Multiagent Systems for Social Computation: The SmartSociety approach

Michael Rovatsos
The University of Edinburgh
CretaMASSS
Chania 22nd July 2013

SmartSociety

- 4-year €6.8M EU FP7 FET Integrated Project, co-ordinated by Trento
- Aim: building hybrid and diversity-aware collective adaptive systems to solve challenging societal problems
- Our focus: social orchestration of multi-level and overlapping concurrent computations + learning them from data
  - By the way, we're looking for a PhD student with machine learning/incentives background

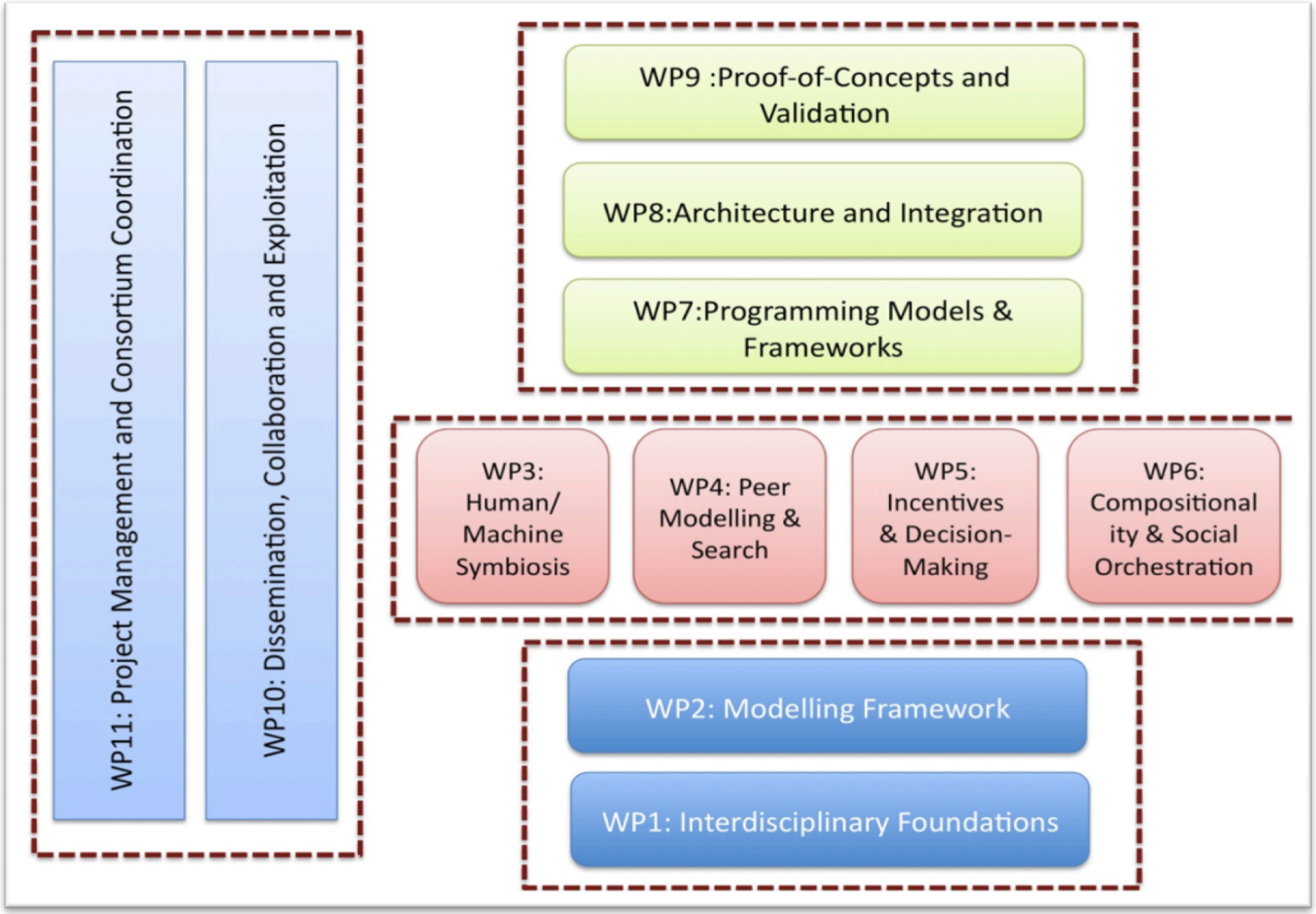
System context

- **Diversity -** we are interested in diverse populations of interacting humans and machines, with different backgrounds, knowledge, skills, objectives, and expectations
- **Hybridity -** humans and machines playing different roles(providing data, performing computations, making decisions), all our systems involve interaction with humans
- **Compositionality -** how do we compose individual interactions to obtain collective action and globally coherent social computations?

Research agenda

- **Design principles -** methods that are needed to build collective adaptive systems, to manage the emergent behaviours they give rise and to validate these designs
- **Operating principles -** principles by which collective adaptive systems operate. How are they controlled and optimised? How can they provide long-term stability? How do they resolve conflicts and failures?
- **Evolutionary principles –** we need to understand the nature of systems evolution, to relate it to individual and collective learning processes, and how these affect operating and design principles

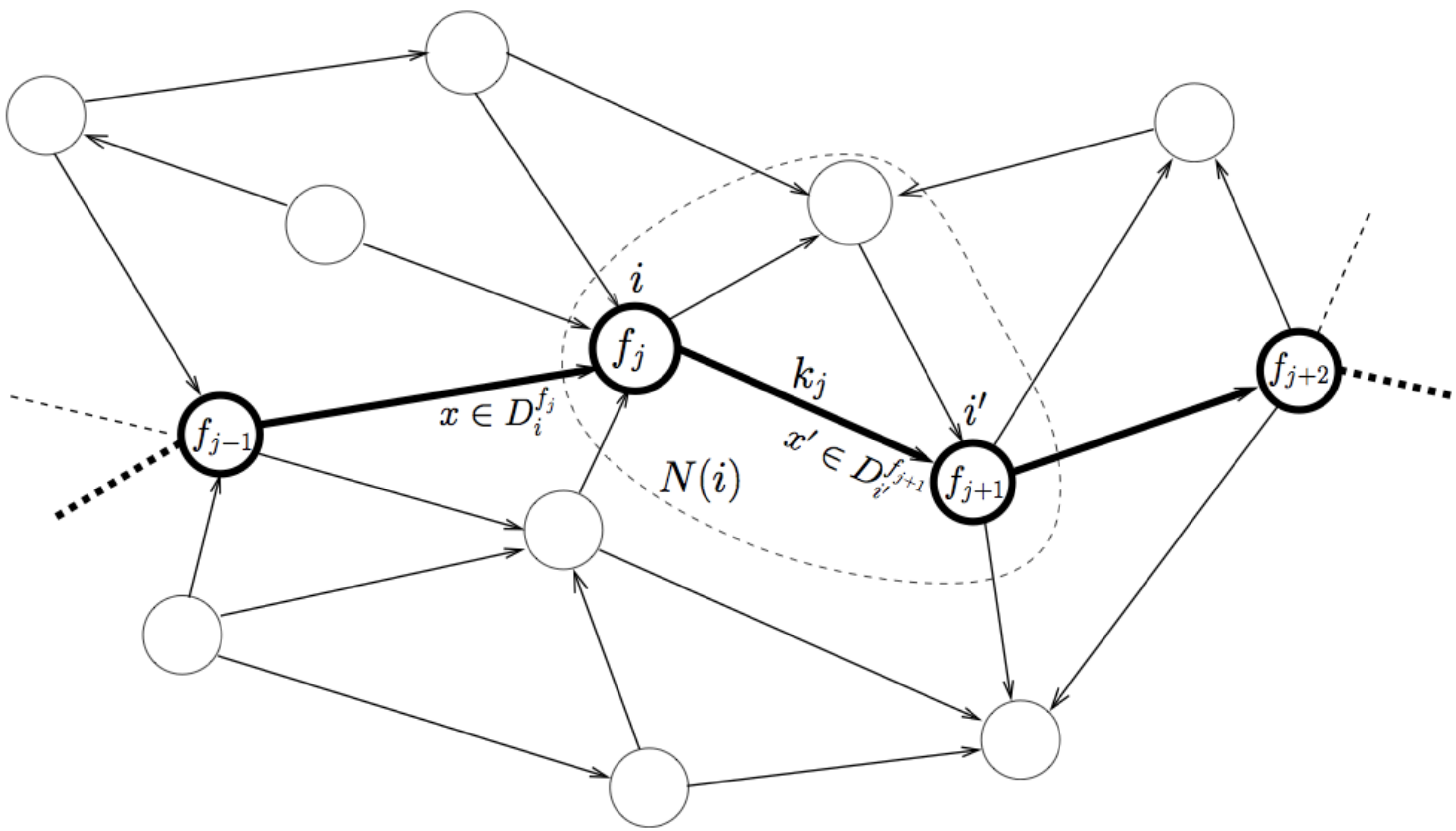
Organisation



Compositionality & Social Orchestration

- Edinburgh part in the project: develop architectures and algorithms for composing and orchestrating social computations
- Provides the link between conceptual modelling layers and actual implementation of social computation systems
- First iteration: **static** lightweight social orchestration
- Second iteration: **dynamic** aspects of social computation
  - identifying emergent social structures, developing adaptive incentive systems, optimising social orchestrations

Abstract architecture



Platform architecture

- Nodes know about (some) capabilities of (some) other nodes, can devise plans to achieve a goal
- Task workflows are advertised on an (initially) centralised platform (later enhanced by peer search & federation)
- Platform provides a small set of meta-protocols for this
  - brokering & network exploration (friend-of-a-friend search)
  - voting (for social choice)
  - auction (for resource or task allocation)
- Meta-protocol execution leads to agreement on task workflow, sub-tasks allocated to peers

Lightweight social orchestration

- Actually, we want to do all of this without a peer-to-peer style platform, without shared state and heavy runtime co-ordination
- Instead, exploit process=data duality in the following way:
  - Social computation specification provided as distributed linked data describing peers, tasks, preferences, goals, norms etc
  - Atomic contributions by humans and machines viewed as lightweight RESTful web services provided by endpoints
  - Execution becomes a distributed query over several datastores, where humans supply the inference steps machines cannot complete

Lightweight social orchestration

- So, actually, we don't want to build a "hybrid human multiagent system"
- We want a Web-style way of crowdsouring computation in an open-ended, arbitrarily scalable way
- Rely only on common Linked Data standards and separate client from data/process model
- With a view to learning and emergence work to be done later, this gives us also directly a "Big Data" view of the world

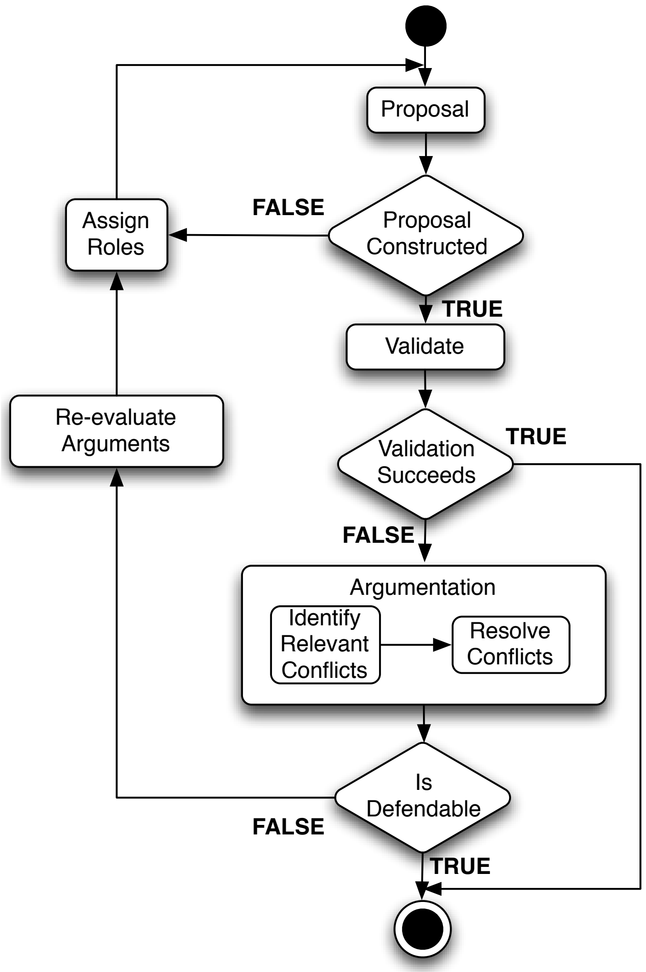
So where do agent techniques come into all this?

- Automated decomposition of planning domains (Crosby & MR)
- **Planning for self-interested agents (Jonsson & MR)**
- **Argumentation-based conflict resolution in planning (Belesiotis & MR)**
- Distributed execution of centralised plans with global constraint maintenance (Herry, Anderson & MR)
- **Automated norm synthesis for planning environments (Christelis & MR)**
- **Qualitative trust modelling in interaction protocols (Serrano & MR)**

Argumentation-Based Conflict Resolution in Planning

- Agents disagree about initial state and action definitions, but share goal
- Our work focuses on *acceptable plans*
  - p is acceptable wrt KB_1 and KB_2 iff $KB_1 \models p$ and $KB_2 \models p$
- Developed argumentation-based method based on evaluating individual agents' proposals to compute defendable plan
- Scalability achieved by using off-the-shelf single-agent planners for sub-tasks in the process

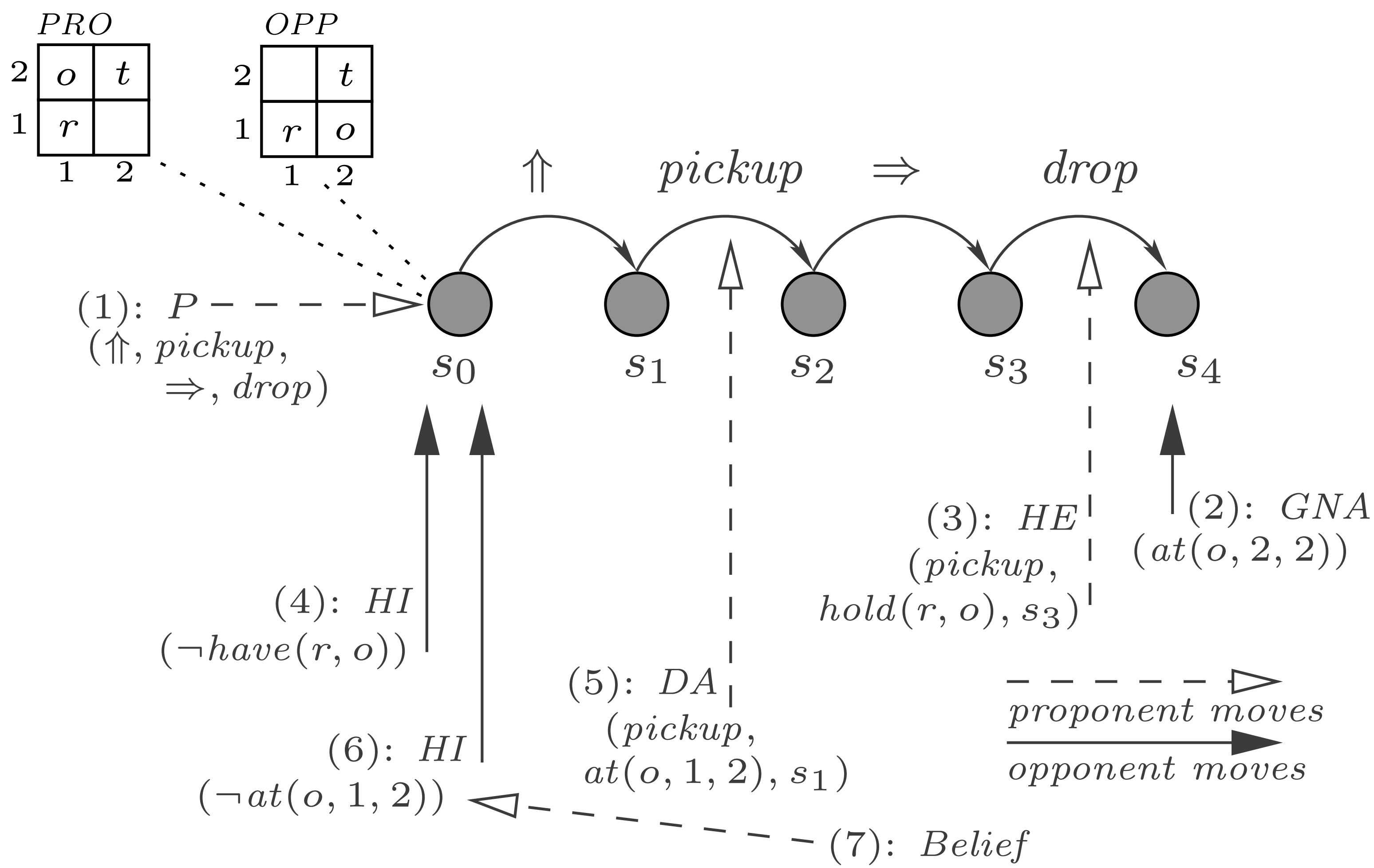
Argumentation-based conflict resolution



- Plan proposal generated by single agent (with any planner)
- Dispute in case of disagreement, argumentation follows
- Ends in successful defence of initial proposal or rejection + belief revision

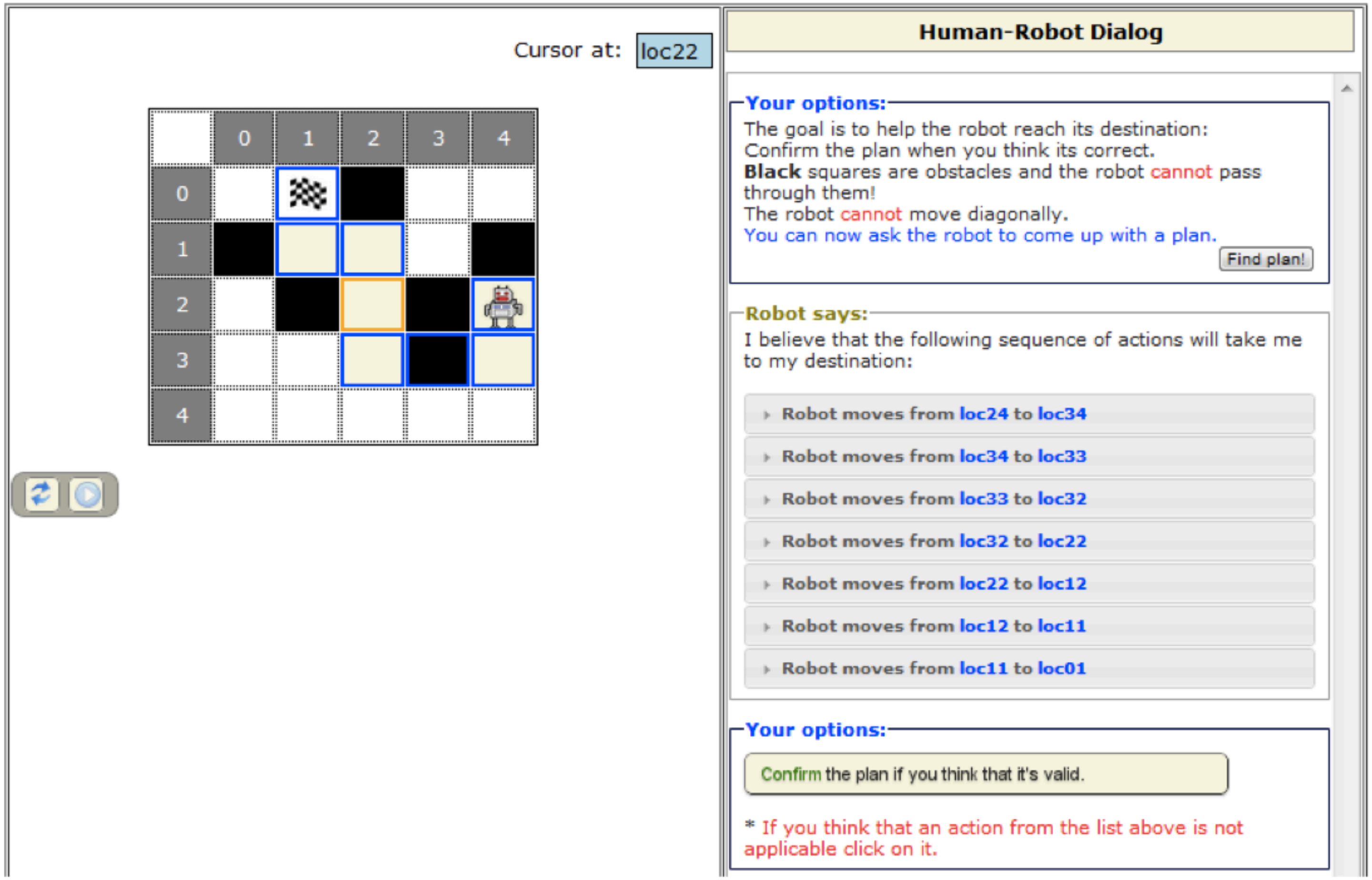
Example

- Robot gridworld domain



Application: ArguDem

- A demonstrator for helping robots navigate:

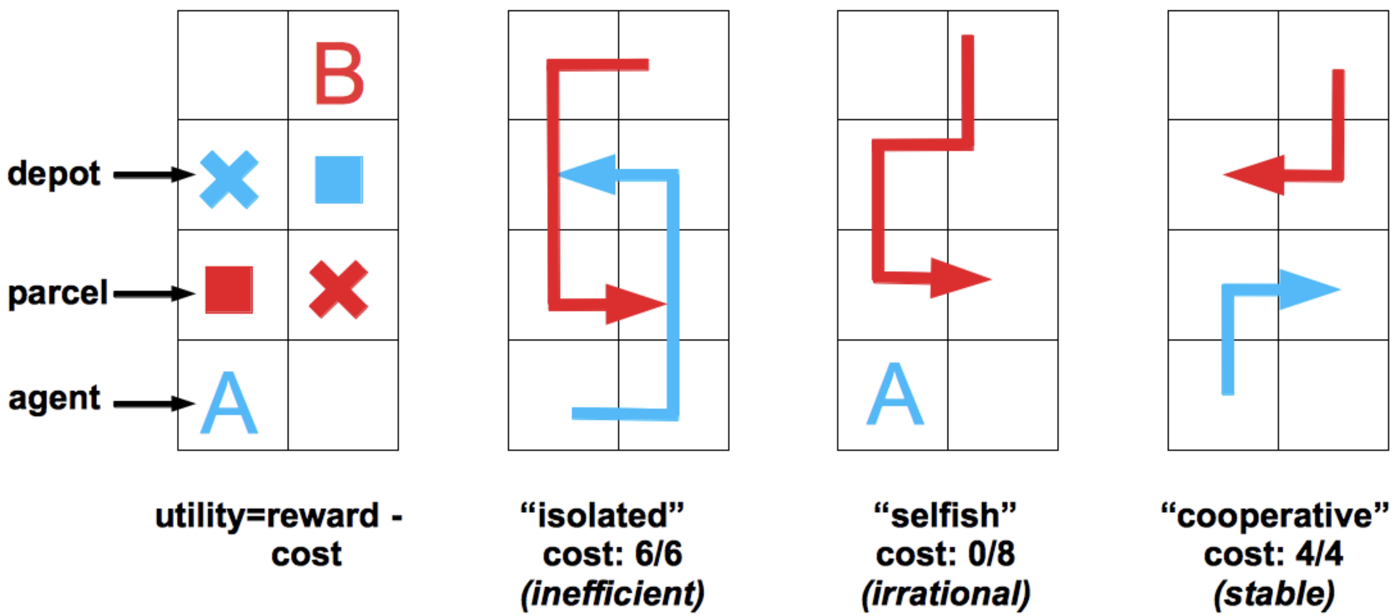


Planning for Self-Interested Agents

- Agents with independent, potentially conflicting (though not inconsistent) goals
- Strategic problem, acceptability based on notions of stability and equilibrium
- Problem depends on whether contracts can be enforced and utility can be transferred
- Like concurrent planning with additional constraints on plan cost to individuals
- Hard to define meaningful solution concepts if goals incompatible or agents untrustworthy

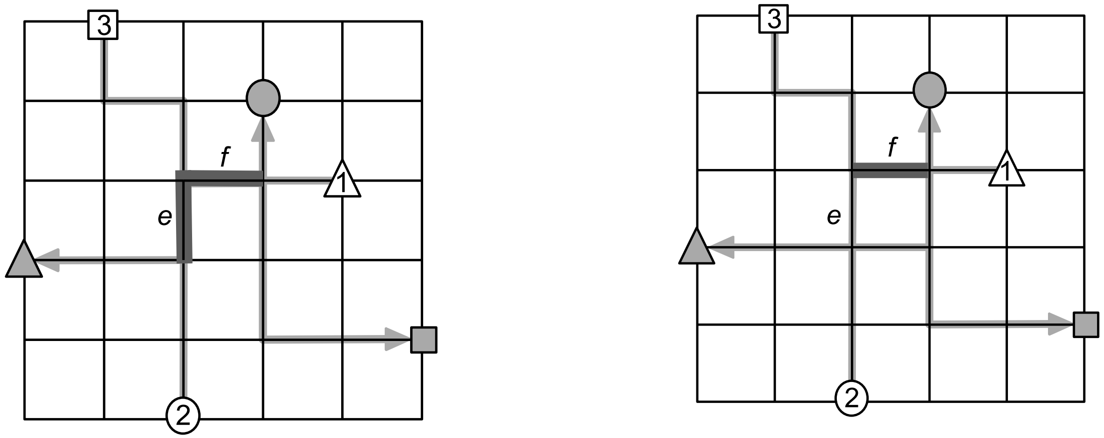
Example

- Delivery domain



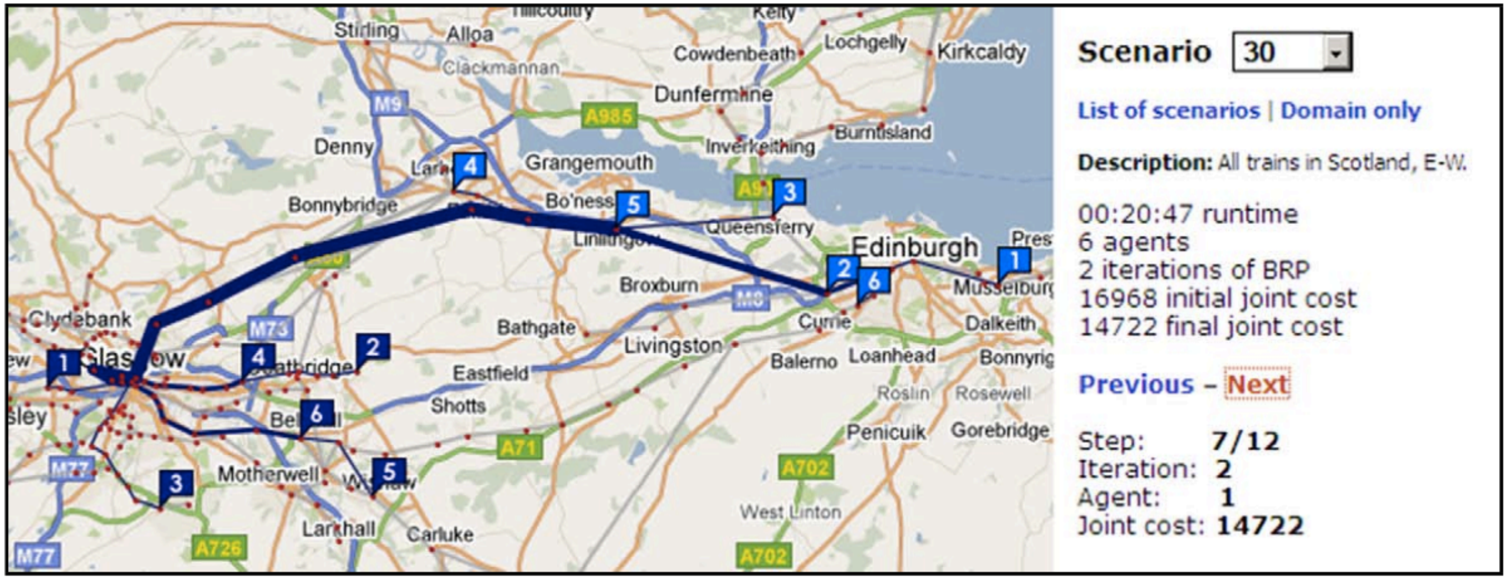
Planning for Self-Interested Agents

- Best-Response Planning (Jonsson & MR):
  - iterative method of optimising agents' individual plans without breaking others' plans
  - computes equilibrium plans fast in congestion games, restricted to interactions regarding cost
  - useful for plan optimisation in unrestricted domains
- Network routing example:



Application: Ride Sharing

- Hrncir's system uses BRP to determine joint travel routes using real-world UK public transportation data (>200,000 connections)

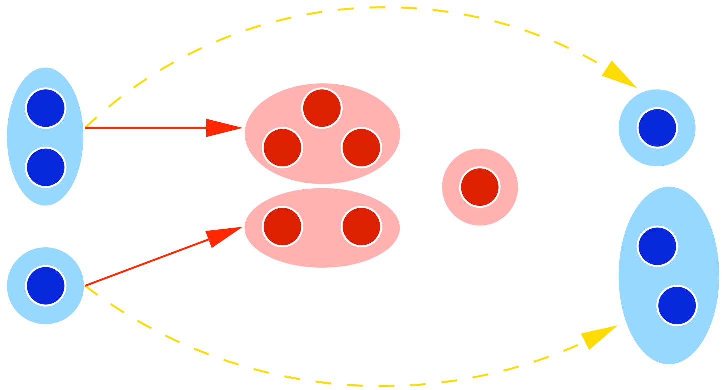


Automated norm synthesis

- Norms ensure global conflict states are never entered by prohibiting actions in certain states
- At the same time agents' private goals should remain achievable
- Automated synthesis of such norms is NP-hard in enumerated state systems
- Existing methods don't exploit abstractions of propositional/first-order domain theories
- Our method: find "detours" around conflict states by local search in generalised state spaces

Automated norm synthesis

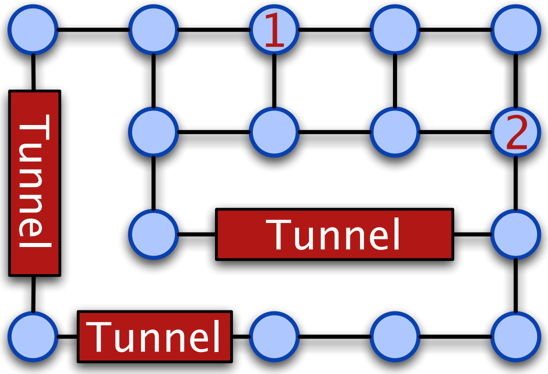
- Iterated process of forward-backward search around conflict state specification:



- Not better than full state-space search in the worst case but we can often "get lucky"
- With simple additional pruning techniques search can often be cut down drastically

Example

- Tunnel world example:

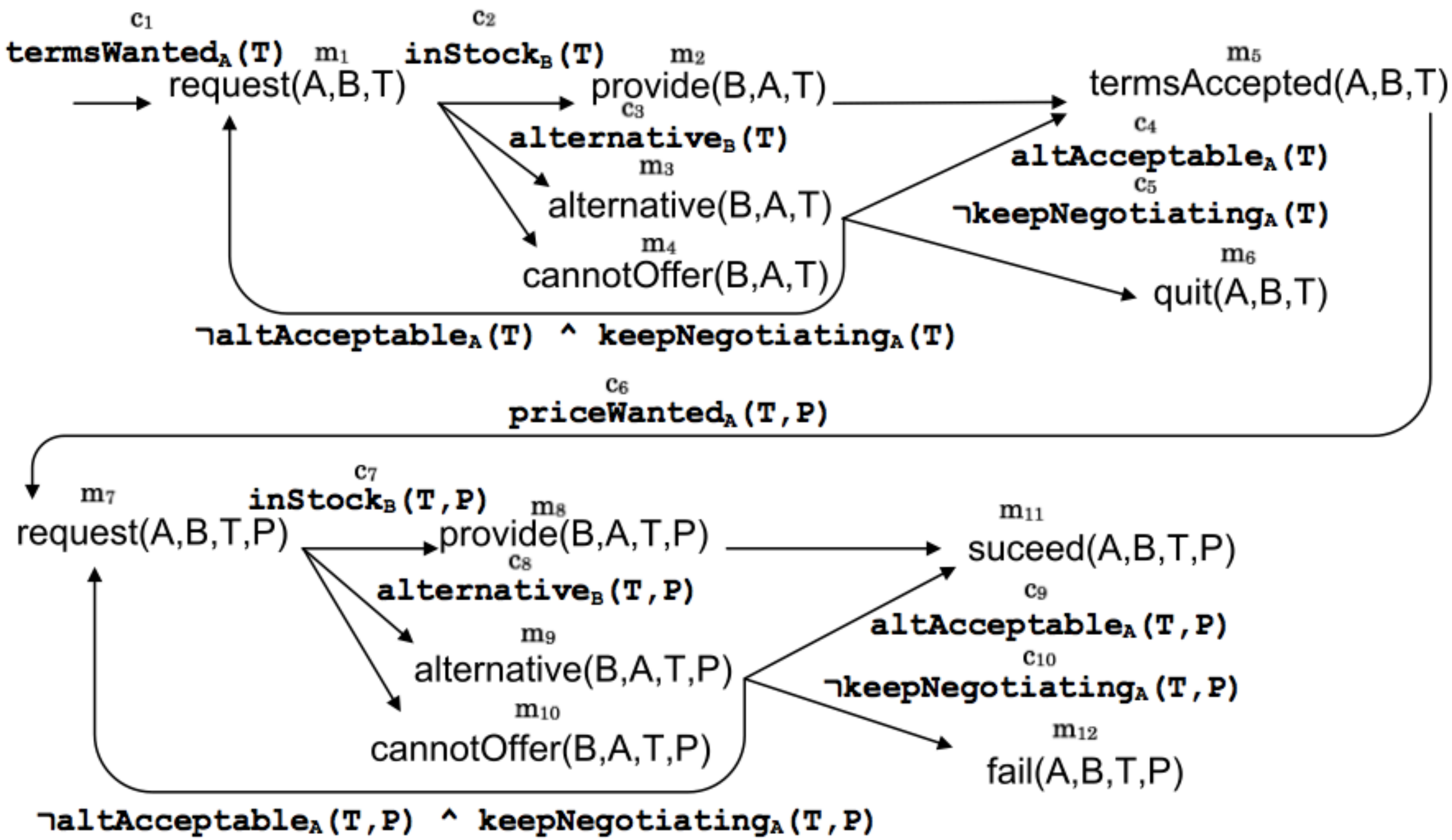


- Agents entering tunnels have to leave them out the opposite end immediately (on entering tunnel, future crash not avoidable)
- Our algorithm solves this by computing a general norm *“if you are next to a tunnel and another agent is at the opposite end, don’t enter the tunnel”*

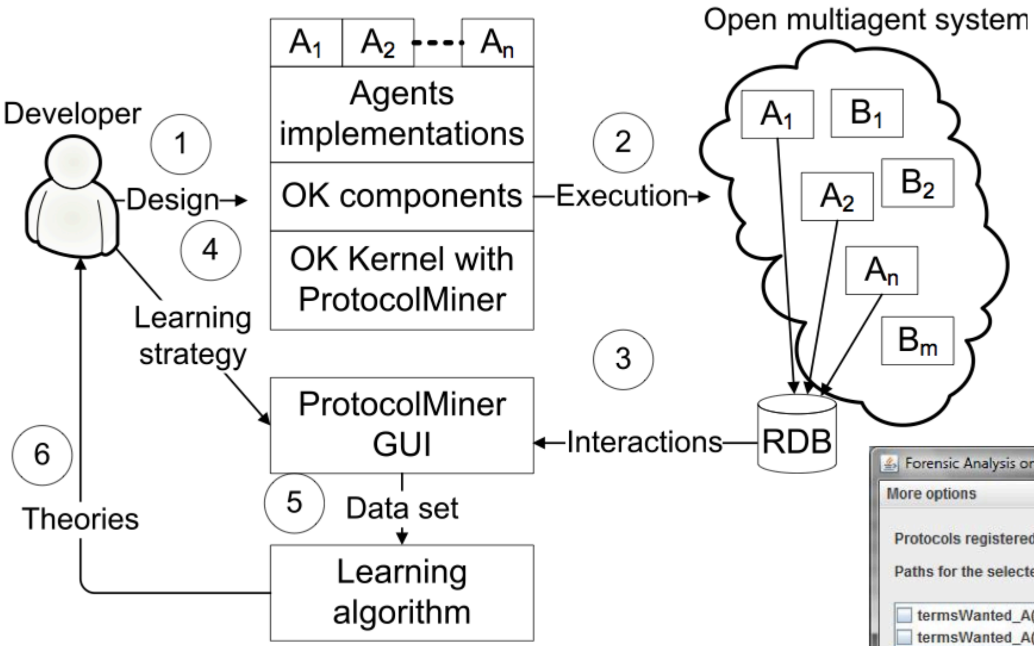
Qualitative Context Mining

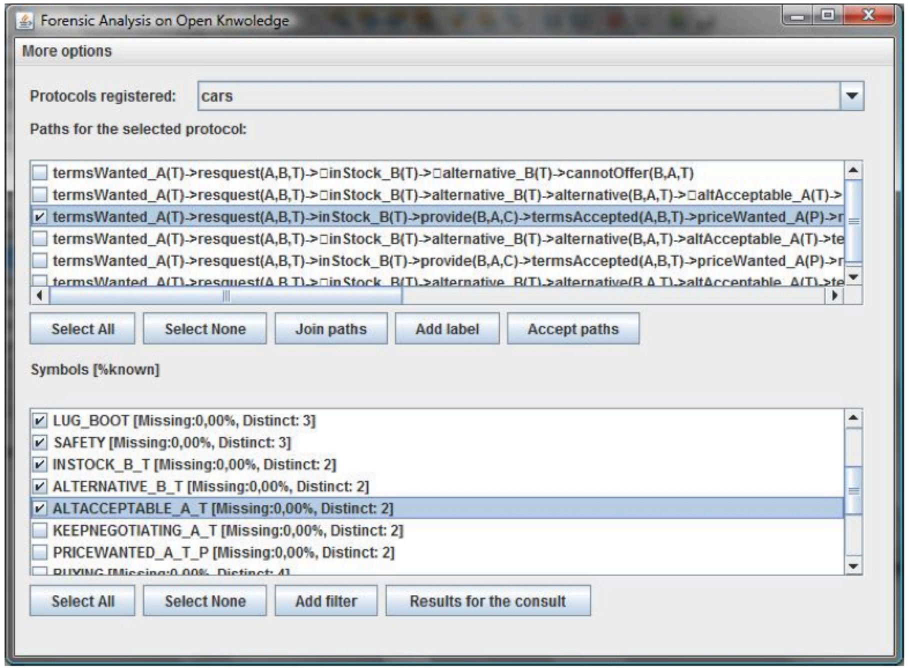
- Qualitative context mining (Serrano&MR)
- Relate constraints in protocols to outcomes, exploiting knowledge-level ACL semantics
- Can be used for
  - predicting outcomes and adjusting strategies
  - identifying misaligned constraint interpretations
  - deriving qualitative trust and reputation measures

Qualitative context modelling



ProtocolMiner





Mining agent protocols

```
persons = 2: F (158)
persons = 4: F (158)
persons = more
|   lug_boot = small
|   |   doors = 2: F (8)
|   |   doors = 3: F (7)
|   |   doors = 4: F (8)
|   |   doors = 5-more: T (105)
|   lug_boot = med
|   |   doors = 2: F (13)
|   |   doors = 3: F (8)
|   |   doors = 4: F (13)
|   |   doors = 5-more: T (120)
|   lug_boot = big: T (402)
```

Conclusions

- Large-scale social computation: a new, challenging domain
- Many multiagent technologies can be used to (partially) automate hard sub-problems
- The challenge is to take them to the real world and confront them with humans
- Focus on developing generic techniques, not a single application that happens to work well

ESSENCE - Evolution of Shared Semantics in Computational Environments

- 4-year, €4M Marie Curie Initial Training Network, co-ordinated by Edinburgh
- Aim: to exploit human methods for negotiating, sharing, and evolving meanings for computational systems
- Focus in Edinburgh: Communication planning from heterogeneous sensor data and ontology learning
  - By the way, we have funding for 11 PhD students and 4 post-docs (but you have to go abroad)