Rigorous Simulation-based Testing for Autonomous Driving Systems Targeting the Achilles' Heel of Four Open Autopilots

(arXiv:2405.16914 [cs.SE])

Joseph Sifakis

Changwen Li
Rongjie Yan
Jian Zhang

Verimag, June 28, 2024

Overview

- ❑ Validation of ADS
- ❑ Compositional Testing
- ❑ Test Methodology
- ❑ Experimental Results
- ❑ Discussion

Validation of ADS – Validation Methods

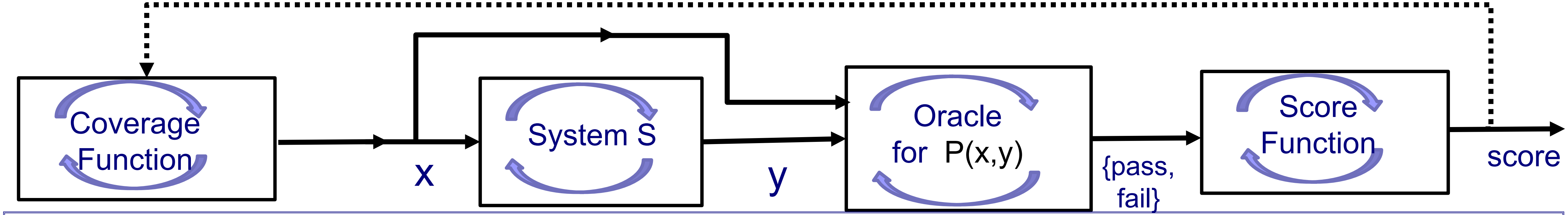
- To validate that "a system S satisfies a property P", systems engineering following epistemic imperatives
  - requires not only that P be rigorously defined, but also that a falsifiable validation method be provided.
  - uses verification or testing techniques.

- Verification consists in comparing a model of S against some specification of P.
  - ~~can examine the whole system behavior described by a model, and decide about the validity of its properties.~~
  - ~~can validate properties involving universal quantification e.g. that all system states are safe, or that for any system run there exists a rejuvenation state.~~
  - is a symbolic (reasoning-based) or a data-based process producing truthful knowledge

- Testing is a controlled experiment on the S (real or virtual) to assess the degree of validity of P.
  - is subject to observability and controllability constraints: distinction between system inputs (controllable) and outputs (observable)
  - is limited to properties P characterizing an I/O relation; properties involving universal quantification can be only falsified.
  - produces empirical knowledge - can fully validate only properties of combinatorial systems with finite input and output domains.
  - Can test methods be applied to AI systems?

Validation of ADS – Test Methods

- Tests are used to validate experimentally that a system y=S(x) satisfies a property P(x,y).
  1. System S: the system under test e.g. an electric bulb, an autopilot or an AI component;
  2. Property P: a predicate (hypothesis) characterizing the I/O behavior of S;
  3. Oracle: is an agent that can decide logically or empirically whether P(x,y) holds producing verdicts *pass* or *fail.*



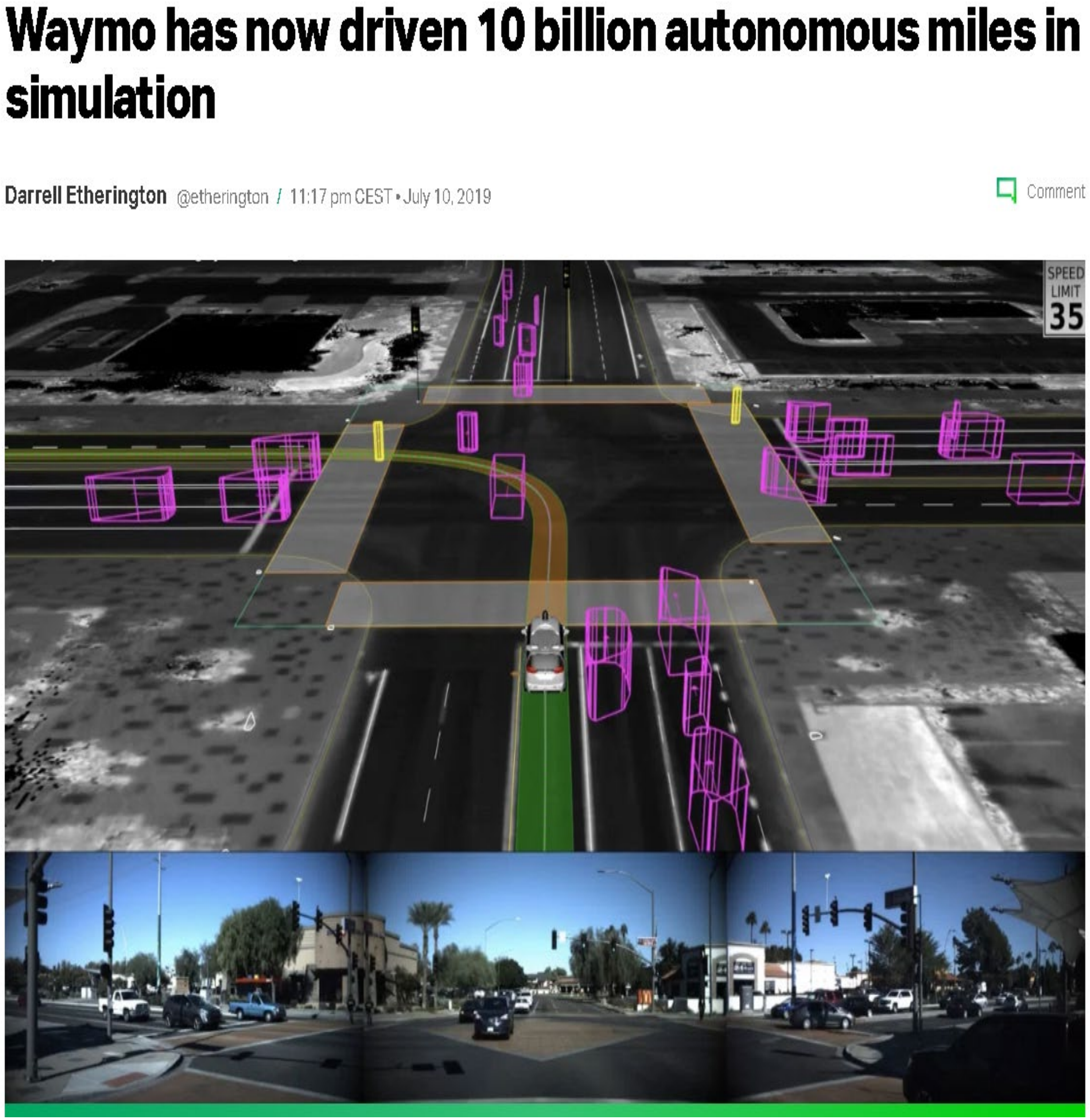
- Test method: How do you choose between possible test cases and decide whether the process is successful or not?

  1. Coverage Function: *coverage*(X)∈[0,1] measures the extent to which the set of test cases X explores the characteristics of the system's behavior in relation to the property P
  2. Score Function: *score*(X,Y) measures for a test set (X,Y) the likelihood that S meets P .

Reproducibility: If (X1,Y1), (X2,Y2) are two sets of tests then:

coverage(X1)=*coverage*(X2) implies *score*(X1,Y1) ~ *score*(X2,Y2)

Validation of ADS – When a Self-driving Car is Safe Enough?



Waymo has now driven 10 billion autonomous miles in simulation

Darrell Etherington @etherington / 11:17 pm CEST • July 10, 2019

Comment

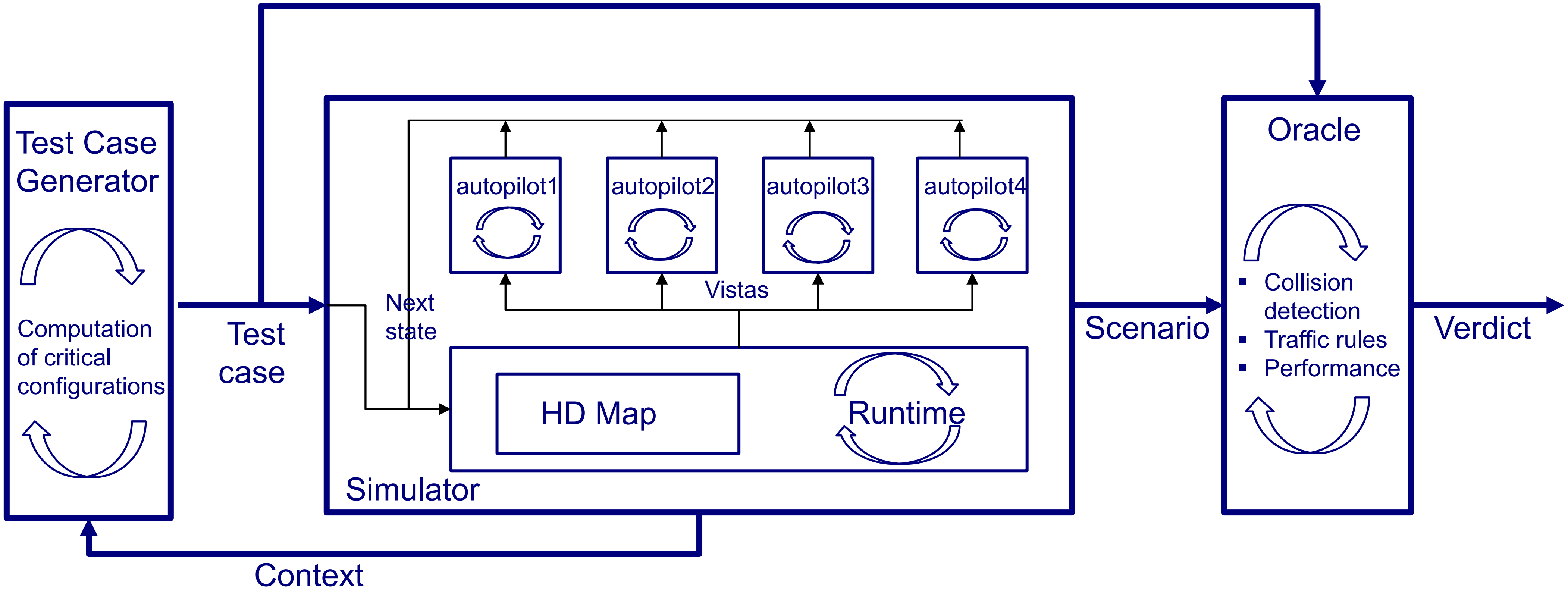
❑ The inability to build global system models limits system validation to simulation and testing.

- Simple simulation is not enough - how a simulated mile is related to a “real mile” ?
- We need evidence, based on coverage criteria, that the simulation deals fairly with the many different situations, e.g., different road types, traffic conditions, weather conditions, etc.

❑ Test methods to calculate, on the basis of statistical analysis, confidence levels for given properties.

- Sampling theory: methods for building sample scenarios that adequately cover real-life situations
- Repeatability: for two samples of scenarios with the same degree of coverage, the estimated confidence levels are approximately the same.

Validation of ADS – Test Environment



Overview

- ❑ Validation of ADS
- ❑ Compositional Testing
- ❑ Test Methodology
- ❑ Experimental Results
- ❑ Discussion

Compositional Testing – Principles

❑ The complexity of ADS can be tamed by applying a compositionality principle.

- **Locality of context:** ADS safety is strongly dependent on the context in which vehicles operate
  - The traffic infrastructure can be seen as the composition of a finite number of patterns comprising different types of roads and intersections with their signalling equipment.
  - We can therefore imagine that a vehicle's safety policy is the composition of elementary policies, each of which is used to drive safely according to the corresponding basic road patterns.

- **Locality of knowledge:** A vehicle's driving policy is based only on local knowledge of the ADS state due to limited visibility.
  - It must therefore drive safely, taking into account the obstacles closest to it, delimited by a visibility zone.
  - In this way, the collective behavior of vehicles in an ADS can be understood and analyzed as the composition of smaller sets of vehicles grouped according to proximity and visibility criteria.

- **Rights-based responsibility**: ADS are a special kind of distributed systems where each agent is responsible for managing a space in its planned route defined by traffic rules.
  - Hence, there is no interaction between the vehicles.
  - Traffic rules guarantee that if each vehicle drives safely in the free space determined dynamically by its rights, then the whole system is safe.
  - This principle of rights-based responsibility greatly simplifies the validation problem as the interaction between vehicles is unidirectional (flow-oriented).
    It is enough to show that a reference vehicle, called the ego vehicle, drives safely in its own free space.

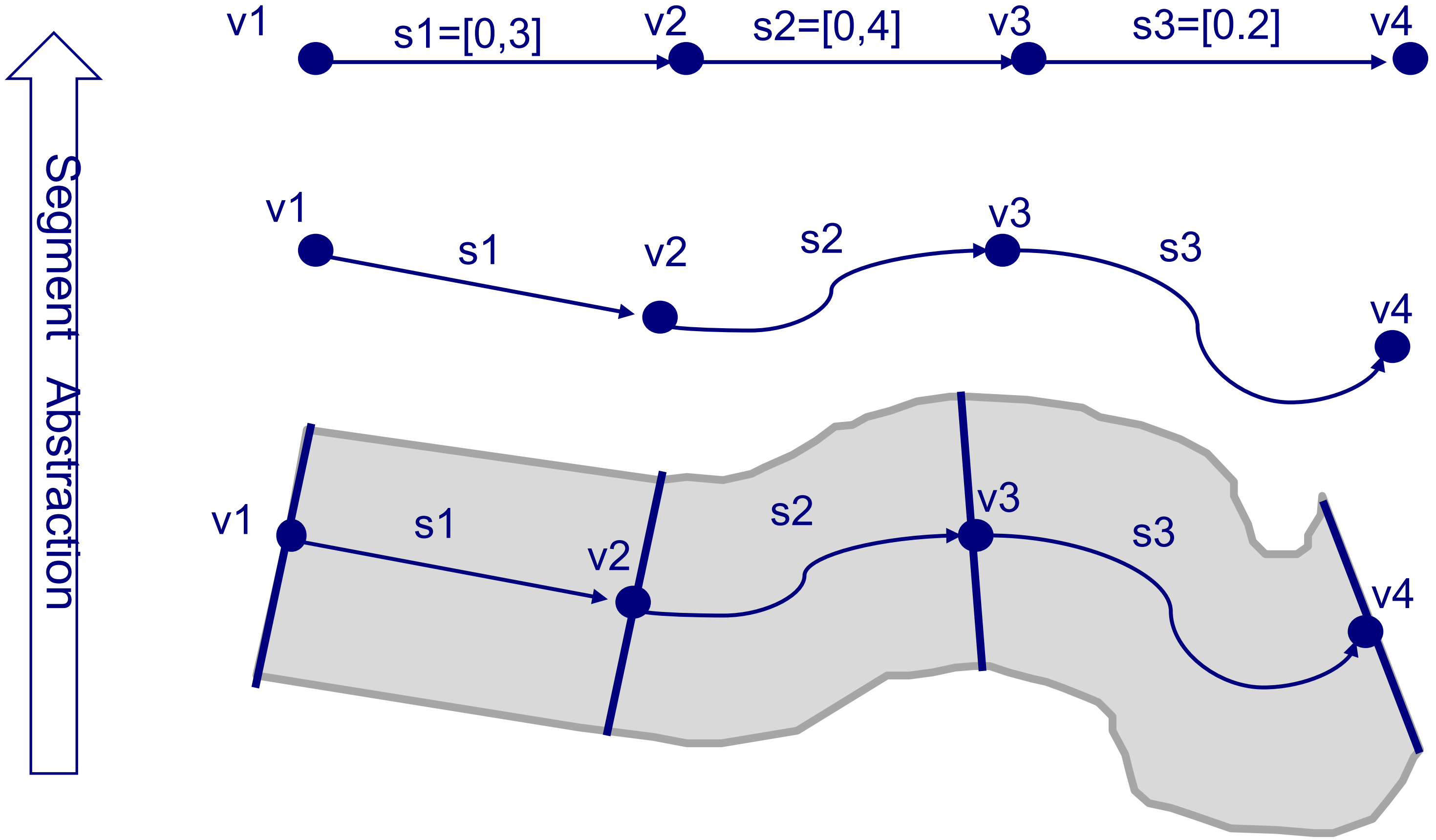
Compositional Testing – Metric Graphs

A Metric Graph is a triple G=(V,S, E) where

- V : set of vertices
- S : set of segments equipped with a partial composition operation ° : $S \times S \rightarrow S \cup \{\perp\}$ and a length norm $||.||: S \rightarrow R_{\geq 0}$
- E : set of edges $E \subseteq V \times S \times V$

Properties of ° :

- associativity;
- additivity: $||s1 \circ s2|| = ||s1|| + ||s2||$;
- refinement: $||s|| = a = a1 + a2 \Rightarrow \exists!\ s1,s2.\ s = s1 \circ s2$ and $||s1|| = a1,\ ||s2|| = a2$



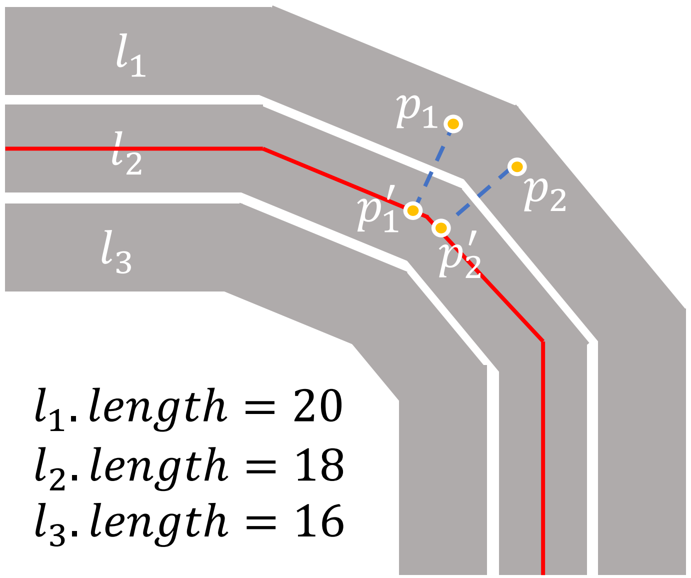
Interval metric graph

Curve metric graph

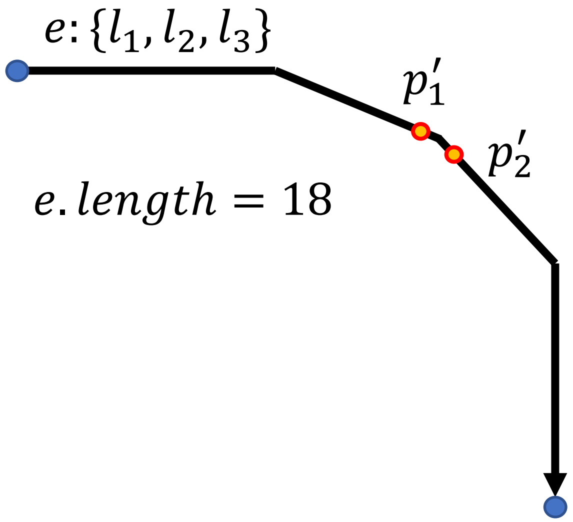
Region metric graph

Compositional Testing – Map Compositionality

From HD maps to their abstract representation



Apollo Map



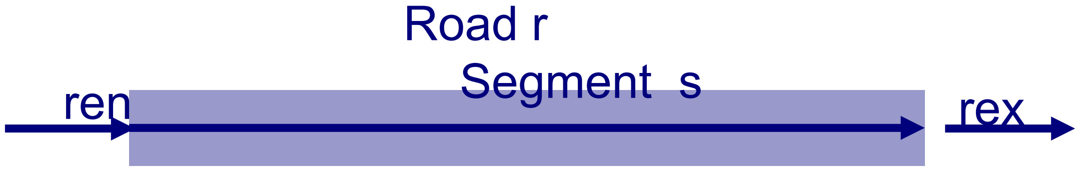
Metric Graph

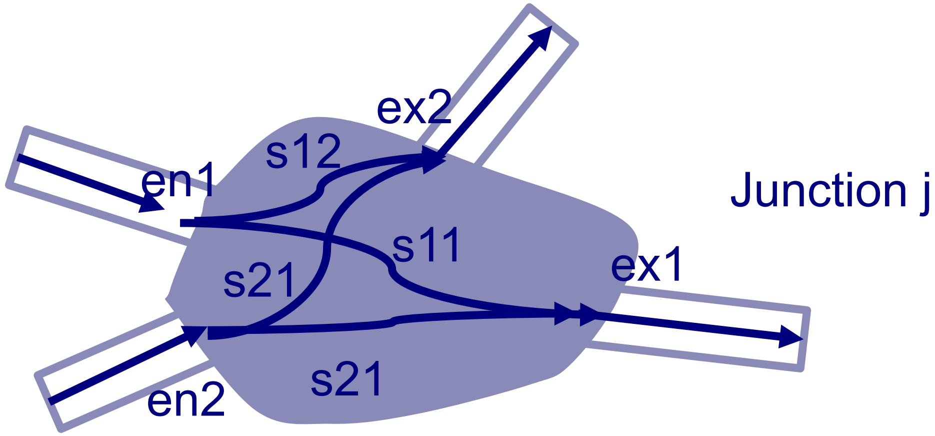
- p_1 on l_1 is mapped to p'_1 on e; p_2 on l_1 is mapped to p'_2 on e.
- The actual distance between p_1 and p_2 is larger than the distance between p'_1 and p'_2
- The difference may affect the checking of safe distance.

Compositional Testing – Maps and ADS

- ❑ A map G ={Gr }$_{r\in R}$ ∪{Gj }$_{j\in J}$ where R and J are respectively the sets of roads and junctions.
  - A road r has a single entrance ren and a single exit rex connected by a segment rs.
  - A junction j is a graph including a set of segments jS of the form sxy where x∈jEN, y∈jEX.

G is obtained by gluing together exits of roads to entrances of junctions and entrances of roads to exits of junctions.





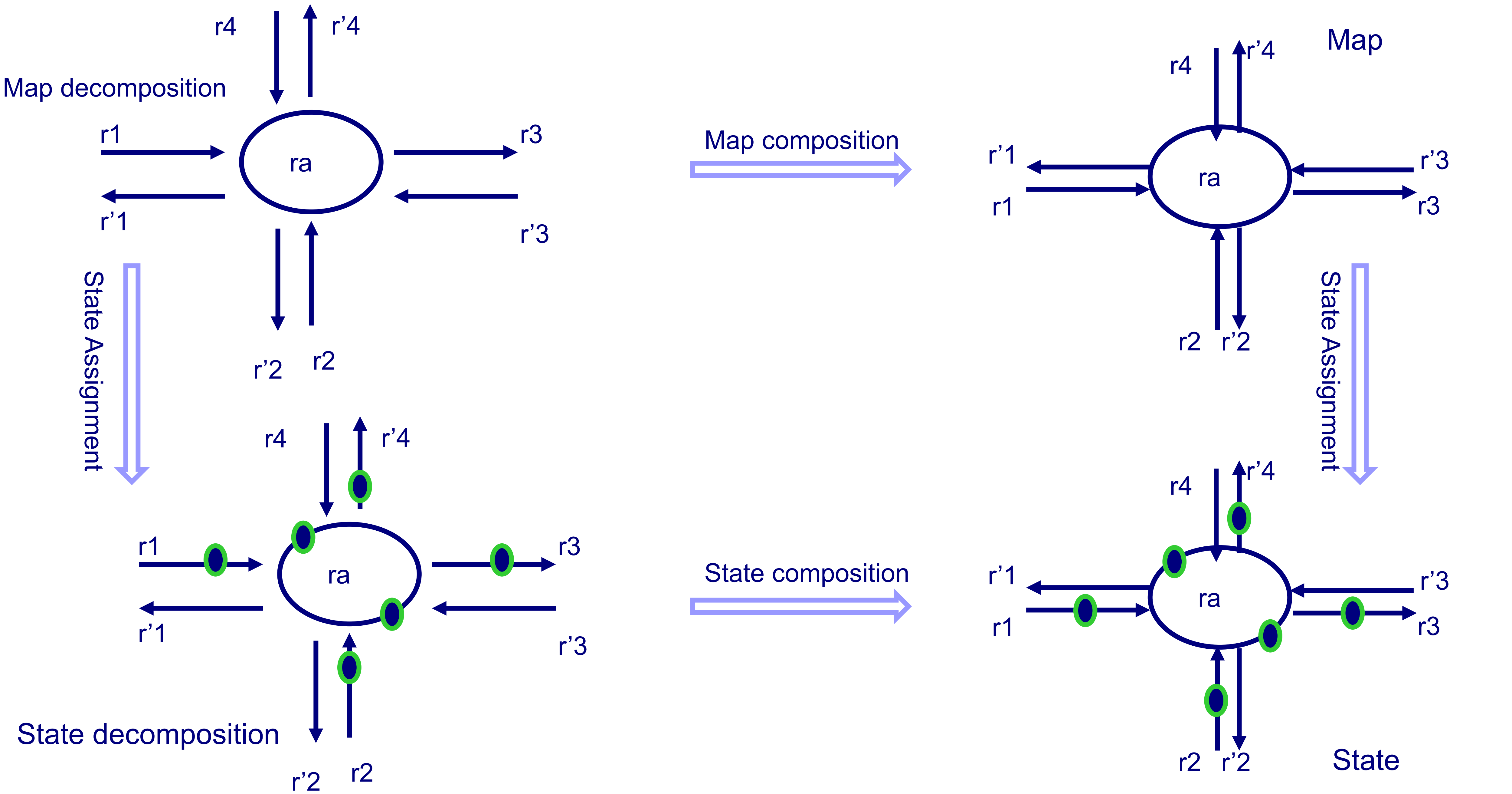
- ❑ An ADS consists of 1) a map G; 2) a set of vehicles C and a set of objects O.

Its state is a set q= {qc}$_{c\in C}$∪ {qo}$_{o\in O}$ =qC∪ qO where,

- qc=<pc, vc, sc, … >, where pc: position of c; vc: the speed of c; sc: a segment of the map describing the route of c
- qo= <po,atto, ….>, where po: position of o; atto is an attribute of o e.g speed limit with vl, state of a traffic light

An ADS is a dynamic system evolving from an initial sate q0=qC0∪ qO0 and through states qi=qCi∪ qOi with qi-- Δt → q(i+1).

Compositional Testing – Map and State Composition



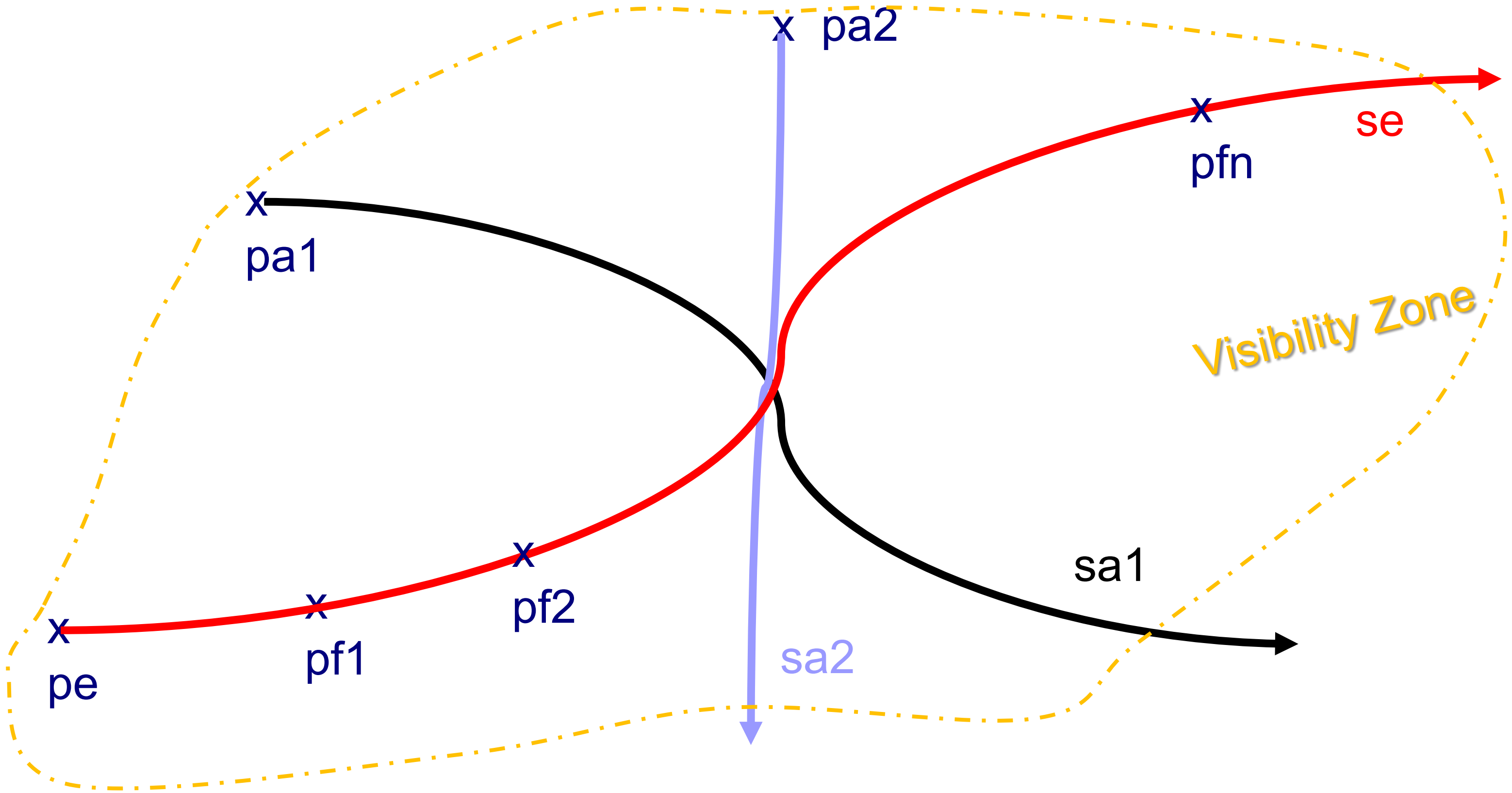
Compositional Testing – The Concept of Vista

A vista is the view of the global state for an ego vehicle including all the relevant obstacles in its neighbourhood depending on its visibility.

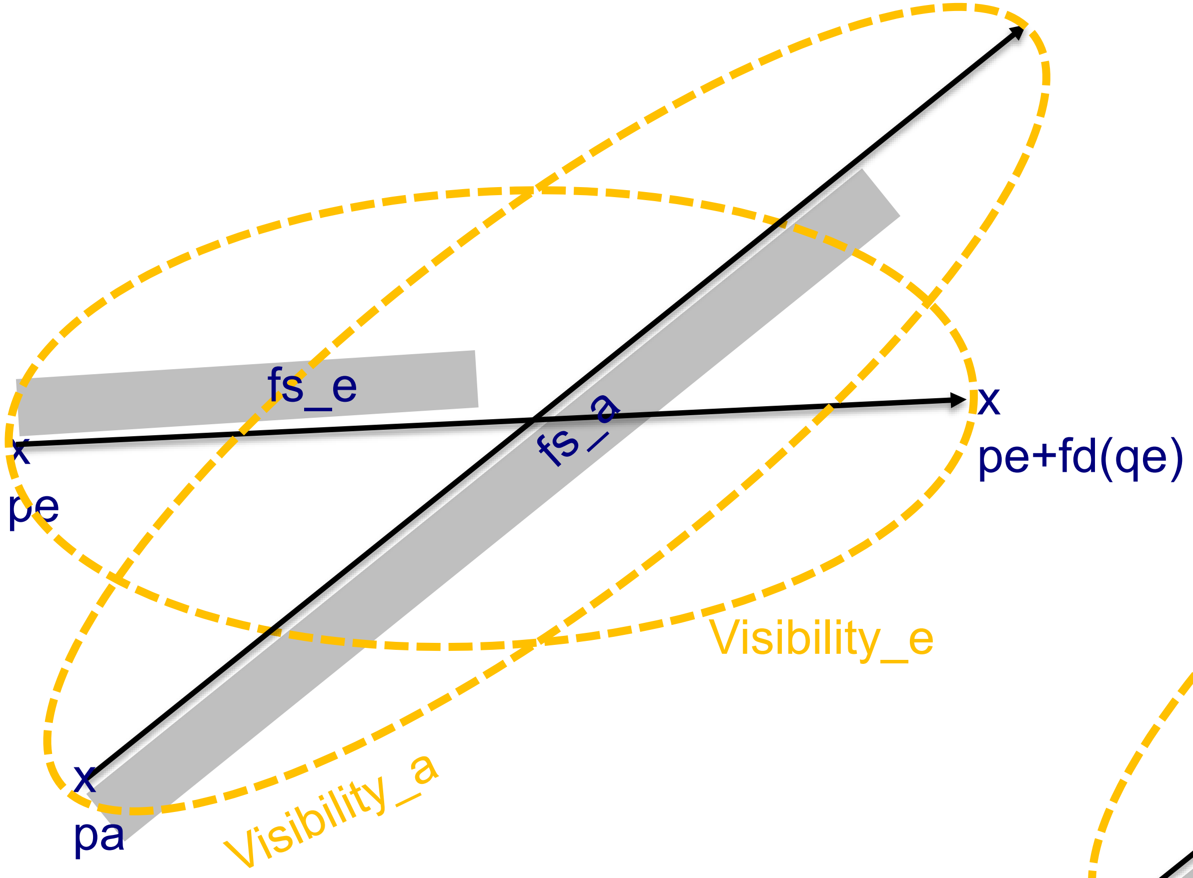
For an ego vehicle its vista vse(q) is defined for a given state q:A vista is a set {<se,ve>, qf1, ,,,, qfn, qa1, …, qak}, with

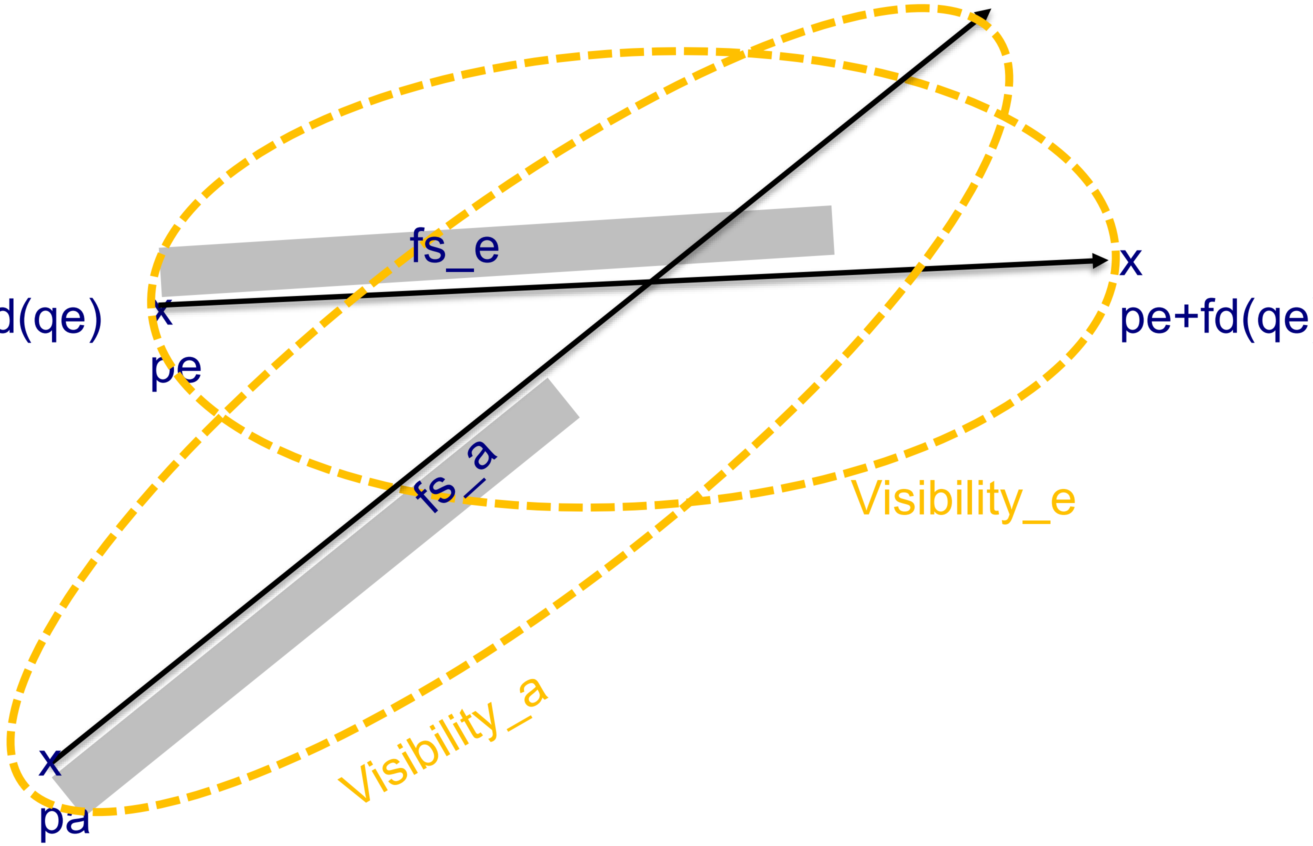
1) the set of the front obstacles F={f1,…, fn} ⊆C∪O located in front of it (on its route segment se) with their position and speed

2) the set of the arriving vehicles A = {a1,…, ak} ⊆C whose route sa may intersect with se or merge into se.



Compositional Testing – The Concept of Free Space

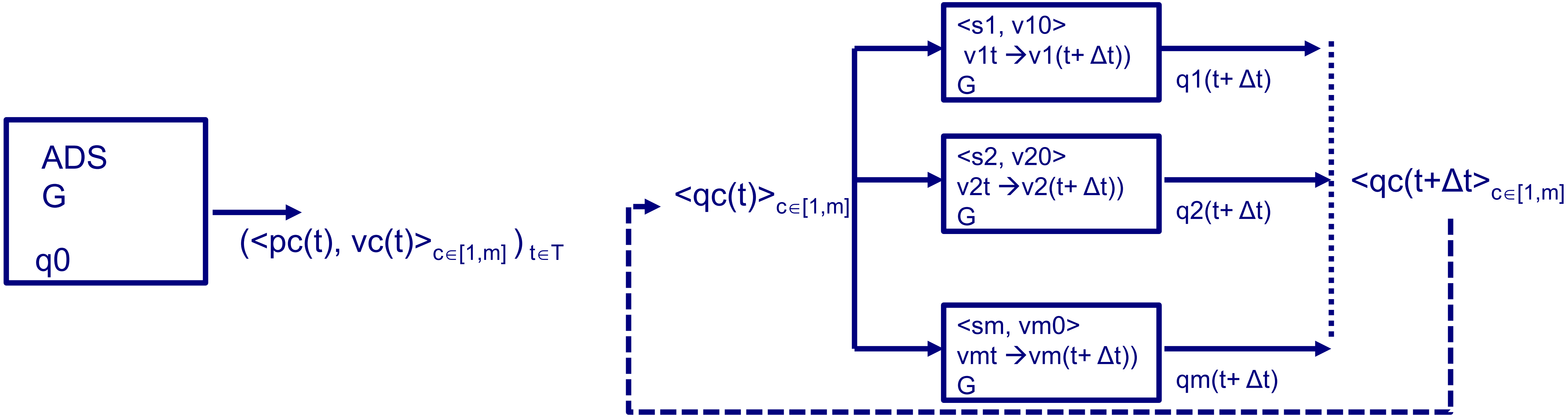


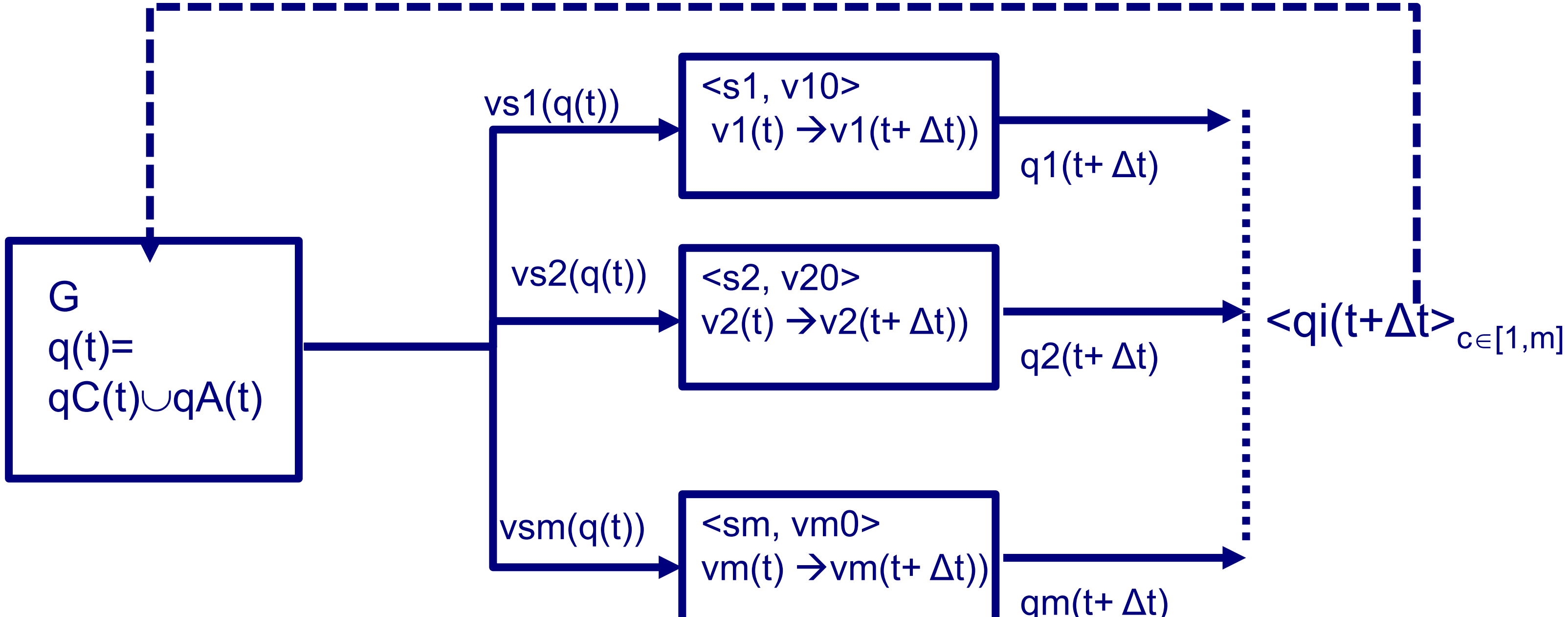


The arriving vehicle has priority over the ego vehicle that is giving way

The ego vehicle decides to cross based on the distance of the arriving vehicle and the enforced speed limit

Compositional Testing – Locality of Interaction

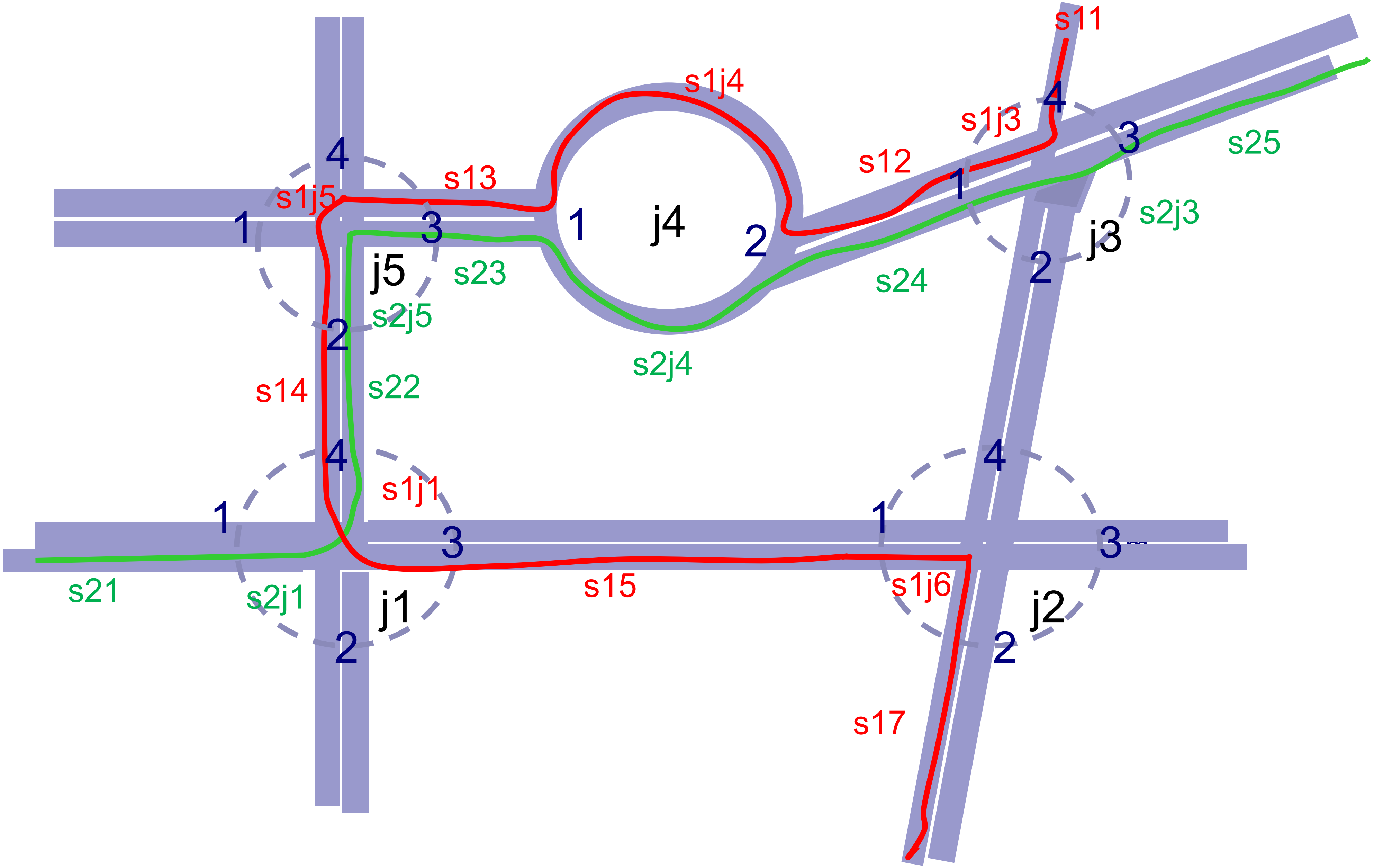




The vista of each vehicle is an abstraction of the relevant system state.

So instead of receiving all the state q(t) it receives only vistas.

Compositional Testing – Composition of Vistas



The route is a sequence of segments from roads, crossings and mergers that define corresponding types of vistas

For s1 we have

s1= s11 s1j3 (s41@j3)

s12 s1j4(s21@j4)

s13 s1j5 (s32@9j5)

s14 s1j1 (s43@j1)

s15 s1j6 (s12@j2)

s17

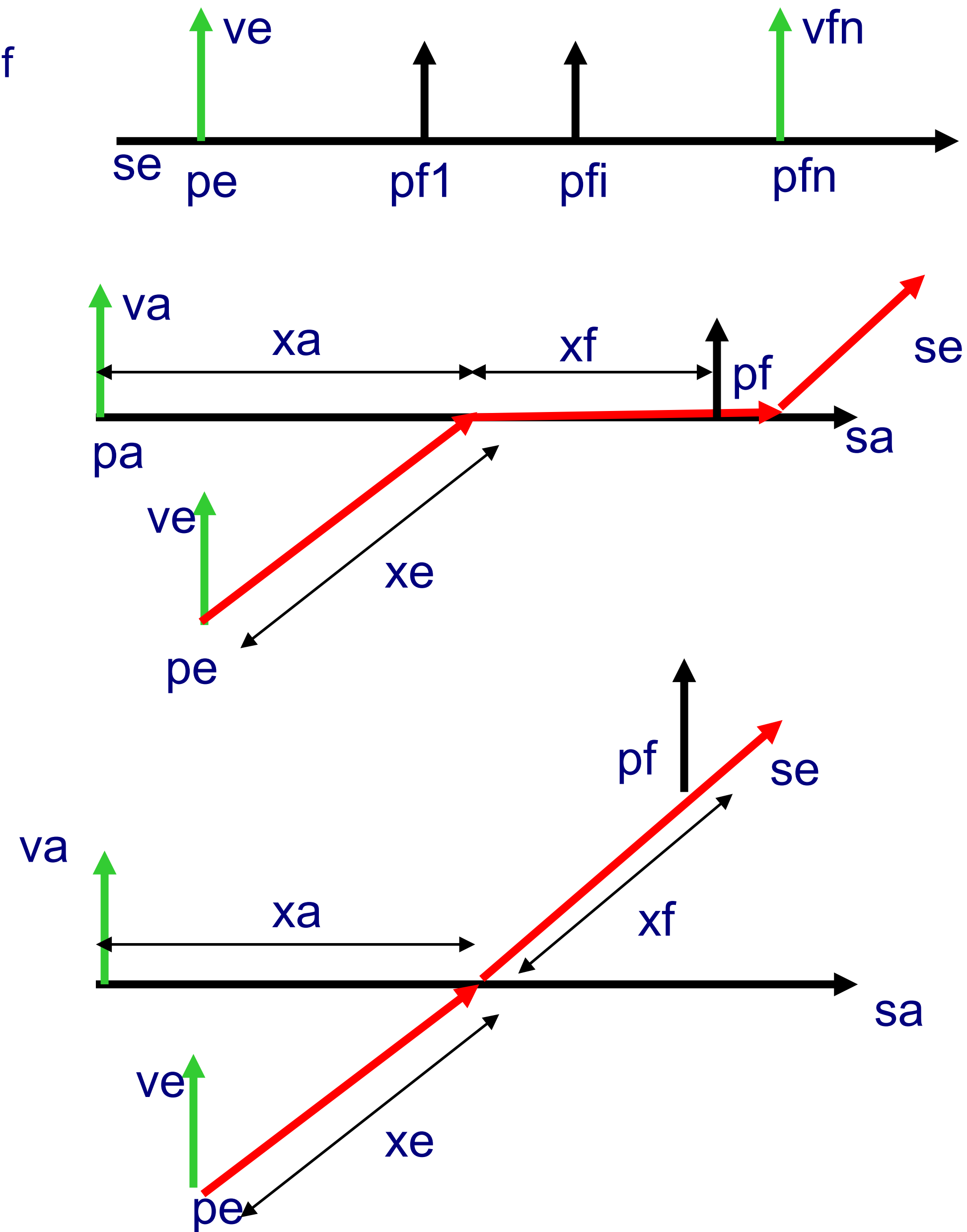
Compositional Testing – The Tree Basic Vista Types

A simple analysis shows that, as a vehicle moves, its autopilot reacts to inputs that are changes in the state of its environment, characterized by three different types of vistas.

1. road vistas: where there are no crossroads in the vehicle’s area of visibility, and the autopilot is tasked with taking into account the obstacles in its route ahead;

2. merging vistas: when the vehicle’s route joins a road or a lane where oncoming vehicles have a higher priority and it must therefore give way to these vehicles;

3. crossing vistas: where the vehicle’s route crosses a junction accessible to other vehicles, and therefore, the vehicle must comply with the traffic rules applicable in this context.



We focus on test cases for merging and crossing vistas trying to determine critical configurations of between the states of the ego vehicle the state of an arriving vehicle and the state of a front vehicle where

- The arriving vehicle moves at the maximal allowed speed limit vl
- The front vehicle is stopped

Overview

- ❑ Validation of ADS
- ❑ Compositional Testing
- ❑ Test Methodology
- ❑ Experimental Results
- ❑ Discussion

Test Methodology – Assumptions about Policies and Vehicle Dynamics

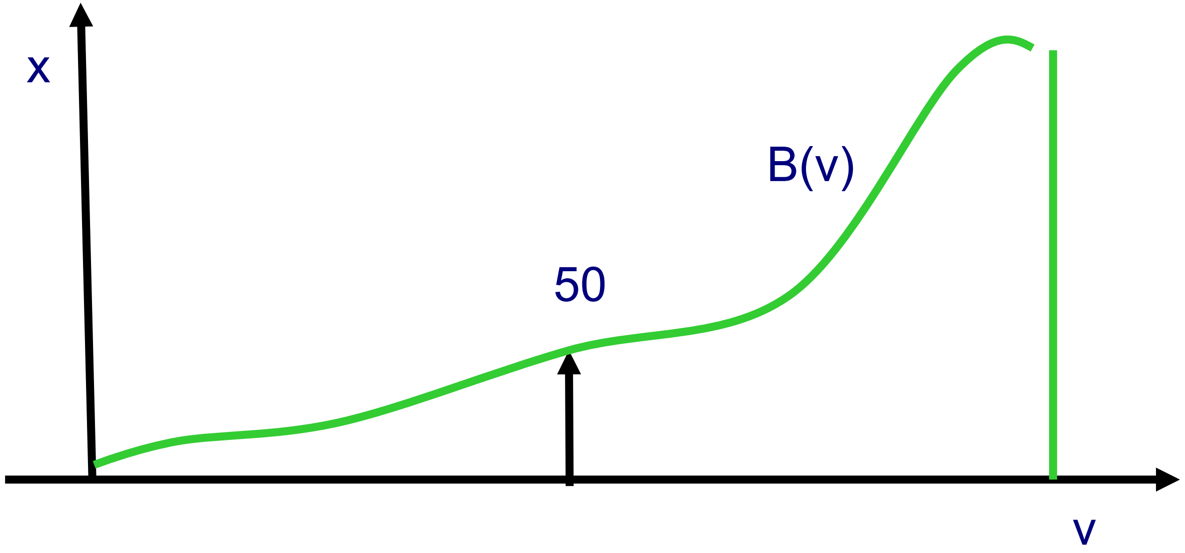
❑ The test should allow detecting failures under the following assumptions:

- The vehicles in the environment of the ego vehicle drive responsibly i.e. respect traffic rules such as speed limits, traffic lights etc.
- Test cases should be realistic: the initial values of the states of the vehicles involved in a vista should be such that
  - The free space of each involved vehicle is large enough to be able to drive safely.
  - If there is violation of a traffic rule or an accident, it should be possible to identify responsibilities i.e. this is due to the fact that some vehicles have not applied a feasible safety control policy.
- The test cases correspond to worst case situations (to be explained).

❑ To estimate feasible safe control policies we need to know the following A/D functions characterizing vehicle dynamics:

- The braking function B(v) that gives the required distance to brake from speed v to speed 0.
- The acceleration time function TA(v0,x) that gives the time taken to accelerate from speed v0 traveling a distance x.
- The acceleration speed function VA(v0,x) that gives the speed reached from v0 by accelerating for distance x.

We assume that the A/D functions are strict and monotonic.



Test Methodology – Caution vs. Progress

A vista describes a relationship between the ego vehicle and other objects that the ego vehicle can eventually change by acting adequately. We distinguish two possible attitudes of the ego vehicle:

- Caution where the ego vehicle acts safely without changing this relationship e.g. following a vehicle, approaching a merge, approaching a crossing.
- Progress where the ego vehicle acts to change this relationship e.g. overtaking a vehicle, merging into a main road, crossing an intersection

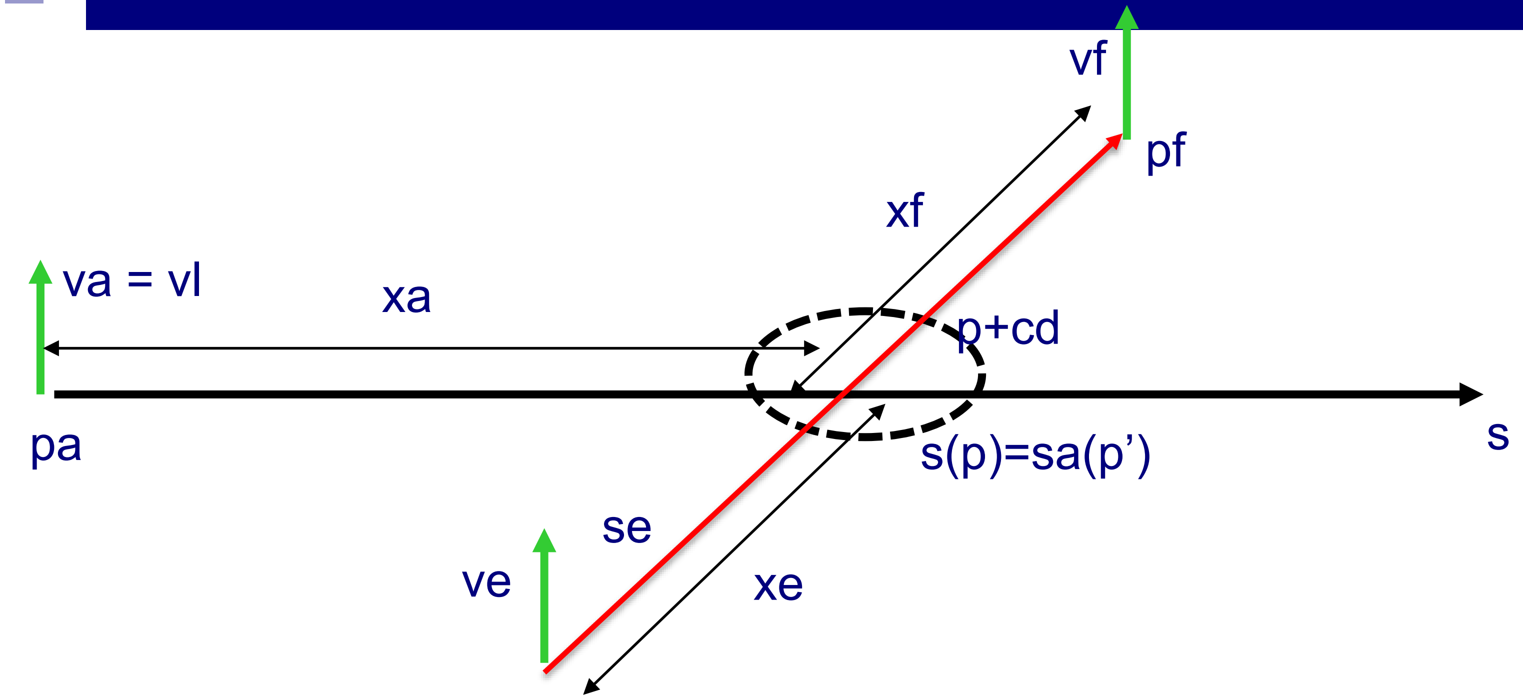
❑ Testing for caution: For each vista, determine the safe conditions for moving the ego vehicle while preserving its initial relationship with the other objects in the vista. Note that caution does not imply the obligation to move - an immobile vehicle remains cautious.

❑ Testing for progress: For each vista, determine the safe conditions for the ego vehicle to change the initial relationship with the other objects in the vista. Note that progress implies the obligation to move.

- Progress implies the ability to perform maneuvers and this allows optimizing road occupancy and avoiding bottlenecks;
- A vehicle can be cautious by applying very conservative control policies, for instance
  - move at low speed in a highway;
  - stop before a yield sign even if the priority road is clear;
  - not overtaking a slow vehicle in front, e.g. a truck, while this is permitted by the performance of the ego vehicle and the external lane is clear;
  - stop before a green light.

Note: The negation of a progress condition is an implicit caution condition that should be tested separately.

Test Methodology – Caution vs. Progress



Caution condition:

- As long as the ego vehicle cannot cross (not C1 or not C2), it can brake safely before reaching the critical region: B(ve) ≤ xe

Progress condition: The ego vehicle will cross if the following conditions hold

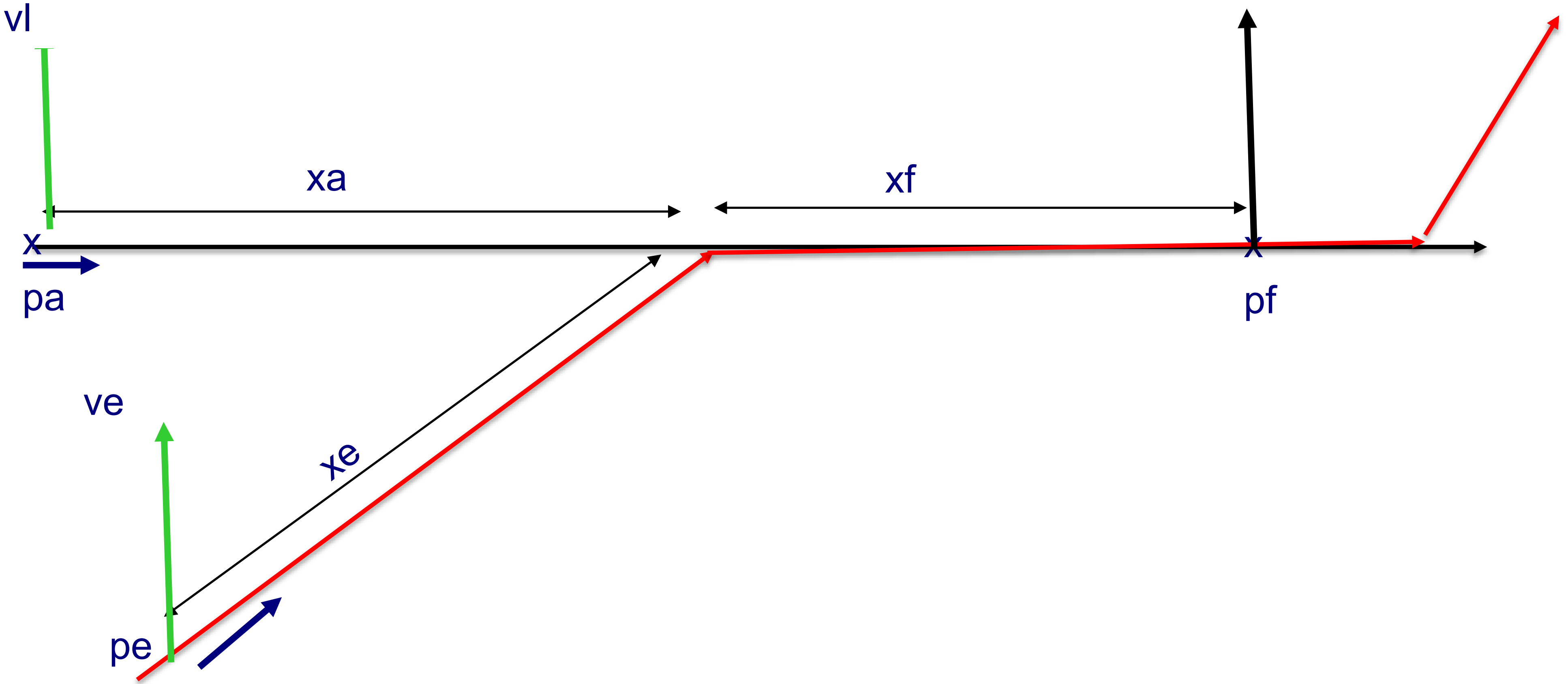
- **C1= xac ≤ xa,** where xac is a condition depending on xe, ve: an arriving vehicle has enough space to avoid collision traveling at the maximal allowed speed vl, d

- **C2= xfc≤ xf,** where xfc is a condition depending on xe, ve: the ego vehicle can keep a safe distance with respect to the first front vehicle

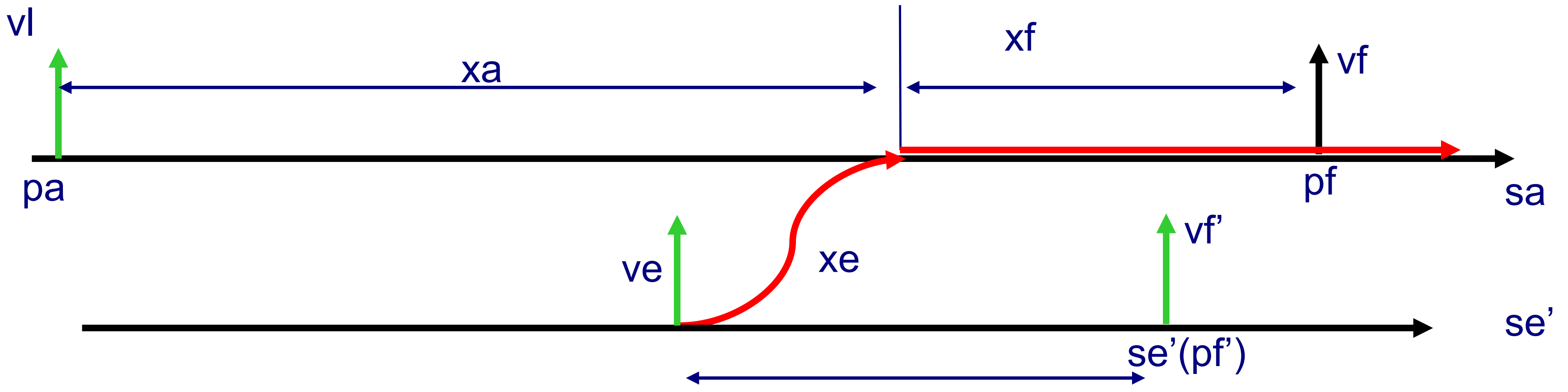
Test Methodology – Testing Merging Maneuver

TA(ve, xe) is the time to travel xe from ve by accelerating to reach speed ve'=VA(ve, xe)

- $C1 = vl.TA(ve, xe)+B(vl) \leq xa$ $\quad$ $C1 = ve \leq a\,[xa/vl - vl/2b]\,([(a+b)/b]^{1/2} - 1)^{-1}$.
- $C2 = B(VA(ve, xe)) \leq xf$ $\quad$ $C2 = ve \leq b\,[2\,xf/(a+b)]^{1/2}$
- Worst case: both C1 and C2 are applicable: for a given ve it determines a unique pair of critical values xac and xfc.



Test Methodology – Lane Change Vista



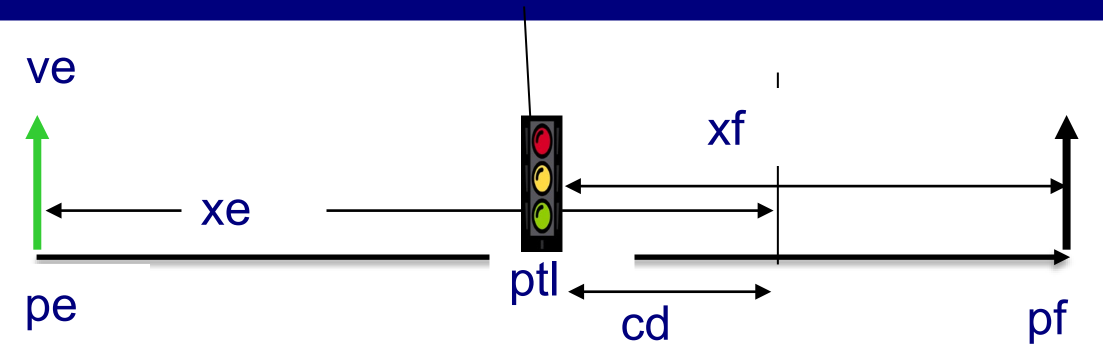
Caution condition :

- As long as the ego vehicle does not perform a lane change (not C1 or not C2), it is in a safe braking distance with respect to the front vehicle f' i.e. B(ve) ≤ pf'-pe.
- The speed of the ego vehicle ve is such that ve≥ vf'.

Progress condition: The ego vehicle will move to merge by accelerating if the following conditions hold:

- C1= vl. ve/xe +B(vl) ≤ xa i.e. an arriving vehicle has enough space to avoid collision traveling at the maximal allowed speed vl,
- C2 = B(ve) ≤ xf i.e., the ego vehicle can keep a safe distance with respect to the first front vehicle on the external lane ,

Test Methodology – Traffic Lights Vista



❑ We consider a crossing with

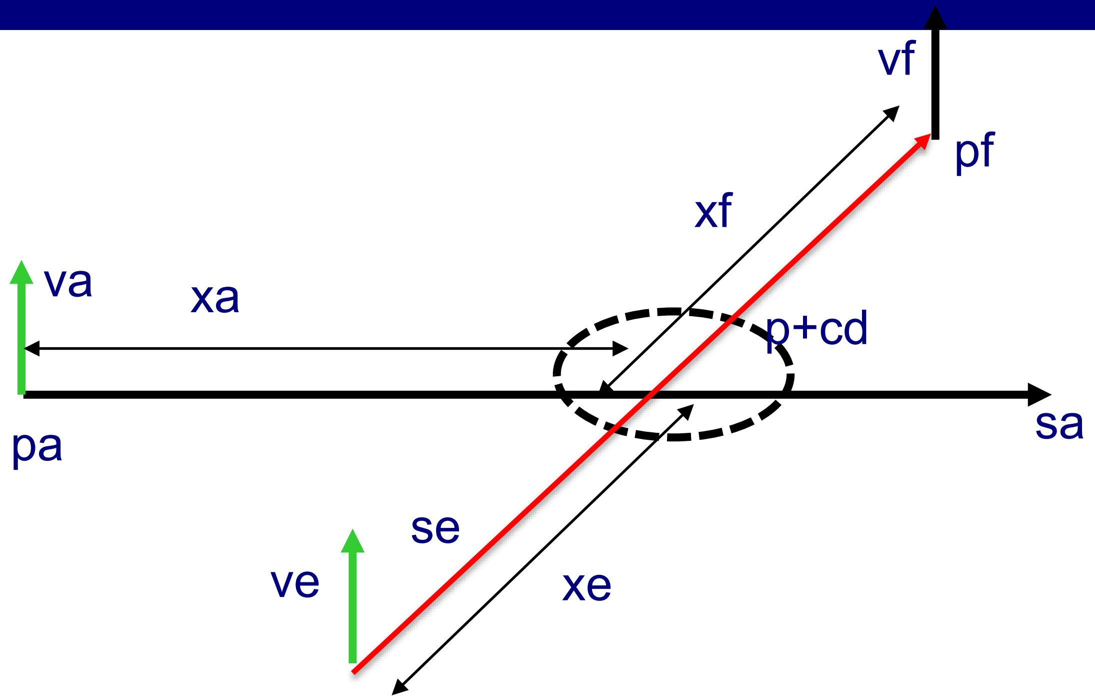
- a critical section of length cd, guarded by traffic lights
- ty is the duration of the yellow light and tar the duration of the “all red” phase.

Caution condition : The ego vehicle has a speed ve such that B(ve) ≤ xe.

Progress condition:

- C1: The ego vehicle will cross by accelerating from distance dr to reach a speed ve’ satisfying the conditions
  - TA(ve, xe) ≤ ty guarantees no running a red light
  - TA(ve, xe+cd) ≤ ty+tar guarantees exiting before the end of the all red phase
- C2: B(VA(ve, xe+cd)) ≤ xf guarantees no collision with the front vehicle

Test Methodology – Crossing Vista

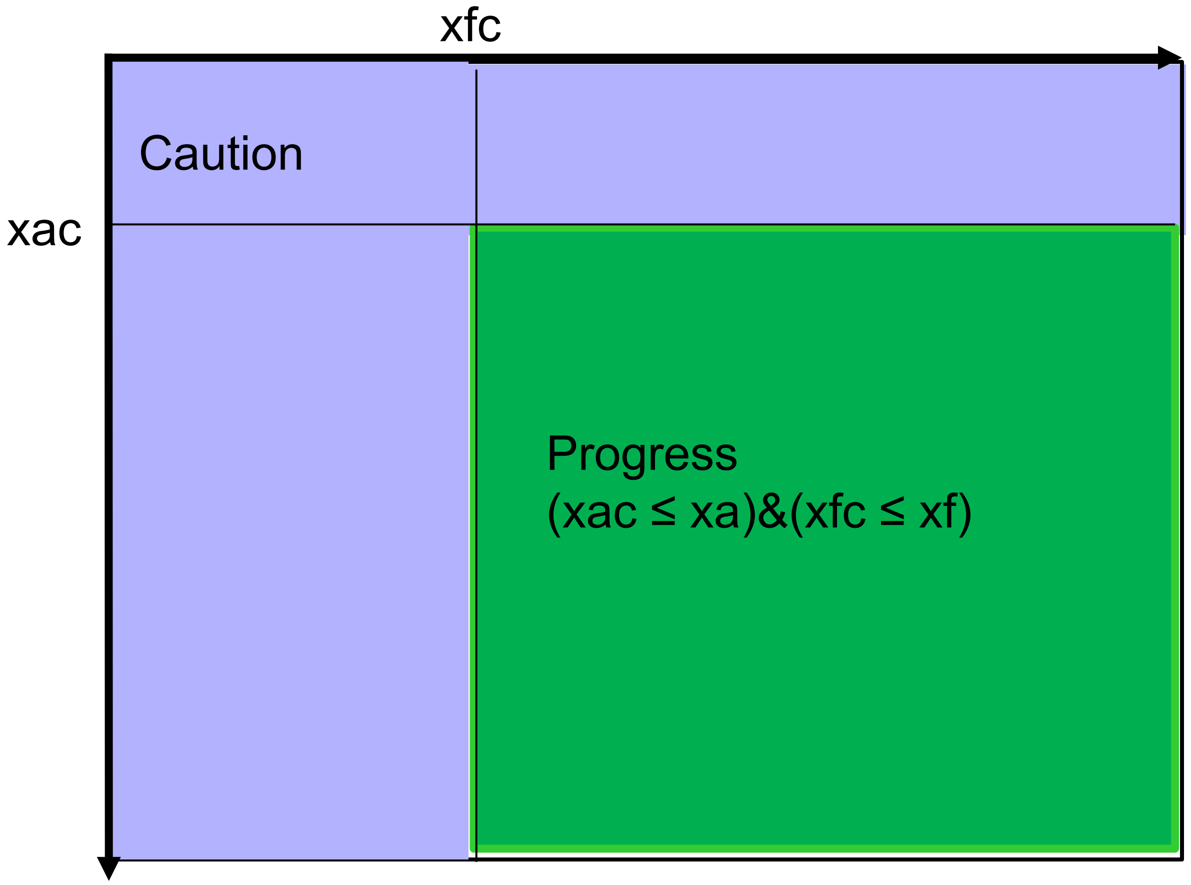


Safety condition for crossing: the vehicle can brake safely before the crossing point: xe ≤B(ve).

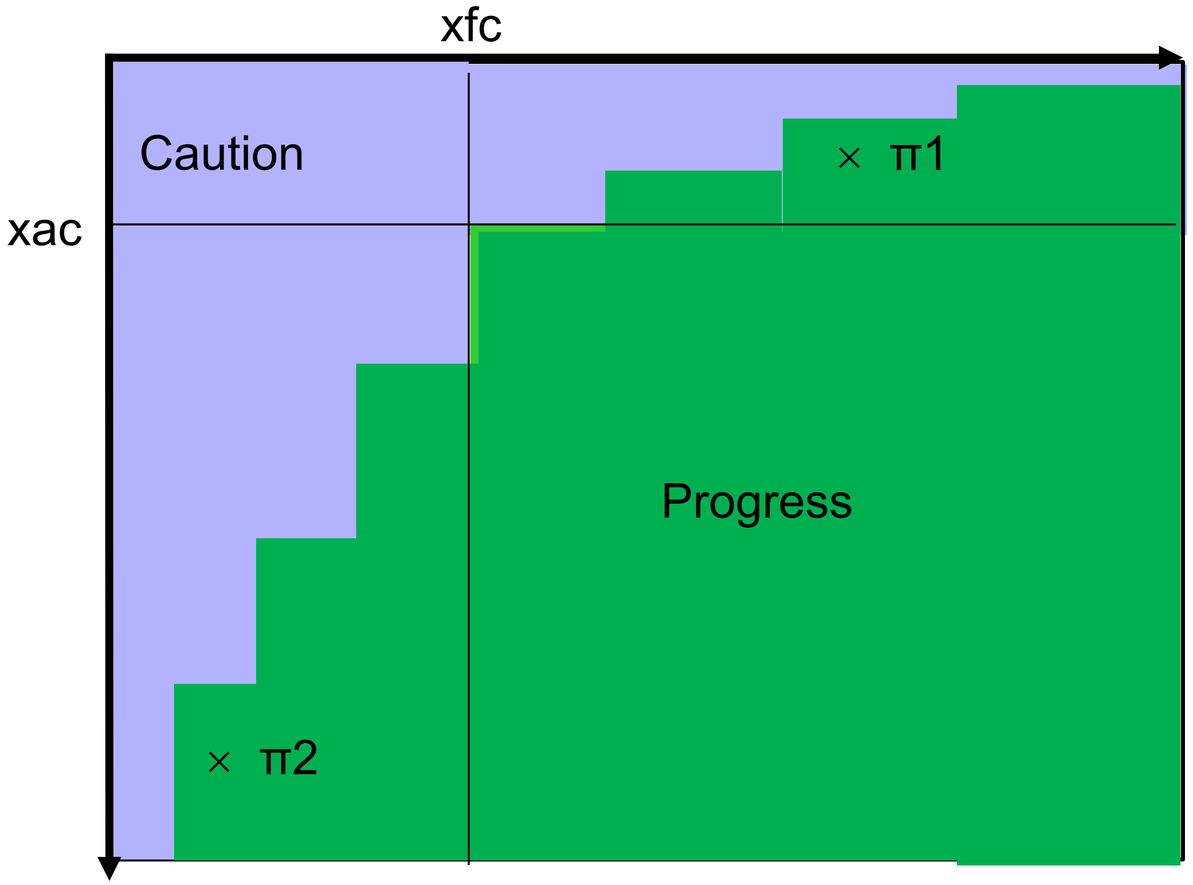
Progress condition: The ego vehicle will cross if the following conditions hold:

- C1 = vl.TA(ve, xe) +B(vl) ≤ xa i.e., an arriving vehicle has enough space to avoid collision traveling at the maximal allowed speed vl.
- C2= : B(VA(ve, xe+cd)) ≤ xft he ego vehicle can keep a safe distance with respect to the first front vehicle

Decomposition of the Test Space



Caution/progress partition for worst case safe policies



Caution/progress partition for feasible safe policies

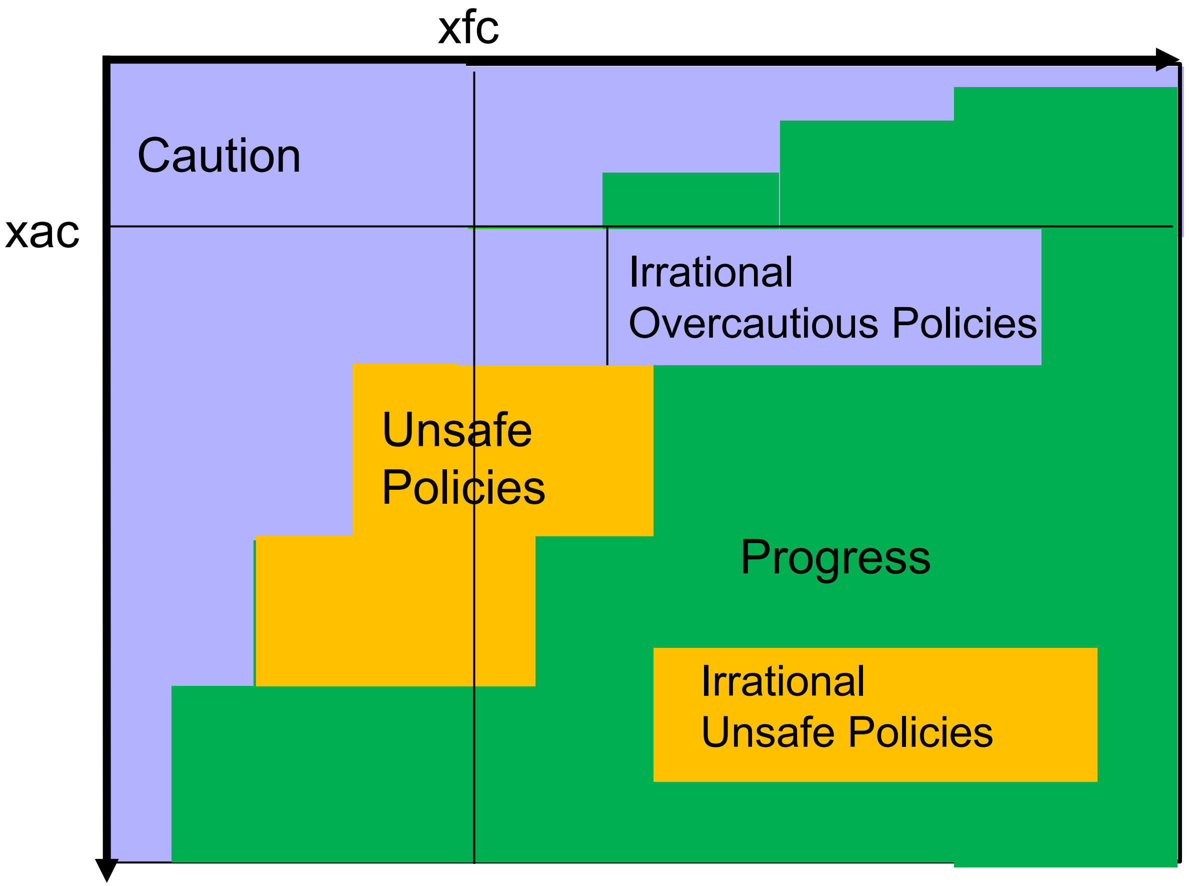
Note: The test space

- defines for given initial system state the function: Test Cases →Verdicts for the considered scenarios (end of the caution or progress phase)
- is different from the decision space that involves also the controllable variables

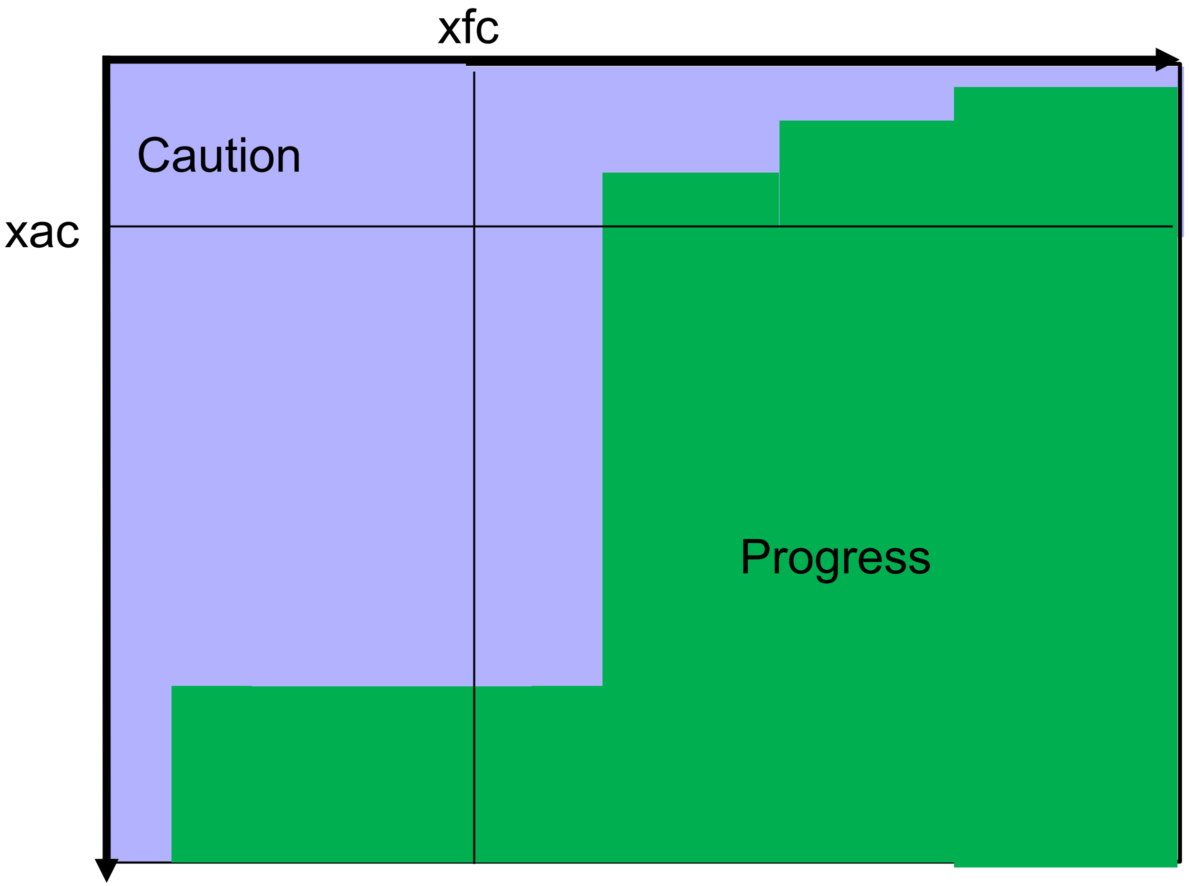
π1: the arriving vehicle decelerates in anticipation of the maneuver

π2: the ego vehicle does not fully accelerate to have enough space to stop before the front vehicle.

Decomposition of the Test Space (2)



Undesirable situations revealed by testing

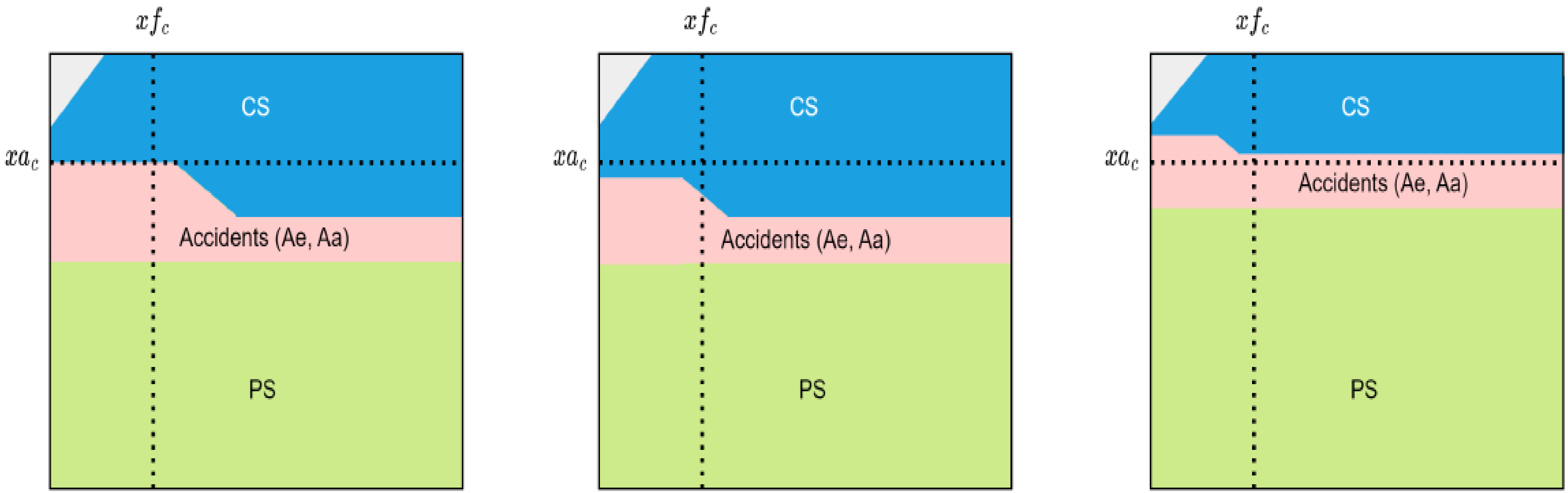


Corresponding feasible safe behavior of a rational controller

Overview

- ❑ Validation of ADS
- ❑ Compositional Testing
- ❑ Test Methodology
- ❑ Experimental Results
- ❑ Discussion

Experimental Results – Detailed Analysis of Apollo



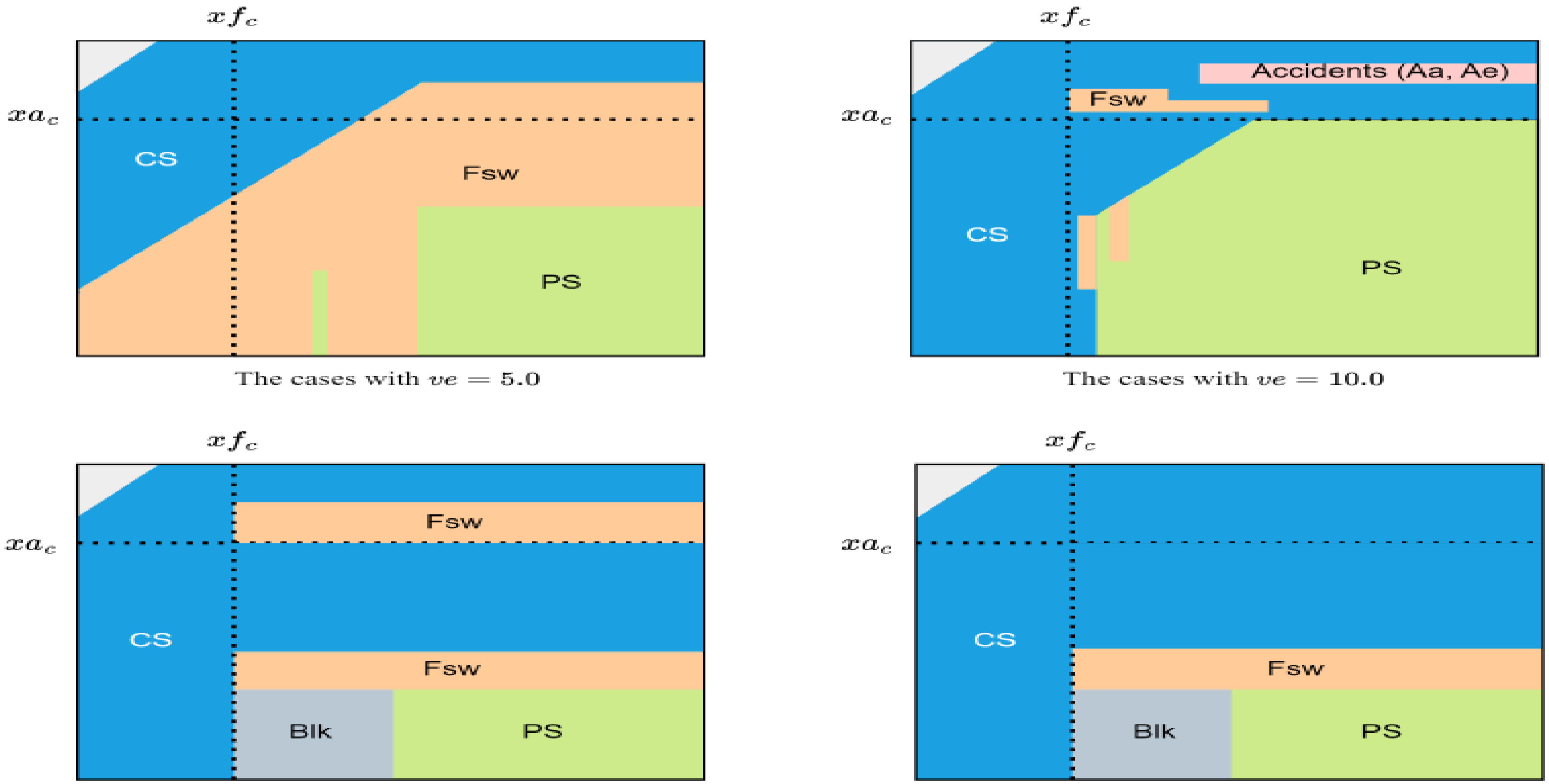
The cases with $ve = 0.0$ and $ve = 5.0$

The cases with $ve = 10.0$

The cases with $ve = 15.0$

Fig. 1: Merging test cases for Apollo Autopilot

Experimental Results – Detailed Analysis of Apollo



The cases with $ve = 5.0$

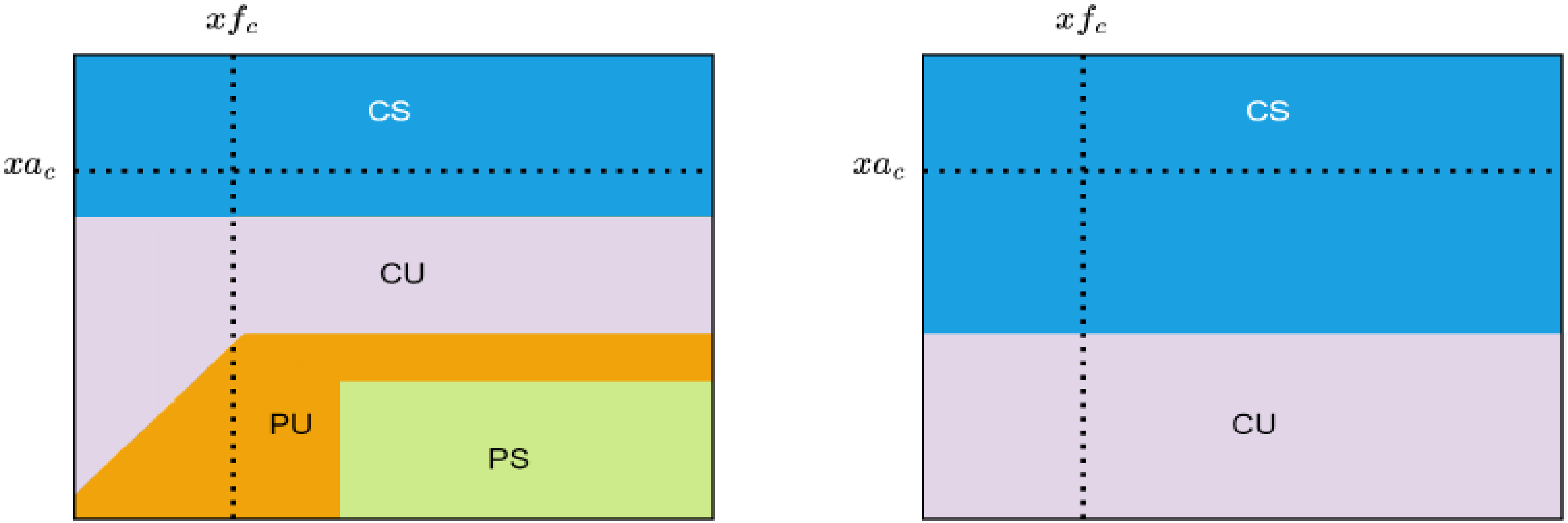
The cases with $ve = 10.0$

The cases with $ve = 15.0$

The cases with $ve = 20.0$

Fig. 2: Lane change test cases for Apollo Autopilot

Experimental Results – Detailed Analysis of Apollo



The cases with $ve = 0.0$ and $ve = 5.0$

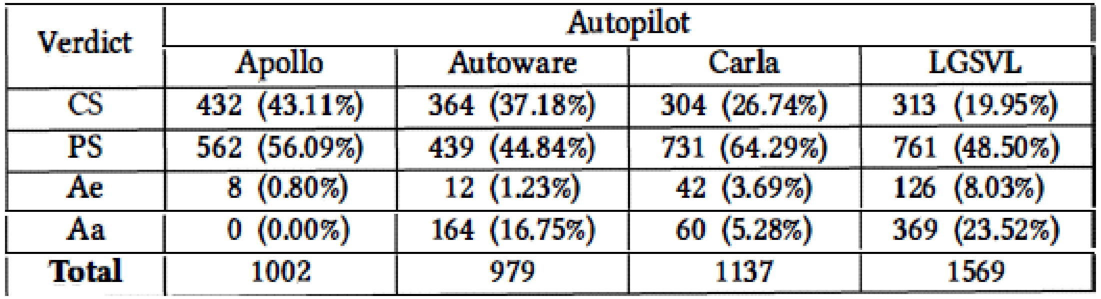
The cases with $ve = 10.0$ and $ve = 15.0$

Fig. 3: Yield-sign crossing test cases for Apollo Autopilot

Experimental Results – Effectiveness of the Method

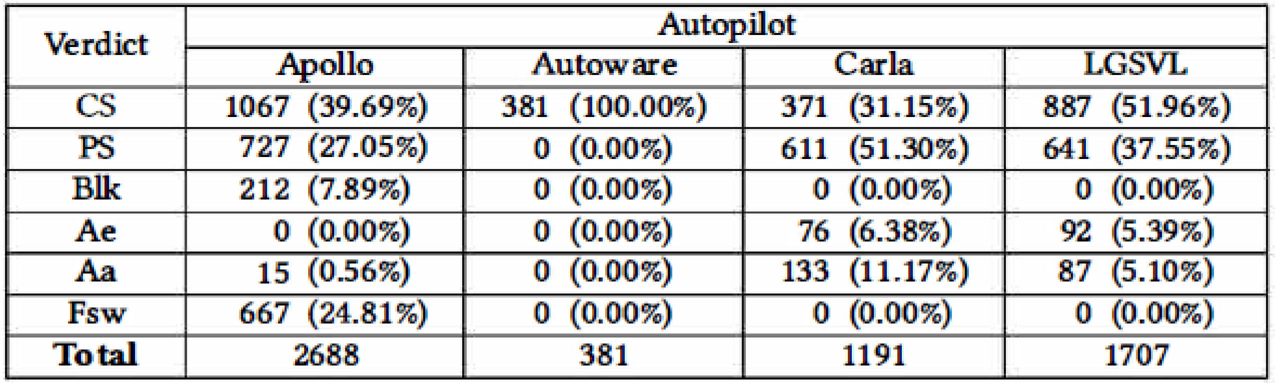
- The method reveals safety problems for all autopilots and all vista types, with the exception of the Autoware autopilot when changing lanes and the Carla autopilot when crossing traffic lights.
- For a total of 14506 test cases, 3962 various types of defects detected, which corresponds to 27.31 % of the test cases.

Table 26. Statistics on verdicts for merging scenarios



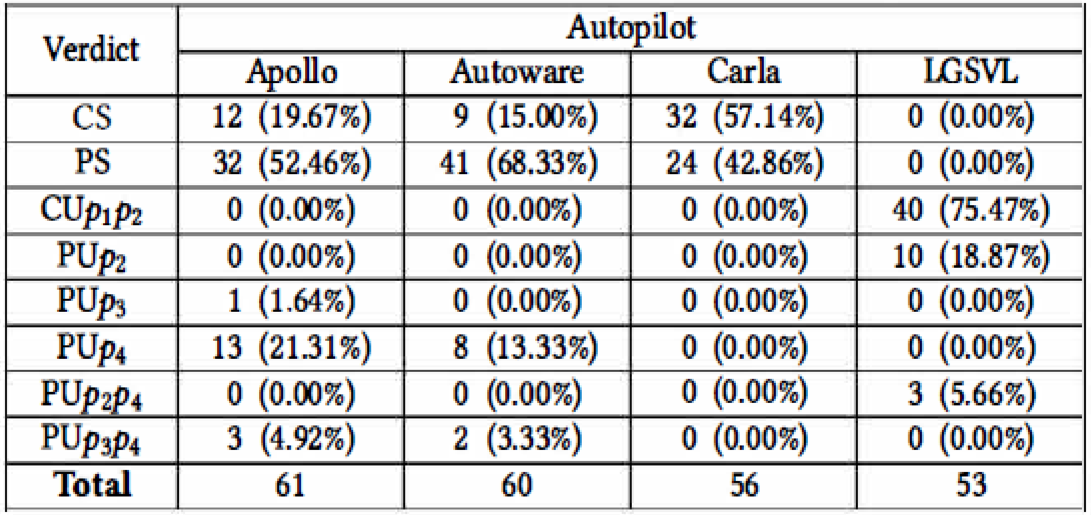
Verdict	Autopilot			
	Apollo	Autoware	Carla	LGSVL
CS	432 (43.11%)	364 (37.18%)	304 (26.74%)	313 (19.95%)
PS	562 (56.09%)	439 (44.84%)	731 (64.29%)	761 (48.50%)
Ae	8 (0.80%)	12 (1.23%)	42 (3.69%)	126 (8.03%)
Aa	0 (0.00%)	164 (16.75%)	60 (5.28%)	369 (23.52%)
Total	1002	979	1137	1569

Table 27. Statistics on verdicts for lane change scenarios



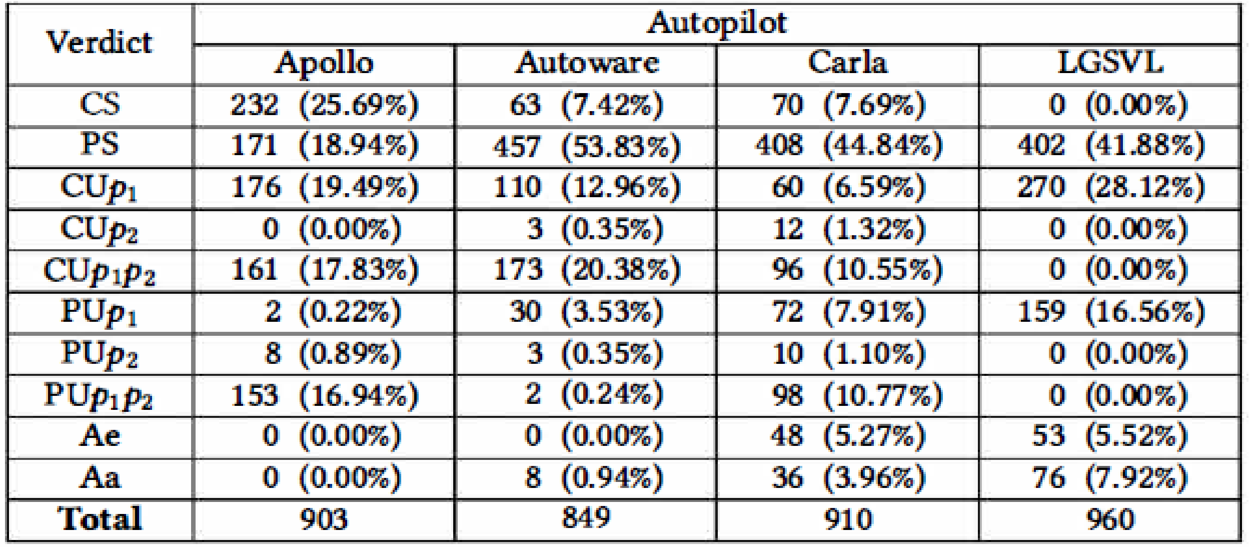
Verdict	Autopilot			
	Apollo	Autoware	Carla	LGSVL
CS	1067 (39.69%)	381 (100.00%)	371 (31.15%)	887 (51.96%)
PS	727 (27.05%)	0 (0.00%)	611 (51.30%)	641 (37.55%)
Blk	212 (7.89%)	0 (0.00%)	0 (0.00%)	0 (0.00%)
Ae	0 (0.00%)	0 (0.00%)	76 (6.38%)	92 (5.39%)
Aa	15 (0.56%)	0 (0.00%)	133 (11.17%)	87 (5.10%)
Fsw	667 (24.81%)	0 (0.00%)	0 (0.00%)	0 (0.00%)
Total	2688	381	1191	1707

Table 29. Statistics on verdicts for crossing with traffic light scenarios



Verdict	Autopilot			
	Apollo	Autoware	Carla	LGSVL
CS	12 (19.67%)	9 (15.00%)	32 (57.14%)	0 (0.00%)
PS	32 (52.46%)	41 (68.33%)	24 (42.86%)	0 (0.00%)
CUp_1p_2	0 (0.00%)	0 (0.00%)	0 (0.00%)	40 (75.47%)
PUp_2	0 (0.00%)	0 (0.00%)	0 (0.00%)	10 (18.87%)
PUp_3	1 (1.64%)	0 (0.00%)	0 (0.00%)	0 (0.00%)
PUp_4	13 (21.31%)	8 (13.33%)	0 (0.00%)	0 (0.00%)
PUp_2p_4	0 (0.00%)	0 (0.00%)	0 (0.00%)	3 (5.66%)
PUp_3p_4	3 (4.92%)	2 (3.33%)	0 (0.00%)	0 (0.00%)
Total	61	60	56	53

Table 28. Statistics on verdicts for crossing with yield sign scenarios



Verdict	Autopilot			
	Apollo	Autoware	Carla	LGSVL
CS	232 (25.69%)	63 (7.42%)	70 (7.69%)	0 (0.00%)
PS	171 (18.94%)	457 (53.83%)	408 (44.84%)	402 (41.88%)
CUp_1	176 (19.49%)	110 (12.96%)	60 (6.59%)	270 (28.12%)
CUp_2	0 (0.00%)	3 (0.35%)	12 (1.32%)	0 (0.00%)
CUp_1p_2	161 (17.83%)	173 (20.38%)	96 (10.55%)	0 (0.00%)
PUp_1	2 (0.22%)	30 (3.53%)	72 (7.91%)	159 (16.56%)
PUp_2	8 (0.89%)	3 (0.35%)	10 (1.10%)	0 (0.00%)
PUp_1p_2	153 (16.94%)	2 (0.24%)	98 (10.77%)	0 (0.00%)
Ae	0 (0.00%)	0 (0.00%)	48 (5.27%)	53 (5.52%)
Aa	0 (0.00%)	8 (0.94%)	36 (3.96%)	76 (7.92%)
Total	903	849	910	960

Overview

□ Validation of ADS

□ Compositional Testing

□ Test Methodology

□ Experimental Results

□ Discussion

Discussion – Key Findings

- The complexity of the test problem can be controlled by decomposing complex scenarios into sequences of simple types of scenarios for a limited number of traffic patterns and configurations of a small number of vehicles.
  - Simulating billions of miles without specifying how they relate to and cover real-life situations is not a convincing argument for safety.
  - Safety critical scenarios are rare and the probability of discovering such situations may be low for a car in simulation, but may become non-negligible for very large number of cars in real-life situations.

- The test method
  - enables detection of defects by means of a detailed analysis of vehicle dynamics, which determines critical situations when the autopilot switches from a cautious control to progress characterized by finely-tuned parameter combinations that are difficult to generate by random simulation.
  - differs from most work which focuses on simple scenarios, typically freeway driving. What's more, most simulators only allow you to control the ego vehicle, which limits the possibility of creating dangerous situations by controlling several vehicles.

Discussion – Key Findings (2)

- The analysis of the dynamic behavior of the autopilots reveals some surprising non realistic features including the following
  - Limited maximal speed limited even on a free road e.g. 25 Km/h for MILE,
  - Non-realistic acceleration and deceleration rates e.g. 12 m/s2
  - Non rational and non monotonic policies make systematic testing impossible as they do not allow the application of the worst case principle “Who can do more can do less”
    - This is fundamental in risk management wherein the planner, in planning for potential disasters, considers the most severe possible outcome that can reasonably be projected to occur in a given situation.
    - Unlike model-based autopilots, which can be made monotonic and rational by design, these properties are difficult to guarantee for data-driven autopilots, in particular because of non reproducibility of their behavior and adversarial examples.

THANK YOU