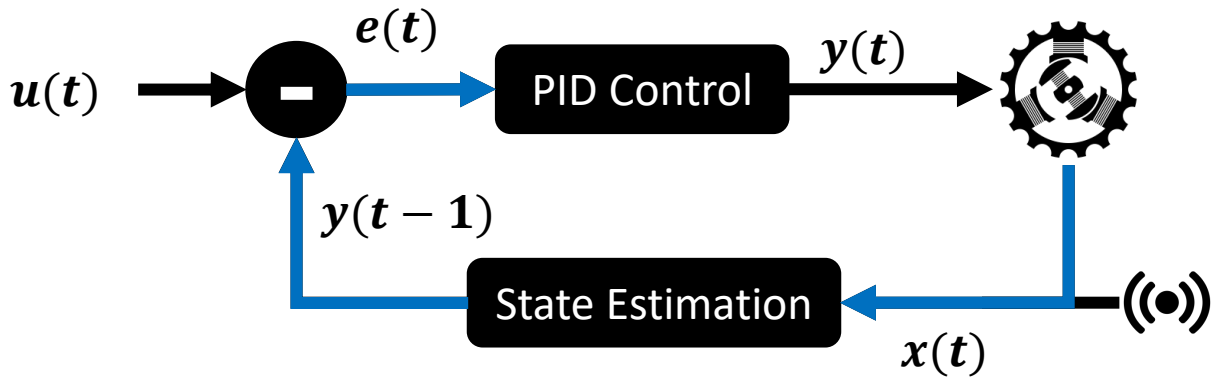


Feedforward Control

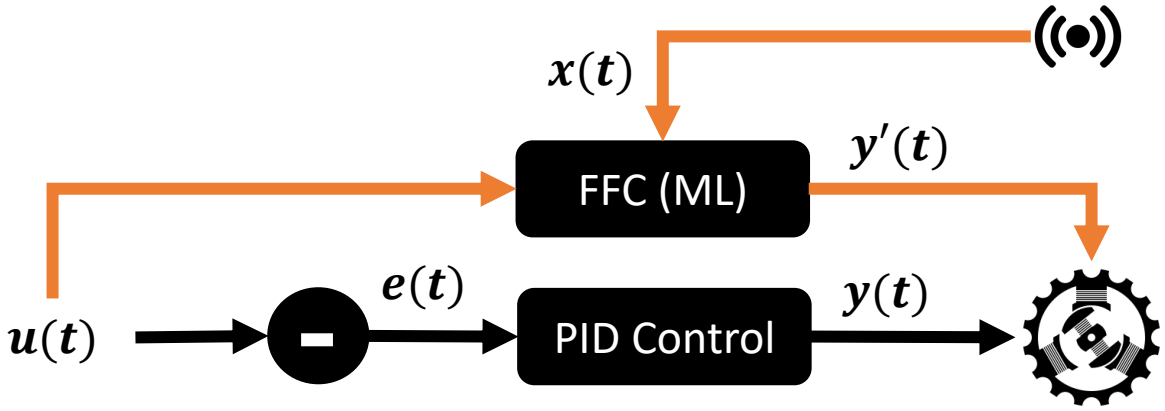


PID-Piper Recovery Approach

PID Control

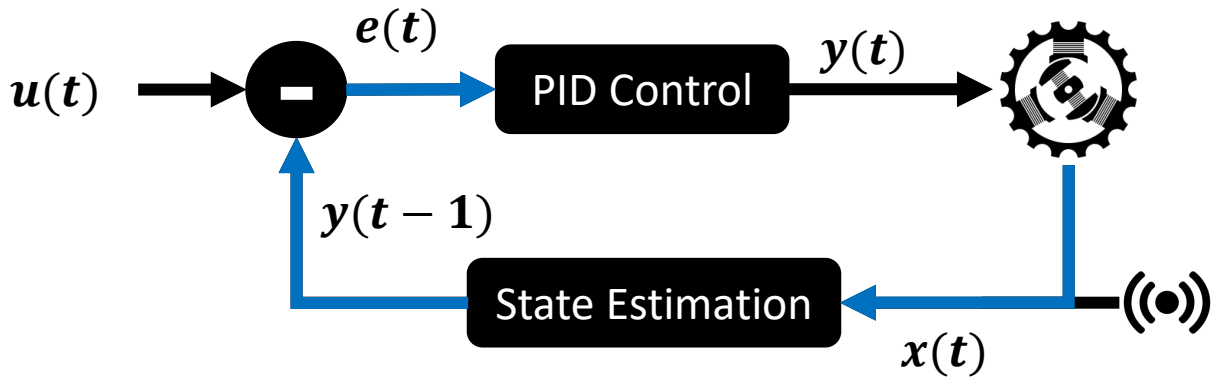


Feedforward Control

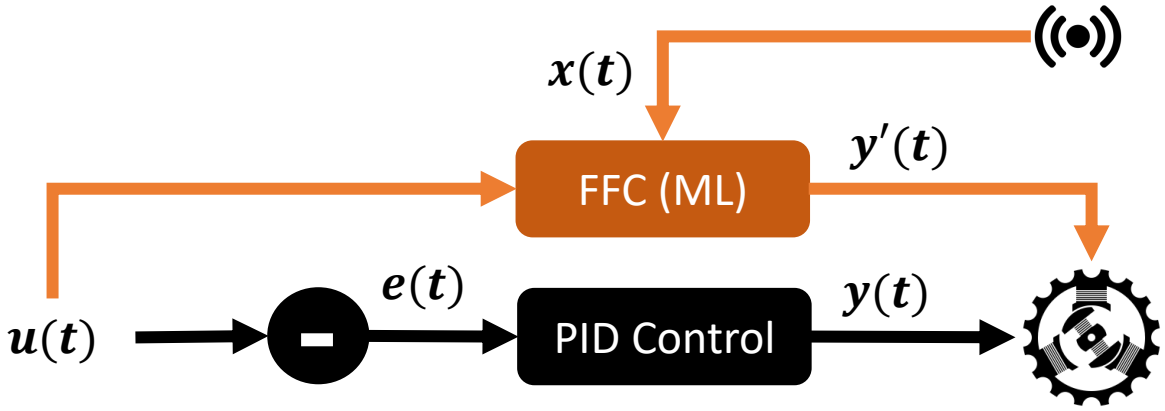


PID-Piper Recovery Approach

PID Control

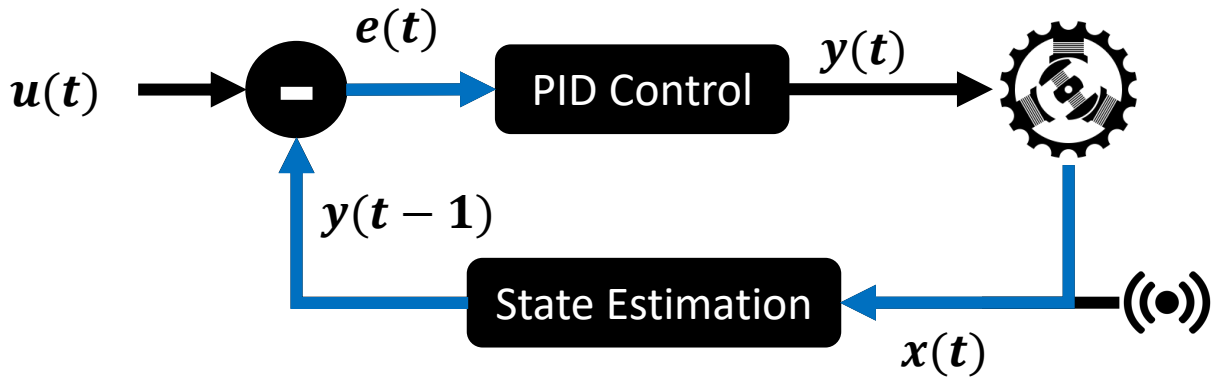


Feedforward Control

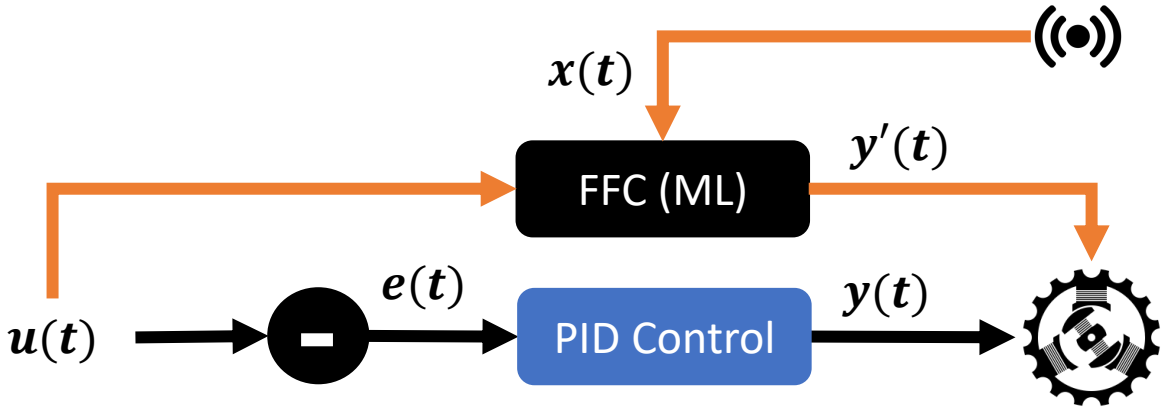


PID-Piper Recovery Approach

PID Control



Feedforward Control



Experimental Setup

ARDUPILOT

PX4 autopilot

Experimental Setup

PID-Piper Implementation

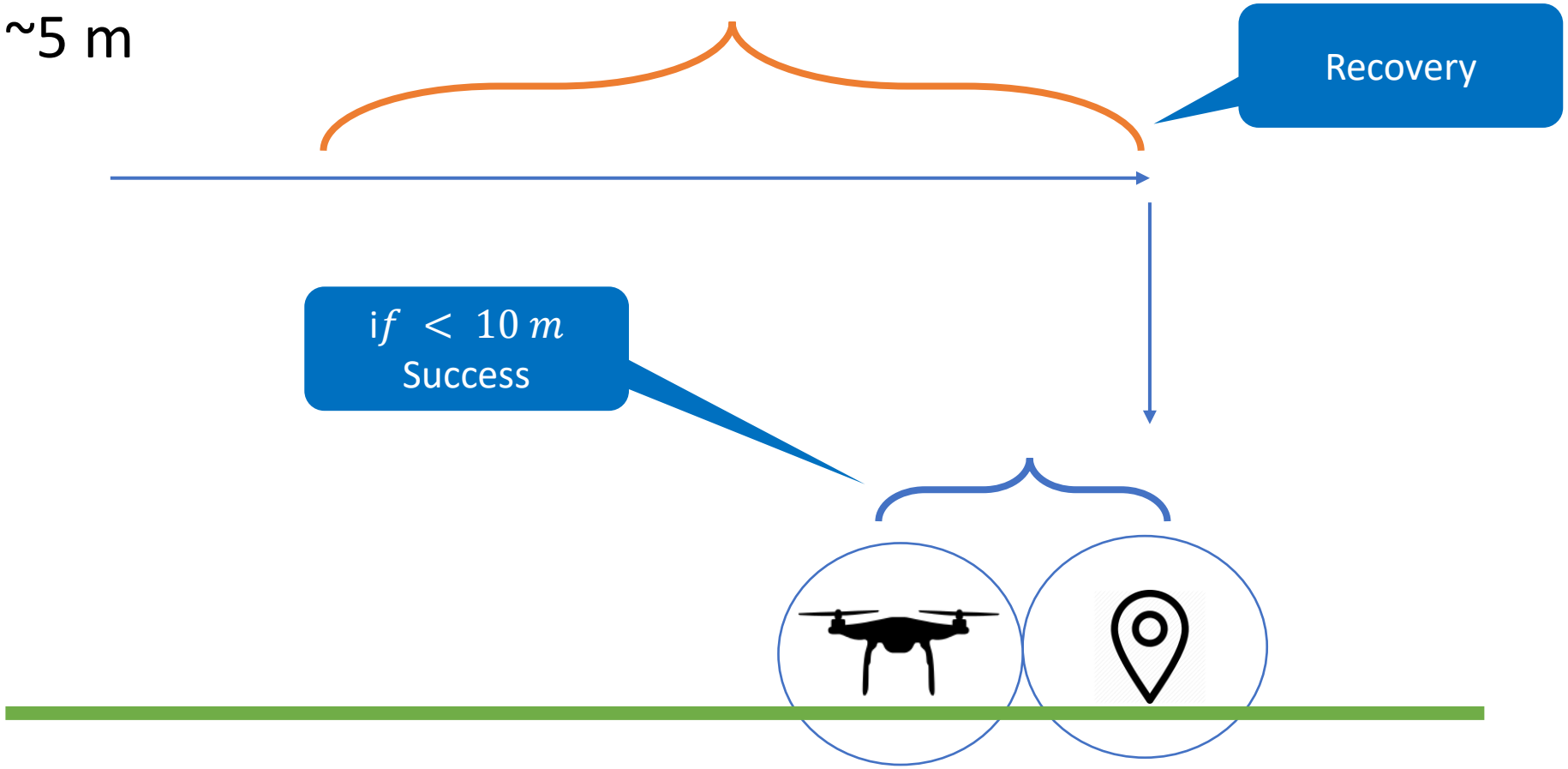
- FFC built using LSTM model (Python)
- Trained (Python)
- Plugged into Autopilot → Firmware (C++)

Training

- 30 RV mission profile data
- Circular, Polygonal, Straight line.

Metric for Mission Success

- GPS Offset ~5 m



PID-Piper: False Positives

Analysis Type	SRR [RAID'20]	PID-Piper [This work]
Recovery Activated	20%	10%
Missions Failed	50%	0%
FPR	**10%**	**0%**

$$FPR = \frac{Number\ of\ missions\ failed}{Total\ number\ of\ missions}$$

PID-Piper: False Positives

Analysis Type	SRR [RAID'20]	PID-Piper [This work]
Recovery Activated	20%	10%
Missions Failed	50%	0%
FPR	**10%**	**0%**

$$FPR = \frac{Number\ of\ missions\ failed}{Total\ number\ of\ missions}$$

PID-Piper: Recovery under Attacks

Analysis Type	SRR [RAID'20]	PID-Piper [This work]
Mission Success	13%	83%
Mission Failed (no Crash)	50%	17%
Crash/Stall	**37%**	**0%**

$$Mission\ Success = \frac{No.\ of\ missions\ with\ deviation < 10\ meters}{Total\ number\ of\ missions}$$

PID-Piper: Recovery under Attacks

Analysis Type	SRR [RAID'20]	PID-Piper [This work]
Mission Success	13%	83%
Mission Failed (no Crash)	50%	17%
Crash/Stall	**37%**	**0%**

$$Mission\ Success = \frac{No.\ of\ missions\ with\ deviation < 10\ meters}{Total\ number\ of\ missions}$$

PID-Piper: Recovery under Attacks

Analysis Type	SRR [RAID'20]	PID-Piper [This work]
Mission Success	13%	83%
Mission Failed (no Crash)	50%	17%
Crash/Stall	**37%**	**0%**

Recovery Successful in 83% of the cases with 0 crashes.

PID-Piper: Overheads

Analysis Type	PID-Piper [This work]
CPU Overhead	~7%
Energy Overhead	~0.9%
Mission delays	Negligible

Summary

Paper# 283

- **PID-Piper: A framework to recover Robotic Vehicles from attacks**
- Feed-forward Control to address overcompensation.
- 3 real and 3 simulated RV systems.
- 83% mission success from attacks, 0% false positives.

Artifacts https://github.com/DependableSystemsLab/pid-piper

Contact Pritam Dash, pdash@ece.ubc.ca

PID-Piper
Recovery Videos