Design and Implementation of Attack-Resilient Cyber-Physical Systems

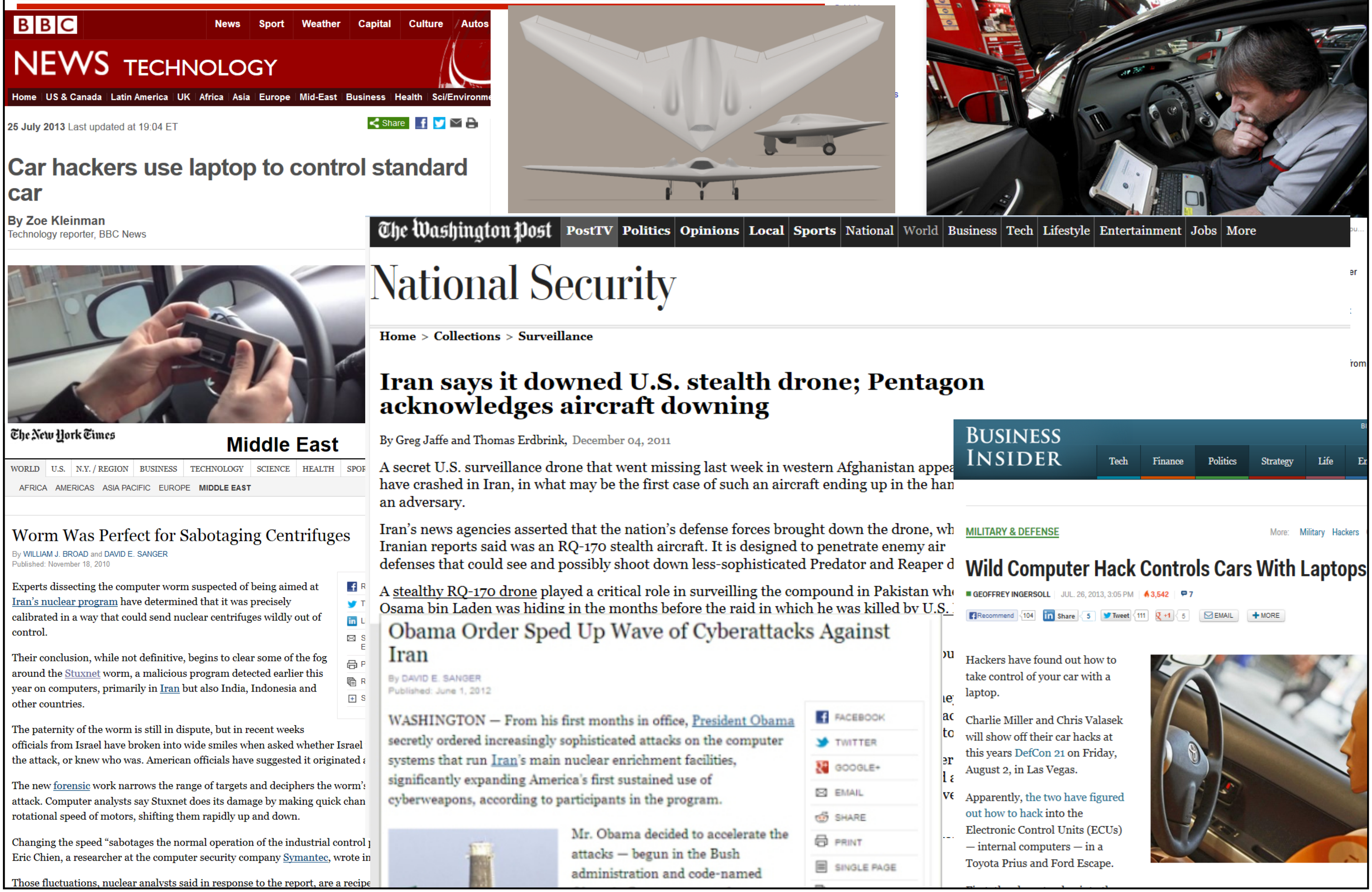
Miroslav Pajic, Nicola Bezzo, James Weimer
Oleg Sokolsky, George J. Pappas, Insup Lee

Precise Center
University of Pennsylvania

Penn Engineering

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Cyber-Physical Systems Security



Hackers find weaknesses in car computer systems

25 July 2013 Last updated at 19:04 ET

Car hackers use laptop to control standard car

By Zoe Kleinman
Technology reporter, BBC News

National Security

Home > Collections > Surveillance

Iran says it downed U.S. stealth drone; Pentagon acknowledges aircraft downing

By Greg Jaffe and Thomas Erdbrink, December 04, 2011

A secret U.S. surveillance drone that went missing last week in western Afghanistan appea have crashed in Iran, in what may be the first case of such an aircraft ending up in the han an adversary.

Iran's news agencies asserted that the nation's defense forces brought down the drone, wh Iranian reports said was an RQ-170 stealth aircraft. It is designed to penetrate enemy air defenses that could see and possibly shoot down less-sophisticated Predator and Reaper d

A stealthy RQ-170 drone played a critical role in surveilling the compound in Pakistan wh Osama bin Laden was hiding in the months before the raid in which he was killed by U.S.

Worm Was Perfect for Sabotaging Centrifuges

By WILLIAM J. BROAD and DAVID E. SANGER
Published: November 18, 2010

Experts dissecting the computer worm suspected of being aimed at Iran's nuclear program have determined that it was precisely calibrated in a way that could send nuclear centrifuges wildly out of control.

Their conclusion, while not definitive, begins to clear some of the fog around the Stuxnet worm, a malicious program detected earlier this year on computers, primarily in Iran but also India, Indonesia and other countries.

The paternity of the worm is still in dispute, but in recent weeks officials from Israel have broken into wide smiles when asked whether Israel the attack, or knew who was. American officials have suggested it originated

The new forensic work narrows the range of targets and deciphers the worm's attack. Computer analysts say Stuxnet does its damage by making quick chan rotational speed of motors, shifting them rapidly up and down.

Changing the speed "sabotages the normal operation of the industrial control Eric Chien, a researcher at the computer security company Symantec, wrote in

Those fluctuations, nuclear analysts said in response to the report, are a recipe

Obama Order Sped Up Wave of Cyberattacks Against Iran

By DAVID E. SANGER
Published: June 1, 2012

WASHINGTON — From his first months in office, President Obama secretly ordered increasingly sophisticated attacks on the computer systems that run Iran's main nuclear enrichment facilities, significantly expanding America's first sustained use of cyberweapons, according to participants in the program.

Mr. Obama decided to accelerate the attacks — begun in the Bush administration and code-named

MILITARY & DEFENSE

More: Military Hackers

Wild Computer Hack Controls Cars With Laptops

GEOFFREY INGERSOLL | JUL. 26, 2013, 3:05 PM | 3,542 | 7

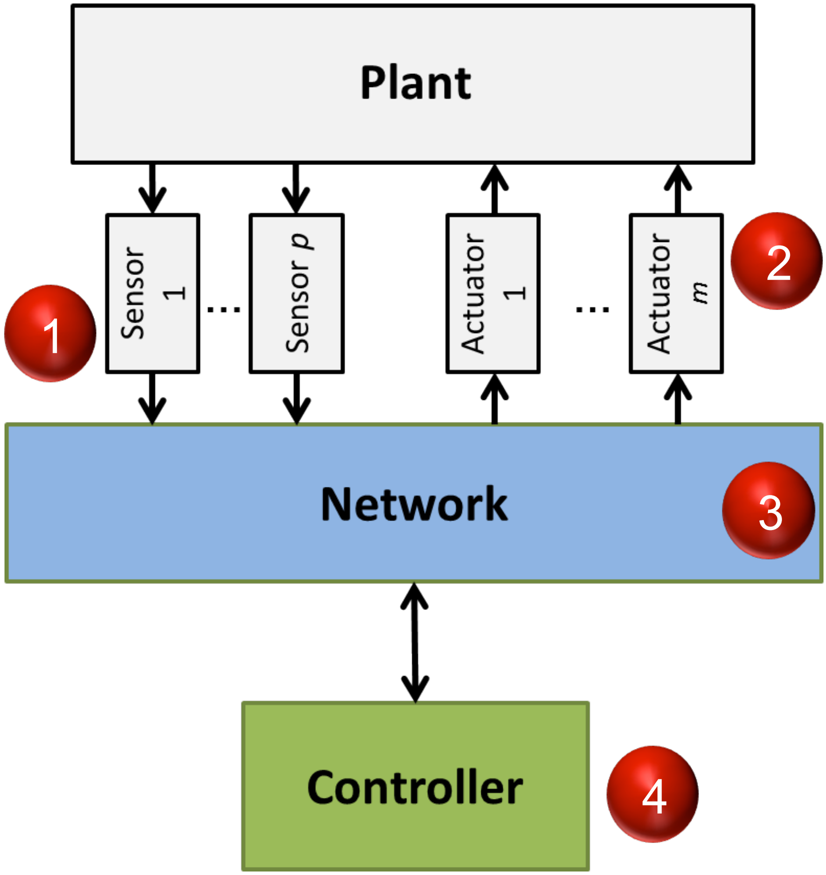
Hackers have found out how to take control of your car with a laptop.

Charlie Miller and Chris Valasek will show off their car hacks at this years DefCon 21 on Friday, August 2, in Las Vegas.

Apparently, the two have figured out how to hack into the Electronic Control Units (ECUs) — internal computers — in a Toyota Prius and Ford Escape.

CPS Attack Classes

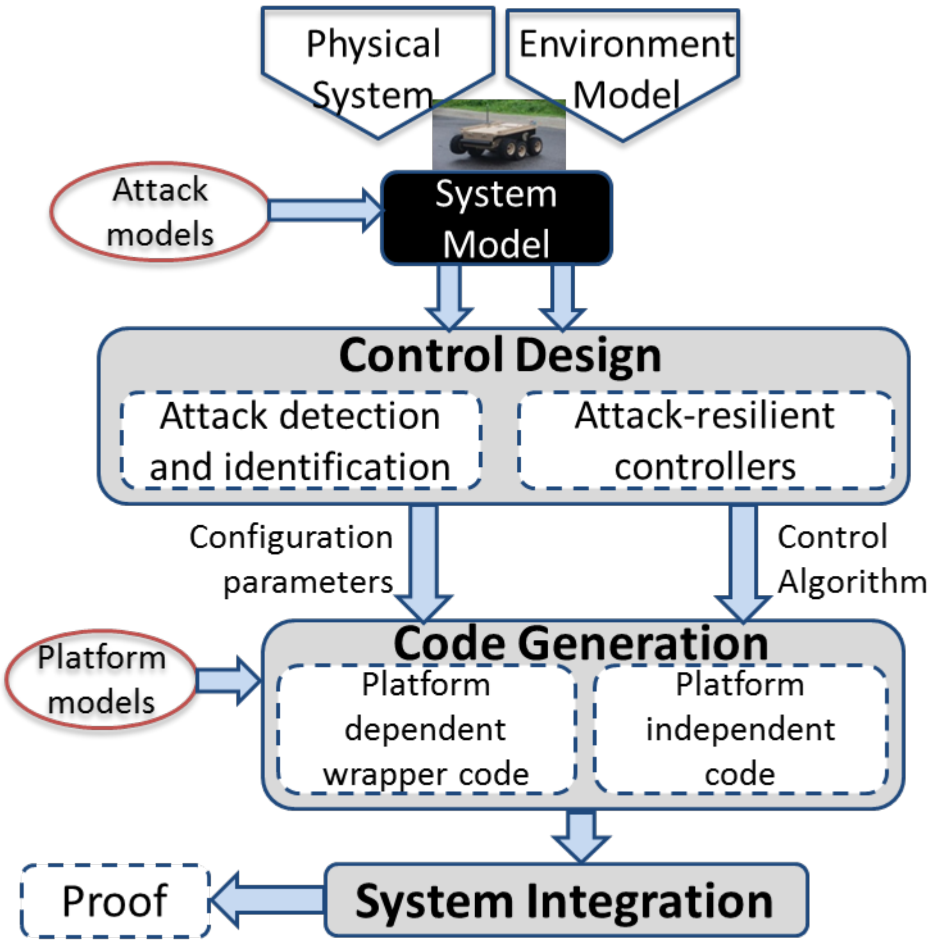
1. Sensor attacks
   - The attacker can arbitrarily change sensor measurements.
2. Actuator attacks
   - The attacker can arbitrarily change actuator values.
3. Communication attacks
   - The attacker can change messages between sensors and controllers, and messages between controllers and actuators.
4. Controller attacks
   - The attacker can change the controllers' parameters (e.g., execution model) or even the controllers' code.



Synthesis of Secure and Attack-Resilient Cyber-Physical Systems

Goal: Develop tools and techniques to ensure that cyber-physical systems maintain a degree of control even when the system is under cyber and/or **physical** attack

- Overall approach



– Control-level techniques

- Attack detection and identification using redundant sensing and model of the system's dynamics
- Attack-resilient control architectures

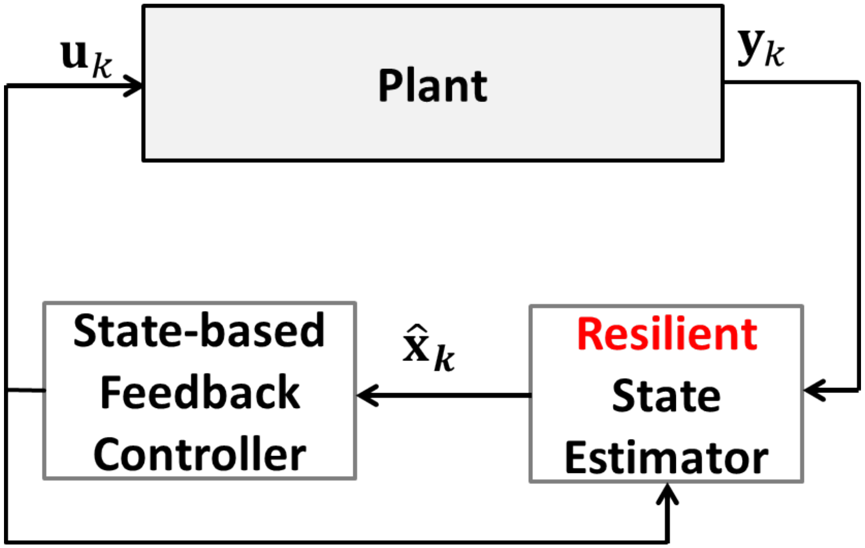
– Code-level techniques

- Ensure that the control code is correctly implemented and **integrated**!
- Preventing malicious code injection into the controller

Why attack-resilient state estimators?

- Attack-resilient control of Cyber-Physical Systems
  - Idea: Design attack-resilient state estimators

- Until now - required an accurate LTI system model
  - Fawzi *et al.* 2012
  - Pasqualetti *et al.* 2013

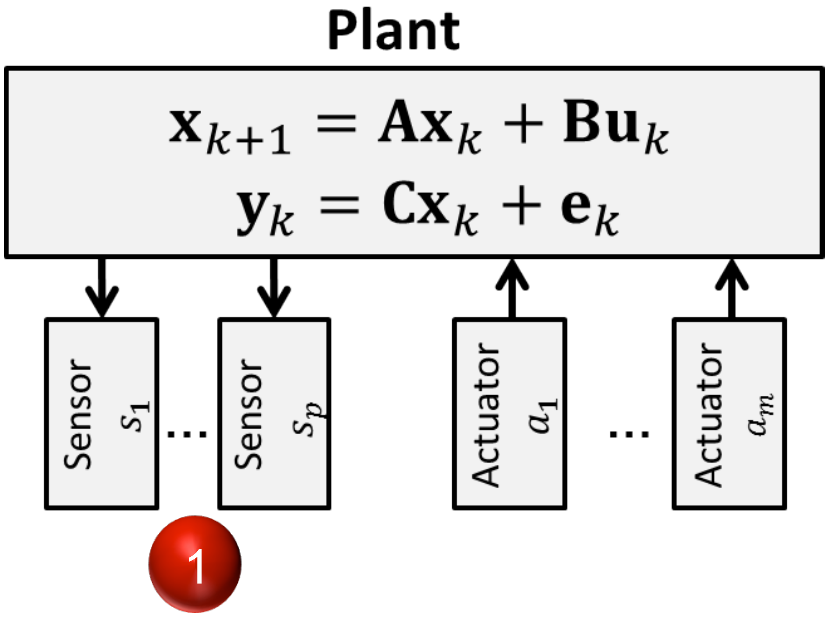


- If the number of attacked sensors is below a threshold, state can be reconstructed from a history of sensor readings
  - Also identifies sensors under attack

Modeling attacks on sensors and *actuators*

- Consider an LTI system
  - $\mathbf{x}_k \in \mathbb{R}^n$ plant's state at time k
  - $\mathbf{u}_k \in \mathbb{R}^m$ plant input at time k
  - $\mathbf{y}_k \in \mathbb{R}^p$ plant output
    - state information is availably only via sensors measurements

$$\mathcal{S} = \{s_1, s_2, \dots, s_p\}$$



$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$
$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{e}_k$$

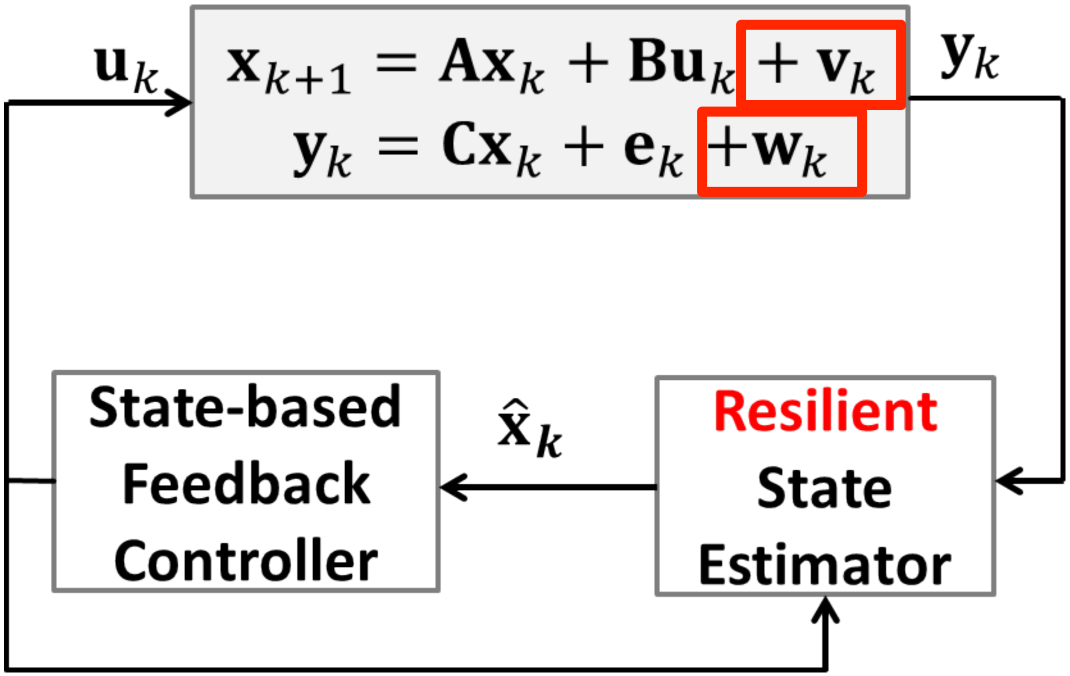
$$\mathcal{K} = \{s_2, s_5\}$$

$$\mathbf{e}_k = \begin{bmatrix} 0 \\ \boxed{1.7} \\ 0 \\ 0 \\ \boxed{-9} \end{bmatrix}$$

- Attacks on sensors in $\mathcal{K} = \left\{s_{i_1}, \dots, s_{i_{\boxed{q}}}\right\} \subseteq \mathcal{S}$
  - modeled with attack vector $\mathbf{e}_k$
  - $\mathbf{e}_{k,i} \neq 0 \iff$ sensor s_i is under attack at time k

Deterministic system with known model?

- In practice we have modeling errors
  - Process and measurement noise
  - Implementation effects - including jitter, latencies, etc



$\mathbf{u}_k$

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \boxed{+ \mathbf{v}_k}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{e}_k \boxed{+\mathbf{w}_k}$$

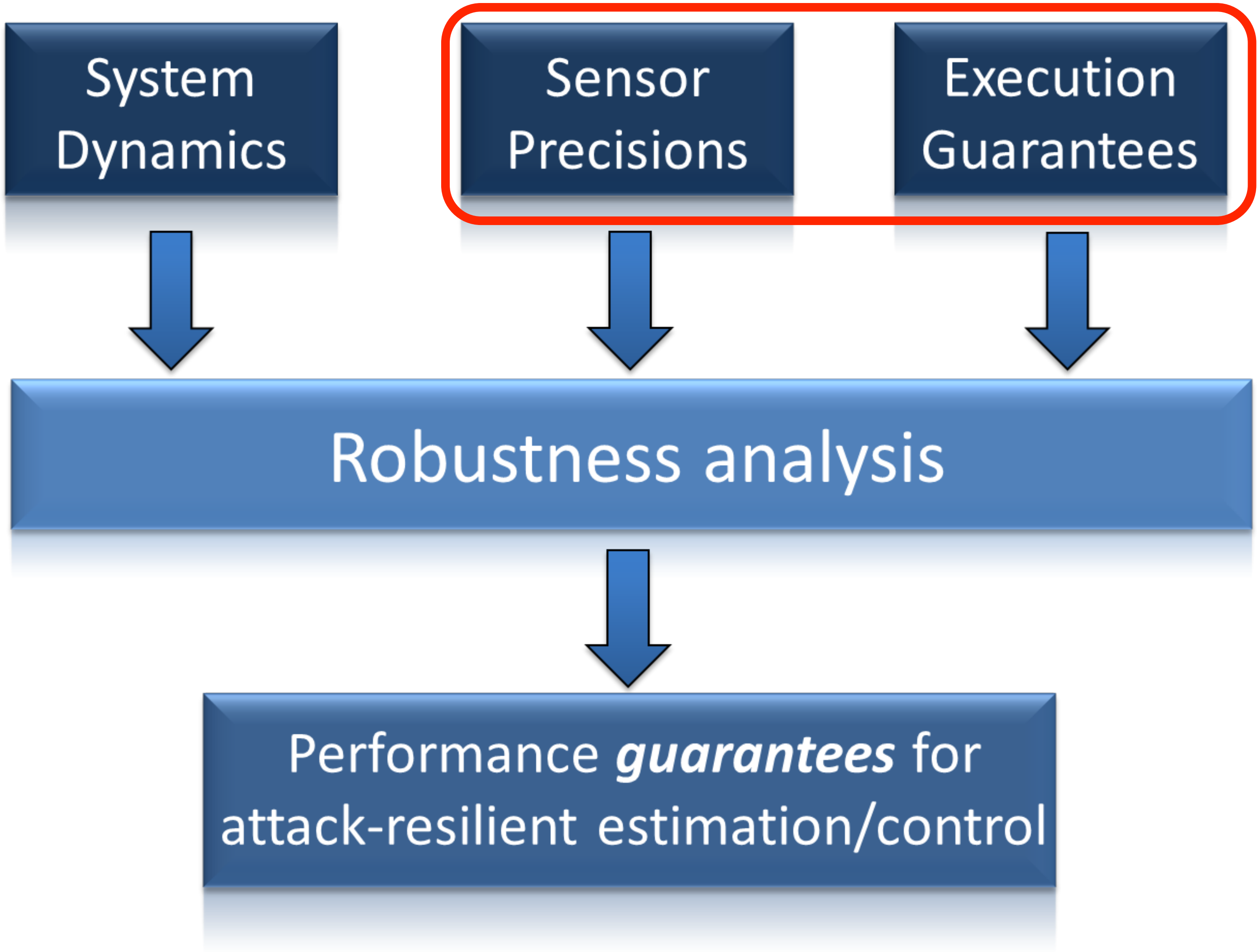
$\mathbf{y}_k$

State-based Feedback Controller

$\hat{\mathbf{x}}_k$

Resilient State Estimator

Goal: Attack-resilient state-estimation with performance guarantees



System Dynamics

Sensor Precisions

Execution Guarantees

Robustness analysis

Performance ***guarantees*** for attack-resilient estimation/control

Modeling Errors Caused by Timing

- Control of Linear-Time Invariant continuous plants

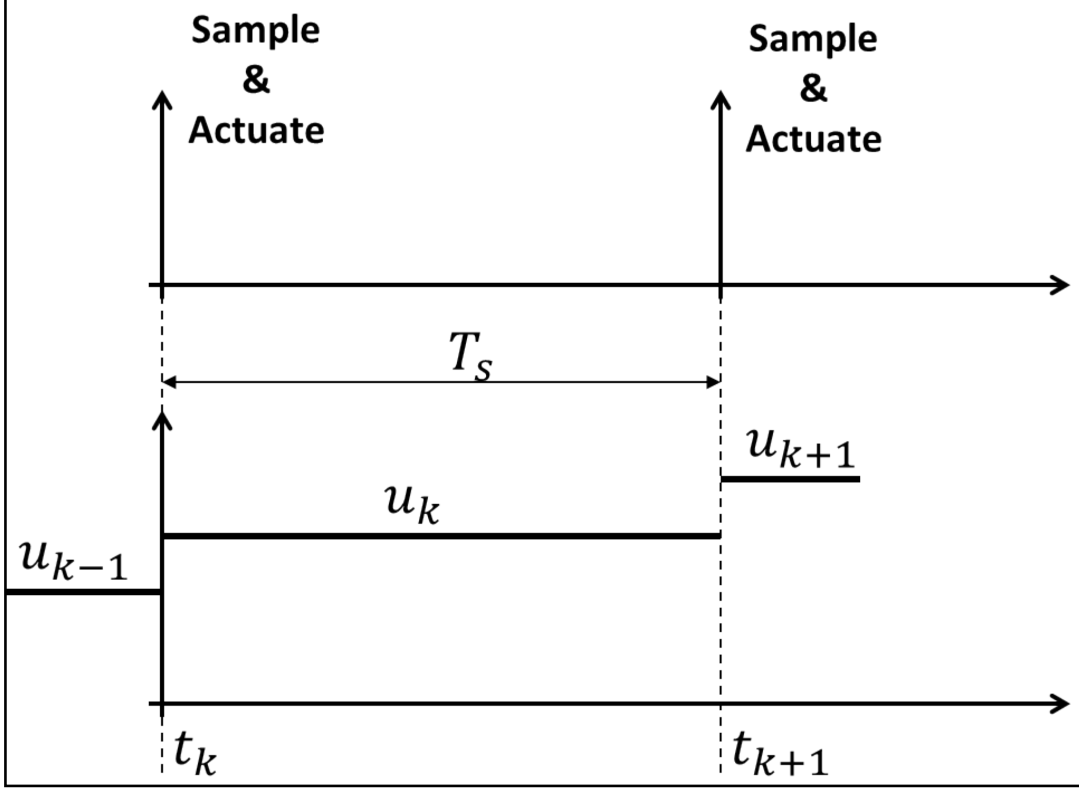
$$\dot{\mathbf{x}} = \mathbf{A}_c\mathbf{x} + \mathbf{B}_c\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x}$$

*Ideal d*iscrete-time plant model

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$
$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k$$

$$\mathbf{A} = e^{\mathbf{A}_c T_s}$$

$$\mathbf{B} = \int_0^{T_s} e^{\mathbf{A}_c\theta}\mathbf{B}_c \mathrm{d}\theta$$



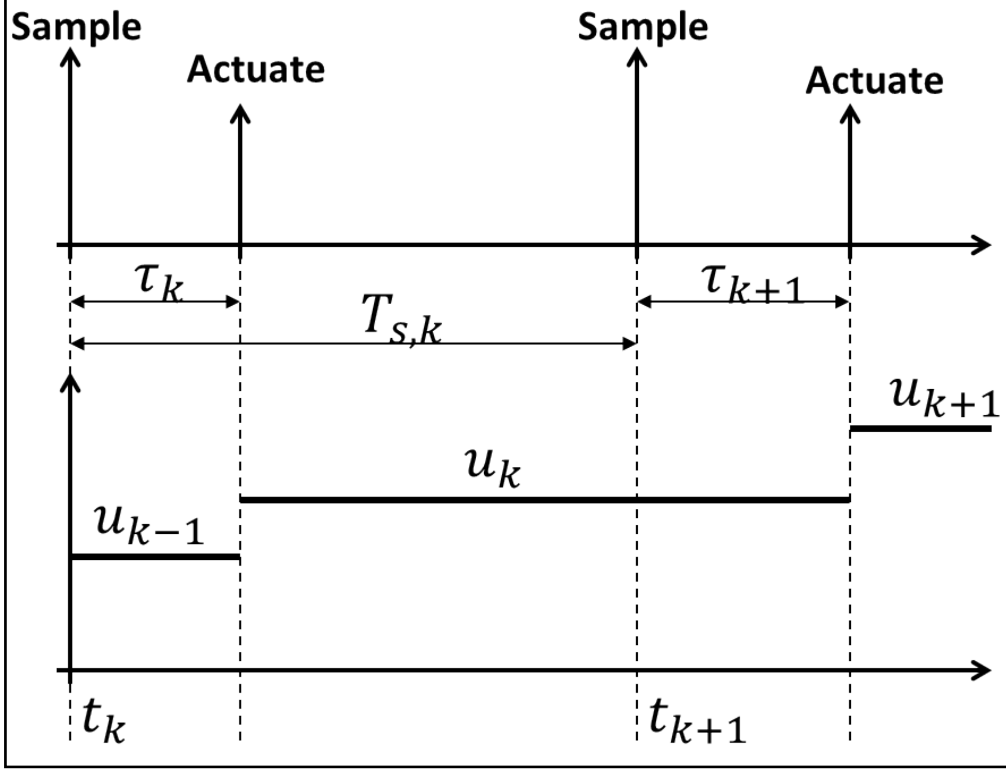
Modeling Errors Caused by Timing

- Control of Linear-Time Invariant continuous plants

$$\dot{\mathbf{x}} = \mathbf{A}_c\mathbf{x} + \mathbf{B}_c\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x}$$

Real **discrete-time plant model**

$$\mathbf{x}_k = \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k + \mathbf{B}_k^-\mathbf{u}_{k-1}$$
$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k$$



$$\mathbf{A}_k = e^{\mathbf{A}_c \boxed{T_{s,k}}}$$

$$\mathbf{B}_k = \int_0^{\boxed{T_{s,k}-\tau_k}} e^{\mathbf{A}_c\theta}\mathbf{B}_c \mathrm{d}\theta$$

$$\mathbf{B}_k^- = \int_{\boxed{T_{s,k}-\tau_k}}^{\boxed{T_{s,k}}} e^{\mathbf{A}_c\theta}\mathbf{B}_c \mathrm{d}\theta$$

Modeling Errors Caused by Timing

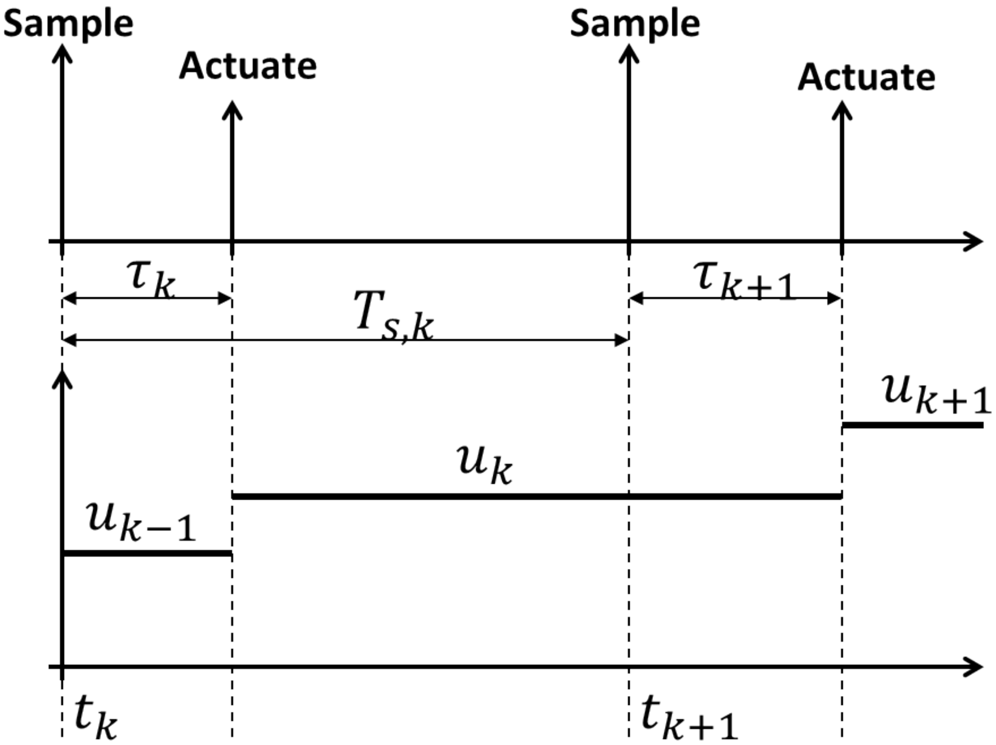
- Control of Linear-Time Invariant continuous plants

$$\dot{\mathbf{x}} = \mathbf{A}_c\mathbf{x} + \mathbf{B}_c\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x}$$

Real **discrete-time plant model**

$$\mathbf{x}_k = \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k + \mathbf{B}_k^{-}\mathbf{u}_{k-1}$$
$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k$$

$$\mathbf{v}_k^{jit} = \left(e^{\mathbf{A}_c T_{s,k}} - e^{\mathbf{A}_c T_s}\right)\mathbf{x}_k + \int_{T_s}^{T_{s,k}-\tau_k} e^{\mathbf{A}_c\theta}\mathbf{B}_c \mathrm{d}\theta\, \mathbf{u}_k + \int_{T_{s,k}-\tau_k}^{T_{s,k}} e^{\mathbf{A}_c\theta}\mathbf{B}_c \mathrm{d}\theta\, \mathbf{u}_{k-1}$$



Effects of synchronization errors between sensors can also be captured

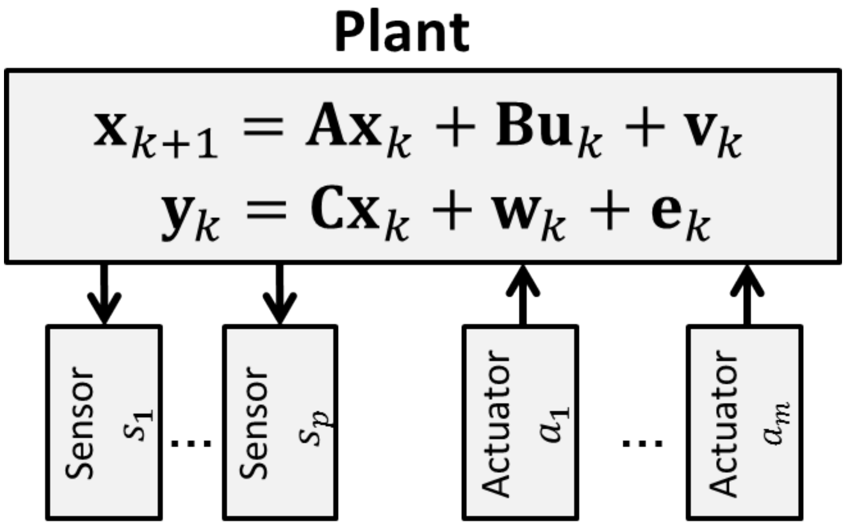
Modeling attacks on sensors

- $\mathbf{v}_k \in \mathbb{R}^n$ and $\mathbf{w}_k \in \mathbb{R}^p$ capture process and measurements noise, and modeling errors at time k

- Assumption

$$|\mathbf{v}_k| \preccurlyeq \epsilon_{\mathbf{v}_k}$$

$$|\mathbf{w}_k| \preccurlyeq \epsilon_{\mathbf{w}_k}$$



Plant

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{v}_k$$
$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{w}_k + \mathbf{e}_k$$

Problems:

- Attack-resilient state estimation with modeling errors?
- Can the attacker exploit the noise to destabilize the system?
- Can we bound the error of the state estimation?

Attack-Resilient State Estimation

Goal: Design a decoder

$$\mathbf{x}_{t-N+1} = \mathrm{D}_N(\boxed{\mathbf{y}_{t-N+1}, \dots, \mathbf{y}_t}, \boxed{\mathbf{u}_{t-N+1}, \dots, \mathbf{u}_{\mathbf{t-1}}})$$

Approach: Consider the difference between the measurement and system evolution due to the initial state/inputs

Measurement and inputs history matrix

System dynamics

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k, \qquad k = t - N + 1$$

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \sum_{i=0}^{k-t+T-2} \mathbf{C}\mathbf{A}^i\mathbf{B}\mathbf{u}_{k-1-i} \qquad k = t - N + 2, \dots, t$$

$$\mathbf{\Phi}(\mathbf{x}) \colon \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}^{p \times N}$$

$$\boxed{\mathbf{Y} = [\tilde{\mathbf{y}}_{t-N+1} | \ \dots \ | \tilde{\mathbf{y}}_t] \qquad \mathbf{\Phi}(\mathbf{x}) = [\mathbf{Cx} \,|\, \mathbf{CAx} \,| \ \dots \ | \, \mathbf{CA}^{N-1}\mathbf{x}]}$$

$$\boxed{\mathbf{E} = \mathbf{Y} - \mathbf{\Phi}(\mathbf{x})}$$

Attack-resilient State Estimation: *Ideal* case

$$\mathbf{E} = \mathbf{Y} - \mathbf{\Phi}(\mathbf{x})$$

$$= [\mathbf{e}_0 \mid \ \dots \mid \mathbf{e}_{N-1}]$$

Optimal attack-resilient state estimator [Fawzi *et al.* 2012]

$$P_0(\mathbf{Y}): \quad \min_{\mathbf{x} \in \mathbb{R}^n} \boxed{||\mathbf{E}||_{l_0}}$$

$$\mathbf{Y} - \mathbf{\Phi}(\mathbf{x}) = \mathbf{E}$$

$$(\mathbf{x}_0, \mathbf{E}) = \operatorname{argmin} P_0(\mathbf{Y})$$

$$q = ||\mathbf{E}||_{l_0} \leq q_{max}$$

$$\mathcal{K} = \{s_2, s_5\}, \mathcal{S} = \{s_1, \dots, s_5\}$$

$$\mathbf{E} = \begin{bmatrix} 0 & 0 & 0 \\ 1.7 & -2 & 6 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -9 & 3.1 & 5 \end{bmatrix}$$

$$||\mathbf{E}||_{l_0} = 2$$

The number of nonzero rows of the matrix

The maximal number of attacked sensors for which the state can be estimated

Attack-resilient State Estimation: With noise and modeling errors

- Consider initial state $\mathbf{x}_0$ and attack matrix $\mathbf{E} = [\mathbf{e}_0 \mid \ldots \mid \mathbf{e}_{N-1}]$

$$P_0(\mathbf{Y}): \quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{l_0}$$

$$\mathbf{Y} - \mathbf{\Phi}(\mathbf{x}) = \mathbf{E}$$

$$(\mathbf{x}_0, \mathbf{E}) = \text{argmin}\, P_0(\bar{\mathbf{Y}})$$

$$\mathrm{P}_{0,\Delta}(\mathbf{Y}): \quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{l_0}$$

$$-\mathbf{\Delta} \preccurlyeq \mathbf{Y} - \mathbf{\Phi}(\mathbf{x}) - \mathbf{E} \preccurlyeq \mathbf{\Delta}$$

$$\left(\mathbf{x}_{0,\Delta}, \mathbf{E}_{\mathbf{\Delta}}\right) = \text{argmin}\, \mathrm{P}_{0,\Delta}(\mathbf{Y})$$

How to initialize $\mathbf{\Delta}$?

Can we bound the error?

$$||\mathbf{x}_{0,\Delta} - \mathbf{x}_0||_2$$

Attack-resilient State Estimation: With noise and modeling errors

- Consider initial state $\mathbf{x}_0$ and attack matrix $\mathbf{E} = [\mathbf{e}_0 \mid \ldots \mid \mathbf{e}_{N-1}]$

$$P_0(\mathbf{Y}): \quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{l_0}$$

$$\mathbf{Y} - \boldsymbol{\Phi}(\mathbf{x}) = \mathbf{E}$$

$$(\mathbf{x}_0, \mathbf{E}) = \text{argmin } P_0(\bar{\mathbf{Y}})$$

$$\mathrm{P}_{0,\Delta}(\mathbf{Y}): \quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{l_0}$$

$$-\boldsymbol{\Delta} \preccurlyeq \mathbf{Y} - \boldsymbol{\Phi}(\mathbf{x}) - \mathbf{E} \preccurlyeq \boldsymbol{\Delta}$$

$$\left(\mathbf{x}_{0,\Delta}, \mathbf{E}_{\boldsymbol{\Delta}}\right) = \text{argmin } \mathrm{P}_{0,\Delta}(\mathbf{Y})$$

How to initialize $\boldsymbol{\Delta}$?

Can we bound the error?

$$||\mathbf{x}_{0,\Delta} - \mathbf{x}_0||_2$$

$$-\boldsymbol{\epsilon}_{v_t} \preccurlyeq \boldsymbol{v}(t) \preccurlyeq \boldsymbol{\epsilon}_{v_t}$$

$$-\boldsymbol{\epsilon}_{w_t} \preccurlyeq w(t) \preccurlyeq \boldsymbol{\epsilon}_{w_t}$$

Initialize $\boldsymbol{\Delta} = [\boldsymbol{\delta}_1 | \boldsymbol{\delta}_2 | \ldots | \boldsymbol{\delta}_T]$

$$\boldsymbol{\delta}_k \succcurlyeq |\mathbf{C}| \sum_{i=0}^{k-1} \left|(\mathbf{A}^{k-1-i})\right| \boldsymbol{\epsilon}_{v_i} + \boldsymbol{\epsilon}_{w_i}$$

Robustness of $\mathrm{P}_{0,\Delta}(\mathrm{Y})$ State Estimation

Penn
UNIVERSITY *of* PENNSYLVANIA

$$\mathrm{P}_0(\mathbf{Y}): \quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{\boldsymbol{l}_0} \qquad\qquad \mathrm{P}_{0,\Delta}(\mathbf{Y}): \quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{\boldsymbol{l}_0}$$

$$\mathbf{Y} - \boldsymbol{\Phi}(\mathbf{x}) = \mathbf{E} \qquad\qquad -\boldsymbol{\Delta} \preccurlyeq \mathbf{Y} - \boldsymbol{\Phi}(\mathbf{x}) - \mathbf{E} \preccurlyeq \boldsymbol{\Delta}$$

$$(\mathbf{x}_0, \mathbf{E}) = \mathrm{argmin}\ \mathrm{P}_0(\bar{\mathbf{Y}}) \qquad\qquad \left(\mathbf{x}_{0,\Delta}, \mathbf{E}_{\boldsymbol{\Delta}}\right) = \mathrm{argmin}\ \mathrm{P}_{0,\Delta}(\mathbf{Y})$$

Theorem: Consider the state estimation error defined as

$$\boldsymbol{\Delta}\mathbf{x} = \ \mathbf{x}_{0,\Delta} - \mathbf{x}_0.$$

Then $||\boldsymbol{\Delta}\mathbf{x}||_2$ is **bounded**.

Robustness of $\mathrm{P}_{0,\Delta}(\mathrm{Y})$ State Estimation

$$\mathrm{P}_0(\mathbf{Y}):\quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{\boldsymbol{l}_0}$$
$$\mathbf{Y}-\mathbf{\Phi}(\mathbf{x})=\mathbf{E}$$

$$\mathrm{P}_{0,\Delta}(\mathbf{Y}):\quad \min_{\mathbf{x}\in\mathbb{R}^n} ||\mathbf{E}||_{\boldsymbol{l}_0}$$
$$-\mathbf{\Delta}\preccurlyeq\mathbf{Y}-\mathbf{\Phi}(\mathbf{x})-\mathbf{E}\preccurlyeq\mathbf{\Delta}$$

$$(\mathbf{x}_0,\mathbf{E})=\text{argmin}\ \mathrm{P}_0(\bar{\mathbf{Y}})\qquad \left(\mathbf{x}_{0,\Delta},\mathbf{E}_{\mathbf{\Delta}}\right)=\text{argmin}\ \mathrm{P}_{0,\Delta}(\mathbf{Y})$$

Corollary: If a stable state-feedback controller utilizes the state estimate $\mathbf{x}_{0,\Delta}$ (i.e., $\mathbf{u}_k=\mathbf{K}\mathbf{x}_{0,\Delta}$, where $\mathbf{A}+\mathbf{B}\mathbf{K}$ is stable)

- Then the closed-loop system will remain stable when at most q_{max} sensors have been compromised.

Bounding the State Estimation Error Algorithm Complexity

- Finding extreme points for every *F*

$$F = 0,1,\dots,p-2q_{max}-1$$

$$|\mathcal{K}| = F, |\mathcal{K}_1| = p-2q_{max}-F$$

$$\mathcal{K}\cup\mathcal{K}_1 \subset S, \mathcal{K}\cap\mathcal{K}_1 = \emptyset$$

$$\max_{\Delta\mathbf{x}\in\mathbb{R}^n} ||\Delta\mathbf{x}||_2$$

$$\mathbf{O}_{\mathcal{K}_\mathbf{F}}\Delta\mathbf{x} = \mathbf{0}$$

$$\mathbf{O}_{\mathcal{K}_{\mathbf{F}_1}}\Delta\mathbf{x} \preccurlyeq 2\boldsymbol{\Delta}_{\mathcal{K}_{\mathbf{F}_1}}$$

$$rank\left(\mathbf{O}_{\mathcal{K}\cup\mathcal{K}_1}\right) = n$$

- *…but we need to do it only at design-time*

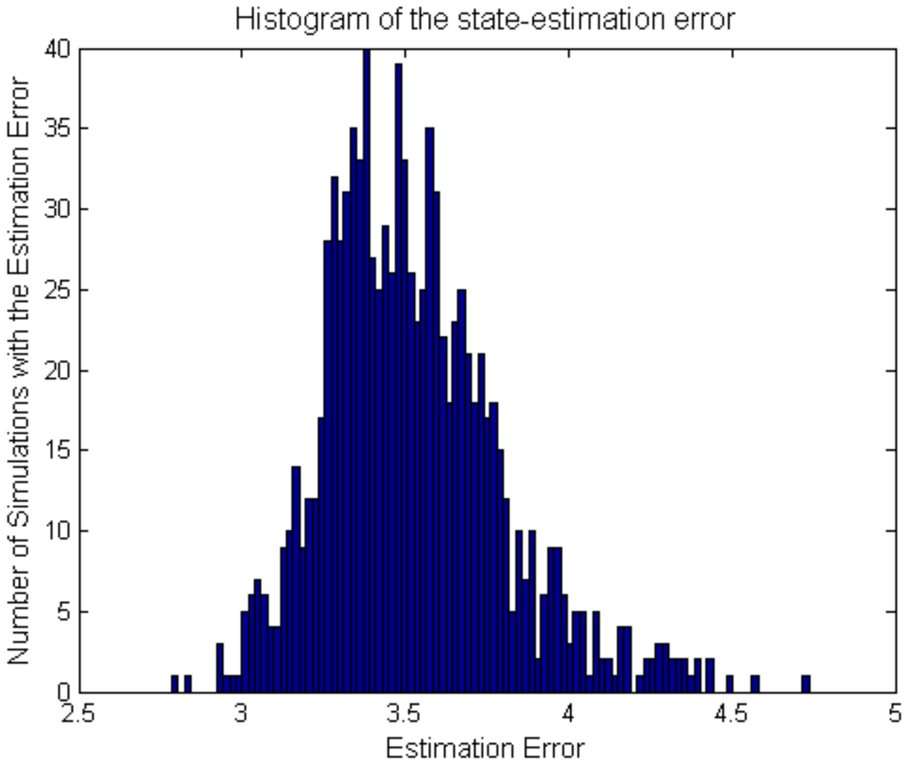
$$\mathbf{O}_{\mathcal{K}} = \begin{bmatrix} P_{\mathcal{K}}\mathbf{C} \\ P_{\mathcal{K}}\mathbf{CA} \\ \vdots \\ P_{\mathcal{K}}\mathbf{CA}^{N-1} \end{bmatrix}$$

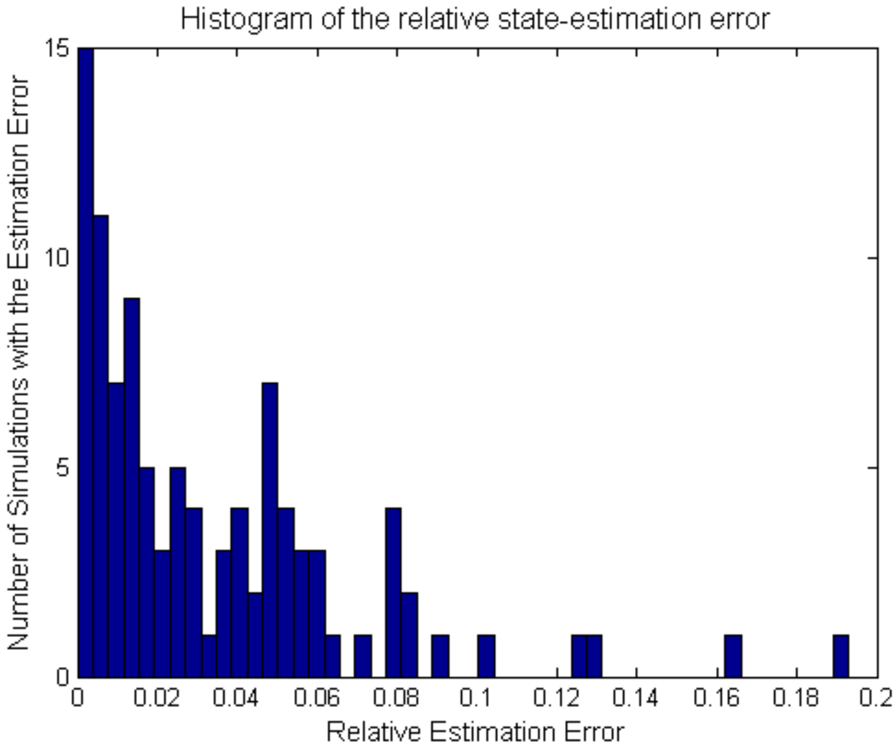
- For almost all systems

$$q_{max} = ceil\left(\frac{p}{2}-1\right) \Rightarrow\ p-2q_{max}-1 \le 1$$

Evaluation of the 'bounding' algorithm

Simulation results for 1000 runs of 100 randomly selected systems with n = 10 states and p = 5 sensors.

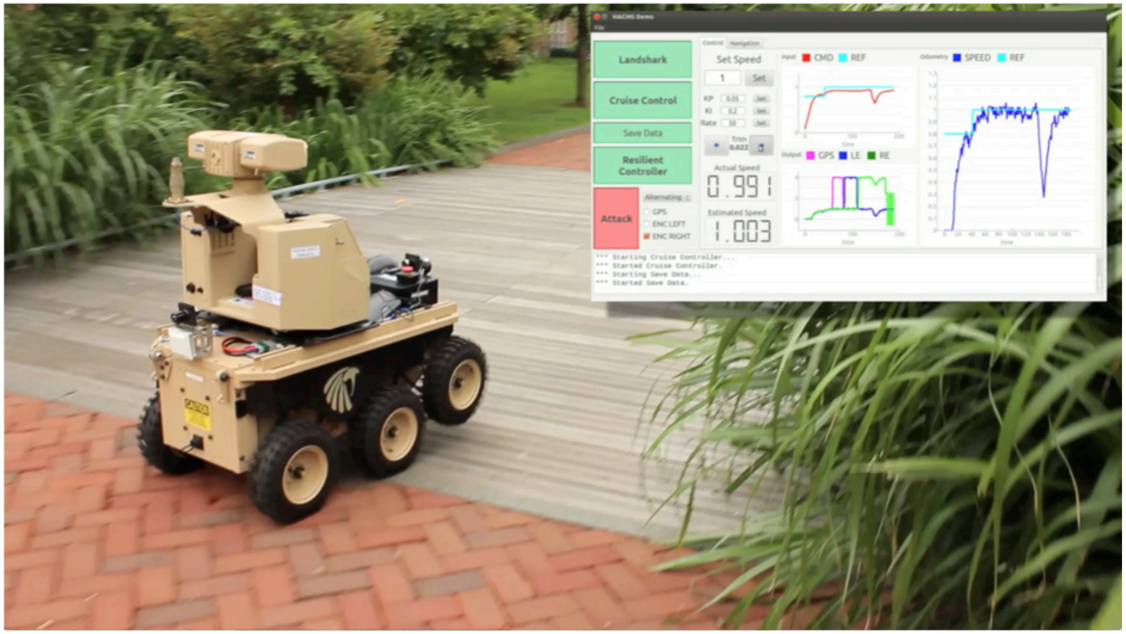




$$Rel\_error_{\mathfrak{S}} = \frac{\max_{i=1:1000} \Delta \mathrm{x}_{\mathfrak{S}}}{MAX\_\|\Delta \mathrm{x}_{\mathfrak{S}}\|_2}$$

Case Study

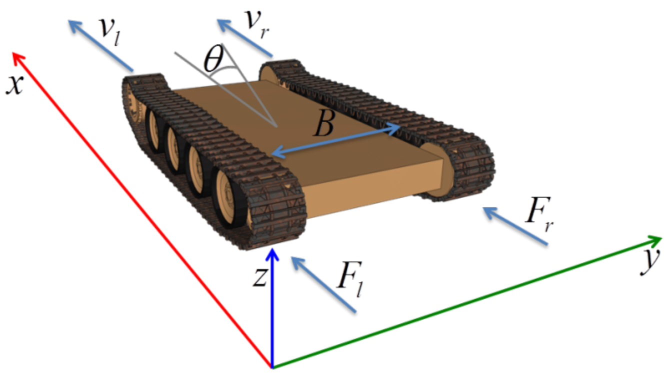
- Constant-speed cruise control for LandShark
  - Ensure that the vehicle can maintain speed when some of the sensors are under attacked



Set Speed: 1
KP: 0.05
KI: 0.2
Rate: 50
Trim: 0.022
Actual Speed: 0.991
Estimated Speed: 1.003
Alternating
☐ GPS
☐ ENC LEFT
☑ ENC RIGHT

```
*** Starting Cruise Controller...
*** Started Cruise Controller.
*** Starting Save Data...
*** Started Save Data.
```

7th order skid steering model

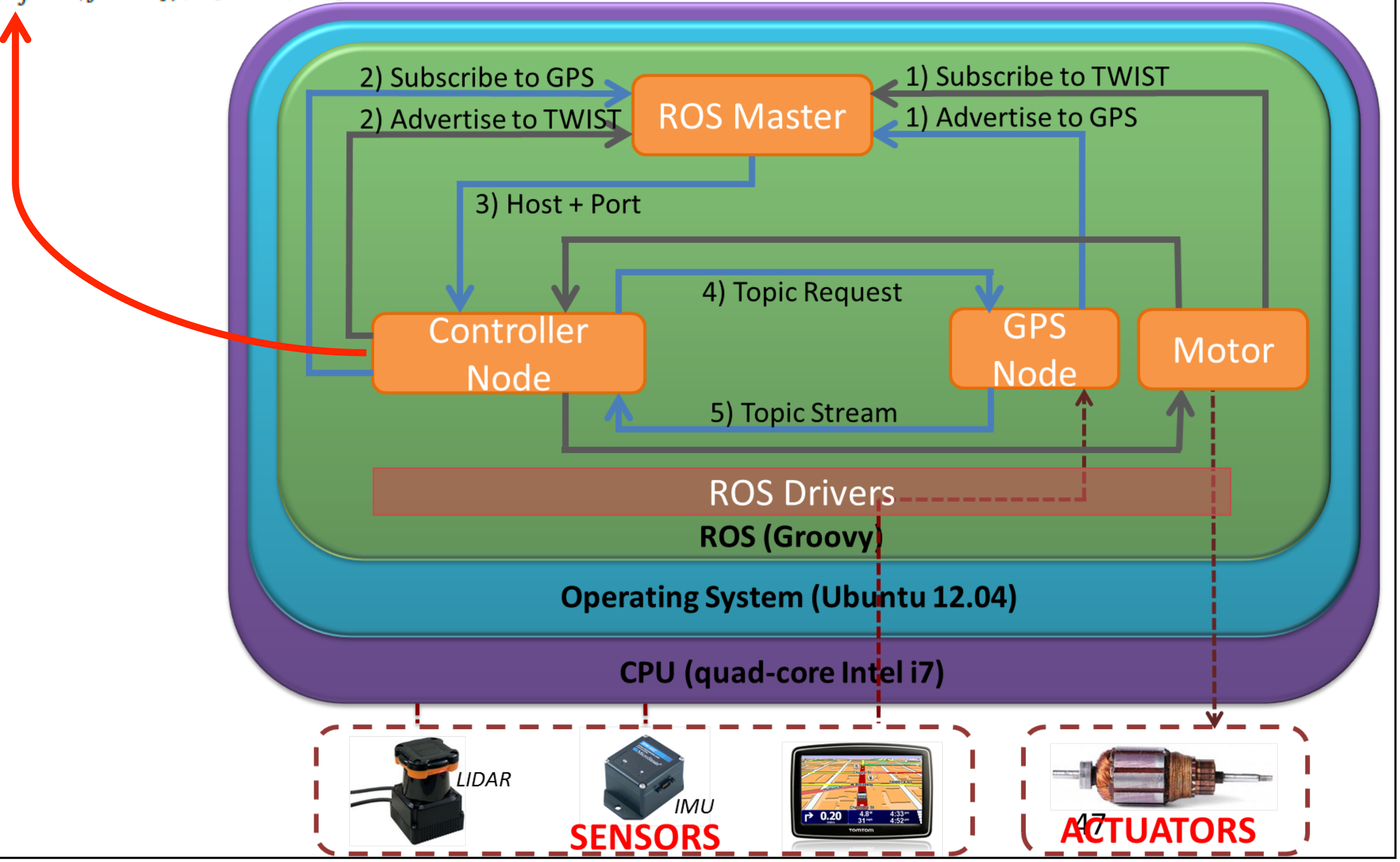


System Architecture in ROS

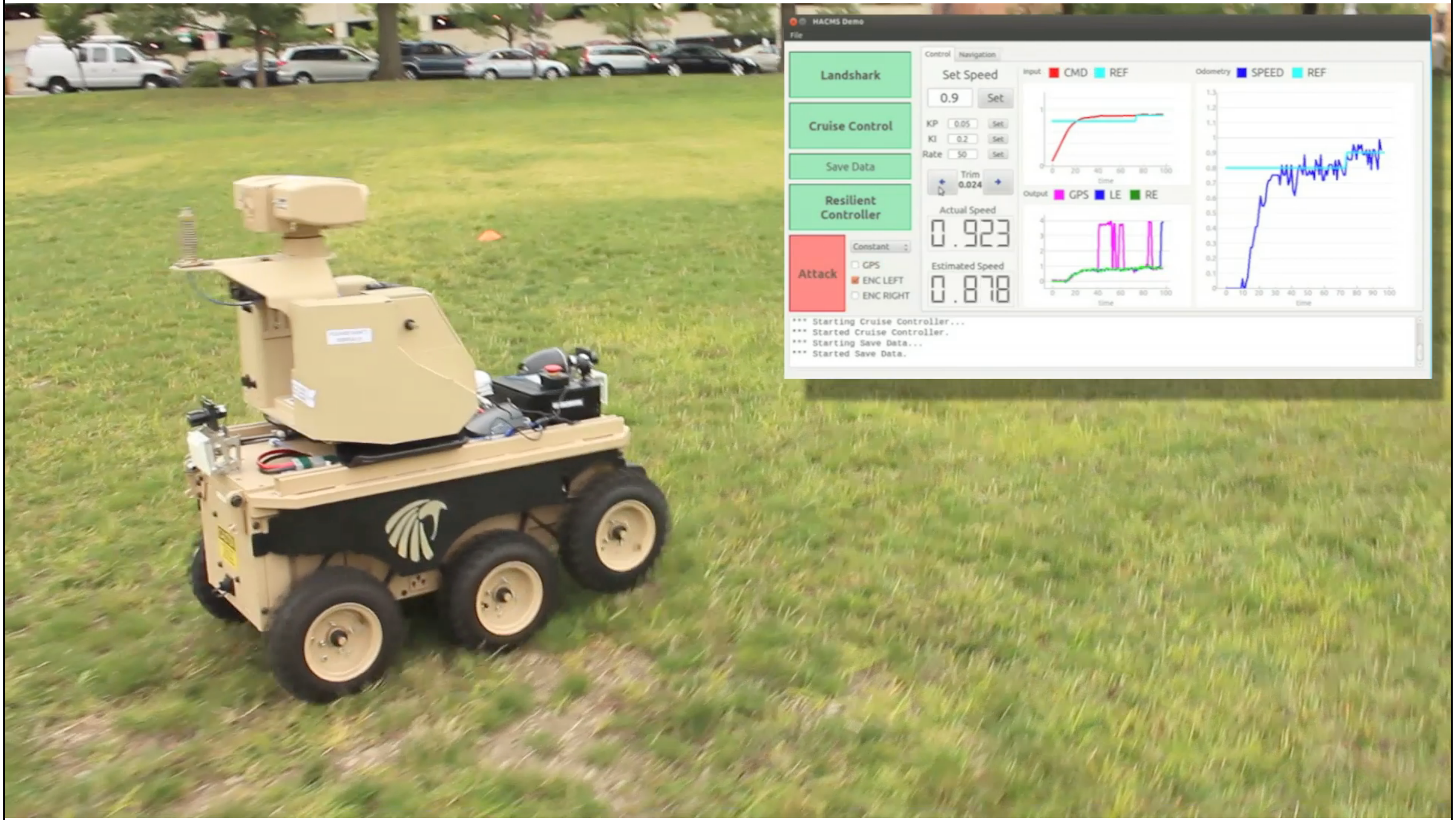
$$\min_{\gamma,\mathbf{E},\mathbf{x}} \quad \mathbf{1}_p^\top \gamma$$

$$-\delta_k \preceq \mathbf{y}_k - \mathbf{C}\mathbf{A}^k\mathbf{x} - \mathbf{e}_k \preceq \delta_k, \quad k=0,...,N-1$$

$$-\gamma_j\alpha\cdot\mathbf{1}'_N \preceq \mathbf{E}'_j \preceq \gamma_j\alpha\cdot\mathbf{1}'_N, \quad j=1,...,p$$



Attack-Resilient Cruise Control Demo



Set Speed: 0.9

KP: 0.05

KI: 0.2

Rate: 50

Trim: 0.024

Actual Speed: 0.923

Estimated Speed: 0.878

Constant

□ GPS

☑ ENC LEFT

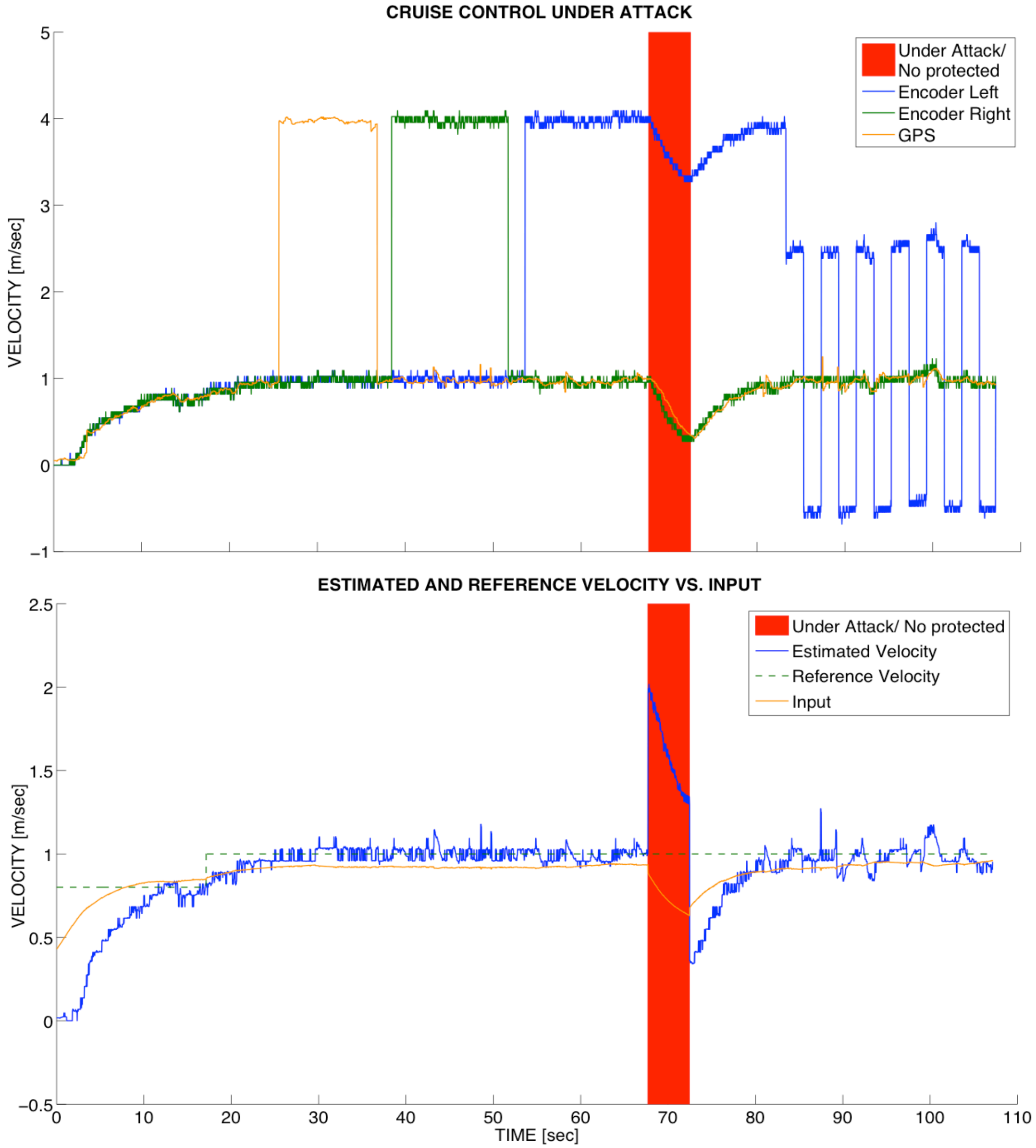
□ ENC RIGHT

```
*** Starting Cruise Controller...
*** Started Cruise Controller.
*** Starting Save Data...
*** Started Save Data.
```

www.seas.upenn.edu/~pajic/research/CPS_security.html

Attack-Resilient Cruise Control Demo

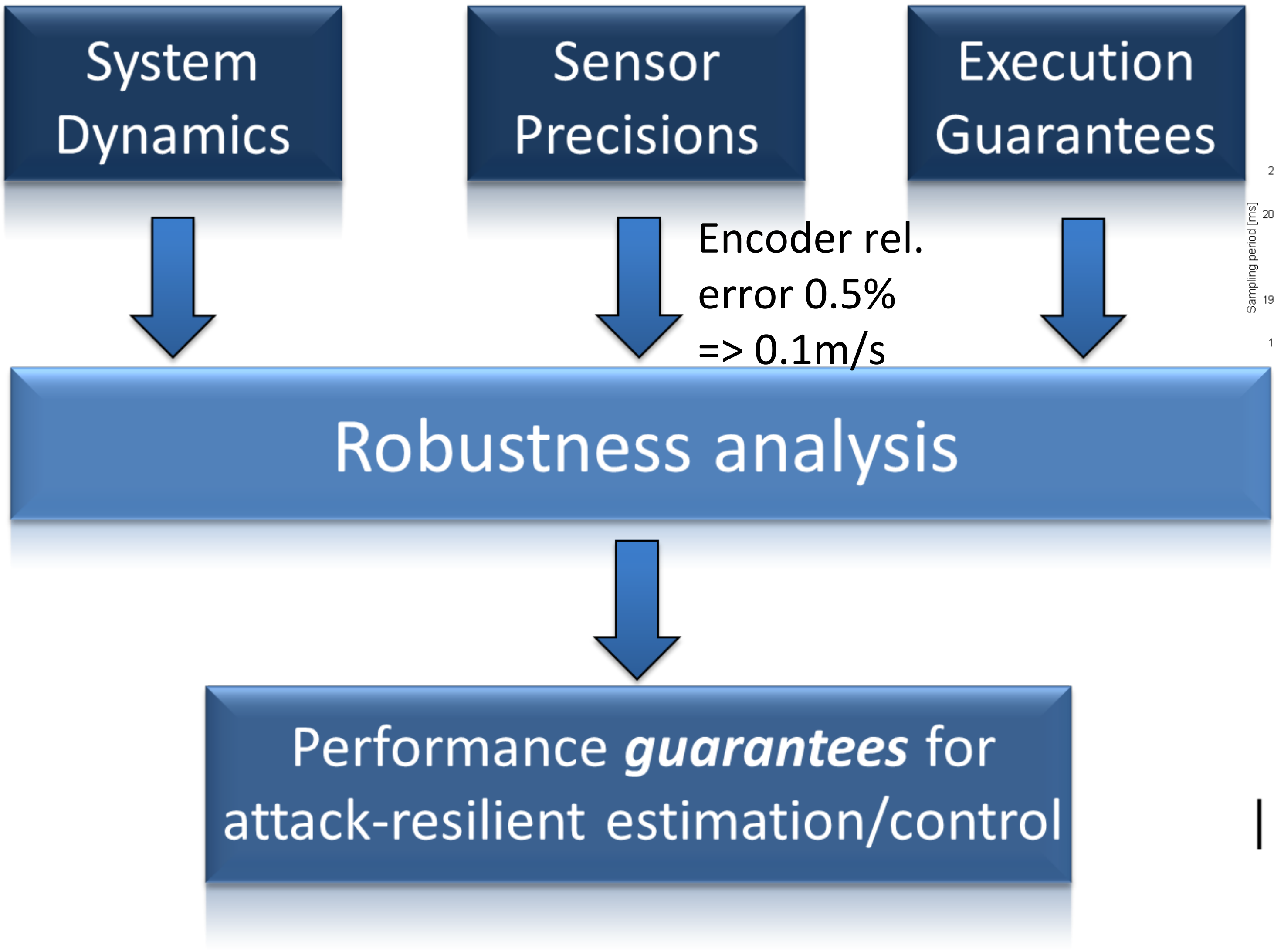
Penn
UNIVERSITY of PENNSYLVANIA



CRUISE CONTROL UNDER ATTACK

ESTIMATED AND REFERENCE VELOCITY VS. INPUT

Robustness Analysis



System Dynamics

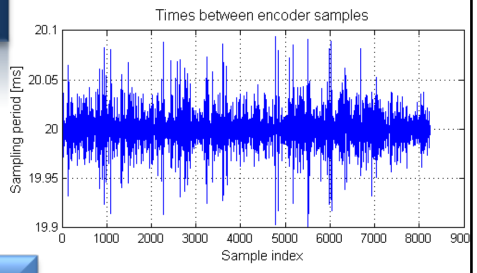
Sensor Precisions

Execution Guarantees

Encoder rel. error 0.5% => 0.1m/s

Robustness analysis

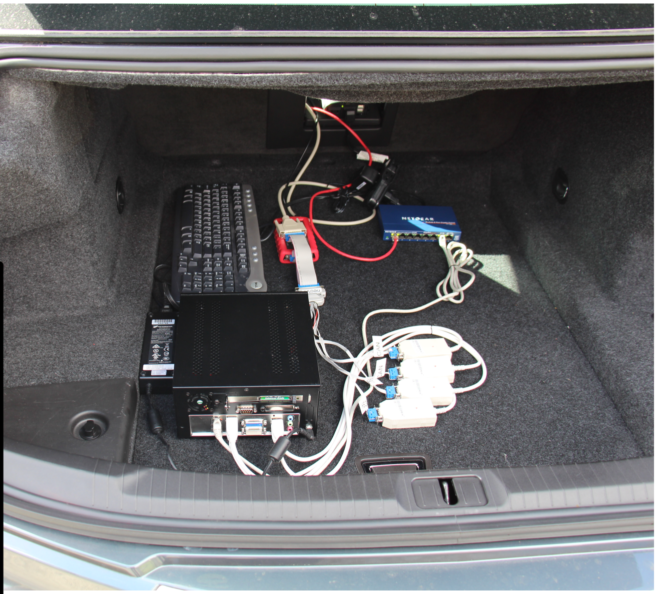
Performance ***guarantees*** for attack-resilient estimation/control

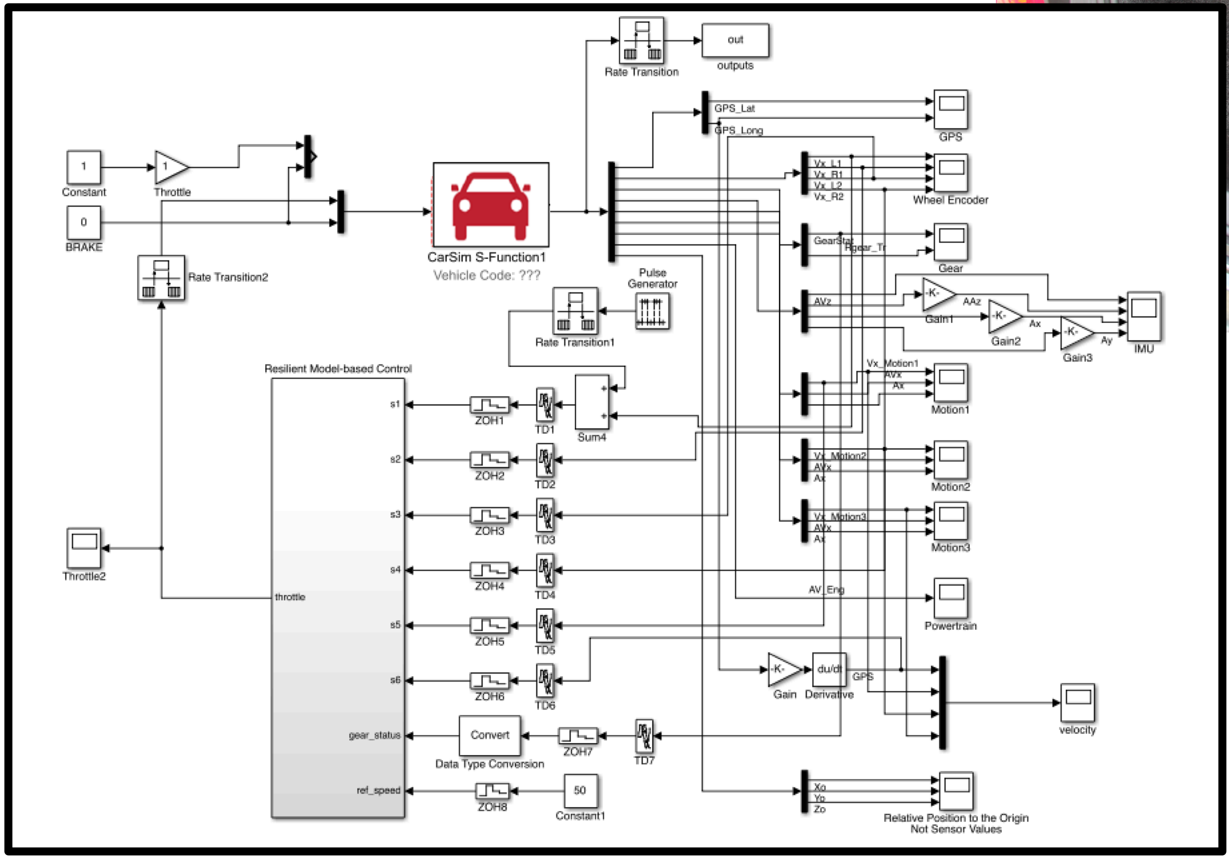


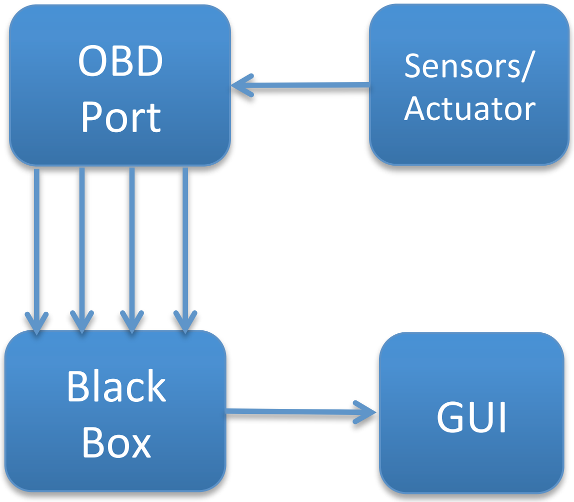
$$||\mathbf{\Delta x}||_2 = 0.72$$

Attack-resilient state estimator for American Built Car

- CarSim Simulation
- In-Car Implementation







Attack-resilient state estimator for American Built Car

www.seas.upenn.edu/~pajic/research/CPS_security.html

PRECISE
PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING
DARPA

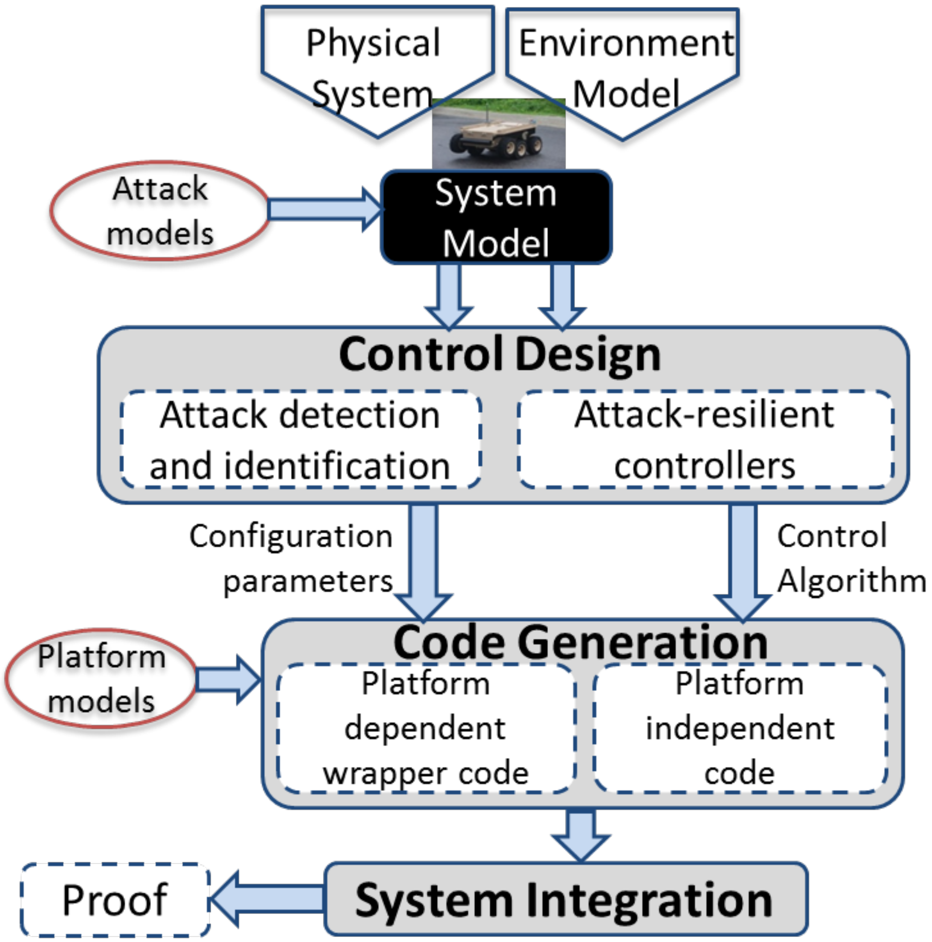
ATTACK RESILIENT STATE ESTIMATOR

Miroslav Pajic, Nicola Bezzo, James Weimer, Oleg Sokolsky,
Paulo Tabuada, George J. Pappas, Insup Lee

Synthesis of Secure and Attack-Resilient Cyber-Physical Systems

Goal: Develop tools and techniques to ensure that cyber-physical systems maintain a degree of control even when the system is under cyber and/or **physical** attack

- Overall approach
  - Control-level techniques
  - Code-level techniques



Physical System

Environment Model

Attack models

System Model

Control Design

Attack detection and identification

Attack-resilient controllers

Configuration parameters

Control Algorithm

Code Generation

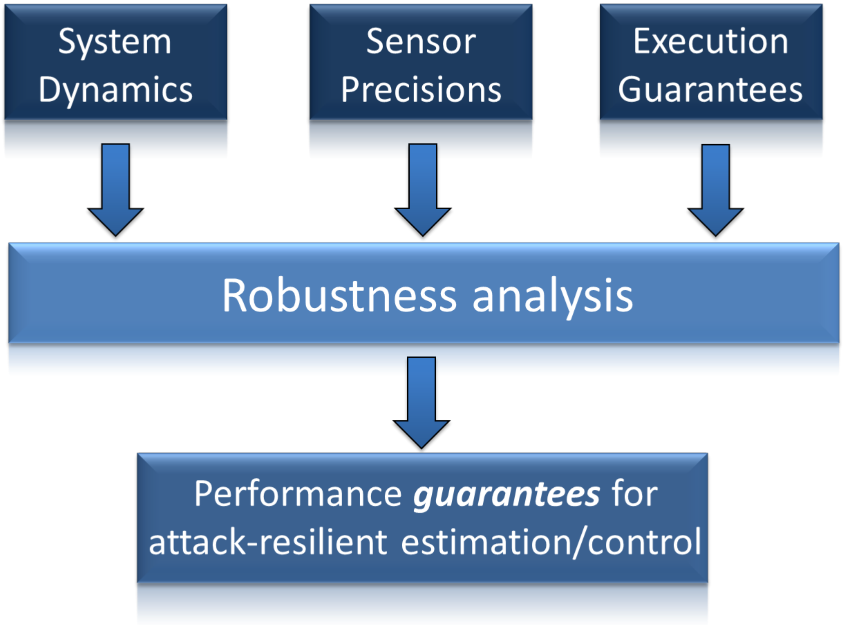
Platform models

Platform dependent wrapper code

Platform independent code

Proof

System Integration



System Dynamics

Sensor Precisions

Execution Guarantees

Robustness analysis

Performance ***guarantees*** for attack-resilient estimation/control

Thank You

PRECISE

PENN RESEARCH IN EMBEDDED COMPUTING AND INTEGRATED SYSTEMS ENGINEERING

Penn
UNIVERSITY of PENNSYLVANIA

DARPA