Reaching a common agreement discourse universe on Multi-Agent Planning

HAIS'10

Alejandro Torreño
Eva Onaindia
Oscar Sapena

Grupo de Tecnología Informática
Inteligencia Artificial

MOTIVATION

- Multi-Agent Planning (MAP) model with heterogeneous agents:
  - Different visions of the environment
  - Private goals
  - Different ontologies
- Coordination through negotiation
- A common agreement discourse universe (ADU) is required to negotiate

MULTI-AGENT PLANNING

- Elements of a MAP task:
  - A description of the initial state.
  - A set of global goals.
  - Two or more agents, each one of them having:
    - A set of actions or abilities.
    - A set of private goals.
- To solve the task, agents should cooperate to find a sequence of actions (plan) that, applied to the initial state, achieves the global and private goals.

AGENTS' INFORMATION MODEL

- At start, planning agents receive a *PDDL2.1*-based information model.
- Some aspects of the agents' models may differ:
  - Operators (actions):
    - Agents can have different abilities, or
    - Different representations of the same abilities.
  - Objects:
    - Agents can have different internal representations of the domain's objects.

DEFINING A COMMON INFORMATION MODEL

- Homogeneous representations of the objects and abilities are required for the agents to negotiate.
- We define a new information layer that:
  - Gives agents a shared ontology.
  - Is coherent with the agents' local models:
    - Agents will manage both models simultaneously.
    - Information can be migrated between models.
  - To build the model, we use a set of mechanisms:
    - Modeling techniques (design guidelines).
    - *PDDL* language extensions.

MODELING TECHNIQUES

- Aimed to define a common model that:
  - Homogenizes the domain descriptions.
  - Includes the simplest description of the objects and operators among the local models.
- Classification:
  - Generalization:
    - Objects in the common model are built as groupings of local objects.
  - Detail reduction:
    - Simplifies the domain description by discarding information from the local models.

OBJECTS GENERALIZATION

- Objects in the common layer are defined as the composition of local objects.
- We distinguish three ways to group objects:
  - Direct mapping:
    - An object at the common level is defined as it is in a local model.
  - Simple grouping:
    - Defines common objects as the composition of several instances of a local object.
  - Multiple grouping:
    - Groups several instances of different local objects into a common-model object.

OPERATORS GENERALIZATION

- Operators on the common model have also to be coherent with their local counterparts.
- We distingush two ways of defining operators:
  - Direct mapping:
    - A local operator is included in the common model as it is, replacing only its parameters.
  - Hierarchical network:
    - Common operators correspond to sequences of local ones.

DETAIL REDUCTION

- Some elements of the local models can be discarded from the common one.
- These elements are considered to be irrelevant from a common perspective.
- An object or an operator is irrelevant if it does not appear in all the local models.

PLANNING LANGUAGE EXTENSIONS

- The extensions are used to translate information between the common layer and the underlying local models.
- Addition of a *mapping* section:
  - Includes several *implies* sentences, that associate common and local predicates.
  - Each *implies* sentence acts as a double implication:
    - common-predicate $\Leftrightarrow$ local-predicate

EXAMPLE (I)

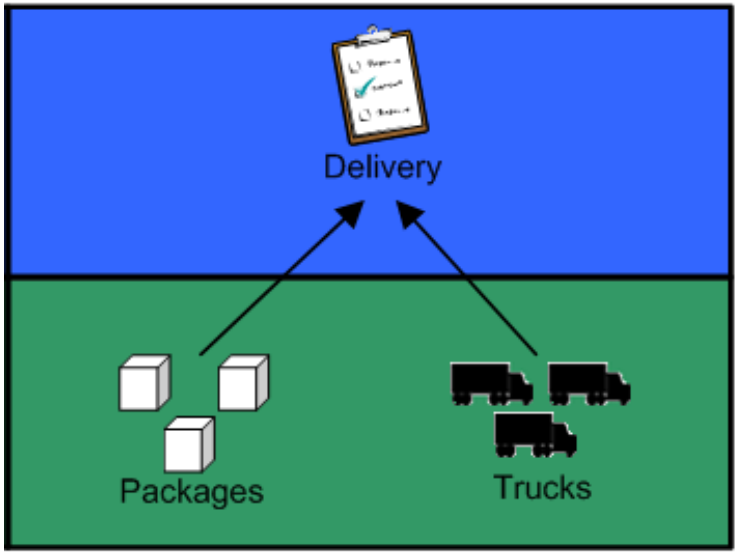
- Transportation domain.
- Local model:
  - Objects:
    - *Truck (T1)*, *Package (P1)*, *City (Madrid, Barcelona, Valencia)*.
  - Operators:
    - Load: Loads a *Package* into a *Truck*.
    - Unload: Unloads a *Package* from a *Truck*.
    - Drive: Drives a *Truck* from a *City* to another.

EXAMPLE (II)

- Global (common) model defined:
  - Objects:
    - *Package (P1)*: Same as in the local model.
    - *Delivery (D1)*: Represent a set of *Trucks* transporting a set of *Packages*.
    - *Area (aSpain)*: Comprises one or more *Cities*.
  - Operators:
    - Collect: Starts the transportation of a *Delivery*.
    - Deliver: Hands over a *Delivery*.
    - Move: Moves a *Delivery* from an *Area* to another.

EXAMPLE (III)

- Correspondences between objects:
  - *Package* → *Package*: direct mapping
  - *City* → *Area*: simple grouping
  - *Truck*: detail reduction
  - *Delivery*: multiple grouping
    - Represents a set of *Trucks* carrying a set of *Packages*



EXAMPLE (IV)

- Mapping section:

```
(:mapping (implies
(and (at D1 aSpain) (in P1 D1))
(and (in P1 T1)
(or (at T1 Madrid) (at T1 Barcelona) (at T1 Valencia)))
))
```

- If the *Delivery* D1, which includes the *Package* P1, is placed at the *aSpain Area*:
  - P1 is loaded in the *Truck* T1, and
  - T1 is placed at Madrid, Barcelona or Valencia.

CONCLUSIONS

- We have presented a mechanism to achieve a common ontology among heterogeneous planning agents.
- A *PDDL*-based model is built coherently on top of the local models of the agents.
- Each agent handles now a two-layered model, that allows it to negotiate with the rest of agents while maintaining its original model.
- A set of design techniques and language extensions is used to define and manage the new model.

THANK YOU FOR YOUR ATTENTION