COMP304 & COMP521: Knowledge Representation

Wiebe van der Hoek
The University of Liverpool
United Kingdom

Knowledge Representation

Part 1: Modal and Description Logics

Wiebe van der Hoek

Lecture 1: Introduction

Module Delivery

Lecturer: Wiebe van der Hoek
Office 223 (Ashton Building)
`wiebe@csc.liv.ac.uk`
Tutor: Petar Iliev (Ashton Building)
`{pvi}@liverpool.ac.uk`

Module notes can be found at
`http://www.csc.liv.ac.uk/~wiebe/Teaching/COMP502/`

Module Delivery

Lectures are:

	Monday	13:00 - 14:00	Room ASHT-LR
AND	Tuesday	10:00 - 11:00	Room BROD-106
AND	Wednesday	11:00 - 12:00	Room ASHT-LR

Tutorials are:

	Tuesday	15:00 - 16:00	Room GHOLT-H223
OR	Thursday	12:00 - 13:00	Room GHOLT-H223

Module Aims

- To introduce Knowledge Representation as a research area.
- To give a complete and critical understanding of the notion of representation languages and logics.
- To study modal logics and their use;
- To study description logic and its use;
- To study epistemic logic and its use
- To study methods for reasoning under uncertainty

foilheadLearning Outcomes

- be able to explain and discuss the need for formal approaches to knowledge representation in artificial intelligence, and in particular the value of logic as such an approach;
- be able to demonstrate knowledge of the basics of propositional logic
- be able to determine the truth/satisfiability of modal formula;

- be able to perform modal logic model checking on simple examples
- be able to perform inference tasks in description logic
- be able to model problems concenring agents' knowledge using epistemic logic;
- be able to indicate how updates and other epistemic actions determine changes on epistemic models;
- have sufficient knowledge to build ”interpreted systems” from a specification, and to verify the ”knowledge” properties of such systems;
- be familiar with the axioms of a logic for knowledge of multiple agents;
- be able to demonstrate knowledge of the basics of probability and decision theory, and their use in addressing problems in knowledge representation;
- be able to model simple problems involving uncertainty, using probability and decision theory;
- be able to perform simple Hilbert-style deductions in modal and epistemic logic;

- be able to use tableau based methods to do inference in description logic.

Example: Russian Cards

Three players, say A, B and C hold 7 cards, say the deal d is such that A holds 0, 1 2, B holds 3, 4, 5 and C has 6. Each player knows its own cards, and it is common knowledge how many cards everybody has and how many cards there are.

Example: Russian Cards

Three players, say A, B and C hold 7 cards, say the deal d is such that A holds 0, 1 2, B holds 3, 4, 5 and C has 6. Each player knows its own cards, and it is common knowledge how many cards everybody has and how many cards there are.

Now, design a protocol P (means here: an exchange of publicly announced messages) after which it is common knowledge that

1 A and B both know d

2 C knows only of card 6 who owns it

Example: Muddy Children

n children have a party, $k \leq n$ of them get muddy. Father calls the children in a circle around him: every child can see the other children, no child sees itself. Let φ be: "at least of you is muddy. If you know that you are, please step forward"

1	Father says: φ!	nothing happens
2	Father says: φ!	nothing happens
...	....	nothing happens
k	Father says: φ!	the k muddy children step forward!

Module objectives

At the end of the module you should

- understand the need for formal approaches to knowledge representation
- understand the value of logic as a formal approach
- understand the basics of modal and description logics and how they are used
- be able to model epistemic problems using Kripke models
- be able to indicate how updates and other epistemic actions determine changes on these models
- be able to determine the truth of epistemic formulae in a given state
- be able to decide whether a given epistemic formula is satisfiable in a given class of models

Module structure (1)

Two parts:

Part 1: Logics for KR& R

- Knowlege representation and reasoning: introduction, logical approach
- Modal Logics: syntax, semantics (Kripke models), model checking, theorem proving
- Description Logics: syntax, semantics, satisfiability checking

Module structure (2)

Two parts:

Part 2: Applications of modal logic: Epistemic logic

- one agent: S5 models, specific properties
- multiple agents: modeling epistemic puzzles, reasoning about other's konwledge and ignorance, alternating bit protocol
- group notions of knowledge: distributed knowledge, common knowledge, muddy children example
- computational models: distributed systems

Module assessment

- There is no coursework for this module (This means you have more time for self study, not more spare time!)
- This module is assessed by exam and two continuous assessment exercises!
- The exam will take 2.5 hours and will assess the module objectives

Optional textbooks

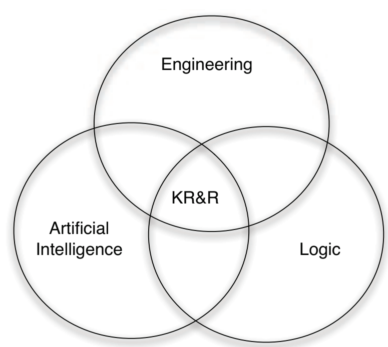
Part 1:

M. R. A. Huth and Mark D. Ryan
Logic in Computer Science: Modelling and reasoning about systems
Cambridge University Press (2000)
ISBN 0-521-65602-8.

Part 1 & 2:

J.-J. Ch. Meyer and W. van der Hoek
Epistemic Logic for Computer Science and Artificial Intelligence
Cambridge Tracts in Theoretical Computer Science 41
Cambridge University Press (1995)
ISBN 0-521-46014.

Knowledge representation and reasoning (1)



Knowledge representation and reasoning (2)

Knowledge representation and reasoning is at the intersection of AI, Logic, and Engineering.

AI The science of understanding intelligent entities and the engineering of intelligent entities

In symbolic AI, intelligent entities have an explicit model of the world, and are able to reason about it

Consequently, we have to find out what these explicit models consist of and how it is possible to reason about them

Knowledge representation and reasoning (3)

Knowledge representation and reasoning is at the intersection of AI, Logic, and Engineering.

Logic The science of reasoning

In particular, logic studies formalisms which can describe partial models of the world and calculi which allow to reason with them

Knowledge representation and reasoning (4)

Knowledge representation and reasoning is at the intersection of AI, Logic, and Engineering.

Engineering The application of science to the design, building, and use of man-made items

Engineering aspects of KR&R include

1. Engineering the logical formalism and calculus suitable for a particular application
2. Engineering the partial world model suitable for an application
3. Engineering a software system capable of performing the desired reasoning task about the partial world model

Knowledge representation and reasoning (5)

The designation ‘Knowledge representation and reasoning’ suggests that we are interested in two related problems:

- The representation of knowledge
- The reasoning about knowledge based on its representation

Historically, the relative importance of the two problems has been subject to a long-lasting debate.

We look at two contributions to this debate:

- Newell and Simon: The physical symbol system hypothesis
- Feigenbaum et al.: The knowledge principle

Physical symbol systems

A physical symbol system consist of

- a set of symbols
- a set of expressions (also called symbol structures)
- a set of procedures that operate on expressions to produce other expressions: Create, Modify, Reproduce, Destroy.

Physical symbol systems

- The symbol structures form a low-level representation of our memory
- The procedures form a low-level realisation of our reasoning processes
- designation: expressions refer to something else. The system is not just a device that computes, but that behaves with respect to a world that is external to itself;
- interpretation: expressions can even refer to computations of the device: rather than a string of symbols, a program can be interpreted as a process

The physical symbol system hypothesis

Newell and Simon (1976)

A physical symbol system has the necessary and sufficient means for general intelligent action.

By necessary we mean that any system that exhibits intelligence will prove upon analysis to be a physical symbol system.

By sufficient we mean that any physical symbol system of sufficient size can be organized further to exhibit general intelligence.

By general intelligent action we wish to indicate the same scope of intelligence as we see in human action.

The knowledge level hypothesis

Newell and Simon

> There exists a distinct computer systems level which is characterized by knowledge as the medium and the principle of rationality as the law of behaviour.

Principle of rationality: if an agent has knowledge that one of its actions will lead to one of its goals then the agent will select that action.

Knowledge: Whatever can be ascribed to an agent such that its behaviour can be computed according to the principle of rationality.

The physical symbol system hypothesis: Critique

- The emphasis of physical symbol systems is on the procedures, therefore, on reasoning and ignores the importance of knowledge
- There is no claim that there is one fundamental physical symbol system but each system/entity showing intelligent behaviour could be a different physical symbol system
- This makes it difficult to falsify the hypothesis:
  - Suppose you show me a physical symbol system of which you claim that it is intelligent
  - I show you an example of 'unintelligent' behaviour of this system
  - Then you simply amend your system to avoid this particular behaviour

The knowledge principle (1)

Feigenbaum (1994)

> The power of AI programs to perform at high levels of competence is primarily a function of the program's knowledge of its task domain, and not of the program's reasoning processes.

Lenat and Feigenbaum (1989)

> A system exhibits intelligent understanding and action at a high level of competence primarily because of the specific knowledge that it can bring to bear: the concepts, facts, representations, methods, models, metaphors, and heuristics about its domain of endeavor.

The knowledge principle (2)

Feigenbaum (1994)

> Physicians, not logicians, treat sick patients.

Underlying assumptions here:

- Logicians are the better reasoners but have little medical knowledge
- Physicians are not as good at reasoning but have the relevant medical knowledge

Obviously, the knowledge principle puts the emphasis on knowledge (and its representation), instead of reasoning

The knowledge principle (3)

> Knowledge is power, and computers that amplify that knowledge will amplify every dimension of power (Feigenbaum)

> The power of an intelligent program is to perform its task well depends primarily on the quantity and quality of knowledge it has about that task (Buchanan and Feigenbaum (1982))

Feigenbaum started to work on the first expert system in 1962. Still, Douglas Lenat's *Cyc* project, builds upon this principle

The knowledge principle: Critique (1)

- Dichotomy between knowledge and reasoning is not clear cut
  - Knowing how to reason correctly is in itself knowledge
  - In addition, 'rules of reasoning' or 'reasoning processes' may themselves be domain-specific

Example: "if Liverpool is the capital of Britain then grass is green"

is true in propositional logic, since "grass is green" is true

However, 'Liverpool is the capital of Britain' has no relevance to 'grass is green'.
So, it might seem counterintuitive that the implication is true.
In relevance logic it would be false.

The knowledge principle: Critique (2)

- The knowledge principle also underestimates the complexity of the reasoning problem
  - For first-order logic (even restricted to 'rules') complete 'reasoning processes' may not terminate
  - For sufficiently expressive, decidable logics complete 'reasoning processes' may not terminate within a reasonable amount of time (e.g. your life span)
- Nevertheless, expert systems (i.e. system based on the knowledge principle) were built around a single basic reasoning procedure, which contributed to their failure
- Consequently, we now often use logics which are specifically tailored for an application domain or single application

The knowledge principle: Critique (3)

- Finally, the knowledge principle shifts the focus from general problem solving ability (which we commonly equate to 'intelligence') to knowing the right answer and knowing the right approach
- It abandons the original aim of AI and shifts the focus to producing useful tools instead of producing intelligent entities
- Consequently, systems based on the knowledge principle which are using an explicit representation of knowledge (as rules) are in direct competition to systems which embody knowledge implicitly in a program (as algorithms)

Synthesis

- The physical symbol system hypothesis and the knowledge principle can be seen as representing two extreme positions concerning KR&R
- A more moderate position can be characterised by saying that

  There are problems and problem domains where an explicit representation of knowledge using a formal language and reasoning about this knowledge using a logical calculus as the primary means of applying this knowledge to a problem, is the best possible approach
- Of course, another extreme position would be that the whole of KR&R is obsolete

Knowledge Representation

Question: How do we *represent* knowledge in a form amenable to computer manipulation?

Desirable features:

- representational adequacy
- inferential adequacy
- inferential efficiency
- well-defined syntax and semantics
- naturalness

Summary

- Module overview
- Knowledge representation: Overview
- The physical symbol system hypothesis
- The physical symbol system hypothesis: Critique
- The knowledge hypothesis
- The knowledge hypothesis: Critique
- Synthesis

Knowledge Representation

Part 1: Modal and Description Logics

Wiebe van der Hoek

Lecture 2 Crash Course in Logic

What is Logic?

- determines whether it is justified to reason from given assumptions to a conclusion
- note: a logician cannot determine whether it rains
- he can conclude it rains from the assumptions if I hear drips on the roof, then it rains and I hear drips on the roof
- formally: $\varphi \rightarrow \psi, \varphi \vdash \psi$

There exist many many logics!

A Formal Approach

Any Logic comes in three parts:

syntax	what are the well-formed formuals (wffs)?
semantics	what do formulas mean, how do we interpret them?
deduction	how to mechanically formulate formulas, giving us for instance the valid ones?

We do the enterprise for Propositional Logic

Syntax

Let $\mathcal{P}$ be a set of atoms $p, q, p_1, p_2, \ldots$. Then $\mathcal{L}(\mathcal{P})$ or $\mathcal{L}_0$ is the smallest set closed unter the following rules:

- $\top, \bot \in \mathcal{L}_0$
- $\mathcal{P} \subseteq \mathcal{L}_0$
- if $\varphi, \psi \in \mathcal{L}_0$, then $(\varphi \wedge \psi)$, $(\varphi \rightarrow \psi), (\varphi \leftrightarrow \psi), (\varphi \vee \psi)$ and $\neg\varphi \in \mathcal{L}_0$

Note: φ, ψ are not formulas, just variables over them.

Syntax, ctd

symbol	name	capital
φ	*phi*	Φ
ψ	*psi*	Ψ
χ	*chi*	
γ	*gamma*	Γ
α	*alpha*	
β	*beta*	

Syntax, ctd

Let $\mathcal{P}$ be a set of atoms $p, q, p_1, p_2, \ldots$. Then $\mathcal{L}(\mathcal{P})$ or $\mathcal{L}_0$ is smallest set:

- $\top, \bot \in \mathcal{L}_0$
- $\mathcal{P} \subseteq \mathcal{L}_0$
- if $\varphi, \psi \in \mathcal{L}_0$, then $(\varphi \wedge \psi)$, $(\varphi \rightarrow \psi), (\varphi \leftrightarrow \psi), (\varphi \vee \psi)$ and $\neg\varphi \in \mathcal{L}_0$

Exercise 2.1

(1) Which of the following are fomulas of $\mathcal{L}_0$, which are not?
- $\neg(p)$
- $p_1 \rightarrow (p_2 \rightarrow p_1)$
- $\neg\top$
- $\neg(\neg p_1 \rightarrow p_2) \vee (p_2 \wedge p_3)$

(2) Argue that $(p_1 \wedge p_2) \neq (p_2 \wedge p_1)$!

Syntax, ctd

Sometimes a more economical set is chosen:

- $\mathcal{P} \subseteq \mathcal{L}_0$
- if $\varphi, \psi \in \mathcal{L}_{\prime}$, then $(\varphi \wedge \psi)$, and $\neg\varphi \in \mathcal{L}_0$

And then define:

$$
\begin{array}{lcl}
\top & \stackrel{\text{def}}{=} & (p \vee \neg p) \\
\bot & \stackrel{\text{def}}{=} & \neg\top \\
(\varphi \vee \psi) & \stackrel{\text{def}}{=} & \neg(\neg\varphi \wedge \neg\psi) \\
(\varphi \rightarrow \psi) & \stackrel{\text{def}}{=} & (\neg\varphi \vee \psi) \\
(\varphi \leftrightarrow \psi) & \stackrel{\text{def}}{=} & ((\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi))
\end{array}
$$

Syntax: Conventions

- In order to minimise the number of brackets, a precedence is assigned to the logical operators and it is assumed that they are left associative. Outermost brackets are omitted.
  Starting from the highest to lowest precedence we have:
  $$\neg \quad \wedge \quad \vee \quad \rightarrow \quad \leftrightarrow$$
- Thus,

$p \rightarrow q$	stands for	$(p \rightarrow q)$
$p \vee q \vee r$	stands for	$((p \vee q) \vee r)$
$p \vee q \wedge r$	stands for	$(p \vee (q \wedge r))$
$p \wedge q \leftrightarrow r$	stands for	$((p \wedge q) \leftrightarrow r)$
$\neg p \rightarrow q$	stands for	$(\neg p \rightarrow q)$
$\neg(p \rightarrow q)$	stands for	$\neg(p \rightarrow q)$

Semantics, ctd

We could say: $\wedge$ means 'and'.....

But is:

I woke up and took a shower

the same as

I took a shower and woke up

???

Semantics, ctd

We specify the semantics of propositional logic in truth tables

$\wedge$	p	q	$(p \wedge q)$
	0	0	0
	0	1	0
	1	0	0
	1	1	1

and

$\neg$	p	$\neg p$
	0	1
	1	0

Semantics, ctd

We specify the semantics of propositional logic in truth tables

$\wedge$	p	q	$(p \wedge q)$
	0	0	0
	0	1	0
	1	0	0
	1	1	0

and

$\neg$	p	$\neg p$
	0	1
	1	0

Exercise 2.2
Make truth-tables for the other connectives $\vee$ and $\leftrightarrow$, using the definition given above.

Implication

$\rightarrow$	p	q	$(p \rightarrow q)$
	0	0	?
	0	1	?
	1	0	?
	1	1	?

Implication

Consider the following statements:

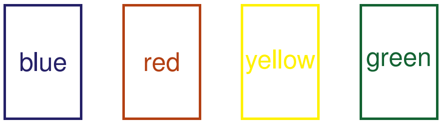
If x is greater than 7, it is also bigger than 4 (which is true)
If y is greater than 7, it is also bigger than 11

The latter is false: take $y = 9$

$\rightarrow$	p	q	$(p \rightarrow q)$	
	0	0	1	$x = 3$
	0	1	1	$x = 5$
	1	0	0	$y = 9$
	1	1	1	$x = 8$

Implication

Every card has a color at each side. Four are displayed on a table:



Implication

Every card has a color at each side. Four are displayed on a table:

blue	red	yellow	green

Claim: *if a card has a red side, it also has a blue side*

Question: how many cards minimally to turn around, and verify claim?

Implication

Every card has a color at each side. Four are displayed on a table:

blue	red	yellow	green

Exercise 2.3 Explain the outcome of the four colours card puzzle using the truth-table of $\rightarrow$.

Semantics

Notation: $\models \varphi$ for: "φ is always true"
Such a φ is also called a tautology.
Procedure: check that in the table, we only have 1's

Examples $p \vee \neg p$, $p \rightarrow p$, $(p \rightarrow q) \rightarrow ((p \rightarrow r) \rightarrow (p \rightarrow (q \wedge r)))$

Example

p	q	r	$(p \rightarrow q)$	$\rightarrow$	$((p \rightarrow r)$	$\rightarrow$	$(p$	$\rightarrow$	$(q \wedge r)))$
0	0	0							
0	0	1							
0	1	0							
0	1	1							
1	0	0							
1	0	1							
1	1	0							
1	1	1							

Example

p	q	r	$(p \to q)$	$\to$	$((p \to r)$	$\to$	$(p$	$\to$	$(q \wedge r)))$
0	0	0	1	1	1	1		1	0
0	0	1	1	1	1	1		1	0
0	1	0	1	1	1	1		1	0
0	1	1	1	1	1	1		1	1
1	0	0	0	1	0	1		0	0
1	0	1	0	1	1	0		0	0
1	1	0	1	1	0	1		0	0
1	1	1	1	1	1	1		1	1

Counterexamples

We have: $\not\models (p \vee q) \to q$

p	q	$(p \vee q)$	$\to$	q
0	0			
0	1			
1	0			
1	1			

Counterexamples

We have: $\not\models (p \vee q) \to q$

p	q	$(p \vee q)$	$\to$	q
0	0	0	1	0
0	1	1	1	1
1	0	1	0	0
1	1	1	1	1

A situation in which p is true, and q is false, is a counterexample.

Semantics

Exercise 2.4 Check which of the following formulas are a tautology. If not, give a counterexample.

(1) $p \to (q \to p)$
(2) $p \to (p \to q)$
(3) $(p \to q) \vee (q \to p)$
(4) $(p \to (q \vee r)) \to ((p \to q) \vee (p \to r))$

Semantics: Consequence

For Γ a set of formulas, and φ a formula,

$\Gamma \models \varphi$ means: if all formulas in Γ is true, φ is also true.

Example $\{p, p \to q\} \models q$

p	q	p	$(p \to q)$	q
0	0	0	1	0
0	1	0	1	1
1	0	1	0	0
1	1	1	1	1

wvdh COMP304/COMP521

Semantics: Consequence

For Γ a set of formulas, and φ a formula,

$\Gamma \models \varphi$ means: if all formulas in Γ is true, φ is also true.

Example $\{p, p \to q\} \models q$

p	q	p	$(p \to q)$	q
0	0	0	1	0
0	1	0	1	1
1	0	1	0	0
1	1	1	1	1

Semantics: Consequence

Exercise 2.4 Check whether the following are true:

(1) $\{\neg q, p \to q\} \models \neg p$
(2) $\{\neg p, p \to q\} \models \neg q$
(3) $\{p, p \to q, (\neg r \to \neg q)\} \models r$
(4) $\{p \vee q, p \to r, q \to r\} \models r$

Deduction

$A1 \quad \varphi \to (\psi \to \varphi)$
$A2 \quad (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$
$A3 \quad (\neg\varphi \to \psi) \to ((\neg\varphi \to \neg\psi) \to \varphi)$

Define $\vdash \varphi$ iff there exists a sequence of formulas $\alpha_1, \ldots \alpha_n$ such that $\alpha_n = \varphi$ and for every $\alpha_i (i \leq k)$:

(1) α_i is an instantiation of $A1, A2$ or $A3$; or
(2) there are $j, k < n$, $\alpha_j = \alpha_k \to \alpha_i$

(2) says that $\vdash \alpha_k, \vdash \alpha_k \to \alpha_i \Rightarrow \vdash \alpha_i$ is a derivation rule: it is called *Modus Ponens*. $A1 - A3$ are called *axioms*.

Deduction

$A1 \quad \varphi \to (\psi \to \varphi)$
$A2 \quad (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$
$A3 \quad (\neg\varphi \to \psi) \to ((\neg\varphi \to \neg\psi) \to \varphi)$

Define $\Gamma \vdash \varphi$ iff there exists a sequence of formulas $\alpha_1, \ldots \alpha_n$ such that $\alpha_n = \varphi$, and for every $\alpha_i (i \leq k)$:

(1) α_i is an instantiation of $A1, A2$ or $A3$; or
(2) there are $j, k < n$, $\alpha_j = \alpha_k \to \alpha_i$
(3) $\alpha_i \in \Gamma$

(2) says that $\vdash \alpha_k, \vdash \alpha_k \to \alpha_i \Rightarrow \vdash \alpha_i$ is a derivation rule: it is called *Modus Ponens*. $A1 - A3$ are called *axioms*.

Deduction: Example

$A1 \quad \varphi \to (\psi \to \varphi)$
$A2 \quad (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$
$A3 \quad (\neg\varphi \to \psi) \to ((\neg\varphi \to \neg\psi) \to \varphi)$
$MP \quad \vdash \varphi, \vdash \varphi \to \psi \Rightarrow \vdash \psi$

Example $\vdash p \to p$

Deduction: Example

$A1 \quad \varphi \to (\psi \to \varphi)$
$A2 \quad (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$
$A3 \quad (\neg\varphi \to \psi) \to ((\neg\varphi \to \neg\psi) \to \varphi)$
$MP \quad \vdash \varphi, \vdash \varphi \to \psi \Rightarrow \vdash \varphi$

Example $\vdash p \to p$

1	$(p \to ((p \to p) \to p)) \to$ $((p \to (p \to p)) \to (p \to p))$	$A2$
2	$p \to ((p \to p) \to p)$	$A1$
3	$(p \to (p \to p)) \to (p \to p)$	1,2, MP
4	$p \to (p \to p)$	$A1$
5	$p \to p$	3,4, MP

Connecting $\models$ and $\vdash$

An inference system is *sound* wrt a semantics if for all φ, $\vdash \varphi \Rightarrow \models \varphi$.

Exercise 2.6

(1) Show that the inference system defined on slide 12 is sound wrt the semantics of slide 4.
(2) A sound inference system is easily found, in a trivial way. How?

An inference system is *complete* wrt a semantics if for all φ, $\models \varphi \Rightarrow \vdash \varphi$.

Exercise 2.7

(1) A complete inference system is easily found, in a trivial way. How?

Completeness

Theorem We have, for all Γ and φ:

$$\Gamma \vdash \varphi \Leftrightarrow \Gamma \models \varphi$$

This is the *adequateness* theorem for propositional logic, stating that our deduction formalism is both sound and complete with respect to the semantics of truth tables.

Further Remarks

- The problem, for given φ whether $\vdash \varphi$, is NP-complete
- A well-known extension of propositional logic is *predicate logic*, with quantifiers $\forall, \exists$.
  - This is basis for a very popular KR formalism
  - *all students are bright* $\forall x(Sx \to Bx)$
  - *some students are lazy* $\exists x(Sx \wedge Lx)$
  - This logic is also known as *first order logic*
  - For this logic, the question wether $\vdash \varphi$ is *undecidable*
- The theory of predicate logic + arithmetic is not even *axiomatisable*!

Knowledge Representation

Part 1: Modal and Description Logics

Wiebe van der Hoek

Lecture 3: Formal logic

KR&R and formal logic

- Our KR&R doctrine:

  There are problems and problem domains where an explicit representation of knowledge using a formal language and reasoning about this knowledge using a logical calculus as the primary means of applying this knowledge to a problem, is the best possible approach
- In the following the combination of formal language, well-defined semantics, and deductive system will be called a formal logic
- Question: Why do we want to use formal logics?

Formal logic

- Formal logic is based on the utilization of symbols which
  - function as substitutes for natural language expressions
  - help us to ignore the irrelevant aspects of natural language expressions
  - help us to point out patterns relevant for consequence
- There are two modes of use of symbols:
  - Regimentative mode
  - Abstractive mode

Symbols: Regimentative mode (1)

Using symbols in the regimentative mode means to disregard irrelevant peculiarities of the grammar of natural language

Example: To express the fact that John loves Mary

we may use various natural language statements, e.g.

John loves Mary, It is John who loves Mary, Mary is loved by John

On the level of formal logics all these are replaced by

loves(John,Mary)

Or, if we employ l to represent loves, c_1 to represent John and c_2 to represent Mary, by

$$l(c_1, c_2)$$

Symbols: Regimentative mode (2)

- Regimentation means the reduction of redundancies in the lexicon and/or grammar of natural language.
- That is, regimentation is simply a kind of sifting of natural language through the sieve of relevance: what is relevant is unambiguously retained in the resulting formal representation, that which is not, vanishes
- Symbols which are utilized for the purposes of regimentation may be considered to be constant: each of them stands constantly for a definite natural language expression or at least for a definite 'pattern' common to several synonymous natural language expressions.

Symbols: Abstractive mode

- In abstractive mode we may use the symbol l to represent an arbitrary binary predicate and the symbols c_1 and c_2 to represent arbitrary terms

  The expression $$l(c_1, c_2)$$

  then represents every statement which shares the form with

  John loves Mary, Mary hates Peter, etc.
- Thus, in abstractive mode we abstract from particular relations, like loves or hates, and particular individuals, like John or Mary, and instead let symbols represent arbitrary relations and individuals
- The symbols are thus not constants, but parameters.

Symbols in standard logics (1)

In standard logics, like propositional and first-order logic, we can distinguish two kinds of symbols:

- there are symbols that are used in the regimentative mode, namely logical connectives, quantifiers, or the equality sign.
  For example, a symbol as $\wedge$ is not meant to be considered once as 'and' and once as 'or'.
  Symbols of this kind are called logical constants
- the other symbols, called extralogical constants, are ambiguous between constants and parameters.
  For example, '$l(c_1, c_2)$' may be understood to represent a concrete statement, such as 'John loves Mary', or it can be understood as an abstract schema amounting to all statements of the relevant form.

Formal languages (1)

- Formal logics are not only characterised by their use of symbols but also by the fact that the way that symbols can be put together to form more complex expressions is governed by a simple grammar which we have to strictly adhere to
- The grammar tells us how symbols can be combined to form the elements of the language of a formal logic, its formal language
  In most cases, the grammar is context-free
- Thus, formal languages contribute to the regimentative character of formal logics

Formal languages (2)

For example, we have already seen that

John loves Mary, It is John who loves Mary, Mary is loved by John

are different ways in which natural language can express the formal statement

loves(John,Mary)

Furthermore,

loves(John,Mary)	loves(Mary,John)
loves(John,John)	loves(Mary,Mary)

are the only ways the symbols loves, John, Mary can be combined to form a well-formed expression (of first-order logic)

Formal languages (3)

- In one of the following lecture we will discuss the distinction between object language and meta language.
- An object language is the formal language of some logic.
  In an object language the meaning of extralogical constants will be given by an interpretation.
- The meta language is the language we use to talk about an object language.
  In the meta language extralogical constants stand for object language expressions. Thus, they are used in abstractive mode.

Semantics (1)

- One of the most important features of a formal logic is that its expressions have a well-defined meaning (semantics)
- This allows for a common understanding of the represented information while also allowing concise descriptions
- In constrast to standard logics, natural language does not have a well-defined meaning. In particular, its handling of both logical connectives and quantifiers is often ambiguous

Semantics (2)

Example:

All dogs hate a cat

First-order logic:

1. $\forall x(dog(x) \rightarrow \exists y(hate(x,y) \wedge cat(y)))$
2. $\exists y(cat(y) \wedge \forall x(dog(x) \rightarrow hate(x,y)))$
3. $\forall x \forall y((dog(x) \wedge cat(y)) \rightarrow hate(x,y))$
4. $\forall x \exists y(dog(x) \rightarrow (hate(x,y) \wedge cat(y)))$

Semantics (3)

- In all the cases we will consider, the semantics of a formal logic is based on the notion of an interpretation
- An interpretation will be given by
  - a mathematical structure

  together with
  - an interpretation function that maps the extralogical constants of the logic to elements of the mathematical structure, and
  - an inductive definition of the meaning of logical connectives and quantifiers, which allows us to define the notion of a true formula

Semantics (4)

- The class of all interpretations can then be used to distinguish between
  - formulae which are not true in any interpretation (unsatisfiable formulae)
  - formulae which are true in some interpretations but not necessarily in all interpretations (satisfiable formulae)
  - formulae which are true in every interpretation (valid formulae)

Semantics (5)

It is important to note that identical formulae can fall into different categories depending on which class of interpretations we are using.

Example:

- Consider a first-order language with constants John and Robin and unary predicate symbol male.
- Suppose that all we know is that
  $$\mathsf{male}(\mathsf{John})$$
  is valid (true in all interpretations)
- Does this mean that
  $$\neg\mathsf{male}(\mathsf{Robin})$$
  is valid?

Semantics (6)

Example (continued):

- Does this mean that
  $$\neg\mathsf{male}(\mathsf{Robin})$$
  is valid?
- In first-order logic, the answer is negative, since in the absence of any additional information, there are two interpretations:
  - one interpretation where $\mathsf{male}(\mathsf{Robin})$ is true,
  - one interpretation where $\mathsf{male}(\mathsf{Robin})$ is false.
- In a database system, the answer is positive, since it will only consider the initial model, that is,
  - the interpretation where $\mathsf{male}(\mathsf{Robin})$ is false.

Deductive systems (1)

- For most non-trivial logics, the class of interpretations contains an infinite number of interpretations.
- Suppose we would like to check whether a formula φ is valid. We cannot simply go through all interpretations checking whether φ is true in all of them, since this process would never terminate.
- Deductive systems provide the means to derive valid formulae without the need to inspect interpretations.
- Deductive systems can be viewed and understood as games where one or more players move according to a given set of rules.

Deductive systems: Example (1)

The following is a game we play on strings consisting only of the letters **M**, **U**, and **I**.

- We always start with the string **MI**
- The four rules of the game are:
  1. If we have a string of the form x**I**, then we can replace it by x**IU**
  2. If we have a string of the form **M**x, then we can replace it by **M**xx
  3. If we have a string of the form x**III**y, then we can replace it by x**U**y
  4. If we have a string of the form x**UU**y, then we can replace it by xy

  where x and y are arbitrary (possibly empty) strings

Deductive systems: Example (2)

The following is a sequence of moves using these rules:

(a)	**MI**	start
(b)	**MII**	from (a) using rule 2
(c)	**MIIII**	from (b) using rule 2
(d)	**MIIIIU**	from (c) using rule 1
(e)	**MUIU**	from (d) using rule 3
(f)	**MUIUUIU**	from (e) using rule 2
(g)	**MUIIU**	from (f) using rule 4

Deductive systems: Example (3)

- At the moment, we don't have a definition of what it means to win our example game.
- In some deductive systems no such definition is required, because every situation is a winning situation.
- In other deductive system we will have some indicator which signifies a winning situation, e.g. if we can reach the string **MU** then we have won.

Aside:

- This game is a simple physical symbol system.

Meta Reasoning about MIU

Is **MU** derivable in the system **MIU**?

NO!

Meta Reasoning about MIU

Is **MU** derivable in the system **MIU**?

NO! We have **Theorem**

Let the *I-count* of a string be the number of **I**-symbols in it. Then: every string x that is derivable from **MI** has an *I-count* which is never a multiple of 3.

Axiom:		**MI**
Rules:	1	if x**I** is a theorem, then so is x**IU**
	2	if **M**x is a theorem, then so is **M**xx
	3	if x**III**y is a theorem, then so is x**U**y
	4	if x**UU**y is a theorem, then so is xy

Summary

- Formal logics allow for a concise description of a problem which disregards irrelevant peculiarities of natural language.
- Formal logics have a well-defined semantics which provides the basis for a common, unambiguous understanding of a problem description.
- Formal logics are accompanied by deductive systems which allow us to derive information which is implied by a problem description, but not necessarily explicitly stated in it.

Exercises

Exercise 3.1 Give translations of the following statements in a first order language. First indicate which symbols you choose for which terms.

(1) students are humans
(2) everybody who loves somebody, is blessed
(3) nobody has read all books, but everybody has read some
(4) two courses are loved by everyone
(5) everybody loves two courses

Exercises

Exercise 3.2 Is the following reasoning sound (m and i are constants)

$$\forall x H(x,m), (H(m,i) \wedge \forall x(H(m,x) \rightarrow x = i)) \vdash m = i$$

Now, take the following translations for the constants and predicates:

$H(x,y)$	x loves y
m	Madonna
i	me

The reasoning would represent:
If everybody loves Madonna, and Madonna loves only me,
then I am Madonna.
What is unsatisfactory in this anyalysis?

Knowledge Representation

Part 1: Modal and Description Logics

Wiebe van der Hoek

Lecture 4: Introduction to Modal Logic

Substitution

$[\alpha/\beta]\varphi$ means 'φ, with subformula β replaced by α'.

Examples

$$[q/p]\ ((p \wedge r) \to \neg p) = ((q \wedge r) \to \neg q)$$
$$[(p \wedge q)/\neg p]\ ((p \wedge q) \to \neg p) = ((p \wedge q) \to (p \wedge q))$$
$$[(p \wedge q)/\neg p]\ ((p \wedge q) \to \neg p) = ((q \wedge q) \to (p \wedge q))$$
$$[q/p]\ ((q \wedge r) \to \neg t) = ((q \wedge r) \to \neg t)$$

Extensionality

A property of propositional and predicate logic:

$$\models (\alpha \leftrightarrow \beta) \to (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

In words: if α and β are equivalent, then you may freely substitute one for the other in any formula without altering its meaning.

The meaning of a formula is also called *extensionality*

Thus: the meaning of a formula only depends on the extensionality of its subformulas, not on their form.

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \to (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

today it rains and I am happy
is equivalent to
today it rains and I study in Liverpool

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \to (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

today it rains and I am happy
is equivalent to
today it rains and I study in Liverpool

I give you a hundred pound or I am happy
is equivalent to
I give you a hundred pound or I study in Liverpool

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

I am happy because I follow Knowledge Representation
is equivalent ??? to
I study in Liverpool because I follow Knowledge Representation

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

I am happy because I follow Knowledge Representation true!!
is equivalent ??? to
I study in Liverpool because I follow Knowledge Representation false!!

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

during all year, I am happy
is equivalent to
during all year, I study in Liverpool

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

during all year, I am happy false!!
is equivalent to
during all year, I study in Liverpool true!!

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

the director of studies knows that I am happy
is equivalent to
the director of studies knows that I study in Liverpool

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

the director of studies knows that I am happy false!!
is equivalent to
the director of studies knows that I study in Liverpool true!!

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

I wish that I am happy
is equivalent to
I wish that I study in Liverpool

Extensionality: Examples

$$\models (\alpha \leftrightarrow \beta) \rightarrow (\varphi \leftrightarrow [\alpha/\beta]\varphi)$$

α: I am happy
β: I study in Liverpool

We are now in situation in which $(\alpha \leftrightarrow \beta)$ is true!

I wish that I am happy true!!
is equivalent to
I wish that I study in Liverpool false!!

Upshot

- propositional logic is extensional;
- for knowledge, time, desires, because ... we do not want extensionality
- Conclusion: propositional logic is not suitable if we want to deal with knowledge, time, desires, because ...
- modal logic will help us out: it is an example of an intensional logic

Modal Logic: Syntax

Let $\mathcal{P}$ be a set of atoms $p, q, p_1, p_2, \ldots$. Then $\mathcal{L}_m(\mathcal{P})$ or $\mathcal{L}$ is smallest set:

- $\top, \bot \in \mathcal{L}$
- $\mathcal{P} \subseteq \mathcal{L}$
- if $\varphi, \psi \in \mathcal{L}$, then $(\varphi \wedge \psi), (\varphi \rightarrow \psi), (\varphi \leftrightarrow \psi), (\varphi \vee \psi), \neg\varphi$ and $\Box\varphi \in \mathcal{L}$

Informal meaning of $\Box$

$\Box\varphi$ has many possible readings:

reading	meaning
alethic	φ is *necessarily* the case
epistemic	φ is *known* to be the case
doxastic	φ is *believed* to be the case
temporal	φ is *always* the case
dynamic	φ is *caused* by a program
provabilistic	φ is *provably* the case
motivational	φ is *desired* to be the case
deontic	φ *ought* to be the case

Informal meaning of $\Box$

$\Box\varphi$ has many possible readings:

reading	meaning	notation
alethic	φ is *necessarily* the case	$\Box\varphi$
epistemic	φ is *known* to be the case	$K\varphi$
doxastic	φ is *believed* to be the case	$B\varphi$
temporal	φ is *always* the case	$\Box\varphi$
dynamic	φ is *caused* by a program	$[\alpha]\varphi$
provabilistic	φ is *provably* the case	$\Box\varphi$
motivational	φ is *desired* to be the case	$D\varphi$
deontic	φ *ought* to be the case	$\bigcirc\varphi$

Modal Semantics

Let p be: it is sunny in Liverpool, and q the same for Manchester.

An agent (1) in Liverpool knows the weather there, but not in Manchester.

He considers two alternatives: p, q and $p, \neg q$.

We call such alternatives *worlds*, with names $w, v, w', \ldots$.

An outsider might be able to distinguish what the *real* world is (designated in *blue*)

Modal Semantics

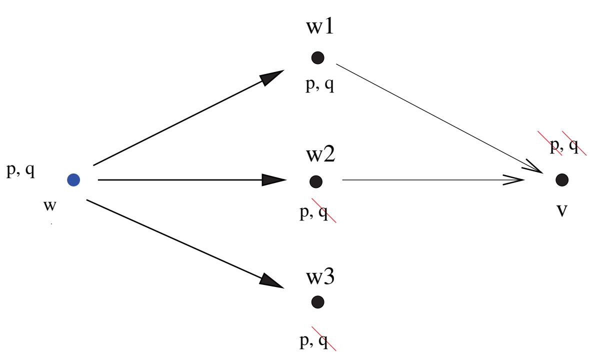
If an agent cannot distinguish between two worlds w and w', we draw an arrow between them.

Of course, he cannot distinguish any world from itself.

We now want to express that, given w, agent 1 knows p, but he does not know q

Formally, this would be $M, w \models (K_1 p \wedge \neg K_1 q \wedge \neg K_1 \neg q)$

Modal Semantics



Definition 4.0. *A* Kripke Model $M = \langle W, R, \pi \rangle$ *where*

- W *is a set of worlds*
- $R \subseteq W \times W$ *is a binary relation*
- $\pi : W \rightarrow \mathcal{P} \rightarrow \{true, false\}$

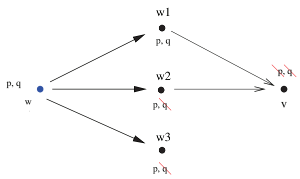
Knowledge Representation

Part 1: Modal and Description Logics

Wiebe van der Hoek

Lecture 5: Introduction to Modal Logic

Modal Semantics



$W =$	$\{w, w1, w2, w3, v\}$
$R =$	$\{(w, w1), (w, w2), (w, w3),$
	$(w1, v), (w2, v)\}$
$\pi :$	$\pi(w)(p) = \pi(w)(q) = true$
	$\pi(v)(p) = \pi(v)(q) = false$
	$\pi(w2)(p) = \pi(w3)(p) = true$
	$\pi(w2)(q) = \pi(w3)(q) = false$
	$\pi(w1) = \pi(w)$

Definition A *Kripke Model* $M = \langle W, R, \pi \rangle$ where

- W is a set of worlds
- $R \subseteq W \times W$ is a binary relation
- $\pi : W \to \mathcal{P} \to \{true, false\}$

Truth Definition

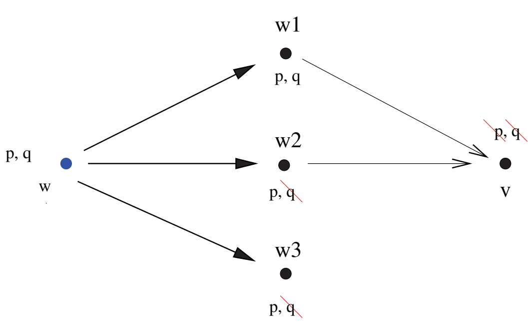
Definition 4.0 A *Kripke Model* $M = \langle W, R, \pi \rangle$ where

- W is a set of worlds
- $R \subseteq W \times W$ is a binary relation
- $\pi : W \to \mathcal{P} \to \{true, false\}$

Definition 5.1. *We define what it means that* $M, w \models \varphi$:

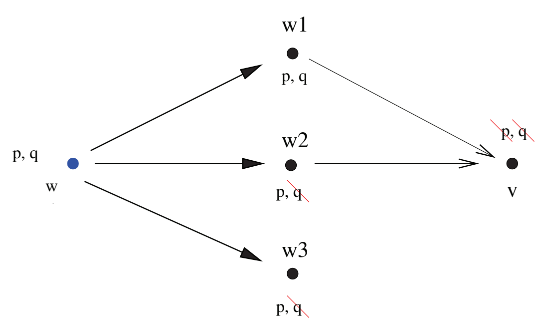
$$
\begin{array}{lll}
M, w \models p & \text{iff} & \pi(w)(p) = true \\
M, w \models \varphi \wedge \psi & \text{iff} & M, w \models \varphi \text{ and } M, w \models \psi \\
M, w \models \neg\varphi & \text{iff} & \text{not: } M, w \models \varphi \\
M, w \models \Box\varphi & \text{iff} & \text{for all } v : (Rwv \Rightarrow M, v \models \varphi)
\end{array}
$$

Modal Semantics: Example



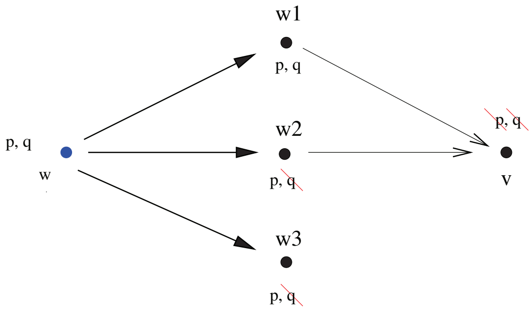
$$
\left.\begin{array}{ll}
M, w \models & (p \leftrightarrow q) \\
M, w \models & \neg(\Box p \leftrightarrow \Box q)
\end{array}\right\} \Rightarrow \text{got rid of extensionality!}
$$

Modal Semantics: Example



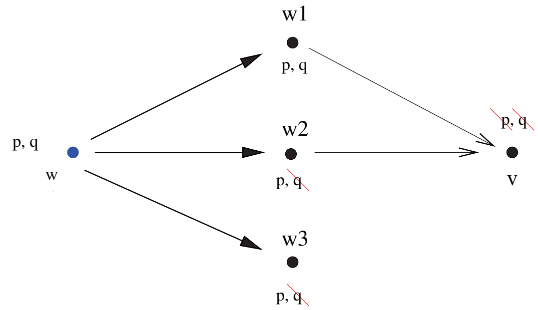
$$
\begin{array}{ll}
M, w \models & (p \leftrightarrow q) \\
M, w \models & \neg(\Box p \leftrightarrow \Box q) \\
M, w \models & \Box(p \to q)? \\
M, w \models & (\Box p \to \Box q)? \\
M, w \models & \Box(q \vee \neg q)? \\
M, w \models & \neg\Box\neg(p \wedge q)?
\end{array}
$$

Modal Semantics: Example



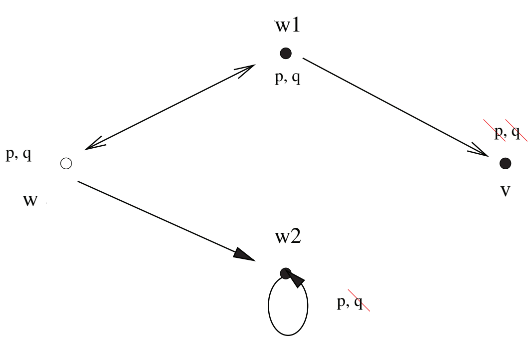
$M, w\mathbf{1} \models \quad \Box(\neg p \wedge \neg q)$
$M, w\mathbf{2} \models \quad \Box(\neg p \wedge \neg q)$
$M, w\mathbf{3} \models \quad \Box(\neg p \wedge \neg q)$**!**
$M, w \models \quad \Box\Box(\neg p \wedge \neg q)$**!**
$M, w \models \quad \Box\Box(\neg p \wedge \neg q)$**!**

Modal Semantics: Exercises



Exercise 5.2. Recall that the definition of $\Diamond\varphi \overset{\text{def}}{=} \neg\Box\neg\varphi$ Show that:
$M, w \models \Diamond\varphi$ iff there is a v: Rwv and $M, v \models \varphi$

Modal Semantics: Exercises



Exercise 5.3. Verify of the model M whether:
$M, w \models \quad (p \leftrightarrow q)$
$M, w \models \quad \Box\Diamond p$
$M, w \models \quad \Diamond\Box(p \leftrightarrow q)$
$M, w \models \quad \Box(\Box p \rightarrow \neg\Diamond q)$