

An Ontology-Based Approach to Relax Traffic Regulation for Autonomous Vehicle Assistance

Philippe Morignot, Fawzi Nashashibi

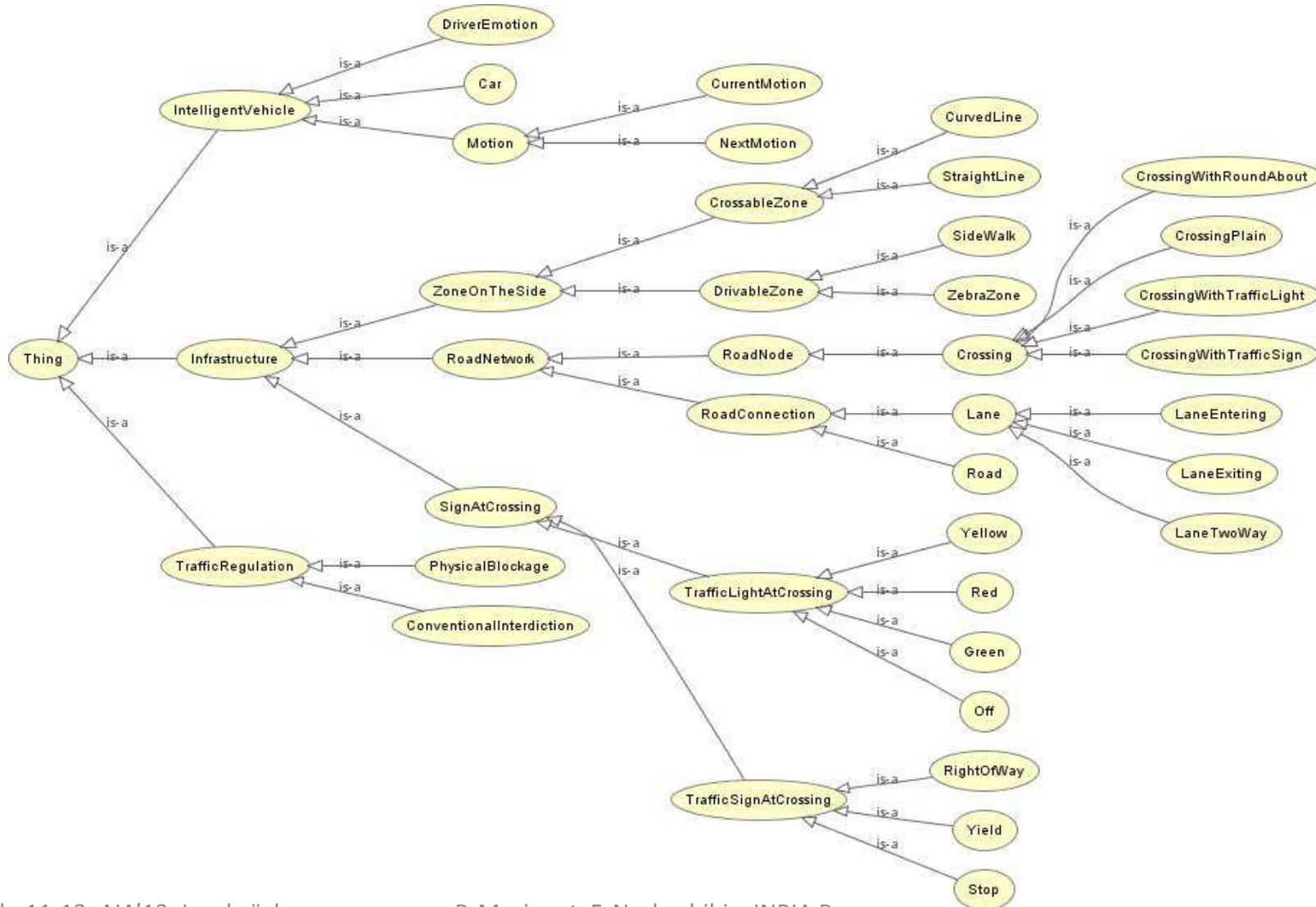
INRIA - Rocquencourt, France

Introduction



- « *Vehicles must circulate on roadways, except in case of absolute emergency* » (Vienna Convention on Road Traffic , 1968).
- Breaking traffic rules is forbidden, but tolerated in unusual situations, e.g., a truck unloading along a continuous line.

Ontology

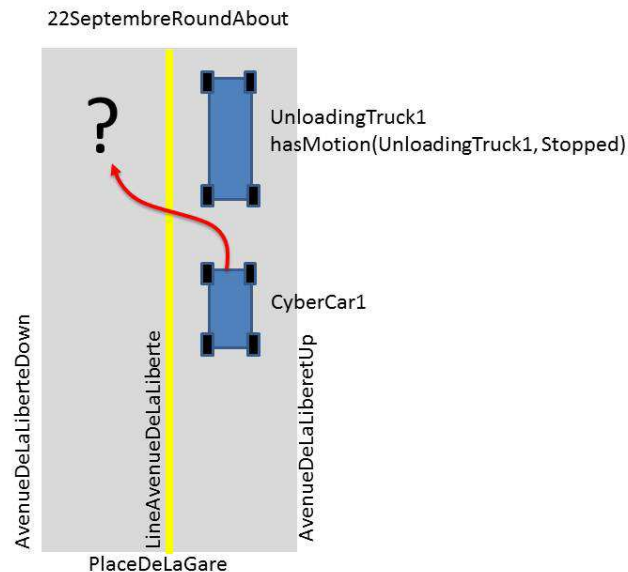


Inference rules

- Axioms (Tboxes) over situations (Aboxes), representing decision at time step n over $n+1$.
- Examples :
 - *Car(?a), Car(?b), CrossableZone(?s), Lane(?l1), Lane(?l2), hasState(?a, Impatient), isAfter(?a, ?b), hasBesides(?l1, ?s), hasBesides(?l2, ?s), hasMotion(?a, Stopped), isOn(?a, ?l1), isOn(?b, ?l1), DifferentFrom(?l1, ?l2), isIllegal(?l1, ?l2), isClear(?l2) -> isNextOn(?a, ?l2)*
 - *Car(?a), Car(?b), isBefore(?a, ?b), hasMotion(?a, Stopped) -> hasMotion(?b, Stopped)*
 - *Car(?a), Car(?b), isBefore(?a, ?b), hasNextMotion(?a, Forward) -> hasNextMotion(?b, Forward)*

Implementation & results

- PROTEGE using DL reasoner PELLET (SWRL).
- Example of inferences:



With hasState(CyberCar1, Impatient) :

hasMotion(CyberCar1, Stopped)
isAfter(CyberCar1, UnloadingTruck1)
hasNextMotion(CyberCar1, Forward)
isNextOn(CyberCar1, AveLiberteDown)

With hasState(CyberCar1, Relax) :

hasMotion(CyberCar1, Stopped)
isAfter(CyberCar1, UnloadingTruck1)
hasNextMotion(CyberCar1, Stopped)

Conclusion

- Unusual / extreme situations in traffic.
- High level topological model of an automated car, other cars and the infrastructure.
- Ontology with inference rules in SWRL.
- Interesting performances.
- Future work:
 - Relating symbols to percepts.
 - Porting onto CyberCars.