Trust-aware Control for Multi-Agent Systems: a future arena for hyperproperties?

Jyotirmoy (Jyo) V. Deshmukh
CS @ University of Southern California

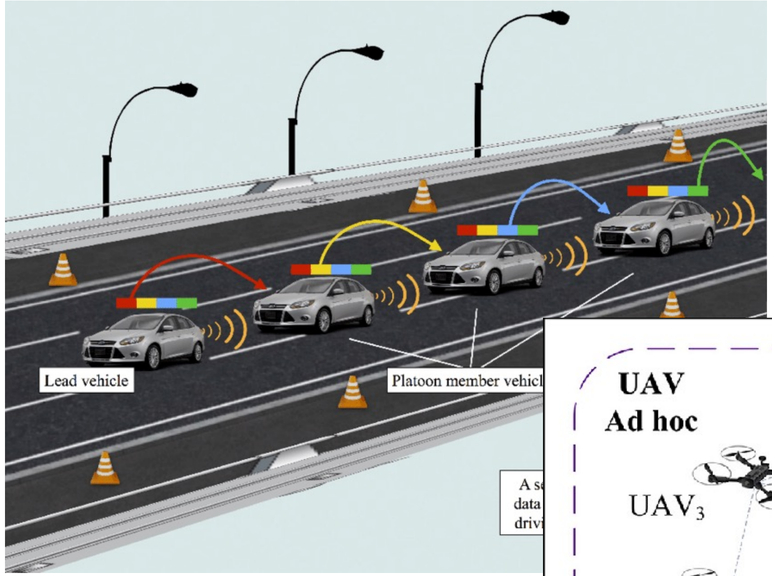
Key collaborators:

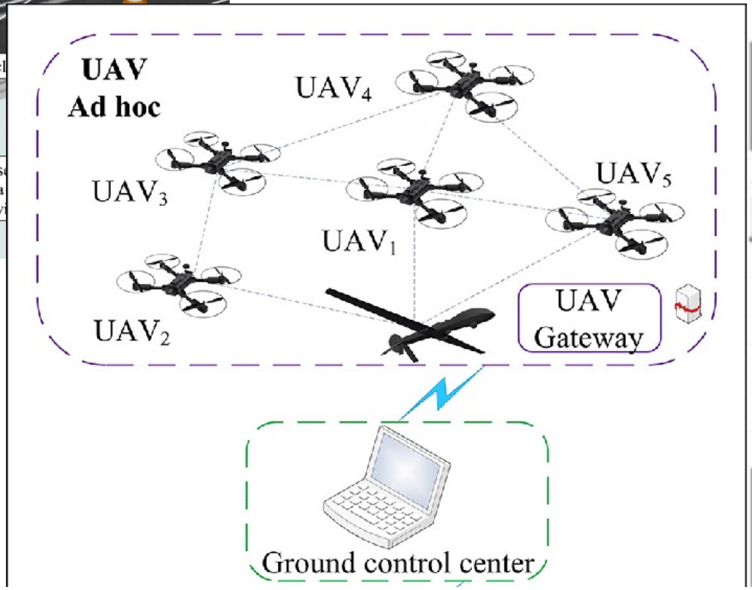
Mingxi Cheng

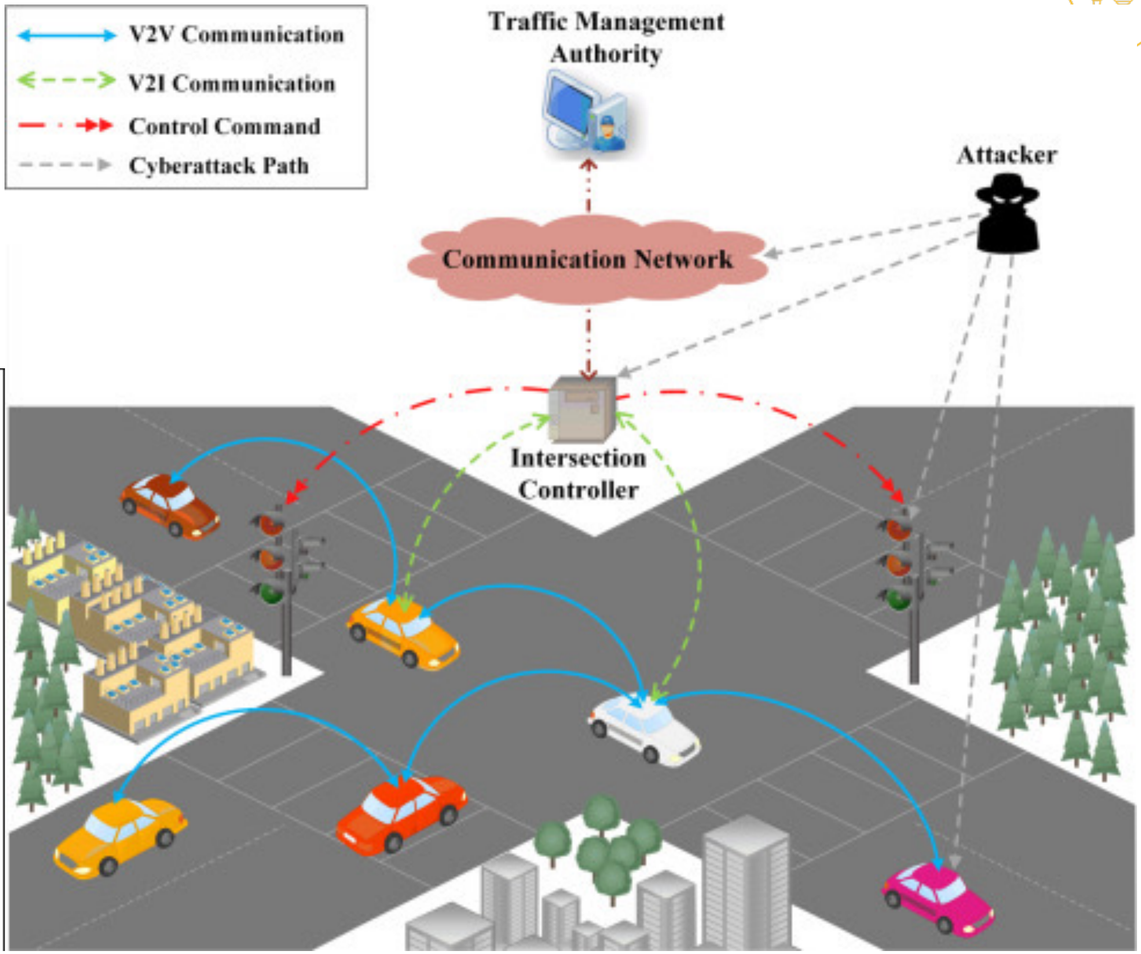
Paul Bogdan

Anand Balakrishnan

Multi-agent systems







[1] Li, Kai, et al. "LCD: Low latency command dissemination for a platoon of vehicles." *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018.
[2] Ma, Yong, et al. "Cooperative communication framework design for the unmanned aerial vehicles-unmanned surface vehicles formation." *Advances in Mechanical Engineering* 10.5 (2018)
[3] Li, Zhiyi, and Mohammad Shahidehpour. "Deployment of cybersecurity for managing traffic efficiency and safety in smart cities." *The Electricity Journal* 30.4 (2017): 52-61.

Multi-agent systems

- Agents
  - assumed to be autonomous
  - report their state (or state of other agents) to the controller
- Controller
  - receives observations from agents
  - gives a sequence of commands to agents
  - can observe some agent actions (e.g. within a bounded area)
- If everyone is **"trustworthy"** all agents ***safely*** achieve objectives

Untrustworthy agents

- Malicious: Agent may consciously desire safety/performance violations
- Faulty: Sensors/Actuators may be malfunctioning
- Uncertain environments: communication may be unreliable, compromised

→ Untrustworthy agents lead to **uncertainty**

Problem Definition

How can agents safely achieve their objectives when some are untrustworthy?

- Security: Focuses on malicious agents or compromised communication
- Verification: Focuses on pessimistic uncertainty of the environment
- Resilience/Fault tolerance: Continued operation in presence of malfunctioning agents

Most approaches have a **pessimistic** view of agents/environment

Fighting pessimism

- Pessimistic assumptions can severely degrade system performance
- Agents may not always be good or bad
  - E.g. faults could be periodic, transient, unpredictable
- Agents may not be equal
  - E.g. some agents may use reliable hardware, could be pre-certified

- What is a general framework to reason about ***trustworthiness***?
- What is ***trustworthiness***?

What is trustworthiness?

- Trustworthy computing[1]
  - Reliability
  - Safety
  - Security
  - Privacy
  - Availability
  - Usability

Trustworthy AI[1]

- Accuracy
- Robustness
- Fairness
- Accountability
- Transparency
- Interpretability/Explainability
- Ethics-aware

[1] Jeanette Wing, Trustworthy AI, https://www.youtube.com/watch?v=WQ6ILBYeKeE

What is it indeed?

- Principles of trust in multi-agent systems[1]
  - *Trust is the subjective probability that individual A expects individual B to perform an action on which B's welfare depends*
  - Trust defined in terms of ability to delegate; deepest trust: no need to monitor
  - Interesting epistemic logic, difficult to apply to multi-agent control
- Trust Quantification for Networked CPS[2]
  - Trustworthiness qualitatively measured in terms of perceptions of ability, benevolence, and integrity
- Trust-based route planning[3]
  - Trustworthiness measured by humans rating controller performance (common in HRI world)

[1] C. Castelfranchi, R. Falcone, R. Principles of trust for MAS: Cognitive anatomy, social importance, and quantification. In *Proceedings International Conference on Multi Agent Systems*, 1998.
[2] Y. Wang, Trust quantification for networked cyber-physical systems." *IEEE Internet of Things Journal*
[3] S. Sheng, et al. Trust-based route planning for automated vehicles. ICCPS 2021.

Talk Overview

- Humanistic Trust
- Review of Dempster Shafer Theory and Subjective Logic
- Trust-aware Control Paradigm
- Connection to hyper-properties

Proposition: A new definition of Trust[1]

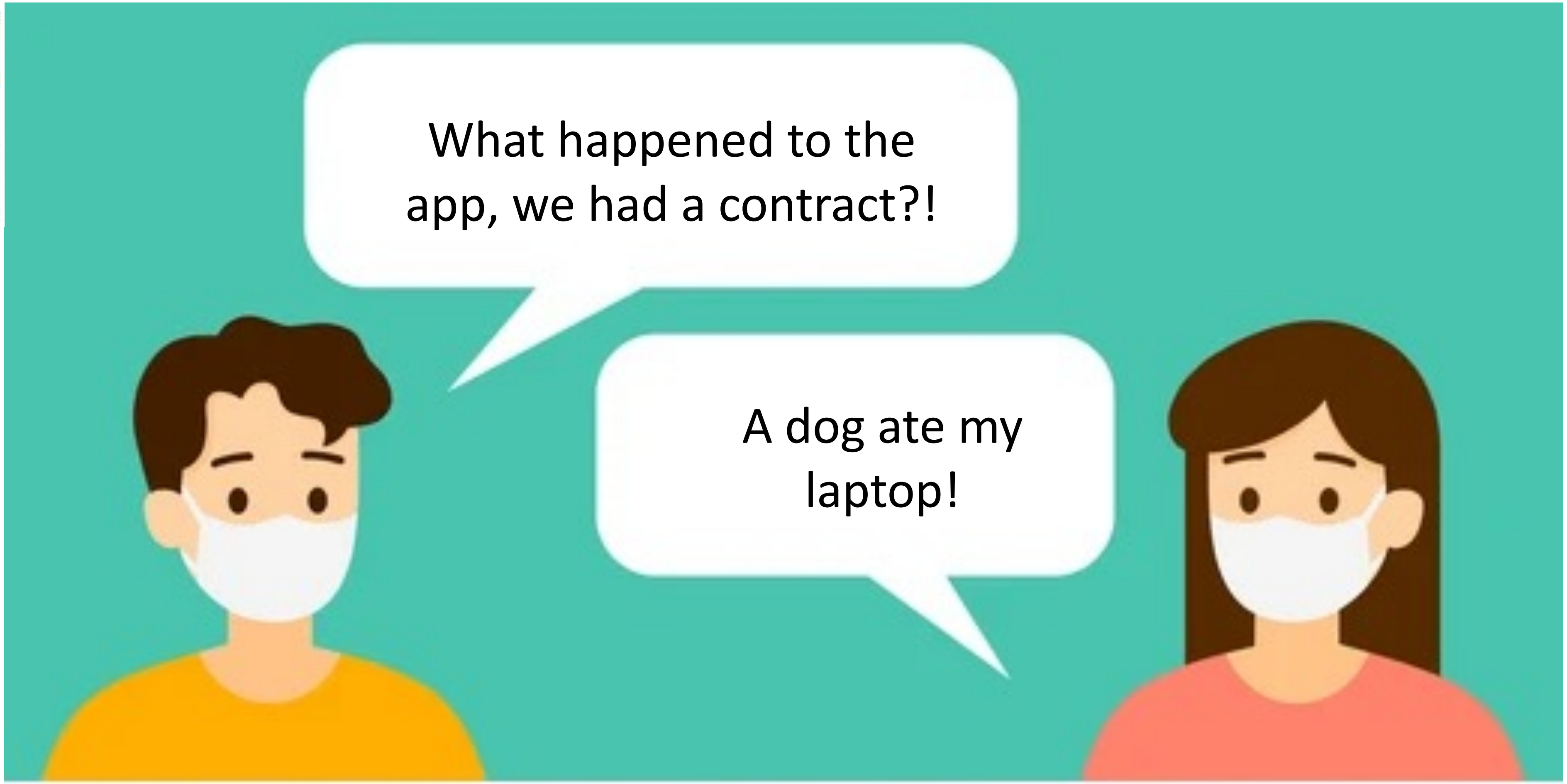
[1] M. Cheng, S. Nazarian, and P. Bogdan. There is hope after all: quantifying opinion and trustworthiness in neural networks. *Frontiers in artificial intelligence* 3 (2020): 54.

How do humans trust?

Scenario 1

Scenario 2


NOVEMBER
1


What happened to the app, we had a contract?!
A dog ate my laptop!

Trust : "they did what I asked them to"

Scenario 1:

Scenario 2:

Uncertainty about the truth

Alice:
Hey can you help me practice this hyperproperties talk ?

Bob: Sorry, I have tons of homework

Scenario 1:

Alice:
Hey can you help me practice this hyperproperties talk ?

Bob: Sorry, I have tons of homework

***Narrator**: No, Bob did not have homework. He had tickets to Lion King.*

The next day:

Alice:
Hey Priya, how was Lion King? You went to see it right?

Priya: Yeah, it was great. Bob is such a softy. He cried when the lion died.

Alice:
Wait, Bob was with you?

Priya: Yeah, didn't he tell you? We had such a good time.

Scenario 2:

Alice:
Hey can you help me practice this hyperproperties talk ?

Bob: Sorry, I have tons of homework

***Narrator**: Bob did have homework.*

The next day:

Alice:
Hey Priya, how was Lion King? You went to see it right?

Priya: Yeah, it was great. Too bad you and Bob missed it.

Alice:
Yeah, Bob had homework, and I had this talk

Priya: Yeah, I know. I saw Bob walking home from the lab at 2am.

Trust : a trusted observer confirmed their actions

Scenario 1:

Scenario 2:

Humanistic trust models → multi-agent framework?

- Agents
  - assumed to be autonomous
  - report their state to the controller (or report the state of other agents)

Could be fake …

- Controller
  - receives observations from agents
  - gives a sequence of commands to agents
  - can observe some agent actions (e.g. within a bounded area)
- If everyone is “**trustworthy**” all agents ***safely*** achieve objectives

Update trust in agent

Reasoning about uncertainty

- Aleatoric uncertainty
  - Arising from inherent stochasticity in the system
  - Variability/Objective uncertainty
- Epistemic uncertainty
  - Arising from lack of knowledge
  - Subjective uncertainty/Ignorance
- Probability (Bayesian) theory traditionally used for both kinds of uncertainty

Dempster Shafer Theory (DST)[1]

- DST: instead of assigning probability to events, assign it to sets of events
- Each fact has a degree of support between 0 and 1
  - 0 : no support for the fact
  - 1 : full support for the fact
- Belief in truth of a proposition p and its negation $\neg p$ may not sum to 1
- Both belief values can be 0 : no evidence for p or $\neg p$
- Given a set of conclusions $\Theta = \{\theta_1, \ldots, \theta_k\}$,
  - DST assigns belief mass m to each subset A of 2^Θ

[1] G. Shafer, "Dempster-shafer theory." *Encyclopedia of artificial intelligence* 1 (1992): 330-331.

DST axioms

Set of conclusions $\Theta = \{\theta_1, \ldots, \theta_k\}$

$$A \subseteq 2^\Theta$$

$$\sum_{A \in 2^\Theta} m(A) = 1$$

$$m(\emptyset) = 0$$

if A $\neq \emptyset$: $m(A) \in (0,1]$

- $belief(A) = \sum_{B \subseteq A} m(B)$
- $plausibility(A) = \sum_{B \in 2^\Theta, B \cap A \neq \emptyset} m(B)$
- $disbelief(A) = belief(\neg A) = 1 - plausibility(A)$
- $prob(A) \in [belief(A), plausibility(A)]$
  - Small Interval = more certainty about belief
- DST: probability of any event is a function of both evidence and uncertainty

Subjective Logic[1]: an epistemic logic

- Alternative to Dempster-Shafer Theory
- DST: belief mass function of evidence and uncertainty
- SL: belief mass function of evidence, uncertainty, **and *a priori probability* in absence of evidence**
- DST & SL allow fusing beliefs (combining evidence) from different sources

[1] Audun Jøsang, *Subjective logic*. Cham: Springer, 2016.

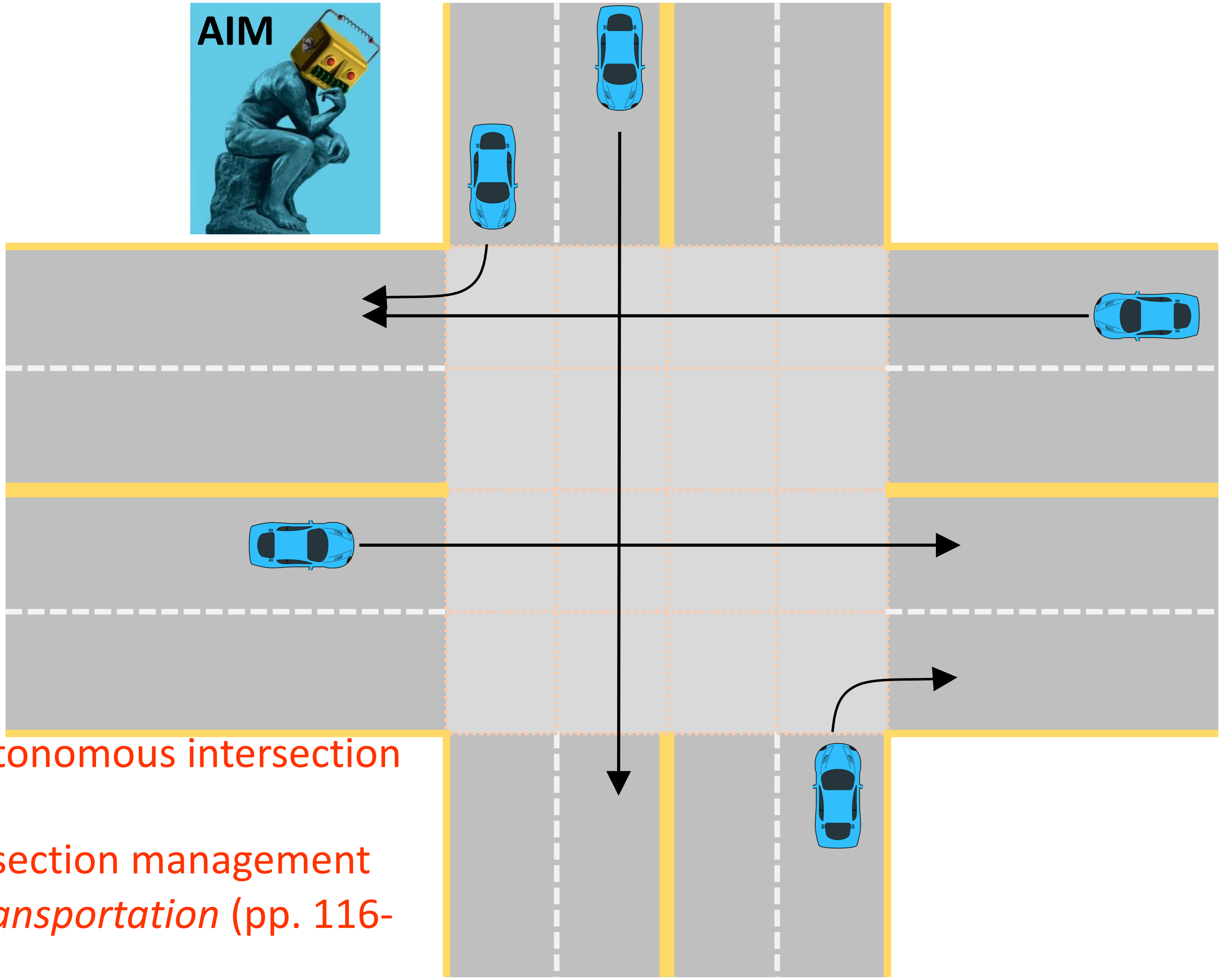
Formalizing SL

- Alice's opinion about Bob – denoted $W_A(B)$
  - $W_A(B) = \left(b_A(B), d_A(B), u_A(B), a_A(B)\right)$
  - $b_A(B)$: belief of A in B
  - $d_A(B)$: disbelief of A in B
  - $u_A(B)$: uncertainty of A in B
  - $a_A(B)$: base rate (prior belief) of A in B
- $b_A(B) + d_A(B) + u_A(B) = 1$
- b, d, u can be viewed as probabilities
- **Trustworthiness** = $b_A(B) + u_A(B) \times a_A(B)$

Evidence → Probabilities

- Evidence obtained when A observes B's behavior
- Assume A has a property φ that is true if B "behaves", and false otherwise
- Let x be a behavior of B, and let X be a set of behaviors of B
- $p = |x \in X \wedge x \vDash \varphi|$: number of behaviors satisfying φ
- $n = |x \in X \wedge x \nvDash \varphi|$: number of behaviors not satisfying φ
- $b_A(B) = \frac{p}{p+n+w}$ $\quad d_A(B) = \frac{n}{p+n+w}$ $\quad u_A(B) = \frac{w}{p+n+w}$
- w: some non-informative prior weight w (depedends on $a_A(B)$)

Autonomous Intersection Management[1,2]

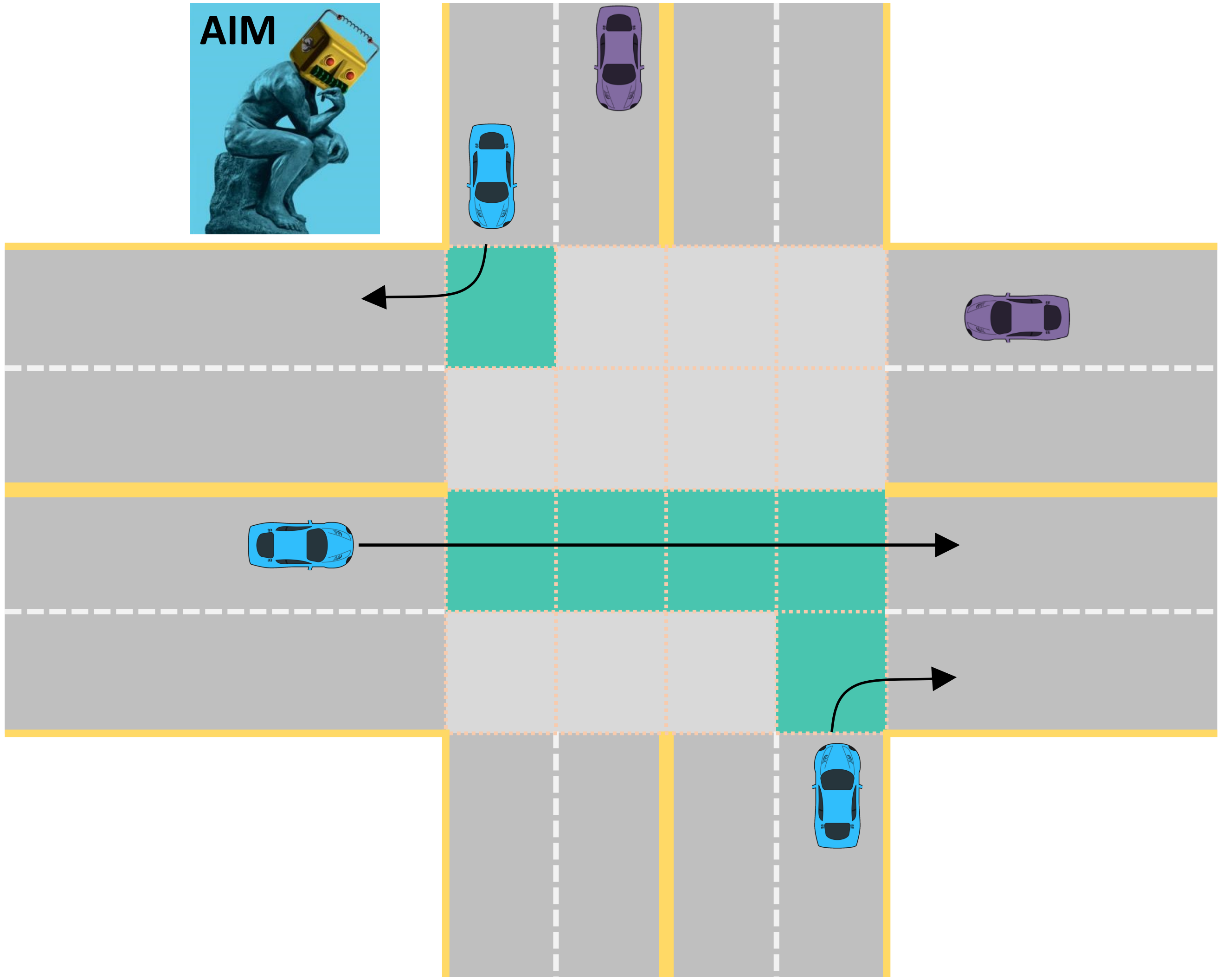


- Cars request use of intersection
- AIM simulates potential car trajectories
- Picks a schedule that does not lead to collisions

[1] K. Dresner, P. Stone, (2008). A multiagent approach to autonomous intersection management. *Journal of artificial intelligence research.*
[2] Au, T. C., Zhang, S., & Stone, P. (2015). Autonomous intersection management for semi-autonomous vehicles. In *Routledge Handbook of Transportation* (pp. 116-132). Routledge

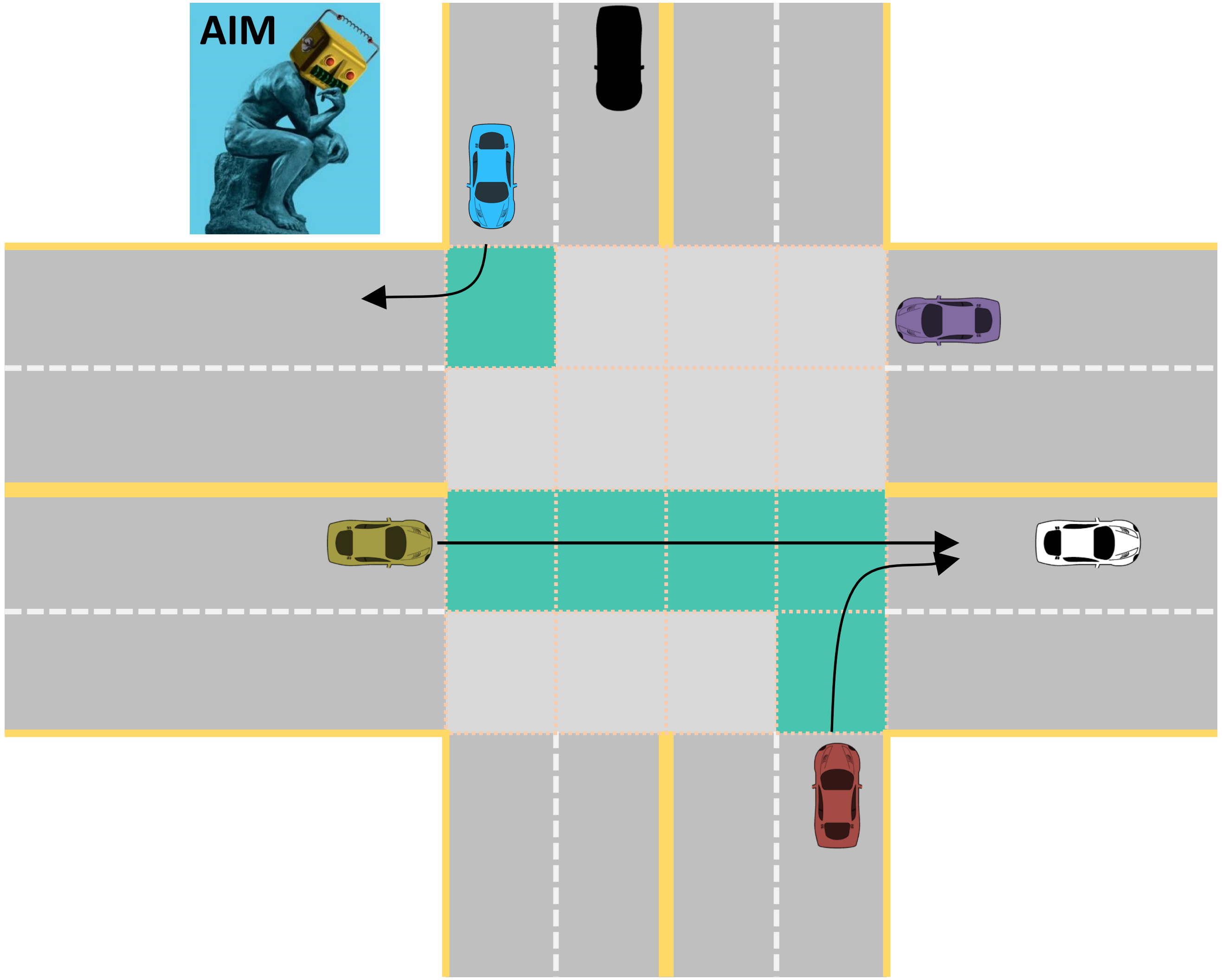
Autonomous Intersection Management

- Green cells in the intersection define “space-time” buffer for cars to use
- Purple cars have to wait their turn
- Assumes that trustworthy cars:
  - strictly follow commands (occupy specified buffer)
  - report location correctly



Autonomous Intersection Management

- What if cars are untrustworthy?
  - What if the red car turns into the wrong lane?
    - Collision!
  - What if the yellow car is actually at the white position
    - Possible Collision!
    - Possible inefficiency (blocked space-time buffer)



Equipping decision-making with trust

- Just like a *credit score* : define a car/driver’s trust score
- Trustworthiness stored in the cloud
- When new evidence is obtained trust is updated

Main idea: Intelligent traffic manager’s decision-making influenced by the agent’s trustworthiness

Trustworthiness scores & Evidence Gathering

- Agent's observed actions and reported sensor data used to compute trustworthiness
- Trustworthiness notions inspired by human interactions
  - does not follow safety instructions from controller → less trustworthy
  - reports false data → less trustworthy
  - reported as less trustworthy by trusted humans → less trustworthy
- Evidence gathering
  - Problem: Evidences can be obtained directly or indirectly
  - Solution: Fusion operators

Trust updates through fusion operators

- Cumulative Fusion operator $\oplus$ computes long-term opinion of A by combining:
  - short-term opinion about B (obtained by observing B)
  - long-term opinion about B
- Discounting operator $\otimes$ computes short-term opinion of A by combining:
  - $P's$ short-term opinion about B (obtained by P observing B)
  - $A's$ opinion about P
- Average fusion operator $\underline{\oplus}$ computes short-term opinion of A by combining:
  - $A's$ short-term opinion about B (obtained by both A and P observing B)
  - $P's$ short-term opnion about B (obtained by both A and P observing B)

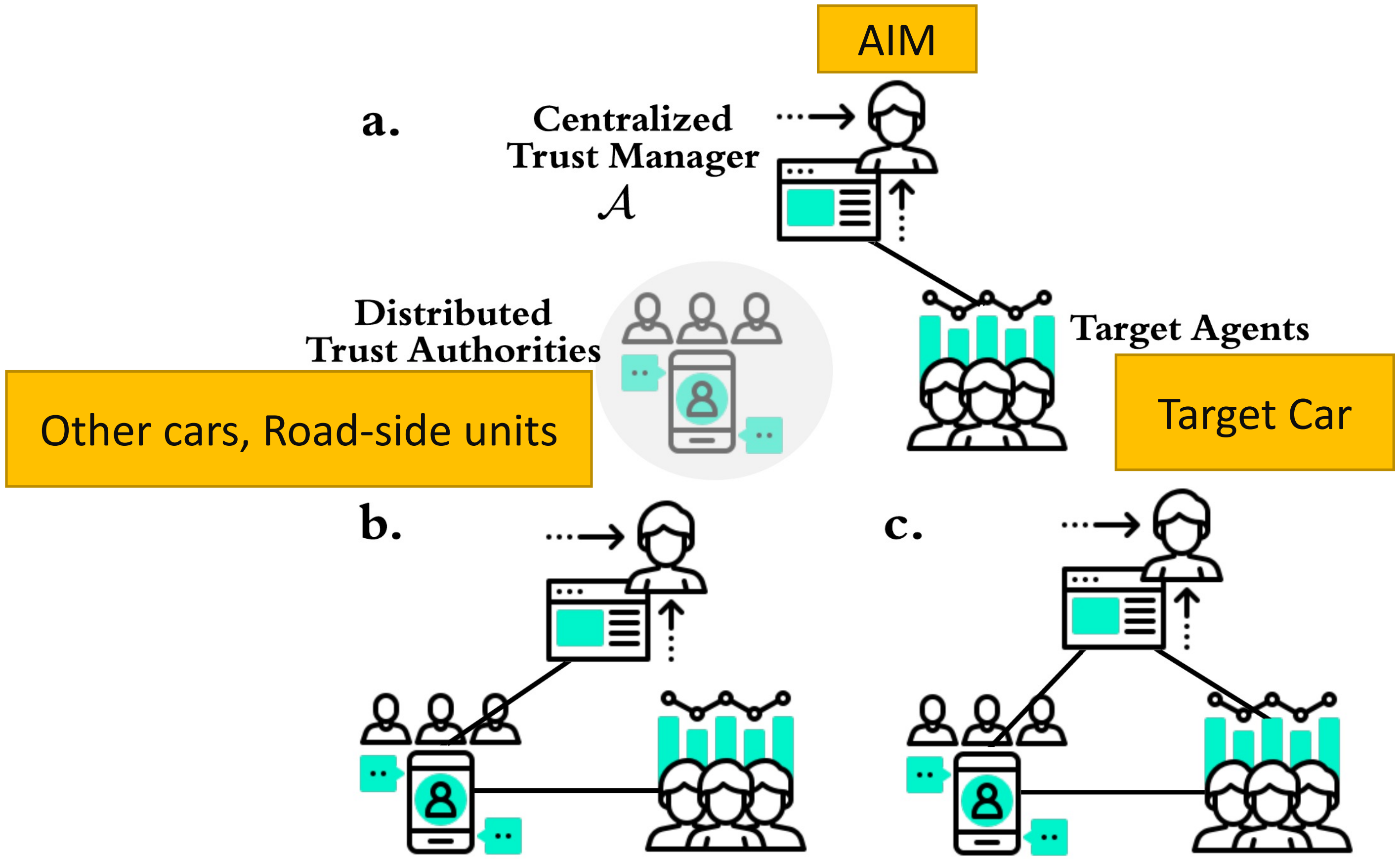
(Details in [1])

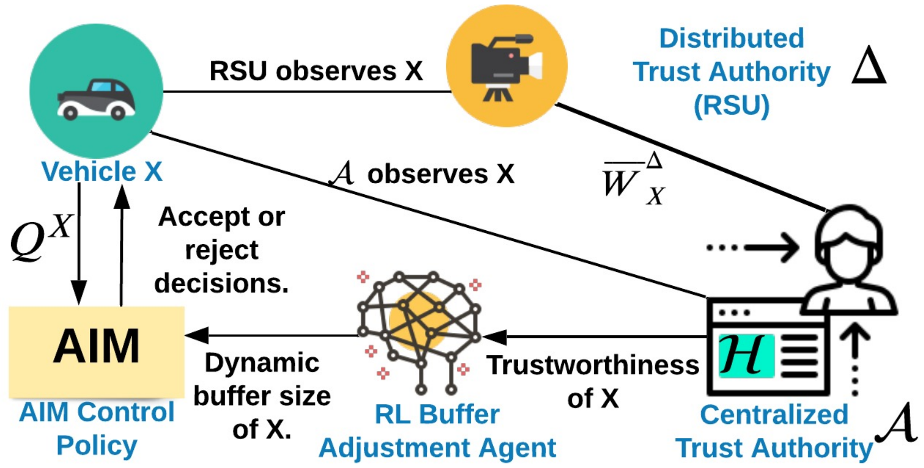
[1] M. Cheng, C. Yin, J. Zhang, S. Nazarian, J. Deshmukh, P. Bogdan, A General Trust Framework for Multi-Agent Systems. In Proc. of *AAMAS 2021*.

Fusion operator scenarios

- Cumulative fusion:
  - AIM knows that Bob has good driving history
  - But Bob caused a collision today (or continued driving excellence today)
- Averaging fusion:
  - Priya sees Bob drive very carefully
  - Miko sees Bob drive like a maniac
- Discounting fusion:
  - AIM receives information from Miko that Bob was driving like a maniac
  - AIM does not trust Miko

Envisioning a Trust-based Cloud/Edge Framework

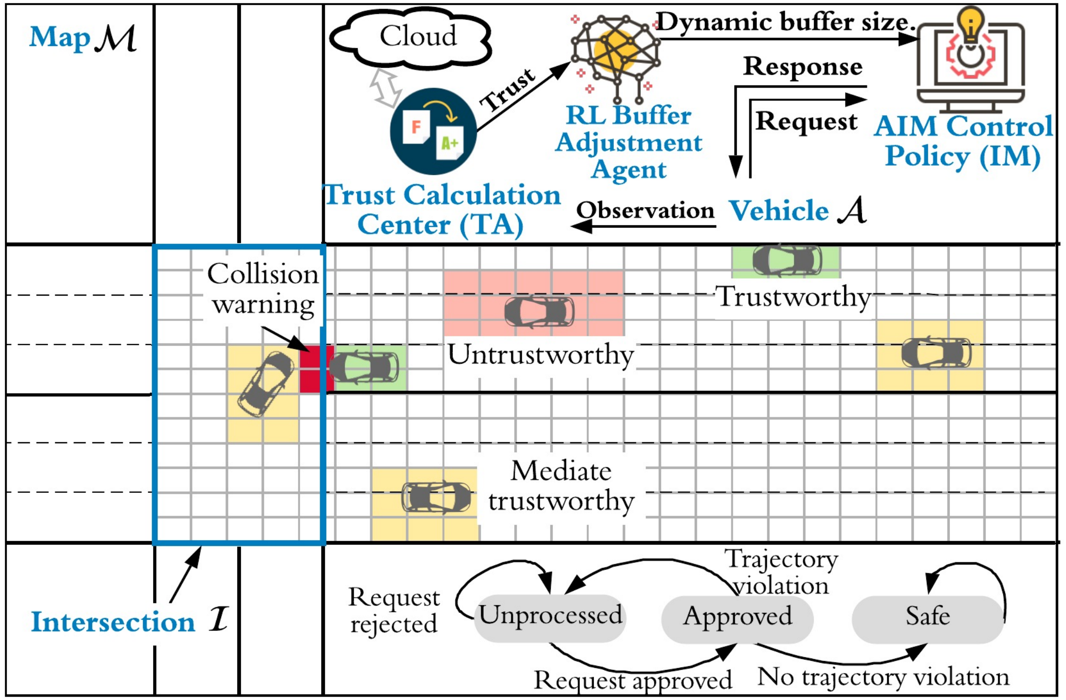




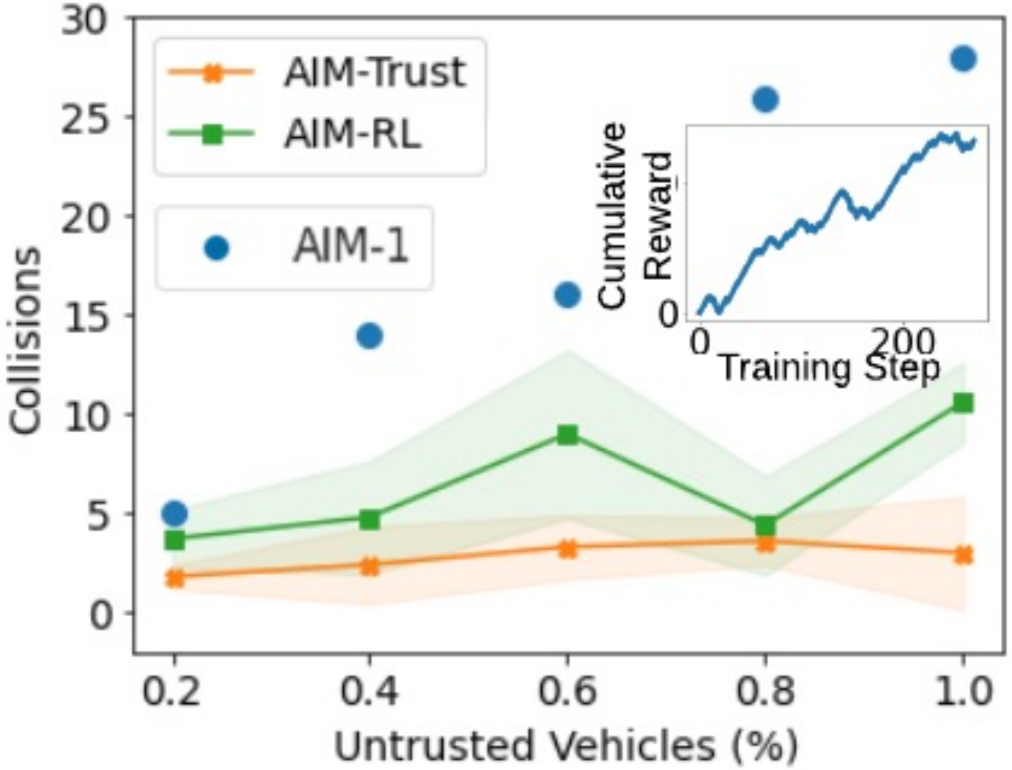
[1] M. Cheng, C. Yin, J. Zhang, S. Nazarian, J. Deshmukh, P. Bogdan, A General Trust Framework for Multi-Agent Systems. In Proc. of *AAMAS 2021*.

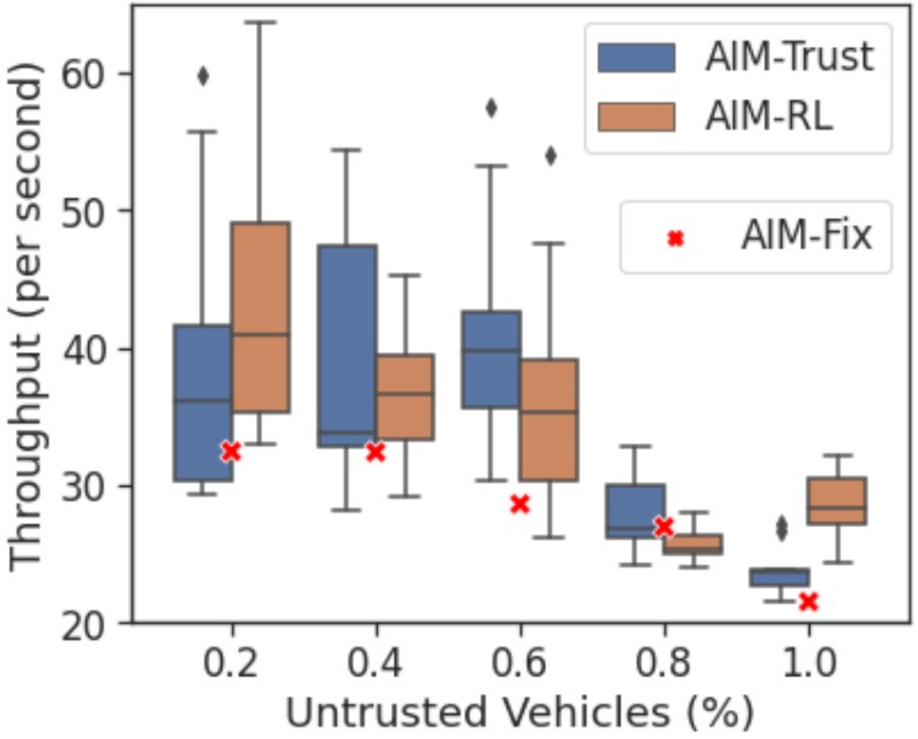
AIM, AIM-RL, AIM-TRUST

- Each vehicle in AIM is assigned space-time reservation buffer
  - Large buffer = high safety
  - Small buffer = high efficiency
- AIM: fixed small buffer
- AIM-RL: utilize reinforcement learning to learn dynamic buffer for each vehicle
- AIM-Trust-RL : dynamic buffer size based on vehicle's trustworthiness



Results



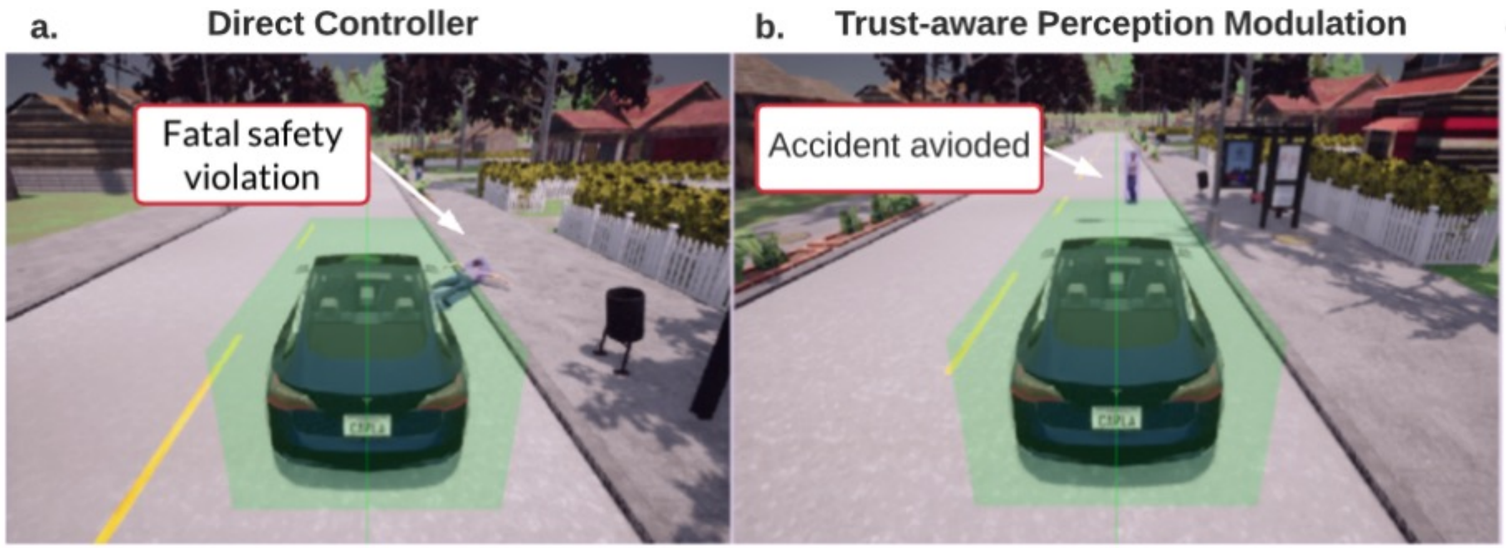


- AIM-Trust has much lower rate of collisions
- AIM-Trust shows favorable throughput compared to traditional AIM

Trust-aware control paradigm

We applied trust-aware control paradigm to:

- Traffic light control[1]
- Autonomous Intersection Management[2]
- Pedestrian avoidance (trusted perception)[3]



[1] M. Cheng, C. Yin, J. Zhang, S. Nazarian, J. Deshmukh, P. Bogdan, A General Trust Framework for Multi-Agent Systems. In Proc. of *AAMAS 2021*.
[2] M. Cheng, J. Zhang, S. Nazarian, J. Deshmukh, P. Bogdan, Trust-aware Control for Intelligent Transportation Systems. In Proc. of IV 2021.
[3] M. Cheng, A. Balakrishnan, J. Deshmukh, P. Bogdan, Dynamic Trust Quantification for Perception, under review.

Trust-aware control paradigm

- Works for any coordination/consensus protocol for a multi-agent system
  - Identify appropriate control variable for each agent
    - AIM: Buffer size, TLC: TL cycle, Pedestrian avoidance: Distance to ped.
  - Modulate agent control inputs according to agent's trustworthiness
  - Update trustworthiness periodically

[1] M. Cheng, C. Yin, J. Zhang, S. Nazarian, J. Deshmukh, P. Bogdan, A General Trust Framework for Multi-Agent Systems. In Proc. of *AAMAS 2021*.
[2] M. Cheng, J. Zhang, S. Nazarian, J. Deshmukh, P. Bogdan, Trust-aware Control for Intelligent Transportation Systems. In Proc. of IV 2021.
[3] M. Cheng, A. Balakrishnan, J. Deshmukh, P. Bogdan, Dynamic Trust Quantification for Perception, under review.

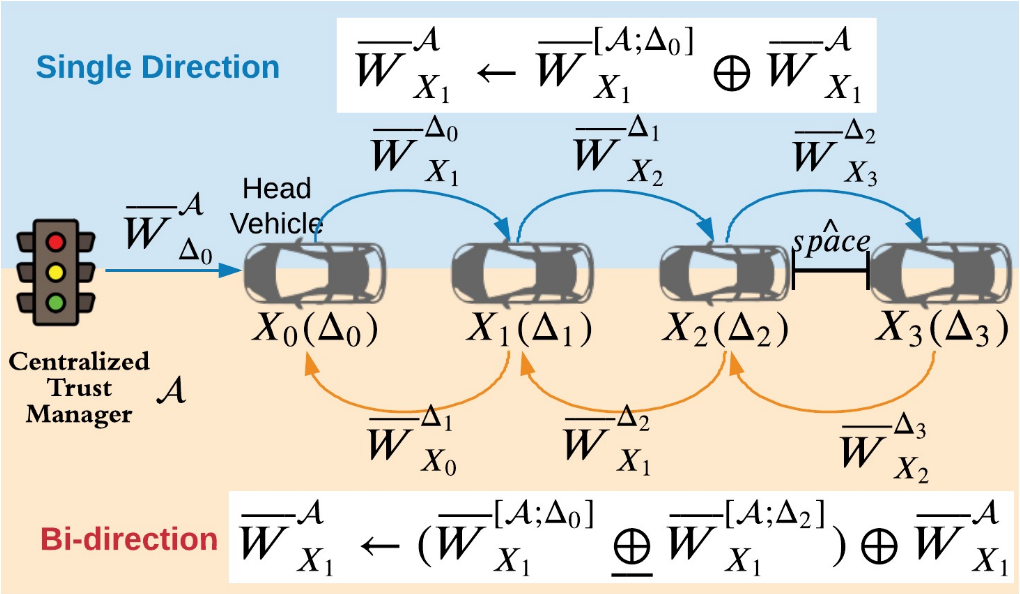
Trust-based attack detection[1]

- CACC platoon:
  - vehicles equipped with V2V/V2X
  - sense surroundings to maintain a constant inter-vehicle space
  - head vehicle controls the platoon
- Attack model (untrustworthy platoon vehicle):
  - jamming attacks
  - V2X data injection
  - sensor manipulation attacks
- Trust-based attack detection: detects acceleration injection attacks

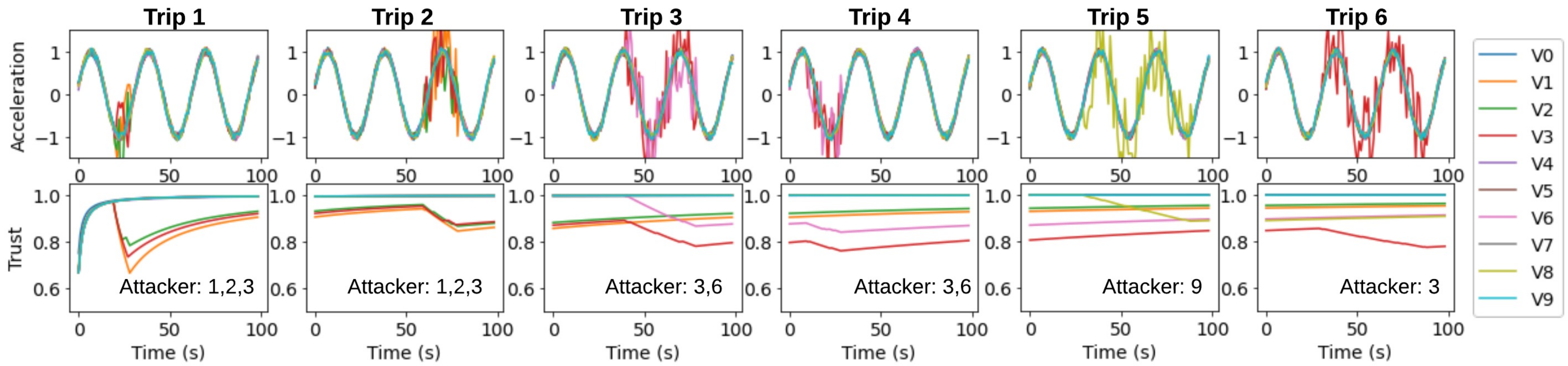
[1] M. Cheng, C. Yin, J. Zhang, S. Nazarian, J. Deshmukh, P. Bogdan, A General Trust Framework for Multi-Agent Systems. In Proc. of *AAMAS 2021.*

Trust-aware CACC

- Centralized trust manager: $\mathcal{A}$
  - Distributed trust vehicles: $X_0, \ldots, X_3$
  - Target vehicles: $X_0, \ldots, X_3$
- Single direction evaluation:
  - Predecessors inspect successors
- Bi-directional evaluation:
  - Predecessors and successors inspect each other
- Positive behaviors:
  - Keep inter-vehicle space
  - Keep speed in desired range
  - No abrupt change in acceleration



Trust-based attack detection

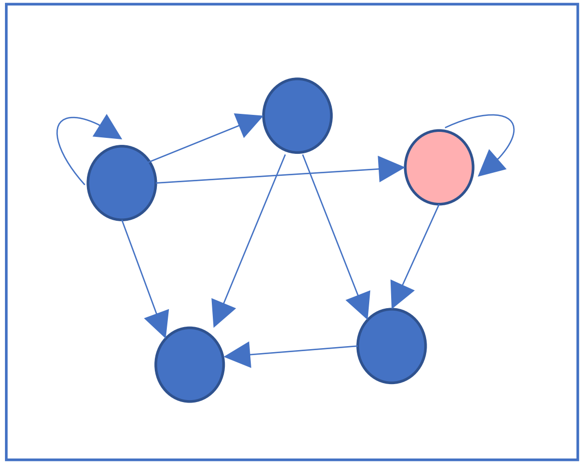


Single-directional attacker detection experimental results. A 10-vehicle platoon completes 6 trips. Assume in the first trip all vehicles are new to the trust system and do not have a trust record. Their records start building from trip 1 and are used in the following trips. The sine waves are required accelerations, and the fuzzy parts are acceleration attacks performed by vehicles.

- Trust framework accurately captures the acceleration attacks in all trips
- Low trust = potential attacker!

Connection to hyper-properties

Agent Model:



- Imagine agent model B that includes faults
- Imagine asking the question:

Are there y more behaviors of B that satisfy φ than those that do not satisfy φ?

OR

Is the probability of B satisfying φ more than the probability of B satisfying $\neg\varphi$?

(or the environment having no information about B)?

Trustworthiness as a hyperproperty

- Trustworthiness can be viewed many ways
- Trustworthiness as a quantitative hyperproperty[1]
  - $W_A(B)$: related to number of positive/negative evidence of B's behavior
  - Good behavior of B: specified by some property φ
  - E.g. φ is a Signal Temporal Logic (STL) property
- Trustworthiness as a HyperPCTL property[2] (or maybe PHL?)
  - Compare probabilities of agent executions (under different models of agent visibility by the environment)

[1] B. Finkbeiner, C. Hahn, and H. Torfah. Model checking quantitative hyperproperties. In *CAV* 2018.
[2] E. Ábrahám, and B. Bonakdarpour. HyperPCTL: A temporal logic for probabilistic hyperproperties. In *QEST 2018*

Attack Detectability is a Hyper-property

- An attack is undetectable if the observed system output is indistinguishable from some valid system behavior
- $y(s, u)$: system output starting in state s with sensor input u
- Attack $u_{\tau'}$ is undetectable (HyperSTL[2]):

$$\exists \tau \exists \tau' (\|s_\tau - s_{\tau'}\| > 0) \wedge \mathrm{Alw} \begin{pmatrix} (u_\tau = 0) \wedge \\ d(y_\tau(s_\tau, u_\tau), y_{\tau'}(s_{\tau'}, u_{\tau'})) < \epsilon \end{pmatrix}$$

- Can we relate attack detectability to trustworthiness of an agent?

[1] Pasqualetti, F., Dörfler, F., & Bullo, F. (2013). Attack detection and identification in cyber-physical systems. *IEEE Trans. on Automatic Control*
[2] Luan Nguyen, et al. Hyperproperties of real-valued signals. MEMOCODE 2017

Open Challenges

- Open Problem 1: Given a white-box agent (or a fault model for the agent), can we *verify* if it is trustworthy? [Model Checking]
- Open Problem 2: Given a white-box model of the controller and the (stochastic) agents, can we design controllers that (probabilistically) guarantee safety/performance? [Synthesis]
- Open Problem 3: Given black-box models of stochastic agents, can we design controllers that give probabilistic guarantees on system safety/performance? [Model-free Synthesis, Statistical Verification[1]]
- Open Problem 4: Can we do runtime monitoring and mitigation in a trust-aware fashion?

[1] Y. Wang, M. Zarei, B. Bonakdarpour, M. Pajic, Statistical verification of hyperproperties for cyber-physical systems. ACM TECS 2019

Thank you for your attention!

Thank our co-authors:

- Junyao Zhang
- Chenzhong Yin
- Shahin Nazarian

Support from: