Dynamic Epistemic Logic

3. Public Announcements

Albert-Ludwigs-Universität Freiburg

Bernhard Nebel and Robert Mattmüller

May 13th, 2019

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Introduction

Public Announcements

So far: Only static knowledge
(Or, where knowledge changed over time, we discussed this change only intuitively, not formally.)

Now: How to model change of knowledge over time?

Note: Knowledge may change in different ways, e. g., via public or private announcements, by sensing, or by ontic (world-changing) actions that affect knowledge along the way.

This chapter: Only public announcements.

Public Announcements

Announcement = public and truthful announcement

Example

I announce the fact: “The sun is shining”.

This announcement makes the fact common knowledge.

This holds for all public announcements of true facts about the world.

It does not generally hold for all public announcements of true statements about knowledge.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Public Announcements

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

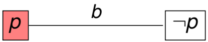
Example (Unsuccessful update)

I announce: "p is true, but Bob does not know it" ($p \wedge \neg K_b p$).

As Bob hears my announcement, he now knows p, and the announced formula $p \wedge \neg K_b p$ becomes false!

Intuition: How should epistemic models look like before and after?

Before:



After: Only those states survive where the announced formula is true.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Example

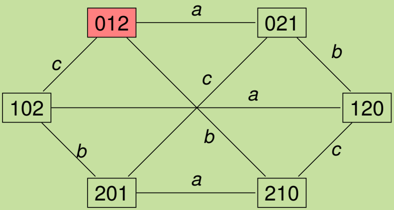
Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn a 0, Bill has drawn a 1 and Cath the 2.

Notation: We write 0_a for the fact that Anne has card 0, etc. In order to describe states, we write three digits for Anne's, Bill's, and Cath's card, e. g., 012 to describe the actual card distribution.

Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Hexa :



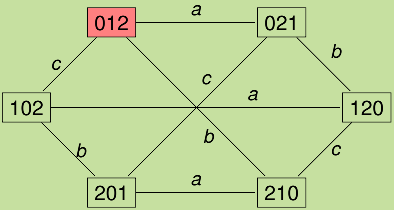
Hexa, $012 \models K_a \neg (K_b 0_a \vee K_b 1_a \vee K_b 2_a)$,
Hexa, $012 \models K_a \neg 1_a$.

Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Anne says: "I do not have card 1". ("$\neg 1_a$")

Hexa :



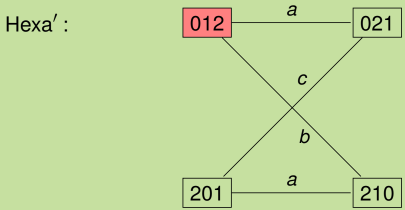
$\text{Hexa}, 012 \models K_a \neg (K_b 0_a \vee K_b 1_a \vee K_b 2_a)$,

$\text{Hexa}, 012 \models K_a \neg 1_a$.

Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Anne says: "I do not have card 1". ("$\neg 1_a$")



$\text{Hexa}', 012 \models K_c 0_a \wedge \neg K_a K_c 0_a$,

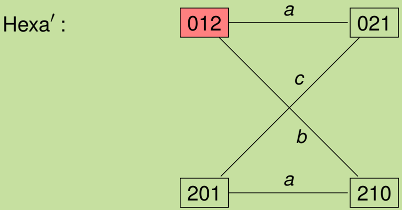
$\text{Hexa}', 012 \models K_b(\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a))$

Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Anne says: "I do not have card 1". ("$\neg 1_a$")

Bill states: "I don't know Anne's card". ("$\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a)$")



$\text{Hexa}', 012 \models K_c 0_a \wedge \neg K_a K_c 0_a$,

$\text{Hexa}', 012 \models K_b(\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a))$

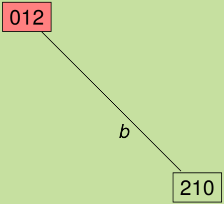
Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Anne says: "I do not have card 1". ("$\neg 1_a$")

Bill states: "I don't know Anne's card". ("$\neg(K_b 0_a \lor K_b 1_a \lor K_b 2_a)$")

Hexa$''$:



$\text{Hexa}'', 012 \models C_{abc}(K_a 0_b \lor K_a 1_b \lor K_a 2_b)$

$\text{Hexa}'', 012 \models K_a(K_a 0_b \lor K_a 1_b \lor K_a 2_b)$,

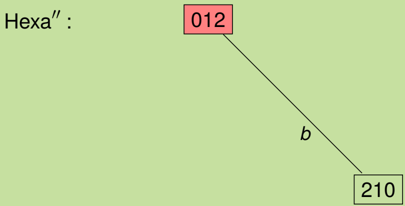
Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Anne says: "I do not have card 1". ("$\neg 1_a$")

Bill states: "I don't know Anne's card". ("$\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a)$")

Anne says: "I know Bill's card". ("$K_a 0_b \vee K_a 1_b \vee K_a 2_b$")



$\text{Hexa}'', 012 \models C_{abc}(K_a 0_b \vee K_a 1_b \vee K_a 2_b)$

$\text{Hexa}'', 012 \models K_a(K_a 0_b \vee K_a 1_b \vee K_a 2_b)$,

Example (ctd.)

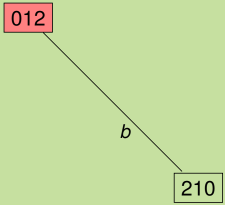
Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Anne says: "I do not have card 1". ("$\neg 1_a$")

Bill states: "I don't know Anne's card". ("$\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a)$")

Anne says: "I know Bill's card". ("$K_a 0_b \vee K_a 1_b \vee K_a 2_b$")

Hexa$''$:



Hexa$''$, $012 \models \neg C_{abc}(0_a \wedge 1_b \wedge 2_c)$,

Hexa$''$, $012 \models K_a(0_a \wedge 1_b \wedge 2_c)$

Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

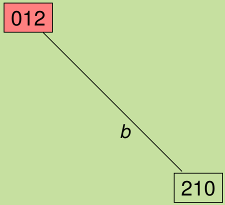
Anne says: "I do not have card 1". ("$\neg 1_a$")

Bill states: "I don't know Anne's card". ("$\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a)$")

Anne says: "I know Bill's card". ("$K_a 0_b \vee K_a 1_b \vee K_a 2_b$")

Anne says: "I have 0, Bill has 1, Cath has 2." ("$0_a \wedge 1_b \wedge 2_c$")

Hexa$''$:



$\text{Hexa}'', 012 \models \neg C_{abc}(0_a \wedge 1_b \wedge 2_c)$,

$\text{Hexa}'', 012 \models K_a(0_a \wedge 1_b \wedge 2_c)$

Example (ctd.)

Anne, Bill and Cath have drawn one card from a stack of three cards, $0, 1, 2$. Anne has drawn 0, Bill has drawn 1 and Cath 2.

Anne says: "I do not have card 1". ("$\neg 1_a$")

Bill states: "I don't know Anne's card". ("$\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a)$")

Anne says: "I know Bill's card". ("$K_a 0_b \vee K_a 1_b \vee K_a 2_b$")

Anne says: "I have 0, Bill has 1, Cath has 2." ("$0_a \wedge 1_b \wedge 2_c$")

Hexa$'''$: 012

$\text{Hexa}''', 012 \models C_{abc}(0_a \wedge 1_b \wedge 2_c)$,

Announcement Syntax

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Definition (Languages $\mathcal{L}_{K[]}$ and $\mathcal{L}_{KC[]}$)

Let P be a countable set of atomic propositions and A be a finite set of agent symbols. Then the language $\mathcal{L}_{KC[]}$ is defined by the following BNF:

$$\varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_a\varphi \mid C_B\varphi \mid [\varphi]\varphi,$$

where $p \in P$, $a \in A$, and $B \subseteq A$.

The language $\mathcal{L}_{K[]}$ is the same without the C_B clause.

$[\varphi]\psi$ reads "after a truthful announcement of φ, it holds that ψ". $\langle\varphi\rangle\psi$ is the dual of $[\varphi]\psi$: "after some truthful announcement of φ, it holds that ψ".

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Example

In (Hexa, 012), after Anne announces $\neg 1_a$, Cath knows that 0_a:

$$\text{Hexa}, 012 \models [\neg 1_a] K_c 0_a$$

After Bill's announcement that he does not know Anne's card, Anne knows Bill's card:

$$\text{Hexa}, 012 \models [\neg 1_a][\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a)] K_a 1_b$$

or: $\text{Hexa}', 012 \models [\neg(K_b 0_a \vee K_b 1_a \vee K_b 2_a)] K_a 1_b$

Announcement Semantics

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Recall that, for models $\mathcal{M}$ with domain S and formulas φ, we write $[\![\varphi]\!]_{\mathcal{M}} = \{s \in S \mid \mathcal{M}, s \models \varphi\}$.

Definition

Let $\mathcal{M} = (S, R, V)$ be an epistemic model and φ a formula. Then $\mathcal{M}|_{\varphi} = (S', R', V')$ with

- $S' = [\![\varphi]\!]_{\mathcal{M}}$,
- $R'_a = R_a \cap (S' \times S')$ for all $a \in A$, and
- $V'(p) = V(p) \cap S'$.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Definition

The truth of an $\mathcal{L}_{K[]}$ (or $\mathcal{L}_{KC[]}$) formula φ in an epistemic state $(\mathcal{M}, s)$, symbolically $\mathcal{M}, s \models \varphi$, is defined as for $\mathcal{L}_K$ (or $\mathcal{L}_{KC}$), with an additional clause for public announcements:

$$\mathcal{M}, s \models [\varphi]\psi \quad \text{iff} \quad (\mathcal{M}, s \models \varphi \text{ implies } \mathcal{M}|_\varphi, s \models \psi).$$

Note: $[\varphi]\psi$ is satisfied in s if φ is not satisfied in s.

The dual $\langle\varphi\rangle\psi = \neg[\varphi]\neg\psi$ has the truth condition $\mathcal{M}, s \models \varphi$ and $\mathcal{M}|_\varphi, s \models \psi$.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Announcements and Revelations

Question: Who actually makes the announcement?

- One of the agents?
- An omniscient external entity?

Observation: This makes a difference!

- If agent a announces φ, she must know φ, and could also announce $K_a\varphi$. This can make a difference!
- If the announcement comes from the outside, it is just $[\varphi]$. This is also called a revelation.

Principles of Public Announcement Logics

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Motivation: In this section, we will prove some valid formulas of the language $\mathcal{L}_{K[]}$ that will ultimately allow us to reduce $\mathcal{L}_{K[]}$ to $\mathcal{L}_K$ and get rid of announcement modalities.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Proposition (Functionality)

It is valid that $\langle\varphi\rangle\psi \rightarrow [\varphi]\psi$.

Proof.

Let $\mathcal{M}, s$ be arbitrary. Assume that $\mathcal{M}, s \models \langle\varphi\rangle\psi$.
This is true if and only if $\mathcal{M}, s \models \varphi$ and $\mathcal{M}|_\varphi, s \models \psi$.
This implies that $\mathcal{M}, s \models \varphi$ implies $\mathcal{M}|_\varphi, s \models \psi$, i. e.,
$\mathcal{M}, s \models [\varphi]\psi$. □

UNI FREIBURG

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Question: What about the opposite direction?
Is $[\varphi]\psi \rightarrow \langle\varphi\rangle\psi$ also valid?

Proposition

$[\varphi]\psi \rightarrow \langle\varphi\rangle\psi$ *is not valid.*

Proof.

Counterexample: model $\mathcal{M}$ with a single state s where atom p is false. Then $\mathcal{M}, s \models [p]p$, but $\mathcal{M}, s \not\models \langle p\rangle p$. □

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Proposition (Partiality)

$\langle\varphi\rangle\top$ *is not valid.*

Proof.

In any epistemic state $(\mathcal{M}, s)$ with $\mathcal{M}, s \not\models \varphi$, we have $\mathcal{M}, s \not\models \langle\varphi\rangle\top$. □

Principles of Public Announcement Logics

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Proposition (Negation)

$[\varphi]\neg\psi \leftrightarrow (\varphi \rightarrow \neg[\varphi]\psi)$ *is valid.*

Proof.

Omitted. Note that the biimplication can be equivalently written as $[\varphi]\neg\psi \leftrightarrow (\neg\varphi \vee \langle\varphi\rangle\neg\psi)$. □

Proposition

All of the following are equivalent:

1. $\varphi \to [\varphi]\psi$
2. $\varphi \to \langle\varphi\rangle\psi$
3. $[\varphi]\psi$

Proof ((1) iff (3); Rest: homework).

$$
\begin{aligned}
\mathcal{M}, s \models \varphi \to [\varphi]\psi \quad & \text{iff} \quad \mathcal{M}, s \models \varphi \text{ implies } \mathcal{M}, s \models [\varphi]\psi \\
& \text{iff} \quad \mathcal{M}, s \models \varphi \text{ implies} \\
& \qquad\qquad (\mathcal{M}, s \models \varphi \text{ implies } \mathcal{M}|_\varphi, s \models \psi) \\
& \text{iff} \quad (\mathcal{M}, s \models \varphi \text{ and } \mathcal{M}, s \models \varphi) \text{ implies} \\
& \qquad\qquad \mathcal{M}|_\varphi, s \models \psi \\
& \text{iff} \quad \mathcal{M}, s \models \varphi \text{ implies } \mathcal{M}|_\varphi, s \models \psi \\
& \text{iff} \quad \mathcal{M}, s \models [\varphi]\psi.
\end{aligned}
$$

□

Principles of Public Announcement Logics

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Proposition

All of the following are equivalent:

1. $\langle\varphi\rangle\psi$
2. $\varphi \wedge \langle\varphi\rangle\psi$
3. $\varphi \wedge [\varphi]\psi$

Proof.

Clear. □

Principles of Public Announcement Logics

Proposition (Composition)

$[\varphi][\psi]\chi$ is equivalent to $[\varphi \wedge [\varphi]\psi]\chi$.

Proof.

For arbitrary $(\mathcal{M}, s)$, we have

$$
\begin{aligned}
s \in \mathcal{M}|_{\varphi\wedge[\varphi]\psi} \quad & \text{iff } \mathcal{M}, s \models \varphi \wedge [\varphi]\psi \\
& \text{iff } \mathcal{M}, s \models \varphi \text{ and} \\
& \qquad (\mathcal{M}, s \models \varphi \text{ implies } \mathcal{M}|_\varphi, s \models \psi) \\
& \text{iff } s \in \mathcal{M}|_\varphi \text{ and } \mathcal{M}|_\varphi, s \models \psi \\
& \text{iff } s \in (\mathcal{M}|_\varphi)|_\psi.
\end{aligned}
$$

□

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Principles of Public Announcement Logics

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Let us now study how knowledge changes with announcements.

We find that $[\varphi]K_a\psi$ is not equivalent to $K_a[\varphi]\psi$.

Counterexample: $\text{Hexa}, 012 \models [1_a]K_c 0_a$, but $\text{Hexa}, 012 \not\models K_c[1_a]0_a$.

Proposition (Knowledge)

$[\varphi]K_a\psi$ is equivalent to $\varphi \to K_a[\varphi]\psi$.

Proof.

$$
\begin{array}{lll}
\mathcal{M}, s \models \varphi \to K_a[\varphi]\psi & \text{iff} & \mathcal{M}, s \models \varphi \text{ implies } \mathcal{M}, s \models K_a[\varphi]\psi \\
& \text{iff} & \mathcal{M}, s \models \varphi \text{ implies} \\
& & \quad (\mathcal{M}, t \models \varphi \text{ implies } \mathcal{M}|_\varphi, t \models \psi) \\
& & \quad\quad \text{for all } t \text{ such that } (s,t) \in R_a \\
& \text{iff} & \mathcal{M}, s \models \varphi \text{ implies} \\
& & \quad (\mathcal{M}, t \models \varphi \text{ and } (s,t) \in R_a \\
& & \quad\quad \text{implies } \mathcal{M}|_\varphi, t \models \psi) \text{ for all } t \in S \\
& \text{iff} & \mathcal{M}, s \models \varphi \text{ implies} \\
& & \quad ((s,t) \in R_a \text{ implies } \mathcal{M}|_\varphi, t \models \psi) \\
& & \quad\quad \text{for all } t \in [\![\varphi]\!] \\
& \text{iff} & \mathcal{M}, s \models \varphi \text{ implies } (\mathcal{M}|_\varphi, s \models K_a\psi) \\
& \text{iff} & \mathcal{M}, s \models [\varphi]K_a\psi.
\end{array}
$$

□

Principles of Public Announcement Logics

Proposition (Reduction)

All of the following schemas are valid:

1. $[\varphi]p \leftrightarrow (\varphi \rightarrow p)$ *for all* $p \in P$
2. $[\varphi](\psi \wedge \chi) \leftrightarrow ([\varphi]\psi \wedge [\varphi]\chi)$
3. $[\varphi](\psi \rightarrow \chi) \leftrightarrow ([\varphi]\psi \rightarrow [\varphi]\chi)$
4. $[\varphi]\neg\psi \leftrightarrow (\varphi \rightarrow \neg[\varphi]\psi)$
5. $[\varphi]K_a\psi \leftrightarrow (\varphi \rightarrow K_a[\varphi]\psi)$
6. $[\varphi][\psi]\chi \leftrightarrow [\varphi \wedge [\varphi]\psi]\chi$

Proof.

We already showed (4), (5), and (6). The others are an easy homework exercise. □

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Principles of Public Announcement Logics

Introduction

Syntax

Semantics

Revelations

Principles

Common Knowledge

Unsuccessful Updates

Axiomatisation

Example

Summary

Note: Using this proposition, one can reduce any $\mathcal{L}_{K[]}$ formula to an $\mathcal{L}_K$ formula. This means that both logics are equally expressive, and that we can use $\mathcal{L}_K$ theorem provers or model checkers for $\mathcal{L}_{K[]}$ as well.

Announcements and Common Knowledge

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Announcements and Common Knowledge

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Question: Can we also systematically eliminate announcement modalities as shown above in the presence of the commom knowledge modality?

Recall:

$$[\varphi]K_a\psi \leftrightarrow (\varphi \rightarrow K_a[\varphi]\psi) \text{ is valid.}$$

Attempted generalization to common knowledge:

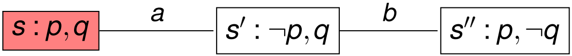
$$[\varphi]C_B\psi \leftrightarrow (\varphi \rightarrow C_B[\varphi]\psi).$$

Problem: This is invalid!

Announcements and Common Knowledge

Counterexample:

Before announcement of p:



After announcement of p:

$$\mathcal{M}, s \models [p]C_{ab}q$$
$$\mathcal{M}, s \not\models p \rightarrow C_{ab}[p]q$$

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

So, how to relate announcements and common knowledge?

Proposition (Announcements and common knowledge)

If $\chi \to [\varphi]\psi$ and $(\chi \wedge \varphi) \to E_B\chi$ are valid, then $\chi \to [\varphi]C_B\psi$ is valid.

Proof.

Let $\mathcal{M}, s$ be arbitrary and suppose that $\mathcal{M}, s \models \chi$. We want to show that $\mathcal{M}, s \models [\varphi]C_B\psi$. Suppose $\mathcal{M}, s \models \varphi$, and let t be in the domain of $\mathcal{M}|_\varphi$ such that sR_B^*t. We prove $\mathcal{M}|_\varphi, t \models \psi$ by induction over the path length from s to t. [...]

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

UNI FREIBURG

Proof (ctd.)

[...]

- Base case: If the path length is 0, then $s = t$ and $\mathcal{M}|_\varphi, s \models \psi$, which follows from $\mathcal{M}, s \models \chi$, $\mathcal{M}, s \models \varphi$, and the validity of $\chi \to [\varphi]\psi$.
- Inductive case: Assume that the path length is $n+1$ for some $n \in \mathbb{N}$, with $sR_a uR_B^* t$ for $a \in B$ and $u \in \mathcal{M}|_\varphi$. From $\mathcal{M}, s \models \chi$, $\mathcal{M}, s \models \varphi$, from the validity of $(\chi \land \varphi) \to E_B\chi$, and $sR_a u$, it follows that $\mathcal{M}, u \models \chi$. Because u is in the doamin of $\mathcal{M}|_\varphi$, we know that $\mathcal{M}, u \models \varphi$. Now, we can apply the induction hypothesis to the length-n path from u to t, which gives us $\mathcal{M}|_\varphi, t \models \psi$. □

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

UNI FREIBURG

Introduction

Syntax

Semantics

Revelations

Principles

Common Knowledge

Unsuccessful Updates

Axiomatisation

Example

Summary

Corollary

$[\varphi]\psi$ is valid iff $[\varphi]C_B\psi$ is valid.

Proof.

($\Leftarrow$) trivial

($\Rightarrow$) previous proposition with $\chi = \top$ □

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Unsuccessful Updates

Unsuccessful Updates

Definition

Given a formula $\varphi \in \mathcal{L}_{KC[]}$ and an epistemic state $(\mathcal{M}, s)$, we define:

- φ is a successful formula iff $[\varphi]\varphi$ is valid.
- φ is an unsuccessful formula iff it is not successful.
- φ is a successful update in $(\mathcal{M}, s)$ iff $\mathcal{M}, s \models \langle\varphi\rangle\varphi$.
- φ is an unsuccessful update in $(\mathcal{M}, s)$ iff $\mathcal{M}, s \models \langle\varphi\rangle\neg\varphi$.

Note:

- Updates with true successful formulas are always successful.
- Updates with unsuccessful formulas can be successful. (Homework: Example?)

Unsuccessful Updates

Question: Can we characterize successful formulas syntactically?

Answer: Not trivially, since it is possible that φ and ψ are successful, but their conjuction or disjunction are not. (Homework: find such formulas and discuss!)

Idea for an easy result: Announcing something that is already public knowledge should not affect existing knowledge. Formally: it we restrict the model in such a way that only "irrelevant" worlds are lost, public knowledge remains public knowledge.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Unsuccessful Updates

Definition (Submodel)

We call a model $\mathcal{M}'$ a submodel of $\mathcal{M}$ if $\mathcal{D}(\mathcal{M}') \subseteq \mathcal{D}(\mathcal{M})$ and R and V are restricted accordingly.

Proposition (Public knowledge updates are successful)

Let $\varphi \in \mathcal{L}_{KC[]}$. Then $[C_A\varphi]C_A\varphi$ is valid.

Proof sketch.

Let $(\mathcal{M}, s)$ be arbitrary and assume that $\mathcal{M}, s \models C_A\varphi$. Then $\mathcal{M}, t \models \varphi$ and even $\mathcal{M}, t \models C_A\varphi$ for all t with sR_A^*t. The R_A^*-reachable submodels of $\mathcal{M}|_{C_A\varphi} = \mathcal{M}|_\varphi$ are identical. Hence $\mathcal{M}|_{C_A\varphi}, s \models C_A\varphi$, i.e., $\mathcal{M}, s \models [C_A\varphi]C_A\varphi$. □

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Unsuccessful Updates

Question: What if $B \subsetneq A$? Is $[C_B\varphi]C_B\varphi$ still valid?

Answer: It is not!

Counterexample: Recall the example from earlier that showed that $[p \land \neg K_b p](p \land \neg K_b p)$ is not valid. Let $B = \{a\}$. Now consider the update formula $[C_B(p \land \neg K_b p)]C_B(p \land \neg K_b p)$. This is not valid, obviously.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Unsuccessful Updates

Back to the previous positive result (public knowledge updates are successful): Let us try to generalize the idea of preservation of truth under submodels.

Definition

The language $\mathcal{L}^0_{KC[]}$ is the following fragment of $\mathcal{L}_{KC[]}$:

$$\varphi ::= p \mid \neg p \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid K_a\varphi \mid C_B\varphi \mid [\neg\varphi]\varphi.$$

Definition

A formula φ is preserved under submodels iff, for all $(\mathcal{M}, s)$ and all submodels $\mathcal{M}'$ of $\mathcal{M}$ with $s \in \mathcal{D}(\mathcal{M}')$, if $\mathcal{M}, s \models \varphi$, then also $\mathcal{M}', s \models \varphi$.

Unsuccessful Updates

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Proposition (Preservation)

Fragment $\mathcal{L}^0_{KC[]}$ is preserved under submodels.

Proof.

By structural induction.

- Base cases: p and $\neg p$ are trivial: assume that $\mathcal{M}'$ is a submodel of $\mathcal{M}$ with $s \in \mathcal{D}(\mathcal{M}')$. Then $\mathcal{M}', s \models p$ iff $\mathcal{M}, s \models p$.
- Inductive case $\varphi \wedge \psi$:

$$\begin{aligned}
\mathcal{M}, s \models \varphi \wedge \psi \quad & \text{iff} \\
\mathcal{M}, s \models \varphi \text{ and } \mathcal{M}, s \models \psi \quad & \text{iff } (2 \times \text{I.H.}) \\
\mathcal{M}', s \models \varphi \text{ and } \mathcal{M}', s \models \psi \quad & \text{iff} \\
\mathcal{M}', s \models \varphi \wedge \psi &
\end{aligned}$$

Unsuccessful Updates

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Proof (ctd.)

- Inductive case $\varphi \vee \psi$: Similar.
- Inductive case $K_a\varphi$: Let $\mathcal{M} = (S, R, V)$ be given and $\mathcal{M}' = (S', R', V')$ a submodel of $\mathcal{M}$. Let $s \in S'$. Suppose $\mathcal{M}, s \models K_a\varphi$. Let $s' \in S'$ and $sR'_a s'$. Then $\mathcal{M}, s' \models \varphi$. By induction hypothesis, $\mathcal{M}', s' \models \varphi$. Therefore $\mathcal{M}', s \models K_a\varphi$.
- Inductive case $C_B\varphi$: Similar.

Proof (ctd.)

- Inductive case $[\neg\varphi]\psi$: Suppose $\mathcal{M},s \models [\neg\varphi]\psi$ and suppose for contradiction that $\mathcal{M}',s \not\models [\neg\varphi]\psi$. This implies $\mathcal{M}',s \models \neg\varphi$ and $\mathcal{M}'|_{\neg\varphi},s \not\models \psi$. Using the contrapositive of the induction hypothesis, we arrive at $\mathcal{M},s \models \neg\varphi$. Moreover $\mathcal{M}'|_{\neg\varphi}$ is a submodel of $\mathcal{M}|_{\neg\varphi}$, because $t \in S'$ only survives if $\mathcal{M}',t \models \neg\varphi$. Again by induction hypothesis, $\mathcal{M},t \models \neg\varphi$, so $[\![\neg\varphi]\!]_{\mathcal{M}'} \subseteq [\![\neg\varphi]\!]_{\mathcal{M}}$. But from $\mathcal{M},s \models [\neg\varphi]\psi$ and $\mathcal{M},s \models \neg\varphi$ it follows that $\mathcal{M}|_{\neg\varphi},s \models \psi$, therefore, by induction hypothesis, $\mathcal{M}'|_{\neg\varphi},s \models \psi$, which is a contradiction. □

Homework: What about formulas of the form $\hat{K}_a\varphi$, or $[\varphi]\psi$? Are they also preserved under submodels? If not, why not? Counterexamples?

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Corollary

Let $\varphi \in \mathcal{L}^{0}_{KC[]}$ and $\psi \in \mathcal{L}_{KC[]}$. Then $\varphi \rightarrow [\psi]\varphi$ is valid.

Proof.

Follows immediately from the previous proposition, since restriction to ψ-states creates a submodel. □

Unsuccessful Updates

Corollary

Let $\varphi \in \mathcal{L}^0_{KC[]}$. Then $\varphi \rightarrow [\varphi]\varphi$ is valid.

Proof.

Previous proposition with $\psi = \varphi$. □

Corollary ($\mathcal{L}^0_{KC[]}$ formulas are successful)

Let $\varphi \in \mathcal{L}^0_{KC[]}$. Then $[\varphi]\varphi$ is valid.

Proof.

Previous corollary using equivalence of $\varphi \rightarrow [\varphi]\varphi$ and $[\varphi]\varphi$. □

UNI FREIBURG

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Remark: The converse does not hold, i. e., there are also formulas not in $\mathcal{L}^0_{KC[]}$ that are successful. Example: $\neg K_a p$. Or:

Proposition

Inconsistent formulas are successful.

Example

$[p \land \neg p](p \land \neg p)$

Axiomatisation

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Without Common Knowledge
With Common Knowledge
Example
Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Without Common Knowledge
With Common Knowledge
Example
Summary

Notation:

- *PA*: The set of all valid $\varphi \in \mathcal{L}_{K[]}$.
- *PAC*: The set of all valid $\varphi \in \mathcal{L}_{KC[]}$.
- **PA**: Axiomatization of $\mathcal{L}_{K[]}$ validities (to be defined below)
- **PAC**: Axiomatization of $\mathcal{L}_{KC[]}$ validities (to be defined below)

Axiomatisation

PA

Axioms and inference rules for logic $\mathcal{L}_{K[]}$ with $a \in A$ and $p \in P$:

- all instantiations of propositional tautologies (Taut.)
- $K_a(\varphi \to \psi) \to (K_a\varphi \to K_a\psi)$ (Distribution of K_a over $\to$)
- $K_a\varphi \to \varphi$ (Truth)
- $K_a\varphi \to K_aK_a\varphi$ (Positive introspection)
- $\neg K_a\varphi \to K_a\neg K_a\varphi$ (Negative introspection)
- $[\varphi]p \leftrightarrow (\varphi \to p)$ (Atomic permanence)
- $[\varphi]\neg\psi \leftrightarrow (\varphi \to \neg[\varphi]\psi)$ (Announcement + negation)
- $[\varphi](\psi \wedge \chi) \leftrightarrow ([\varphi]\psi \wedge [\varphi]\chi)$ (Announcement + conj.)
- $[\varphi]K_a\psi \leftrightarrow (\varphi \to K_a[\varphi]\psi)$ (Announcement + knowledge)
- $[\varphi][\psi]\chi \leftrightarrow [\varphi \wedge [\varphi]\psi]\chi$ (Composition of announcements)
- From φ and $\varphi \to \psi$, infer ψ. (Modus ponens)
- From φ, infer $K_a\varphi$. (Necessitation)

Note: in example derivations, we will get sloppier over time and occasionally skip steps, especially those that involve purely propositional reasoning. Hence, the given derivations may not be derivations in the formal sense, strictly speaking, but it should always be clear how to fill in the missing details/steps.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Without Common Knowledge
With Common Knowledge
Example
Summary

UNI FREIBURG

Example

We want to show that $\vdash [p]K_a p$:

1. $p \rightarrow p$ (prop. taut.)
2. $[p]p \leftrightarrow (p \rightarrow p)$ (atomic permanence)
3. $[p]p$ (1, 2, another prop. tautology, MP)
4. $K_a[p]p$ (3, necessitation)
5. $p \rightarrow K_a[p]p$ (4, prop. taut.)
6. $[p]K_a p \leftrightarrow (p \rightarrow K_a[p]p)$ (announcements + knowledge)
7. $[p]K_a p$ (5, 6, prop. taut.)

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Without Common Knowledge
With Common Knowledge
Example
Summary

UNI FREIBURG

Theorem

The axiomatisation **PA** *of PA is sound and complete.* □

Note:

- We already showed that the axioms involving announcements are sound.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Without Common Knowledge
With Common Knowledge
Example
Summary

UNI FREIBURG

Axioms and inference rules for logic $\mathcal{L}_{KC[]}$ ($B \subseteq A$):

- all axioms and inference rules of $\mathcal{L}_{K[]}$
- $C_B(\varphi \to \psi) \to (C_B\varphi \to C_B\psi)$ (Distribution of C_B over $\to$)
- $C_B\varphi \to (\varphi \land E_B C_B\varphi)$ (Mix)
- $C_B(\varphi \to E_B\varphi) \to (\varphi \to C_B\varphi)$
  (Induction of common knowledge)
- From φ, infer $C_B\varphi$.
  (Neccessitation of common knowledge)
- From φ, infer $[\psi]\varphi$. (Neccessitation of announcements)
- From $\chi \to [\varphi]\psi$ and $\chi \land \varphi \to E_B\chi$, infer $\chi \to [\varphi]C_B\psi$.
  (Mix of announcements and common knowledge)

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Without Common Knowledge
With Common Knowledge
Example
Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Without Common Knowledge
With Common Knowledge
Example
Summary

Theorem

The axiomatisation **PAC** *of PAC is sound and complete.* □

Note:

- We already showed soundness for (most of) the additional rules and axioms.

UNI FREIBURG

Example

We show that $\vdash [\neg p]C_A\neg p$:

1. $\neg p \rightarrow \neg(\neg p \rightarrow p)$ (prop. taut.)
2. $[\neg p]p \leftrightarrow (\neg p \rightarrow p)$ (atomic permanence)
3. $\neg p \rightarrow \neg[\neg p]p$ (1, 2, prop. taut.)
4. $[\neg p]\neg p \leftrightarrow (\neg p \rightarrow \neg[\neg p]p)$ (announcements + negation)
5. $[\neg p]\neg p$ (3, 4, prop. taut.)
6. $\top \rightarrow [\neg p]\neg p$ (5, prop. taut.)
7. $\top$ (prop. taut.)
8. $K_a\top$ (7, necessitation)
9. $\top \wedge \neg p \rightarrow K_a\top$ (8, prop. taut.)
10. $\top \wedge \neg p \rightarrow E_A\top$ (9, for all $a \in A$, prop. taut.)
11. $\top \rightarrow [\neg p]C_A\neg p$ (10, 6, ann. + common knowledge)
12. $[\neg p]C_A\neg p$ (11, prop. taut.)

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Without Common Knowledge
With Common Knowledge
Example
Summary

Example: Muddy Children

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Muddy Children

UNI FREIBURG

Example (Muddy children)

- There are n children. Some of them have a muddy forehead.
- They only see whether the other children are muddy, not themselves.
- They are perfect reasoners/logicians.
- Their father says (repeatedly): “At least one of you is muddy. Those of you who know whether they are muddy please raise your hand.”

Announcements: Raising hands or not.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Muddy Children

We look at example with three children (a, b, and c), where a and b are muddy, while c not, i. e., $m_a \land m_b \land \neg m_c$.

Some abbreviations:

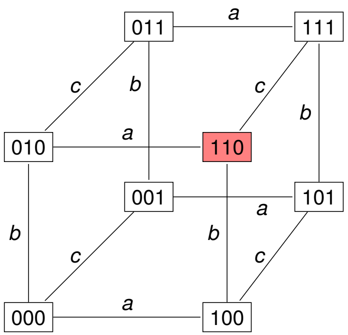
$$\begin{aligned}
\text{muddy} &= m_a \lor m_b \lor m_c. \\
\text{knowmuddy} &= (K_a m_a \lor K_a \neg m_a) \lor (K_b m_b \lor K_b \neg m_b) \lor (K_c m_c \lor K_c \neg m_c). \\
\text{abknowmuddy} &= (K_a m_a \lor K_a \neg m_a) \land (K_b m_b \lor K_b \neg m_b).
\end{aligned}$$

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Muddy Children

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccessful Updates
Axiomatisation
Example
Summary

Model Cube:

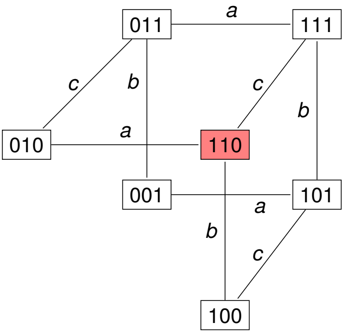


Cube, 110 $\models E_{abc}$muddy

Cube, 110 $\not\models C_{abc}$muddy (110 R_a 010 R_b 000 and Cube, 000 $\not\models$ muddy).

Muddy Children

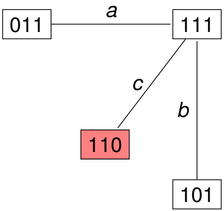
Model Cube′ = Cube$|_{\text{muddy}}$ (after announcement of muddy):



Cube′, 110 $\models C_{abc}$muddy

Cube′, 110 $\not\models$ knowmuddy

Model Cube$''$ = Cube$'|_{\neg\text{knowmuddy}}$ (after no-one raises their hand):



Cube$'$, 110 $\models \langle\neg$knowmuddy$\rangle$knowmuddy (unsuccessful update)

Cube$''$, 110 $\models$ abknowmuddy

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Muddy Children

Model Cube$'''$ = Cube$''|_{\text{abknowmuddy}}$ (after *a* and *b* raise their hands):

110

Cube$'''$, 110 $\models$ knowmuddy
Cube$'''$, 110 $\models C_{abc}(m_a \wedge m_b \wedge \neg m_c)$

Summary

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary

Summary

- Public announcements change knowledge state.
- Semantics: via submodels
- Without common knowledge: $\mathcal{L}_{K[]}$ can be reduced to $\mathcal{L}_K$.
- With common knowledge: not.
- Announcements can be successful or unsuccessful. Preserved formulas are successful
- Sound and complete axiomatizations exist.

Introduction
Syntax
Semantics
Revelations
Principles
Common Knowledge
Unsuccess-ful Updates
Axiomatisati-on
Example
Summary