Threat Detection for Collaborative Adaptive Cruise Control for Connected Cars

Matthew Jagielski*, Nicholas Jones*, Chung-Wei Lin+, Cristina Nita-Rotaru*, Shinichi Shiraishi+

* Northeastern University

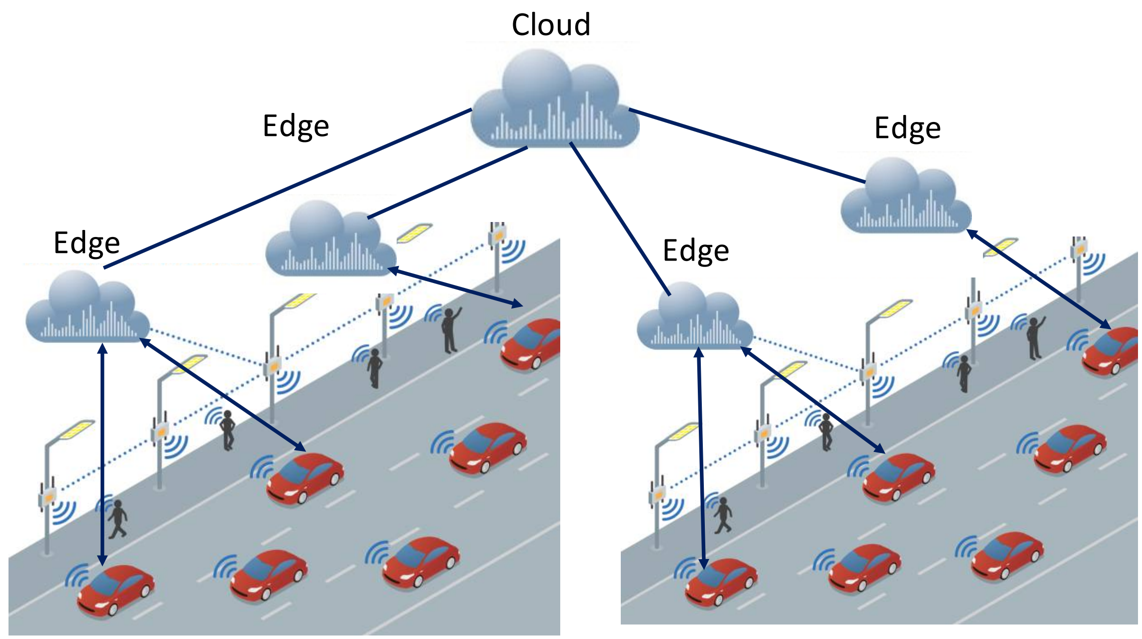
\+ Toyota ITC

Connected Cars Deployment: DSRC

- General Motors:
  - Available in Cadillac CTS sedans since 2017
- Toyota:
  - Toyota and Lexus enabled with DSRC-based V2V communications in Japan since 2015
  - Announced plans to begin deployment of V2V and V2I technology in the U.S. market starting in 2021
- Volkswagen:
  - Announced in 2017 that will have DRSC in Europe beginning in 2019

Safety Applications

- Traffic and congestion control
- Collision avoidance
- Intersection management
- Assisted-turn
- Collaborative adaptive cruise control



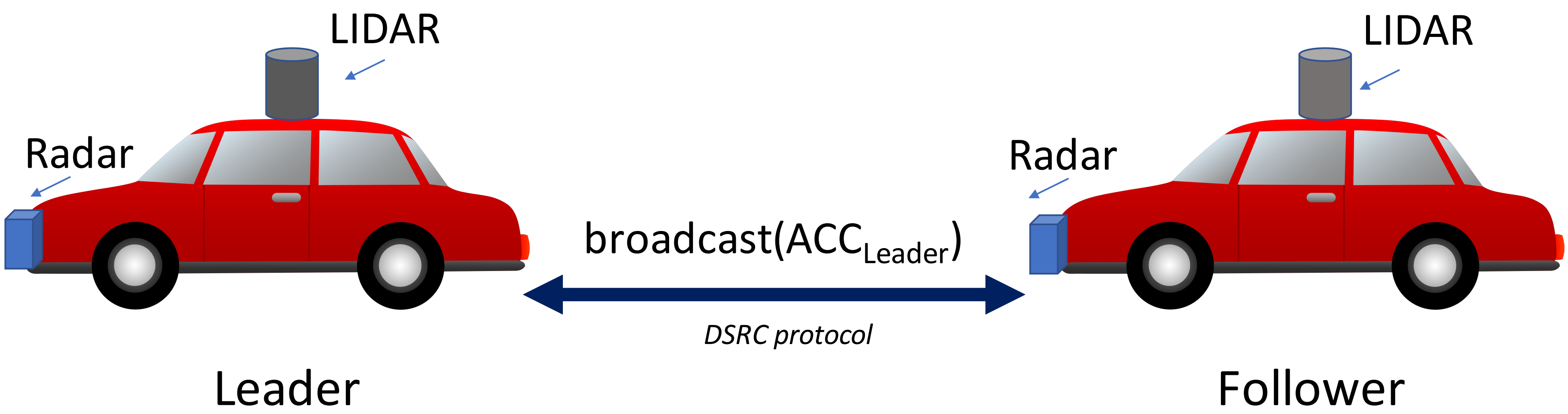
How to ensure that safety applications achieve their goal in an adversarial environment?

This Talk

- Consider collaborative adaptive cruise control for connected cars architectures using DSRC
  - Demonstrate the impact of attacks on safety applications
  - Design mitigation techniques

A group of self-driving cars successfully formed a platoon (July 2017)
https://www.volpe.dot.gov/

Collaborative Adaptive Cruise Control



- Each car:
  - Periodically broadcasts its own acceleration
- Each follower:
  - Uses input:
    - Preceding car acceleration received via network, i.e. DRSC
    - Local sensors for speed and distance of previous car
  - Computes the new acceleration to maintain a safety time gap

CACC

$$g_{safe} = v * 0.1 + \frac{v^2}{2D^{max}} - \frac{{v_p}^2}{2{D_p}^{max}} + 1.0$$

$$a_{t+1} = K_a a_t + K_v(v_p - v) + K_g(g - G_{min} - vT_g)$$

Mani Amoozadeha,Hui Dengb,H.Michael Zhangb Chen-Nee Chuaha,and Dipak Ghosalc. 2015. Platoon Management with Cooperative Adaptive Cruise Control Enabled by VANET. Veh. Commun. 2, 2 (April 2015).

CACC Goals

- Safety:
  - Cars need to maintain a minimum safe time-gap g^t_{safe}
- Efficiency:
  - Platoon of cars should be traveling with as little distance as possible between them
- Passenger comfort:
  - Avoid abrupt changes

$$crash = \max_{T_j} \left\{ 0, \max_i \frac{g^t_{safe} - g^t_i}{g^t_{safe}} \right\}$$

$$waste_i = \int_{t=0}^{t_{end}} \left(g^t_i - g^t_{safe} \right) dt$$

$$jerk = \frac{da}{dt}$$

Attacker Goal and Capabilities

Goal: impact safety, efficiency and passenger comfort by influencing the computation of the new acceleration

- Influence acceleration of car preceding the victim
  - Attacker has compromised the car preceding the victim and sends incorrect acceleration values via DSRC communication
- Influence RADAR and/or LIDAR sensors of the victim.
  - Attacker has control over just the LIDAR, just the RADAR, and over both LIDAR and RADAR
  - Can manipulate data from the victim’s sensors, either directly, by compromising a subset of the victim car, or indirectly, by remotely manipulating the sensor’s physical layer signals

How to Model Attacks

- (ACL) Lying about acceleration
  - Passenger comfort

$$a_{fake} = a_{true} + c_a \sin(ft)$$

- (VEL) Lying about velocity
  - Efficiency

$$v_{fake} = v_{true} - c_v t$$

- (POS) Lying about distance
  - Safety

$$d_{fake} = d_{true} + c_d t$$

- (VEL-POS) Lying about velocity and distance
  - Safety

$$v_{fake} = v_{true} + c_v t$$
$$d_{fake} = d_{true} + c_d t$$

Defenses: Leveraging Invariants

- Cars are physical objects, their behavior in terms of position, velocity, and acceleration must follow certain well defined laws of kinematics
- By using these laws, we can detect inconsistencies between these values as a result of an attack

PHY (ε_p ,ε_v)

$$v_{min}t_d + 0.5a_{min}t_d^2 - \varepsilon_p \leq p_{new} - p_{old}$$

$$p_{new} - p_{old} \leq v_{max}t_d + 0.5\, a_{max}\, t_d^2 + \varepsilon_p$$

$$a_{min}t_d - \varepsilon v \leq v_{new} - v_{old} \leq a_{max}t_d + \varepsilon_v$$

Defenses: Hidden Markov Models

- Use a Hidden Markov Model, an anomaly detection mechanism, to fit the time series data of CACC and learn temporal dependencies

HMM (δ_h)

- a synchronization phase where cars create the safe gaps
- a stable phase, where cars stay at a roughly fixed velocity.

Simulations Setup

- Simulation is discrete, a run is 400 steps, each step is 0.1s
- Platoon of 7 cars, car length is 5m, cars start at 1m/s with a distance between cars of 10m
- Sensor measurement error with Gaussian noise, with standard deviation of 3cm for LIDAR and 0.1m/s for RADAR
- CACC algorithm: minimum safe-gap is 0.55s, with a 2m leeway, resulting in a 2.55m gap (or 7.55m from front to front including car length); Maximum deceleration is $5m/s^2$
- PHY is invoked at each step, and HMM every 50 steps

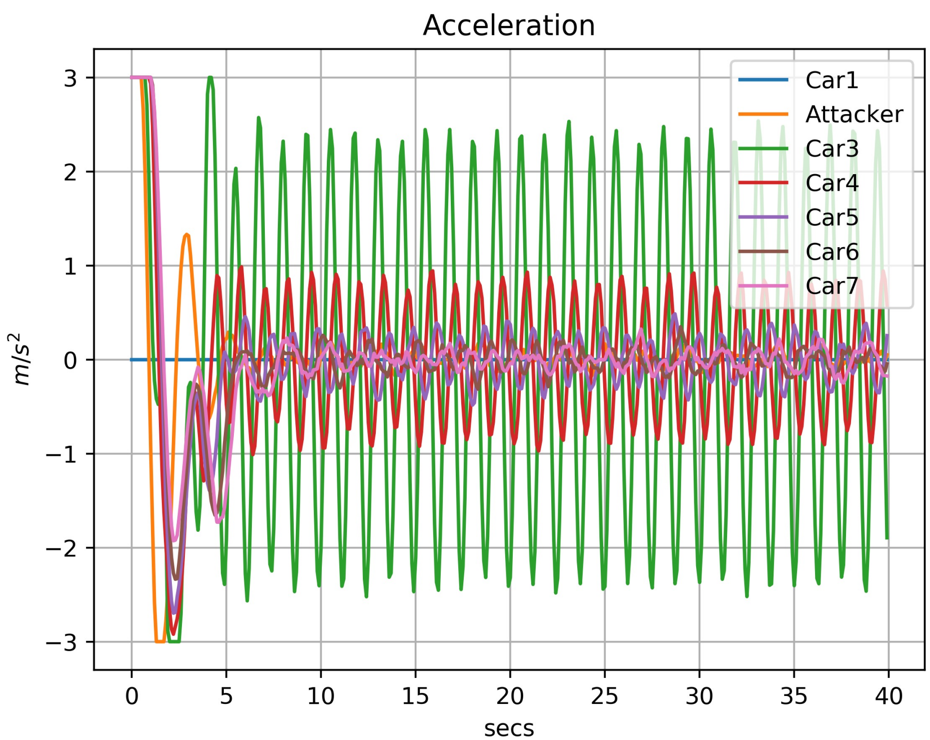
Summary of Attacks

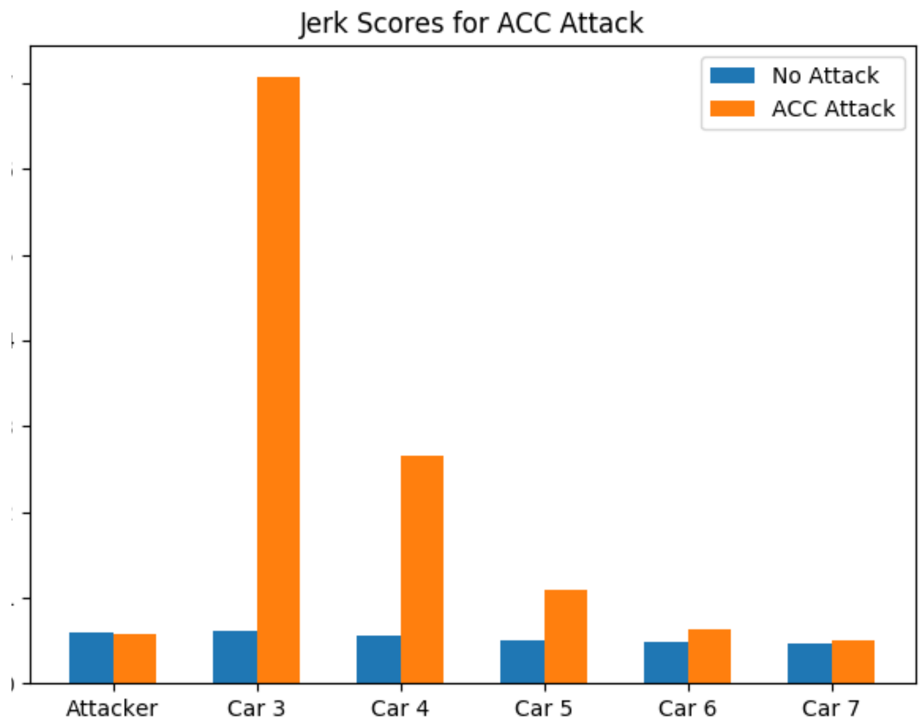
Attack	Jerk	Waste	Crash
No attack	0.56	2.10	0
ACL	7.07	3.14	0
VEL	0.59	9.32	0
POS	0.73	0.69	1 (crash)
VEL-POS	0.86	0.60	1 (crash)

Detection Rate

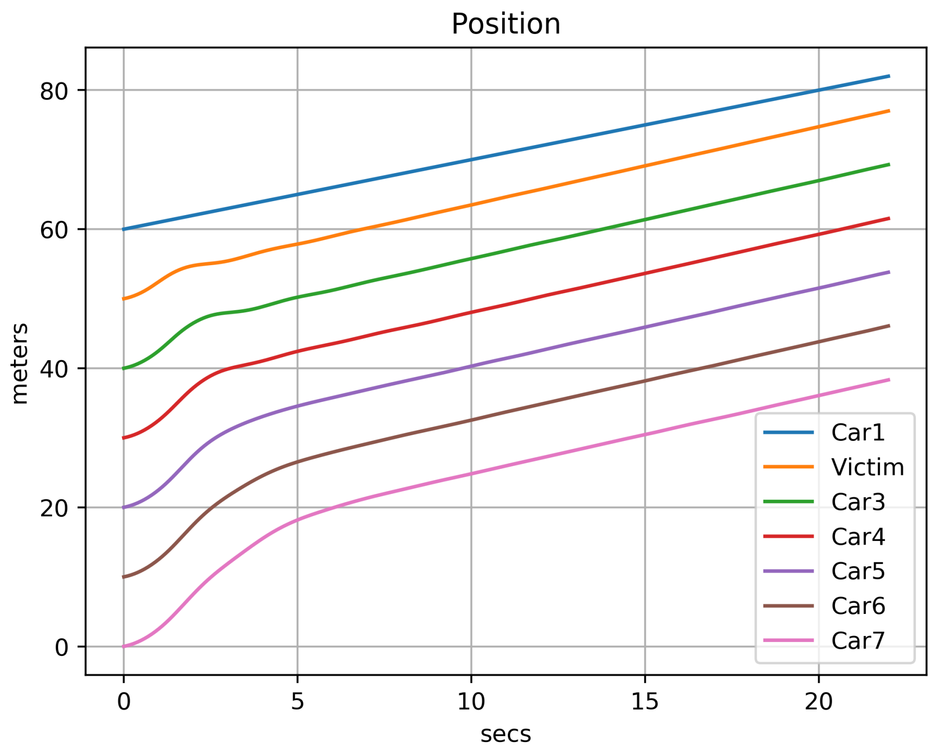
Attack	PHY	HMM
No attacks (false positives)	0.35	1.5
ACL ($c_a = 5$, $f = 5$)	25.75	77.5
VEL ($c_v = 1$)	95.13	83.5
VEL ($c_v = 0.1$)	0.58	79.5
VEL ($c_v = 0.05$)	0.45	79.5
POS ($c_d = 0.1$)	0.25	74.0
VEL-POS ($c_v = 0.2$, $c_d = 0.1$)	0.13	90.0

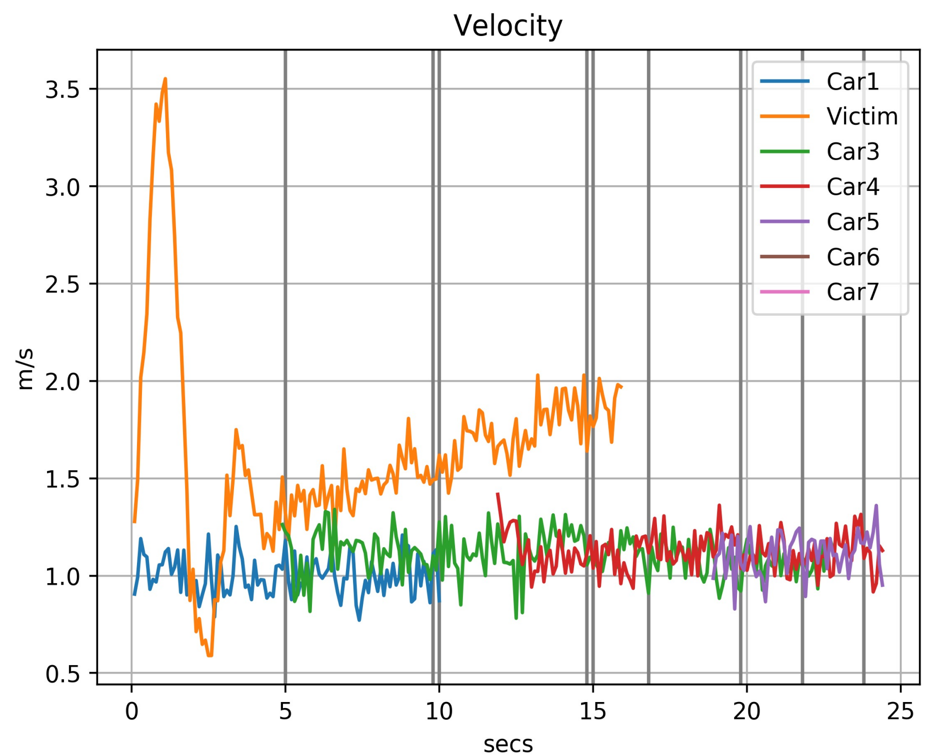
ACL Attack


Acceleration
3
2
1
0
−1
−2
−3
m/s^2
0
5
10
15
20
25
30
35
40
secs
Car1
Attacker
Car3
Car4
Car5
Car6
Car7


Jerk Scores for ACC Attack
No Attack
ACC Attack
Attacker
Car 3
Car 4
Car 5
Car 6
Car 7

VEL-POS Attack Detection





Crash occurs at 21.72 s (distance of 5 m means a crash has occurred)

HMM detects the crash before it occurs !

Conclusion

- One can not have safety without security:
  - We were able to show how attackers can create crashes
- We also showed attacks that impact efficiency and passenger comfort
- Proposed mitigation techniques that were able to detect the attacks before the crash occurred

https://nds2.ccs.neu.edu/