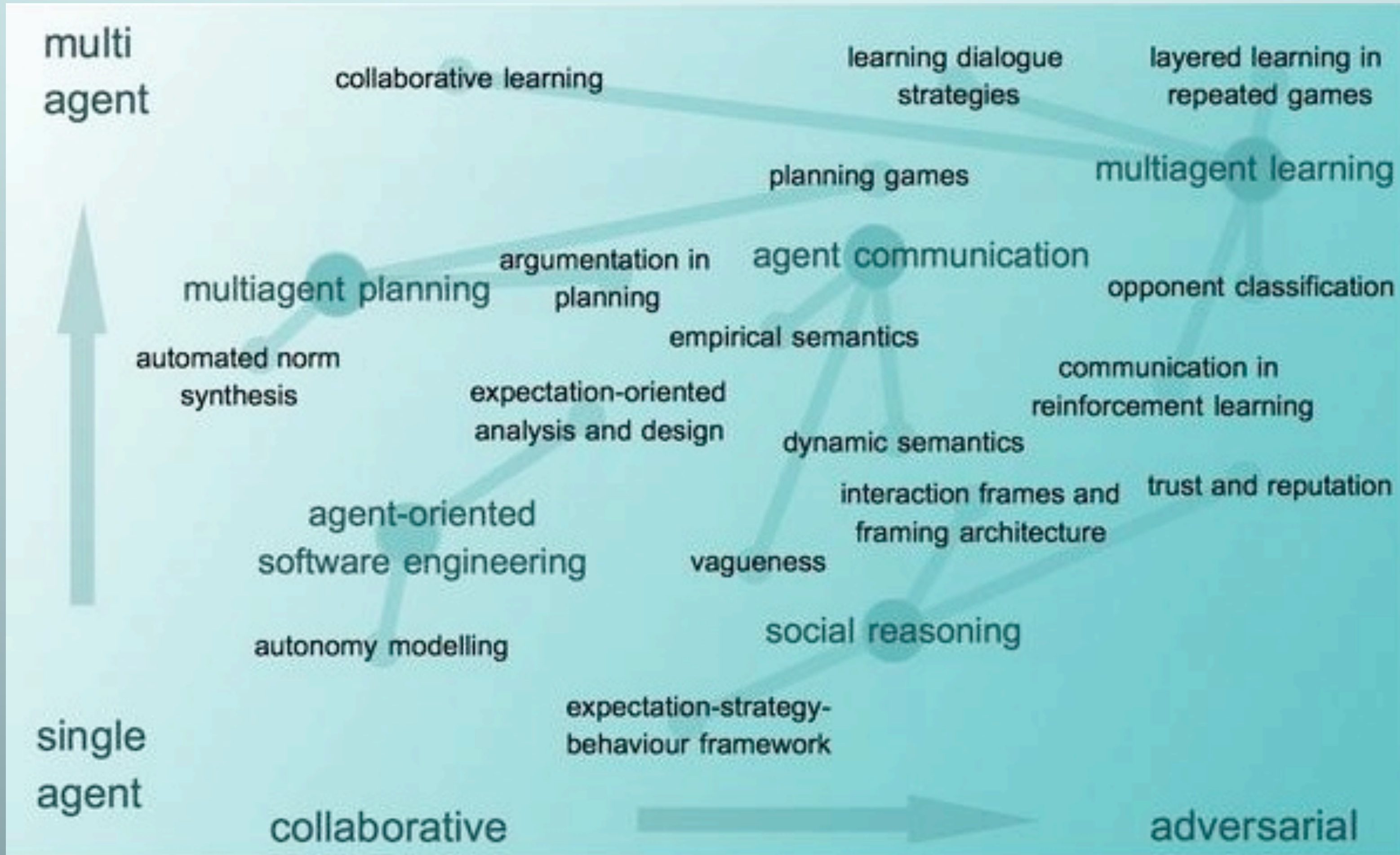
Strategic Knowledge-Based Technologies for the Web

Michael Rovatsos
(joint work with Alexandros Belesiotis, George Christelis, Matt Crosby)

Centre of Intelligent Systems and their Applications
School of Informatics, University of Edinburgh
mrovatso@inf.ed.ac.uk

Background



Visit www.cisa.inf.ed.ac.uk/agents for details

Background

- What is special about agents? Interaction in a common environment
- To make agents intelligent and autonomous, we need to automate such interaction
- Interested in knowledge-based reasoning about interaction
- Reasoning about interaction is by definition practical reasoning
- Vision: given a specification of the interaction problem, automatically synthesise behaviour

Background

- We are interested in building systems, not only specifying them formally
- We want to tell them what to achieve, not how, abstraction desirable
- This suggests using knowledge representation techniques
- Planning is the interface between KR methods and practical reasoning
- But multiagent planning underdeveloped, no simple common framework

The "Strategic" Web

- Many interactions on the Web are strategic, i.e. involve potentially divergent views and objectives of users
- Currently, very little support for this on the Web (with exception of some eCommerce applications)
- Applications rely on hardcoded policies, large-scale data mining, or manual user intervention
- Vision: represent knowledge about interests of users to be able to reason about them
- *Warning: Same dangers as Semantic Web (standards, burden of annotation, scalability etc)*

Examples

BUYER-SELLER

B: I would like an art history book.

S: Good art history books range from $35-$55.

B: I would like something cheaper.

S: There's "Art for Kids" at $15.

B: I want a book for adults.

S: There's "Art History for Dummies" at $25.

B: Great, I'll take that.

(execution follows, including payment, delivery, etc)

PEER-TO-PEER

P: I'd like to stream a music concert in high quality tomorrow night.

Q: Who will be performing?

P: It's a "best-of" transmission from a festival.

Q: I don't like watching concerts unless I know what bands are playing.

P: Could I still borrow your bandwidth?

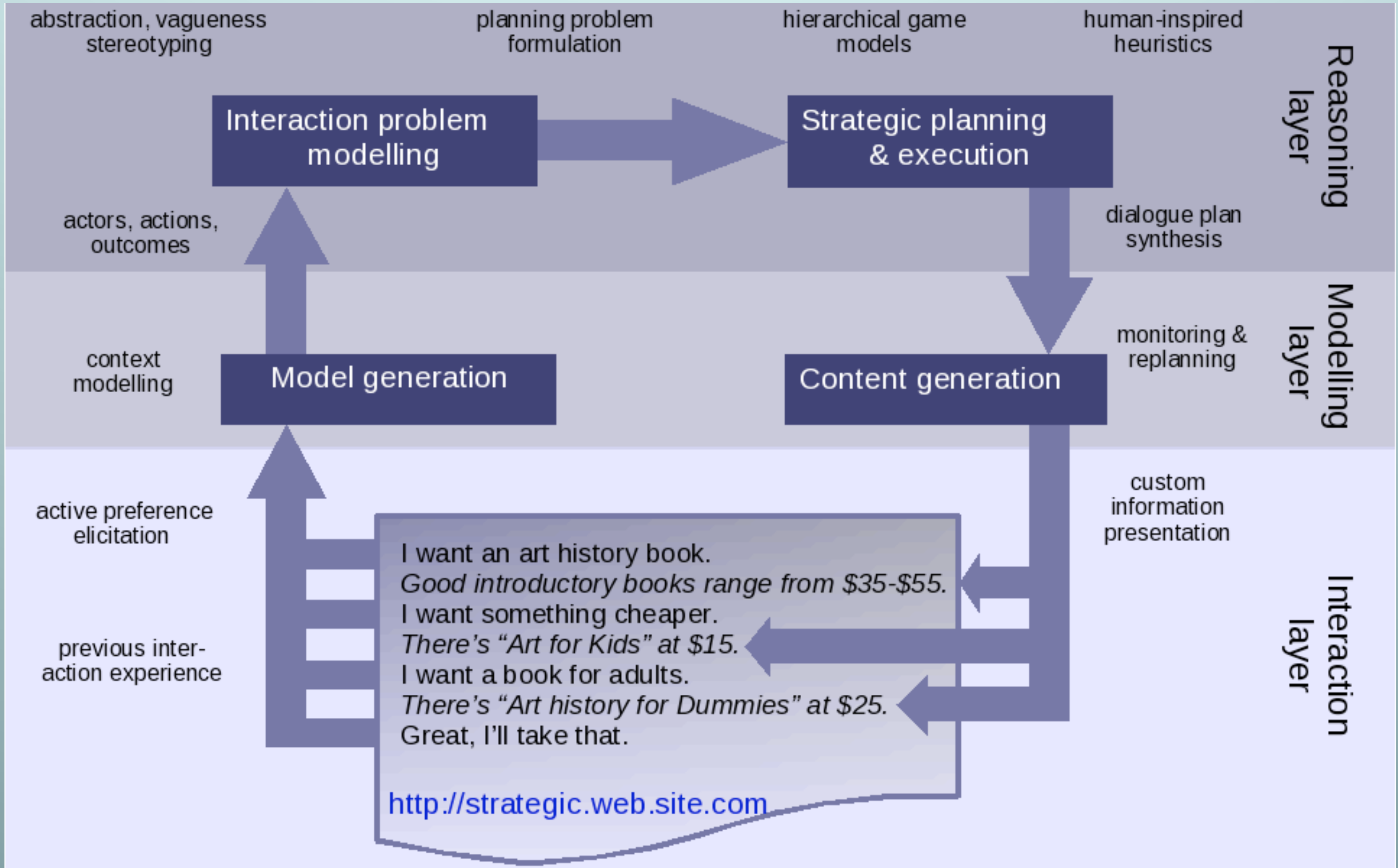
Q: OK, if you grant me prioritised access to yours for seven days after that.

(execution follows, including settings to preference in P2P system, actual streaming actions, etc)

The dialogue metaphor

- Examples deliberately looked like conversations, a simple, intuitive way of thinking about Strategic Web
- Dialogue planning metaphor covers synthesis, negotiation, and execution aspect
  – If communication actions are interpreted in a planning-based way, we should be able to plan them just like physical actions
  – But hard to decide about communication strategy before having synthesised collaborative plans
  – Actions planned for deception detection ahead of execution may affect suggested deals

The technology



Challenges

- Languages for describing strategic interaction situations on the Web
- Tractable (approximate?) inference and plan synthesis algorithms
- Preference elicitation and content presentation techniques
- Human-centric & interdisciplinary approach required

Game theory? Yes, *but*…

- Game-theoretic methods very popular currently and address the problem of reasoning about interaction
- Information in real-world domains available in relational terms (e.g. on the Web), not enumerated state actions as assumed in game theory
- Non-incremental: unable to express how a game changes when we incrementally change background knowledge
- Knowledge-based methods might be useful in lifting overly restrictive assumptions (full rationality, perfect knowledge, etc)
- Intuition: many large-scale games might be actually "easier" than we think *(this is speculative)*

Previous work

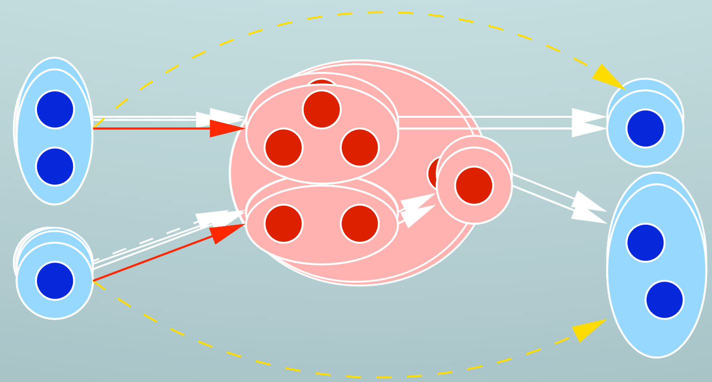
- Two examples of our current work on knowledge-based reasoning about interaction:
  - Automated norm synthesis
  - Argumentation-based conflict resolution
- Address general multiagent systems problems :
  - Setting up social laws to avoid undesirable states
  - Exchanging information to align divergent views
- From a general computer science point of view:
  - Designer-level specification of system constraints
  - Integration of distributed sources of data

Automated norm synthesis in a planning environment

- 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

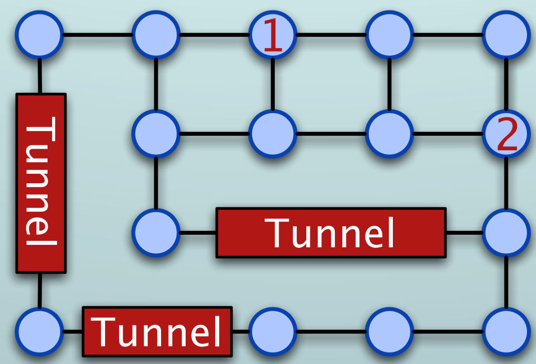
Iterated process of forward-backward search around conflict state specification:



- Not better than full state-space search in the worst case but often get lucky
- With simple additional pruning techniques search can often be cut down drastically
- Currently working on synthesising sanctions

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

  $(\{at_1(N), at_2(N'), tunnel(T), conn(N,T), conn(T,N')\}, move_1(N,T))$

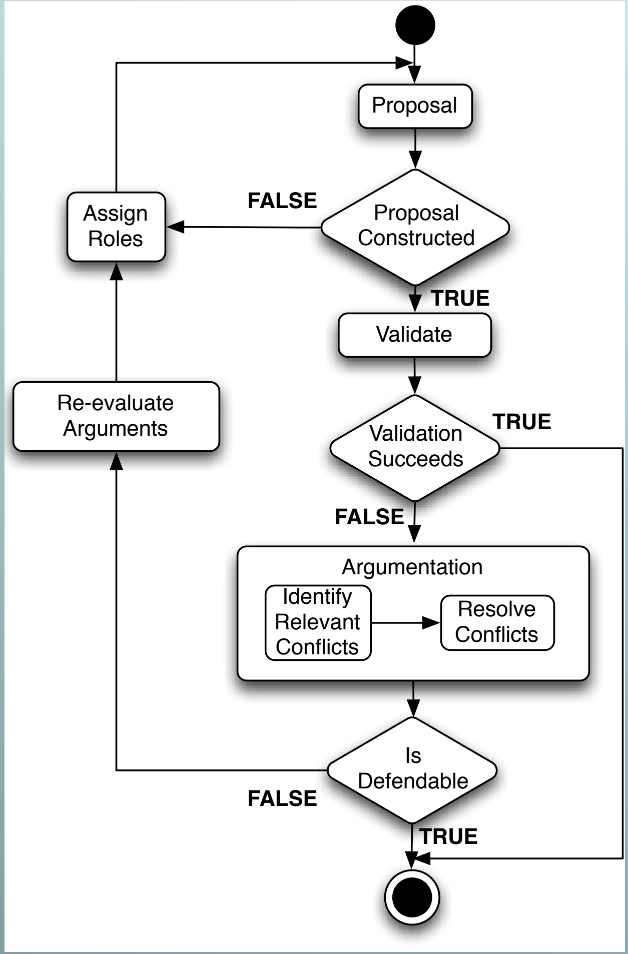
Argumentation-based conflict resolution in planning

- **Argumentation** is a method for determining the status of propositions in the presence of conflicting information
- Different acceptability-based semantics and protocols that implement these
- Rarely used for reasoning about action, our intuition is that this can be done more efficiently due to domain structure
- Suggest framework for **acceptable** planning:

  A plan P is acceptable wrt (potentially conflicting) knowledge bases KB_1 and KB_2

  iff $KB_1 \models P$ and $KB_2 \models P$

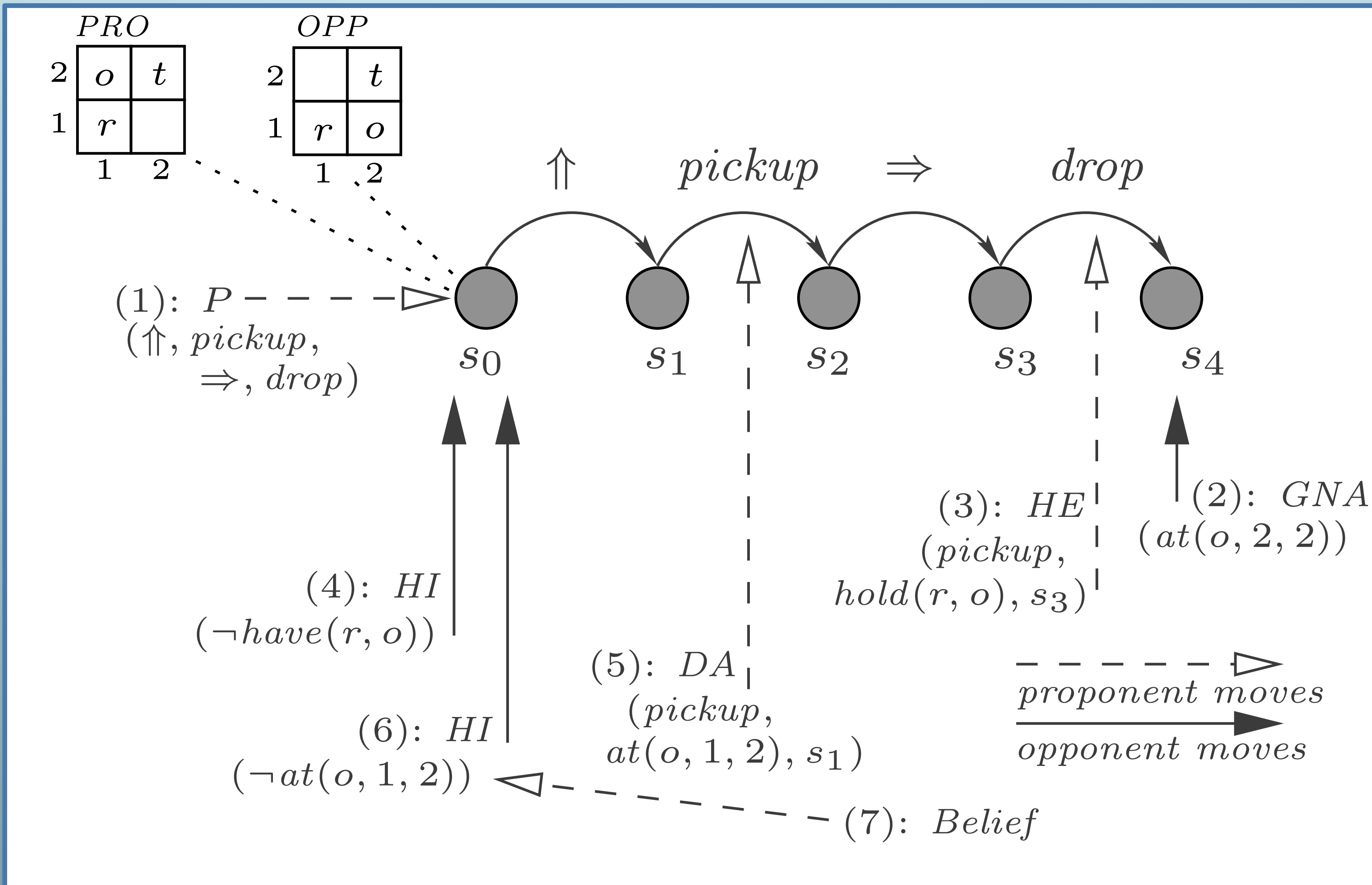
Argumentation-based conflict resolution



- Plan proposal generated by single agent (with any planner)
- Validation based on simple plan projection
- Dispute in case of disagree-ment, argumentation follows
- Ends in successful defence of initial proposal or rejection
- An alternative to generating one *P* that works under both *KBs*

Example

- **Robot gridworld domain**

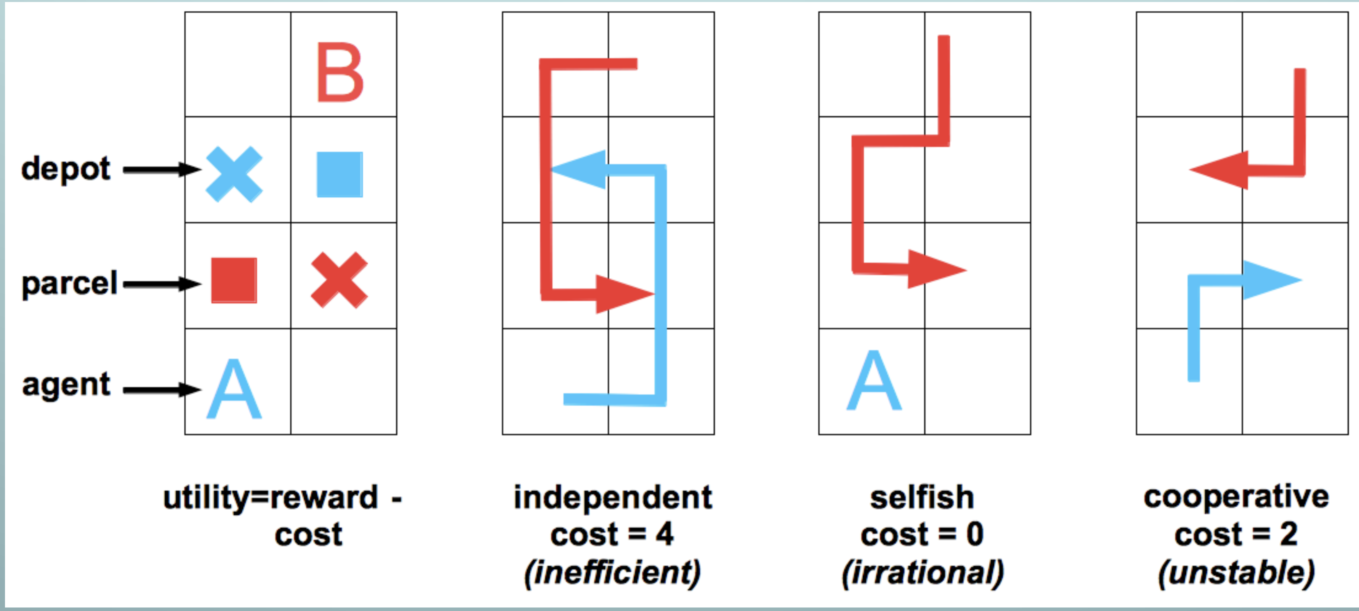


Planning Games

- Examples illustrate use of knowledge-based methods for reasoning about interaction
- But so far not concerned with strategic interaction
- Currently trying to look at more general framework of strategic multiagent planning
- Why planning? At the frontline of what is possible in terms of scalability while maintaining “knowledge-level” flavour

Examples

- Parcel delivery: the simplest (?) domain which raises interesting issues



- Fundamental question: how can domain structure help here?

Brafman/Domshlak/Engel/ Tennenholtz (IJCAI 2009)

- Introduce notion of coalition-planning game (reward for goal, cost for plan, no action = 0)
- Solution stable if no set of agents can increase utility by jointly adopting other plan
- Formally: plan π *stable* for iff no plan π exists for any subset Φ' of agents Φ such that $u_\phi(\pi') > u_\phi(\pi)$ for all ϕ in Φ'
- Present an algorithm for computing stable plans, but complexity issues (enumeration of strategies necessary)

Interesting problems

Three general problems seem interesting:

- How to compute acceptable plan given a solution criterion (in particular adapting existing planning heuristics)
- How to search plan space incrementally for generating proposals during negotiation
- How to use background knowledge to guide plan recognition and optimal response generation

Computing acceptable plans

- ***How do you find a plan that is a reasonable compromise?***
- **Initial idea: look at actions that A performs "for" B and vice versa**
- **Adjust planning heuristics by "discounting" the cost of actions done for the other**
- **This is currently being implemented in state of the art planners**

Incremental plan space search

- ***Given a joint plan P, how do you find P' that is slightly more/less acceptable?***
- **Important for negotiation: finding a selfish/selfless plan is trivial, search for proposal in between hard**
- **Hierarchical representations should help: if sub-tasks can be identified they can be re-assigned to different agents**

Guiding plan recognition and response

- ***How can knowledge about other's preferences help filter possible plans given action sequence observation*?**
- **Plan recognition can already be done in a scalable way**
- **When jointly executing an agreed plan, likely alternative execution paths are contingent on (joint) preferences**
- **Important for detecting deception or risk of deception, and responding to it**

Conclusions

- Argued for “Strategic” Web as an interesting field for agent applications
- Personal view: automated reasoning about strategic interaction is key contribution of agent technology
- Examples of previous work indicate practical reasoning algorithms are possible
- Current goal is to develop similar methods for settings of strategic interaction
- A lot of scope for doing things in a multiagent planning setting, very little previous work

Thank you. Questions?

Material based on
Christelis, MR & Petrick @ AAMAS 2009/2010
Belesiotis, MR & Rahwan @ AAMAS 2010

Find out more/get involved at
http://www.cisa.inf.ed.ac.uk/agents