Strategic mitigation against wireless attacks on autonomous platoons
Supplementary materials

No Author Given

No Institute Given

A Details of platoon model

A.1 Communication topology

Communication among vehicles are usually established using dedicated short-range communication or cellular vehicle-to-everything technology over dedicated and secured links. There are multiple communication topologies available for platoon members to share critical information (e.g., location, velocity etc.) to others ([2]). In this paper, we adopt the *predecessor-leader following* topology as in Figure 1. Information from the leading vehicle is transmitted to all following vehicles. In addition, each following vehicle also receives information from its immediate proceeding vehicle.

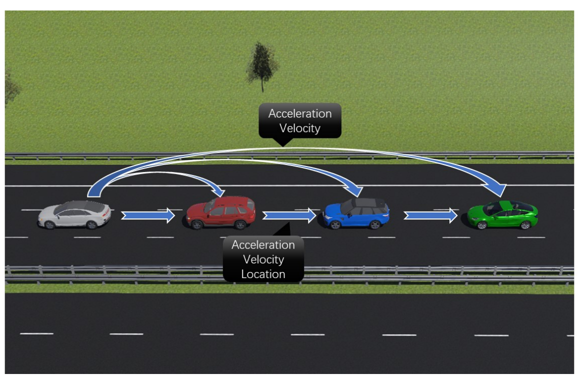

Fig. 1. Predecessor-leader following information topology.

A.2 Platoon control strategies

Cooperative Adaptive Cruise Control (CACC) CACC control algorithm utilizes transmitted messages from the leader and the preceding vehicle to maintain a desired inter-vehicle distance or gap distance gap_{des}. The targeted acceleration $\ddot{x}_i$ of the i-th vehicle in the platoon is calculated by:

$$\ddot{x}_i = \alpha_1 \ddot{x}_{i-1} + \alpha_2 \ddot{x}_0 + \alpha_3 \dot{\epsilon}_i + \alpha_4 \dot{\epsilon}_0 + \alpha_5 \epsilon_i \tag{1}$$

$$\epsilon_i = x_i - x_{i-1} + l_{i-1} + gap_{des} \tag{2}$$

$$\dot{\epsilon}_i = \dot{x}_i - \dot{x}_{i-1} \tag{3}$$

$$\dot{\epsilon}_0 = \dot{x}_i - \dot{x}_0, \tag{4}$$

where $\ddot{x}_{i-1}$ and $\ddot{x}_0$ are the accelerations (second derivative of location x_{i-1} and x_0) of the preceding vehicle and the leader respectively. Similarly, $\dot{x}_{i-1}$ and $\dot{x}_0$ represent their corresponding speed (first derivative of location x_{i-1} and x_0). The gap distance error ϵ_i is calculated based on a desired gap distance gap_{des} and vehicle length l_{i-1}. $\dot{\epsilon}_i$ denotes the speed error with respect to the preceding vehicle, while $\dot{\epsilon}_0$ is the speed error with respect to the leader. Coefficients α's can be found in Appendix A.3.

Adaptive Cruise Control (ACC) ACC is a relatively mature technology widely deployed in many modern vehicles by now. It improves driving experience by performing the longitudinal following control task utilizing sensor readings from Radar etc. Compared with message-based control algorithms such as CACC, inaccurate sensor readings from nearby noisy and dynamic environment raise concerns about the safety risks and robustness of such technologies especially for vehicle platoon where the inter-vehicle distance is much narrower. Nevertheless, ACC could be treated as a back-up controller when the message-based controller is suspicious or completely compromised due to malicious attacks. In this work, we implemented a Proportional-Integral (PI) controller based on measured relative velocity v_{radar} (i.e. $\dot{x}_{i-1} - \dot{x}_i$) and gap distance L_{radar}(i.e. $x_{i-1} - x_i$) between a platoon member and its immediate predecessor. Let the target velocity of i-th vehicle to be $\dot{x}_i$,

$$\ddot{x}_i = K_P \cdot v_{radar} + K_I \cdot \epsilon_{radar} \tag{5}$$

$$\epsilon_{radar} = gap_{des} - L_{radar} \tag{6}$$

where ϵ_{radar} is the distance error between the desired gap distance gap_{des} and the current radar measurement L_{radar} with vehicle length l_{i-1} taken into account. Controller gains can be found in Table 1.

A.3 Controller coefficients

The α parameters in (1) are calculated by:

$$\alpha_1 = 1 - C_1; \quad \alpha_2 = C_1; \quad \alpha_5 = -\omega_n^2 \tag{7}$$

$$\alpha_3 = -(2\xi - C_1(\xi + \sqrt{\xi^2 - 1}))\omega_n \tag{8}$$

$$\alpha_4 = -C_1(\xi + \sqrt{\xi^2 - 1})\omega_n \tag{9}$$

where C_1 represents the weighting between the preceding and the leader vehicle, ξ is the damping ratio and ω_n is the bandwidth of the controller. The default values taken from [1] can be found in Table 1. In our work, anomaly detector training of the feed-forward deep neural networks is based on the five types of data in (1) , i.e. $\ddot{x}_{i-1}$, $\ddot{x}_0$, $\dot{\epsilon}_i$, $\dot{\epsilon}_0$ and ϵ_i.

Table 1. key controller parameters

parameters	values
C_1	0.5
ξ	1
ω_n	0.2 Hz
K_P	1
K_I	-0.25

B WorldInfo parameters

The general information about the simulated world is contained in WorldInfo node. For example, basicTimeStep node configures the simulation step size executed by Webots, which was chosen based on the trade-off between simulation speed and accuracy.

Table 2. WorldInfo parameters

Environment parameters	Values
gravity	9.81 g/m^2
ERP	0.6
basicTimeStep	10 ms
optimalThreadCount	4
FPS	60 (frames per second)
physicsDisableTime	1 s
physicsDisableLinearThreshold	0.01 m/s
physicsDisableAngularThreshold	0.01 rad/s
defaultDamping	NULL
inkEvaporation	0
coordinateSystem	NUE
lineScale	1
randomSeed	0

C Sensors equipped on the vehicle model

Essential information is gathered by various types of built-in sensor modules in Webots. The following table contains types of sensors, their property as well as their locations embedded on each vehicle in our simulation.

Table 3. Vehicle's equipment

Positions	Names	Parameters	
sensorsSlotFront	Radar	maxRange	20
		horizontalFiledOfView	0.16
		verticalFiledOfView	0.3
		speedNoise	0.3
		rangeNoise	0.3
		frequency	24
	Receiver (leader)	type	radio
		aperture	-1
		channel	1
		signalStrengthNoise	0
	Receiver (predecessor)	channel	2
	GPS	type	satellite
		accuracy	0
		speedNoise	0
sensorsSlotRear	Emitter	type	radio
		aperture	-1
		channel	3
		baudRate	-1
		maxRange	-1
sensorsSlotCenter	Accelerometer	xAxis	TRUE
		yAxis	TRUE
		zAxis	TRUE

D Parameters for simulation of Urban Mobility (SUMO)

Parameters in the following tables define SUMO vehicle properties for vehicle route generation. Table 4 defines acceleration, deceleration, maximum reachable velocity and length for four vehicle types, whereas Table 5 contains car-following model parameter *sigma*, the eagerness for performing lane changing to gain speed *lcSpeedGain*, the for following the obligation to keep right *lcKeepRight* as well as the vehicles expected multiplicator for lane speed limits *speedFactor*. Other parameters are set as their default values.

Table 4. SUMO definition of vehicles (part 1)

vehicle type	accel (m/s^2)	decel (m/s^2)	maxSpeed (m/s)	length (m)
car	2.6	4.5	55.5	5
bus	0.8	4.5	22.22	10
motorcycle	1.2	4.5	26.4	3
truck	0.8	4.5	18	14

Table 5. SUMO definition of vehicles (part 2)

vehicle type	sigma	lcSpeedGain	lcKeepRight	speedFactor
car	0.5	5	5	normc(1,0.05,0.5,2)
bus	0.8	1	50	normc(1,0.1,0.5,2)
motorcycle	0.5	1	1	1
truck	0.8	1	100	1

References

1. Segata, M., Joerer, S., Bloessl, B., Sommer, C., Dressler, F., Cigno, R.L.: Plexe: A platooning extension for veins. In: 2014 IEEE Vehicular Networking Conference (VNC). pp. 53–60. IEEE (2014)
2. Zheng, Y., Li, S.E., Wang, J., Cao, D., Li, K.: Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies. IEEE Transactions on intelligent transportation systems **17**(1), 14–26 (2015)