Chapter 9

Modal Language, Syntax, and Semantics

In chapter 6 we saw that $\mathbb{PL}$ is not expressive enough to represent valid arguments and semantic relationships that employ quantified expressions "some" and "all". To deal with this, a more expressive language was developed, one that extended PL with two, one-place operators ("$\forall$" and "$\exists$") and that represented the relations between objects. *Modals* are expressions that qualify a proposition. For example, "John is happy" might be qualified by saying that "John is *typically* happy" or "John *ought* to be happy" or "John *was* happy" or "*It is possible that* John is happy." When translating a sentence into $\mathbb{PL}$, we ignore modal expressions and translate each of the above sentence with a unique letter. Unfortunately, this ignores the fact that the validity of many arguments and semantic relationships are sensitive to modal expressions involved. For example, consider the following argument:

> It is necessary that John is happy.
> Therefore, John is happy.

Intuitively, the above argument is valid. If John *must* be happy, then it is impossible for John not to be tall. However, consider the following representation of the above argument in PL:

> N
> H

The above argument is clearly invalid. In order to bring English arguments like the one above into the realm of symbolic logic, it is necessary to develop a formal language that captures expressions that signify the various *modes* in which a proposition may be true or false. This is the language of *modal logic*. We will symbolize it as $\mathbb{ML}$.

9.1 Expressing Necessity and Possibility

There are a variety of terms in English and other natural languages that express different modes (modalities) in which a proposition may be true or false. Some of these include words like "can," "must," "necessary," and "possible." Many of these occur in real life sentences and play critical roles in the formalization and representation of sentences.

> (1) "It always seems *impossible* until it is done" – Nelson Mandela
> (2) "It is *impossible* to rightly govern a nation without God and the Bible" – George Washington
> (3) "I *can* accept failure, everyone fails at something. But I *can't* accept not trying." – Michael Jordan.

(4) “I *can* live without money, but I *cannot* live without love.” – Judy Garland

There is no operator in PL to represent the different modes in which a proposition is true. Thus, symbolizing (1)–(4) requires the addition of some new operators. ML adds to the language of PL two one-place operators: ◊ (the diamond) and □ (the box). The diamond captures the notion of possibility, which is represented by terms like “can” and “it is possible that” while the box captures the notion of necessity, which is reprsented by terms like “must” and “it is necessary that.” As the syntax of ML allows for affixing the diamond “◊” and box “□” to the left of a proposition **P**, various English sentences can be translated into ML. For instance, consider the following two simple sentences:

(5) It is possible that John will win the Olympics.
(6) It is necessary that we lift the smoking ban.

Symbolizing “John will win the Olympics” as “O” and “We will lift the smoking ban” as “S”, (5) and (6) can be translated into the following two ML formulas:

(5*) ◊J
(6*) □S

9.2 Possible Worlds and Accessibility

Before diving deeper into how ML can be used to translate English sentences, it is useful to get a clearer understanding of two key notions in ML, along with its syntax and semantics. First, one useful tool for understanding the nature of modality (the modes by which a proposition can be true) are possible worlds. A *possible world* (represented by *w*) is a *complete* and *possible* scenario, i.e. it is a fully specific way in which a world can be.

possible world (*w*)	A *possible world* is a *complete* and *possible* scenario, i.e. it is a fully specific way in which a world can be.

At least from the outset, the notion of “possibility” is used in the widest sense. Namely, provided a scenario is free from contradiction or inconsistency, it is possible. Thus, while it is possible for many things that are not the case to be the case, e.g. a hippopatomus dancing, pigs flying, or for humans not to exist, some things are not possible, e.g. for a person “John” to both exist and not exist. In saying that a possible world is *complete*, what is meant is that the scenario or way in which things could be is fully specific. That is, in no possible world are certain details about a given object left out, e.g. whether they exist or whether they have a certain property like *tall* or *red*.

Since there is more than one complete and possible way in which things could be, there is more than one possible world. An upper-case W is used to represent a non-empty set of possible worlds, and lower-case w’s with or without numerical subscripts (or with superscripted primes, e.g. w′) are used to represent specific worlds. Thus, w1 and w2 represent two possible worlds, as do w′ and w″. In addition, w1 ∈ W represents that possible world w1 belongs to (is a member of) the set W of possible worlds.

In ML, the actual world is considered a possible world as it is a complete and possible scenario. While it is no doubt the case that the actual world differs from possible worlds as the actual world is the one you and I live in, the one that in fact obtains, and the one we tend to be most interested in, from the perspective of the logician, the actual world differs from other possible worlds only in terms of its content (the objects and relations between these objects that obtain). For example, consider the following proposition, 'Mitt Romney is a U.S. President'. While this proposition is false in our world, we could talk about other possible worlds or situations in which 'Mitt Romney is a U.S. President' is true. Thus,

w_1: 'Mitt Romney is a U.S. President' is true
w_2: 'Mitt Romney is a U.S. President' is false.

Finally, the language of modal logic involves a binary *accessibility relation* R on worlds such that certain worlds are said to be "accessible" from other words. Thus, 'Rw_1w_2' reads:

w_1 accesses w_2
w_2 is accessible from w_1
there is access from w_1 to w_2

Alternatively, 'w_2Rw_1' reads:

w_2 accesses w_1
w_1 is accessible from w_2
there is access from w_2 to w_1

By contrast, w_2 is not accessible to w_1 when w_2 is not possible relative to w_1. The intuitive idea behind a world w_2 being accessible to w_1 is that w_2 is *possible relative to* w_1. Another way to think about this is that you and I both live in a world, let's call this w_1. From our perspective in w1, some complete scenarios (worlds) are possible and some complete scenarios (worlds) are not (exactly what are and aren't is a subject for debate). Thus, we say that our world w1 accesses a number of other worlds, e.g. Rw1w2, Rw1w3, Rw1w3, if and only if those worlds are possible relative to w1.

The idea that certain worlds are possible relative to other worlds may, at first glance, seem a tad strange. One might, for example, think that a single, fixed set of scenarios are possible. However, the notion of accessibility allows for capturing the a variety of different notions of possibility. For example, consider that in this world it is both the case that spaceships do not travel at faster-than-light speeds and it is not technically (or physically) possible for a spaceships to travel at faster-than-light speeds. Thus, our world does not access any world where spaceships travel at faster-than-light speeds. But, now consider a world whose physical laws are much different than our own, one's that permit faster-than-light speed. In such a world, while it is not the case that spaceships travel at faster-than-light speed, it is physically possible for them to do so. In other words, those worlds access worlds where faster-than-light speed does occur.

9.3 Language & Syntax

The language and syntax of $\mathbb{ML}$ extends the language and syntax of $\mathbb{PL}$. As such, $\mathbb{ML}$ retains the symbols used to represent propositions, truth-functional operators, and scope.

1	Uppercase Roman (unbolded) letters ('A1,' 'A2,' 'B,' 'C,' …, 'Z') with or without subscripted integers for propositional letters
2	Truth-functional operators ($\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$)
3	Parentheses, braces, and brackets to indicate the scope of operators

$\mathbb{ML}$ extends $\mathbb{PL}$ with the following two one-place operators: ◊ (the diamond) and □ (the box). The syntax of $\mathbb{ML}$ allows for affixing the diamond "◊" to the left of a proposition **P** such that "◊P" is read as "It is possible that P" or "P is possible." Similarly, the box "□" can be affixed to the left of a proposition **P** such that "□P" is read as "It is necessary that P" or "P is necessary."

4	Two one-place modal operators (◊ and □)

As with $\mathbb{PL}$ and $\mathbb{RL}$, the syntax for $\mathbb{ML}$ can be formulated using a set of formation rules. Namely, a proposition in $\mathbb{ML}$ is a well-formed formula (wff) if and only if can be formulated by the following set of rules:

(1) Single propositional letters A, B, C, ..., Z with and without numerical subscripts are wffs.
(2) If **P** is a wff, then so is ¬(**P**).
(3) If **P** and **R** are wffs, then so are **P**∧**R**, **P**∨**R**, **P**→**R**, **P**↔**R.**
(4) If **P** is a wff, then so is ◊(**P)** and □(**P**).
(5) Nothing else is a wff except in virtue of (1)–(5).

Similar to the syntax of $\mathbb{PL}$ and $\mathbb{RL}$, two conventions are adopted for the scope of 1-place operators. *Convention #1* is that parentheses are optional when a 1-place operator operates upon a single propositional letter or another 1-place operator. Thus, ¬(**P**), ◊(**P**), ◊(□(**P**)) can be simplified to ¬**P**, ◊**P**, and ◊□**P,** respectively when "**P**" is a single propositional letter. *Convention #2* involves the use of braces [ ] and brackets { } when a set of parentheses is contained in another set of parentheses. Thus, P∨(S∨(M∨(L∧T))) is more perspicuously symbolized as P∨{S∨[M∨(L∧T)]}.

As in $\mathbb{PL}$, formation rules are used recursively to create any wff in $\mathbb{ML}$. To illustrate, consider the following:

Show that □◊P is a wff.		
1	P is a wff	Rule 1
2	If P is a wff, then ◊P is a wff.	Rule 4 + Line 1
3	If ◊P is a wff, then □◊P is a wff	Rule 4 + Line 2

The use of the formation rules for ML begin with identifying the needed propositional letters, proceed to attaching those operators having the least scope, until the operator with the most scope is attached. As a final example, consider a more complicated use of the formation rules:

Show that ¬(□◊P→□R) is a wff.		
1	P and R are wffs	Rule 1
2	If R is a wff, then □R is a wff.	Rule 4 + Line 1
3	If P is a wff, then ◊P is a wff.	Rule 4 + Line 2
4	If ◊P is a wff, then □◊P is a wff.	Rule 4 + Line 3
5	If □◊P and □R are wffs, then □◊P→□R is a wff.	Rule 3 + Line 2, 4
6	If □◊P→□R is a wff, then ¬(□◊P→□R) is a wff.	Rule 2 + line 5

Exercise Set #1

A. *Syntax: Identifying well-formed formulas*. State whether the following are well-formed formulas in ML.
1. ◊P
2. □¬P
3. ¬□(P→R)
4. □□(P→R)
5. □P→R
6. P□∧P
7. P□→P◊
8. ¬◊¬P
9. ◊(□P)
10. ¬P□¬P

B. *Syntax: Using the formation rules*. Using the formation rules provided in this chapter, show that the following formulas are wffs.
1. ◊P
2. □P
3. ¬□P
4. □(P→R)
5. □¬(P→R)
6. ◊□P
7. □◊P
8. □(◊P↔R)
9. ◊□P∧¬□◊P
10. (□P→□□P)→(□□P→□P)

9.4 Types of Relations: Preparation for 9.5*

Intuitively, a relation is some way in which two or more objects are connected or taken together. For example, if John is taller than Liz, we say that one way that John and Liz can be taken together is through the taller-than relation. The standing-between relation is another example. If

John is standing between Liz and Sam, we say that that *standing-between* is a way in which three objects (John, Liz, and Sam) can be connected.

Relations specify the number of objects that they connect or take together. The taller-than relation is a binary (2-place) relation as it connects two objects. The standing-between relation is a tertiary (3-place) relation as it connects three objects. While common relations are unary (1-place), binary (2-place), tertiary (3-place), a relation can connect any number of objects. To keep open the possibility of talking about relations that connect four, five, or more objects, any given relation is an *n*-place relation, where '*n*' is a placeholder for the number of objects it connects.

But what is a relation? A *relation* is just a set of n-tuples, an n-tuple being an ordered list of elements. The *tall* relation is the 1-place relation that is just the set of tall objects. The taller-than relation is a 2-place relation and so it is a set whose members consist of ordered pairs <x, y> where the first element x is taller than the second element y. The standing-between relation is the 3-place relation consisting of triples <x, y, z> where x is standing between y and z. And, finally, an n-place relation is a set of n-tuples.

Relation	An n-place relation is a set of n-tuples.

When an n-tuple like <a, b> is a member of a relation R, we say that a and b "stand in" R (i.e., stand in relation R). Thus, if John is taller than Liz, we can say that John and Liz "stand in" the taller-than relation. To abbreviate

There are several important binary (2-place) relations worth mentioning:

R is *serial* in A iff for every $x \in$ A there is some y in A such that Rxy
R is *reflexive* in A iff for every $x \in$ A, Rxx
R is *irreflexive* in A iff for every $x \in$ A, it is not the case that Rxx
R is *symmetric* in A iff for every x and $y \in$ A, if Rxy, then Ryx
R is *asymmetric* in A iff for every x and $y \in$ A, if Rxy, then not Ryx
R is *transitive* in A iff for every x, y, and $z \in$ A if Rxy and Ryz, then Rxz
R is an *equivalence relation* in A iff R is symmetric, transitive, and reflexive in A.
R is *total* in A iff for every x and $y \in$ A, Rxy

The *equal-to* relation is reflexive for the set of positive integers {1, 2, 3, …} since for every integer x in {1, 2, 3, …}, x is equal to x. In contrast, the *taller-than* relation is an irreflexive relation as no object is taller than itself. Further, the *being married* relation is a symmetric relation for if John is married to Liz, then Liz is also married to John.

There are three additional points worth considering about relations. First, a relation can be more than one type of relation. One example are asymmetric relations, which are irreflexive. To consider a second example, let's reflect on the *greater-than-or-equal-to* relation (we'll abbreviate this as "R") for the set of positive integers {1, 2, 3, …}. This relation is the set of ordered pairs <x, y> where x is greater than y:

{<1,1>, <2,1>, <2,2>, <3,1>, <3,2>, <3,3>, <4,1>, …}

Notice that this relation is *serial* since for every integer x, there is always some number y where x is greater than y. It is also reflexive since as integer x is greater than or equal to itself. However, it is not symmetric as 5 is greater than or equal to 4, but 4 is not greater than or equal to 5.

Second, some relations only apply to specific sets. The *sister-of* relation is asymmetric for the set of all people but it is not for a set whose members consist of a group of sisters. If we have a set consisting of Liz and John, Liz may be the sister of John but John is not the sister of Liz. R is asymmetric in this set. In contrast, in a set that consists of Liz and Sam, where both are sisters of each other, the *sister-of* relation is symmetric rather than asymmetric since Liz is a sister of Sam and Sam is a sister of Liz.

Third, it is important to keep in mind that the above relations are defined using variables x and y of a set A. Although the variables are notationally distinct—i.e., the variable x is not the variable y—x and y can have the same value. As an illustration, consider whether *x loves y* relation is a total relation for a set where everyone loves everyone except themselves. For this set, it is true that if we substitute a member for x and a distinct member for y, then for every x and y in the set, x loves y. But x and y range over the entire set and so they could choose the same member. In that case, *x loves y* would not be a total relation.

Exercise Set #2

A. *Examples of Binary Relations*: Come up with an example of the following binary relations (be sure to indicate the set that the relation applies to)
1. Serial
2. Equivalence
3. Total
4. Irreflexive
5. Reflexive

B. *Binary Relations*: Consider the following examples of binary relations. Identify what kind of binary relation these relations instantiate (at least one) and explain why, e.g. "*greater than* is a transitive relation for the positive integers since for any positive number x, y, and z, if x is greater than y, and y is greater than z, then x is greater than z."
1. Holding hands, as in x holds y's hand
2. Less than, as in x is numerically less than y.
3. Loving, as in x loves y
4. Father of, as in x is the father of y.
5. Boss of, as in x is the boss of y.

9.5 Modal Logic Semantics

The approach to ML semantics considered here is known as *possible-worlds semantics*. This approach is based on the idea that a proposition P is *necessarily true* if and only if it is true in *all* possible worlds and P is *possibly true* if and only if it is true in *at least one* possible world. Propositions will not be assigned truth values absolutely but always *relative* to possible worlds. Thus, the truth value of formulas like P, $\neg$P, P$\wedge$R, $\Box$P are always determined relative to a given possible world. The truth values of propositions without modal operators (e.g., P$\wedge$R) will be determined within a single world w_1 using truth-table definitions for operators, e.g. P$\wedge$R is true in a world if and only if P and R are true in w_1. In other words, we won't need to consider the truth values of "P" and "R" at various possible worlds w_2, w_3, w_4 in order to determine the truth value of "P∧R" at w_1. In contrast, the truth values of propositions with modal operators (e.g., □P and ◊P) at w_1 will require a consideration of the truth value of P at all of the possible worlds that are accessible to w_1. In other words, determining the truth value of □P at w_1 may require a consideration of the truth values of P at w_1, w_2, w_3, and so on.

Possible-world semantics begins with a generic definition of a *model* that is later tailored to specific modal systems. A *model* is a triple (or three-part structure) consisting of a non-empty set of possible worlds $\mathcal{W}$, a binary relation $\mathcal{R}$ on $\mathcal{W}$ (the "accessibility relation"), and an interpretation $\mathcal{I}$ function that assigns truth values (T or F) to each sentence letter in each world. The part of the model consisting of $\mathcal{W}$ and $\mathcal{R}$ is known as the model's *frame*. Intuitively, the frame of the model represents the way in which the world is by telling us what possible worlds there are and which worlds access each other. The part of the model consisting of $\mathcal{I}$ is known as the model's *interpretation*. Intuitively, this part of the model tells us the meaning of the non-logical expressions in ML. It does this by assigning truth values (T or F) to each propositional letter at each world.

An $\mathbb{ML}$ model $\mathcal{M}$ is an ordered triple $<\mathcal{W}, \mathcal{R}, \mathcal{I}>$	
Frame	$\mathcal{W}$ is a non-empty set of objects (possible worlds)
	$\mathcal{R}$ is a binary relation on W (accessibility relation)
Interpretation	$\mathcal{I}$ is a binary function (interpretation function) that assigns truth values (T or F) to each propositional letter in each world.

Whereas the model's interpretation function assigns truth values to propositional letters, a model's *valuation function* (symbolized as "*v*") is a two-place function that assigns truth values to every well-formed formula (atomic and complex propositions) relative to worlds. More specifically, a valuation $v_{\mathcal{M}}$ of a $\mathbb{ML}$ model $\mathcal{M}$ is a two-place function that assigns truth values (T or F) to every wff relative to each member of $\mathcal{W}$. For example, $v_{\mathcal{M}}$(P, w_1)=T is read as 'the valuation *v* of model $\mathcal{M}$ or P at world w_1 is true.'

Using the valuation function, we can state how the truth value of every wff in $\mathbb{ML}$ is determined:

Where **P** is any propositional letter, and ***w*** is any member of $\mathcal{W}$::

(1) $v_{\mathcal{M}}$ (**P**, w) = $\mathcal{I}$(**P**, w)

Where **P** and **Q** are any well-formed formula, and ***w*** and ***u*** are members of $\mathcal{W}$:

(2) $v_{\mathcal{M}}$ (**P**∧**Q**, w) = T iff $v_{\mathcal{M}}$ (**P**, w) = T and $v_{\mathcal{M}}$ (**Q**, w) = T
(3) $v_{\mathcal{M}}$ (**P**∨**Q**, w) = T iff $v_{\mathcal{M}}$ (**P**, w) = T or $v_{\mathcal{M}}$ (**Q**, w) = T
(4) $v_{\mathcal{M}}$ (**P**→**Q**, w) = T iff $v_{\mathcal{M}}$ (**P**, w) = F or $v_{\mathcal{M}}$ (**Q**, w) = T
(5) $v_{\mathcal{M}}$ (**P**↔**Q**, w) = T iff $v_{\mathcal{M}}$ (**P**, w) = $v_{\mathcal{M}}$ (**Q**, w)
(6) $v_{\mathcal{M}}$ (◊**P**, w) = T iff there is at least one *u* such that $\mathcal{R}$*wu* and $v_{\mathcal{M}}$ (**P**, u)=T
(7) $v_{\mathcal{M}}$ (□**P**, w) = T iff for every *u* that $\mathcal{R}$*wu*, $v_{\mathcal{M}}$ (**P**, u)=T

To clarify (6): ‘It is possibly the case that **P**’ is true at a world w just in the case that **P** is true at some world u that is accessible to w. To clarify (7): ‘It is necessarily the case that **P**’ is true at a world w just in the case that P is true at every world u that w accesses.

Using the valuation function described above, it is possible to determine the truth values given a description of a model. For example, let’s consider the following model $\mathcal{M}$:

$\mathcal{M}$ =<$\mathcal{W}$, $\mathcal{R}$, $\mathcal{I}$>, where
$\mathcal{W} = \{w_1, w_2\}$,
$\mathcal{R} = \{<w_1, w_2>, <w_2, w_1>\}$, and
$\mathcal{I}(P, w_1)$=F, $\mathcal{I}(P, w_2)$=F, $\mathcal{I}(S, w_1)$=T, $\mathcal{I}(S, w_2)$=T

The frame of this model consists of two possible worlds w_1 and w_2, where each of these worlds accesses the other but neither accesses itself. There are two propositional letters for which the interpretation function assigns truth values relative to the two worlds. Using this model and the valuation function specified above, we can determine the truth values of any wff relative to its world. For wffs without any modal operators, the procedure is straightforward: for a given world, use the interpretation of propositional letters along with the truth-functional definitions for the operators to determine the truth value of the wff.

To illustrate, let’s consider $v_{\mathcal{M}}$ (P∧S, w_1):

Determine the truth value of $v_{\mathcal{M}}$ (P∧S, w_1)		
1	$v_{\mathcal{M}}$ (P, w_1)=F since $\mathcal{I}(P, w_1)$=F	Rule 1
2	$v_{\mathcal{M}}$ (S, w_1)=T since $\mathcal{I}(S, w_1)$=T	Rule 1
3	$v_{\mathcal{M}}$ (P∧S, w_1)=F since $v_{\mathcal{M}}$ (P, w_1)=F and $v_{\mathcal{M}}$ (S, w_1)=T	1, 2 + Rule 2

Notice that in the above example, it is only necessary to consider the truth values of “S” and “P” at w_1. The case, however, is different when it comes to wffs with modal operators. In these cases, we valuate a proposition **P** relative to a world ***w***, it is necessary to consider the worlds that ***w*** accesses. To illustrate, let’s consider $v_{\mathcal{M}}$ (□S, w_1):

Determine the truth value of $v_{\mathcal{M}}$ (□S, w_1)		
1	$v_{\mathcal{M}}$ (S, w_2)=T since $\mathcal{I}(S, w_2)$=T	Rule 1
2	$v_{\mathcal{M}}$ (□S, w_1)=T since $\mathcal{R}w_1w_2$ and $v_{\mathcal{M}}$ (S, w_2)=T	1, Rule 7

Since “□S” is true at w_1 if and only if “S” is true at *every* world that w_1 accesses, w_1 accesses w_2 and no other world (even itself), the truth value of “□S” at w_1 depends upon the truth value of “S” at w_2 and no other world.

Note: Add discussion of Duality & Diagrammic Representation of Interpretation (see Non-Classical Logics)

Exercise Set #3

A. *Valuating wffs at a world in a Model*: Consider the following model and determine whether or not the following propositions are true or false:

> $\mathcal{M} = \langle \mathcal{W}, \mathcal{R}, \mathcal{I} \rangle$ where
> $\mathcal{W} = \{w_1, w_2\}$,
> $\mathcal{R} = \{\langle w_1, w_2 \rangle, \langle w_1, w_1 \rangle\}$, and
> $\mathcal{I}(P, w_1)=T$, $\mathcal{I}(P, w_2)=F$, $\mathcal{I}(S, w_1)=T$, $\mathcal{I}(S, w_2)=T$

1. $v_{\mathcal{M}}(\neg P, w_1)$
2. $v_{\mathcal{M}}(P \wedge S, w_2)$
3. $v_{\mathcal{M}}(P \rightarrow S, w_1)$
4. $v_{\mathcal{M}}(\Diamond P, w_1)$
5. $v_{\mathcal{M}}(\Diamond P, w_2)$
6. $v_{\mathcal{M}}(\Box S, w_1)$
7. $v_{\mathcal{M}}(\Box S, w_2)$
8. $v_{\mathcal{M}}(\Diamond \neg P, w_1)$
9. $v_{\mathcal{M}}(\neg \Box P, w_1)$
10. $v_{\mathcal{M}}(\Diamond \Box P, w_1)$

B. *Testing Your Understanding*: Using the above model, what additional information is needed to determine the truth value of the following proposition: $\Diamond \neg P$. What does this say about modal logic semantics and the nature of possibility and necessity?

9.5 Modal Systems: S, D, T, B, S4, S5

In the previous section, a generic definition of a *model* was provided along with a description of how to assign truth values to wffs given a fleshed-out model. In this section, we consider the semantics for various modal systems. There are several obstacles for constructing a semantics for $\mathbb{ML}$. First, as the modal operators $\lozenge$ and $\square$ are not truth-functional, it is not possible to represent the semantics of $\mathbb{ML}$ using truth tables or truth-tree diagrams. Recall that a propositional operator is truth-functional if and only if the truth value of the complex proposition **P**, composed of propositions **R** and **S**, is determined entirely by the truth values of **R** and **S**. That is, since the propositional operator "$\wedge$" is truth-functional, we can determine the truth-value of "$A \wedge B$" given the truth values of "A" and "B". In contrast, "possibly" and "necessarily" (and their corresponding operators) are not truth-functional. That is, the truth values of "possibly **P**" and "necessarily **P**" are not determined by the truth value of **P**. If "possibly' and "necessarily" were truth-functional, then it would be possible to determine the truth value of sentences like "It is possible that John is planning a picnic" from the truth value of "John is planning a picnic." Unfortunately, this cannot be determined for if we suppose that John is *not* planning a picnic, the truth or falsity of "It is possible that John is planning a picnic" will depend upon additional factors we don't have access to: is John even alive to do the planning? is John capable of planning a picnic?

The second obstacle is more problematic, philosophical, and irksome. In propositional and predicate logic, barring a few isolated cases, there is a strong consensus about the formula that should count as tautologies in PL and RL. Proposition of the form "$\mathbf{P} \vee \neg \mathbf{P}$", "$\mathbf{P} \rightarrow \mathbf{P}$", "$\mathbf{P} \rightarrow (\mathbf{Q} \rightarrow \mathbf{P})$" are true independent of the specific propositions we substitute for "**P**" and "**Q**". For a language involving $\lozenge\mathbf{P}$ and $\square\mathbf{P}$, what propositions should count as tautologies and what arguments should count as valid? While there are some formula that clearly should count as tautologies, e.g. $\square(P \vee \neg P)$ and $\neg\lozenge(P \wedge \neg P)$, there are several formula where it is uncertain whether they should count as tautologies, e.g., $\lozenge\square P \rightarrow P$.

Rather than solving this philosophical problem, we sidestep it by developing a variety of different modal systems. These systems are named K, D, T, B, S4, and S5. While each modal system shares a common set of symbols and syntax, and uses the generic notion of a model, each of these systems is distinguished by their semantics. Specifically, K, D, T, B, S4, and S5 are distinguished according to what formal properties belong to the accessibility relation $\mathcal{R}$. These formal properties (and their corresponding systems) are as follows:

Modal System	Property of Accessibility Relation
K	None
D	Serial
T	Reflexive
B	Reflexive and Symmetric
S4	Reflexive and Transitive
S5	Reflexive, Symmetric, and Transitive

Thus, in system S4 (i.e., in a S4-model), the accessibility relation $\mathcal{R}$ is both *reflexive* and *transitive*. That is, for any set of worlds $\mathcal{W}$ in a S4-model, every member of $\mathcal{W}$ accesses itself and for every world in $\mathcal{W}$, if w' accesses w'' and w'' accesses w''', then w' accesses w'''. In contrast, in a D model, the accessibility relation $\mathcal{R}$ is serial such that for every world $w' \in \mathcal{W}$, there is some $w'' \in \mathcal{W}$ such that $\mathcal{R}w'w''$.

The different formal properties which characterize the accessibility relation have an effect on which well-formed formula will count as tautologies, contradictions, contingencies, which arguments will count as valid, and which formula count as being a semantic consequence from a set of formula. That is, in some modal systems, a given argument will count as valid, while in others, it will not.

Valid	An argument $\vDash$ **P** is *valid* in a ML-model ($\mathcal{M}$ =<$\mathcal{W}$, $\mathcal{R}$, $\mathcal{I}$>) iff for every w$\in$W, $v_{\mathcal{M}}$ (**P**, w)=T
Semantic Consequence	A wff **R** is a semantic consequence of a set of propositions {**P**} in an ML-model ($\mathcal{M}$ =<$\mathcal{W}$, $\mathcal{R}$, $\mathcal{I}$>) iff for every w$\in$W, if $v_{\mathcal{M}}$(**S**, w)=T for each **S**$\in${**P**}, then $v_{\mathcal{M}}$(**R**, w)=T.

- Illustration of How Some arguments are invalid & Counter-models
- Illustrations of How Some Arguments are valid in S5 but not in K
- Translations
- Metaphysical Issues.

Exercise Set #4

Exercise Set #2

A. Semantics
1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

B. Models
1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

9.4.2 Normal Modal Logic Semantics

K and its restrictions

for Extensions of K

8.

Exercise Set #3

A. Semantics
1.
2.
3.
4.
5.
6.

7.
8.
9.
10.

9.5 Modal Logic Translation

With the symbols, syntax, and semantics of ML articulated, ML allows for the symbolization of a number of propositions previously impossible with PL. For example, consider the argument put forward at the beginning of this chapter:

It is necessary that John is happy.
Therefore, John is happy.

The language of ML, now makes it possible to translate this argument as follows:

$\Box J \vdash J$

In addition to the above translations, the language of ML makes possible a number of translations that were previously impossible in PL:

English Sentence	**Translation into ML**
It is possible that John is a zombie.	$\Diamond J$
It is necessarily the case that Frank is a vampire.	$\Box F$
It is possible that both John is a zombie and Mary is not a mutant.	$\Diamond (J \wedge \neg M)$
It is not necessarily the case that Frank is a not a vampire or Mary is not a mutant.	$\neg\Box(\neg F \vee \neg M)$

In many cases, translating an English sentence involving the wff in ML can be done in a straightforward manner, assigning upper-case letters to simple sentences, assigning truth-functional operators like "$\neg$" and "$\wedge$" to propositional operators like "not" and "and," and then the modal operators "$\Diamond$" and "$\Box$" to modal terms like terms like "can" and "must." Let's call any translation of this type, a *surface translation*.

Fortunately or unfortunately, not every English sentence can be given a surface translation. For consider the following sentence

(1) If Vic is a bachelor, then he must not be married.

If we represent 'Vic is a bachelor' as 'B' and 'John is not married' as '$\neg$M', a si

(2) $B \rightarrow \Box \neg M$

However, if the above conditional is true and also that *Vic is a bachelor* is true, then it would follow that it is *necessarily* the case that *John is unmarried*. The problem, however, is that it is possible for the

$$(3)\ \Box(B \rightarrow \neg M)$$

The language of ML is capable of disambiguating several different

Exercise Set #4

A. *Translation*: Translate the following wffs from ML into English sentences.

1. It is possible that John is not happy.
2. It is necessary that Liz plays guitar.
3. If John isn't happy, then he should stop drinking.
4. It is possible that Liz will go to the party and it is possible that she won't.
5. John may or may not get the job.
6. It is not possible for Barack Obama to win a third term.
7.
8.
9.
10.

B. *Translation*: Translate the following wffs English sentences into wffs in ML.

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

9.6 What are possible worlds?

The semantics of modal logic may strike you as a useful way of representing the truth and falsity of propositions that involve modal terms, but the notion of a *possible world* raises several questions about the status of possible worlds and perhaps even cast some doubt on the nature of modal logic as a whole. Do possible worlds really exist? If they do, in what sense do they exist? If they don't, what does this mean for modal logic? How do possible worlds relate to the actual world?

9.6.1 Modal Realism: Possible worlds exist as concrete objects

9.6.2 Modal Actualism: Possible worlds are abstract entities, they don't really exist

9.6.3 Meinongianism: Possible worlds are non-existent objects

Exercise Set #5

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

Chapter 10

Modal Logic Trees

In chapters 4 and 7, truth trees were used to represent various semantic properties of propositions, sets of propositions, and arguments. Truth trees, much like truth-tables, are an effective decision procedure for propositional logic and a partial-decision procedure for predicate logic. In this chapter, the truth-tree method is formulated for modal logic.

but it will also allow for us to test propositions, sets of propositions, and arguments for various logical properties (consistency, contingency, validity, etc.).

10.1. Modal Logic Tree Setup and Representation

Propositional modal logic trees (hereafter modal trees) are setup in the same way that truth trees are in propositional logic. In terms of decomposition, as ML is an extension of PL, propositional modal logic trees extend the truth-tree method from propositional logic. Thus, many propositions will be decomposed using the propositional logic decomposition rules. For example, "P∧R" will be decomposed using ∧D.

However, one difference between the propositional and modal trees is that the truth of every proposition in a modal tree is relative to a world. In order to indicate this, we write *w* with a subscripted integer so as to specify which world that particular proposition is being evaluated at.

Modal Logic Tree

1	$\neg[P\rightarrow\neg(P\vee Q)]$, w_0	P
2	$\neg P$, w_0	1 $\neg\rightarrow$D
3	$\neg(P\vee Q)$, w_0	1 $\neg\rightarrow$D
4	$\neg P$, w_0	3 $\neg\vee$D
5	$\neg Q$, w_0	3 $\neg\vee$D

However, to simplify, we will abbreviate our notation by using positive integers and dropping the use of *w* at each line. Thus, the modal tree above can be simplified as follows:

1	$\neg[P\rightarrow\neg(P\vee Q)]$, $_0$	P
2	$\neg P$, $_0$	1 $\neg\rightarrow$D
3	$\neg(P\vee Q)$, $_0$	1 $\neg\rightarrow$D

4	¬P, $_0$	3 ¬∨D
5	¬Q, $_0$	3 ¬∨D

There are four quantifier rules

10.2. Negated Box and Diamond Decomposition

The first two modal tree decomposition rules concern propositions where a negation operates upon a modal operator.

Negated Diamond Decomposition (¬◊D)	**Negated Box Decomposition (¬□D)**
¬◊**P**, *i* ✓ □¬**P**, *i*	¬□**P**, *i* ✓ ◊¬**P**, *i*

Negated diamond decomposition and negated box decomposition.

1	¬◊(P∧R), $_0$ ✓	P
2	¬□(P→R), $_1$ ✓	P
3	□¬(P∧R), $_0$	1¬◊D
4	◊¬(P→R), $_1$	2¬□D

10.3. Box and Diamond Decomposition

The second two modal tree decomposition rules concern propositions where the main operator is a modal operator.

Diamond Decomposition (◊D)	**Box Decomposition (□D)**
◊P, *I* ✓ $_i R_j$ P, *j* where *j* is a world that does not previously occur in the branch	□P, *i* $_i R_j$ ~P, *j* where $w_i R w_j$ occurs somewhere in the branch

In the case of diamond decomposition

In the case of box decomposition

10.4 Analysis with Modal Logic Trees

At the beginning of the chapter 9, it was noted that the following argument is valid but that there was no way to express this validity in the less expressive PL:

> It is necessary that John is happy.
> Therefore, John is happy.

It

Chapter 11

Modal Logic Derivations