Towards mobile conversational agents:
abstract and concrete models

1.1 Abstract vs. concrete models

We shall try and avoid two “plagues”

- The “hard core theory” plague: states as tuples (= vectors in Cartesian products) + undefined functions
- The AI plague: “the theory is the program”
1.1 Abstract vs. concrete models

We shall distinguish three levels of descriptions:

- *abstract models*: functional signatures
- *concrete models*: procedural definitions
- *implementations*: Prolog programs

1.2 Core vs. communication models

Intelligent autonomous agents are

- *reactive*: they are situated in an environment and respond to changes
- *pro-active*: they have a goal directed behavior
- *social*: they interact with other agents (and possibly humans)
1.2 Core vs. communication models

• *core models*: define the reactivity and pro-activeness of individual agents
• *communication models*: needed for modeling the social ability of individual agents in multi-agent systems (*MAS*)

1.2.1 The BDI core agent model

A well studied and documented core agent model based on modeling

• *beliefs*: what the agent knows about his current situation (i.e., the world around him)
• *desires*: the options that he might take at any time (they come and go)
• *intentions*: the options he has committed to at some point (they persist, but may be dropped)
1.2.1 The BDI core agent model

Let $P$, $Bel$, $Des$, $Int$ and $Act$ be the sets of current percepts, beliefs, desires, intentions and possible actions, respectively.

Then we have:
Agent state: a 3-tuple $l = (B,D,I) \in L$, where $B \subseteq Bel$, $D \subseteq Des$ and $I \subseteq Int$.

Functional signatures:

$brf$: $\wp(Bel) \times P \rightarrow \wp(Bel)$

$options$: $\wp(Bel) \times \wp(Int) \rightarrow \wp(Des)$

$filter$: $\wp(Bel) \times \wp(Des) \times \wp(Int) \rightarrow \wp(Int)$

$execute$: $\wp(Int) \rightarrow Act$
1.2.1 The BDI core agent model

Finally, we have the following signature and definition for the action decision function

\[
\text{action} : L \times P \rightarrow \text{Act}
\]

\[
\text{function action}(l,p) \\
B := \text{brf}(B,p); \\
D := \text{options}(B,I); \\
I := \text{filter}(B,D,I); \\
\text{return}(\text{execute}(I))
\]

Remarks and critics:

- the \text{brf}, \text{options} and \text{filter} calls should probably be allowed in any order, or even better in parallel rather than in sequence
- most functions are left \text{undefined}, and it is not clear that the distinction between desires and intentions will hold in all cases
- in actual implementations, this process is better described as elaborating a \text{hierarchical plan} structure
1.2.2 The FIPA communication model

The Foundation for Intelligent Physical Agent was established in 1995 as a non-profit organization for producing specifications for open agent interfaces. FIPA drafts (http://drogo.cselt.it/fipa) include proposed standards for

- agent management
- agent communication
- agent/software integration
- agent/human interaction

- The FIPA Agent Communication Language was defined as the successor of KQML (knowledge query and manipulation language) whose semantics was not universally agreed
- It is based on a set of normative communication acts (or messages) which represent the building blocks of a dialogue or conversation between agents
1.2.2 The FIPA communication model

Messages are of two types:

- *primitive*
- *composite*

Composite messages (not to be confused with *protocols*, yet to come) are defined in terms of primitive ones.

They are 4 primitives messages:

- *confirm*
- *disconfirm*
- *inform*
- *request*
1.2.2 The FIPA communication model

The inform message

Intuitively, the inform message is used by agent $s$ (= sender), to tell agent $r$ (= receiver), that some proposition $\varphi$ is true.

Some strong constraints exist with respect to the beliefs of each agent. Formally, inform is defined as follows:

$$\Box s \varphi \land \neg \Box s \{ (\Box r \varphi \lor \Box r \neg \varphi) \lor (\U r \varphi \lor \U r \neg \varphi) \}$$

where $\Box i \varphi$ and $\U i \varphi$ mean agent $i$ believes $\varphi$ and is uncertain about $\varphi$, respectively.

In other words, $s$ is sincere (= he believes what he says), and does not believe that $r$ either believes or is uncertain that proposition $\varphi$ is either true or false.
1.2.2 The FIPA communication model

The inform message
Rational effect (= an intention, but not a strict post-condition):
\[ B_r \phi \]

The inform message can be contrasted with the confirm message which has the same effect but whose precondition is
\[ B_s \phi \land B_s U_r \phi \]

Remarks and critics:
- Preconditions impose reciprocal knowledge of/on communicating agents (too much ?)
- Actual outcome is “uncertain” (some say “rightly”)
- Intended outcome could be achieved by matching neutral pairs (=“never speak unless invited to”)

Casa2003 Mobile conversational agents 17
1.2.2 The FIPA communication model

• At a higher level, agent interaction protocols define generic sequences of messages representing a complete conversation or dialogue between agents.

• This enables agents to anticipate each other’s response according to some conversation patterns.

• FIPA specifications contain predefined protocols, but do not impose upon agent to follow these standards (i.e. they can adopt their own protocols).

• Such conversations have a fixed structure that can be represented by graphs from deterministic finite automata, colored Petri nets, or UML sequence diagrams.
1.2.2 The FIPA communication model

Example: the FIPA request protocol

- The agent sending a request message (within this agreed protocol) can anticipate either a non-understood, refuse or agree response
- It is not possible for a protocol to compose existing protocols on demand e.g., to call another protocol at run-time and build a dynamic embedding of protocols (=protocols are static)
1.3 Towards mobile conversational agents

An additional concept: **mobility**

- weak mobility (=involves code transfer)
- strong mobility (=involves execution state transfer)

A virtual machine e.g., as for Java applets, allows for both

**a virtual machine for agents**

---

1.3 Towards mobile conversational agents

• We need a *balanced* agent model integrating reactive, proactive and social abilities
• We shall start by elaborating a core model of reactive and proactive agent
• We will then present a “neutral“ communication model
• We finally will combine these two models into an integrated MAS