Multiagent Learning: Towards a New Synthesis

Michael Rovatsos

Centre for Intelligent Systems and their Applications

AAMAS Symposium, Paris, 21st March 2005

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**
  - Large (and changing) agent populations

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**
  - Large (and changing) agent populations
  - Heterogeneous agent designs

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**
  - Large (and changing) agent populations
  - Heterogeneous agent designs
  - Internal state not accessible to other agents

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**
  - Large (and changing) agent populations
  - Heterogeneous agent designs
  - Internal state not accessible to other agents
  - Agents potentially self-interested or malicious

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**
  - Large (and changing) agent populations
  - Heterogeneous agent designs
  - Internal state not accessible to other agents
  - Agents potentially self-interested or malicious
- Example application areas:

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**
  - Large (and changing) agent populations
  - Heterogeneous agent designs
  - Internal state not accessible to other agents
  - Agents potentially self-interested or malicious
- Example application areas:
  - eCommerce, Semantic Web, Web Services, Grid computing, mobile/ubiquitous computing, P2P computing

The Open Systems Challenge

- Multiagent systems (MAS) are increasingly moving towards **open systems**
  - Large (and changing) agent populations
  - Heterogeneous agent designs
  - Internal state not accessible to other agents
  - Agents potentially self-interested or malicious
- Example application areas:
  - eCommerce, Semantic Web, Web Services, Grid computing, mobile/ubiquitous computing, P2P computing
- This is true regardless of our highbrow academic theories of agents, it is happening in the real world!

The Open Systems Challenge

- Openness is inevitable:

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":
  - Uncertainty & AI

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":
  - Uncertainty & AI
  - Machine Learning

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":
  - Uncertainty & AI
  - Machine Learning
  - Game-Theoretic Approaches

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":
  - Uncertainty & AI – don't rely on your beliefs!
  - Machine Learning
  - Game-Theoretic Approaches

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":
  - Uncertainty & AI – don't rely on your beliefs!
  - Machine Learning – use your experience!
  - Game-Theoretic Approaches

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":
  - Uncertainty & AI – don't rely on your beliefs!
  - Machine Learning – use your experience!
  - Game-Theoretic Approaches – think strategically!

The Open Systems Challenge

- Openness is inevitable:
  - We cannot impose restrictions on agent autonomy
  - We cannot predict the behaviour of the system
- What are others doing about this?
- Very few areas in AI can cope well with "openness":
  - Uncertainty & AI – don't rely on your beliefs!
  - Machine Learning – use your experience!
  - Game-Theoretic Approaches – think strategically!
- How about multiagent learning?

The Case for Multiagent Learning

- Multiagent learning (MAL) incorporates these aspects

The Case for Multiagent Learning

- Multiagent learning (MAL) incorporates these aspects
  - Treat current model of the world as a hypothesis

The Case for Multiagent Learning

- Multiagent learning (MAL) incorporates these aspects
  - Treat current model of the world as a hypothesis
  - Built-in flexibility: try to improve knowledge with experience

The Case for Multiagent Learning

- Multiagent learning (MAL) incorporates these aspects
  - Treat current model of the world as a hypothesis
  - Built-in flexibility: try to improve knowledge with experience
  - Active learning involves strategic decision-making

The Case for Multiagent Learning

- Multiagent learning (MAL) incorporates these aspects
  - Treat current model of the world as a hypothesis
  - Built-in flexibility: try to improve knowledge with experience
  - Active learning involves strategic decision-making

➡ MAL should be ideally suited for open systems!

The Case for Multiagent Learning

- Multiagent learning (MAL) incorporates these aspects
  - Treat current model of the world as a hypothesis
  - Built-in flexibility: try to improve knowledge with experience
  - Active learning involves strategic decision-making

➡ MAL should be ideally suited for open systems!

- And yet, is MAL achieving its full potential as this would lead us to expect?

The Case for Multiagent Learning

- Multiagent learning (MAL) incorporates these aspects
  - Treat current model of the world as a hypothesis
  - Built-in flexibility: try to improve knowledge with experience
  - Active learning involves strategic decision-making

➡ MAL should be ideally suited for open systems!

- And yet, is MAL achieving its full potential as this would lead us to expect?
- Certainly, quite some successes in (roughly) last ten years
  - Examples: Learning opponent models, learning organisational roles, multiagent reinforcement learning, learning in market environments, imitation learning, learning and negotiation, learning auction strategies, language evolution

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)
  - *Problems*: Can you define the key problems of MAL research?

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)
  - *Problems*: Can you define the key problems of MAL research?
    - ML: the problem of inductive inference, noise & overfitting, exploration vs. exploitation, sampling strategies etc.

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)
  - *Problems*: Can you define the key problems of MAL research?
    - ML: the problem of inductive inference, noise & overfitting, exploration vs. exploitation, sampling strategies etc.
  - *Algorithms*: Can you name a successful MAL algorithm that is widely used in MAS and whose properties are well-studied?

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)
  - *Problems*: Can you define the key problems of MAL research?
    - ML: the problem of inductive inference, noise & overfitting, exploration vs. exploitation, sampling strategies etc.
  - *Algorithms*: Can you name a successful MAL algorithm that is widely used in MAS and whose properties are well-studied?
    - ML: neural networks, reinforcement learning, ILP, decision trees, Bayesian learning etc.

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)
  - *Problems*: Can you define the key problems of MAL research?
    - ML: the problem of inductive inference, noise & overfitting, exploration vs. exploitation, sampling strategies etc.
  - *Algorithms*: Can you name a successful MAL algorithm that is widely used in MAS and whose properties are well-studied?
    - ML: neural networks, reinforcement learning, ILP, decision trees, Bayesian learning etc.
  - *Applications*: Do you know practical applications in which MAL techniques have really made a difference?

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)
  - *Problems*: Can you define the key problems of MAL research?
    - ML: the problem of inductive inference, noise & overfitting, exploration vs. exploitation, sampling strategies etc.
  - *Algorithms*: Can you name a successful MAL algorithm that is widely used in MAS and whose properties are well-studied?
    - ML: neural networks, reinforcement learning, ILP, decision trees, Bayesian learning etc.
  - *Applications*: Do you know practical applications in which MAL techniques have really made a difference?
    - ML: standard dataset repositories, Bayesian learning in user modelling, inference of biochemical rules with ILP

The Current Landscape

- Claim: **Multiagent learning is far from where it should be** (e.g. compared to general machine learning)
  - *Problems*: Can you define the key problems of MAL research?
    - ML: the problem of inductive inference, noise & overfitting, exploration vs. exploitation, sampling strategies etc.
  - *Algorithms*: Can you name a successful MAL algorithm that is widely used in MAS and whose properties are well-studied?
    - ML: neural networks, reinforcement learning, ILP, decision trees, Bayesian learning etc.
  - *Applications*: Do you know practical applications in which MAL techniques have really made a difference?
    - ML: standard dataset repositories, Bayesian learning in user modelling, inference of biochemical rules with ILP
- (Granted: ML has been around much longer, agents are much more complex and diverse than "disembodied" ML algorithms)

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!
- *Diversity*: Add (any type of) learning to any aspect of a MAS, and you obtain a different MAL flavour!

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!
- *Diversity*: Add (any type of) learning to any aspect of a MAS, and you obtain a different MAL flavour!
- *Conceptual confusion*: How can we map all work done in MAL to a single, coherent set of problems?

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!
- *Diversity*: Add (any type of) learning to any aspect of a MAS, and you obtain a different MAL flavour!
- *Conceptual confusion*: How can we map all work done in MAL to a single, coherent set of problems?
- And above all: **Lack of a unifying framework** which could be used to

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!
- *Diversity*: Add (any type of) learning to any aspect of a MAS, and you obtain a different MAL flavour!
- *Conceptual confusion*: How can we map all work done in MAL to a single, coherent set of problems?
- And above all: **Lack of a unifying framework** which could be used to
  - develop a shared understanding among MAL researchers

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!
- *Diversity*: Add (any type of) learning to any aspect of a MAS, and you obtain a different MAL flavour!
- *Conceptual confusion*: How can we map all work done in MAL to a single, coherent set of problems?
- And above all: **Lack of a unifying framework** which could be used to
  - develop a shared understanding among MAL researchers
  - come up with a set of well-defined research problems

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!
- *Diversity*: Add (any type of) learning to any aspect of a MAS, and you obtain a different MAL flavour!
- *Conceptual confusion*: How can we map all work done in MAL to a single, coherent set of problems?
- And above all: **Lack of a unifying framework** which could be used to
  - develop a shared understanding among MAL researchers
  - come up with a set of well-defined research problems
  - define canonical problem instances & performance measures

Problems

- *Complexity*: MAS are difficult to control and analyse, let alone if we add learning capabilities to them!
- *Diversity*: Add (any type of) learning to any aspect of a MAS, and you obtain a different MAL flavour!
- *Conceptual confusion*: How can we map all work done in MAL to a single, coherent set of problems?
- And above all: **Lack of a unifying framework** which could be used to
  - develop a shared understanding among MAL researchers
  - come up with a set of well-defined research problems
  - define canonical problem instances & performance measures
  - communicate our results to the wider MAS/AI community

Towards a New Synthesis

- This talk proposes such a unifying framework

Towards a New Synthesis

- This talk proposes such a unifying framework
- (Maybe surprisingly) this will not be a

Towards a New Synthesis

- This talk proposes such a unifying framework
- (Maybe surprisingly) this will not be a
  - new (class of) learning algorithm(s)

Towards a New Synthesis

- This talk proposes such a unifying framework
- (Maybe surprisingly) this will not be a
  - new (class of) learning algorithm(s)
  - a set of interesting learning problems

Towards a New Synthesis

- This talk proposes such a unifying framework
- (Maybe surprisingly) this will not be a
  - new (class of) learning algorithm(s)
  - a set of interesting learning problems
  - or a methodology for building MAL systems

Towards a New Synthesis

- This talk proposes such a unifying framework
- (Maybe surprisingly) this will not be a
  - new (class of) learning algorithm(s)
  - a set of interesting learning problems
  - or a methodology for building MAL systems
- Instead, we propose an abstract architecture for **Practical Social Reasoning Systems**

Towards a New Synthesis

- This talk proposes such a unifying framework
- (Maybe surprisingly) this will not be a
  - new (class of) learning algorithm(s)
  - a set of interesting learning problems
  - or a methodology for building MAL systems
- Instead, we propose an abstract architecture for **Practical Social Reasoning Systems**
- Introducing the **Expectation-Strategy-Behaviour** model for practical social reasoning

Towards a New Synthesis

- Line of reasoning:

Towards a New Synthesis

- Line of reasoning:
  - Models of practical reasoning systems (in particular, the BDI model) don't account for reasoning about interaction between agents

Towards a New Synthesis

- Line of reasoning:
  - Models of practical reasoning systems (in particular, the BDI model) don't account for reasoning about interaction between agents
  - We need a generic model for social reasoning to bridge the gap between cognitive and social processes

Towards a New Synthesis

- Line of reasoning:
  - Models of practical reasoning systems (in particular, the BDI model) don't account for reasoning about interaction between agents
  - We need a generic model for social reasoning to bridge the gap between cognitive and social processes
  - An abstract framework for PSRS can serve as a starting point for grounding different MAL methods in a common basic model

Towards a New Synthesis

- Line of reasoning:
  - Models of practical reasoning systems (in particular, the BDI model) don't account for reasoning about interaction between agents
  - We need a generic model for social reasoning to bridge the gap between cognitive and social processes
  - An abstract framework for PSRS can serve as a starting point for grounding different MAL methods in a common basic model
- Caution:

Towards a New Synthesis

- Line of reasoning:
  - Models of practical reasoning systems (in particular, the BDI model) don't account for reasoning about interaction between agents
  - We need a generic model for social reasoning to bridge the gap between cognitive and social processes
  - An abstract framework for PSRS can serve as a starting point for grounding different MAL methods in a common basic model
- Caution:
  - Lots of hand-waving and speculative argument involved

Towards a New Synthesis

- Line of reasoning:
  - Models of practical reasoning systems (in particular, the BDI model) don't account for reasoning about interaction between agents
  - We need a generic model for social reasoning to bridge the gap between cognitive and social processes
  - An abstract framework for PSRS can serve as a starting point for grounding different MAL methods in a common basic model
- Caution:
  - Lots of hand-waving and speculative argument involved
  - This is work in progress, shown here to generate discussion

The ESB Architecture

- Expectation-Strategy-Behaviour

The ESB Architecture

- Expectation-Strategy-Behaviour
- Key ideas:

The ESB Architecture

- Expectation-Strategy-Behaviour
- Key ideas:
  - Models of agents' interaction behaviour are stored as expectations and updated with new observations

The ESB Architecture

- Expectation-Strategy-Behaviour
- Key ideas:
  - Models of agents' interaction behaviour are stored as expectations and updated with new observations
  - Set of current expectations creates a strategy space

The ESB Architecture

- Expectation-Strategy-Behaviour
- Key ideas:
  - Models of agents' interaction behaviour are stored as expectations and updated with new observations
  - Set of current expectations creates a strategy space
  - Own behaviour chosen from these strategies in accordance with agent's goals

The ESB Architecture

- Expectation-Strategy-Behaviour
- Key ideas:
  - Models of agents' interaction behaviour are stored as expectations and updated with new observations
  - Set of current expectations creates a strategy space
  - Own behaviour chosen from these strategies in accordance with agent's goals
- Concept of expectation used to bridge gap between cognitive and social system layer

The ESB Architecture

- Expectation-Strategy-Behaviour
- Key ideas:
  - Models of agents' interaction behaviour are stored as expectations and updated with new observations
  - Set of current expectations creates a strategy space
  - Own behaviour chosen from these strategies in accordance with agent's goals
- Concept of expectation used to bridge gap between cognitive and social system layer
- Suitable for integration with the BDI architecture

Expectations

Informal definition:

> *An expectation is a* **conditional prediction** *whose fulfillment will be* **verified** *and* **reacted** *upon.*

Expectations

Informal definition:

> *An expectation is a* **conditional prediction** *whose fulfillment will be* **verified** *and* **reacted** *upon.*

Semi-formal description:

> *We write* $(EXP\ a\ C\ E\ \varphi\ \rho^+\ \rho^-)$ *iff agent a expects E to hold true under condition C, and is going to verify this using test* φ*. If the expectation is fulfilled he will react with* ρ^+*, otherwise with* ρ^-*.*

Expectations

Informal definition:

> *An expectation is a* **conditional prediction** *whose fulfillment will be* **verified** *and* **reacted** *upon.*

Semi-formal description:

> *We write* $(EXP\ a\ C\ E\ \varphi\ \rho^+\ \rho^-)$ *iff agent a expects E to hold true under condition C, and is going to verify this using test* φ. *If the expectation is fulfilled he will react with* ρ^+, *otherwise with* ρ^-.

(Preliminary) formal definition:

$$
\begin{aligned}
(EXP\ a\ C\ E\ \varphi\ \rho^+\ \rho^-) \Leftrightarrow & (BEL\ a\ [(BEL\ a\ \varphi \wedge C) \Rightarrow (BEL\ a\ E) \wedge (INT\ a\ \rho^+)]) \\
& \wedge (BEL\ a\ [(BEL\ a\ \neg\varphi \wedge C) \Rightarrow (INT\ a\ \rho^-)])
\end{aligned}
$$

Expectations

- Example:

$$(EXP\ a\ (DO\ a\ request(A,B,X))\ (DO\ B\ X)\ Done(X)\ nil\ retract)$$

Expectations

- Example:

  $$(EXP\ a\ (DO\ a\ request(A, B, X))\ (DO\ B\ X)\ Done(X)\ nil\ retract)$$

- This "formalisation" only used for illustration purposes, any cause-and-effect model will do (in particular, probabilistic representations will be more suitable in many cases)

Expectations

- Example:

$$(EXP\ a\,(DO\ a\ request(A,B,X))\,(DO\ B\ X)\ Done(X)\ nil\ retract)$$

- This "formalisation" only used for illustration purposes, any cause-and-effect model will do (in particular, probabilistic representations will be more suitable in many cases)
- Some remarks:

Expectations

- Example:

$$(EXP\ a\ (DO\ a\ request(A, B, X))\ (DO\ B\ X)\ Done(X)\ nil\ retract)$$

- This "formalisation" only used for illustration purposes, any cause-and-effect model will do (in particular, probabilistic representations will be more suitable in many cases)
- Some remarks:
  - E will usually be an action expression, but can also contain others' mental states (or expectations)

Expectations

- Example:

$$(EXP\ a\ (DO\ a\ request(A, B, X))\ (DO\ B\ X)\ Done(X)\ nil\ retract)$$

- This “formalisation” only used for illustration purposes, any cause-and-effect model will do (in particular, probabilistic representations will be more suitable in many cases)
- Some remarks:
  - E will usually be an action expression, but can also contain others' mental states (or expectations)
  - However, “mentalism” is prevented because conditions for inferring others' mental states are made explicit

Expectations

- Example:

$$(EXP\ a\ (DO\ a\ request(A, B, X))\ (DO\ B\ X)\ Done(X)\ nil\ retract)$$

- This "formalisation" only used for illustration purposes, any cause-and-effect model will do (in particular, probabilistic representations will be more suitable in many cases)
- Some remarks:
  - E will usually be an action expression, but can also contain others' mental states (or expectations)
  - However, "mentalism" is prevented because conditions for inferring others' mental states are made explicit
  - If E is an action expression, φ will simply be defined as an observation of these actions (or their consequences)

Expectations

- Example:

$$(EXP\ a\ (DO\ a\ request(A, B, X))\ (DO\ B\ X)\ Done(X)\ nil\ retract)$$

- This "formalisation" only used for illustration purposes, any cause-and-effect model will do (in particular, probabilistic representations will be more suitable in many cases)
- Some remarks:
  - E will usually be an action expression, but can also contain others' mental states (or expectations)
  - However, "mentalism" is prevented because conditions for inferring others' mental states are made explicit
  - If E is an action expression, φ will simply be defined as an observation of these actions (or their consequences)
  - Often ρ^+/ρ^- will simply consist of strengthening/weakening (or even retracting) the expectation, but can also involve overt action

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)
  - **self-referential** (which – unlike normal belief – permits agents to change them themselves)

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)
  - **self-referential** (which – unlike normal belief – permits agents to change them themselves)
  - **recursive** (expectations towards the reasoning agent herself)

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)
  - **self-referential** (which – unlike normal belief – permits agents to change them themselves)
  - **recursive** (expectations towards the reasoning agent herself)
  - **generalised** (valid for whole sets of agents/actions, especially in the case of **communicative** expectations)

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)
  - **self-referential** (which – unlike normal belief – permits agents to change them themselves)
  - **recursive** (expectations towards the reasoning agent herself)
  - **generalised** (valid for whole sets of agents/actions, especially in the case of **communicative** expectations)
- This makes them essential in reasoning about open systems!

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)
  - **self-referential** (which – unlike normal belief – permits agents to change them themselves)
  - **recursive** (expectations towards the reasoning agent herself)
  - **generalised** (valid for whole sets of agents/actions, especially in the case of **communicative** expectations)
- This makes them essential in reasoning about open systems!
- Strong relationship to

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)
  - **self-referential** (which – unlike normal belief – permits agents to change them themselves)
  - **recursive** (expectations towards the reasoning agent herself)
  - **generalised** (valid for whole sets of agents/actions, especially in the case of **communicative** expectations)
- This makes them essential in reasoning about open systems!
- Strong relationship to
  - non-monotonic reasoning and truth maintenance systems

Expectations

- Essentially expectations are a special kind of belief, but conditional upon the verification condition and tied to consequences
- But why should this be useful for practical social reasoning?
- Expectations can be
  - **adaptive** (and hence grounded in observation)
  - **self-referential** (which – unlike normal belief – permits agents to change them themselves)
  - **recursive** (expectations towards the reasoning agent herself)
  - **generalised** (valid for whole sets of agents/actions, especially in the case of **communicative** expectations)
- This makes them essential in reasoning about open systems!
- Strong relationship to
  - non-monotonic reasoning and truth maintenance systems
  - empirical agent communication semantics

Strategies

- At any given time, the current set of expectations defines a **strategy space**

Strategies

- At any given time, the current set of expectations defines a **strategy space**
- Results from space of actions that will lead to (non-)fulfillment of verification conditions

Strategies

- At any given time, the current set of expectations defines a **strategy space**
- Results from space of actions that will lead to (non-)fulfillment of verification conditions
- Important properties of strategies:

Strategies

- At any given time, the current set of expectations defines a **strategy space**
- Results from space of actions that will lead to (non-)fulfillment of verification conditions
- Important properties of strategies:
  - Strategies concerns others' actions as much as one's own

Strategies

- At any given time, the current set of expectations defines a **strategy space**
- Results from space of actions that will lead to (non-)fulfillment of verification conditions
- Important properties of strategies:
  - Strategies concerns others' actions as much as one's own
  - Not all different action(s) (sequences) are different strategies, effect on expectations is what matters

Strategies

- At any given time, the current set of expectations defines a **strategy space**
- Results from space of actions that will lead to (non-)fulfillment of verification conditions
- Important properties of strategies:
  - Strategies concerns others' actions as much as one's own
  - Not all different action(s) (sequences) are different strategies, effect on expectations is what matters
  - Strategy space includes potential effects on expectations

Strategies

- At any given time, the current set of expectations defines a **strategy space**
- Results from space of actions that will lead to (non-)fulfillment of verification conditions
- Important properties of strategies:
  - Strategies concerns others' actions as much as one's own
  - Not all different action(s) (sequences) are different strategies, effect on expectations is what matters
  - Strategy space includes potential effects on expectations
  - Mostly, however, identifying fixed points in the expectation update function will not be feasible or too complex
    ⇒ restrict analysis to an "expectation horizon"

Strategies

- At any given time, the current set of expectations defines a **strategy space**
- Results from space of actions that will lead to (non-)fulfillment of verification conditions
- Important properties of strategies:
  - Strategies concerns others' actions as much as one's own
  - Not all different action(s) (sequences) are different strategies, effect on expectations is what matters
  - Strategy space includes potential effects on expectations
  - Mostly, however, identifying fixed points in the expectation update function will not be feasible or too complex
    ➡ restrict analysis to an "expectation horizon"
- Strategies define the vocabulary of behaviours that may affect expectations so that an assessment of the desirability of these behaviours can follow

Behaviours

- After analysing different stategies (of others and oneself), agents determine their behaviour

Behaviours

- After analysing different stategies (of others and oneself), agents determine their behaviour
- But things are not that simple!

Behaviours

- After analysing different stategies (of others and oneself), agents determine their behaviour
- But things are not that simple!
- As far as own strategies are concerned, the agent can pick a strategy (like intentions are selected from desires in the BDI model) ➡ but how about what others will do?

Behaviours

- After analysing different stategies (of others and oneself), agents determine their behaviour
- But things are not that simple!
- As far as own strategies are concerned, the agent can pick a strategy (like intentions are selected from desires in the BDI model) ➡ but how about what others will do?
- No general statements can be made here, this depends on the implementation

Behaviours

- After analysing different stategies (of others and oneself), agents determine their behaviour
- But things are not that simple!
- As far as own strategies are concerned, the agent can pick a strategy (like intentions are selected from desires in the BDI model) ➡ but how about what others will do?
- No general statements can be made here, this depends on the implementation
  - Simple strategies: consider only opponents' most likely/worst-case strategy and adjust own strategy to this

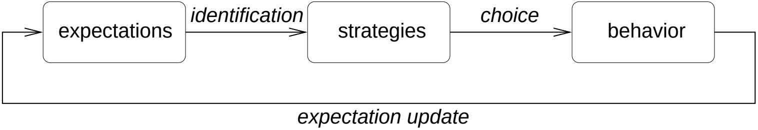
Behaviours

- After analysing different stategies (of others and oneself), agents determine their behaviour
- But things are not that simple!
- As far as own strategies are concerned, the agent can pick a strategy (like intentions are selected from desires in the BDI model) ➡ but how about what others will do?
- No general statements can be made here, this depends on the implementation
  - Simple strategies: consider only opponents' most likely/worst-case strategy and adjust own strategy to this
  - The range and temporal scope of validity of a chosen strategy may vary (when will strategies be reconsidered?)

Behaviours

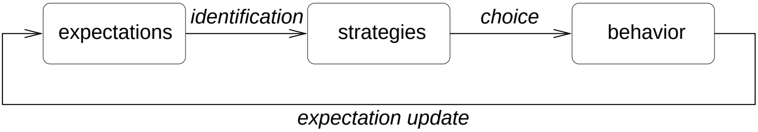
- After analysing different stategies (of others and oneself), agents determine their behaviour
- But things are not that simple!
- As far as own strategies are concerned, the agent can pick a strategy (like intentions are selected from desires in the BDI model) ➡ but how about what others will do?
- No general statements can be made here, this depends on the implementation
  - Simple strategies: consider only opponents' most likely/worst-case strategy and adjust own strategy to this
  - The range and temporal scope of validity of a chosen strategy may vary (when will strategies be reconsidered?)
- In any case we end up with some behavioural constraints imposed on the agent herself (and, implicitly, on that of others)

The ESB Feedback Loop



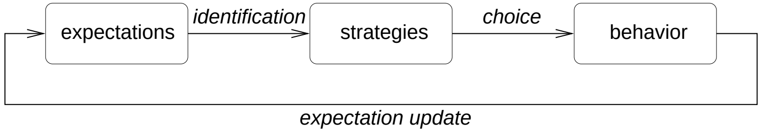
- Expectations generate strategies, these generate behaviours, and the observation of these behaviours leads to new expectations

The ESB Feedback Loop



- Expectations generate strategies, these generate behaviours, and the observation of these behaviours leads to new expectations
- A closer look reveals that this nothing but a learning loop for **interaction learning**

The ESB Feedback Loop



- Expectations generate strategies, these generate behaviours, and the observation of these behaviours leads to new expectations
- A closer look reveals that this nothing but a learning loop for **interaction learning**
- Central insight: ESB as an abstract model for practical social reasoning *necessitates* learning

From MAL to ESB (and back)

- LAYLA (Rovatsos, 1999): A layered learning agent architecture for adaptive behaviour in repeated games

From MAL to ESB (and back)

- LAYLA (Rovatsos, 1999): A layered learning agent architecture for adaptive behaviour in repeated games
  - Three layers: one to learn the payoff matrix (ANN), one to learn best response strategies (GA) and one to learn the cooperation potential given each other's preferences (POM)

From MAL to ESB (and back)

- LAYLA (Rovatsos, 1999): A layered learning agent architecture for adaptive behaviour in repeated games
  - Three layers: one to learn the payoff matrix (ANN), one to learn best response strategies (GA) and one to learn the cooperation potential given each other's preferences (POM)
- AdHoc (Rovatsos and Wolf, 2002): A heuristic for opponent classification in repeated games

From MAL to ESB (and back)

- LAYLA (Rovatsos, 1999): A layered learning agent architecture for adaptive behaviour in repeated games
  - Three layers: one to learn the payoff matrix (ANN), one to learn best response strategies (GA) and one to learn the cooperation potential given each other's preferences (POM)
- AdHoc (Rovatsos and Wolf, 2002): A heuristic for opponent classification in repeated games
  - Grows a variable set of classes of opponents and learns an optimal strategy against each one of them (US-L* + Q-learning + new classification heuristic)

From MAL to ESB (and back)

- LAYLA (Rovatsos, 1999): A layered learning agent architecture for adaptive behaviour in repeated games
  - Three layers: one to learn the payoff matrix (ANN), one to learn best response strategies (GA) and one to learn the cooperation potential given each other's preferences (POM)
- AdHoc (Rovatsos and Wolf, 2002): A heuristic for opponent classification in repeated games
  - Grows a variable set of classes of opponents and learns an optimal strategy against each one of them (US-L* + Q-learning + new classification heuristic)
- InFFrA (Rovatsos, 2004): Learning interaction patterns

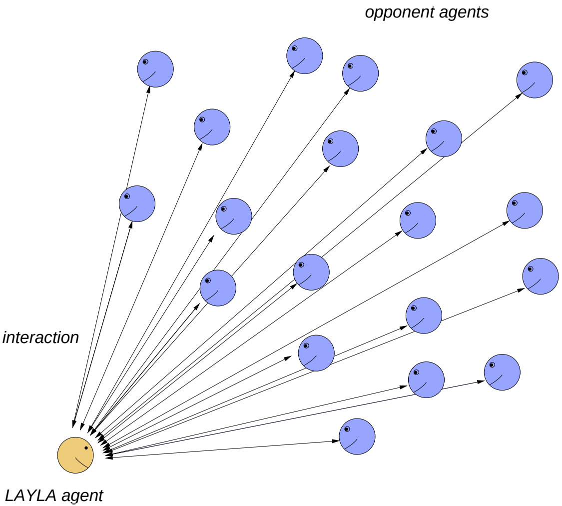
From MAL to ESB (and back)

- LAYLA (Rovatsos, 1999): A layered learning agent architecture for adaptive behaviour in repeated games
  - Three layers: one to learn the payoff matrix (ANN), one to learn best response strategies (GA) and one to learn the cooperation potential given each other's preferences (POM)
- AdHoc (Rovatsos and Wolf, 2002): A heuristic for opponent classification in repeated games
  - Grows a variable set of classes of opponents and learns an optimal strategy against each one of them (US-L* + Q-learning + new classification heuristic)
- InFFrA (Rovatsos, 2004): Learning interaction patterns
  - Patterns describe conversation runs together with context conditions, optimisation uses hierarchical RL + instance-based methods + cluster validation techniques to abstract from observed individual instances

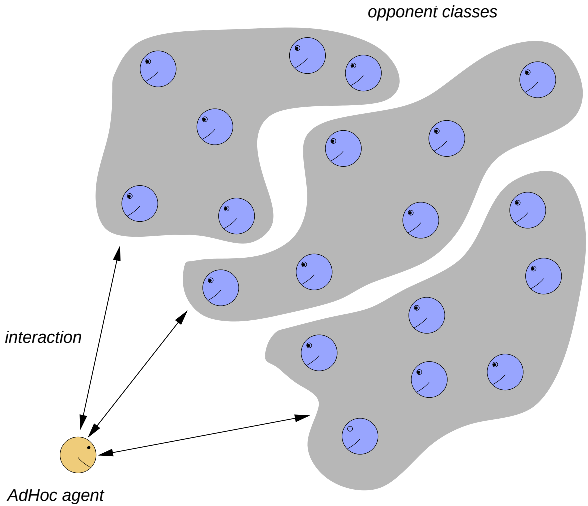
From MAL to ESB (and back)

- LAYLA (Rovatsos, 1999): A layered learning agent architecture for adaptive behaviour in repeated games
  - Three layers: one to learn the payoff matrix (ANN), one to learn best response strategies (GA) and one to learn the cooperation potential given each other's preferences (POM)
- AdHoc (Rovatsos and Wolf, 2002): A heuristic for opponent classification in repeated games
  - Grows a variable set of classes of opponents and learns an optimal strategy against each one of them (US-L* + Q-learning + new classification heuristic)
- InFFrA (Rovatsos, 2004): Learning interaction patterns
  - Patterns describe conversation runs together with context conditions, optimisation uses hierarchical RL + instance-based methods + cluster validation techniques to abstract from observed individual instances
- Gradual shift towards expectation-centered learning methods

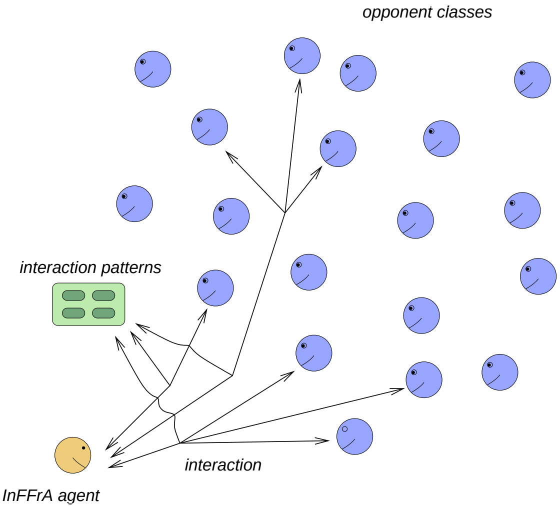
From MAL to ESB



From MAL to ESB



From MAL to ESB



Expressiveness

- Three categories of methods for reasoning about interaction

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)
  - Problem: Not feasible in open systems

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)
  - Problem: Not feasible in open systems
- **Objectivist**: impose some kind of deontic apparatus on the system to regulate agent behaviour

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)
  - Problem: Not feasible in open systems
- **Objectivist**: impose some kind of deontic apparatus on the system to regulate agent behaviour
  - Methods abound: commitments and conventions, norms, roles, deontic logics, organisational approaches, electronic institutions

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)
  - Problem: Not feasible in open systems
- **Objectivist**: impose some kind of deontic apparatus on the system to regulate agent behaviour
  - Methods abound: commitments and conventions, norms, roles, deontic logics, organisational approaches, electronic institutions
  - Problem: no unifying model, no grounding in agent cognition

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)
  - Problem: Not feasible in open systems
- **Objectivist**: impose some kind of deontic apparatus on the system to regulate agent behaviour
  - Methods abound: commitments and conventions, norms, roles, deontic logics, organisational approaches, electronic institutions
  - Problem: no unifying model, no grounding in agent cognition
- **Rationalistic**: devise interaction mechanisms such that system objectives are achieved despite agents' self-interest

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)
  - Problem: Not feasible in open systems
- **Objectivist**: impose some kind of deontic apparatus on the system to regulate agent behaviour
  - Methods abound: commitments and conventions, norms, roles, deontic logics, organisational approaches, electronic institutions
  - Problem: no unifying model, no grounding in agent cognition
- **Rationalistic**: devise interaction mechanisms such that system objectives are achieved despite agents' self-interest
  - Examples: game-theoretic approaches (mechanism design, etc.)

Expressiveness

- Three categories of methods for reasoning about interaction
- **Mentalistic**: assume a model of mental states of other agents (so that behaviour can essentially be fully predicted)
  - Example: Mentalistic ACL semantics (e.g. in FIPA-ACL)
  - Problem: Not feasible in open systems
- **Objectivist**: impose some kind of deontic apparatus on the system to regulate agent behaviour
  - Methods abound: commitments and conventions, norms, roles, deontic logics, organisational approaches, electronic institutions
  - Problem: no unifying model, no grounding in agent cognition
- **Rationalistic**: devise interaction mechanisms such that system objectives are achieved despite agents' self-interest
  - Examples: game-theoretic approaches (mechanism design, etc.)
  - Problem: simplification of interaction mechanisms to guarantee properties, "worst-case reasoning"

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them
- Concept of expectation can be applied to all three types of mechanisms:

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them
- Concept of expectation can be applied to all three types of mechanisms:
  - Encode assumptions about mental states, deontic frameworks, and agent rationality in expectations

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them
- Concept of expectation can be applied to all three types of mechanisms:
  - Encode assumptions about mental states, deontic frameworks, and agent rationality in expectations
- Added flexibility through adaptiveness of expectations:

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them
- Concept of expectation can be applied to all three types of mechanisms:
  - Encode assumptions about mental states, deontic frameworks, and agent rationality in expectations
- Added flexibility through adaptiveness of expectations:
  - Revise mentalistic assumptions as soon as agent behaviour indicates they are not valid

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them
- Concept of expectation can be applied to all three types of mechanisms:
  - Encode assumptions about mental states, deontic frameworks, and agent rationality in expectations
- Added flexibility through adaptiveness of expectations:
  - Revise mentalistic assumptions as soon as agent behaviour indicates they are not valid
  - Design social laws (e.g. a commitment mechanism) with a focus on handling failure to meet social requirements

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them
- Concept of expectation can be applied to all three types of mechanisms:
  - Encode assumptions about mental states, deontic frameworks, and agent rationality in expectations
- Added flexibility through adaptiveness of expectations:
  - Revise mentalistic assumptions as soon as agent behaviour indicates they are not valid
  - Design social laws (e.g. a commitment mechanism) with a focus on handling failure to meet social requirements
  - Drop rationality assumptions in mechanism design if agents behave irrationaly

Expressiveness

- ESB does not solve the basic problems, but it provides a uniform set of concepts to deal with them
- Concept of expectation can be applied to all three types of mechanisms:
  - Encode assumptions about mental states, deontic frameworks, and agent rationality in expectations
- Added flexibility through adaptiveness of expectations:
  - Revise mentalistic assumptions as soon as agent behaviour indicates they are not valid
  - Design social laws (e.g. a commitment mechanism) with a focus on handling failure to meet social requirements
  - Drop rationality assumptions in mechanism design if agents behave irrationaly
- (Of course, this remains to be shown for concrete existing approaches)

Integration with the BDI model

- All this may sound good, but how do we build ESB agents?

Integration with the BDI model

- All this may sound good, but how do we build ESB agents?
- The ESB model does not provide methods for (general) practical reasoning

Integration with the BDI model

- All this may sound good, but how do we build ESB agents?
- The ESB model does not provide methods for (general) practical reasoning
- Integration with the BDI model?

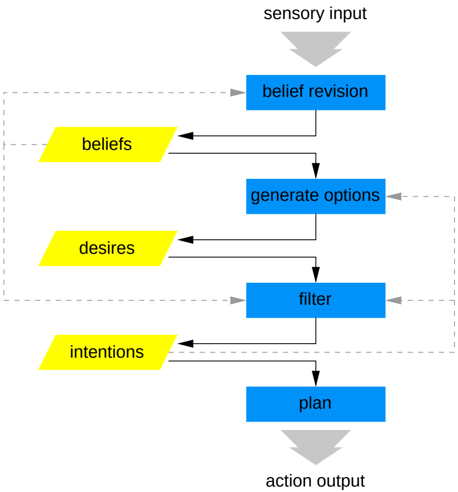
Integration with the BDI model

- All this may sound good, but how do we build ESB agents?
- The ESB model does not provide methods for (general) practical reasoning
- Integration with the BDI model?
- Interestingly, ESB and BDI look quite similar from an agent's point of view

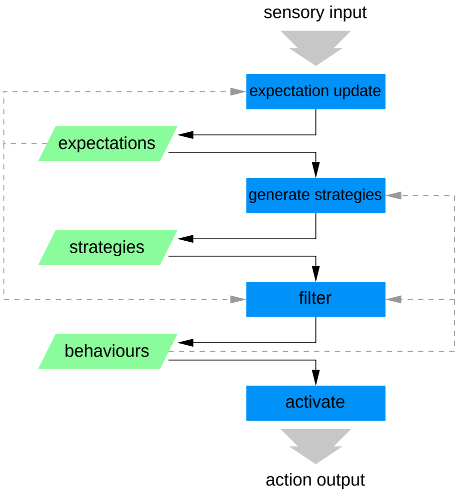
Integration with the BDI model

- All this may sound good, but how do we build ESB agents?
- The ESB model does not provide methods for (general) practical reasoning
- Integration with the BDI model?
- Interestingly, ESB and BDI look quite similar from an agent's point of view
- However, there are also some intricacies ...

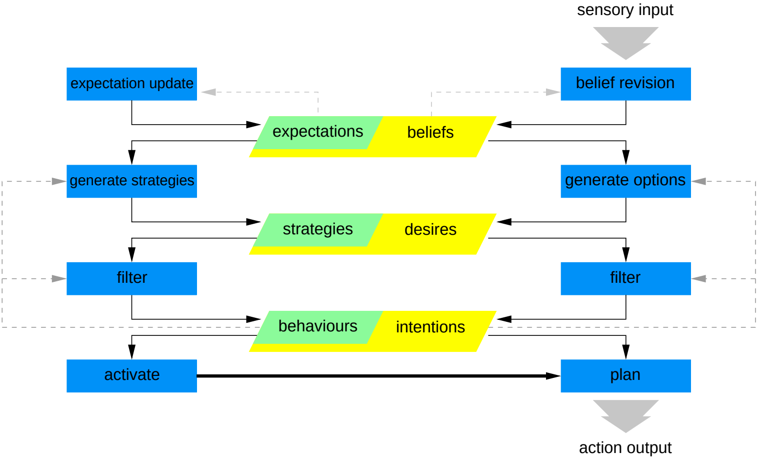
The BDI Model (in 60 seconds)



ESB – Agent Perspective



Integration: BDI^{ESB}



Introduction

The ESB Architecture

Integration with the BDI model

A Prototypical Example

- The Reputation Game
- The ESB/Learning View

Conclusions

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent
  - Agents may change names and re-enter the market

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent
  - Agents may change names and re-enter the market
  - The agent's own participation in market exchange is hard to predict

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent
  - Agents may change names and re-enter the market
  - The agent's own participation in market exchange is hard to predict
- Agents play a very simple game repeatedly:

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent
  - Agents may change names and re-enter the market
  - The agent's own participation in market exchange is hard to predict
- Agents play a very simple game repeatedly:
  - A potential buyer and seller are matched

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent
  - Agents may change names and re-enter the market
  - The agent's own participation in market exchange is hard to predict
- Agents play a very simple game repeatedly:
  - A potential buyer and seller are matched
  - They decide whether they want to make a deal or not

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent
  - Agents may change names and re-enter the market
  - The agent's own participation in market exchange is hard to predict
- Agents play a very simple game repeatedly:
  - A potential buyer and seller are matched
  - They decide whether they want to make a deal or not
  - They agree to pay and deliver the goods simultaneously

The Reputation Game

- A minimal example to illustrate how expectations interface between social behaviour, cognitive decision-making processes
- Imagine a market of "complete strangers" where agents are unlikely to interact more than once with the same partner
  - They cannot collect enough evidence about some particular opponent
  - Agents may change names and re-enter the market
  - The agent's own participation in market exchange is hard to predict
- Agents play a very simple game repeatedly:
  - A potential buyer and seller are matched
  - They decide whether they want to make a deal or not
  - They agree to pay and deliver the goods simultaneously
  - They may choose to cooperate (i.e. pay/deliver) or not

The Reputation Game

- Assume there is a temptation to defect
  ➡ Situation very similar to Prisoner's Dilemma game

The Reputation Game

- Assume there is a temptation to defect
  ➡ Situation very similar to Prisoner's Dilemma game
- There is a central reputation mechanism that measures the frequency of defection (t/t' for the two players p and p')

The Reputation Game

- Assume there is a temptation to defect
  ➡ Situation very similar to Prisoner's Dilemma game
- There is a central reputation mechanism that measures the frequency of defection (t/t' for the two players p and p')
  - Reputation value = ratio between numbers of time p defected and total purchases he participated in

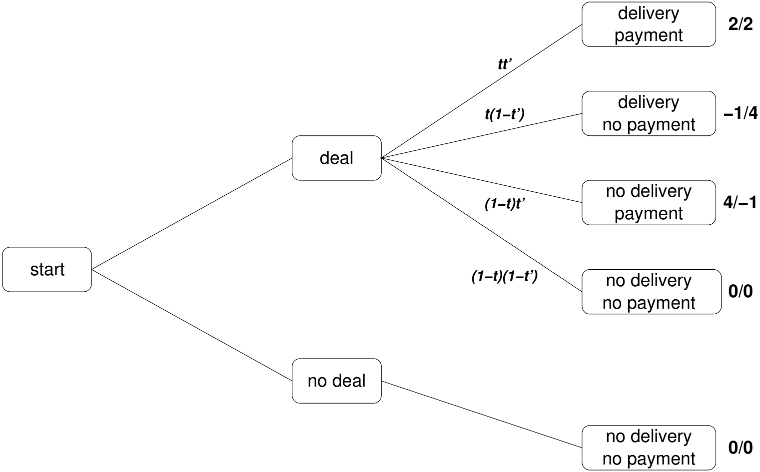
The Reputation Game

- Assume there is a temptation to defect
  ➡ Situation very similar to Prisoner's Dilemma game
- There is a central reputation mechanism that measures the frequency of defection (t/t' for the two players p and p')
  - Reputation value = ratio between numbers of time p defected and total purchases he participated in
- Resembles the rating mechanisms used on eBay, Amazon, etc.

The Reputation Game

- Assume there is a temptation to defect
  ➡ Situation very similar to Prisoner's Dilemma game
- There is a central reputation mechanism that measures the frequency of defection (t/t' for the two players p and p')
  - Reputation value = ratio between numbers of time p defected and total purchases he participated in
- Resembles the rating mechanisms used on eBay, Amazon, etc.
- Reputation value is a prototypical example for an expectation structure

The Reputation Game



The ESB perspective

- "Reputation manager": the source of expectation information in this game

The ESB perspective

- "Reputation manager": the source of expectation information in this game
- Agent strategy is a function

$$f(history, own\ reputation, other's\ reputation$$

(this is all the information that is available, after all)

The ESB perspective

- "Reputation manager": the source of expectation information in this game
- Agent strategy is a function

$$f(history, own\ reputation, other's\ reputation$$

  (this is all the information that is available, after all)
- Each agent knows other agents are taking her own reputation value into account

The ESB perspective

- "Reputation manager": the source of expectation information in this game
- Agent strategy is a function

$$f(history, \; own\ reputation, \; other's\ reputation$$

(this is all the information that is available, after all)

- Each agent knows other agents are taking her own reputation value into account
- Crucial: taking the future effect of current behaviour on expectation structures into account in decision making (ESB loop)

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of “I will cooperate”

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of "I will cooperate"
  - This is the simplest case of an "empicial semantics" approach (language consists of a single symbol)

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of "I will cooperate"
  - This is the simplest case of an "empicial semantics" approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of “I will cooperate”
  - This is the simplest case of an “empicial semantics” approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:
  - Does the reputation mechanism lead to more cooperation?

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of "I will cooperate"
  - This is the simplest case of an "empicial semantics" approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:
  - Does the reputation mechanism lead to more cooperation?
  - Or can agents learn to "beat" the reputation mechanism?

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of “I will cooperate”
  - This is the simplest case of an “empicial semantics” approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:
  - Does the reputation mechanism lead to more cooperation?
  - Or can agents learn to “beat” the reputation mechanism?
  - If so, can we improve the mechanism itself over time?

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of "I will cooperate"
  - This is the simplest case of an "empicial semantics" approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:
  - Does the reputation mechanism lead to more cooperation?
  - Or can agents learn to "beat" the reputation mechanism?
  - If so, can we improve the mechanism itself over time?
  - What should initial reputation values be?

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of "I will cooperate"
  - This is the simplest case of an "empicial semantics" approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:
  - Does the reputation mechanism lead to more cooperation?
  - Or can agents learn to "beat" the reputation mechanism?
  - If so, can we improve the mechanism itself over time?
  - What should initial reputation values be?
  - How about the update mechanism?

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of “I will cooperate”
  - This is the simplest case of an “empicial semantics” approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:
  - Does the reputation mechanism lead to more cooperation?
  - Or can agents learn to “beat” the reputation mechanism?
  - If so, can we improve the mechanism itself over time?
  - What should initial reputation values be?
  - How about the update mechanism?
  - Can an agent learn whether to trust the mechanism or not?

The Learning Perspective

- Essentially, the reputation value is an expectation structure that captures the social semantics of "I will cooperate"
  - This is the simplest case of an "empicial semantics" approach (language consists of a single symbol)
- Offers a great variety of interesting learning problems:
  - Does the reputation mechanism lead to more cooperation?
  - Or can agents learn to "beat" the reputation mechanism?
  - If so, can we improve the mechanism itself over time?
  - What should initial reputation values be?
  - How about the update mechanism?
  - Can an agent learn whether to trust the mechanism or not?
  - How about exchanging information about expectations among agents?

Summary

- Considering the challenges of open systems, multiagent learning has a huge potential

Summary

- Considering the challenges of open systems, multiagent learning has a huge potential
- Yet a simple, unifying framework for MAL is still amiss

Summary

- Considering the challenges of open systems, multiagent learning has a huge potential
- Yet a simple, unifying framework for MAL is still amiss
- Argued that this is due to a lack of a simple practical social reasoning architecture for agents

Summary

- Considering the challenges of open systems, multiagent learning has a huge potential
- Yet a simple, unifying framework for MAL is still amiss
- Argued that this is due to a lack of a simple practical social reasoning architecture for agents
- Proposed such an architecture based on central notion of expectations

Summary

- Considering the challenges of open systems, multiagent learning has a huge potential
- Yet a simple, unifying framework for MAL is still amiss
- Argued that this is due to a lack of a simple practical social reasoning architecture for agents
- Proposed such an architecture based on central notion of expectations
- Described its capacity to capture different types of current social reasoning mechanisms

Summary

- Considering the challenges of open systems, multiagent learning has a huge potential
- Yet a simple, unifying framework for MAL is still amiss
- Argued that this is due to a lack of a simple practical social reasoning architecture for agents
- Proposed such an architecture based on central notion of expectations
- Described its capacity to capture different types of current social reasoning mechanisms
- Some initial ideas for its integration with general practical reasoning systems

Summary

- Considering the challenges of open systems, multiagent learning has a huge potential
- Yet a simple, unifying framework for MAL is still amiss
- Argued that this is due to a lack of a simple practical social reasoning architecture for agents
- Proposed such an architecture based on central notion of expectations
- Described its capacity to capture different types of current social reasoning mechanisms
- Some initial ideas for its integration with general practical reasoning systems
- Presented a prototypical example for expectation-centered learning

A Vision for MAL Research

So, finally, here is the deal ...

- The Multiagent Learning Problem:

A Vision for MAL Research

So, finally, here is the deal ...

- The Multiagent Learning Problem:
  - Target function: a certain behaviour of the system

A Vision for MAL Research

So, finally, here is the deal . . .

- The Multiagent Learning Problem:
  - Target function: a certain behaviour of the system
  - Learning data: mutually observed agent behaviour

A Vision for MAL Research

So, finally, here is the deal . . .

- The Multiagent Learning Problem:
  - Target function: a certain behaviour of the system
  - Learning data: mutually observed agent behaviour
  - Hypothesis space: available ESB processing functions

A Vision for MAL Research

So, finally, here is the deal . . .

- The Multiagent Learning Problem:
  - Target function: a certain behaviour of the system
  - Learning data: mutually observed agent behaviour
  - Hypothesis space: available ESB processing functions
- *The Multiagent Learning endeavour should be an investigation of how expectation processing mechanisms can be devised to achieve a target functionality in a certain class of task environments and multiagent system structures*

A Vision for MAL Research

So, finally, here is the deal ...

- The Multiagent Learning Problem:
  - Target function: a certain behaviour of the system
  - Learning data: mutually observed agent behaviour
  - Hypothesis space: available ESB processing functions
- *The Multiagent Learning endeavour should be an investigation of how expectation processing mechanisms can be devised to achieve a target functionality in a certain class of task environments and multiagent system structures*
- My suspicion: many approaches are already very successful at doing this, but a shared understanding prevents focused and effective research on key issues

A Vision for MAL Research

So, finally, here is the deal . . .

- The Multiagent Learning Problem:
  - Target function: a certain behaviour of the system
  - Learning data: mutually observed agent behaviour
  - Hypothesis space: available ESB processing functions
- *The Multiagent Learning endeavour should be an investigation of how expectation processing mechanisms can be devised to achieve a target functionality in a certain class of task environments and multiagent system structures*
- My suspicion: many approaches are already very successful at doing this, but a shared understanding prevents focused and effective research on key issues
- Today I described one way of attacking this problem

The End

Thank you for your attention!