Tutorial on Distributed Knowledge Graphs for the WoT, Pt IX: Link Following, Planning, Model Checking

Tobias Käfer (KIT) and Andreas Harth (FAU)

Tutorial @ 10th International Conference on the Internet of Things (IoT), 2020

Outlook

- Read-Write Linked Data (or the equivalent Web of Things technologies) serve as environment on which user agents operate
- With declarative specifications of agent behaviour, many exciting directions for further research can be pursued:
  - “Service descriptions” for Read-Write Linked Data, in particular related to following links
  - Planning and model checking
  - Reasoning about the behaviour of single agents and groups of agents
  - Supporting users to specify agent behaviour

Reasoning

- Given RDF (or more generally graphs) as basis for knowledge representation, the next logical step would be to apply the semantics of RDFS and OWL (the Web Ontology Language) to world state
- Reasoning on RDF is a well-studied research field

- Related to reasoning is the choice of modelling dynamics, i.e., the difference between state representations and event representations

Link Following

- Link following is a tenet of (decentralised) web architecture
- Hypermedia as the Engine of Application State (HATEOAS) in Representational State Transfer (REST) architectures

- Functional descriptions (e.g., OWL-S) and affordance descriptions (e.g., WoT) could be modelled as hyperlinks

Artificial Intelligence Planning

- Given functional descriptions or affordance descriptions, AI planning algorithms could be applied
- Given a goal, the agent selects the “best” (according to some metrics: fastest, cheapest…) plan
- We already know how to execute agent plans (see Tobias’ presentation)

Model Checking

- Given functional descriptions or affordance descriptions, model checking algorithms could be applied
- Given a behaviour specification (a plan) in declarative form, check for safety and liveness conditions
- Model checking is important in systems which can harm people (many IoT systems with actuators can)

Many Agents

- We currently have only presented methods for specifying single agents
- Deploying many agents that work concurrently presents new challenges
- The web supports basic mechanisms for concurrency (conditional updates via Etag headers)

User Interfaces

- A declarative way to specify agent behaviour (plans) implies a clean syntax and semantics
- A clean syntax and semantics helps in designing consistent graphical user interfaces
- A clean syntax and semantics is also needed for conversational user interfaces

Platforms

- The presented approach, based on web architecture, allows for a decentralised implementation of the Internet of Things
- We can use similar technologies for a decentralised Internet of Things as the Social Linked Data (Solid) project does for decentralised social networking (https://solidproject.org/)

- Without the need for IoT platforms controlled by large companies