1

MODAL LOGIC: A SEMANTIC PERSPECTIVE

Patrick Blackburn and Johan van Benthem

1 INTRODUCTION

This chapter introduces modal logic from a semantic perspective. That is, it presents modal logic as a tool for talking about *structures* or *models*. But what kind of structures can modal logic talk about?

There is no single answer. For example, modal logic can be given an *algebraic semantics*, and under this interpretation modal logic is a tool for talking about what are known as boolean algebras with operators. And modal logic can be given a *topological semantics*, so it can also be viewed as a tool for talking about topologies. But although we briefly discuss algebraic and topological semantics, for the most part this chapter focuses on modal logic as a tool for talking about *graphs*. To put it another way, this chapter is devoted to what is known as the *relational* or *Kripke* semantics for modal logic. This is the best known and (with the possible exception of algebraic semantics) the best explored style of modal semantics. It is also, arguably, the most intuitive. Over the years modal logic has been applied in many different ways. It has been used as a tool for reasoning about time, beliefs, computational systems, necessity and possibility, and much else besides. These applications, though diverse, have something important in common: the key ideas they employ (flows of time, relations between epistemic alternatives, transitions between computational states, networks of possible worlds) can all be represented as simple graph-like structures. And as we shall see, modal logic is an interesting tool for talking about such structures: it provides an internal perspective on the information they contain.

But modal logic is not the only tool for talking about graphs, and this brings us to one of the major themes of the chapter: the relationship between modal logic and other forms of logic. As we shall see, under the graph-based perspective discussed here, modal logic is closely linked to both first- and second-order classical logic. This immediately raises interesting questions. How does modal logic compare with these logics as a tool for talking about graphs? Can modal expressivity over graphs be characterised in terms of classical logic? We shall ask (and answer) such questions in the course of the chapter.

Games (in various guises) are another recurring motif. The simple way that modal formulas are interpreted on graphs naturally gives rise to games and game-like concepts. The most important of these is the notion of *bisimulation*. This is a relation between two models, weaker than isomorphism, which can be thought of as giving rise to a transition-matching game between two players. As we shall see, this concept holds the key to modal model theory and characterises the link with first-order logic.

This chapter has two pedagogical goals. The first is to provide a bread-and-butter introduction to relational semantics for modal logic that can be used as a basis for tackling the more advanced chapters in this handbook. Thus the reader will find here definitions and discussions of all the basic tools needed in modal model theory (such as the standard translation, generated submodels, bounded morphisms, and so on). Basic results about these concepts are stated and some simple proofs are given. But we have a second, more ambitious, goal: to help the reader start thinking semantically. We want to give the reader a sense of how modal logicians view structure, and what they look for when exploring new logics. To this end we have tried to isolate the intuitions that guide working modal logicians, and to present them vividly. We also make numerous asides, some of which touch on advanced logical topics. Their purpose is to situate the key ideas in a wider context, and even beginners should try to follow them.

Here is our plan. In Section 2, we introduce basic modal languages and the graphs over which they are interpreted. We give the satisfaction definition (which tells us how to interpret modal formulas in graphs) and the standard translation (which links modal logic with classical logic).

With these preliminaries out of the way, we are ready to go deeper. What can (and cannot) modal languages say about graphs? In Section 3 we introduce the notion of bisimulation and use it to develop some answers; among other things, we characterise modal logic as a fragment of first-order logic. In Section 4 we examine the computability and computational complexity of modal logic. A shift of topic? Not at all. In essence, this section examines modal logic as a tool for talking about *finite* graphs. In Section 5 we move to the level of frames and re-examine the link between modal and classical logic. As we shall see, at this level the fundamental correspondence is between modal logic and (monadic) second-order logic. In Section 6 we move beyond the basic modal language and discuss a number of richer languages that offer more expressivity. But what makes them all modal? As we shall see, many of the themes explored in earlier sections re-emerge, and point towards an idea that seems to lie at the heart of modal logic: guarding. Moreover, in some cases it is possible to prove Lindström-style characterisation results. In Section 7 we discuss three alternatives to relational semantics, namely algebraic, neighbourhood, and topological semantics. We conclude in Section 8.

Two final remarks. First, although we introduce modal logic from scratch, we assume that the reader has at least a basic understanding of classical first-order logic (especially its model-theoretic semantics) and some grasp of the notion of computability. Any standard introduction to mathematical logic (Enderton [37] is a good choice) supplies more than enough material to follow the main line of the chapter. Second, we *don't* discuss modal proof-theory or related notions such as completeness in any detail (these topics are the focus of Chapter 2 of this handbook). Although we haven't banished all mention of normal modal logics and completeness from the chapter, in our view traditional introductions to modal logic tend to overemphasise these topics. We want this chapter to act as a counterbalance. As we hope to convince the reader, simply asking the question "But what can I *say* with these languages?" swiftly leads to interesting territory.

2 BASIC MODAL LOGIC

In this section we introduce the basic modal language and its relational semantics. We define basic modal syntax, introduce models and frames, and give the satisfaction definition. We then draw the reader's attention to the internal perspective that modal languages offer on relational structure, and explain why models and frames should be thought of as graphs. Following this we give the standard translation. This enables us to convert any basic modal formula into a first-order formula with one free variable. The standard translation is a bridge between the modal and classical worlds, a bridge that underlies much of the work of this chapter.

2.1 First steps in relational semantics

Suppose we have a set of proposition symbols (whose elements we typically write as p, q, r and so on) and a set of modality symbols (whose elements we typically write as m, m', m'', and so on). The choice of PROP and MOD is called the *signature* (or *similarity type*) of the language; in what follows we'll tacitly assume that PROP is denumerably infinite, and we'll often work with signatures in which MOD contains only a single element. Given a signature, we define the *basic modal language* (over the signature) as follows:

$$\varphi \quad ::= \quad p \mid \top \mid \bot \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \varphi \to \psi \mid \varphi \leftrightarrow \psi \mid \langle m \rangle \varphi \mid [m]\varphi.$$

That is, a basic modal formula is either a proposition symbol, a boolean constant, a boolean combination of basic modal formulas, or (most interesting of all) a formula prefixed by a diamond