Interaction is Meaning: A New Model for Communication in Open Systems

Michael Rovatsos, Matthias Nickles, Gerhard Weiβ
Department of Informatics
Technical University of Munich
Motivation

The diagram illustrates the interaction between an Initiator and a Participant in a motivational context. The sequence of events includes:

- **Initiator**
  - cfp
  - refuse
  - not-understood
  - propose
  - reject-proposal
  - accept-proposal
  - failure
  - inform-ref
  - inform-done

- **Participant**
  - inform-ref
  - inform-done
Overview

▸ Motivation
▸ Communication semantics: desiderata
▸ Empirical semantics framework
▸ Analysis
▸ Conclusion
▸ Future Work
Overview

- Motivation
- **Communication semantics: desiderata**
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Communication vs. Open Systems

- open multiagent systems
Communication vs. Open Systems

- open multiagent systems
  - dynamic populations
  - self-interested agents
  - black-box agents
Communication vs. Open Systems

- open multiagent systems
  - dynamic populations
  - self-interested agents
  - black-box agents

- how can we predict what agents will do seeing only what they say?
Communication vs. Open Systems

- open multiagent systems
  - dynamic populations
  - self-interested agents
  - black-box agents
- how can we predict what agents will do seeing only what they say?
- how can we explain link between illocution and perlocution?
Communication vs. Open Systems

- open multiagent systems
  - dynamic populations
  - self-interested agents
  - black-box agents

- how can we predict what agents will do seeing only what they say?

- how can we explain link between illocution and perlocution?

- view “semantics” as an emergent, evolving phenomenon
Goals

- function of semantics: predicting other agents’ actions
Goals

► function of semantics: predicting other agents’ actions

► provide causal model of social processes
Goals

► function of semantics: predicting other agents’ actions
► provide causal model of social processes
► differences to other causal models:
Goals

► function of semantics: predicting other agents’ actions
► provide causal model of social processes
► differences to other causal models:
  ■ autonomy of other agents
  ■ homogeneity (to some degree), e.g. rationality
  ■ communication $\neq$ physical action
Goals

- function of semantics: predicting other agents’ actions
- provide causal model of social processes
- differences to other causal models:
  - autonomy of other agents
  - homogeneity (to some degree), e.g. rationality
  - communication ≠ physical action
- semantics must be expectation-based
Expectations & Communication

- experience with communication creates expectations
Expectations & Communication

- experience with communication creates expectations
- strategic use of information about expectations
Expectations & Communication

- experience with communication creates expectations
- strategic use of information about expectations
- generalisation of communicative expectations
Expectations & Communication

- experience with communication creates expectations
- strategic use of information about expectations
- generalisation of communicative expectations
- two (potentially conflicting) goals:
Expectations & Communication

- experience with communication creates expectations
- strategic use of information about expectations
- generalisation of communicative expectations
- two (potentially conflicting) goals:
  - reduce uncertainty
  - break undesirable expectations
Semantics should be...  

- **consequentialist**: meaning of utterance is defined by its consequences.
Semantics should be... 

- consequentialist: meaning of utterance is defined by its consequences
  - reactions of self and others to message
    (“first-order”)
  - impact on expectation structures
    (“second-order”)

- empirical: expectations grounded in past experience

- constructivist: meaning is in the eye of the observer
Semantics should be... 

- **consequentialist**: meaning of utterance is defined by its consequences 
  - reactions of self and others to message ("first-order")
  - impact on expectation structures ("second-order")

- **empirical**: expectations grounded in past experience
Semantics should be...

- **consequentialist**: meaning of utterance is defined by its consequences
  - reactions of self and others to message ("first-order")
  - impact on expectation structures ("second-order")
- **empirical**: expectations grounded in past experience
- **constructivist**: meaning is in the eye of the observer
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Probabilistic semantics

An example:
Probabilistic semantics

An example:

messages
- request(A,B,X) → propose(B,A,Y)
- reject(B,A,X)
- accept(B,A,X)
- propose(B,A,Y)
- confirm(A,B,X)
- reject(B,A,X)

probabilities
- accept(B,A,X) = 0.3
- propose(B,A,Y) = 0.5
- confirm(A,B,X) = 1.0
- reject(B,A,X) = 0.5
- do(B,X) = 0.9
- accept-proposal(A,B,Y) = 0.5
- reject-proposal(A,B,Y) = 0.5
- do(B,X) = 0.1
- do(A,Y) = 0.77
- do(B,X) = 0.23
- do(B,X) = 0.9

utilities
- [+]10
- [-5]

physical actions
Probabilistic semantics

- assume agent maintains such a tree \( \mathcal{F} \), and encounters are sequences \( w = w_1w_2 \cdots w_n \)
Probabilistic semantics

- assume agent maintains such a tree $\mathcal{F}$, and encounters are sequences $w = w_1w_2 \cdots w_n$

- easy to compute future distribution $I_{\mathcal{F}}(w)$ for any current $w$
Probabilistic semantics

- assume agent maintains such a tree $\mathcal{F}$, and encounters are sequences $w = w_1w_2 \cdots w_n$
- easy to compute future distribution $I_{\mathcal{F}}(w)$ for any current $w$
- calculate expected utility after encounter prefix $w$:

$$\bar{u}(w) = \sum_{w'} I_{\mathcal{F}}(w)(w') \cdot u(w')$$

- assuming that $u(w') =$sum of the utilities of physical actions along $w'$
Example

Let $w = \langle request(A, B, X), propose(B, A, Y) \rangle$:
Example

Let $w = \langle \text{request}(A, B, X), \text{propose}(B, A, Y) \rangle$:

$I_F(w) = \left\{ \langle \text{accept-proposal}(A, B, Y), \text{do}(B, X), \text{do}(A, Y) \rangle, 0.3456 \rangle, \langle \text{accept-proposal}(A, B, Y), \text{do}(B, X) \rangle, 0.1035 \rangle, \langle \text{accept-proposal}(A, B, Y) \rangle, 0.05 \rangle, \langle \text{reject-proposal}(A, B, Y) \rangle, 0.5 \rangle \right\}$

$\bar{u}(w) = -10 \cdot 0.3456 + (-5) \cdot 0.103 + (0 \cdot 0.05 + 0 \cdot 0.5) = -3.971$
Entropy Measures

- define measures to determine degree of *uncertainty* and own *autonomy*
Entropy Measures

- define measures to determine degree of uncertainty and own autonomy

\[
EE_F(w) = \sum_{w'} -P(w') \log_2 P(w')
\]

\[
UD_F(w) = \sqrt{\sum_{w'} (u(w') - \bar{u}(w'))^2}
\]
Entropy Measures

- define measures to determine degree of uncertainty and own autonomy

\[
EE_{\mathcal{F}}(w) = \sum_{w'} -P(w') \log_2 P(w')
\]

\[
UD_{\mathcal{F}}(w) = \sqrt{\sum_{w'} (u(w') - \bar{u}(w'))^2}
\]

- total entropy as combined measure:

\[
E_{\mathcal{F}}(w) = EE_{\mathcal{F}}(w) \cdot UD_{\mathcal{F}}(w)
\]
**InFFrA architecture**

Framing Architecture

- **perceived frame**
  - perception update
  - current model

- **activated frame**
  - descriptive model
  - normative model

- **difference model**
  - generate
  - roles
  - trajectories
  - contexts
  - beliefs

- **frame matching module**
- **frame enactment module**
- **frame adjustment module**
- **frame repository**

- **frame assessment module**
  - determine frame adequacy
  - determine frame validity
  - determine frame desirability

- **frame adjustment module**
  - frame updates
  - alternative frames
  - switch
  - modify
  - create

- **frame enactment module**
  - derive commitments
  - trial instantiate

- **behaviour generation module**

- **perception update**
  - current model

- **private goals/values**
  - sub-social level

---

AAMAS’2003, Melbourne, July 14-18, 2003 – p.16/43
Minimal InFFrA Agents

- a simple variant of InFFrA
Minimal InFFrA Agents

- a simple variant of InFFrA
- agents that record (and count) two-party encounters
Minimal InFFrA Agents

- a simple variant of InFFrA
- agents that record (and count) two-party encounters
- frames = simple message sequences + counters + conditions
Minimal InFFrA Agents

- a simple variant of InFFrA
- agents that record (and count) two-party encounters
- frames = simple message sequences + counters + conditions
- roles/relationships, contexts and beliefs packed into conditions
Minimal InFFrA Agents

- a simple variant of InFFrA
- agents that record (and count) two-party encounters
- frames = simple message sequences + counters + conditions
- roles/relationships, contexts and beliefs packed into conditions
- main goal: maximise expected utility
Minimal InFFrA Agents

- a simple variant of InFFrA
- agents that record (and count) two-party encounters
- frames = simple message sequences + counters + conditions
- roles/relationships, contexts and beliefs packed into conditions
- main goal: maximise expected utility
- entropy considerations useful?
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Example

Considering undesirable action in a simple request protocol:

```
request(A,B,X) do(B,X) [-10]
request(A,B,X) do(B,X) [-10]
request(A,B,X) do(B,X) [-10]
request(A,B,X) do(B,X) [-10]
```

```
perform do(B,X)
```

```
total entropy
```

```
do nothing
```

AAMAS’2003, Melbourne, July 14-18, 2003 – p.20/43
Example

Slightly more sophisticated protocol:

\begin{itemize}
\item \text{accept}(B,A,X)
\item \text{reject}(B,A,X)
\item \text{confirm}(A,B,X)
\item \text{request}(A,B,X)
\item \text{do}(B,X)
\end{itemize}

utility deviation

4.76

0.3 \quad \text{accept}(B,A,X) \quad 1.0 \quad \text{confirm}(A,B,X)

0.7 \quad \text{reject}(B,A,X)

6.40

0.9 \quad \text{do}(B,X)

0.1

[-10]
Example

Entropies: before executing undesirable action

- accept(B,A,X) 0.7
- reject(B,A,X) 0.3
- confirm(A,B,X) 1.0
- do(B,X) 0.1
- request(A,B,X) 0.9

4.86
Example

Entropies: after executing undesirable action

```
<table>
<thead>
<tr>
<th>Action</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>request(A,B,X)</td>
<td>4.86</td>
</tr>
<tr>
<td>accept(B,A,X)</td>
<td>0.3</td>
</tr>
<tr>
<td>confirm(A,B,X)</td>
<td>1.0</td>
</tr>
<tr>
<td>reject(B,A,X)</td>
<td>0.7</td>
</tr>
<tr>
<td>do(B,X)</td>
<td>1.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>request(A,B,X)</td>
<td>4.89</td>
</tr>
<tr>
<td>accept(B,A,X)</td>
<td>0.306</td>
</tr>
<tr>
<td>confirm(A,B,X)</td>
<td>0.903</td>
</tr>
<tr>
<td>reject(B,A,X)</td>
<td>0.693</td>
</tr>
<tr>
<td>do(B,X)</td>
<td>0.907</td>
</tr>
</tbody>
</table>
```
Example

External paths: the effect of “reject”

```
accept(B,A,X)  3.00
                └── confirm(A,B,X)  3.00
                    └── do(B,X)  0.9
                            0.1
                       [−10]

reject(B,A,X)  0.7
request(A,B,X)  0.3
request(A,B,X)  0.297
reject(B,A,X)   0.703
```

```
accept(B,A,X)  3.00
                └── confirm(A,B,X)  3.00
                    └── do(B,X)  0.9
                            0.1
                       [−10]

reject(B,A,X)  0.7
request(A,B,X)  0.3
request(A,B,X)  0.297
reject(B,A,X)   0.703
```

AAMAS’2003, Melbourne, July 14-18, 2003 – p.24/43
Example

Critical paths: the effect of “cheating”

```
accept(B,A,X) reject(B,A,X)
```

```
confirm(A,B,X)
```

```
request(A,B,X)
```

```
do(B,X)
```

<table>
<thead>
<tr>
<th>4.86</th>
<th>3.00</th>
<th>1.0</th>
<th>3.00</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>request(A,B,X)</td>
<td>accept(B,A,X)</td>
<td>confirm(A,B,X)</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>0.7</td>
<td>0.9</td>
<td>0.1</td>
</tr>
<tr>
<td>reject(B,A,X)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5.05</th>
<th>3.45</th>
<th>1.0</th>
<th>3.45</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>request(A,B,X)</td>
<td>accept(B,A,X)</td>
<td>confirm(A,B,X)</td>
<td></td>
</tr>
<tr>
<td>0.306</td>
<td>0.693</td>
<td>0.87</td>
<td>0.13</td>
</tr>
<tr>
<td>reject(B,A,X)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[−10] do(B,X)

AAMAS’2003, Melbourne, July 14-18, 2003 – p.25/43
Example

Back to complex protocol:

```
Example
Back to complex protocol:

```

*accept*(*B,A,X*)  
*confirm*(*A,B,X*)  
*do*(*B,X*)  
*reject*(*B,A,X*)  
*request*(*A,B,X*)  
*propose*(*B,A,Y*)  
*accept-proposal*(*A,B,Y*)  
*reject-proposal*(*A,B,Y*)

\[ \begin{align*}
0.5 & \quad 0.9 \\
0.1 & \quad 0.77 \\
0.23 & \quad \downarrow
\end{align*} \]

\[ \begin{align*}
0.9 & \quad 0.1 \\
0.5 & \quad \downarrow
\end{align*} \]

\[ \begin{align*}
0.5 & \quad 0.5 \\
0.1 & \quad \downarrow
\end{align*} \]

\[ \begin{align*}
0.3 & \quad 0.2 \\
0.5 & \quad \downarrow
\end{align*} \]

\[ \begin{align*}
0.2 & \quad 1.0 \\
0.9 & \quad \downarrow
\end{align*} \]

\[ \begin{align*}
0.1 & \quad \downarrow
\end{align*} \]
Example

Successful completion:

- `accept(B,A,X)`
- `confirm(A,B,X)`
- `do(B,X)`
- `reject(B,A,X)`
- `request(A,B,X)`
- `propose(B,A,Y)`
- `accept−proposal(A,B,Y)`
- `reject−proposal(A,B,Y)`
- `do(A,Y)`

Weights:
- `0.5`
- `0.9`
- `0.1`
- `0.23`
Example

A cheats:

- accept(B,A,X)
- confirm(A,B,X)
- do(B,X)
- reject(B,A,X)
- request(A,B,X)
- propose(B,A,Y)
- accept-proposal(A,B,Y)
- reject-proposal(A,B,Y)
- do(A,Y)

Weights:
- accept(B,A,X): 1.0
- confirm(A,B,X): 0.9
- do(B,X): 0.9
- reject(B,A,X): 0.1
- request(A,B,X): 0.2
- propose(B,A,Y): 0.5
- accept-proposal(A,B,Y): 0.5
- reject-proposal(A,B,Y): 0.5
- do(A,Y): 0.77

Penalties:
- [−10]
- [−5]
Example

$B$ cheats:

- request$(A, B, X)$
- propose$(B, A, Y)$
- reject$-(B, A, X)$
- accept-propose$(A, B, Y)$
- reject-propose$(A, B, Y)$
- accept$(B, A, X)$
- confirm$(A, B, X)$
- do$(B, X)$

Weights:
- $0.2$ for request
- $0.3$ for request
- $0.5$ for reject
- $0.5$ for accept
- $1.0$ for confirm
- $0.9$ for do
- $0.1$ for do
- $0.77$ for do
- $0.23$ for do

Penalties:
- $[-10]$ for confirm
- $[-5]$ for do
- $[-5]$ for do
Example

Rejection:

Diagram showing the flow of actions including:
- Request (A, B, X) with probability 0.2
- Propose (B, A, Y) with probability 0.5
- Accept (B, A, X) with probability 1.0
- Accept-proposal (A, B, Y) with probability 0.5
- Reject-proposal (A, B, Y) with probability 0.5
- Do (B, X) with probabilities 0.9, 0.1
- Do (A, Y) with probabilities 0.77, 0.23
- Rejected (B, A, X) with probability 0.1
- Confirmed (A, B, X) with probability 0.9

The diagram also includes values for the costs [-10] and [-5].
Trajectory Shapes

- analyse effects of each of the trajectories on $\text{propose}(A, B, X) \rightarrow \ldots \rightarrow \text{do}(A, Y)$
Trajectory Shapes

- analyse effects of each of the trajectories on \( \text{propose}(A, B, X) \rightarrow \ldots \rightarrow \text{do}(A, Y) \)

- Observations:
Trajectory Shapes

- analyse effects of each of the trajectories on \( \text{propose}(A, B, X) \to \ldots \to \text{do}(A, Y) \)

- Observations:
  - total entropy of request much higher than before (14.41)
  - accept/reject decrease entropy to 14.38/14.35
  - effects of “\( A \) cheats” much worse than “\( B \) cheats”
**Trajectory Shapes**

- analyse effects of each of the trajectories on $\text{propose}(A, B, X) \rightarrow \ldots \rightarrow \text{do}(A, Y)$

- Observations:
  - total entropy of request much higher than before (14.41)
  - accept/reject decrease entropy to 14.38/14.35
  - effects of “$A$ cheats” much worse than “$B$ cheats”

- “perfect” entropy curves consist of autonomy and commitment part
Trajectory shapes
Trajectory shapes
Conflict Potential

If $\mathcal{F}'$ is the product of $w'$ in $\mathcal{F}$, define:

$$\Delta \mathcal{E}_{\mathcal{F}}(w, w') = \mathcal{E}_{\mathcal{F}'}(w) - \mathcal{E}_{\mathcal{F}}(w)$$
Conflict Potential

- If $F'$ is the product of $w'$ in $F$, define:

$$
\Delta E_F(w, w') = E_{F'}(w) - E_F(w)
$$

- If $w'$ was expected, and $w''$ occurred, define:

$$
CP_F(w'', w', w) = \int_{w[1]}^{w[|w|]} \Delta E_F(w, w'') - \Delta E_F(w, w')\,dx
$$
Conflict Potential

- If $\mathcal{F}'$ is the product of $w'$ in $\mathcal{F}$, define:

$$\Delta \mathcal{E}_{\mathcal{F}}(w, w') = \mathcal{E}_{\mathcal{F}'}(w) - \mathcal{E}_{\mathcal{F}}(w)$$

- If $w'$ was expected, and $w''$ occurred, define:

$$\text{CP}_{\mathcal{F}}(w'', w', w) = \int_{w[1]} w[|w|] \Delta \mathcal{E}_{\mathcal{F}}(w, w'') - \Delta \mathcal{E}_{\mathcal{F}}(w, w') \, dx$$

- Example:

$$\Delta \mathcal{E}(\text{“success”}, \text{“A cheats”}) - \Delta \mathcal{E}(\text{“success”}, \text{“success”})$$
Conflict Potential

![Graph showing entropy change for different messages: request(A,B,X), propose(B,A,Y), accept-proposal(A,B,Y), do(B,X), do(A,Y). Legend: success, A cheats, B cheats, rejection, conflict curve.](image-url)
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Conclusion

Introduced general framework for empirical semantics
Conclusion

Introduced general framework for empirical semantics

- few assumptions about agents and application domain
Conclusion

Introduced general framework for empirical semantics

- few assumptions about agents and application domain
- allows for analysis of emergent and evolving meaning
Conclusion

Introduced general framework for empirical semantics

- few assumptions about agents and application domain
- allows for analysis of emergent and evolving meaning
- suggested methods for analysis
Conclusion

Introduced general framework for empirical semantics

- few assumptions about agents and application domain
- allows for analysis of emergent and evolving meaning
- suggested methods for analysis
- domain-independent definition of conflict (potential)
Conclusion

Introduced general framework for empirical semantics

- few assumptions about agents and application domain
- allows for analysis of emergent and evolving meaning
- suggested methods for analysis
- domain-independent definition of conflict (potential)
- ready to be used by agents (and designers)
Conclusion

- derived desirable properties of protocols
Conclusion

- derived desirable properties of protocols
  - autonomy-respecting and contingency-reducing
  - provide external paths
  - utility deviation high $\Rightarrow$ expectation entropy low
  - alternatives for different utility configurations
Conclusion

derived desirable properties of protocols

- autonomy-respecting and contingency-reducing
- provide external paths
- utility deviation high $\Rightarrow$ expectation entropy low
- alternatives for different utility configurations

performatives as markers for different “runs” of encounters (content for reference to objects)
Conclusion

- derived desirable properties of protocols
  - autonomy-respecting and contingency-reducing
  - provide external paths
  - utility deviation high $\Rightarrow$ expectation entropy low
  - alternatives for different utility configurations

- performatives as markers for different “runs” of encounters (content for reference to objects)

- reasoning about “utility” of semantics
  - link to agent interests
  - meaning
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Overview

- Motivation
- Communication semantics: desiderata
- Empirical semantics framework
- Analysis
- Conclusion
- Future Work
Outlook

- relationship to ontologies
Outlook

- relationship to ontologies
- conflict resolution (reification of expectation structures)
Outlook

- relationship to ontologies
- conflict resolution (reification of expectation structures)
- decision-theoretic framework for second-order utility of semantics
Outlook

- relationship to ontologies
- conflict resolution (reification of expectation structures)
- decision-theoretic framework for second-order utility of semantics
- global impact of local expectation structures
Outlook

- relationship to ontologies
- conflict resolution (reification of expectation structures)
- decision-theoretic framework for second-order utility of semantics
- global impact of local expectation structures
- homogeneity, rationality and content communication
Thank you for your attention!