VePMAD: A Vehicular Platoon Management Anomaly Detection System

A Case Study of Car-following Mode, Middle Join and Exit Maneuvers

Weaam Bayaa

Examiner: Panagiotis Papadimitratos
Supervisor: Mohammad Khodaei
NSS Group

Sep 13, 2021

Platooning

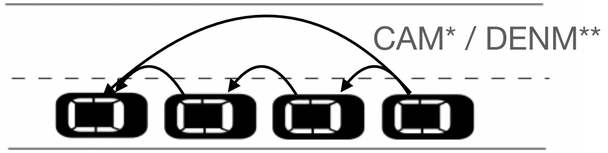
1. Communication between vehicles and infrastructure enhanced the capabilities of vehicles and enabled the rise of Cooperative Intelligent Transport Systems (C-ITS). New applications are introduced such as CACC and Platooning

2. Platoon-enabled vehicles: have all required hardware, software and subscriptions in platooning service. If it is a member of platoon and can be either leader or follower.

3. "PlatooningContainer" is added to CAM to carry information about vehicle and its ability of platooning.

4. Platoon modes of operation: Car-following mode and maneuver mode. In maneuver mode, Platoon Management Protocol (PMP) and its operations will allow platoon maneuvers (e.g., middle join and exit)

5. Benefits of platooning: increase road capacity and reduction of energy consumption and exhaust emissions.




```
CamParameters ::= SEQUENCE {
    basicContainer BasicContainer,
    highFrequencyContainer HighFrequencyContainer,
    lowFrequencyContainer LowFrequencyContainer OPTIONAL,
    specialVehicleContainer SpecialVehicleContainer OPTIONAL,
    ...,
    platooningContainer    PlatooningContainer OPTIONAL,
    ...
}
```

Example of placement of a specific platooning container in CAM [1]

*CAM (Cooperative Awareness Messages)
**DENM (Decentralized Environmental Notification Message)

Motivation

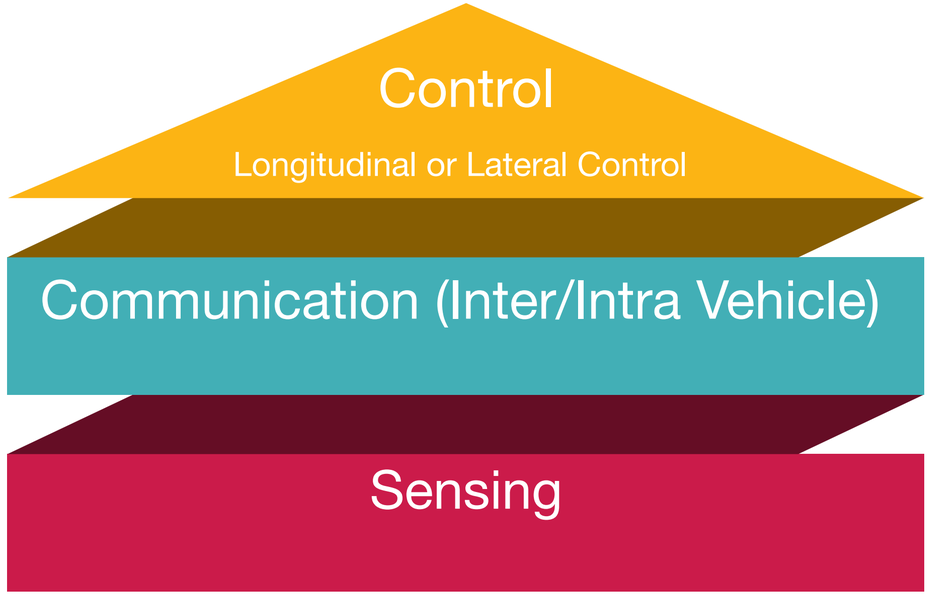
1. Communication is more effective than distance sensors in terms of platoon safety.

2. Platooning benefits cannot be achieved if the system is vulnerable to invalid and malicious behavior.

3. Proactive mechanisms: restrict access to Vehicular Communication (VC) network (protect the system from external attackers through VPKI).

4. Reactive mechanisms “Misbehavior Detection Scheme (MDS)”: detect attacks launched by authenticated vehicles (internal attackers).

5. Many proposed MDSs in literature assumes that vehicles in platoon are driving in a single straight lane. Considering all possible attack scenarios (i.e., during maneuver) is essential for large deployment of platooning



Vulnerabilities in three layer framework

Research Questions

1. By considering PMP, what aspects need to be considered while designing MDS deployed in each vehicle?
2. How to design an MDS that can take into consideration different maneuvers in PMP?
3. Can MDS reused for different controllers or a new MDS is needed for each controller?
4. How to design the MDS in a way that enables it to recognize the undergoing maneuver and intelligently detect deviations from expected behavior?

Contribution

1. A state-of-the-art MDS that can take into consideration dynamic platoon environment is introduced. Unlike existing MDSs, proposed framework has the ability to discern misbehavior from maneuver operation and accordingly, avoid treating a maneuver as a misbehavior.

2. Leveraging machine learning algorithms, the MDS has the ability to recognize the mode of the operation of the platoon (car-following or maneuver mode) at early stage and predict the misbehavior.

3. The whole framework is extendable in terms of maneuvers and controllers. Only platoon controllers are considered (e.g., PATH and Flatbed) but the framework can be extended to include Cooperative Adaptive Cruise Control (CACC) controllers (e.g., Consensus and Ploeg).

Adversary Model

1. Internal attacker is assumed which means it can communicate with platoon members. Both vehicles and RSUs are vulnerable to be attacked

2. By compromising the vehicle, it's assumed that the attacker has full control of inputs of vehicles and has the ability to inject falsified beacons such as manipulating speed, acceleration and position of compromised vehicle.

3. Attacker can compromise either platoon leader or platoon member.

4. Standardized security mechanisms are assumed to be in place: pseudonymous certificates to protect driver's privacy and storage of corresponding private keys in a Hardware Security Module (HSM) to protect it against tampering. VPKI is available to limit validity of of pseudonyms (countermeasure for Sybil attack)

Methodology (Hidden Markov Model)

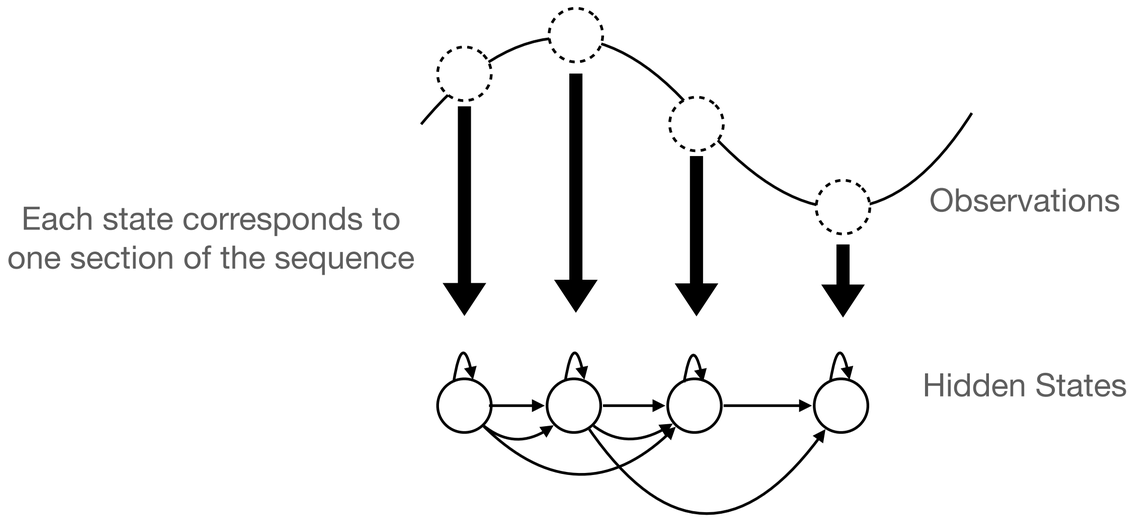
Introduction

1. Hidden Markov Model (HMM) has been widely used to model driving behavior due to its powerful ability to describe dynamic processes.

2. Two stochastic processes: an observable process which represents the sequence of observations of the system and hidden process which can be indirectly inferred by analyzing the the sequence of observations.

3. In the context of driver intention recognition, Example of Discrete HMM (DHMM): classify the current driver as sporty or defensive. Example of Continuous HMM (CHMM) is advanced driver assistant systems

4. Main requirement for the MDS is to recognize the platoon behavior at early stage and not to wait till the behavior is completed.



Methodology (Hidden Markov Model)

Fundamental Concepts

Observed time series $O = o_1, o_2, \ldots o_T$

N possible states $S = s_1, s_2, \ldots s_N$

Hidden sequence $z = z_1, z_2, \ldots z_T$

HMM model is represented in compact form as: $\lambda = (\pi, A, B)$

Initial Probabilities (π)

$$\pi_i = P(z_1 = s_i)$$

Transition Probabilities (A)

$$a_{i,j} = P(z_{t+1} = s_i \mid z_t = s_j)$$

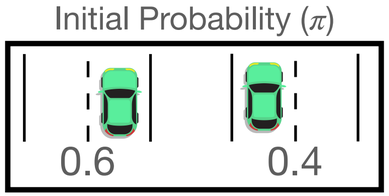
Emission Probabilities (B)

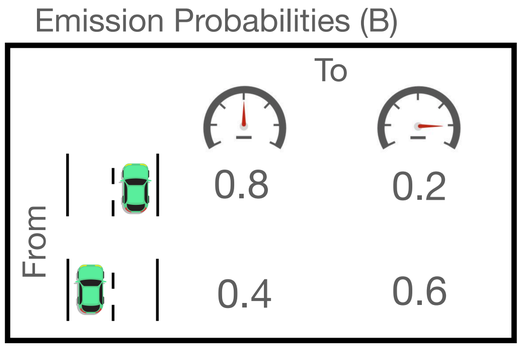
If discrete (Observations can belong to a codebook $V = (v_1, \ldots v_k)$)

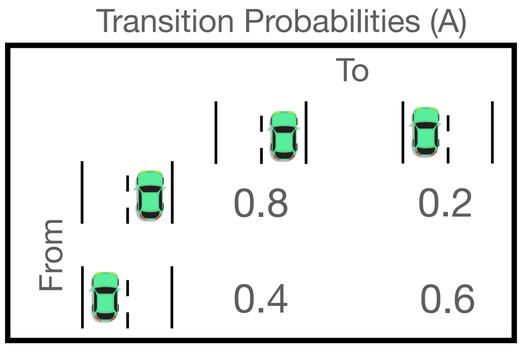
$$b_i(o_t) = P(o_t = v_k \mid z_t = s_i)$$

If continuous (Observations follow a specific distribution i.e., Gaussian or mixture of Gaussians)

$$b_i(o_t) = \sum C_{i,m} S(o_t) \mid \mu_{i,m}, \Sigma_{i,m}$$



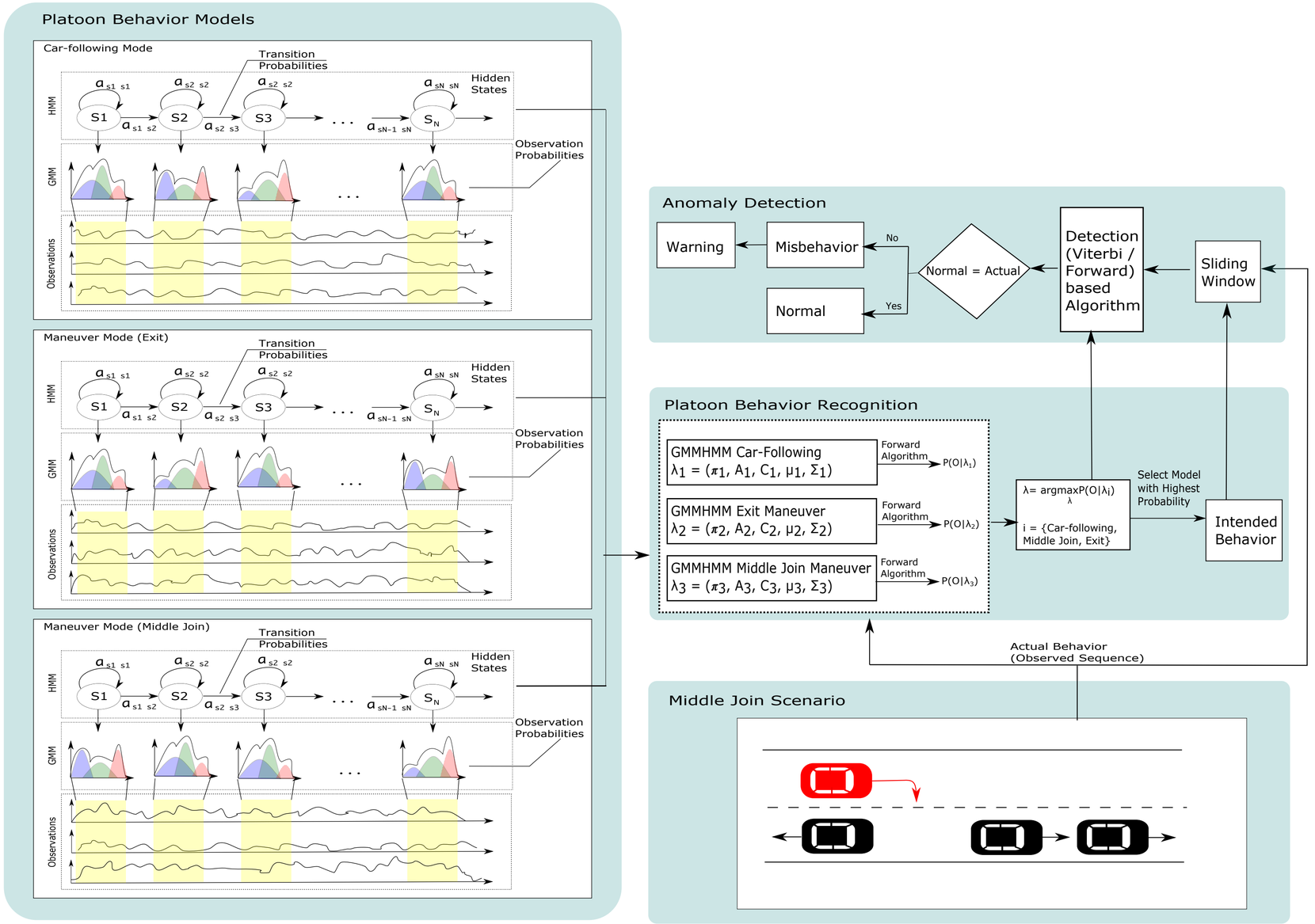




GMMHMM is HMM with GMM emission probability distribution B which means that observation probabilities of HMM states are modeled with GMMs

HMM model is represented in compact form as:
$\lambda = (\pi, A, C, \mu, \Sigma)$

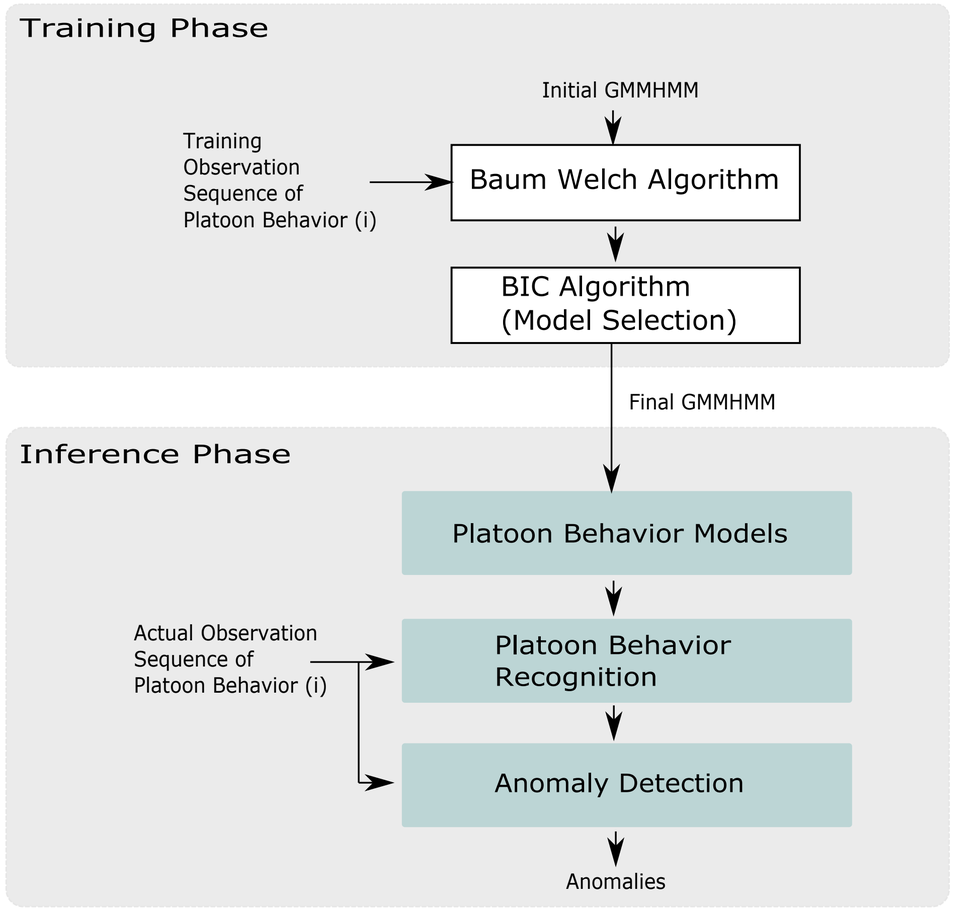
Proposed Framework



Training and Inference Phases

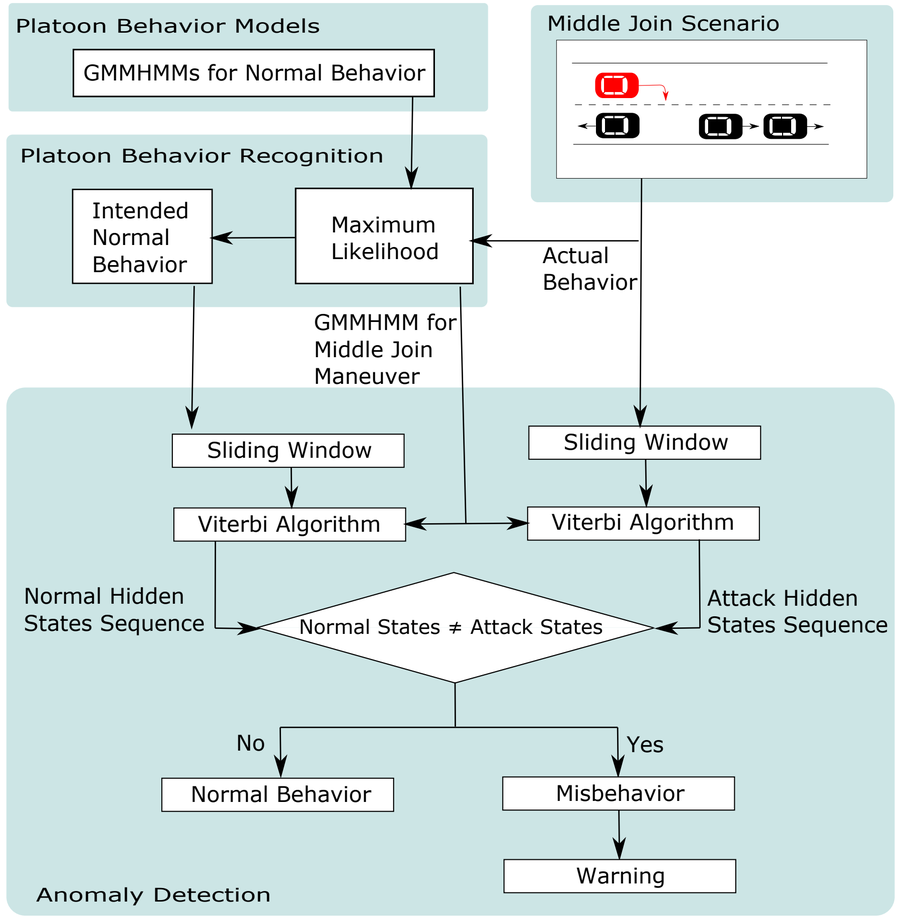
1. Baum Welch algorithm used to optimize GMMHMM model using training data. BIC criterion to determine number of components -> Generate GMMHMM models that models nominal behavior

2. Determine which GMMHMM model to use based on observed sequence and use it for anomaly detection. Sliding window of 1 second is used to recognize the maneuver and detect anomalies as early as possible

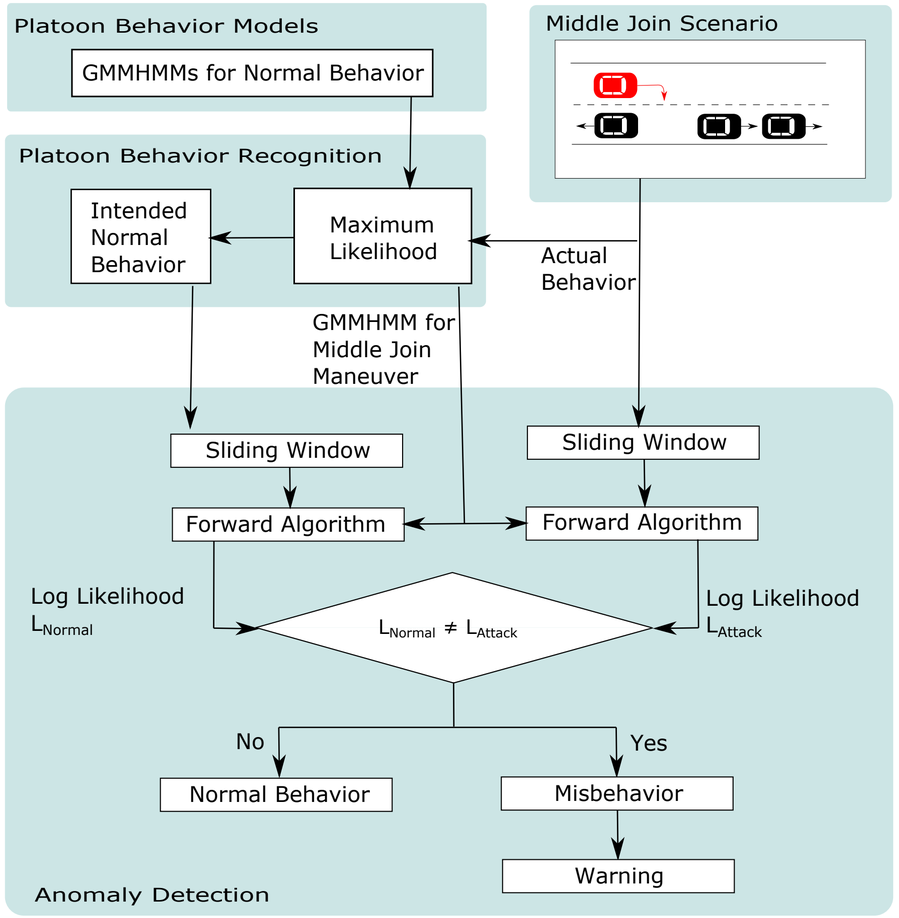


Anomaly Detection Approaches

Viterbi-based Anomaly Detection

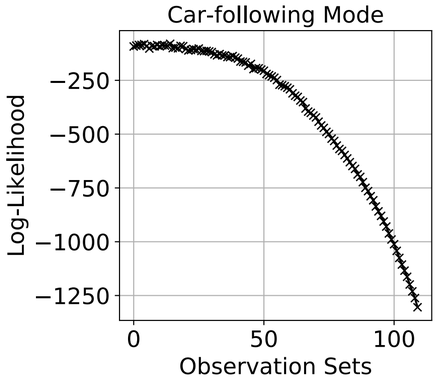


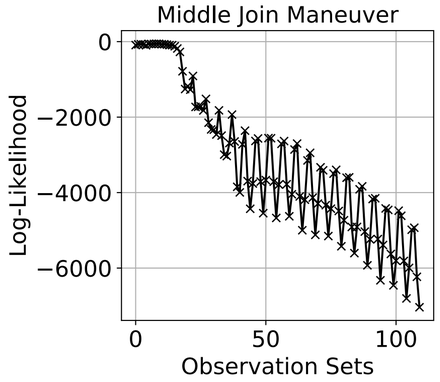
Forward-based Anomaly Detection

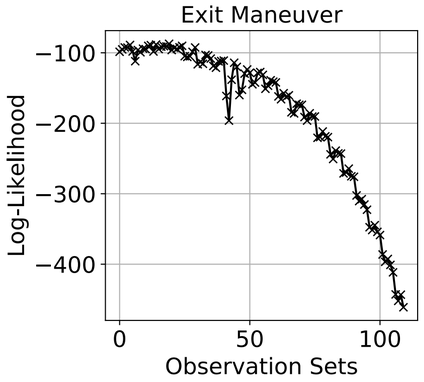


Results & Performance of Platoon Behavior Recognition

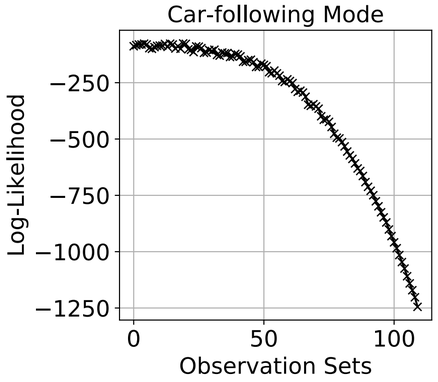
Flatbed Controller

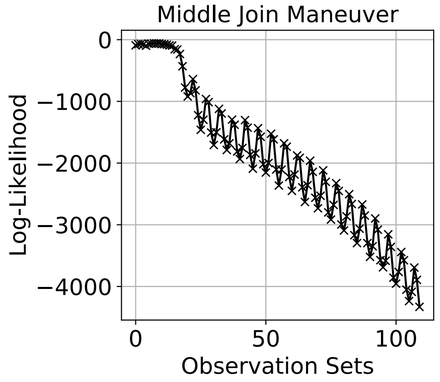


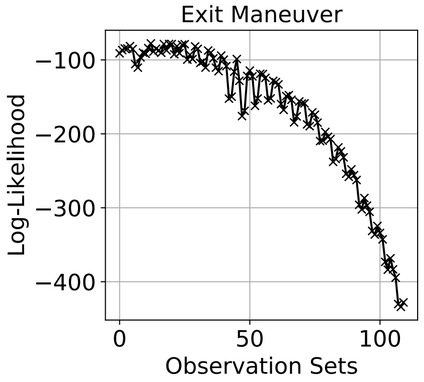




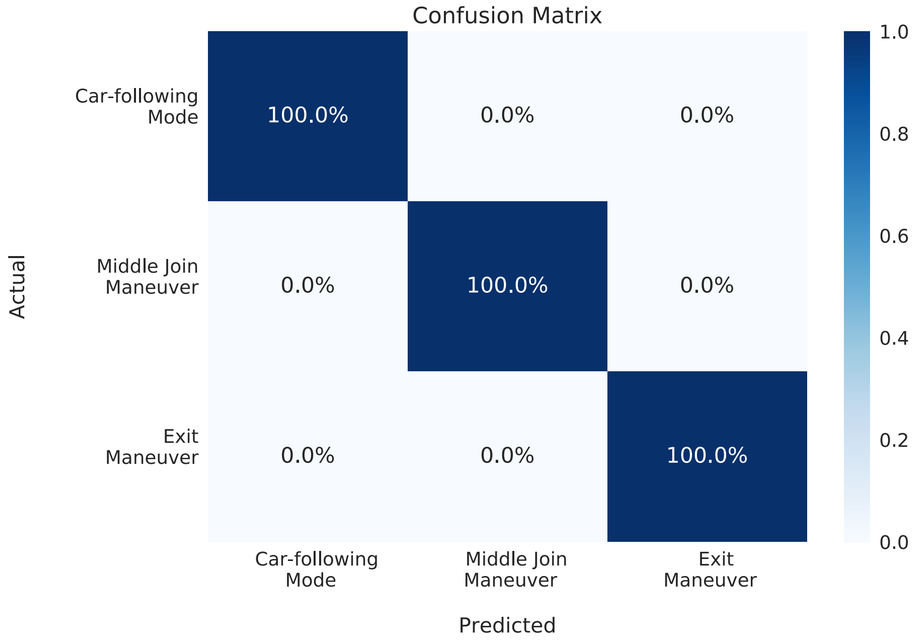
Path Controller







Performance of Platoon Behavior Recognition



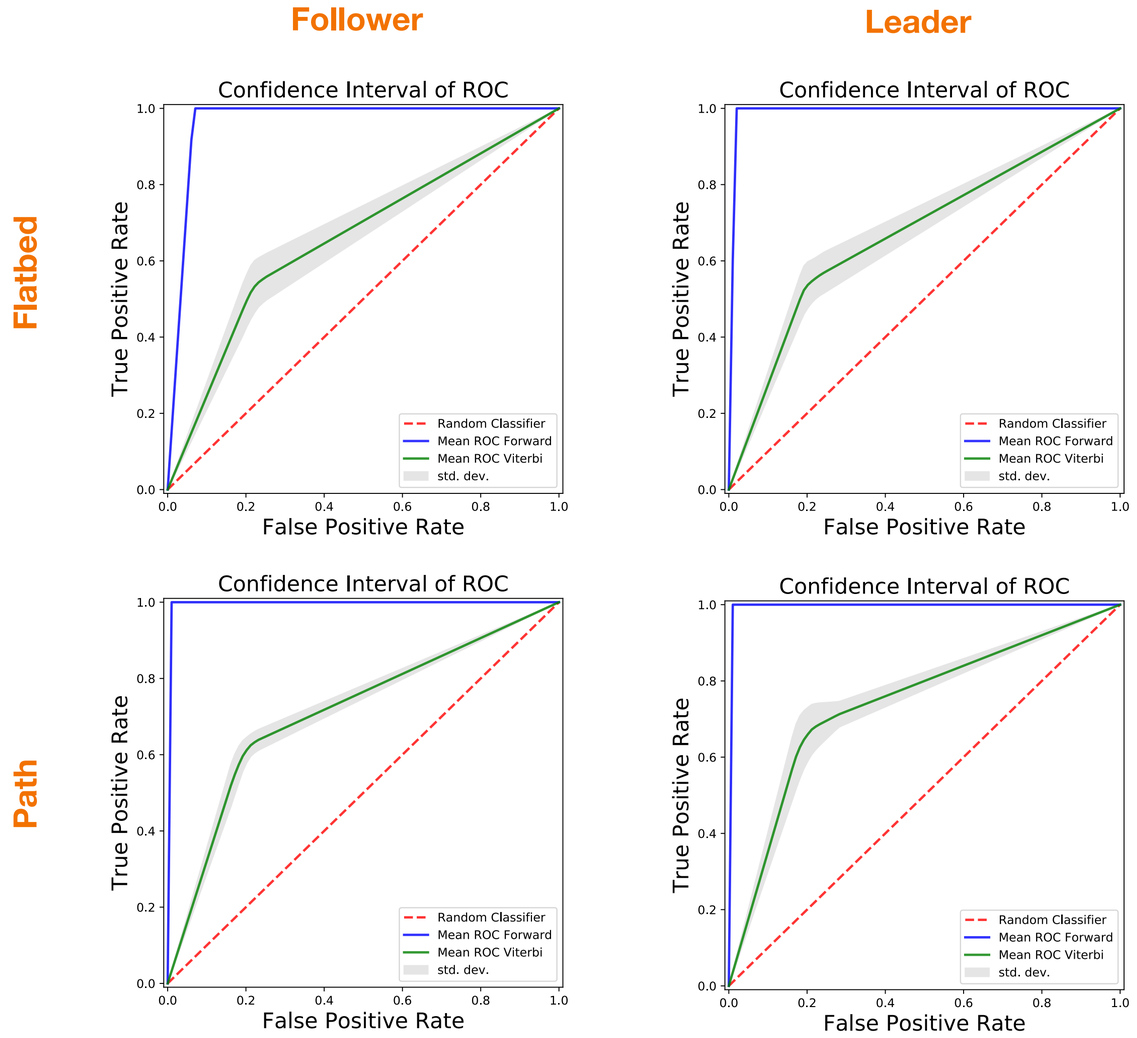
Performance Evaluation

Comparing Performance of Anomaly Detection Approaches

1. Area Under the Curve (AUC) is used to compare the performance of Viterbi-based and Forward-based anomaly detection approaches

2. By repeating the experiment of detecting AccInjectionAttack at Speed 50 for 50 times, AUC is computed

3. Forward-based algorithm is approaching 1 which means that the model has a good performance in separating the positive class (e.g., misbehavior) and the negative class (e.g., benign behavior).

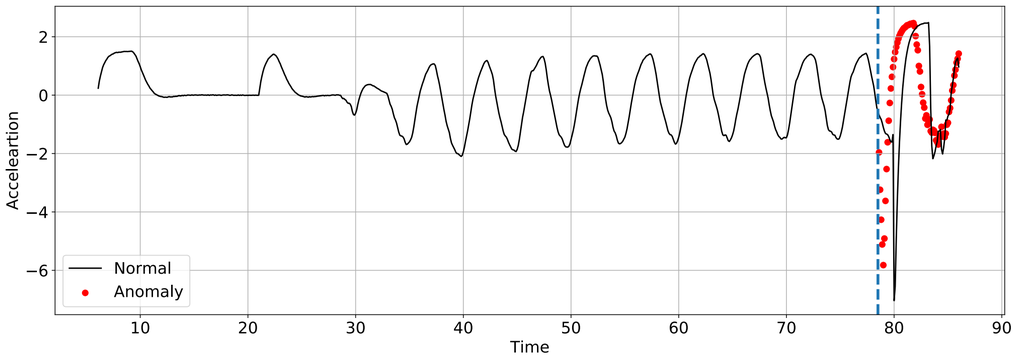


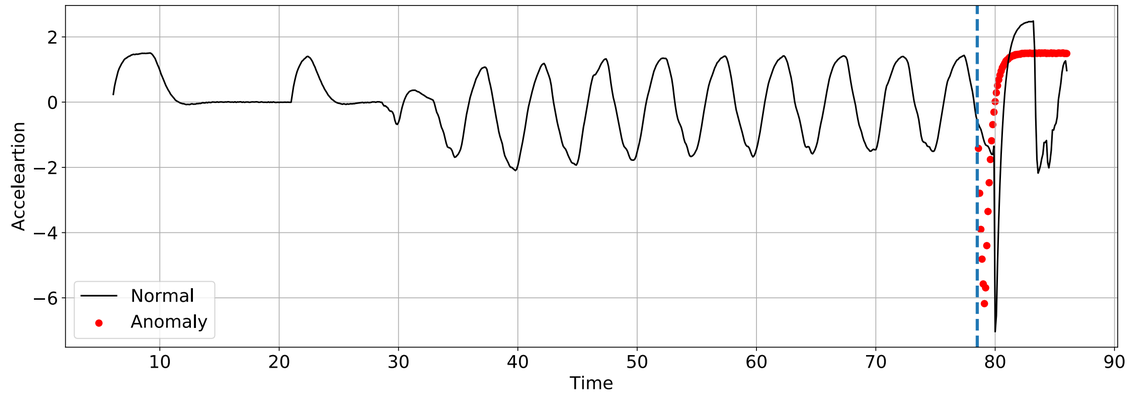
Results of Anomaly Detection

Middle Join Maneuver

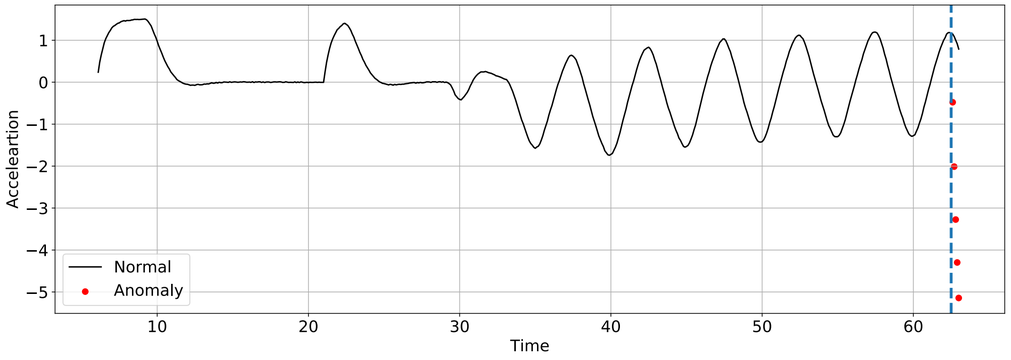
Follower | **Leader**

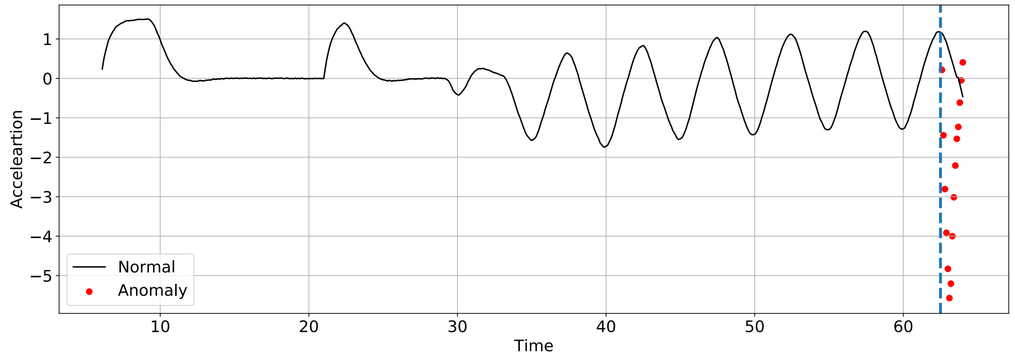
Flatbed





Path





SinusoidalSmartPosFalsificationAttack (Speed 150)

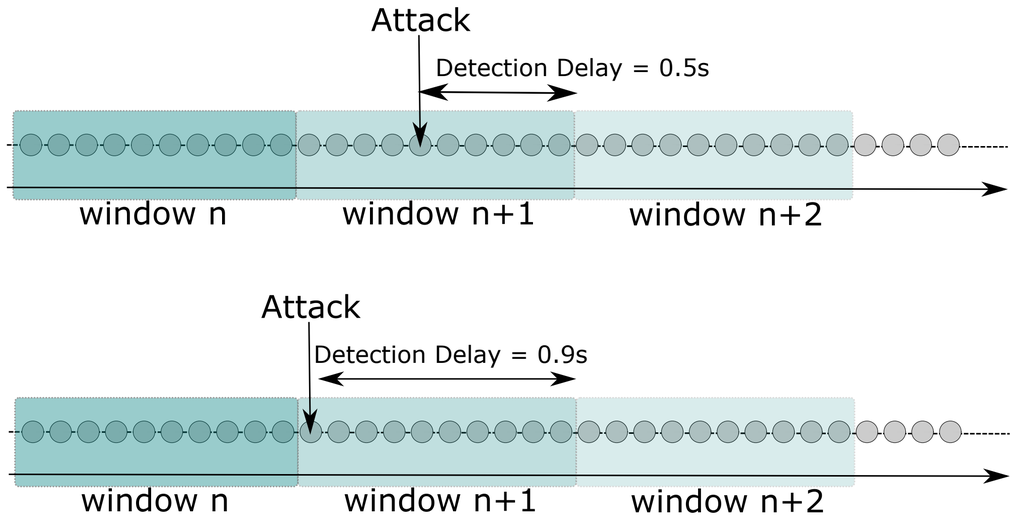
SinusoidalSmartPosFalsificationAttack (Speed 150)

Detection Delay

1. The proposed MDS works in a sliding window. Ten observation samples (1 second) are defined as a sliding window.

2. Detection delay is calculated by adding processing delay (0.4s) and time between attack and end of sliding window. Detection delay varies between 0.5s to 1.3s.

3. Worst case scenario in terms of detection delay would happen if the attack is launched at the beginning of the sliding window



Conclusion

Platoon Behavior Recognition

The framework is able to recognize a set of driving maneuvers at an early stage even if only a limited set of input signals are used (small window of 1 second). the framework is generic and can be extended to consider other maneuvers

Anomaly Detection

Two approaches are proposed to detect 21 different attacks. Forward-based algorithm showed better and consistent performance comparing to Viterbi-based algorithm. It has been shown that a single run is enough for learning the normal behavior. The detector is very sensitive and accordingly it’s able to detect very small deviations. This high sensitivity may introduce false alarms for jerkiness scenarios.

Overall Performance

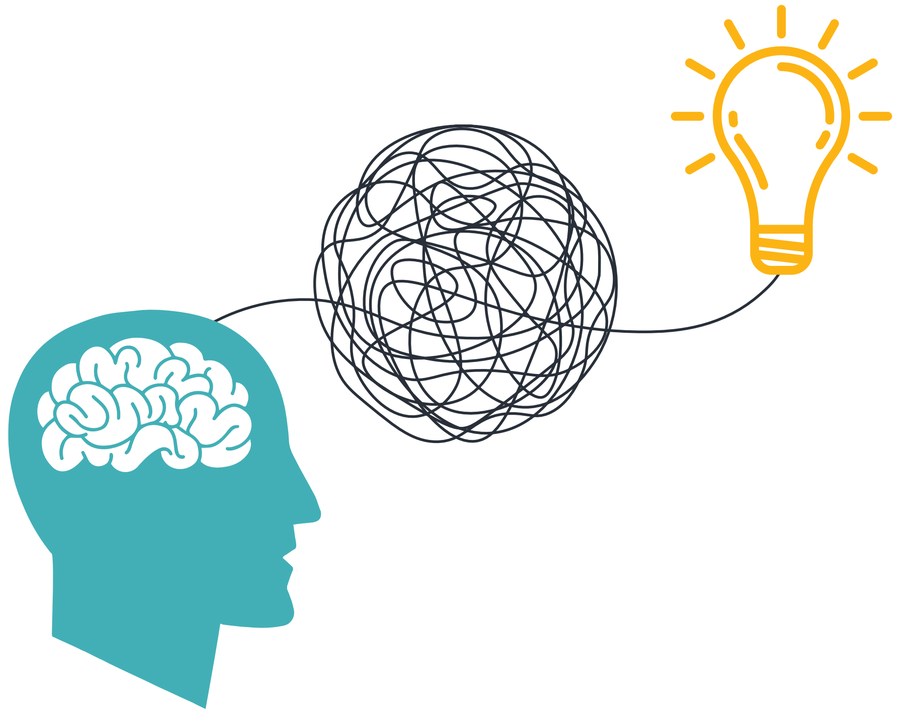
Overall performance of maneuver recognition and anomaly detection shows excellent results in terms of performance metrics such as accuracy, recall, precision, f1 score and detection delay. Detection delay varies between 0.9s to 1.3s

Future Work

1. Investigate if ML features can be re-engineered in way that makes GMMHMM model independent of platoon speed

2. Run the framework on all vehicles and compare results to see if sharing misbehavior reports between vehicles will make the the framework more robust

3. Sharing misbehavior reports with other vehicles opens the issue of fake reports. Cooperative schemes assume that reports from other vehicles are trustworthy if they carry correct signatures. Validation of received misbehavior reports need to be investigated.



VePMAD: A Vehicular Platoon Management Anomaly Detection System

A Case Study of Car-following Mode, Middle Join and Exit Maneuvers

Weaam Bayaa

Examiner: Panagiotis Papadimitratos
Supervisor: Mohammad Khodaei
NSS Group

Sep 13, 2021

References

- [1] ETSI TR 103 298 V0.0.4
- [2] K. Garlichs, A. Willecke, M. Wegner and L. C. Wolf, "TriP: Misbehavior Detection for Dynamic Platoons using Trust," *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 455-460, doi: 10.1109/ITSC.2019.8917188.
- [3] Marco Iorio, Fulvio Risso, Riccardo Sisto, Alberto Buttiglieri, and Massimo Reineri. Detecting injection attacks on cooperative adaptive cruise control. In 2019 IEEE Vehicular Networking Conference (VNC), pages 1{8, Los Angeles, CA, USA, December 2019. IEEE.
- [4] UCAR, Seyhan; ERGEN, Sinem Coleri; OZKASAP, Oznur. IEEE 802.11 p and visible light hybrid communication based secure autonomous platoon. *IEEE Transactions on Vehicular Technology*, 2018, 67.9: 8667-8681.
- [5] ABDO, Ahmed, et al. Application level attacks on Connected Vehicle Protocols. In: 22nd International Symposium on Research in Attacks, Intrusions and Defenses ({RAID} 2019). 2019. p. 459-471.
- [6] ELAMASSIE, Mohammed, et al. Effect of fog and rain on the performance of vehicular visible light communications. In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. IEEE, 2018. p. 1-6.
- [7] KALOGIANNIS, Konstantinos. Investigating Attacks on Vehicular Platooning and Cooperative Adaptive Cruise Control. 2020.
- [8] Steven So, Prinkle Sharma, and Jonathan Petit. Integrating plausibility checks and machine learning for misbehavior detection in VANET. In 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), pages 564{571. IEEE, December 2018.
- [9] Matthias Matousek, Mahmoud Yassin, Rens van der Heijden, Frank Kargl, et al. Robust detection of anomalous driving behavior. In 2018 IEEE 87th Vehicular Technology Conference (VTC Spring), pages 1{5, Porto, Portugal, June 2018. IEEE.
- [10]

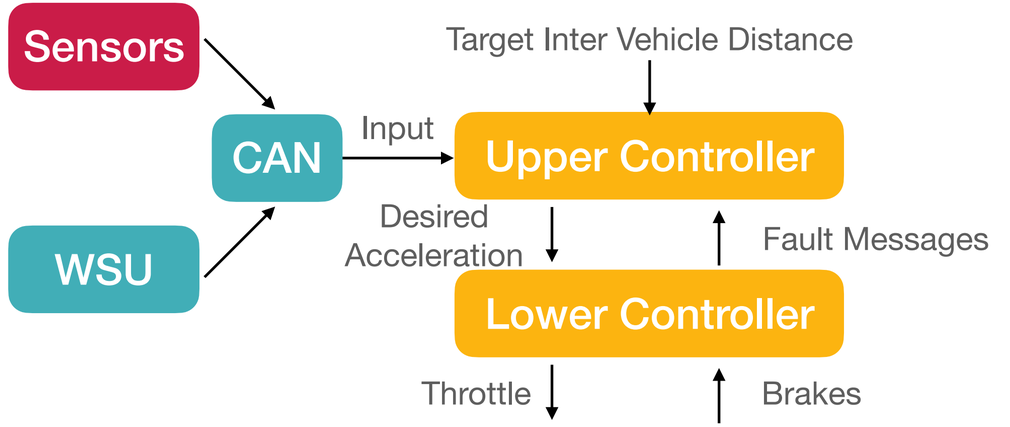
Platoon Controllers

1. Car-following mode -> longitudinal controller only used. Platoon maneuver mode -> combined lateral and longitudinal controller alongside PMP are needed.

2. PATH and Flatbed are using CVS (Constant Vehicle Spacing) policy.

3. These controllers are relying on both sensors and V2V for sharing information.

4. PATH is susceptible to fake beacons due to its dependence on multiple variables while Flatbed is susceptible to speed falsification perpetrated by the leader (it simply require all members to share a common speed value)



A basic architecture of a driving control system

Constant-Time Gap (CTG) = Constant Time Headway (CTH)
Constant Distance Gap (CDG) = Constant Vehicle Spacing (CVS)

Controller	Policy	Predecessor			Leader		
		d	s	a	p	s	a
ACC	CTH	✓	✓				
PATH	CVS	✓	✓	📶		📶	📶
Consensus	Both	✓			📶	📶	
Flatbed	CVS	✓	✓			📶	
Ploeg	CTH	✓	✓	📶			

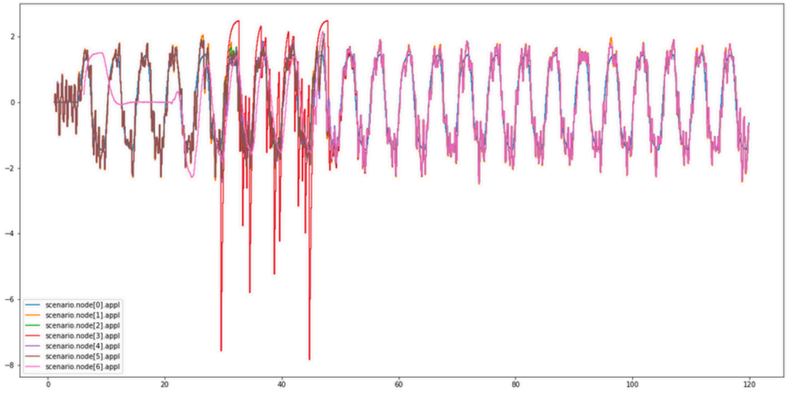
Table taken from [3]

Example of Normal Behavior

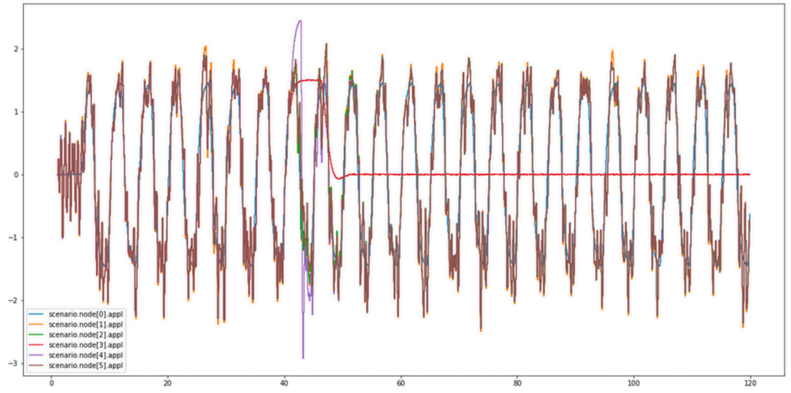
Different Maneuvers, Different Speeds, Same Controller (i.e. Flatbed)

50 km/h

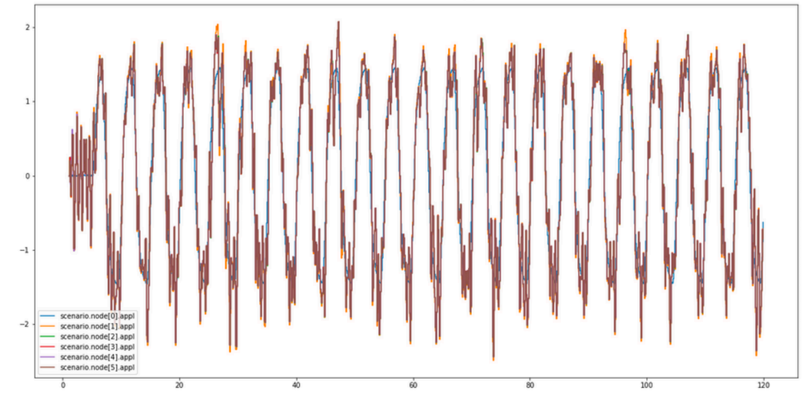
Middle Join Maneuver



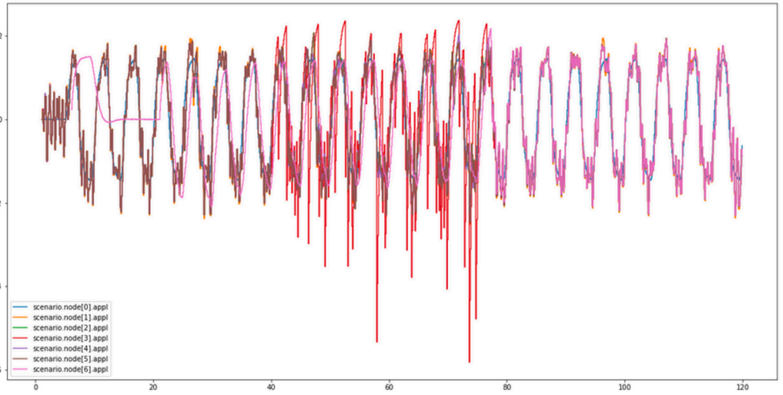
Exit Maneuver

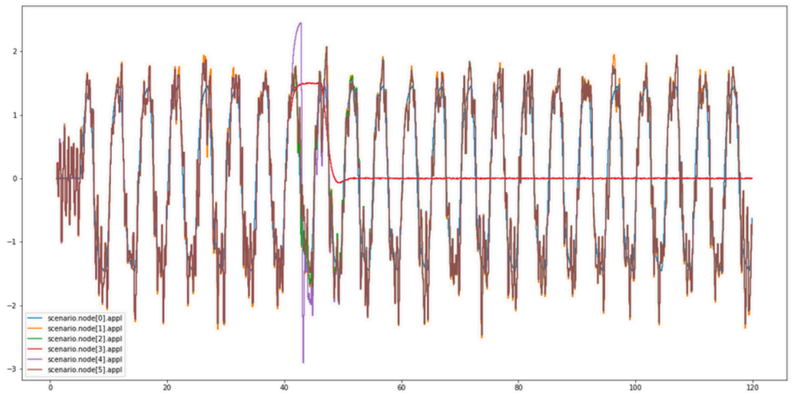


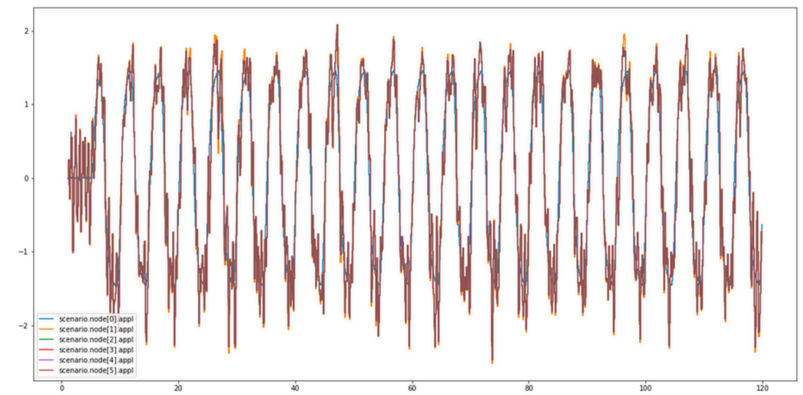
Car-following Mode



80 km/h







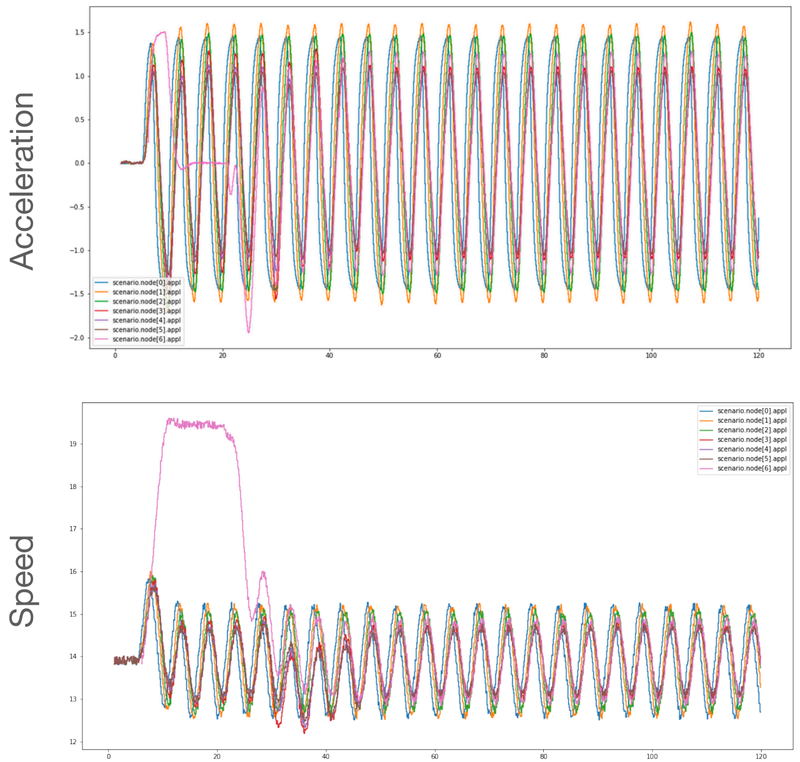
Behavior is different for different maneuvers and different speed

Example of Normal Behavior

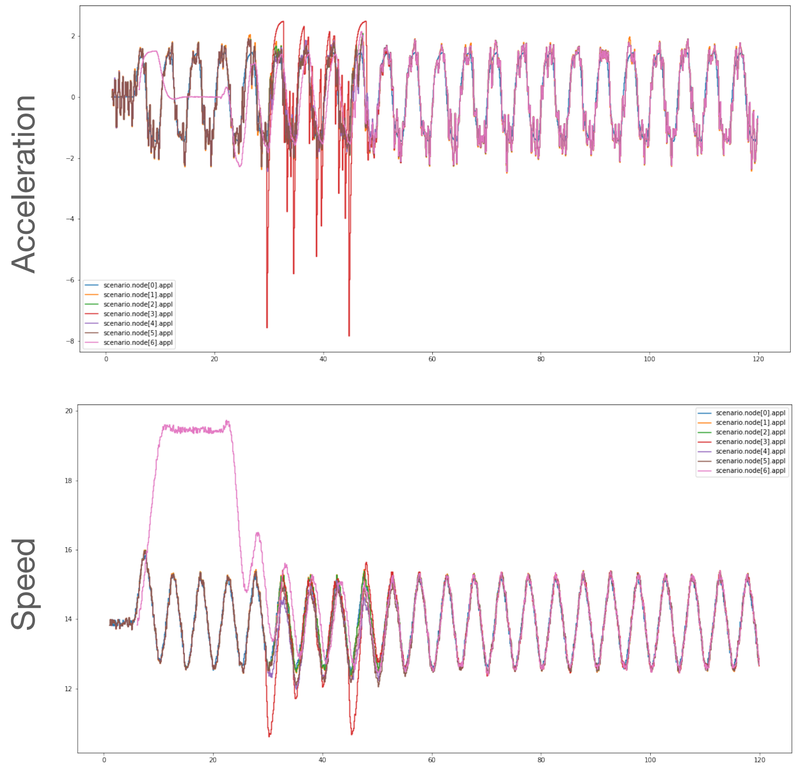
Same Maneuver, Different Controllers

Middle Join Maneuver:

Path



Flatbed



Behavior is different if different controllers are considered

Limitation of Existing Solutions

Securing CACC and VANET

1. Rule-based mechanisms [2]:
Multiple rules to do plausibility checks or create a trust score in a reputation system

2. Kalman Filter [3]:
It has been used in the context of CACC to detect injection attack by detecting increase of deviation of received measurements through V2V from Kalman Filter estimations

3. ML-based mechanisms [8], [9]:
Any linear ML model that relies on linear function for its prediction function cannot be applied if system dynamics and observation models are non-linear.

Securing Platoon During Maneuvers

1. SP-VLC Protocol [4]:
VLC may not work reliably under adverse weather conditions (e.g. rain and fog). In addition, the road surface may impact VLC system performance [6]

2. Pre-Approval Protocol [5]:
RSUs are assumed to be trusted while RSUs are usually distributed outside and vulnerable to be compromised by attackers.

3. Kalman Filter [7]:
It has been attempted to use Kalman Filter during middle join and exit maneuvers. Results have shown that Kalman Filter detected two vehicles as malicious during the normal operation.

Existing Solutions

Securing Platoon During Maneuvers

SP-VLC Protocol [4]

1. Proposal: new hybrid security protocol (SP-VLC) which combines both IEEE 802.11p and Visible light communication (VLC).
2. Road side attacker is assumed to transmit either fake maneuver request packet or a fake maneuver response packet.
3. VLC is used for secret key establishment to construct the initial secret key securely between each pair of consecutive platoon members.
4. A mechanism to switch to either transmission over both IEEE 802.11p and VLC or VLC only according to conducted attack.

VLC may not work reliably under adverse weather conditions (e.g. rain and fog). In addition, the road surface may impact VLC system performance [6]

Pre-Approval Protocol [5]

1. Proposal: send maneuvers requests to RSUs which in turn will verify it with relevant information. If all checks are verified, maneuver approval will be sent to requestor.
2. Attacker attempt to exploit the functionality of the PMP implementation
3. During validation process, platoon will be switched to safe mode where it moves within the safety speed limit, i.e 20 mph.
4. If rejection is received from RSU, maneuver will be aborted.

RSUs are assumed to be trusted while RSUs are usually distributed outside and vulnerable to be compromised by attackers.

Attacks

Attack	Attack Value	Attack Position	Maneuver	Controller
PosInjectionAttack	[3, 5, 7, 9, 11] m	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
SpeedInjectionAttack	[-50, 0, 50, 100, 150] km/h	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
AccInjectionAttack	[-30, -10, 0, 10, 30] m/s2	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
GradualPosFalsificationAttack	[-10,40] m	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
GradualSpeedFalsificationAttack	[-10,17] m/s	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
GradualAccFalsificationAttack	[-10,10] m/s2 m	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
SmartPosFalsificationAttack	[-10,10] m	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
SmartSpeedFalsificationAttack	[-10,10] m/s	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed
SmartAccFalsificationAttack	[-10,10] m/s2	[0, 2]	Car-following, Middle Join, Exit	Path, Flatbed

Table taken from [7]

21 attacks for 2 controllers, 2 attacker positions and 3 maneuvers considering 4 speed scenarios:
Total of 1008 tests

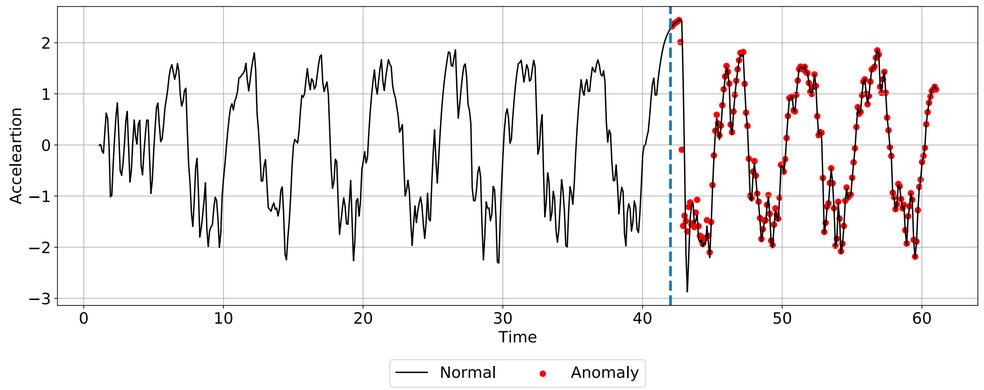
Results of Anomaly Detection

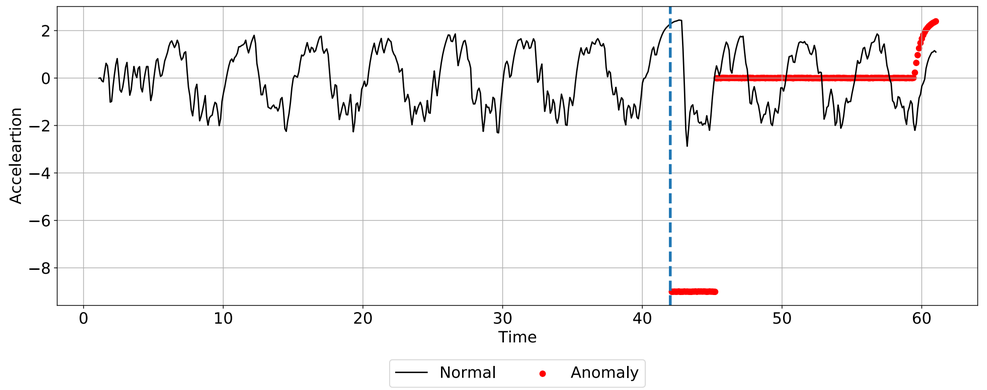
Exit Maneuver

Follower

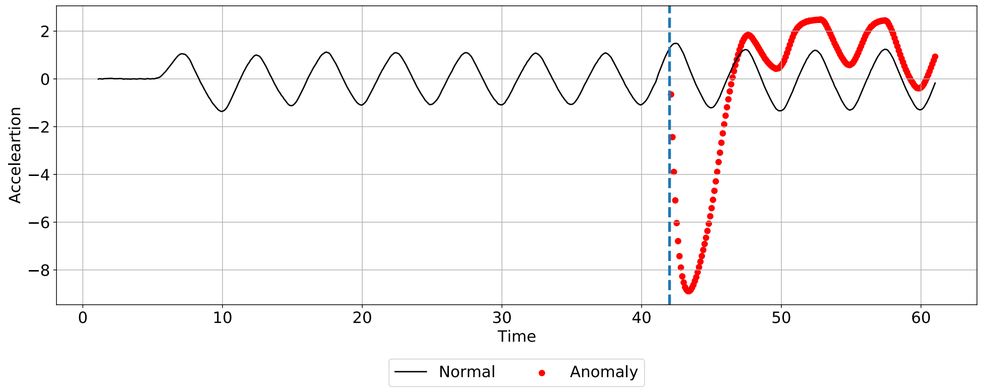
Leader

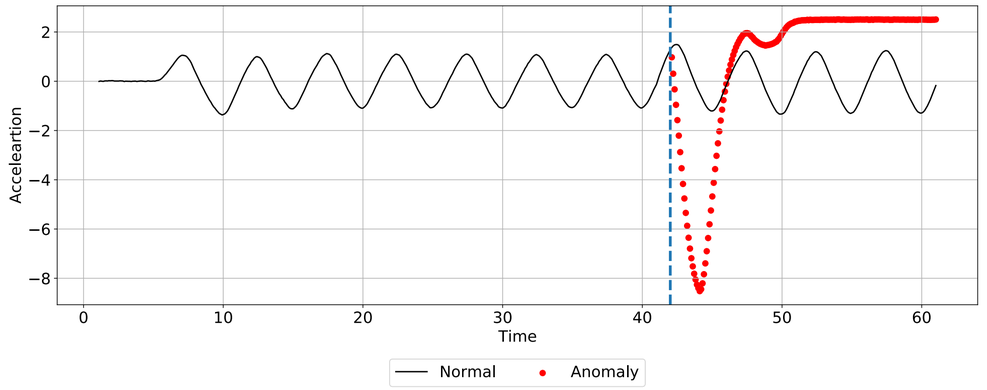
Flatbed





Path





SinusoidalSpeedInjectionAttack (Speed 100)

SinusoidalSpeedInjectionAttack (Speed 100)

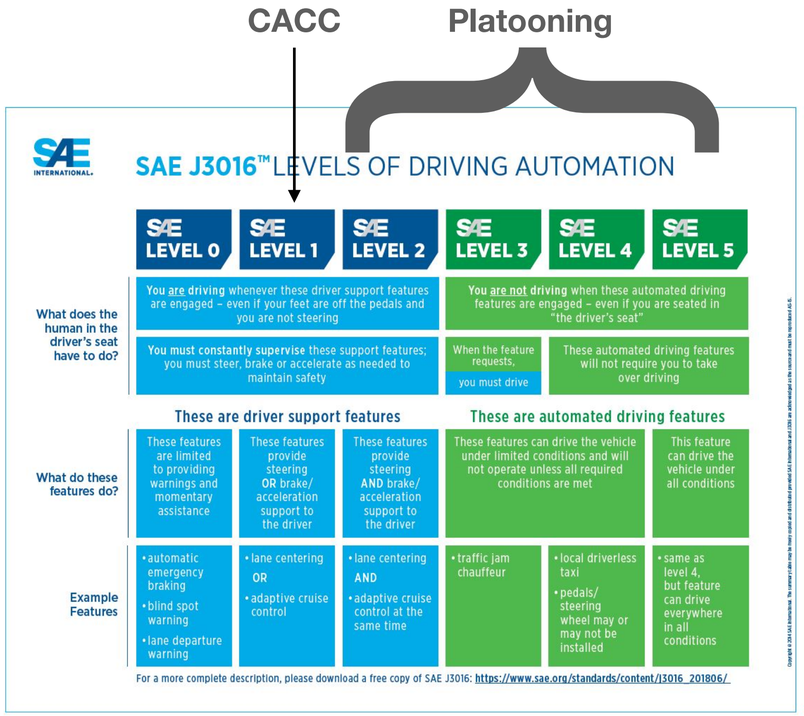
CACC vs Platooning

1. Cooperative Adaptive Cruise Control (CACC) is a term that has been used loosely in recent years and is often mistakenly assumed to be synonymous with platooning.

2. Important distinctions between CACC systems and automated truck platooning systems:

   A. For CACC, only truck speed control will be automated (usually addresses only longitudinal control). The drivers will still be responsible for actively steering the vehicle, lane keeping, and monitoring roadway and traffic conditions.

   B. Truck platooning systems relies on a Constant Distance Gap (CDG) control strategy “separation between vehicles remains unchanged with speed”. The CACC control strategy is based on a Constant-Time Gap (CTG) “distance between vehicles is proportional to the speed”.



Picture taken from [10]

Methodology (Hidden Markov Model)

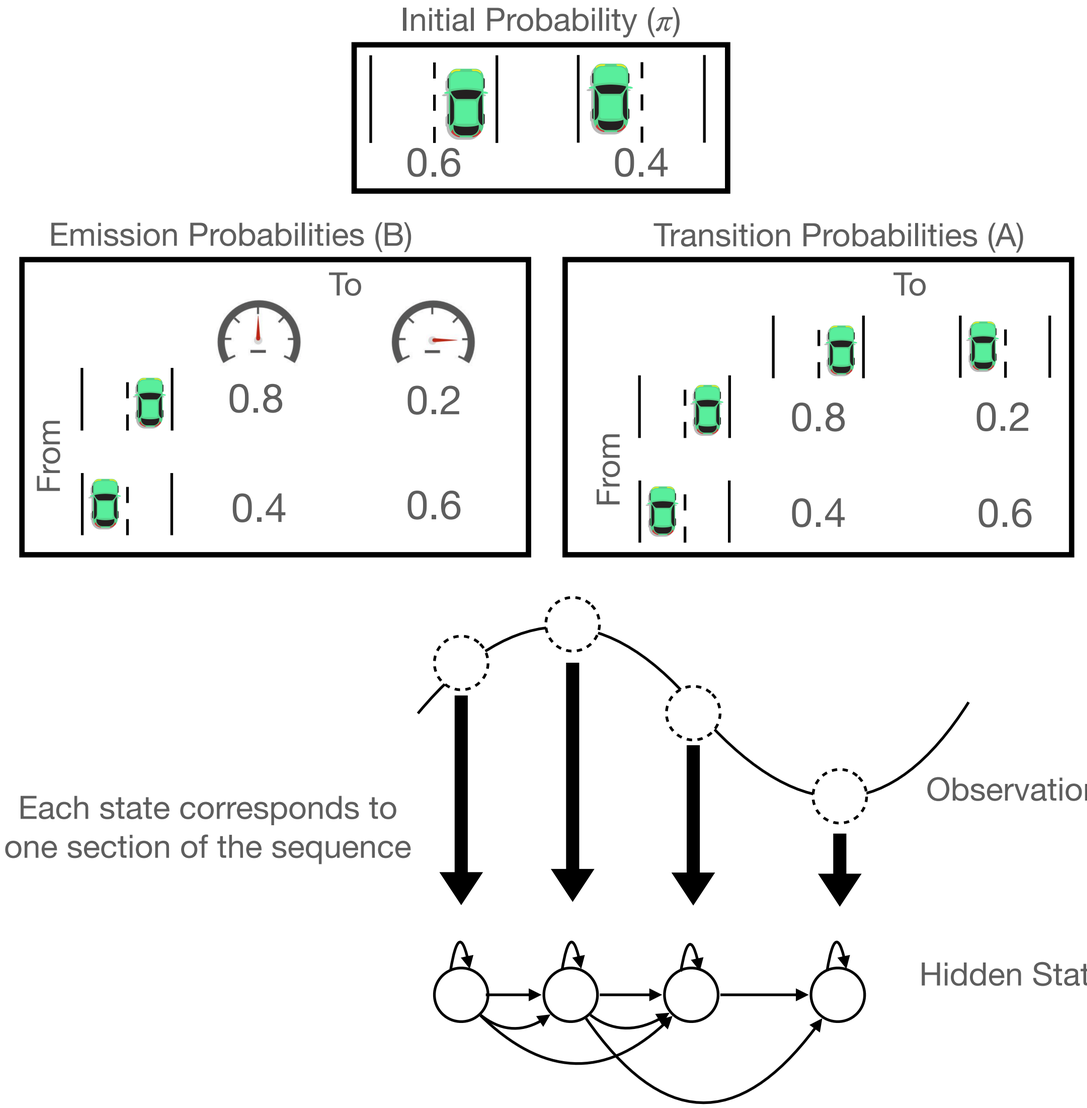
Fundamental Concepts

1. Markov Model: Stochastic model for changing systems. Markov property assumes that future state depends on current state

2. Hidden Markov Model (HMM) is statistical Markov Model in which the system being modeled is assumed to be a Markov process with unobserved (i.e. hidden) states.

3. Hidden Markov Models application include Automatic Speech Recognition (ASR), gesture recognition, …etc.

4. Hidden States: left lane and right lane
   Observables: speed (Discrete: high speed, low speed)
   How to infer lane from speed?

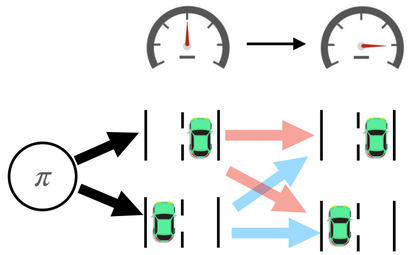


Three Main Tasks for HMM

1

Evaluation/Scoring: Given a HMM model M and x, estimate the probability of observation: Find P(x|M). -> forward-backward algorithm

Given the model parameters and observed data (sequence of speed observations) , calculate the model likelihood (what is the most likely current lane?)

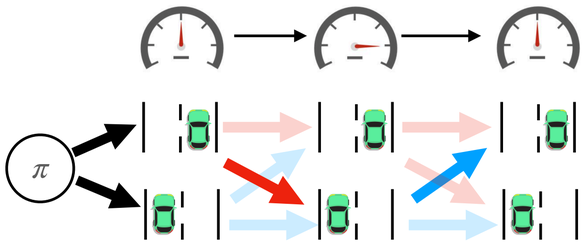


Probability of path and sequence given a model

2

Decoding: Given a HMM model M and observed sequence x, compute the hidden sequence that best models the observations: Find z. -> Viterbi Algorithm

Given the model parameters and observed data (sequence of speed observations), what is the most likely underlying lane sequence?



Best path (estimate the optimal sequence of hidden states)

3

Learning: Given the observed sequence x, estimate the most likely HMM model M using the maximum likelihood method: Find M. -> Maximum Likelihood Estimation or Estimation Maximization

Supervised Learning: Some observation sequences (Speed) and their associated states (lane) have been included in training dataset (MLE)

Unsupervised Learning: In case it is not possible to sample from hidden states (EM)

($\text{argmax}_\lambda P(O_{train}|\lambda)$)

Model Selection

Bayesian Information Criterion (BIC)

1. One problem during training of HMM model is the assumption of knowing HMM topology beforehand which is not a realistic assumption in real-world applications. Previous example was simple, e.g. 2 states (left lane and right lane).

2. Bayesian Information Criterion (BIC) is a criterion to select models among a finite set of models. It's based on likelihood function. BIC is a commonly used criterion to balance between likelihood of data and number of free parameters.

3. When fitting models, it's possible to increase the likelihood by adding parameters but that would result in overfitting. BIC resolves this problem by introducing a penalty for the number of parameters in the model.