A Quest of Self-Explainability:
When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES’21

21. September 2021
Motivation

Why do you accelerate?
Because the crossing ahead is free!
This Talk

What do I not do?
- What types of explanations are necessary?
- How should an explanation be delivered?
- ...

Instead:
- How do I retrieve an explanation model from a system model?
- Vision: How can the explanation model be updated?
Part 1 (Contribution)

Retrieving an Explanation Model from a System Model
System Model

Crossing Controller for autonomous turn manoeuvres at intersections

- Some functional requirements have been proven:
  - Safety (collision freedom),
  - Liveness (something good happens) and
  - Fairness (no queue-jumping).

- Now: Introduce self-reflective/ non-functional capability to the Crossing Controller
  - Self-explainability (of the controller's actions)

- Semantic Model of Crossing Controller:
  - Automotive-Controlling Timed Automata (ACTA)

[BS19] Bischopink, C., S., M.: Verification of fair controllers for urban traffic manoeuvres at intersections (FMAS@FMWeek19)

A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres
Dr. Maike Schwammberger @RE4ES’21
System Model

Structure of (Automotive-Controlling) Timed Automata:

- Location 0
  - Guard / Action
    - $x > 10$ / $x := 0$
  - Or

- Location 0
  - Guard
    - $x > 10$
  - C
    - Action
      - $x := 0$
  - Location 1

Explanation type that we consider: “ACTION happened because of GUARD.”
Example: “$x := 0$ happened because of $x > 10$.“
Example: Crossing Controller

1st Phase:
Far away

View(E)

A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres
Dr. Maike Schwammberger @RE4ES’21
Example: Crossing Controller

Urban Multi-lane Spatial Logic:

\[ ca(E) \equiv re(E) \land free^{<dc} \land \neg\langle cs \rangle \land cs \]

(“Crossing ahead check”)


A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES'21
Example: Crossing Controller

2nd Phase: Crossing ahead

Claim
("Setting the turn signal")

View(E)

POTENTIAL COLLISION!


A Quest of Self-Explainability: When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES’21
Example: Crossing Controller

2nd Phase:
Crossing ahead

View(E)

0 1 2 3
V iew(E)


A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres
Dr. Maike Schwammberger @RE4ES'21
Example: Crossing Controller

2nd Phase:
Crossing ahead

Urban Multi-lane Spatial Logic:
\[ \text{ca}(E) \equiv \text{re}(E) \triangleright \text{free} < \text{dc} \land \neg \langle \text{cs} \rangle \triangleright \text{cs} \]


A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES'21
Example: Crossing Controller

3nd Phase: On crossing

View(E)

Reservation


A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES'21
Communicating Priorities

**Fairness:** *No car has to wait unreasonably long in front of an intersection.*

**Approach:**
- Send priority on arriving at intersection ("prio! (D, 10)"")
- Helper Controllers determine whether priority is large enough
  - Own helper (e.g. B) determines own priority is too small: ("withdraw! (B)"")
  - Other helper (e.g. A) determines D's priority is too small: ("no! (D)"")

---

[BS19] Bischopink, C., S., M.: *Verification of fair controllers for urban traffic manoeuvres at intersections (FMAS@FMWeek19)*

A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES'21
Crossing Controller

Phases:

1. **STAGE 1:** PHASE: FAR AWAY

2. **STAGE 2:** PHASE: CROSSING AHEAD

3. **STAGE 3:** PHASE: ON CROSSING

Crossing Controller (Overview):

**FAR AWAY**

- Claim crossing path
- Check for potential collisions
- Check priorities
- Priority + no potential collision: reserve crossing segments

**ON CROSSING**

- Finished manoeuvre

**CROSSING AHEAD**

- Crossing ahead
Crossing Controller

- **Focus:** 2nd Phase (crossing ahead), **action** “withdraw claim”
- **Three possible types of transitions:**
  1. Message “withdraw[ego]?” received
  2. Message “no[ego]?” received
  3. Potential collision detected (after $t$ time) (“exists (c: carid_t) pc(c)”) (**cause 1**)
  ⇒ “no” and “withdraw” sent by helper controller
  ⇒ Identify guards behind these messages in helper controller
Helper Controller

- **withdraw**: 
  - “Other has intersecting claim (pthcc(d))” and “Priority of other is larger” (cause 2)
  - “Other has no int. claim (!pthcc(d))” and “Priority of other is significantly (s) larger” (cause 3)
- **no**: Similar (“inverse”) to withdraw
- **no**: “Other car already on crossing (pthcoll(c))” (cause 4)

[Bis19] Bischopink, C., S., M.: *Verification of fair controllers for urban traffic manoeuvres at intersections (FMAS@FMWeek19)*

A Quest of Self-Explainability: — When Causal Diagrams meet Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES’21
Explanation Model

Causal Diagram related to Crossing Controller:

- "claim crossing path" $cc(E)$
  - because "a crossing is ahead" $ca(E)$
  - because "no potential collision exists" $\neg \exists c: pc(c)$
  - because "another car has intersecting claim" $pthcc(c)$

- "reserve crossing path" $rc(E)$
  - because "another car is on crossing" $pthcoll(c)$

- "withdraw claim" $wdcc(E)$
  - because "potential collision exists" $\exists c: pc(c)$
  - because "another car has higher priority" $prior(c) > prior(E)$
  - because "another car has no claim" $!pthcc(c)$

- "withdraw reservation" $wrdc(E)$
  - because "another car has a significantly higher priority" $prior(c) > prior(E) + s$
  - because "manoeuvre finished after $t_{cr}$ time" $x \geq t_{cr}$
Self-Explainability

Follow the MAB-EX Framework for Self-Explainability:

**Monitoring:** Observe system (e.g. with Observer Automata)

**Analysis:** Detect need for explanation (e.g. transition was triggered)

**Build Explanation:** Extract explanation path from explanation model

**Explain:** Give explanation to recipient (user, other car,...)


A Quest of Self-Explainability: — When Causal Diagrams meet, Autonomous Urban Traffic Manoeuvres

Dr. Maike Schwammberger @RE4ES'21
Example Explanation

Extracted explanation path:

\[ \text{because}(wd \ cc(E), \text{and} ( \neg \text{pthcc}(c), \text{prior}(c) > \text{prior}(E) + s)) , \]

Explanation:

“Car E did withdraw claim, because another car has no claim but a significantly higher priority.”
Summary Part 1

- Extraction of Explanation Model from a System Model

- Answer to the “Build Explanation” Phase of MAB-EX Framework

- Universality of Semantic Model “ACTA”: Not only self-explainability for Crossing Controller
Open Questions (Part 1)

– Check technical completeness of explanation model
  – E.g. is it enough to connect guards/ causes with actions?
  – What are the requirements for an explanation model?

– Automatically extract explanation models for (automotive-controlling) timed automata

– What is provable about explanations?
  – E.g. correctness or completeness of explanation model?

– Improve presentation of explanations (e.g. through HMI methods)

– Examine the explanation model wrt actually needed explanation types
  – Technical explanation model vs. non-technical explanation model?
Part 2 (Vision)

Dynamic Updates of Explanation Models
Explanation Model Updates

- System model needs to be updated at run-time (i.e. after it was sold)
  ⇒ Different approaches for this exist (e.g. component-wise updates)
- Easy: An updated system model leads to an updated explanation model
- But what if the explanation model needs to be updated without a system model update?

Part 1: Contribution

System Model

Part 2: Vision

Update

Explanation Model
Example

- Crossing Controller does not distinguish between different prioritised vehicles
  - But: Passenger might be interested which type of vehicle gets right of way (e.g. emergency vehicle)
  - Refinement of explanation model needed, but not of system model

\[
\text{“claim crossing path” } \quad \text{cc}(E)
\]

\[
\text{“reserve crossing path” } \quad \text{rc}(E)
\]

\[
\text{“withdraw claim” } \quad \text{wd cc}(E)
\]

\[
\text{“withdraw reservation” } \quad \text{wd rc}(E)
\]

\[
\text{“a crossing is ahead” } \quad \text{ca}(E)
\]

\[
\text{“no potential collision exists” } \quad \neg \exists c: \text{pc}(c)
\]

\[
\text{“another car is on crossing” } \quad pthcoll(c)
\]

\[
\text{“potential collision exists” } \quad \exists c: \text{pc}(c)
\]

\[
\text{“manoeuvre finished after } t_{cr} \text{ time” } \quad x \geq t_{cr}
\]

\[
\text{“another car has an intersecting claim” } \quad pthcc(c)
\]

\[
\text{“another car has higher priority” } \quad \text{prior}(c) > \text{prior}(E)
\]

\[
\text{“another car has no claim” } \quad !\text{pthcc}(c)
\]

\[
\text{“another car has a significantly higher priority” } \quad \text{prior}(c) > \text{prior}(E) + s
\]

\[
<\text{type}> \quad \text{emergency vehicle}
\]

\[
<\text{type}> \quad \text{normal vehicle}
\]
Vision

Research Question:

- How do we identify that a run-time update of the explanation model is needed?

Possible Direction:

- Run-time/ Dynamic Requirements Engineering
  - E.g. after an unsatisfactory/ incomplete explanation was provided
  - With methods of dynamic RE, refinement of nodes in explanation model (?)
  - Benefit: Explanation model can be tailored to specific user
Open Questions

Part 2:

- How is the need for an update of the explanation model detected?
- What types of updates for the explanation model are necessary?
Open Questions

Part 2:
- How is the need for an update of the explanation model detected?
- What types of updates for the explanation model are necessary?

Part 1:
- Check technical completeness of explanation model
  - E.g. is it enough to connect guards/ causes with actions?
  - What are the requirements for an explanation model?
- Automatically extract explanation models for (automotive-controlling) timed automata
- What is provable about explanations?
  - E.g. correctness or completeness of explanation model?
- Improve presentation of explanations (e.g. through HMI methods)
- Examine the explanation model wrt actually needed explanation types
  - Technical explanation model vs. non-technical explanation model?


