Verification of Intelligent Driving Systems

Project Number 1329593 *N. F. Maxemchuk, Shou-pon Lin, Yitian Gu*

Why we need formal verification?

Test tracks and simulations cannot provide the level confidence required in automotive applications.

- Faulty ignition switches in General Motors vehicles have resulted 52 crashes.
- At least 2.6 million vehicles have been recalled
- If each vehicle spent only 1,000 hours on the road, there has been less than 1 crash every 5 x 10^7 hours of operation

Simulations and testing cannot reliably detect events that occur this infrequently

The crashes occurred because of the interaction between the switch and the steering and braking systems – which is a protocol issue

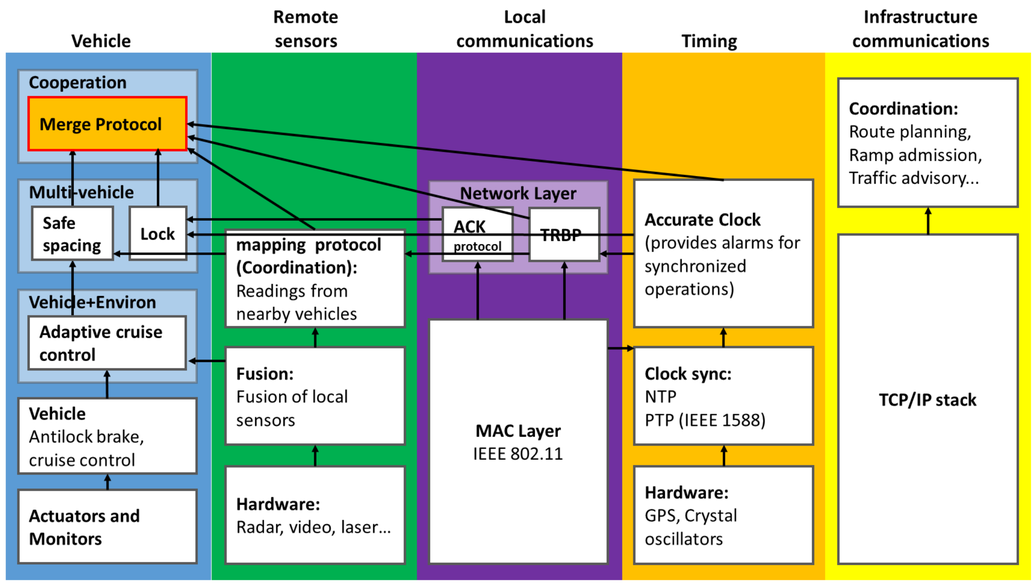
Challenges for the formal verification of vehicle protocols

1. Intelligent driving applications interact with the physical world in multiple ways (*Communications protocols interact in one way*)
2. Vehicles execute time-critical actions, such as braking or merging
3. When multiple vehicles interact with one another, the composite state space grows exponentially with the number of vehicles

1) Multiple stack architecture

A driver-assisted merge protocol, shown below, interacts with the physical world through a) the operation of the vehicle, b) sensors that map the surrounding environment, c) the communications channel, and d) the time domain.

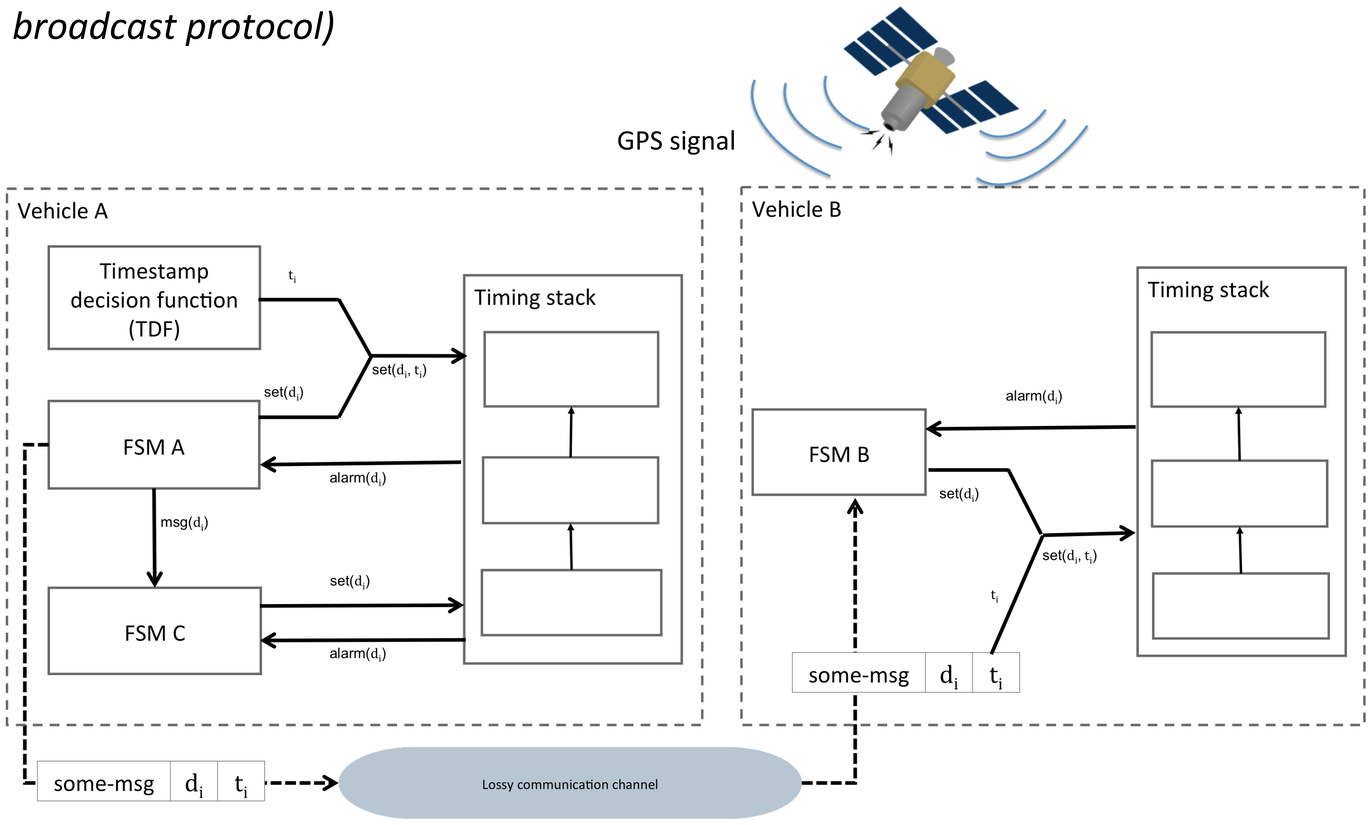
Layered architectures a) break a big verification problem into smaller pieces, and b) allow different modelling and verification procedures to be used for logical functions and physical functions



2) Synchronize clocks in the vehicles and separate time from the logical operation of protocols

Extracting time from protocols allows a simpler verification of the logic

Synchronizing clocks a) reduces the number of execution sequences and b) enables protocols with unique guarantees *(ie. A lock protocol and a fail safe broadcast protocol)*



3) Stratified Probabilistic Verification

Uses linear programming to calculate a tighter bound on the probability of unexplored states than the original probabilistic verification.

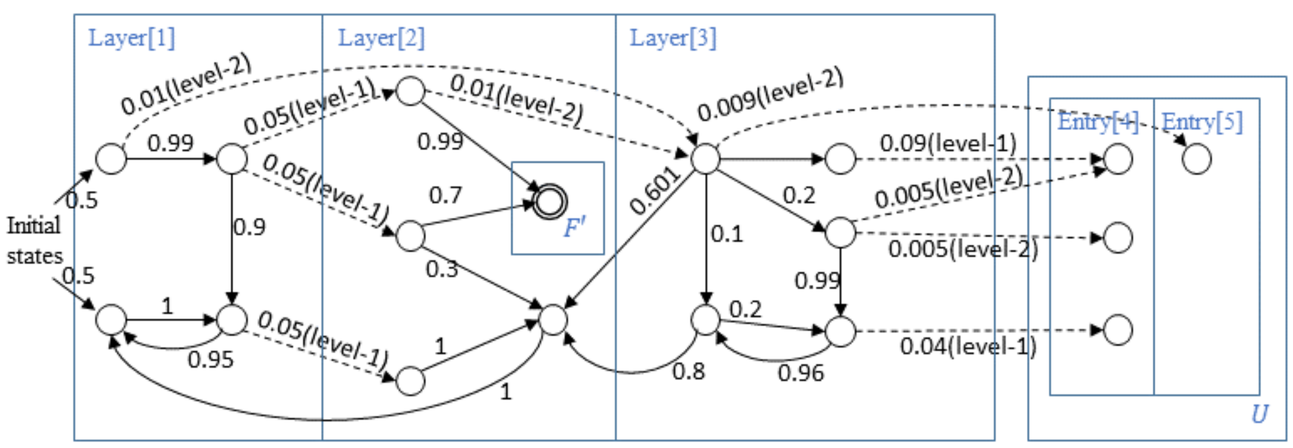


Table 1: Comparison of probability bounds obtained by the original probabilistic verification and by using linear programming

	Lock with 2 conflicting reqs		Driver-assisted merging	
	Original	Stratified	Original	Stratified
Sequences w/o low prob. edge	$4.44 \times 10^6 p$	$24.0012p$	$12837p$	$0.0015p$
Up to sequences bounded by p	$2.68 \times 10^7 p^2$	$105.008p^2$	$3.78 \times 10^6 p^2$	$13.0002p^2$
Up to sequences bounded by p^2	$7.91 \times 10^7 p^3$	$62.0082p^3$	$1.11 \times 10^9 p^3$	$2.0013p^3$
Up to sequences bounded by p^3	$1.39 \times 10^8 p^4$	$4p^4$	TLE	$15p^4$