Multi-Agent Systems

BDI Logic (Cohen and Levesque)

Albert-Ludwigs-Universität Freiburg

Bernhard Nebel, Rolf Bergdoll, and Thorsten Engesser

Winter Term 2019/20

Recap

- Epistemic/doxastic logic: What an agent knows/beliefs.
- Deontic logic: What an agent ought to bring about.
- Missing: What an agent desires and intends.

BDI (Belief, Desire, Intention) Agent

function BDI-AGENT(*percept*)
  global *beliefs*, *desires*, *intentions*
  beliefs ← UPDATE-BELIEF(*beliefs*, *percept*)
  desires ← OPTIONS(*beliefs*, *intentions*)
  intentions ← FILTER(*beliefs*, *intentions*, *desires*)
  action ← MEANS-END-REASONING(*intentions*)
  beliefs ← UPDATE-BELIEF(*action*)
  return *action*
end function

- BDI agents start out with some beliefs and intentions.
- Intentions are goals the agent has actually chosen to bring about (can be adopted and dropped).
- Beliefs and intentions constrain what the agent desires.
- Together, B, D, and I determine the agent's future intentions.

Signatures of main processes

- The alternatives for action (options) for an agent is a set of desires dependent on the agent's beliefs and its intentions:

$$options : 2^{Bel} \times 2^{Int} \rightarrow 2^{Des}$$

- To select between competing options, an agent uses a filter function. This choice depends on the agent's beliefs, current options (desires), and intentions:

$$filter : 2^{Bel} \times 2^{Des} \times 2^{Int} \rightarrow 2^{Int}$$

$\Rightarrow$Prior intentions serve as input! They provide a filter of admissibility for options, and thereby "provide a [...] purpose for deliberation, rather than merely a general injunction to do the best." (Bratman, 1987, p. 33)

Intentions: Main properties

- Intentions drive means-ends reasoning: If I adopt an intention, I will attempt to achieve it.
- Intentions persist: Once adopted they will not be dropped until achieved, deemed unachievable, or reconsidered.
- Intentions constrain future deliberation: Filter of admissibility. Options inconsistent with current intentions will not be entertained.
- Intentions influence beliefs upon which future practical reasoning is based: Rationality requires that I believe that I can achieve my intentions.

Comparison: Intention vs. Desire

- Desires, similar to intentions, are states of affairs considered for achievement (or actions considered for execution), i.e., basic preferences of an agent.
- Unlike desires, intentions involve a commitment to bringing them about.
- Unlike desires, intentions must be consistent.

(Bratman, 1990, after Wooldridge, p. 67)

My desire to play basketball this afternoon is merely a potential influence of my conduct this afternoon. It must vie with my other relevant desires [. . .] before it is settled what I will do. In contrast, once I intend to play basketball this afternoon, the matter is settled: I normally need not continue to weigh the pros and cons. When the afternoon arrives, I will normally just proceed to execute my intentions.

Role in explanations

- "I want to have some icecream, and I believe there is icecream in the freeze, and I choose to have some icecream, therefore, I go to the freeze to get some icecream."
- Each of these three clauses constitutes an adequate explanation.
- Beliefs, desires, and intentions are reason-giving forces.

Cohen & Levesque, 1990 [1]

- Ingredients:
  - Action and time
  - Belief and preference
  - Definition of intention

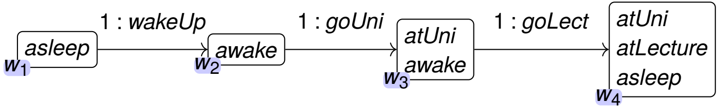
[1] The following notations are according to Meyer, Broersen, Herzig (2015). They slightly deviate from the original notations in Cohen, Levesque (1990).

Semantics for BDI: Kripkean Model

A BDI Kripke model is a tuple $M = (W, R, B, P, V)$, where:

- W is a set of possible worlds.
- $R : I \times A \to W \times W$
  - Accessibility relations $R_{i:\alpha} \subseteq W \times W$ for each action $i : \alpha$.
  - (W, R) is a linear transition system.
- $B : I \to W \times W$
  - Accessibility relations $B_i \subseteq W \times W$ for each agent i.
  - Every B_i is serial, transitive, Euclidean (**KD45**) modelling belief.
- $P : I \to W \times W$
  - Accessibility relations $P_i \subseteq B_i \subseteq W \times W$ for each agent i modelling preferences.
  - Every P_i is serial (**KD**).
- $V : \mathscr{P} \to 2^W$
  - Maps atomic propositions to their extension $V(p) \subseteq W$.

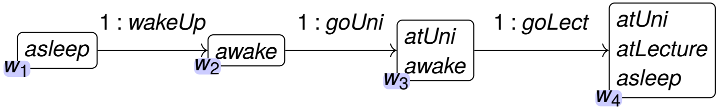
Actions: Example I



Actions: Operators

- $M, w \models Happ_{i:\alpha}\varphi$ iff there is a $w' \in W$ s.th. $(w, w') \in R_{i:\alpha}$ and $M, w' \models \varphi$ ($\Rightarrow$diamond operator).
- $M, w \models IfHapp_{i:\alpha}\varphi$ iff $M, w \models \neg Happ_{i:\alpha}\neg\varphi$ ($\Rightarrow$box operator).
- $M, w \models \exists\alpha Happ_{i:\alpha}\varphi$ iff for agent i, there exists an action type α and w' s.th. $(w, w') \in R_{i:\alpha}$ and $M, w' \models \varphi$.

Actions: Example II

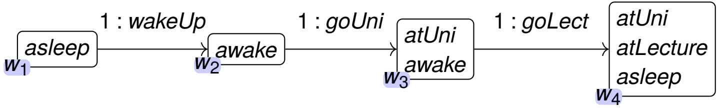


- $M, w_1 \models Happ_{1:wakeUp} awake$
- $M, w_2 \models \exists \alpha Happ_{1:\alpha} \exists \beta Happ_{1:\beta} atLecture$

Time

- $M, w \models X\varphi$ iff $M, w' \models \varphi$ for some w' s.th. $(w, w') \in R_{i:\alpha}$ for some $i : \alpha$.
- $M, w \models F\varphi$ iff $M, w \models \varphi$ or $M, w \models XF\varphi$.
- $M, w \models G\varphi$ iff $M, w \models \neg F \neg \varphi$.
- $M, w \models \psi U \varphi$ iff $M, w \models \varphi$ or ($M, w \models \psi$ and $M, w' \models \psi U \varphi$) for some w' s.th. $(w, w') \in R_{i:\alpha}$ for some $i : \alpha$.

Time: Example



- $M, w_1 \models X(awake \mathbf{U} atLecture)$
- $M, w_1 \models atSleep \wedge XFatSleep$
- $M, w_1 \models G(atSleep \leftrightarrow \neg awake)$
- $M, w_1 \models F \exists \alpha Happ_{1:\alpha} atLecture$

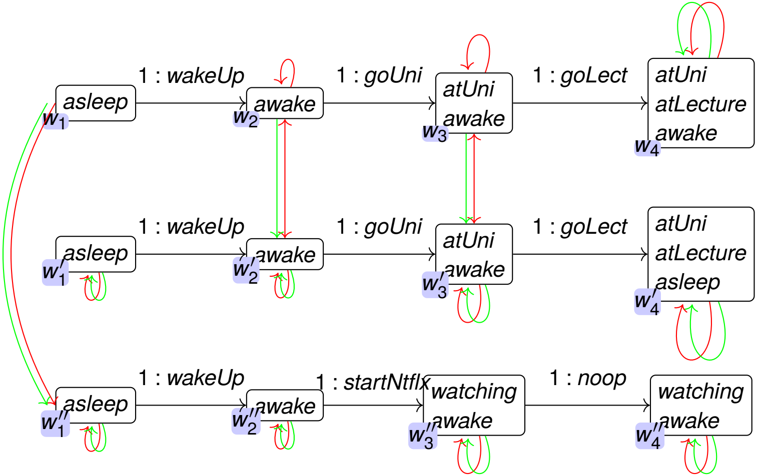
Belief and Preference

- $M,w \models Bel_i\varphi$ iff for all w' s.th. $(w,w') \in B_i$: $M,w' \models \varphi$.
  - $Know_i\varphi \stackrel{\text{def}}{=} \varphi \wedge Bel_i\varphi$.
- $M,w \models Pref_i\varphi$ iff for all w' s.th. $(w,w') \in P_i$: $M,w' \models \varphi$.
  - In the original *Pref* is called *Goal*. Some authors call it *Choice*. It is meant to be a “chosen desire” (consistent!).

Properties

- For Bel_i all properties for **KD45** operators.
- For $Pref_i$ all properties for **KD** operators.
- $\models Bel_i\varphi \rightarrow Pref_i\varphi$(Realism)
- $\models (Pref_i\varphi \wedge Bel_i(\varphi \rightarrow \psi)) \rightarrow Pref_i\psi$.

Belief and Preference: Example



Preferences alone are too weak

- Because of realism, all believed propositions are preferred propositions. But it only makes sense for an agent to adopt some goal φ if φ is believed to be false.

Achievement Goal

- Agent i has the achievement goal that φ iff i prefers that φ is eventually true and believes that φ is currently false:

$$AGoal_i \varphi \stackrel{\text{def}}{=} Pref_i F \varphi \wedge Bel_i \neg \varphi$$

Example

In the Netflix-vs.-Lecture dilemma:

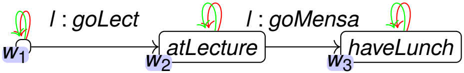
- $M, w_1 \not\models AGoal_1(asleep)$
- $M, w_1 \models AGoal_1(watching)$

Achievement Goal: Properties

- $\models AGoal_i \neg\varphi \rightarrow \neg AGoal_i \varphi$.
  - Check that $AGoal_i \neg\varphi \wedge AGoal_i \varphi$ is unsatisfiable, because the achievement goal that $\neg\varphi$ implies to believe φ, and the achievement goal that φ implies to believe $\neg\varphi$. This contradicts axiom D ($Bel_i \varphi \rightarrow \neg Bel_i \neg\varphi$). □
- $\not\models AGoal_i(\varphi \wedge \psi) \rightarrow AGoal_i \varphi \wedge AGoal_i \psi$ (for exercise).
- $\not\models AGoal_i \varphi \wedge AGoal_i \psi \rightarrow AGoal_i(\varphi \wedge \psi)$.
- $\not\models AGoal_i(\varphi \vee \psi) \rightarrow AGoal_i \varphi \vee AGoal_i \psi$.
- $\not\models AGoal_i \varphi \vee AGoal_i \psi \rightarrow AGoal_i(\varphi \vee \psi)$.

$\not\models AGoal_i\varphi \land AGoal_i\psi \rightarrow AGoal_i(\varphi \land \psi)$

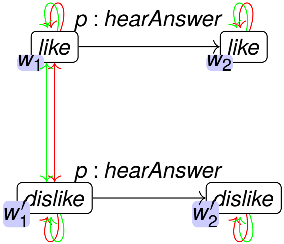
"Lisa has the goal to listen to the lecture and she has the goal to have lunch" vs. "Lisa has the goal to listen to the lecture and to have lunch"



- $M, w_1 \models AGoal_l(atLecture) \land AGoal_1(haveLunch)$
- $M, w_1 \not\models AGoal_1(atLecture \land haveLunch)$

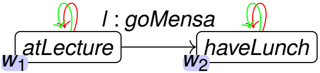
$\not\models AGoal_i(\varphi \vee \psi) \rightarrow AGoal_i\varphi \vee AGoal_i\psi$.

"Paul asks Lisa whether she likes him." (Paul does not prefer any of the two possible answers.)



- $M, w_1 \models AGoal_p(Know_p like \vee Know_p dislike)$
- $M, w_1 \not\models AGoal_p(Know_p like)$
- $M, w_1 \not\models AGoal_p(Know_p dislike)$

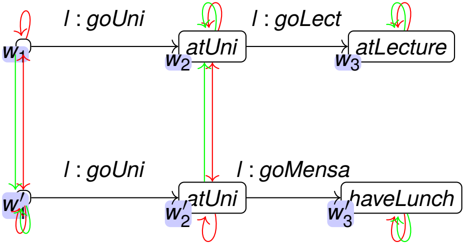
$\not\models AGoal_i \varphi \vee AGoal_i \psi \rightarrow AGoal_i(\varphi \vee \psi)$



- $M, w_1 \models AGoal_1(haveLunch)$
- $M, w_1 \not\models AGoal_1(atLecture)$
  - Reason: $M, w_1 \not\models Bel_l \neg atLecture$
- $M, w_1 \not\models AGoal_1(atLecture \vee haveLunch)$
  - Reason: $M, w_1 \not\models Bel_l(\neg(atLecture \vee haveLunch))$

Achievement Goal: Too weak for Intention

- Agents can change their preferences whenever they like: Lack of commitment!



- $M, w_1 \models AGoal_1(haveLunch)$
- $M, w_2 \models \neg AGoal_1(haveLunch)$

The Nell problem

Say a problem solver is confronted with the classic situation of a heroine, called Nell, having been tied to the tracks while a train approaches. The problem solver, called Dudley, knows that “If Nell is going to be mashed, I must remove her from the tracks.” When Dudley deduces that he must do something, he looks for, and eventually executes, a plan for doing it. This will involve finding out where Nell is, and making a navigation plan to get to her location. Assume that he knows where she is, and he is not too far away; then the fact that the plan will be carried out will be added to Dudley’s world model. Dudley must have some kind of database consistency maintainer to make sure that the plan is deleted if it is no longer necessary. Unfortunately, as soon as an apparently successful plan is added to the world model, the consistency maintainer will notice that “Nell is going to be mashed” is no longer true. But that removes any justification for the plan, so it goes too. But that means “Nell is going to be mashed” is no longer contradictory, so it comes back in. And so forth. (McDermmott 1982)

Commitment (1)

Cohen & Levesque postulate that intentions are choice and commitment.

You are having trouble with your new household robot. You say “Willie, bring me a beer.” The robot replies “OK, boss.” Twenty minutes later, you screech “Willie, why didn’t you bring that beer?” It answers “Well, I intended to get you the beer, but I decided to do something else.” Miffed, you send the wise guy back to the manufacturer, complaining about a lack of commitment.

After retrofitting, Willie is returned, marked *Model C: The Committed Assistant.*

Again, you ask Willie to bring a beer. Again, it accedes, replying “Sure thing.”

Then you ask: “What kind did you buy?” It answers: “Genessee.” You say

“Never mind.” One minute later, Willie trundles over with a Genessee in its

gripper. This time, you angrily return Willie for overcommitment.

Commitment (2)

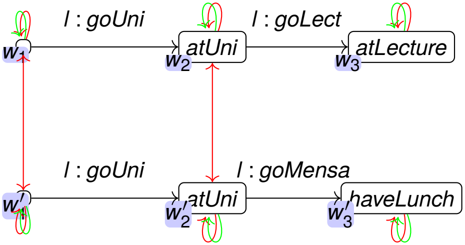
After still more tinkering, the manufacturer sends Willie back, promising no more problems with its commitments. So, being a somewhat trusting consumer, you accept the rascal back into your household, but as a test, you ask it to bring you your last beer. Willie again accedes, saying “Yes, Sir.” (Its attitude problem seems to have been fixed.) The robot gets the beer and starts towards you. As it approaches, it lifts its arm, wheels around, deliberately smashes the bottle, and trundles off. Back at the plant, when interrogated by customer service as to why it had abandoned its commitments, the robot replies that according to its specifications, it kept its commitments as long as required—commitments must be dropped when fulfilled or impossible to achieve. By smashing the last bottle, the commitment became unachievable.

Despite the impeccable logic, and the correct implementation, Willie is dismantled.

Persistent Goal

- Agent i has the persistent goal that φ iff i has the achievement goal that φ and will keep that goal until it is either fulfilled or believed to be out of reach:

$$PGoal_i \varphi \overset{\text{def}}{=} AGoal_i \varphi \wedge (AGoal_i \varphi) U (Bel_i \varphi \vee Bel_i G \neg \varphi)$$



- $M, w_1 \models PGoal_l(atLecture)$

Intention

- Agent i has the intention that φ iff i has the persistent goal that φ and believes that (s)he can achieve φ by an action.

$$Intend_i \varphi \stackrel{\text{def}}{=} PGoal_i \varphi \wedge Bel_i F \exists \alpha Happ_{i:\alpha} \varphi$$

- Intending is acting! An agent 1 cannot intend that some other agent 2 does something. However, 1 may intend to make 2 do something.
- Viz., $Intend_1 Happ_{2:act} \top$ expands to $PGoal_1 Happ_{2:act} \top \wedge Bel_1 F \exists \alpha Happ_{1:\alpha} Happ_{2:act} \top$

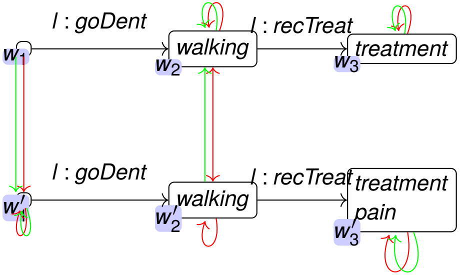
Intention: Property

- $\not\models (\mathit{Intend}_i \varphi \wedge \mathit{Bel}_i G(\varphi \rightarrow \psi)) \rightarrow \mathit{Intend}_i \psi$.

Proof

We provide a model for $\mathit{Intend}_i \varphi \wedge \mathit{Bel}_i G(\varphi \rightarrow \psi) \wedge \neg \mathit{Intend}_i \psi$: John intends to go to the dentist. He believes that going to the dentist always implies pain. At the dentist, John gets some painkiller.

Dentist Example



- $M, w_1 \models Intend_I(treament) \wedge Bel_I G(treament \rightarrow pain)$, but:
- $M, w_2 \not\models AGoal_I(pain)$, thus:
- $M, w_1 \not\models PGoal_I(pain)$, thus:
- $M, w_1 \not\models Intend_I(pain)$

Applications of Logics in MAS

- Specification
  - The intended behavior of a MAS can be specified using a logical specification language. The concrete program is derived from the specification (manually, in most cases).
- Verification
  - Once a program $\mathscr{P}$ is built, one wishes to be able to proof that it behaves according to its specification φ_p, i.e., $\mathscr{P} \models \varphi_p$.
- Agent programming
  - Agents themselves can be realized deductive reasoners: What an agent knows is represented as formulae of a formal language. The agent can reason about these formulae to derive new formulae, or to determine what to do next.

Model Checking

Definition

Model checking is an automated technique that, given a finite-state model of a system and a formal property, systematically checks whether this property holds for (a given state in) that model.

- Model of the system $\Rightarrow$How the system actually behaves.
- Formal properties $\Rightarrow$How the system should behave.
  - Safety: something bad never happen
  - Liveness: something good eventually happens
  - Fairness: if something may happen frequently, it will happen

Runtime Verification

Definition

Runtime verification is the discipline of computer science that deals with the study, development, and application of those verification techniques that allow checking whether a run of a system under scrutiny satisfies or violates a given correctness property.

$\Rightarrow$Testing using formal methods.

Requirements

- Question: Does a given BDI agent act right (viz., according to some specified properties)?
- Required
  - Representation of the agent's execution.
  - Language to specify the wanted properties.
  - Algorithm to check if some given properties hold in some represention of an execution.

Remark: Runtime Verification

- Sketch
  1. Observe the execution of the system to be verified (e.g., log state of the environment, mental state of the agents, the agents' actions).
  2. Represent the execution log using the semantics of Cohen & Levesque.
  3. Model check representation against the agents' specification, e.g.:
     - $G(\mathit{goldNear} \rightarrow \mathit{Intend}(\mathit{hasGold}))$
     - $G(\mathit{Bel}(\mathit{goldNear}) \rightarrow \mathit{Intend}(\mathit{hasGold}))$
     - $G(\mathit{battLow} \rightarrow \mathit{Intend}(\neg \mathit{battLow}))$
  4. Find time points where the specification evaluates false $\Rightarrow$Fault detection.

Literature

UNI FREIBURG

- M. Wooldridge, An Introduction to MultiAgent Systems, 2nd Edition, John Wiley & Sons, 2009.
- Bratman, M. (1987). Intention, plans, and practical reason. Harvard University Press.
- Cohen, P. R., & Levesque, H. J. (1990). Intention is choice with commitment. Artificial intelligence, 42(2-3), 213–261.
- Meyer, J.-J. Ch., Broersen, J., Herzig, A. (2015). BDI Logics. In H. van Ditmarsch, J. Y. Halpern, W. van der Hoek, B. Kooi (Eds.) Handbook of Epistemic Logic. College Publications.