

Developing a BDI-style Domain Specific Language for Accountability of Autonomous Vehicles

Motivation

Recently¹, one of Google's Autonomous Vehicles (AV) collided with a bus in California when driving autonomously. This is the first time Google acknowledges that its AV is liable. How should we decide when an AV is liable when a collision occurs? What if the AV obeys all traffic rules when the collision occurs? Should it be held liable? If not, how could we then design an AV which always obeys traffic laws? Could Artificial Intelligence (AI) solve this problem? If AI can, what elements of AI do we need to solve this problem?



Image taken from independent.co.uk

Description

Our idea to solve this problem is by using Knowledge Representation (KR) technique from AI. We wish to formalise traffic laws in BDI (Beliefs-Desires-Intentions) logic such that each traffic rule is precise and *verifiable*. This thesis focuses on implementing a BDI-style Domain Specific Language (DSL) for 'programming' an agent, like AgentSpeak(L)², and on devising a method for verifying this 'program' against the formalised traffic rules.

Tasks

1. Implement a prototypical language for programming a BDI-style agent in Isabelle/HOL³.
2. Devise a method for verifying the program in this language with respect to BDI logic.
3. Construct an agent and verify formally that it satisfies the traffic rules formalised in BDI-logic.

¹Google Inc. *Google Self-Driving Car Project Monthly Report February 2016*. Tech. rep. Alphabet, 2016.

²Anand S. Rao. "AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language". In: *Agents Breaking Away, Modelling Autonomous Agents in a Multi-Agent World, Proceedings*. Springer, 1996.

³Tobias Nipkow and Gerwin Klein. *Concrete Semantics - With Isabelle/HOL*. Springer, 2014.



Technische Universität München



Fakultät für Informatik

Lehrstuhl für Echtzeitsysteme und Robotik

Supervisor:

Prof. Dr.-Ing. Matthias Althoff

Advisor:

Albert Rizaldi, M.Sc.²

Research project:

PUMA

Type:

BA / MA

Research area:

Automated Driving, Formal Methods, Logics, Semantics of Programming Language, Theorem Proving

Programming language:

Any functional programming language

Required skills:

Interacting with Isabelle
Theorem Prover

Language:

English

Date of submission:

March 17th, 2016

For more information please contact us:

Phone: +49.89.289.18140

E-Mail: rizaldi@in.tum.de

Internet:

www6.in.tum.de/Main/Rizaldi