Toward AI-based autonomy: safety and security in cyber-physical systems

Mohammad Javad Khojasteh

Autonomous Systems Lab (ASL)
Stanford University

July 2020

Taking robots into the real world

Brittle hand-designed dynamics models work for lab operation but fail to account for the complexity and uncertainty of real-world operation

Learning for dynamics and control

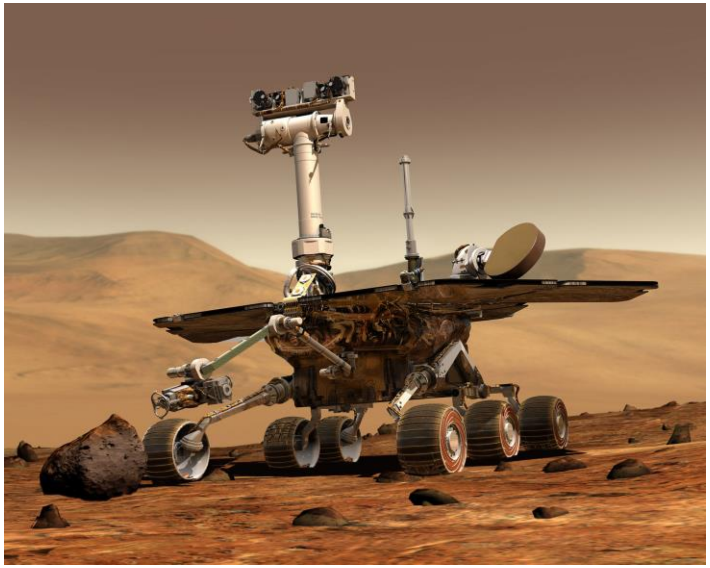
Cyber

learning online relying on
streaming data

Physical

control objectives and
guaranteeing safe operation

Example: space missions



We need to address

1. individual safety: e.g. avoiding the obstacles
2. joint safety: e.g. avoiding the collision with other agents

Example: sandtrap

Train is a major source of risks for Mars rovers:

- Spirit → embeded in sand
- Opportunity → got stuck in soft sand for 6 weeks

Outline

Part I: Safety

1. Probabilistic Safety Constraints for Learned High Relative Degree System

Joint work with:

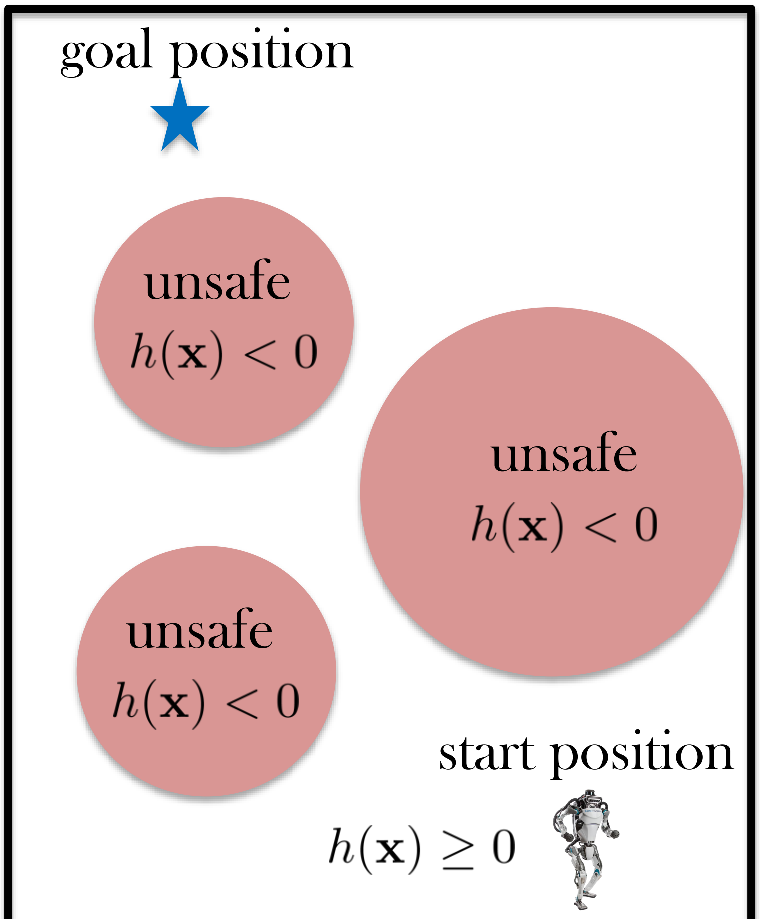
- Vikas Dhiman, UCSD
- Massimo Franceschetti, UCSD
- Nikolay Atanasov, UCSD

2. Safe Multi-Agent Interaction through CBF with Learned Uncertainties

Part II: Security

Learning-based attacks in cyber-physical systems

Problem formulation



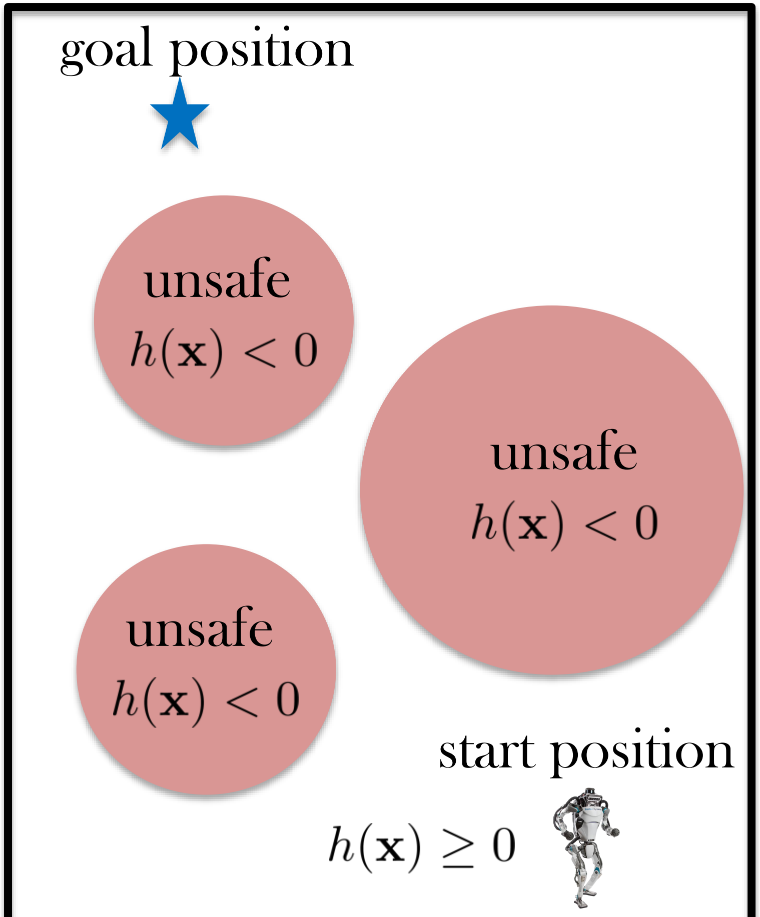
$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} \\ &= \begin{bmatrix} f(\mathbf{x}) & g(\mathbf{x}) \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{u} \end{bmatrix} \\ &= F(\mathbf{x})\underline{\mathbf{u}}\end{aligned}$$

drift term $\quad f : \mathbb{R}^n \to \mathbb{R}^n$

input gain $\quad g : \mathbb{R}^n \to \mathbb{R}^{n \times m}$

We study the problem of enforcing probabilistic safety when f and g are unknown

Problem formulation



$$\dot{\mathbf{x}} = F(\mathbf{x})\underline{\mathbf{u}}$$

$$vec(F(\mathbf{x})) \sim \mathcal{GP}\left(vec(\mathbf{M}_0(\mathbf{x})), \mathbf{K}_0(\mathbf{x}, \mathbf{x}')\right)$$

baseline control policy

$$\min_{\mathbf{u}_k \in \mathcal{U}} \|\mathbf{u}_k - \pi(\mathbf{x}_k)\|_Q$$

$$\text{s.t.} \quad \mathbb{P}(\text{safety}) \geq p_k$$

user-specified risk tolerance

Approach

1. Bayesian learning
2. Propagate uncertainty to the safety condition
3. Self-triggered control: extension to continous time
4. Extension to higher relative degree systems

Gaussian processes for machine learning

$$\dot{\mathbf{x}} = F(\mathbf{x})\underline{\mathbf{u}}$$

$$vec(F(\mathbf{x})) \sim \mathcal{GP}\left(vec(\mathbf{M}_0(\mathbf{x})), \mathbf{K}_0(\mathbf{x}, \mathbf{x}')\right)$$

The controller observes $\mathbf{X}_{1:k} := [\mathbf{x}(t_1), \ldots, \mathbf{x}(t_k)]$, $\mathbf{U}_{1:k} := [\mathbf{u}(t_1), \ldots, \mathbf{u}(t_k)]$ without noise,

but the measurements $\dot{\mathbf{X}}_{1:k} = [\dot{\mathbf{x}}(t_1), \ldots, \dot{\mathbf{x}}(t_k)]$ might be noisy.

In general, there may be a correlation among different components of f and g.

Thus, we need to develop an efficient factorization of $\mathbf{K}_0(\mathbf{x}, \mathbf{x}')$.

Matrix variate Gaussian processes (MVGP)

$$vec(F(\mathbf{x})) \sim \mathcal{GP}\left(vec(\mathbf{M}_0(\mathbf{x})), \underbrace{\mathbf{K}_0(\mathbf{x}, \mathbf{x}')}_{\mathbf{B}_0(\mathbf{x}, \mathbf{x}') \otimes \mathbf{A}}\right)$$

$\mathbf{B}_0(\mathbf{x}, \mathbf{x}') \otimes \mathbf{A} \rightarrow$ Louizos and Welling (ICML 2016)
Sun et al. (AISTATS 2017)

The above parameterization is efficient because we need to learn smaller matrices $\mathbf{B}_0(\mathbf{x}, \mathbf{x}') \in \mathbb{R}^{(m+1)\times(m+1)}$ and $\mathbf{A} \in \mathbb{R}^{n\times n}$. Also, this parameterization preserves its structure during inference.

Inference

$$vec(F(\mathbf{x}_*)) \sim \mathcal{GP}(vec(\mathbf{M}_k(\mathbf{x}_*)), \mathbf{B}_k(\mathbf{x}_*, \mathbf{x}'_*) \otimes \mathbf{A})$$

$$F(\mathbf{x}_*)\underline{\mathbf{u}}_* = f(\mathbf{x}_*) + g(\mathbf{x}_*)\mathbf{u}_* \sim \mathcal{GP}(\mathbf{M}_k(\mathbf{x}_*)\underline{\mathbf{u}}_*, \underline{\mathbf{u}}_*^\top \mathbf{B}_k(\mathbf{x}_*, \mathbf{x}'_*)\underline{\mathbf{u}}_* \otimes \mathbf{A})$$

$\mathbf{M}_k(\mathbf{x}_*)$ and $\mathbf{B}_k(\mathbf{x}_*, \mathbf{x}'_*)$ are calculated in our paper

Two alternative approaches

1. Develop a decoupled GP regression per system dimension:

   Does not model the dependencies among different components of f and g

   Inference computational complexity:

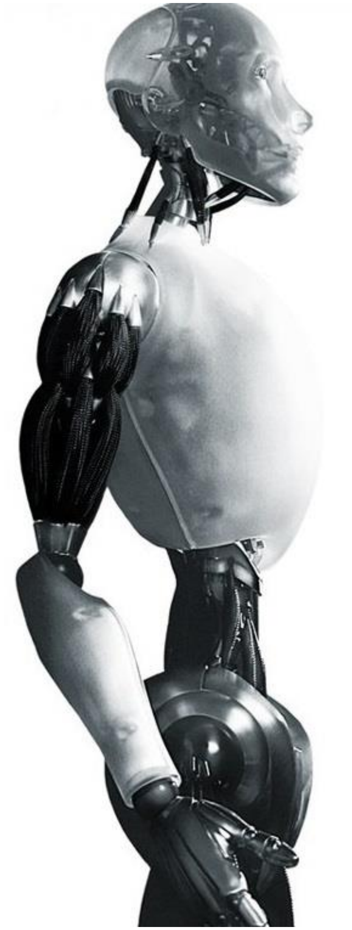
   decoupled GP $O((1+m)k^2) + O(k^3)$ MVGP $O((1+m)^3k^2) + O(k^3)$

2. Coregionalization models [Alvarez et al. (FTML 2012)]:

$$\mathbf{K}_0(\mathbf{x}, \mathbf{x}') = \mathbf{\Sigma} \underbrace{\kappa_0(\mathbf{x}, \mathbf{x}')}_{\text{scalar state-dependent kernel}}$$

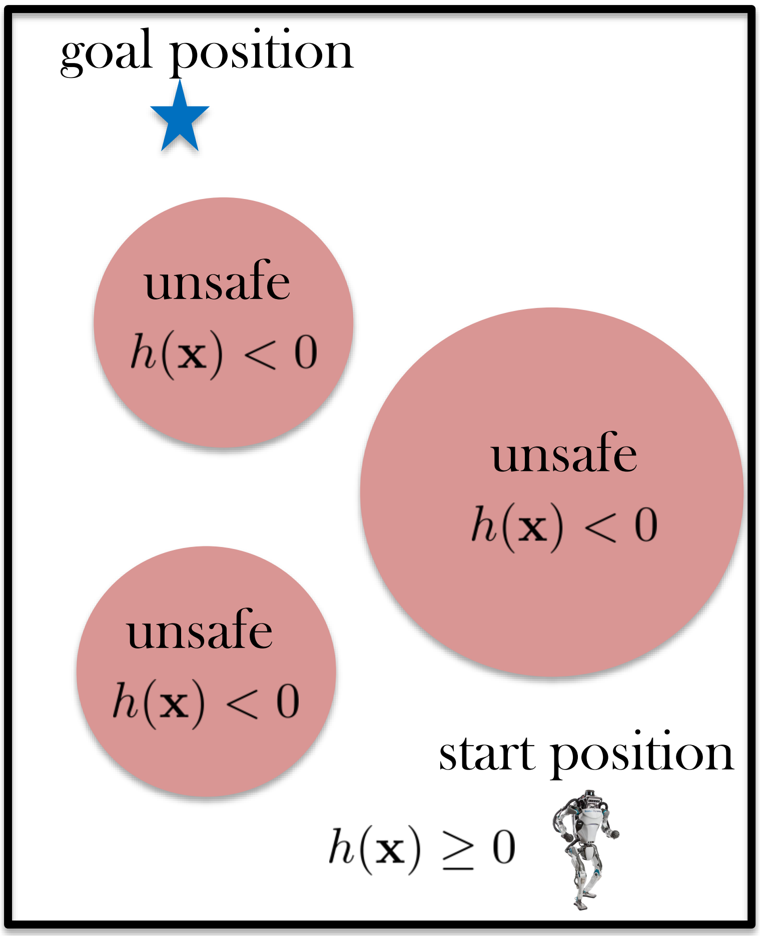
The nice matrix-times-scalar-kernel structure is not preserved in the posterior

Approach



1. Bayesian learning
2. Propagate uncertainty to the safety condition
3. Self-triggered control: extension to continous time
4. Extension to higher relative degree systems

Control Barrier Functions (CBF)



Previously, CBF are used to dynamically enforce the safety for known dynamics

Ames et al. (ECC 2019)

Control Barrier Condition (CBC)

$$\mathrm{CBC}(\mathbf{x}, \mathbf{u}) := \underbrace{\mathcal{L}_f h(\mathbf{x}) + \mathcal{L}_g h(\mathbf{x})\mathbf{u}}_{\nabla_{\mathbf{x}} h(\mathbf{x}) F(\mathbf{x}) \underline{\mathbf{u}}} + \underbrace{\alpha}_{\alpha > 0} h(\mathbf{x}) \geq 0$$

A lower bound on the derivative

Uncertainity propagation to CBC

$$\mathrm{CBC}(\mathbf{x},\mathbf{u}) = \underbrace{\mathcal{L}_f h(\mathbf{x}) + \mathcal{L}_g h(\mathbf{x})\mathbf{u}}_{\nabla_{\mathbf{x}} h(\mathbf{x}) F(\mathbf{x})\underline{\mathbf{u}}} + \underbrace{\alpha h(\mathbf{x})}_{\alpha > 0}$$

$$vec(F(\mathbf{x}_*)) \sim \mathcal{GP}(vec(\mathbf{M}_k(\mathbf{x}_*)), \mathbf{B}_k(\mathbf{x}_*, \mathbf{x}'_*) \otimes \mathbf{A})$$

We have shown given $\mathbf{x}_k$ and $\mathbf{u}_k$, $\mathrm{CBC}(\mathbf{x}_k, \mathbf{u}_k)$ is a Gaussian random variable with the following parameters

$$\mathbb{E}[\mathrm{CBC}_k] = \nabla_{\mathbf{x}} h(\mathbf{x}_k)^\top \mathbf{M}_k(\mathbf{x}_k)\underline{\mathbf{u}}_k + \alpha h(\mathbf{x}_k)$$

$$\mathrm{Var}[\mathrm{CBC}_k] = \underline{\mathbf{u}}_k^\top \mathbf{B}_k(\mathbf{x}_k, \mathbf{x}_k)\underline{\mathbf{u}}_k \nabla_{\mathbf{x}} h(\mathbf{x}_k)^\top \mathbf{A} \nabla_{\mathbf{x}} h(\mathbf{x}_k)$$

Note: mean and variance are Affine and Quadratic in $\mathbf{u}$ respectively.

Deterministic condition for controller

$$\min_{\mathbf{u}_k \in \mathcal{U}} \|\mathbf{u}_k - \pi(\mathbf{x}_k)\|_Q$$

$$\text{s.t.} \quad \mathbb{P}(\text{CBC}(\mathbf{x}_k, \mathbf{u}_k) \geq \zeta > 0 | \mathbf{x}_k, \mathbf{u}_k) \geq \tilde{p}_k$$

Kh-Dhiman-Franceschetti-Atanasov 2020

$$(\mathbb{E}[\text{CBC}(\mathbf{x}_k, \mathbf{u}_k)] - \zeta)^2 \geq 2\text{Var}[\text{CBC}(\mathbf{x}_k, \mathbf{u}_k)] \left(\text{erf}^{-1}(1 - 2\tilde{p}_k)\right)^2$$

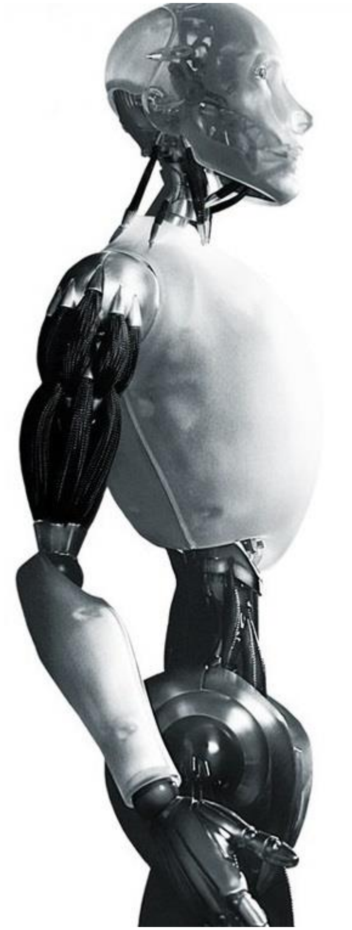
$$\mathbb{E}[\text{CBC}(\mathbf{x}_k, \mathbf{u}_k)] - \zeta \geq 0$$

A safe optimization-based controller which is a Quadratically Constrained Quadratic Program (QCQP)

This QCQP might not be convex →

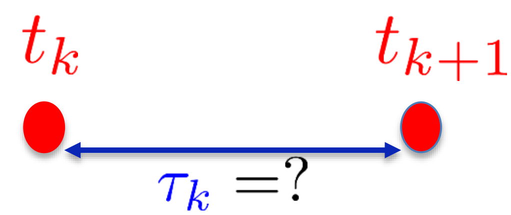
Second Order Cone Program (SOCP)

Approach



1. Bayesian learning
2. Propagate uncertainty to the safety condition
3. Self-triggered control: extension to continous time
4. Extension to higher relative degree systems

Safety beyond triggering times



Safety at triggering times

$$\min_{\mathbf{u}_k \in \mathcal{U}} \|\mathbf{u}_k - \pi(\mathbf{x}_k)\|$$

$$\text{s.t.} \quad \mathbb{P}(\mathrm{CBC}(\mathbf{x}_k, \mathbf{u}_k) \geq \zeta > 0 | \mathbf{x}_k, \mathbf{u}_k) \geq \tilde{p}_k$$

Safety during the inter-triggering times

$\mathbf{u}(t) \equiv \mathbf{u}_k$ zero-order hold (ZOH) control mechanism $\forall t \in [t_k, t_k + \tau_k)$

$\tau_k = ?$ $\mathbb{P}(\mathrm{CBC}(\mathbf{x}(t), \mathbf{u}_k) \geq 0) \geq p_k$ $\forall t \in [t_k, t_k + \tau_k)$

Self-triggered Control with Probabilistic Safety Constraints

We assume the sample paths of the GP used to model the dynamics are locally Lipschitz with sufficiently large probability q_k

QCQP

$\mathbb{P}(\text{safety}) \geq \tilde{p}_k$

t_k

QCQP

$\mathbb{P}(\text{safety}) \geq \tilde{p}_{k+1}$

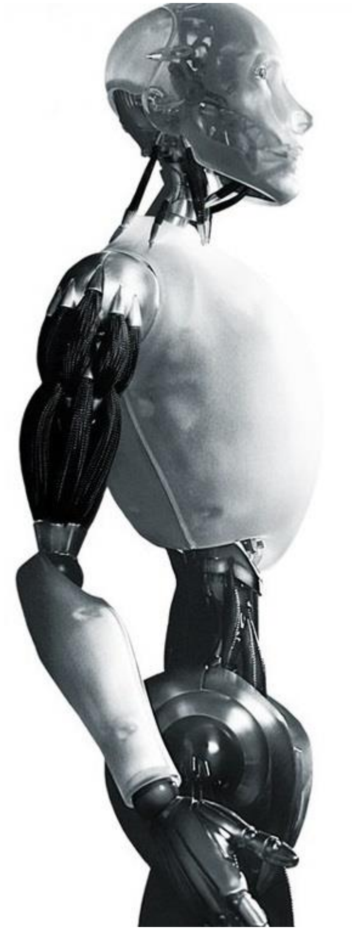
t_{k+1}

$$\tau_k \leq \frac{1}{L_k} \ln \left(1 + \frac{L_k \zeta}{(\chi_k L_k + L_{\alpha \circ h}) \|\dot{\mathbf{x}}_k\|} \right)$$

The parameters are calculated in our paper

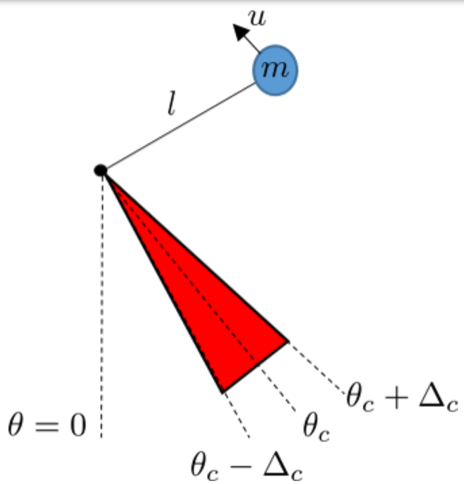
$$\mathbb{P}(\mathrm{CBC}(\mathbf{x}(t), \mathbf{u}_k) \geq 0) \geq p_k = \tilde{p}_k q_k \qquad \forall t \in [t_k, t_k + \tau_k)$$

Approach



1. Bayesian learning
2. Propagate uncertainty to the safety condition
3. Self-triggered control: extension to continous time
4. Extension to higher relative degree systems

Higher relative degree CBFs



$$\mathbf{x} = [\theta, \omega]^\top$$

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}$$

$$f(\mathbf{x}) = \left[\omega, -\tfrac{g}{l}\sin(\theta)\right]^\top \; g(\mathbf{x}) = \left[0, \tfrac{1}{ml}\right]^\top$$

We want to avoid a radial region $[\theta_c - \Delta_c, \theta_c + \Delta_c]$

CBF: $h(\mathbf{x}) = \cos(\Delta_c) - \cos(\theta - \theta_c)$

Notice $\mathcal{L}_g h(\mathbf{x}) = \nabla h(\mathbf{x}) g(\mathbf{x}) = 0$

$\mathrm{CBC}(\mathbf{x}, \mathbf{u}) = \mathcal{L}_f h(\mathbf{x}) + \mathcal{L}_g h(\mathbf{x})\mathbf{u} + \alpha h(\mathbf{x})$ is independent of $\mathbf{u}$

Exponential Control Barrier Functions (ECBF)

Let $r \geq 1$ be the relative degree of $h(\mathbf{x})$, that is, $\mathcal{L}_g \mathcal{L}_f^{(r-1)} h(\mathbf{x}) \neq 0$ and $\mathcal{L}_g \mathcal{L}_f^{(k-1)} h(\mathbf{x}) = 0,\ \forall k \in \{1, \ldots, r-2\}$.

ECBC:

$$\mathrm{CBC}^{(r)}(\mathbf{x}, \mathbf{u}) := \mathcal{L}_f^{(r)} h(\mathbf{x}) + \mathcal{L}_g \mathcal{L}_f^{(r-1)} h(\mathbf{x})\mathbf{u} + K_\alpha \begin{bmatrix} h(\mathbf{x}) \\ \mathcal{L}_f h(\mathbf{x}) \\ \vdots \\ \mathcal{L}_f^{(r-1)} h(\mathbf{x}) \end{bmatrix}$$

If K_α is chosen appropriately, $\mathrm{CBC}^{(r)} \geq 0$ enforce the safety for known dynamics. → Ames et al. (ECC 2019)
Nguyen and Sreenath (ACC 2016)

Chance constraint over ECBC

$$\min_{\mathbf{u}_k \in \mathcal{U}} \|\mathbf{u}_k - \pi(\mathbf{x}_k)\|$$

$$\text{s.t.} \quad \mathbb{P}(\text{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k) \geq \zeta > 0 | \mathbf{x}_k, \mathbf{u}_k) \geq \tilde{p}_k$$

Cantelli's inequality

$$(\mathbb{E}[\text{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)] - \zeta)^2 \geq \frac{\tilde{p}_k}{1-\tilde{p}_k} \text{Var}[\text{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)]$$

$$\mathbb{E}[\text{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)] - \zeta \geq 0$$

We proved $\mathbb{E}[\text{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)]$ and $\text{Var}[\text{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)]$ are Affine and Quadratic in $\mathbf{u}_k$ respectively.

→ QCQP (might be non-convex)

→ Second Order Cone Program (SOCP)

Safe controller using ECBF

$$\min_{\mathbf{u}_k \in \mathcal{U}} \|\mathbf{u}_k - \pi(\mathbf{x}_k)\|$$

$$\text{s.t.} \quad (\mathbb{E}[\mathrm{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)] - \zeta)^2 \geq \tfrac{\tilde{p}_k}{1-\tilde{p}_k} \mathrm{Var}[\mathrm{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)]$$

$$\mathbb{E}[\mathrm{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)] - \zeta \geq 0$$

Solving this program requires the knowledge of the mean and variance of

$$\mathrm{CBC}^{(r)}(\mathbf{x}_k, \mathbf{u}_k)$$

In general, Monte Carlo sampling could be used to estimate these quantities.

Relative degree two $(r = 2)$

We also explicitly quantified $\mathbb{E}[\mathrm{CBC}^{(2)}(\mathbf{x}_k, \mathbf{u}_k)]$ and $\mathrm{Var}[\mathrm{CBC}^{(2)}(\mathbf{x}_k, \mathbf{u}_k)]$ in our paper for relative-degree-two systems.

Algorithm 1: Algorithm to compute Mean and variance of CBF of relative degree 2

Data: Training data $\mathbf{X}$, $\mathcal{U}_{1:k}$ at discretization interval τ. Gaussian process priors $\mathbf{A}$ and $\mathbf{B}_0(\mathbf{x}, \mathbf{x}')$. Test state $\mathbf{x}_*$ and $\mathbf{u}_*$.

Result: $\mathbb{E}[\mathrm{CBC}^{(2)}(\mathbf{x}; \mathbf{u})]$ and $\mathit{Var}(\mathrm{CBC}^{(2)}(\mathbf{x}; \mathbf{u}))$

1 Compute approximate state time derivative $\dot{\mathbf{x}}_t \leftarrow \frac{\mathbf{x}_{t+1} - \mathbf{x}_t}{\tau}$ for all $t \in [1, \dots, d-1]$.

2 Collect $\dot{\mathbf{X}} = [\dot{\mathbf{x}}_1^\top, \dots, \dot{\mathbf{x}}_{d-1}^\top]^\top$.

3 Compute $\mathbf{M}_k(\mathbf{x}_*)$ and $\mathbf{B}_k(\mathbf{x}_*, \mathbf{x}_*)$ from (12).

4 Compute mean and variance of $\mathcal{L}_{f_k} h(\mathbf{x}) = \nabla h(\mathbf{x})^\top f(\mathbf{x})$ using Corollary 2

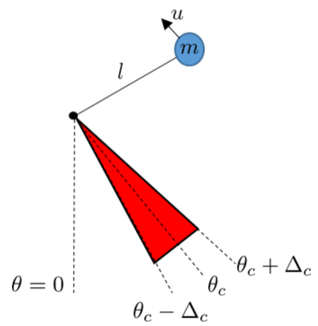
5 Compute mean, variance and covariance of $\nabla \mathcal{L}_{f_k} h(\mathbf{x})$ using Lemma 3,

6 Compute mean, variance and covariance of $\nabla[\mathcal{L}_{f_k} h(\mathbf{x})]^\top F(\mathbf{x})\underline{\mathbf{u}}$ using Lemma 2,

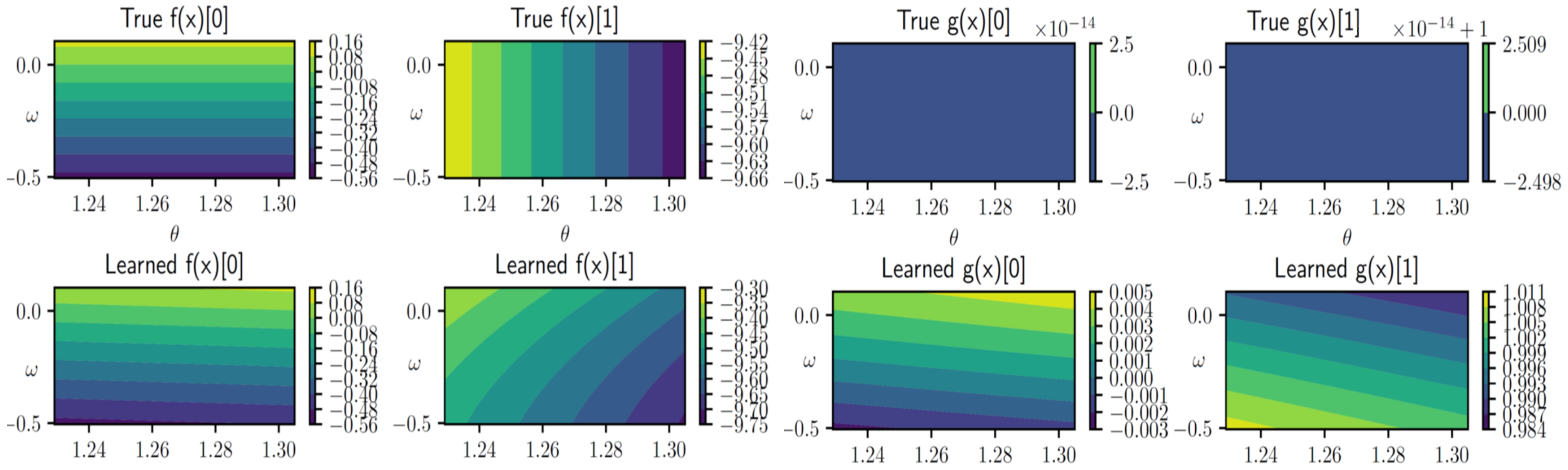
7 Plug the above values into Theorem 1 to get $\mathbb{E}[\mathrm{CBC}^{(2)}(\mathbf{x}; \mathbf{u})]$ and $\mathit{Var}(\mathrm{CBC}^{(2)}(\mathbf{x}; \mathbf{u}))$.

Bipedal and car-like robots are examples of these systems.

Example



$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}$$



Outline

Part I: Safety

1. Probabilistic Safety Constraints for Learned High Relative Degree System
2. Safe Multi-Agent Interaction through CBF with Learned Uncertainties

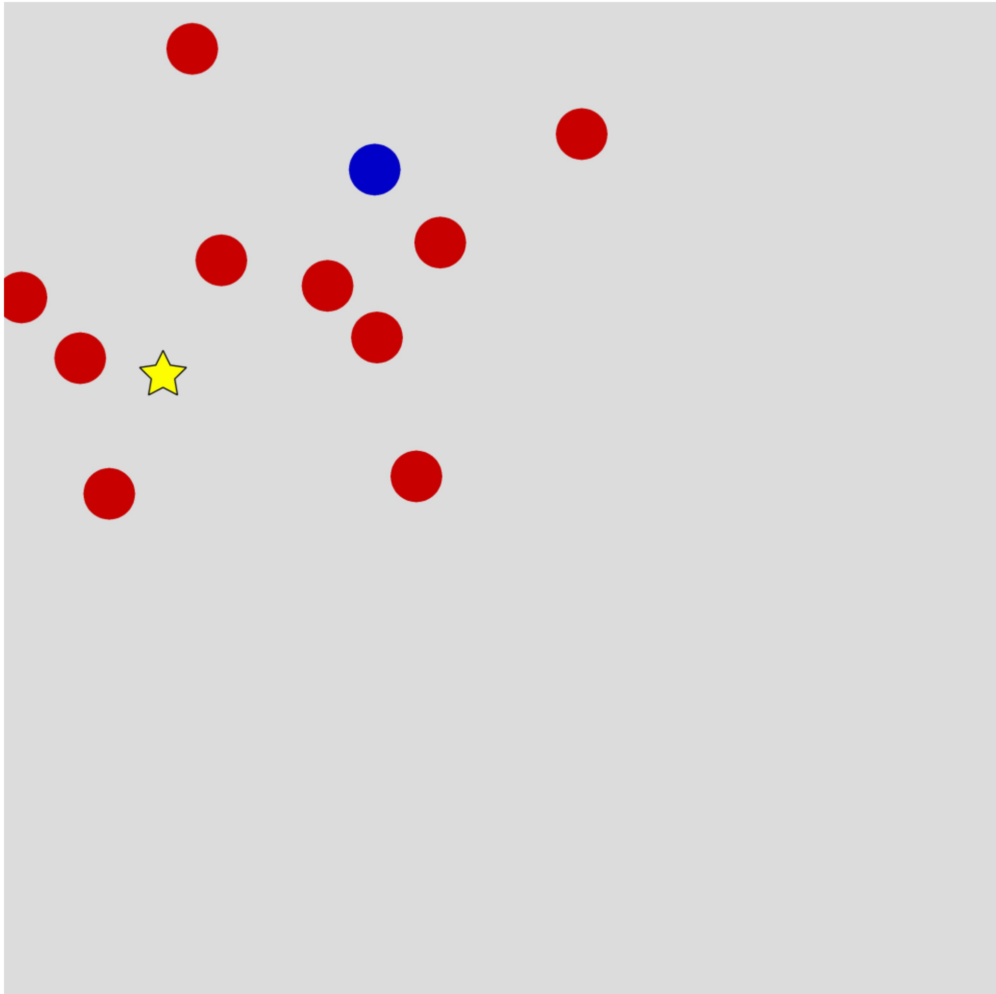
Joint work with:

- Richard Cheng, Caltech
- Aaron D. Ames, Caltech
- Joel W. Burdick, Caltech

Part II: Security

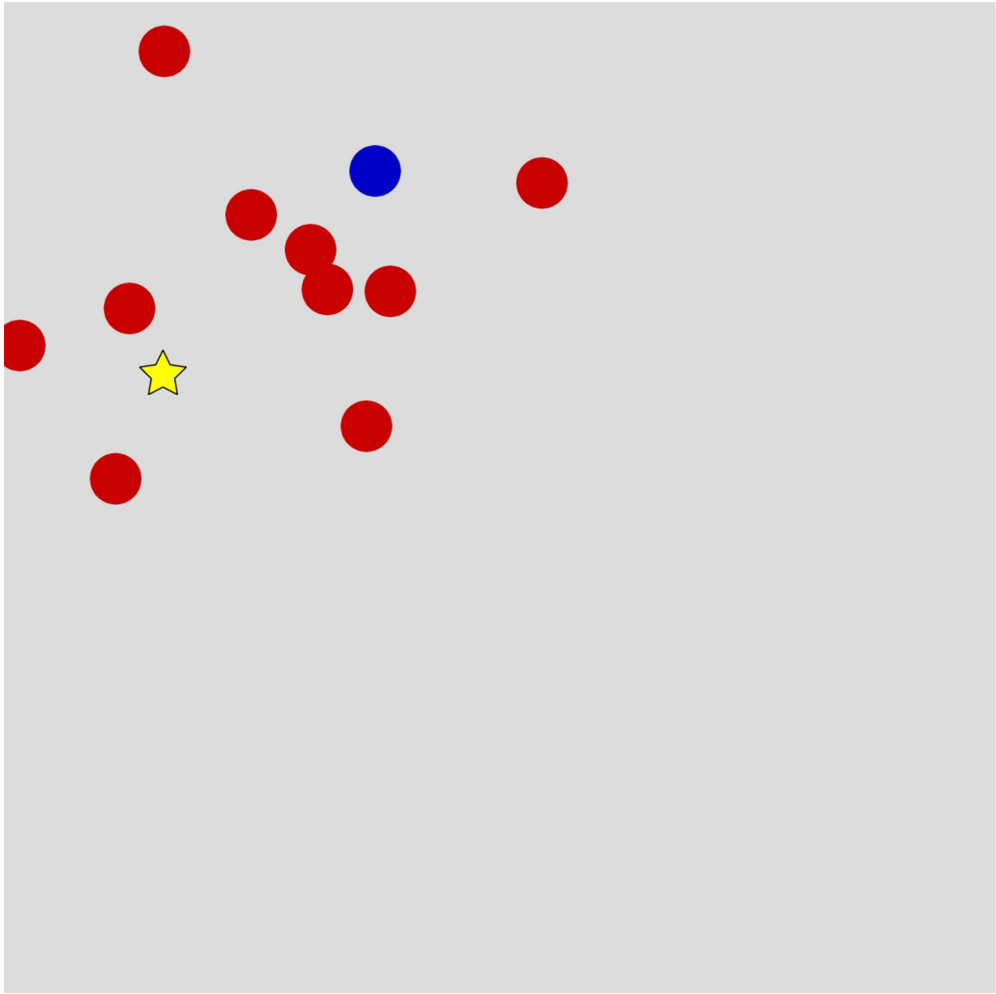
Learning-based attacks in cyber-physical systems

Navigation in Unstructured Environment



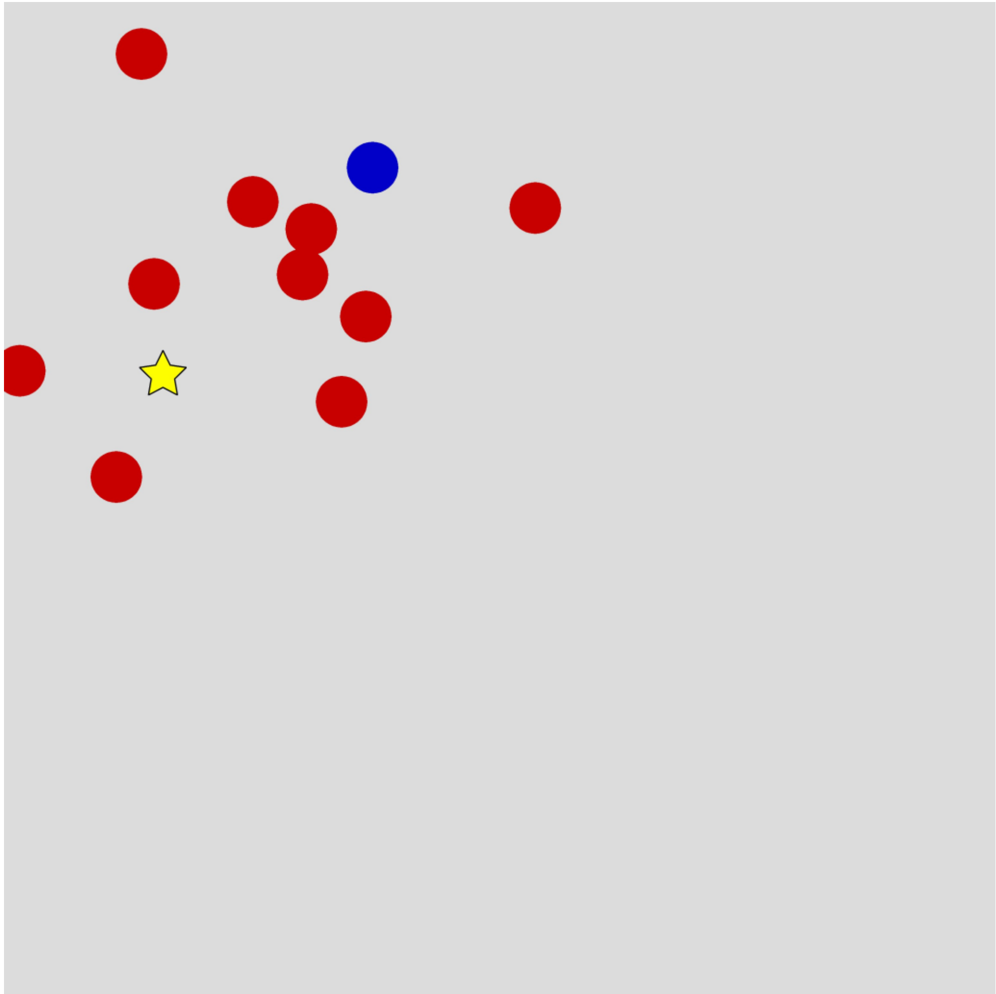
The robot (blue) tries to navigate from a start position to random goal position (yellow star) while avoiding collisions with other agents (red)

Navigation in Unstructured Environment



Approximately half of the other agents blindly travel towards their own randomly chosen goal, while the rest exhibit varying degrees of collision-avoidance behavior (the robot does not know their behavior apriori)

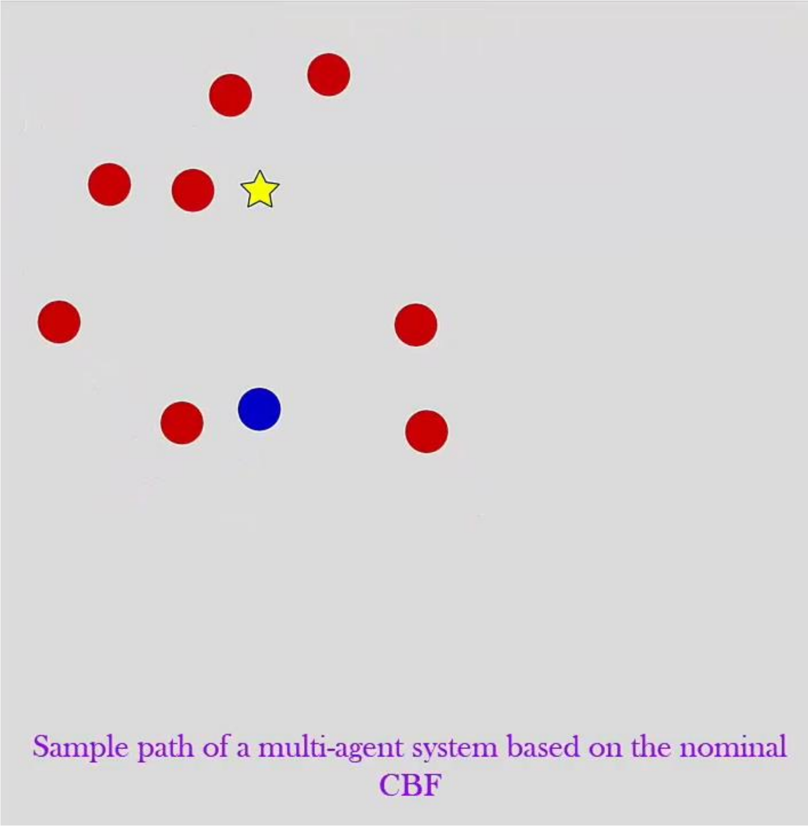
Navigation in Unstructured Environment



Example 1: Sample path of a multi-agent system based on the nominal CBF

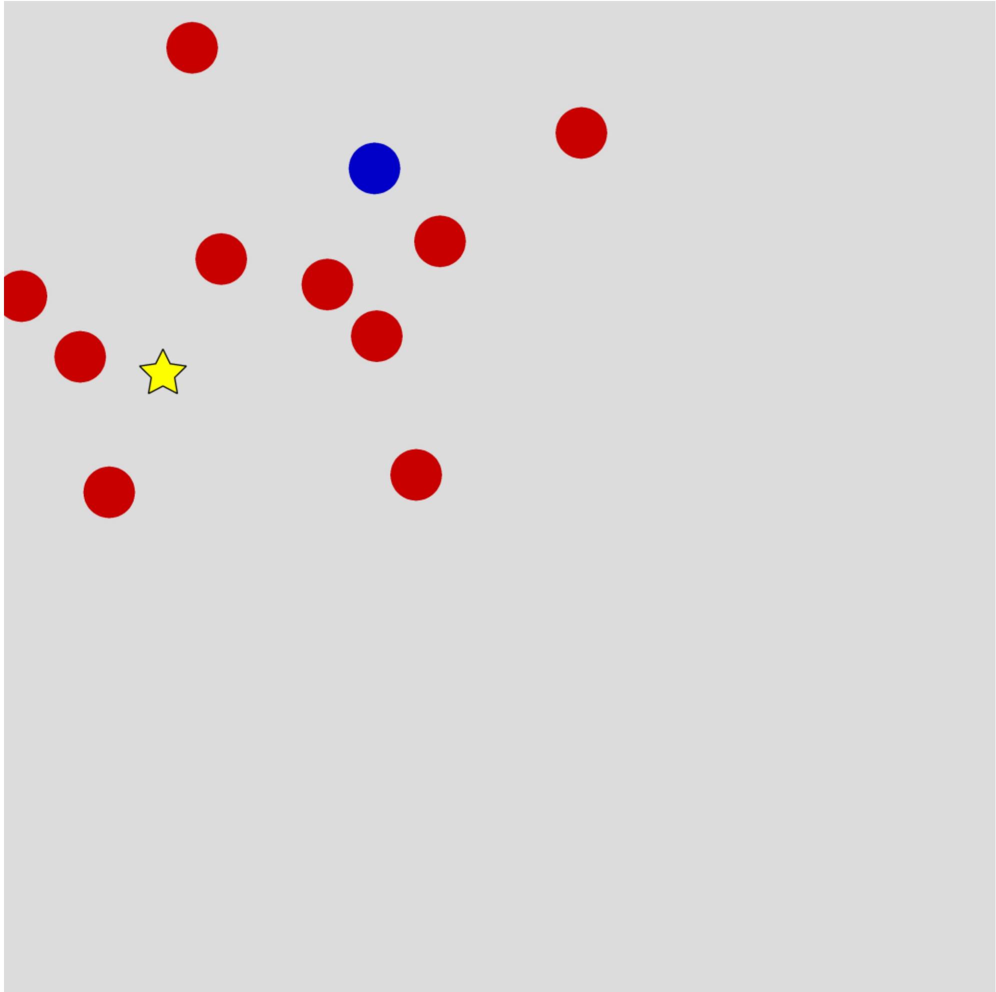
Borrmann et al. (IFAC 2015)

Navigation in Unstructured Environment



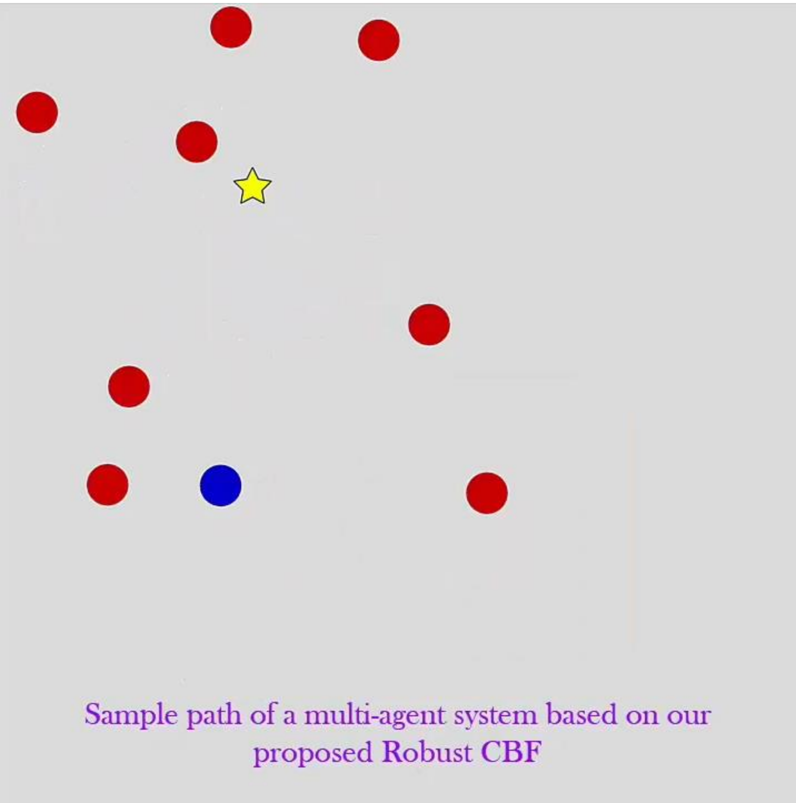
https://youtu.be/hXg5kZO86Lw

Navigation in Unstructured Environment



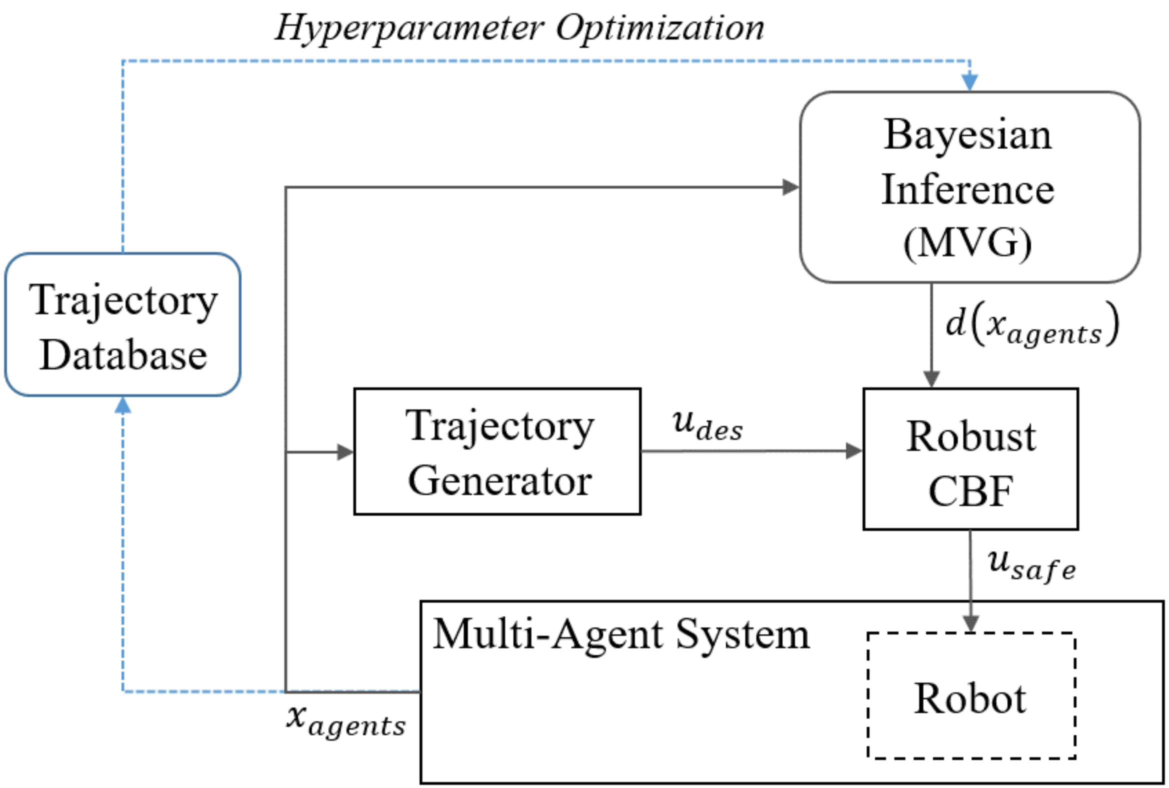
Example 2: Sample path of a multi-agent system based on our proposed Robust CBF

Navigation in Unstructured Environment

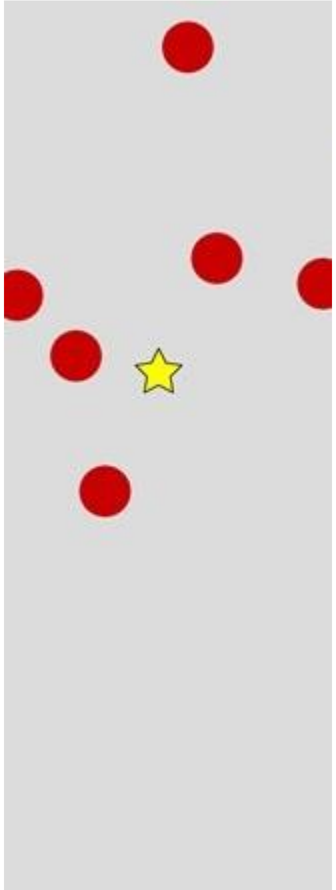


https://youtu.be/hXg5kZO86Lw

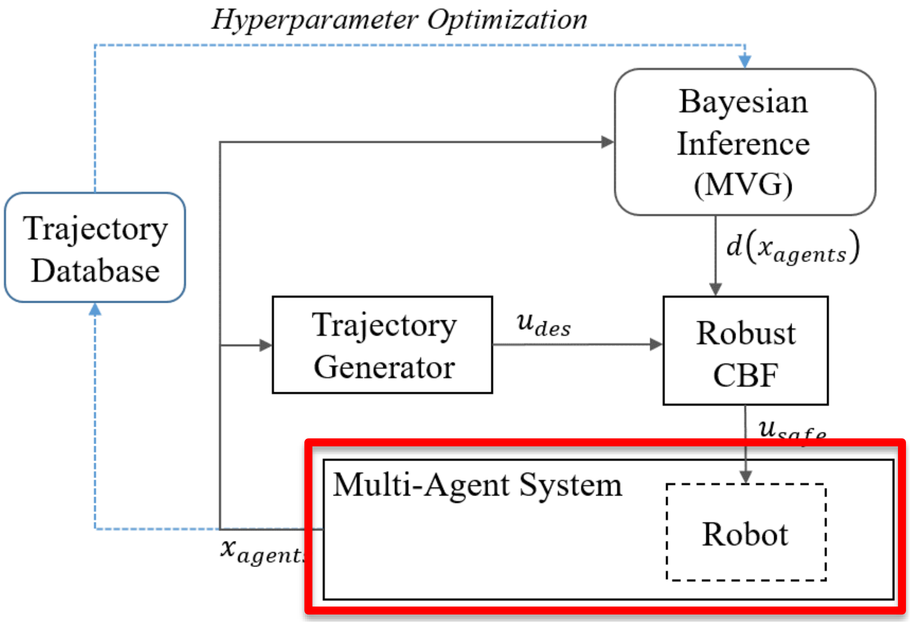
Overview of the the control structure



Approach



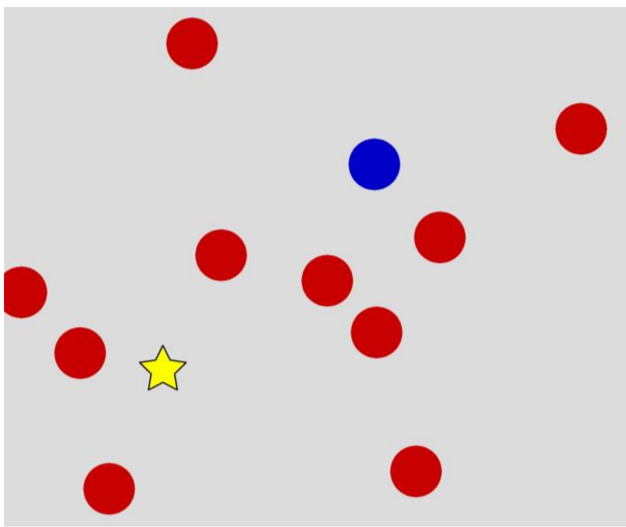
1. Multi-agent CBF
2. Incorporating Robustness into CBF
3. Learning Uncertity bound



Multi-agent system

Our robot dynamics

$$x_{t+1} = \begin{bmatrix} p_{t+1} \\ v_{t+1} \\ z_{t+1} \end{bmatrix} = \underbrace{\begin{bmatrix} f_p(x_t) \\ f_v(x_t) \\ f_z(x_t) \end{bmatrix}}_{f(x_t)} + \underbrace{\begin{bmatrix} g_p(x_t) \\ g_v(x_t) \\ g_z(x_t) \end{bmatrix}}_{g(x_t)} u + \underbrace{\begin{bmatrix} d_p(x_t) \\ d_v(x_t) \\ d_z(x_t) \end{bmatrix}}_{d(x_t)}$$



f and g are known

d is unknown

$p \in \mathbb{R}^2$ position

$v \in \mathbb{R}^2$ velocity

$z \in \mathbb{R}^{n-4}$ other states

$\|u\|_2 \leq u_{max}$

actuation bound

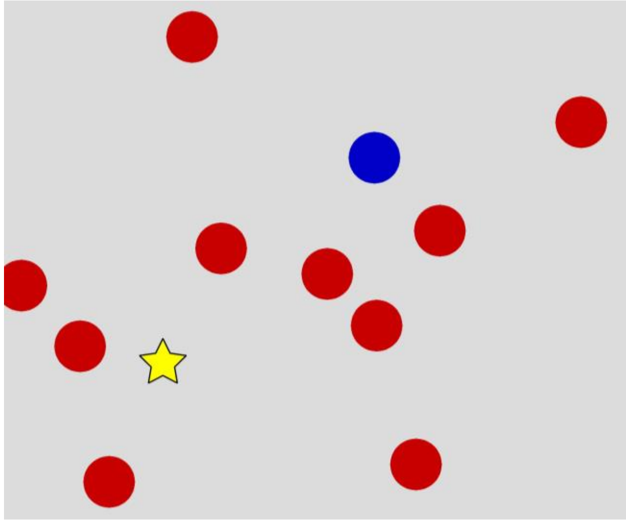
$g_p(x) = 0_{2\times 2}$

system has relative degree 2 w.r.t. position

Multi-agent system

Other agents

$$x_{t+1}^{(i)} = \begin{bmatrix} p_{t+1}^{(i)} \\ v_{t+1}^{(i)} \\ z_{t+1}^{(i)} \end{bmatrix} = \underbrace{\begin{bmatrix} f_p^{(i)}(x_t) \\ f_v^{(i)}(x_t) \\ f_z^{(i)}(x_t) \end{bmatrix}}_{f^{(i)}(x_t)} + \underbrace{\begin{bmatrix} d_p^{(i)}(x_t) \\ d_v^{(i)}(x_t) \\ d_z^{(i)}(x_t) \end{bmatrix}}_{d^{(i)}(x_t)}$$



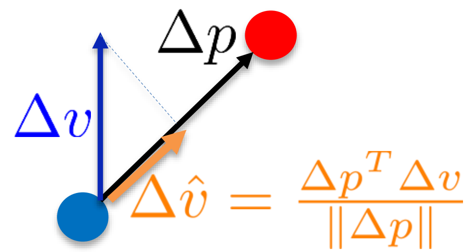
f is known

d is unknown

We assume the control input for other agents are a function of their state (we do not show their control inputs explicitly)

Multi-agent control barrier functions (MA-CBF)

$$h(x) = \underbrace{\frac{\Delta p^T \Delta v}{\|\Delta p\|}}_{\Delta \hat{v}} + \sqrt{a_{max}(\|\Delta p\| - D_s)}$$



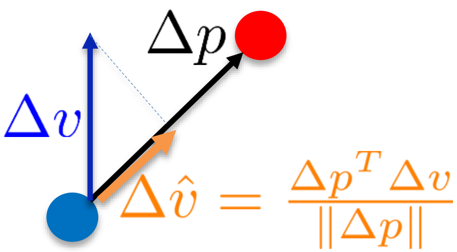
$\Delta p = p - p^{(i)}$	positional difference between the agents
$\Delta v = v - v^{(i)}$	velocity difference between the agents
$\Delta \hat{v} = \frac{\Delta p^T \Delta v}{\|\Delta p\|}$	velocity porojected in the direction of collision
a_{max}	our robot's max acceleration in the collision direction
D_s	collision margin

Multi-agent control barrier functions (MA-CBF)

collision can be avoided if we match the other agents velocity by the time we reach them

a_{max} our robot's max acceleration in any direction

We can achieve $\Delta\hat{v} = 0$ within time $T_c = \frac{-\Delta\hat{v}(x_t)}{a_{max}}$



collision avoidance is guaranteed:

$$\Delta\hat{v}(x_t)T_c + ||\Delta p|| \geq D_s$$

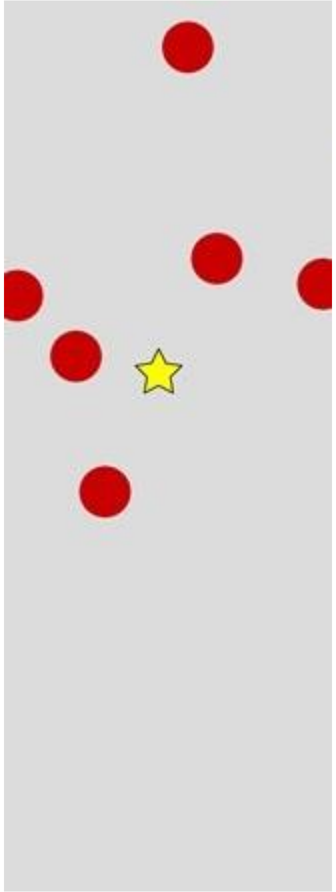
$$h(x) = \Delta\hat{v} + \sqrt{a_{max}(||\Delta p|| - D_s)} \geq 0$$

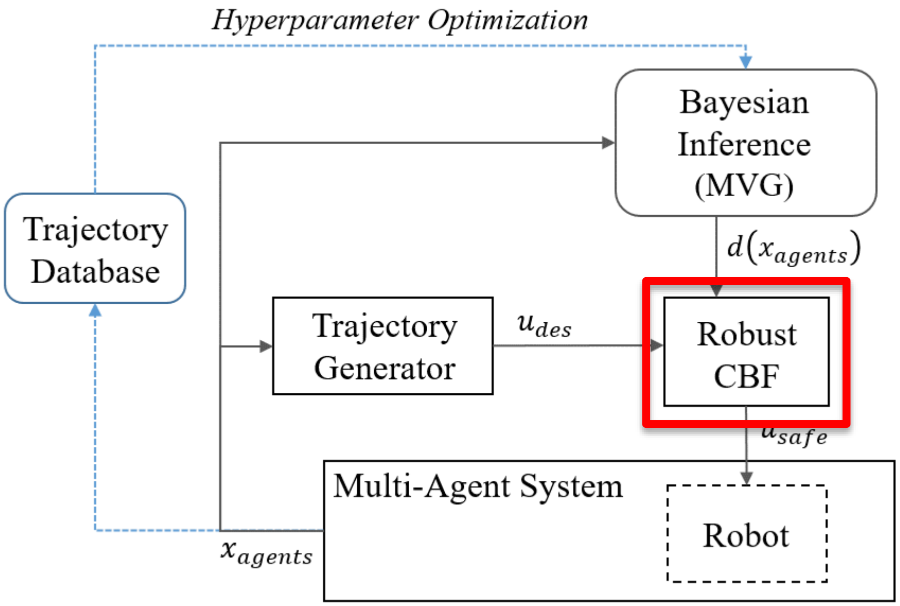
provided the acceleration is sufficiently large $a_{max}(u_{max}) > c'(d)$

The parameter c' is calculated in our paper

Approach

1. Multi-agent CBF
2. Incorporating Robustness into CBF
3. Learning Uncertity bound





Robust multi-agent CBF

$$h(x) = \Delta \hat{v} + \sqrt{a_{max}(\|\Delta p\| - D_s)}$$

$$CBC(x_t, u_t) = h(x_{t+1}(u_t)) + (\eta - 1)h(x_t)$$

$$\boxed{\begin{aligned} &\min_u \|u - u_{des}\| \\ &\text{s.t.} \quad \min_{d(x_t)} CBC(x_t, u, d_t) \geq 0 \\ &\qquad \text{where } d(x_t) \in \mathcal{D} \\ &\qquad \|u\| \leq u_{max} \end{aligned}}$$

$\rightarrow$ nonlinear (not convex)

polytopic bounds on the uncertainties: $\{d \in \mathbb{R}^n \mid Gd \leq g\}$

lower bound on CBC:

$$CBC(x_t, u_t, d_t) \geq k_c(x_t) - H_1(x_t)d_t - u_t^T H_2(x_t)d_t - H_3(x_t)u_t$$

The parameters are calculated in our paper

Robust multi-agent CBF

polytopic bounds on the uncertainties: $\{d \in \mathbb{R}^n \mid Gd \le g\}$

lower bound on CBC:

$$CBC(x_t, u_t, d_t) \ge k_c(x_t) - H_1(x_t)d_t - u_t^T H_2(x_t)d_t - H_3(x_t)u_t$$

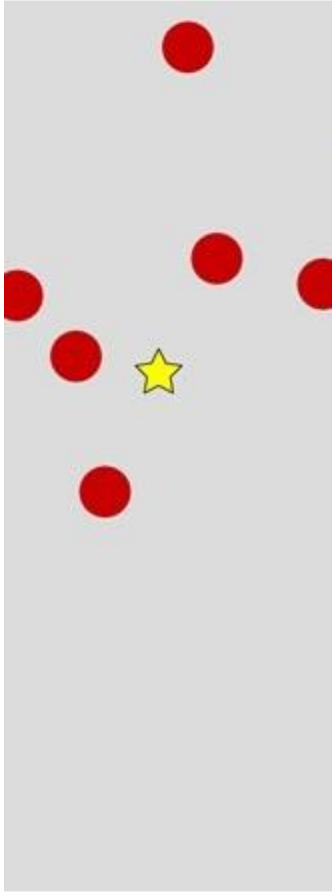
$$\begin{aligned}
&\min_u \|u - u_{des}\| \\
&\quad \text{s.t.} \quad \min_{d(x_t)} CBC(x_t, u, d_t) \ge 0 \\
&\qquad\quad \text{where } d(x_t) \in \mathcal{D} \\
&\qquad\quad \|u\| \le u_{max}
\end{aligned}$$

$\rightarrow$

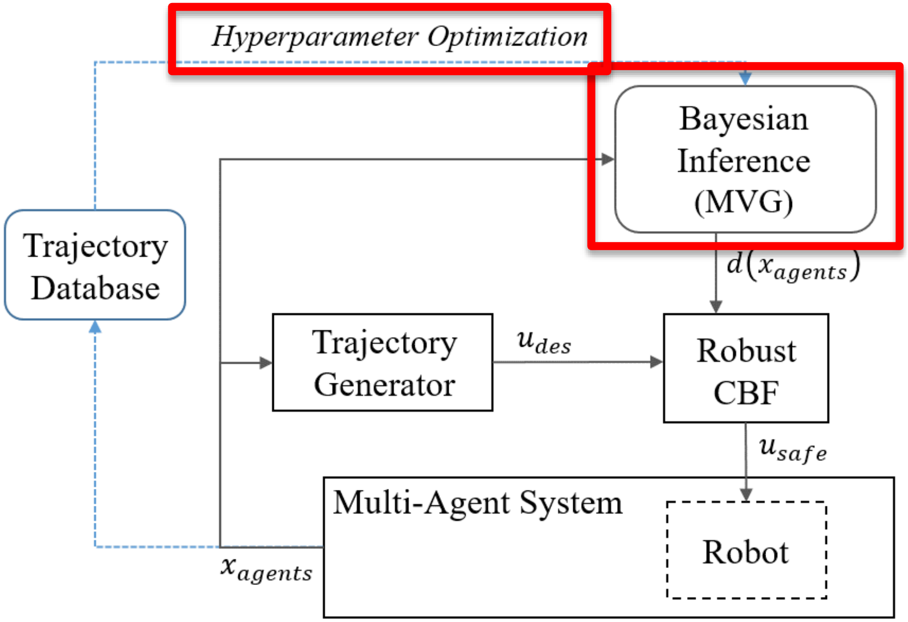
$$\begin{aligned}
\min_{u,\xi} \quad & \|u - u_{des}\|_2 \\
\text{s.t.} \quad & H_3(x_t)u + \xi g \le k_c(x_t) \\
& H_1(x_t) + u^T H_2(x_t) = \xi G \\
& \xi \ge \mathbf{0} \\
& \|u\| \le u_{max}
\end{aligned}$$

QP

Approach



1. Multi-agent CBF
2. Incorporating Robustness into CBF
3. Learning Uncertity bound



Hyperparameter optimization

Bayesian learning (Matrix-Variate Gaussian Process)

$$vec(d(x_1), \ldots, d(x_N)) \sim \mathcal{N}(\mathbf{0},\ \Sigma(x) \otimes \Omega)$$

$$\Sigma_{i,j} = \kappa(x_i, x_j)$$

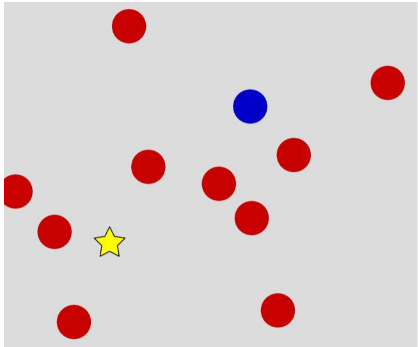
$$\kappa(x_i, x_j) = \sigma^2 \exp\left(\frac{-\|x_i - x_j\|^2}{2l^2}\right)$$

some agents might behave predictably and others might behave more erratically, and hyperparameter optimization is necessary to capture these uncertainty profiles in our Bayesian inference

We optimize kernel parameters

$$\sigma, l, \Omega$$

to obtain better prior.
(We learn them offline from data)



Learning Uncertity bound (online)

Bayesian learning (Matrix-Variate Gaussian Process)

$$vec(d(x_1), \ldots, d(x_N)) \sim \mathcal{N}(\mathbf{0},\ \Sigma(x) \otimes \Omega)$$

Posterior mean

Posterior variance

$$(d - \mu_d)^T \Sigma_d^{-1} (d - \mu_d) \sim \chi_N^2$$

$$(d - \mu_d)^T \Sigma_d^{-1} (d - \mu_d) \leq k_\delta$$

with probability $1 - \delta$

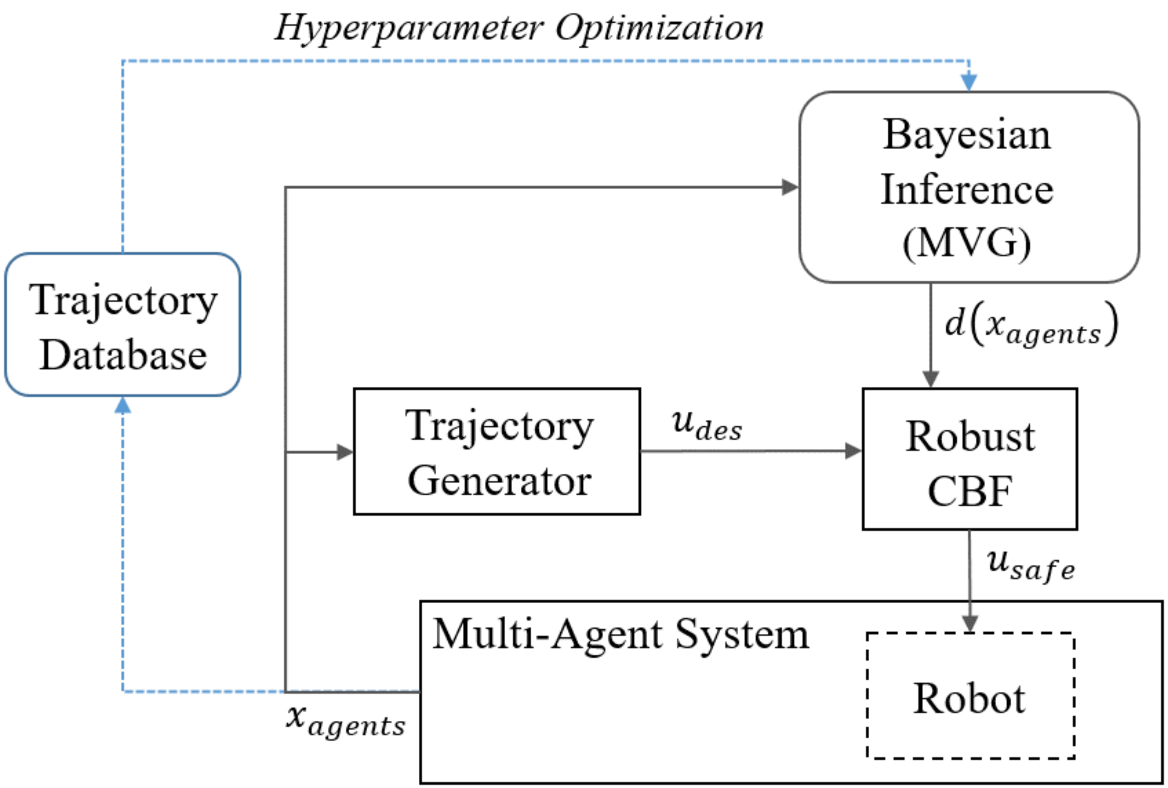
Polytopic bounds

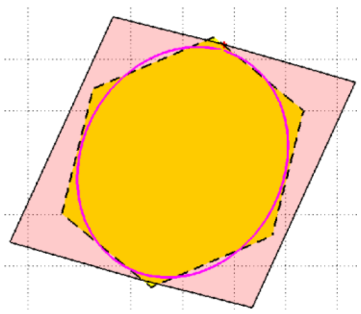
QP

$$\begin{aligned}
\min_{u,\xi} \quad & \|u - u_{des}\|_2 \\
\text{s.t.} \quad & H_3(x_t)u + \xi g \leq k_c(x_t) \\
& H_1(x_t) + u^T H_2(x_t) = \xi G \\
& \xi \geq \mathbf{0} \\
& \|u\| \leq u_{max}
\end{aligned}$$

High-Confidence Safety Guarantee

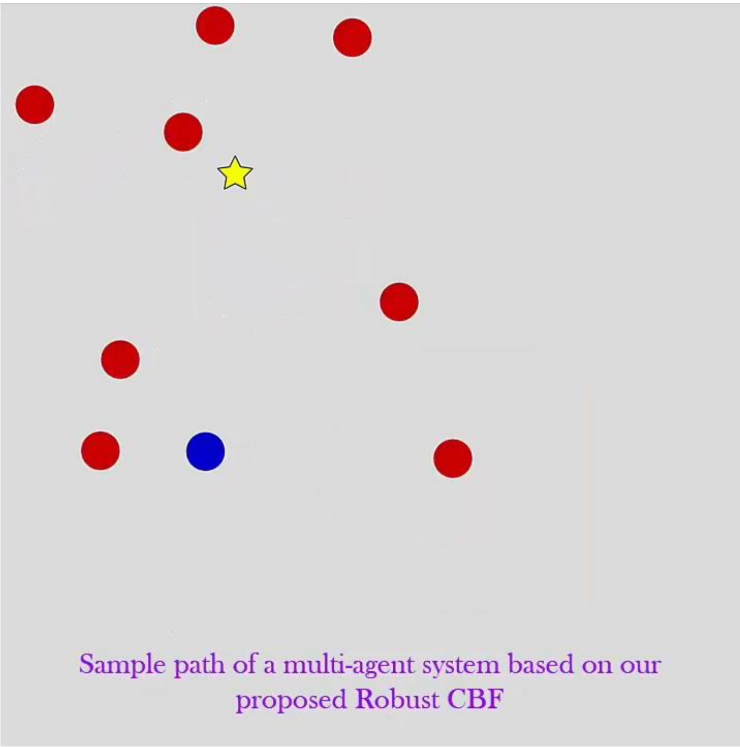
Overview of the the control structure





QP

Navigation in Unstructured Environment



https://youtu.be/hXg5kZO86Lw

By running 1000 simulated tests in randomized environments, we show that our robust CBF avoids collision in 98.5% of cases performing much better than the nominal multi-agent CBF, which avoids collisions in 85.0% of cases.

Outline

Part I: Safety

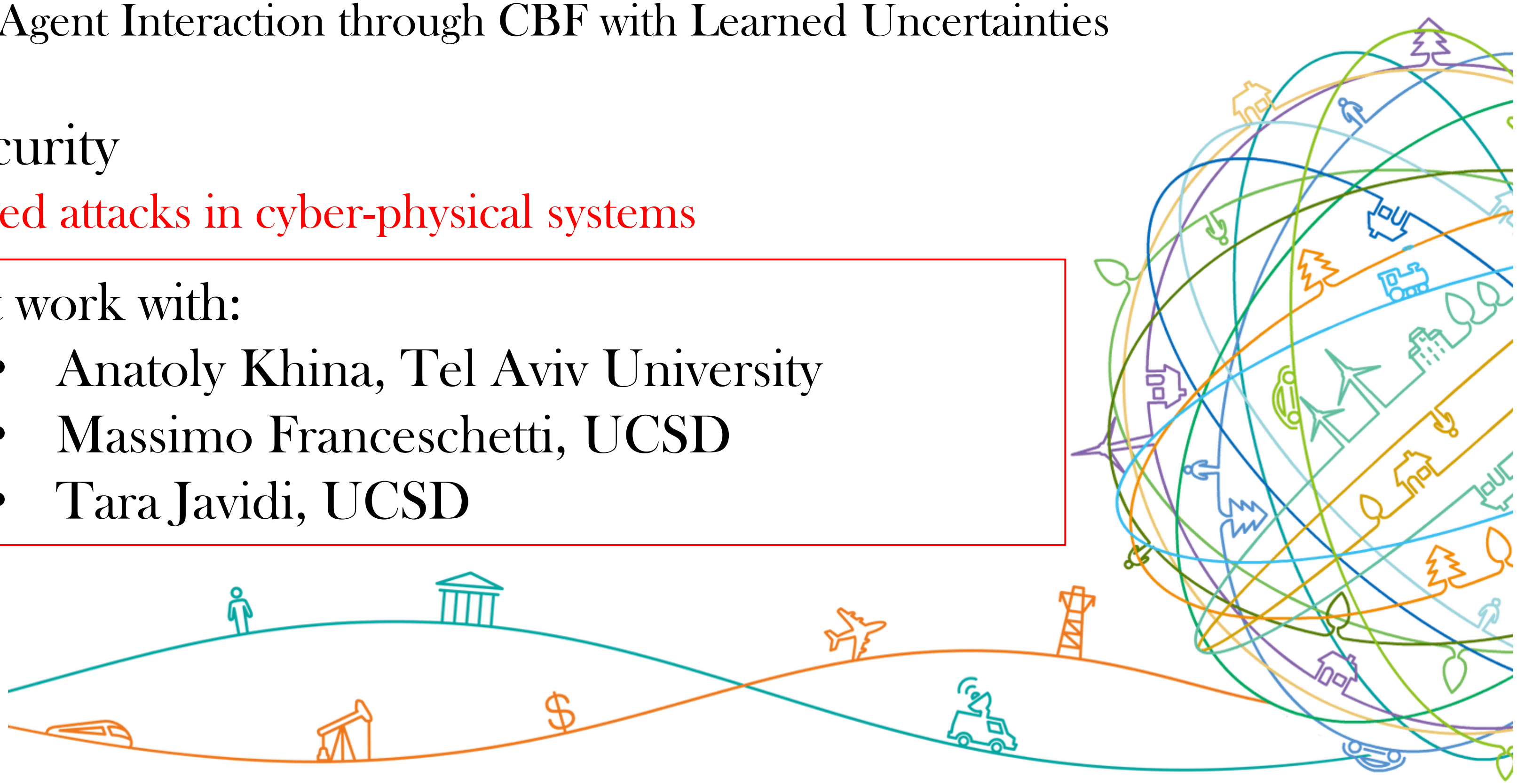
1. Probabilistic Safety Constraints for Learned High Relative Degree System
2. Safe Multi-Agent Interaction through CBF with Learned Uncertainties

Part II: Security

Learning-based attacks in cyber-physical systems

Joint work with:

- Anatoly Khina, Tel Aviv University
- Massimo Franceschetti, UCSD
- Tara Javidi, UCSD



Cloud robots and automation systems

Security

We need to address physical security in addition to cyber security

News reports

Port of San Diego suffers cyber-attack, second port in a week after Barcelona

Hacker jailed for revenge sewage attacks

Job rejection caused a bit of a stink

HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY—WITH ME IN IT

News reports

The Stuxnet outbreak

A worm in the centrifuge

The Economist

An unusually sophisticated cyber-weapon is mysterious but important

Computer virus Stuxnet a 'game changer,' DHS official tells Senate

CNN

Symantec

"It has changed the way we view the security threat"

The man in the middle

A fictitious plant for the controller

A malicious controller for the plant

Mathematical formulation

- Linear dynamical system

$$X_{k+1} = aX_k + U_k + W_k$$

$$\{W_k\} \text{ are i.i.d. } \mathcal{N}(0, Var[W])$$

- The controller, at time k, observes Y_k and generates a control signal U_k as a function of all past observations Y_1^k.

$Y_k = X_k$ Under normal operation

$Y_k = V_k$ Under attack

- The attacker feeds a malicious input $\tilde{U}_k$ to the plant.

- How can the controller detect that the system is under attack?

Anomaly detection

- The controller is armed with a detector that tests for anomalies in the observed history Y_1^k.

$$X_{k+1} = aX_k + U_k + W_k \qquad \{W_k\} \text{ are i.i.d. } \mathcal{N}(0, Var[W])$$

- Under legitimate system operation $(Y_k = X_k)$ we expect

$$Y_{k+1} - aY_k - U_k(Y_1^k) \sim \quad \text{i.i.d.} \quad \mathcal{N}(0, Var[W])$$

- The detector performs the variance test

$$Var[W] = \mathbb{E}[W^2]$$

Anomaly detection

- Under legitimate system operation we expect

$$Y_{k+1} - aY_k - U_k(Y_1^k) \sim \quad \text{i.i.d.} \quad \mathcal{N}(0, Var[W])$$

- The controller performs a threshold-based detection

$$\frac{1}{T}\sum_{k=1}^{T}\left[Y_{k+1} - aY_k - U_k(Y_1^k)\right]^2 \in (Var[W] - \delta, Var[W] + \delta).$$

- What kind of attacks can we detect?

The man in the middle attack types

Replay attack

Stuxnet

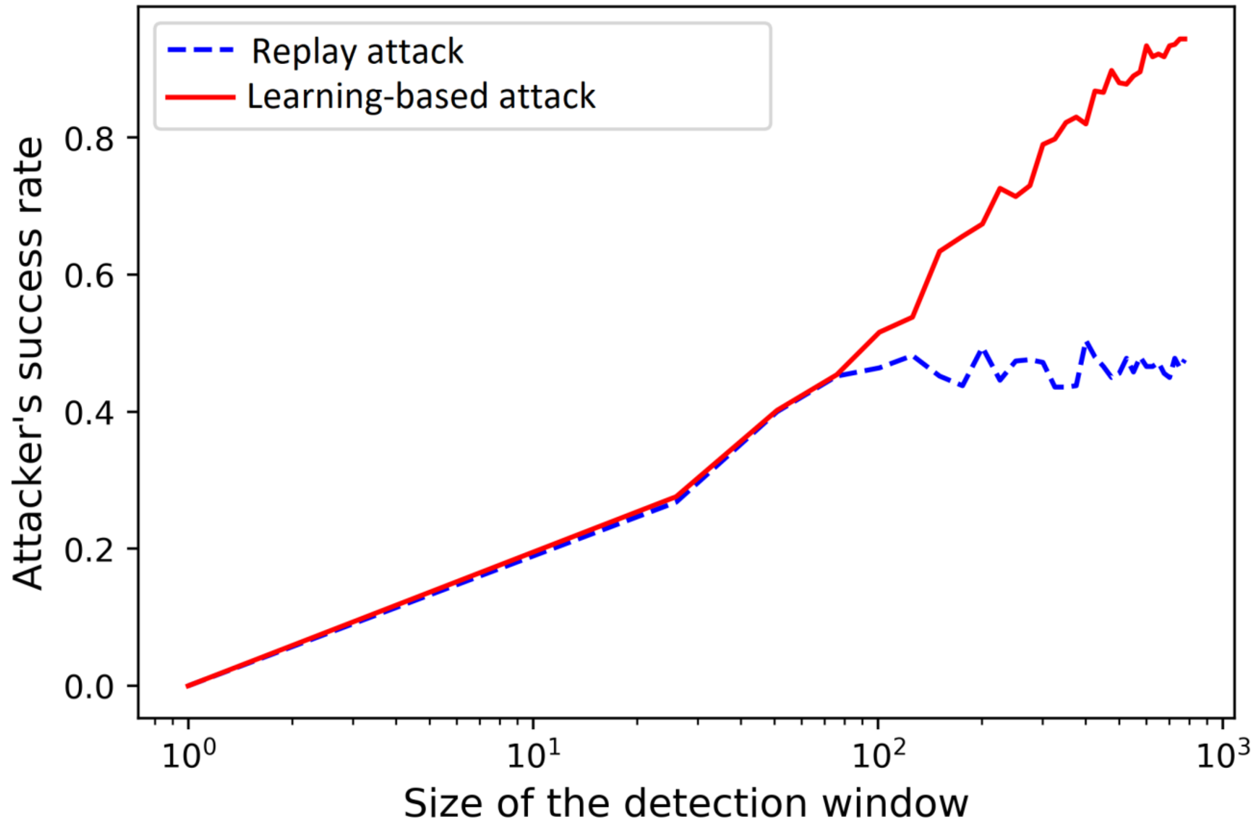
Y. Mo, B. Sinopoli (2009)

Learning-based attack

$$X_{k+1} = aX_k + U_k + W_k$$

MJ Khojasteh et al. (2019)

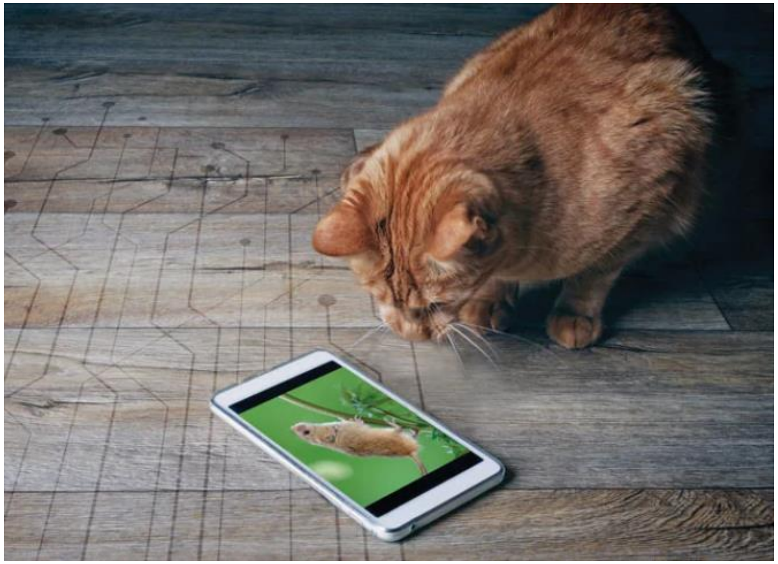
Comparison with a replay attack



MJ Khojasteh et al. (2019)

Stuxnet

Defense against learning-based attack

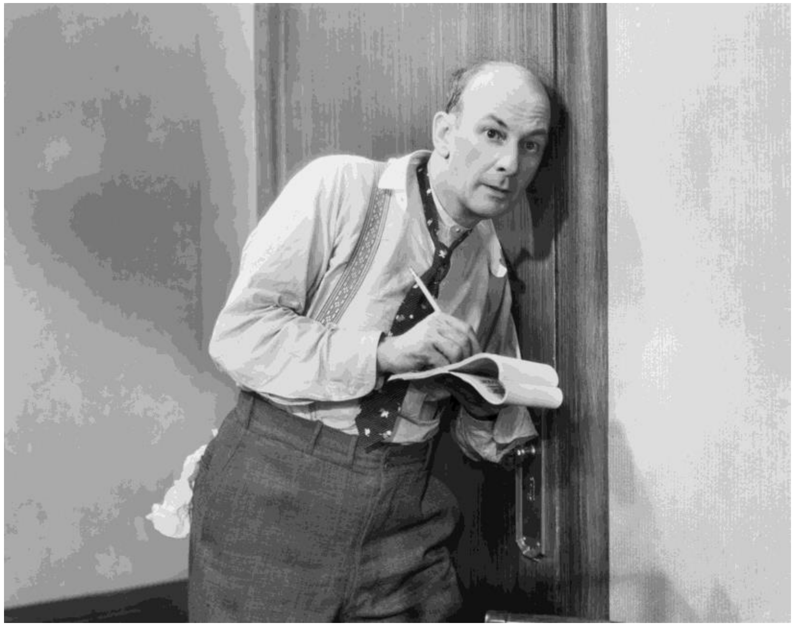


$$X_{k+1} = aX_k + U_k + W_k.$$

- The attacker has access to both X_k and U_k and knows the distribution of W_k and of the initial condition X_0, but it should learn the open loop gain a of the plant.

Two phases of the learning-based attack

Learning (exploration) phase



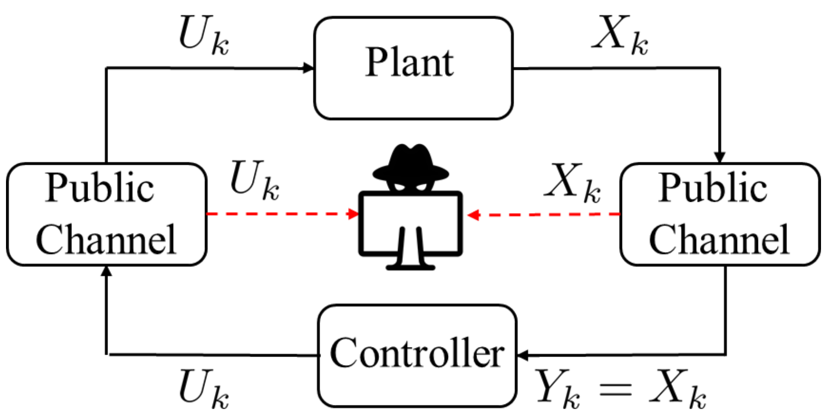
Eavesdropping and learning

Hijacking (exploitation) phase

Hijacking the system

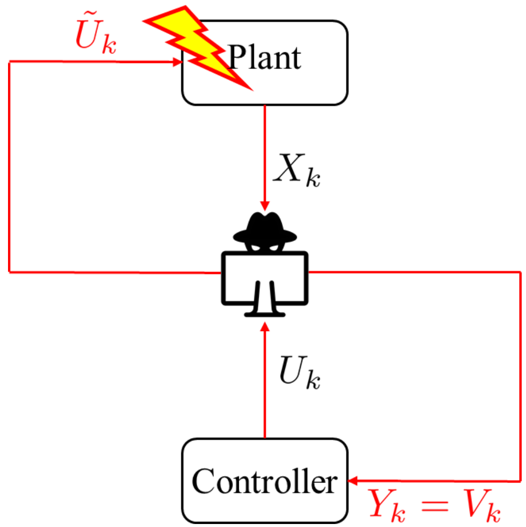
Two phases of the learning-based attack

Learning (exploration) phase



Eavesdropping and learning

Hijacking (exploitation) phase



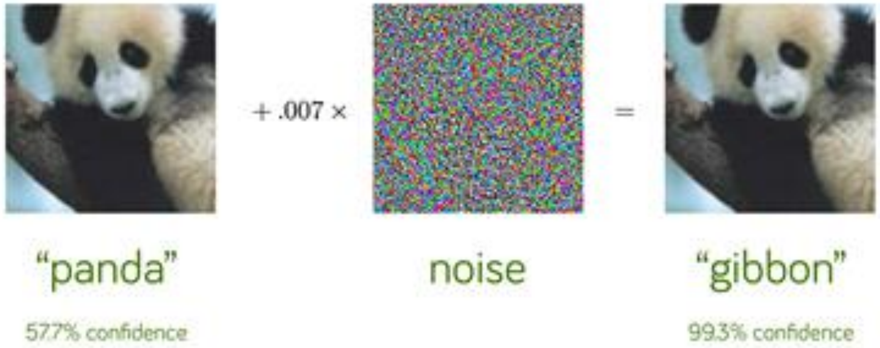
Hijacking the system

Defense against learning-based attack

Impede the learning process of the attacker

$$U_k = \text{modify}(\underbrace{\bar{U}_k}_{\text{Nominal control policy}})$$

The controller, by potentially sacrificing the optimally of the control task, can act in an adversarial machine learning setting



Defense against learning-based attack

Controller

knows the dyanamics

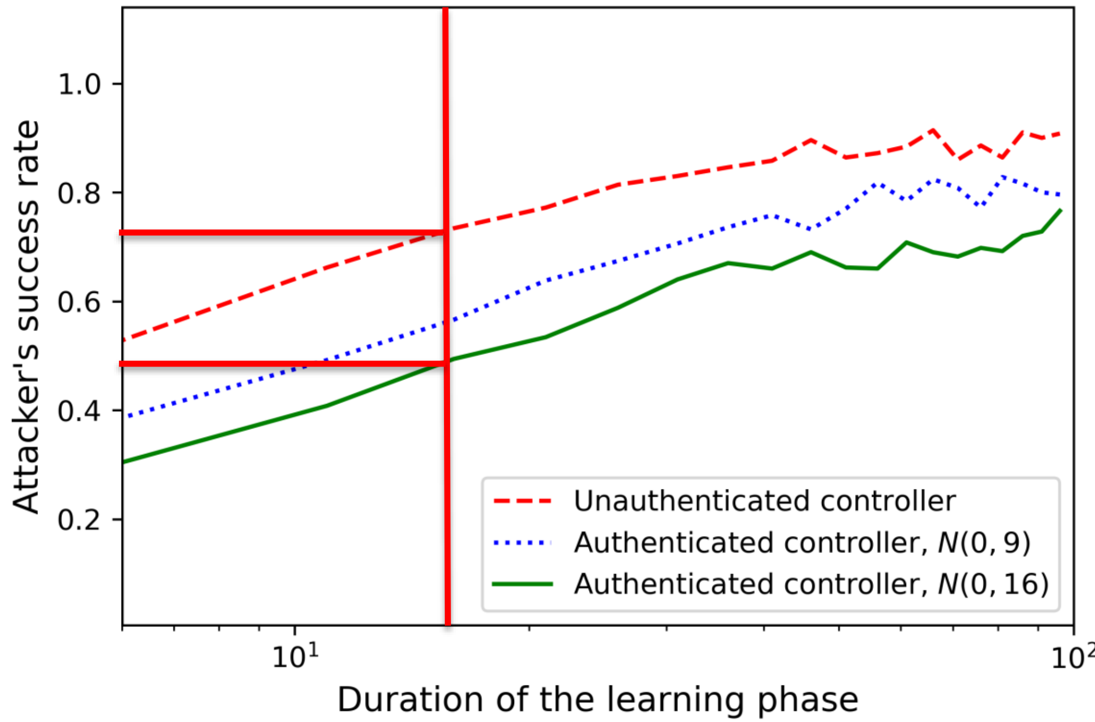
wants to Learn the dyanamics

$$\min_{U_k} \|U_k - \bar{U}_k\|$$

$$I(f; X_1^L, U_1^L)$$

to enhance the dyanamics privacy

Privacy-enhancing signal

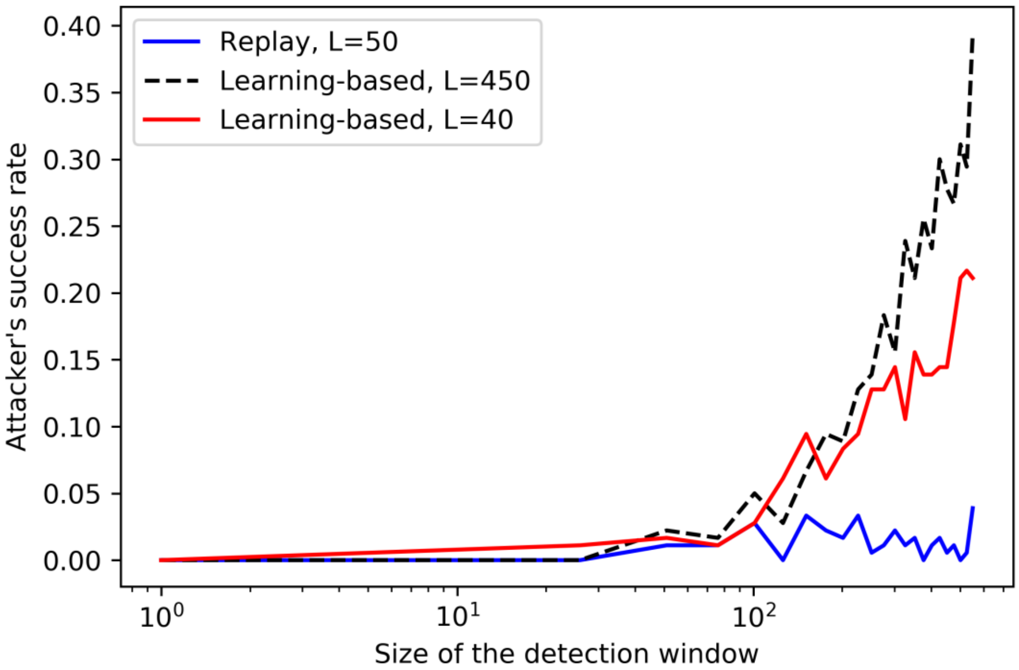


MJ Khojasteh et al. (2019)

Learning-based attack: vector systems

$$A \longrightarrow \mathbf{A} = \begin{bmatrix} ? \end{bmatrix}$$

$$\mathrm{Var} \longrightarrow \mathrm{Cov}$$



MJ Khojasteh et al. (2019)

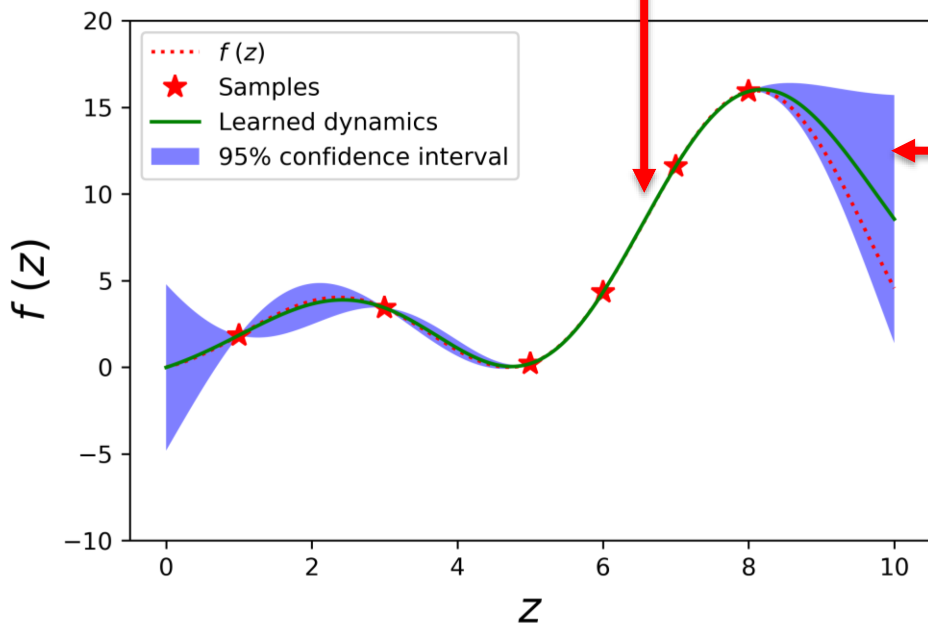
Stuxnet

Nonlinear learning-based attack

$A \longrightarrow f(X, U) \in$ Reproducing Kernel Hilbert Space (RKHS)

Linear regression $\longrightarrow$ Bayesian learning: Gaussian processes (GP)

Vulnerable region



Lower attacker's success rate

References

- Khojasteh MJ, Dhiman V, Franceschetti M, Atanasov N
  Probabilistic safety constraints for learned high relative degree system dynamics.
  Learning for Dynamics and Control. 2020, July; 781-792
- Cheng R, Khojasteh MJ, Ames A D, Burdick JW
  Safe multi-agent interaction through robust control barrier functions with learned uncertainties.
  59th IEEE Conference on Decision and Control (CDC 2020)
- Khojasteh MJ, Khina A, Franceschetti M, Javidi T
  Authentication of cyber-physical systems under learning-based attacks
  IFAC-PapersOnLine. 2019 Jan 1; 52(20): 369-74
- Khojasteh MJ, Khina A, Franceschetti M, Javidi T
  Learning-based attacks in cyber-physical systems
  arXiv preprint arXiv:1809.06023, 2020