Rationality in communication: from learning to planning and back

Michael Rovatsos
School of Informatics
University of Edinburgh
mrovatso@inf.ed.ac.uk

King's College AIS Seminar, 4th July 2013

The story behind this talk

- Christos contacted me about work done a long time ago on learning argumentation strategies
- Remembering that line of work brought me back to question that's driven much of my later research, too
- In this talk I make an attempt to talk about the different pieces in the puzzle that I've tried to work on

The Grand Challenge

- Understanding the principles of communication as rational action
  - *What should I say to whom to achieve my goals?*
- A key problem in AI, philosophy, linguistics, widely studies in multiagent systems
  - Speech act theory
  - Signalling games
  - Dialog systems
  - Argumentation theory

etc

My Small Challenge

- Given what you have observed in previous communication and action, what is the best thing you should say from a bounded set of options?
- This involves:
  - calculating what one might say
  - tracking success/failure
  - generalising over instance experiences
- Problems:
  - observations are (mostly) statistical, language is (mostly) symbolic
  - semantic models (mostly) not very practical for computation, content languages (mostly) infinite
  - immediate vs long-term utility, "cheap talk"

Learning communication strategies

- My PhD work was about *interaction frames* and reinforcement learning over them:

$$F = \Big\langle \overbrace{\langle \xrightarrow{5} \texttt{request}(A,B,X) \xrightarrow{3} \texttt{do}(B,X) \rangle}^{T},$$

$$\langle \overbrace{\{can(B,X)\}}^{\Theta_1}, \overbrace{\{can(B,\texttt{pay}(S))\}}^{\Theta_2} \rangle$$

$$\langle \overbrace{\xrightarrow{2} \langle [A/\texttt{a}], [B/\texttt{b}], [X/\texttt{pay}(\$100)] \rangle}^{C_1},$$

$$\overbrace{\xrightarrow{1} \langle [A/\texttt{b}], [B/\texttt{a}], [X/\texttt{pay}(S)] \rangle}^{C_2} \rangle \Big\rangle$$

Learning communication strategies

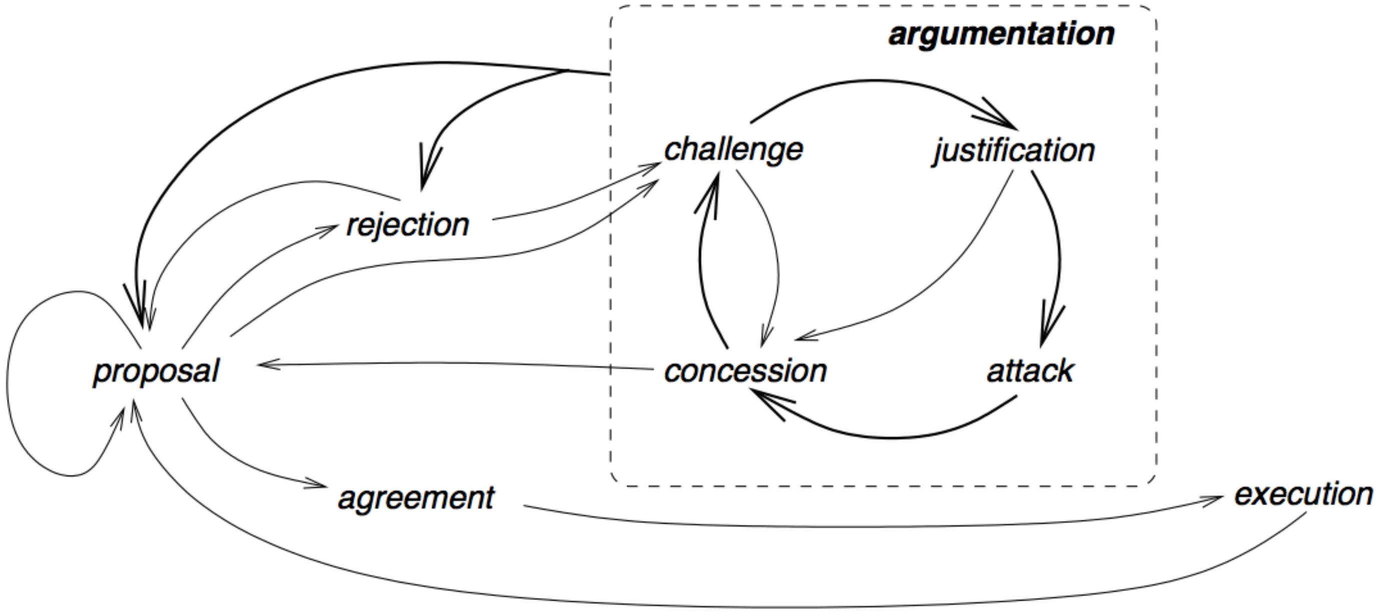
- Mix of symbolic and numerical representation
- Problem: modelling communication state
  - what determines whether a communication choice is appropriate?
  - only domain-dependent solution provided, strategy value depending on goals
- Learning over finite sets of pre-defined options regarding speech acts and content
  - though some induction over patterns

Learning communication strategies

$$\texttt{request}(A,B,X) \to \texttt{accept}(B,A,X) \to \texttt{confirm}(A,B,X) \to \texttt{do}(B,X)$$
$$\texttt{request}(A,B,X) \to \texttt{propose}(B,A,Y) \to \texttt{accept}(A,B,Y) \to \texttt{do}(B,Y)$$
$$\begin{aligned}\texttt{request}(A,B,X) \to\ & \texttt{propose-also}(B,A,Y) \to \texttt{accept}(A,B,Y) \to \\ & \texttt{do}(B,X) \to \texttt{do}(A,Y)\end{aligned}$$
$$\texttt{request}(A,B,X) \to \texttt{reject}(B,A,X)$$
$$\texttt{request}(A,B,X) \to \texttt{propose}(B,A,Y) \to \texttt{reject}(B,A,Y)$$
$$\texttt{request}(A,B,X) \to \texttt{propose-also}(B,A,Y) \to \texttt{reject}(B,A,Y)$$

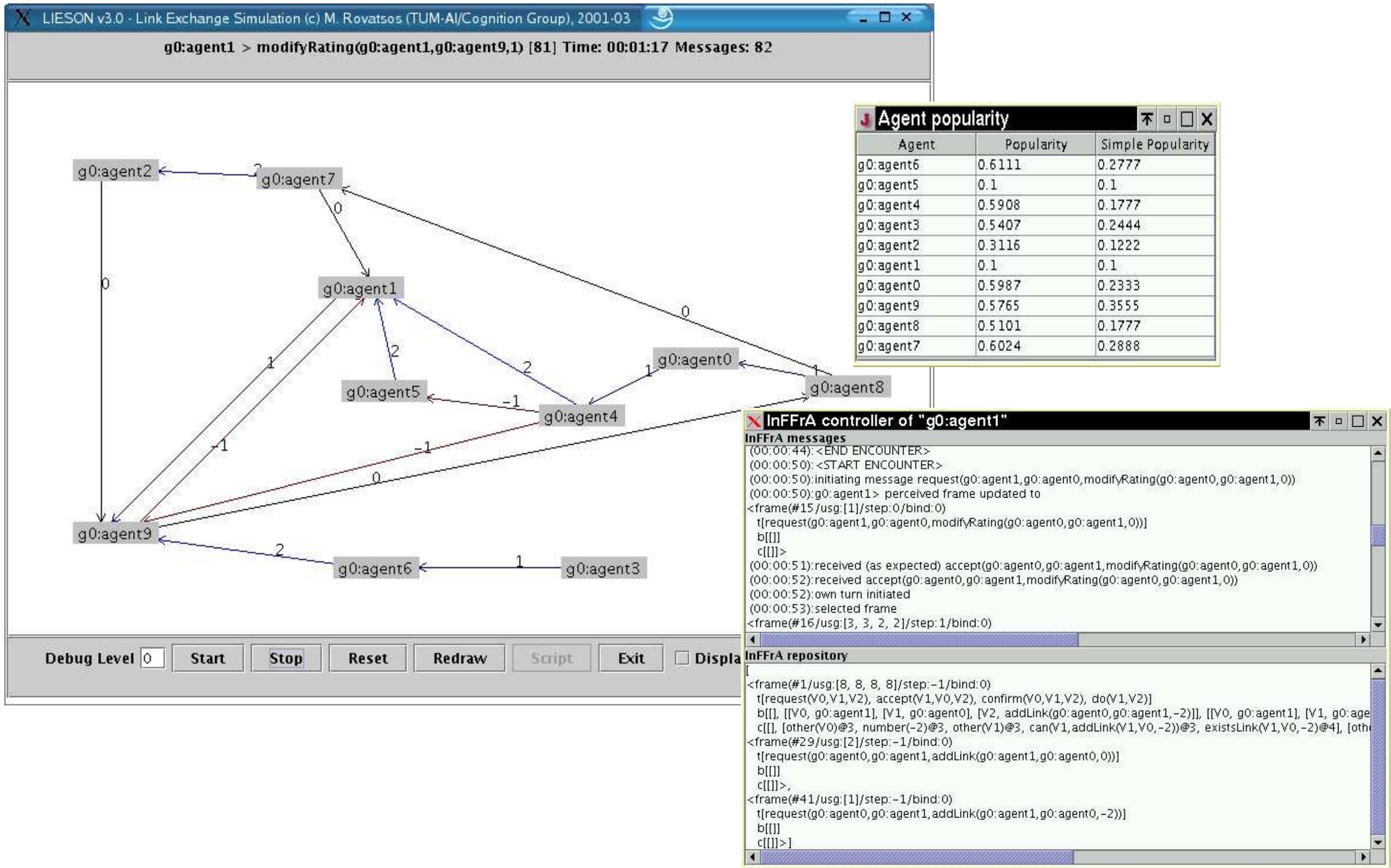
Learning argumentation strategies

- MR & Rahwan applied this to argumentation strategies based on model of interest-based negotiation



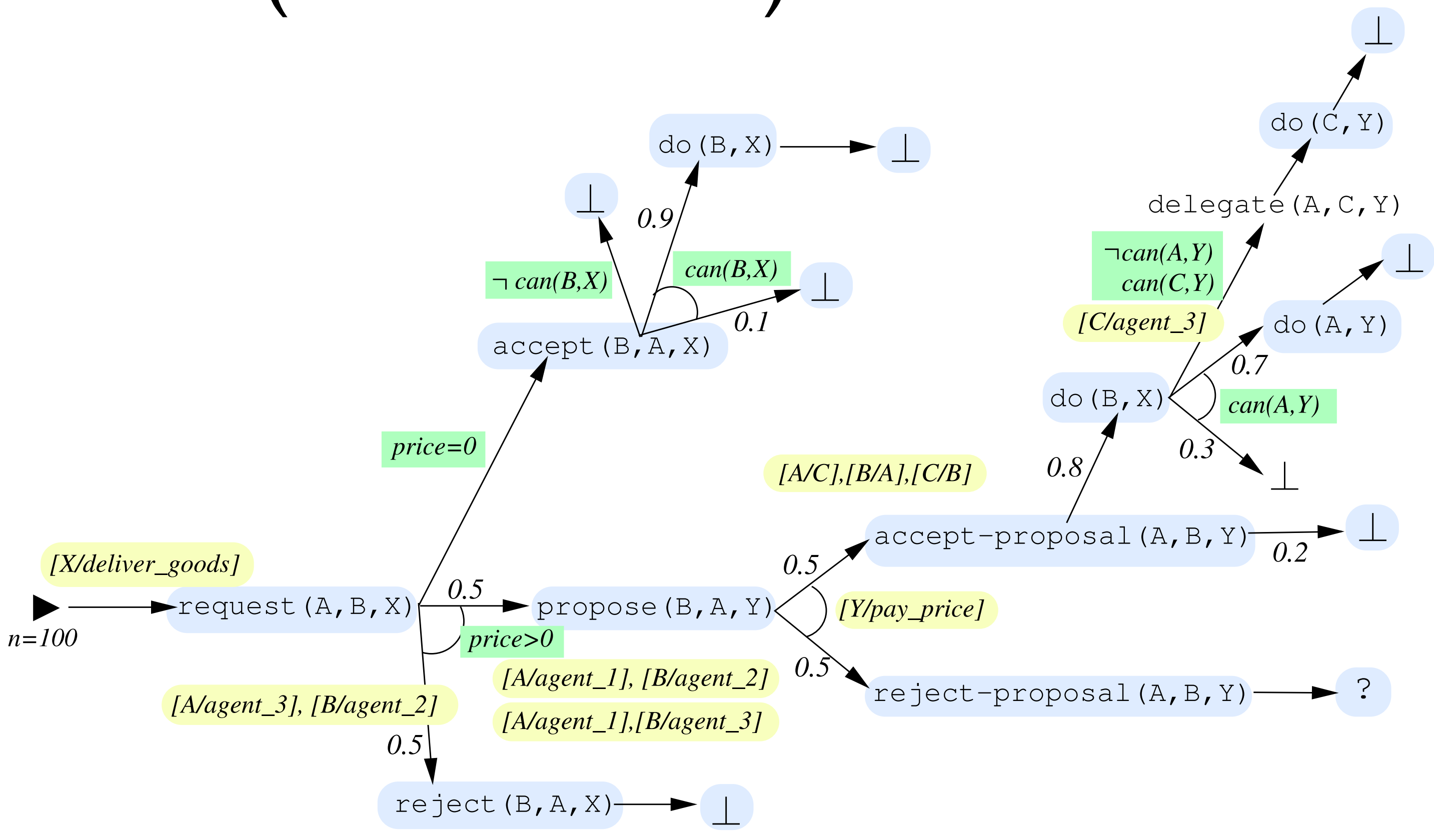
Example: link exchange simulation

- Early example of social computing (now slightly outdated)

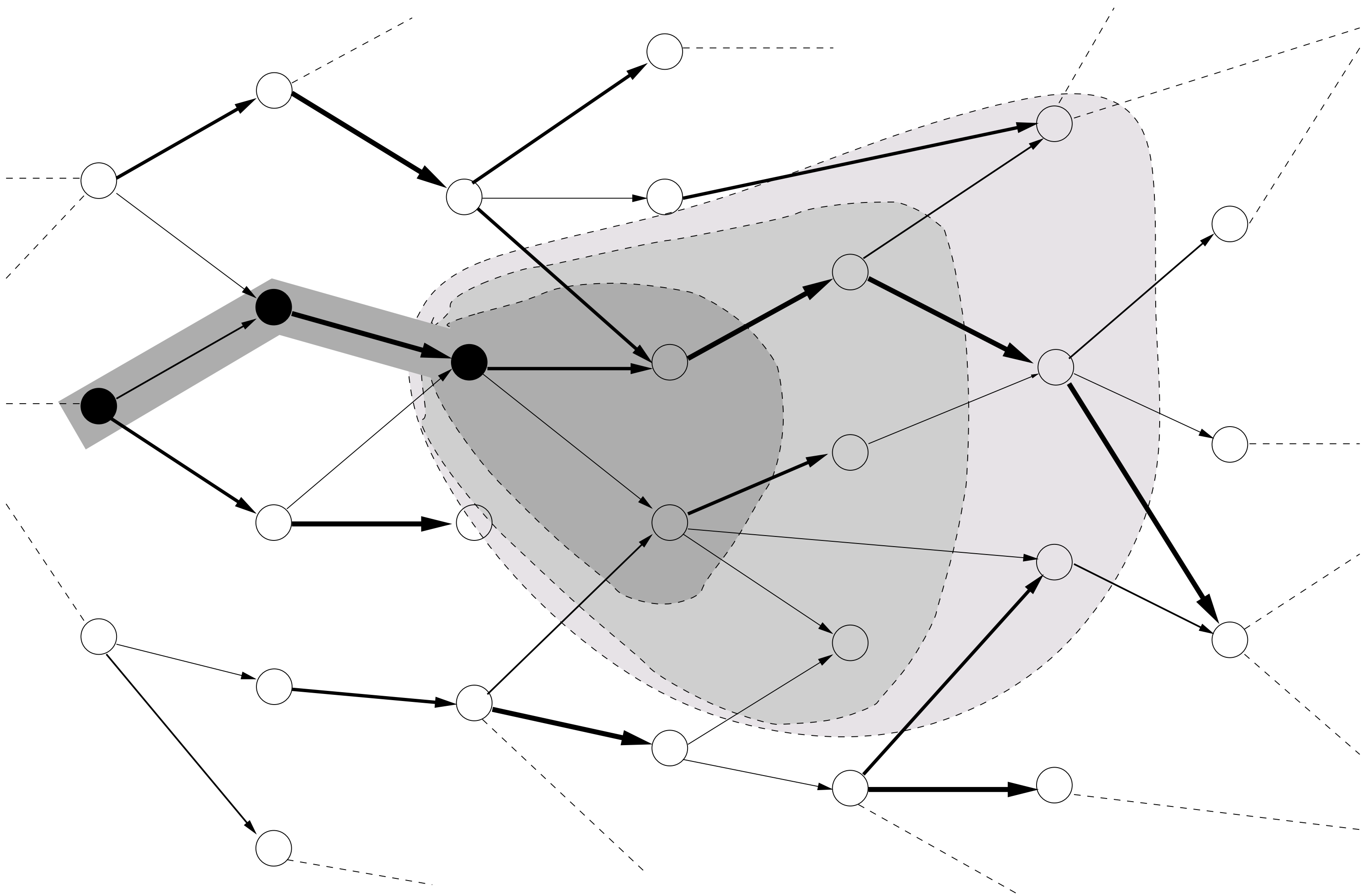


Expectation networks

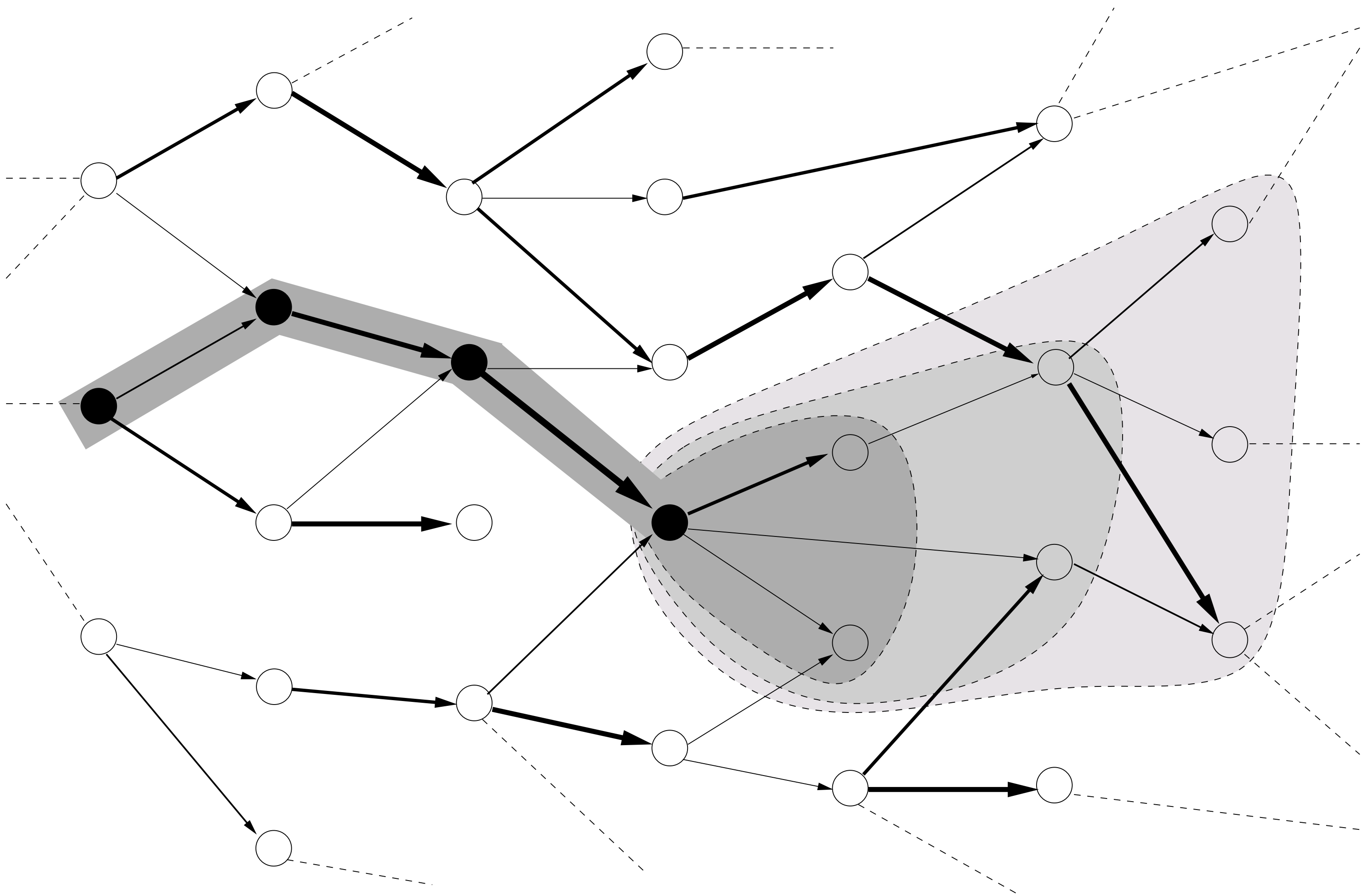
- Generalisation of the interaction frames idea (Nickles & MR)



Prediction dynamics



Prediction dynamics



(Where) Did we go wrong?

- Novel in terms of modelling semantics in terms of experience and prediction
- But no account for generation of *what* is talked about
- Leads to thinking about what agents mostly talk about: *their own activities*
- Most expressive model for modelling complex activity (while remaining tractable): *planning*
- Tricky: how can you know know what to say when you first have to compute the things you might talk about?

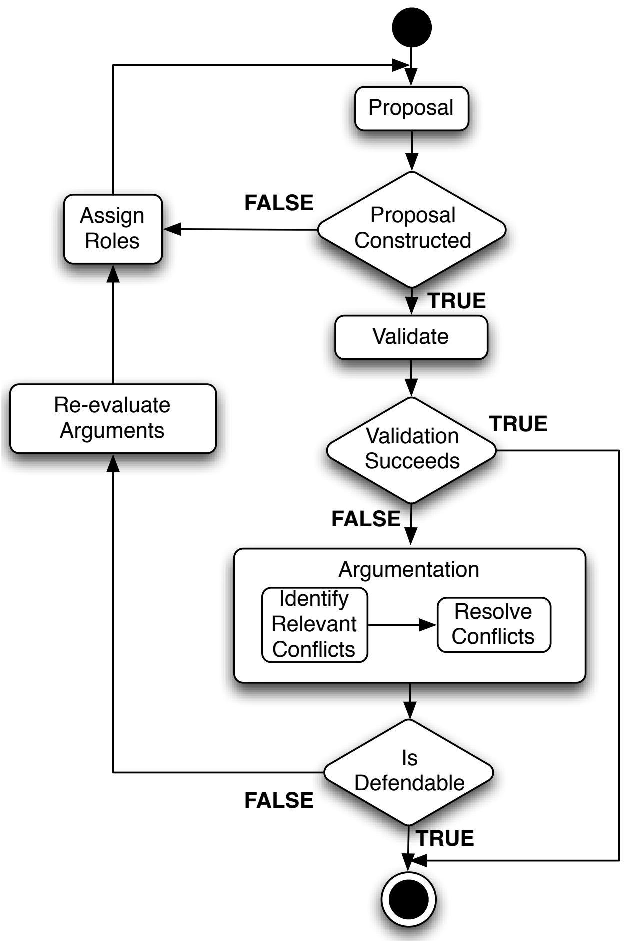
The relevance of planning

- Classical planning problem $P=<F,I,A,G>$, fluents F, initial state I, actions A, goal G
- Extend the above framework (naively) to accommodate multiagent aspects
  - $P=<F,\{\boldsymbol{A_i}\},\{\boldsymbol{I_i}\},G>$: multi-perspective planning
  - $P=<F,A,I,\{\boldsymbol{G_i}\}>$: multi-objective planning
  - $P=<\{\boldsymbol{F_i}\},\{\boldsymbol{A_i}\},I,G>$: multi-ontology planning
- Ignore concurrency, uncertainty, execution

Multi-Perspective Planning

- Agents disagree about initial state and action definitions, but share goal: P_i=<F,**{A_i},{I_i}**,G>
- Our work focuses on *acceptable plans*
  - p is acceptable wrt KB_1 and KB_2 iff $KB_1 \models p$ and $KB_2 \models p$
- Belesiotis & MR 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

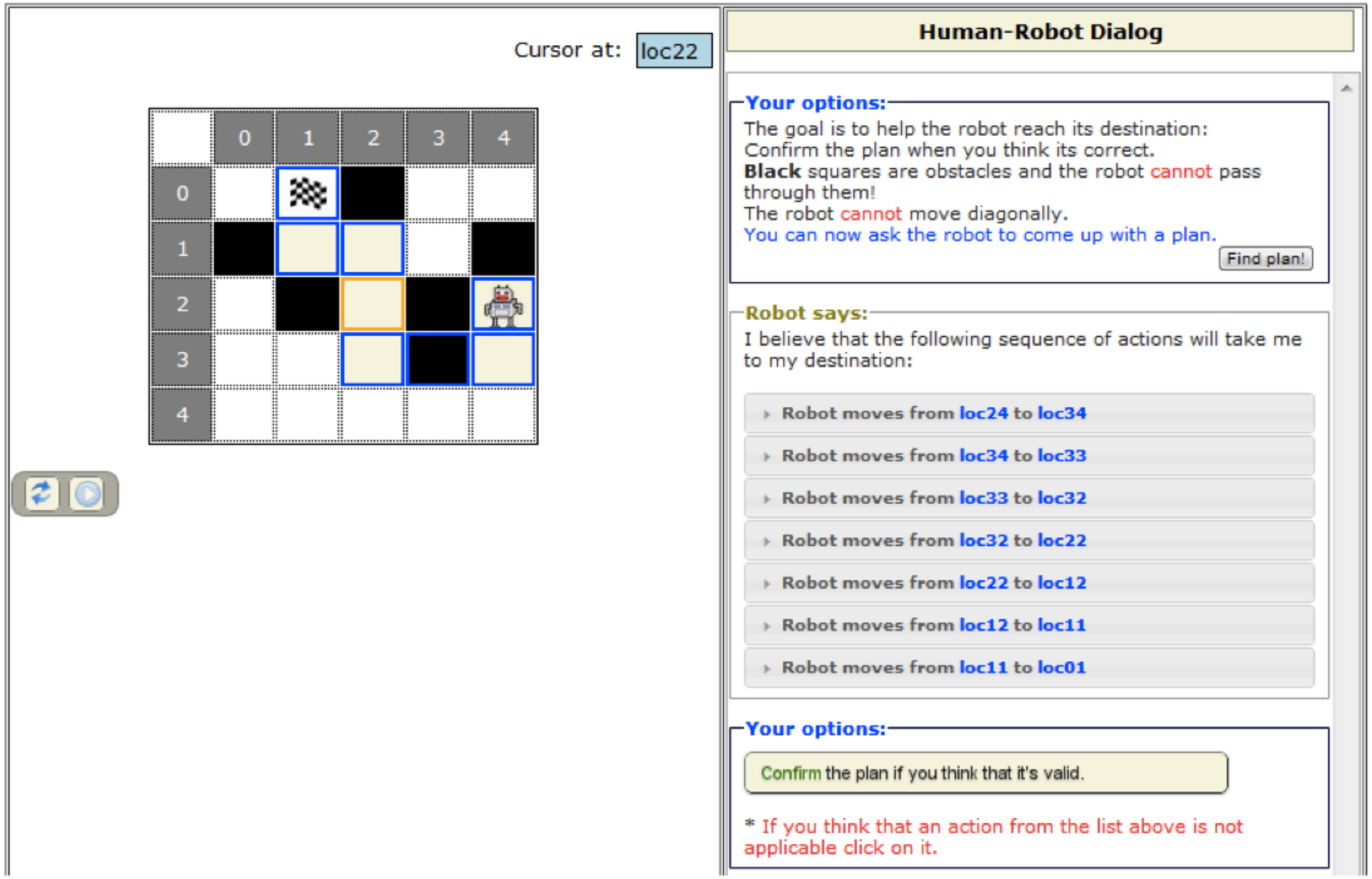
Argumentation-based conflict resolution in planning environments



- 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

Application: ArguDem

- A demonstrator for helping robots navigate:

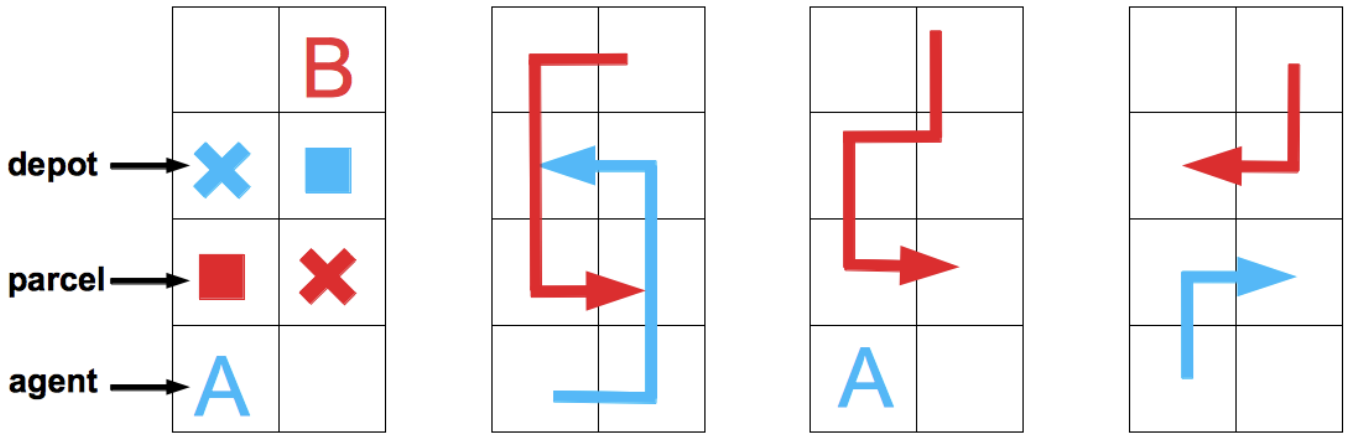


Multi-Objective Planning

- Introduce independent goals: $P=<F,A,I,\{\mathbf{G}_i\}>$
- 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



utility=reward - cost

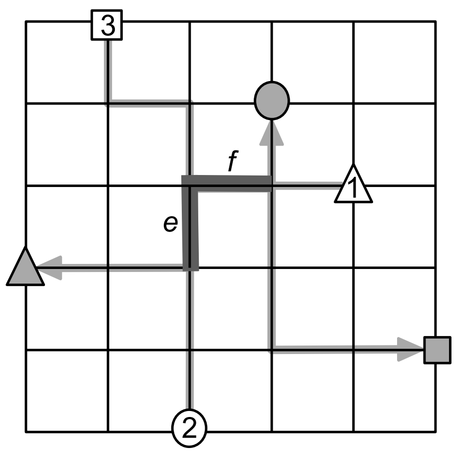
"isolated"
cost: 6/6
(inefficient)

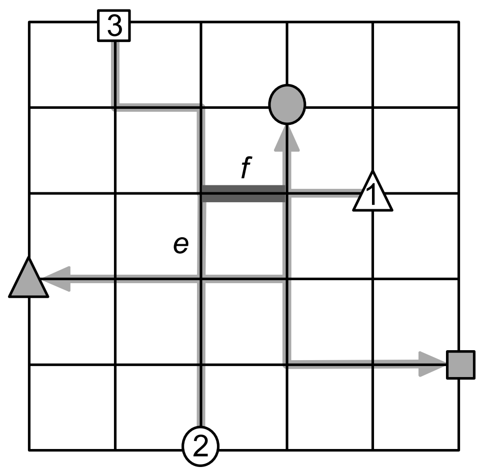
"selfish"
cost: 0/8
(irrational)

"cooperative"
cost: 4/4
(stable)

Multi-Objective Planning

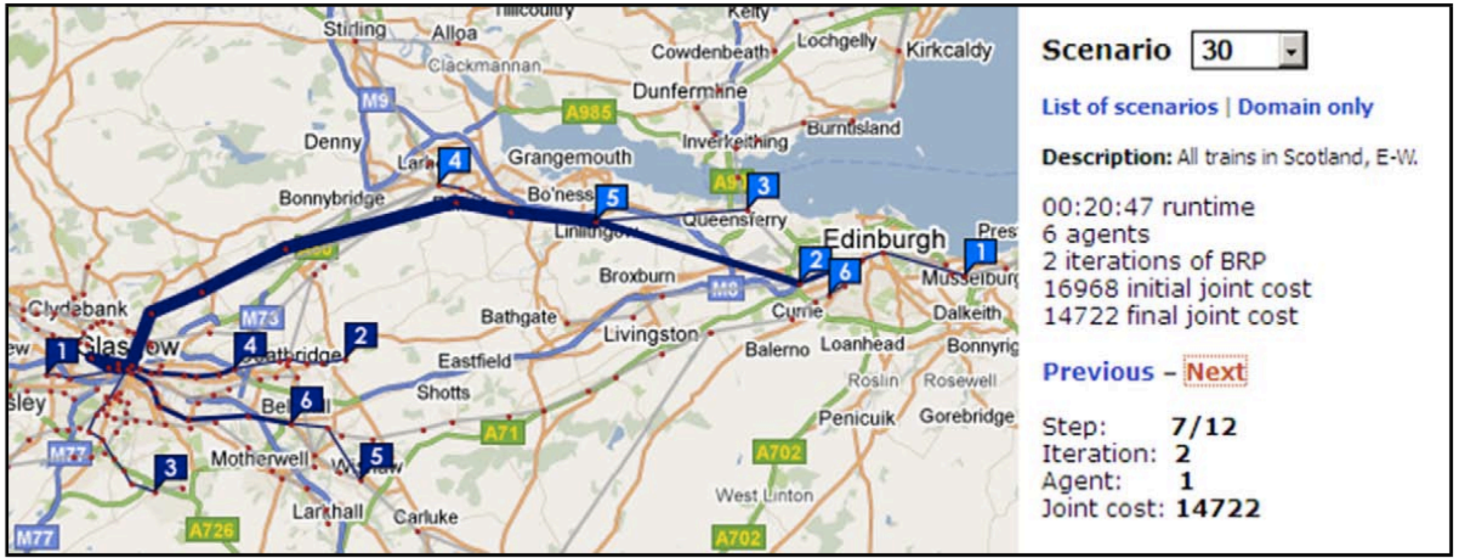
- Best-Response Planning (Jonsson & MR):
  - iterative method of optimising agents' individual plans without breaking others' plans
  - computes equilibrium plans fast in congestion games, restricted to interactions regarding cost
  - useful for plan optimisation in unrestricted domains
- Network routing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





Application: Trip Sharing

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

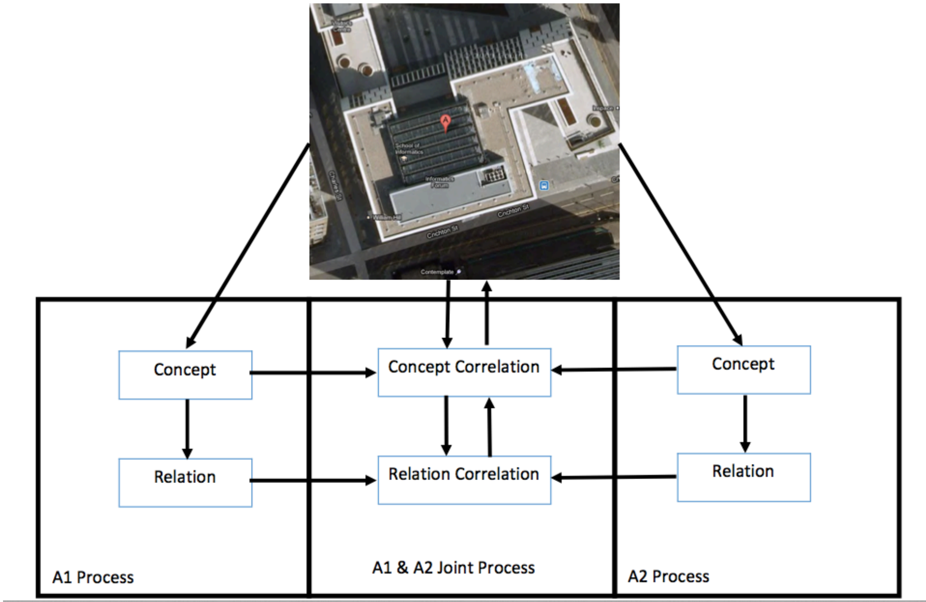


Multi-Ontology Planning

- P=<$\{\boldsymbol{F_i}\}$,$\{\boldsymbol{A_i}\}$,I,G>: multi-ontology planning
- Systems like ORS (McNeill & Bundy) address the issue of creating plans under ontological disagreement
- But how does this relate to the data agents' local models come from?
- Moreover, even if we assume local models are initially known, how about change?
- These questions bring us back to thinking about data and symbol grounding

Ontology evolution & language games

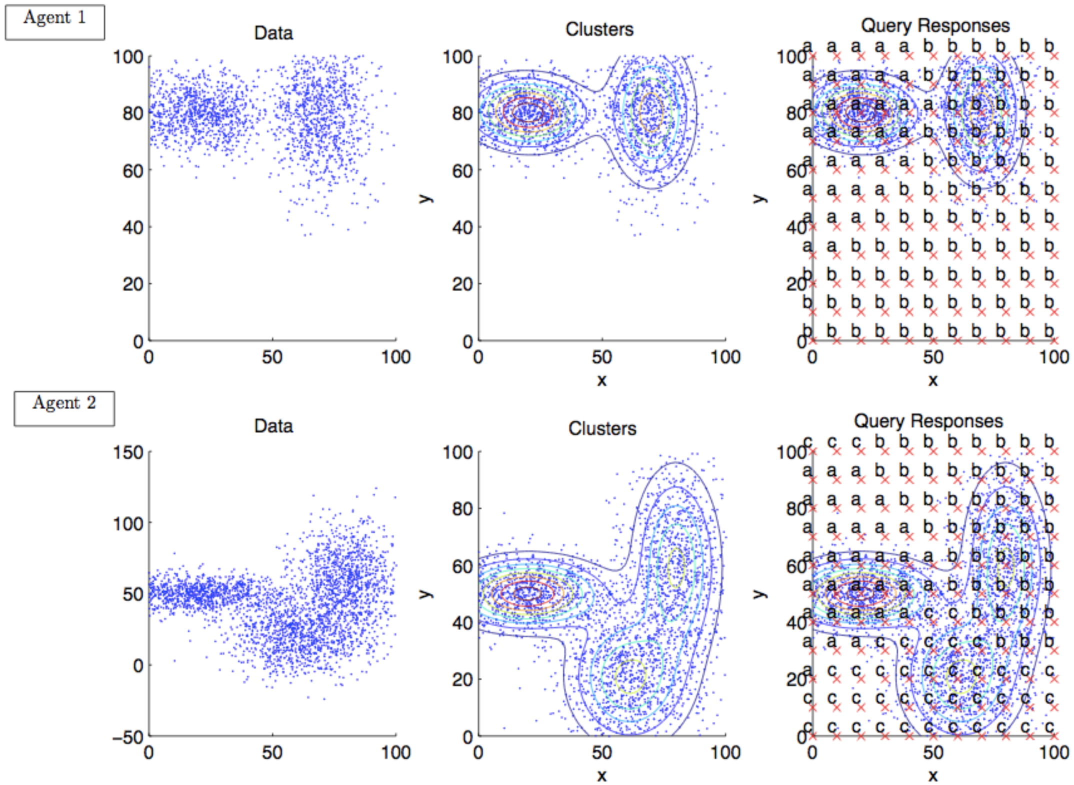
- Anslow's prototype of co-evolving concepts through mutual querying



- Concept/relation correlation can be through symbolic ontologies

Example: tracking objects/events

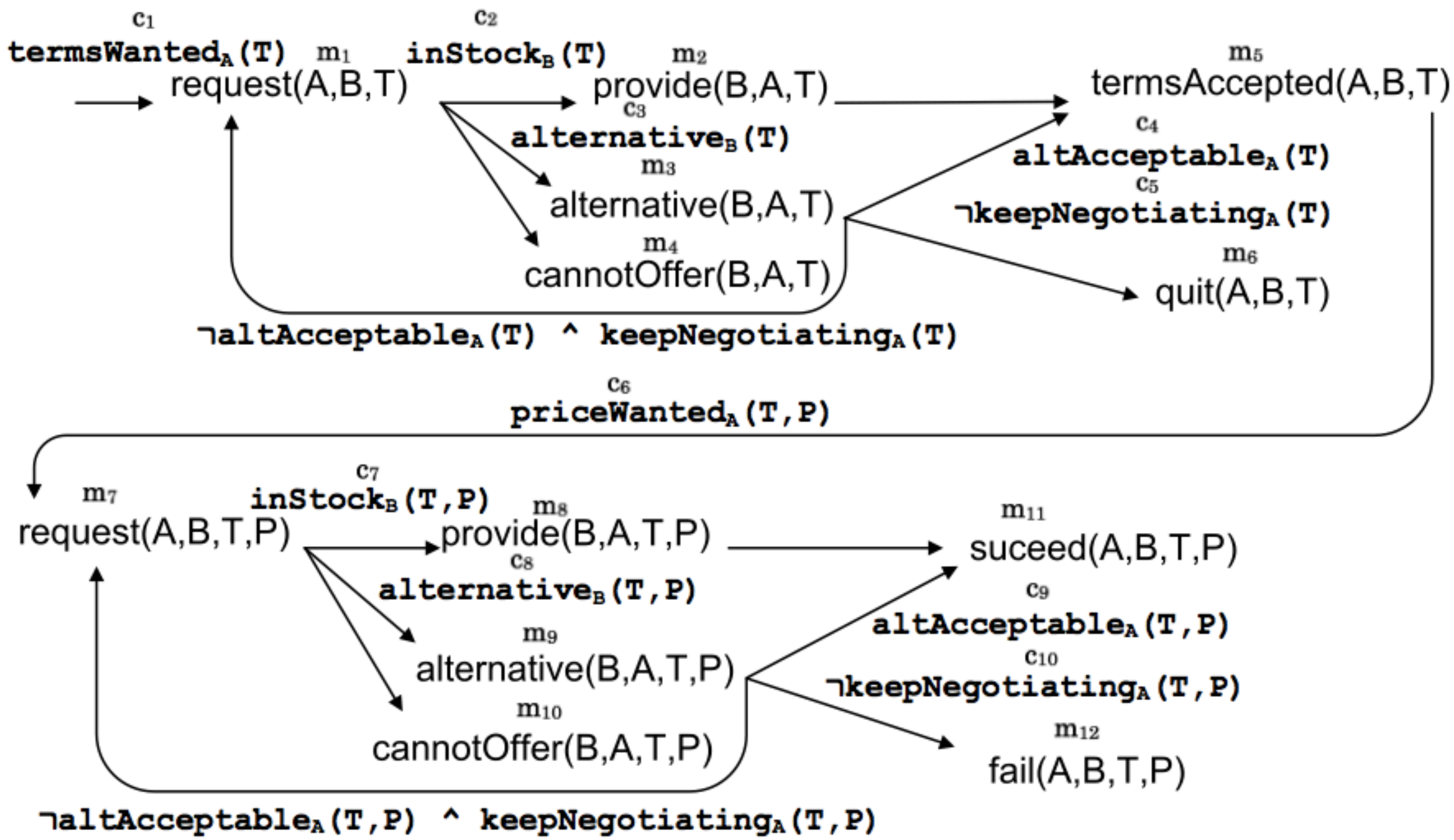
- Heterogeneous sensors clustering "interesting" events



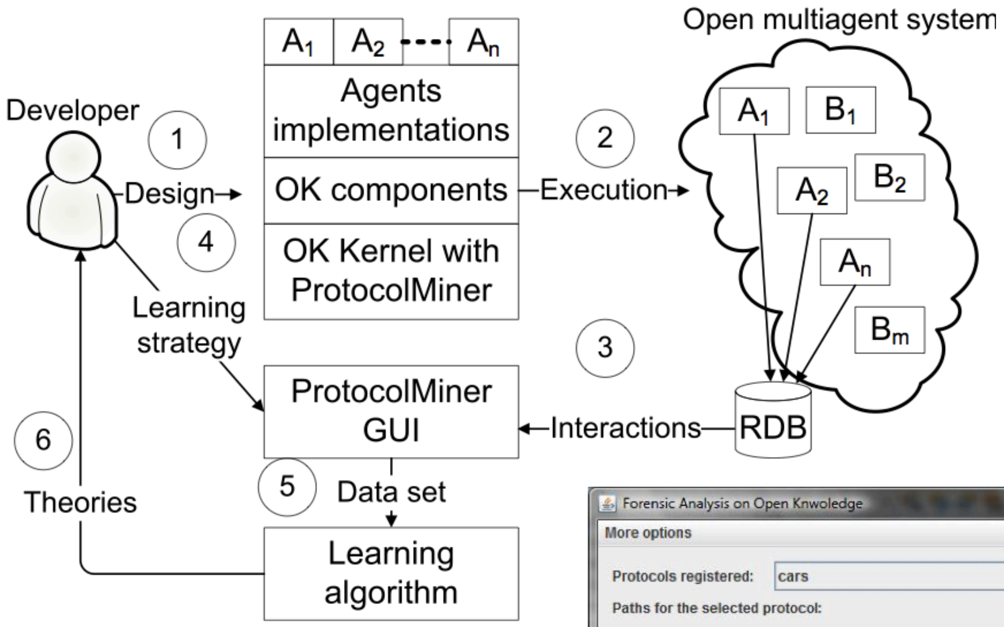
Learning communication strategies revisited

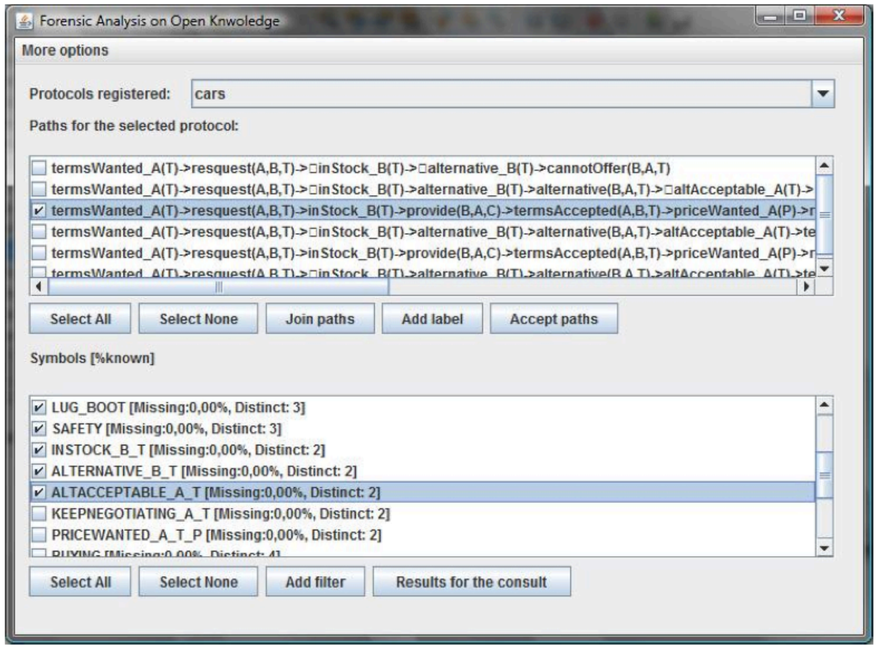
- Qualitative context mining (Serrano&MR)
- Relate constraints in protocols to outcomes
- Can be used for
  - predicting outcomes and adjusting strategies
  - identifying misaligned constraint interpretations
  - deriving qualitative trust and reputation measures
- A much more generic, simpler view of interaction frames

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)
```

The missing link

- Bottom-up data-driven methods enable us to build models of ontologies and strategies
- Top-down specification methods enable us to structure interaction space, “compute content”, limit search space
- How can we bring these two sides further together? **With the help of humans!**

The way forward: Social Computation

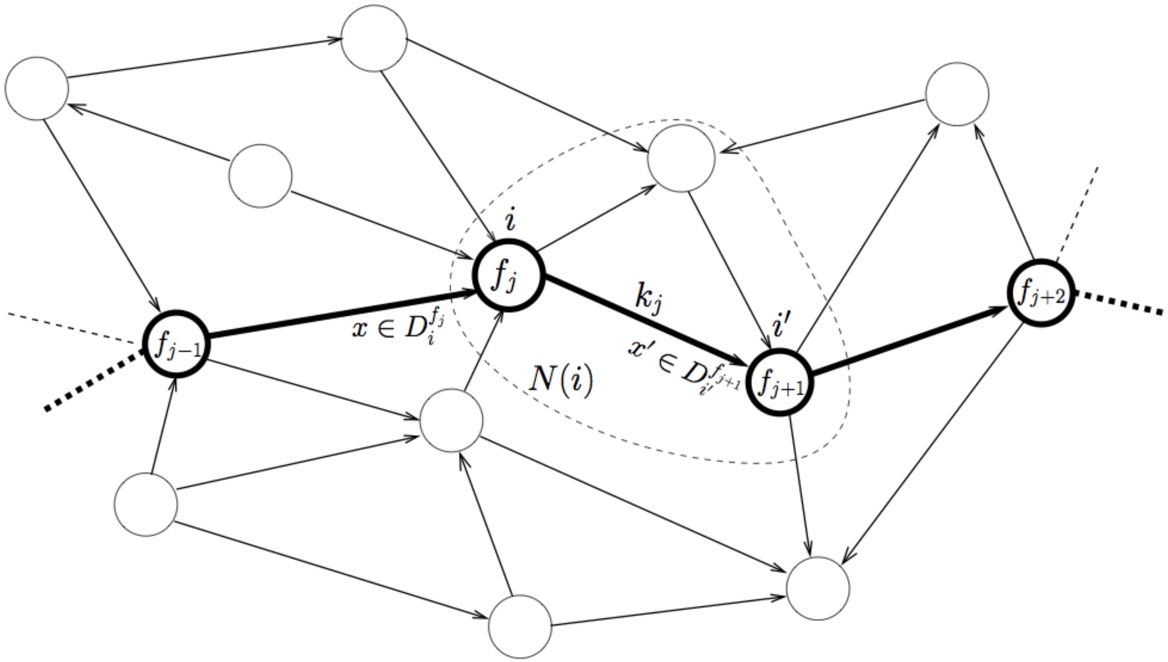
- Imagine large-scale, hybrid, heterogeneous networks of humans and machines
- Crowdsource human intelligence where computational problem is too hard
- Conversely, support human users with automation for computation tasks
- Two major projects: SmartSociety and ESSENCE

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

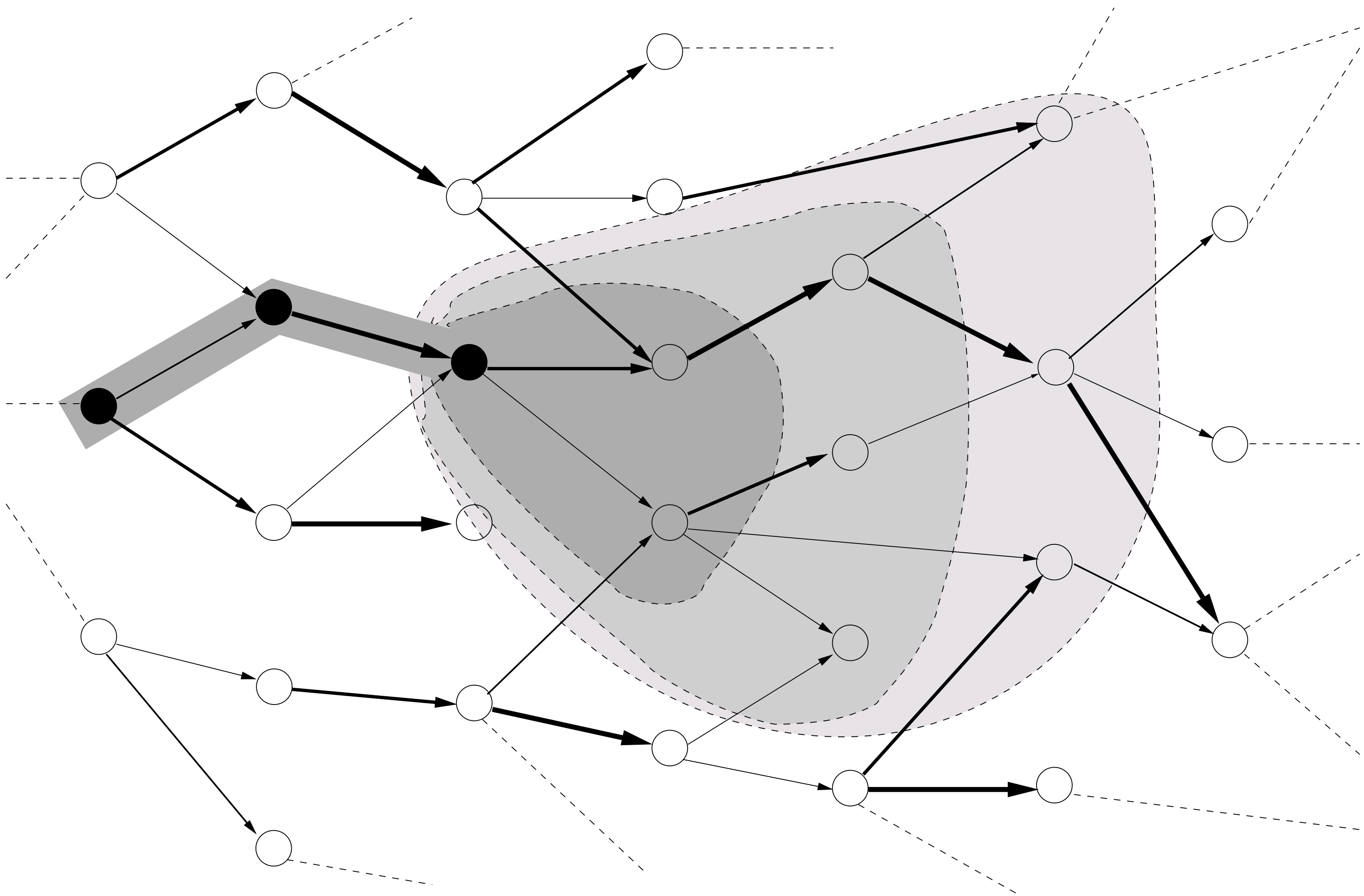
Lightweight social orchestration

- Generalised planning and identification of emergent patterns in networks of computation

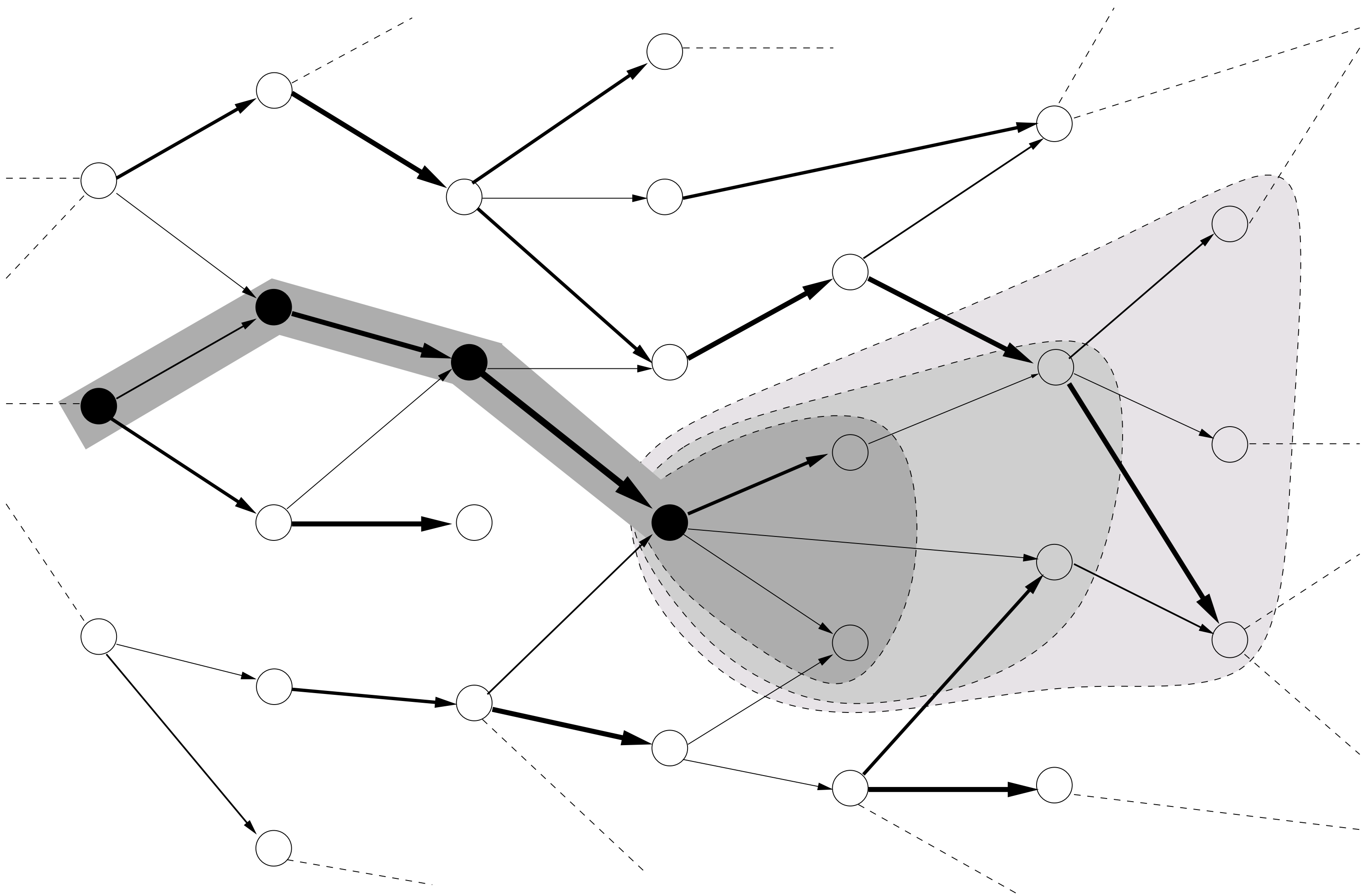


- Does this graph remind you of something?

Prediction dynamics



Prediction dynamics



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
- Our focus: 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)

Conclusions

- Presented a challenge but did not really propose a solution
- Described various methods for dealing with different parts of the problem
- Pervasive theme: tension between top-down and bottom-up methods
- The challenge lies in integrating them – new types of social computation systems may help