ECASL: A Model of Rational Agency for Communicating Agents

Shakil M. Khan and Yves Lespérance
Dept. of Computer Science & Engineering
York University
Toronto, ON, Canada
Motivation

• Agent theories (e.g. Cohen & Levesque ‘90, Rao & Georgeff ‘91) model
  – the different mental attitudes of the agents
  – the relation between these different mental attitudes of the agents and physical state of the world

• But most of them do not account for
  – when agents are able to achieve their goals
  – how they plan to achieve their goals and commit to plans

• Success Theorem – does not deal with ability
Motivation

- Others (e.g. CASL - Shapiro & Lesperance ‘02) introduce a procedural component
  - supports modeling of complex multi-agent systems
- However an agent’s actions need not be consistent with her intentions, or do anything to achieve them: problematic since it makes it impossible to predict behavior
- An agent’s future directed intentions (goals) should lead to present directed intentions (actions)
Motivation

• We present
  – a mechanism for relating future and present directed intentions –
  a formal model of means-end reasoning suitable for a multi-agent context
  – a simple model of cooperative ability
  – a simple model of rational plan selection

• We show how to get a Success Theorem of the form:
  – if an agent intends to achieve a goal, and is able to do so, she will eventually achieve it provided that she behaves rationally, i.e., commits to a rational plan, and executes it
Outline

- CASL
- Ability in multiagent context
- Communicative acts
- Rational Plans and commitment to rational plans
- Rational behavior
- Success Theorem
- Related work, extensions, conclusion and future work
The Cognitive Agent Specification Language (CASL)

- A framework for specifying and verifying complex communicating multi-agent systems
- Combines:
  - a declarative action theory (Reiter ’01) defined in the Situation Calculus
  - a rich programming/process language, ConGolog (De Giacomo, Lesperance & Levesque ‘00), which has a transition semantics
  - a model of various mental states, such as, knowledge (Scherl & Levesque ‘03) and goals (Shapiro et al. ‘05), and their dynamics
CASL

• Knowledge
  – S5 logic
  – knowledge expansion

• Communicative Acts
  – $\text{inform}(inf,agt,\varphi) : inf$ informs $agt$ that $\varphi$ holds
  – $\text{inform\_Whether}(inf,agt,\psi) : inf$ informs $agt$ whether $\psi$ holds
  – $\text{inform\_Ref}(inf,agt,\theta) : inf$ informs $agt$ the value of $\theta$

• No revision of beliefs – all actions are considered to be public to avoid revision
CASL

• Goals/Intentions
  – KD logic
  – goal expansion (as a result of a request)
  – limited form of goal contraction (as a result of a cancelRequest)

• Communicative Acts
  – request(req,agt,φ) : req requests agt that φ
  – cancelRequest(req,agt,φ) : req cancels the request that φ
Single-Agent Ability

- An agent is able to achieve a goal if she knows of a plan whose execution brings about her goal.
- The agent is able to execute the plan:
  - she can *physically* execute all the actions in the plan.
  - she *has enough knowledge* to execute the plan.
    - e.g.: to open a safe, she needs to know the correct combination.
Epistemically Feasible Deterministic Programs (EFDPs)

• A program is an EFDP (De Giacomo et al ‘02) if
  – at every stage of execution, the executing agent always knows what step to take next, or knows that the program has terminated
• The agent never gets stuck due to lack of knowledge
• An agent can achieve a goal if she
  – knows of a physically executable EFDP, and
  – knows that this plan achieves the goal
Epistemically and Intentionally Feasible Deterministic Programs (EIFDPs)

- If the planning agent wants to get help from other agents, she should also take their knowledge and intentions into account.
- We extend EFDPs to be suitable for *limited* multi-agent domains: we call these *EIFDPs*.
- Limitations
  - the planning agent does all the deliberation, she must know the whole plan in advance.
  - all other agents are simple executors and do not plan.
  - program delegation is possible, but not sub-goal delegation.
EIFDPs

• At every stage of execution of the program –
  – if it is the planning agent’s turn to act, then same as EFDP
  – if it is some other agent’s turn to act, then the planning agent knows
    • that the other agent *knows what step to take* next
    • that the other agent *intends to execute it* next
    • what the remaining program is, so she knows how to continue
• The executor of the plan never gets stuck due to lack of knowledge, or other agents’ lack of intention/knowledge to perform the next action
The Safe Example

- Safe with a combination lock
- Dialing the correct combination will open the safe
- Dialing an incorrect combination will cause the safe to explode
- In the initial situation $S_0$
  - $Agt_1$ intends to open the safe, but does not know the correct combination
  - $Agt_1$ knows that $Agt_2$ knows the combination
The Safe Example : EIFDP

- $\delta_{safe}$:
  - $Agt_1$ requests $Agt_2$ to inform her of the combination;
  - $Agt_2$ informs $Agt_1$ of the combination;
  - $Agt_1$ dials the combination.

- Theorem
  - $\text{EIFDP}(Agt_1, \delta_{safe}, S_0)$, provided that $Agt_1$ knows that $Agt_2$ does not intend in $S_0$ not to inform her the combination of the safe
The Safe Example: EIFDP

$S_0$ → $A_1$ requests $A_2$ to inform Comb → $A_2$ informs $A_1$ of Comb → $A_1$ dials Comb

- $A_2$ does not intend to inform Comb
- $A_2$ intends to inform Comb
- $A_1$ knows Comb
Cooperative Ability

• An agent *can achieve* a goal $\varphi$ in situation $s$ iff she knows of a plan $\delta$, such that:
  – $\delta$ is *EIFDP* in $s$
  – $\delta$ is physically executable starting in $s$
  – any execution of $\delta$ starting in $s$ achieves $\varphi$

• Theorem
  – $\text{Can}(\text{Agt}_1, \neg \text{Locked}, S_0)$
Communicative Acts

- *inform, informWhether, informRef*: primitive
- Requests defined in terms of inform
  - \(\text{request}(\text{req}, \text{agt}, \varphi)\): \(\text{req}\) informs \(\text{agt}\) that \(\text{req}\) intends that \(\varphi\)
  - \(\text{requestAct}(\text{req}, \text{agt}, \delta)\): \(\text{req}\) informs \(\text{agt}\) that \(\text{req}\) intends that \(\text{agt}\) executes \(\delta\) starting from the next situation
- The requested goal is adopted via cooperation principles
- Canceling Requests also defined
  - \(\text{cancelRequest}(\text{req}, \text{agt}, \varphi)\): \(\text{req}\) informs \(\text{agt}\) that she no longer intends that \(\varphi\)
  - \(\text{cancelRequestAct}(\text{req}, \text{agt}, \delta)\): \(\text{req}\) informs \(\text{agt}\) that \(\text{req}\) no longer intends that \(\text{agt}\) executes \(\delta\)
Rational Plans

- \((\text{agt}, \delta_1, \delta_2, s)\) : weak domination
  - a plan \(\delta_1\) is \textit{as good as} another plan \(\delta_2\) for an agent \(\text{agt}\) in situation \(s\), if it achieves \(\text{agt}'s\) goals in all epistemic alternatives where \(\delta_2\) does.

\[
\begin{align*}
\delta_1 &> \delta_3 \\
\delta_2 &> \delta_3 \\
\delta_1 &\not> \delta_2 \\
\delta_2 &\not> \delta_1
\end{align*}
\]
Rational Plans

• A plan $\delta$ is *rational* for an agent $agt$ in situation $s$ iff
  – $\delta$ is dominant, i.e. $\delta$ is *as-good-as* any other plan that is *as-good-as* $\delta$ in $s$,
  – $\delta$ is an *EIFDP* for $agt$ in $s$, and

• Theorem
  – $\text{Rational}(Agt1,\delta_{safe},S0)$
The *commit* Action

- A bridge between future directed intentions and present directed ones
- The execution of $\text{commit}(\text{agt}, \delta)$ updates $\text{agt}$’s intentions such that she intends to execute $\delta$ starting in the next situation
- An agent can commit to a plan $\delta$ iff she does not intend not to execute $\delta$ next
Generic Meta-Controller for Rational Agent Behavior

- $\text{BehaveRationallyUntil}(agt, \psi)$ is defined as
  pick a plan $\delta$ that is \textit{rational} for $agt$ in the current situation;
  \textit{commit} $agt$ to $\delta$;
  While $\psi$ is not achieved and $agt$ intends to execute some act next
    If $agt$ intends to perform some action next
      perform that action;
    Else
      ($agt$ intends that $agt'$ performs some action next)?
      The action happens;
  endWhile.
From Commitment and Ability to Eventuality (Success Theorem)

• Theorem:
  – if
    • \textit{agt} has the \textit{intention} in situation \textit{s} to achieve \(\psi\)
    • \textit{agt} \textit{can achieve} all her intentions in situation \textit{s}
  – then
    • if \textit{agt} behaves rationally until \(\psi\), she will successfully achieve it (no matter what rational plan she picks)

• Assumption: no unintended (exogenous) actions

• Corollary:
  – \texttt{AllDo(BehaveRationallyUntil(\textit{Agt1,\neg Locked}),S_0)}
Related Work

- (Cohen & Levesque ‘90), (Rao & Georgeff ‘91)
  - all intentions eventually get dropped (AKA no infinite deferral)
  - success not related to ability
- (Sadek ‘94)
  - backward chaining planning mechanism
  - uses per-locutionary / rational effects of actions, rather than actual effects
- The KARO Framework (van Linder, van der Hoek, Meyer ‘96)
  - commit action
  - does not model rationality or provide a success theorem
Additional Results in (Khan 05)

- How to handle unintended (exogenous) actions and intention revision due to these actions
- Developed a notion of Conditional Commitment/Intention and additional communicative acts to handle conditional requests
- Modeled some simple interaction protocols (including protocols to handle conditional requests)
Conclusion & Future Work

• Main contributions
  – formalized a simple notion of cooperative ability
  – defined rational plans
  – established a link between future directed intentions and present directed ones

• Future work
  – allow sub-goal delegation
  – model interaction protocols with multiple planning agents
  – develop implementation and tools