2 The modal $\mu$-calculus

This chapter is a first introduction to the modal $\mu$-calculus. We define the language, discuss some syntactic issues, and then proceed to its game-theoretic semantics. As a first result, we prove that the modal $\mu$-calculus is bisimulation invariant, and has a strong, ‘bounded’ version of the tree model property. We start with an example.

Example 2.1 Consider the formula $\diamond (d^*)p$ from propositional dynamic logic. By definition, this formula holds at those points in an LTS $S$ from which there is a finite $R_d$-path, of unspecified length, leading to a state where $p$ is true.

We leave it for the reader to prove that

$$S, s \models (d^*)p \leftrightarrow (p \lor \langle d \rangle (d^*)p)$$

for any pointed transition system $(S, s)$ (here we write $(d)$ rather than $\diamond_d$). Informally, one might say that $(d^*)p$ is a fixed point or solution of the ‘equation’

$$x \leftrightarrow p \lor (d)x.$$  \hspace{1cm} (5)

One may show, however, that $(d^*)p$ is not the only fixpoint of (5). If we let $\infty_d$ denote a formula that is true at those states of a transition system from which an infinite $d$-path emanates, then the formula $(d^*)p \lor \infty_d$ is another fixed point of (5).

In fact, one may prove that the two mentioned fixpoints are the smallest and largest possible solutions of (5), respectively.

As we will see in this chapter, the modal $\mu$-calculus allows one to explicitly refer to such smallest and largest solutions. For instance, as we will see further on, the smallest and largest solution of the ‘equation’ (5) will be written as $\mu x.p \lor (d)x$ and $\nu x.p \lor (d)x$, respectively.

To arrive at the semantics of modal fixpoint formulas one can take two roads. In Chapter 3 we will introduce the algebraic semantics of $\mu x.\varphi$ and $\nu x.\varphi$ in an LTS $S$, in terms of the least and greatest fixpoint, respectively, of some algebraically defined meaning function. For this purpose, we will consider the formula $\varphi$ as an operation on the power set of (the state space of) $S$, and we have to prove that this operation indeed has a least and a greatest fixpoint. As we will see, this formal definition of the semantics of the modal $\mu$-calculus may be mathematically transparent, but it is of little help when it comes to unravelling and understanding the actual meaning of individual formulas. In practice, it is much easier to work with the evaluation games that we will introduce in this chapter.

This framework builds on the game-theoretical semantics for ordinary modal logic as described in subsection 1.2, extending it with features for the fixpoint operators and for the bound variables of fixpoint formulas (such as $x$ in the formula $\mu x.p \lor \diamond x$). The key difference lies in the fact that when a match of an evaluation game reaches a position of the form $(x, s)$, with $x$ a bound variable, then an equation such as (5) is used to unfold the variable $x$ into its associated formula (in the example, the formula $p \lor \diamond x$).

As a consequence, the flavour of these games is remarkably different from the evaluation games we met before. Recall that in evaluation matches for basic modal formulas, the formula
Lectures on the modal $\mu$-calculus

is broken down, step by step, until we can declare a winner of the match. From this it follows that the length of such a match is bounded by the length of the formula. Evaluation matches for fixpoint formulas, on the other hand, can last infinitely long, if some fixpoint variables are unfolded infinitely often. Hence, the game-theoretic semantics for fixpoint logics takes us to the area of infinite games. In this chapter we keep the discussion on infinite games informal, in Chapter 5 the reader can find precise definitions of all notions that we introduce here.

2.1 Syntax

As announced already in the previous chapter, in the case of fixpoint formulas we will usually work with formulas in positive normal form in which the only admissible occurrences of the negation symbol is in front of atomic formulas.

Definition 2.2 Given a set $D$ of atomic actions, we define the collection $\mu ML_D$ of (poly-) modal fixpoint formulas as follows:

$$\varphi ::= \top | \bot | p | \neg p | \varphi \land \varphi | \varphi \lor \varphi | \Diamond_d \varphi | \Box_d \varphi | \mu x. \varphi | \nu x. \varphi$$

where $p$ and $x$ are propositional variables, and $d \in D$. There is a restriction on the formation of the formulas $\mu x. \varphi$ and $\nu x. \varphi$, namely, that all occurrences of $x$ in $\varphi$ are positive. That is, no occurrence of $x$ in $\varphi$ may be in the scope of the negation operator $\neg$.

In case the set $D$ of atomic actions is a singleton, we will simply speak of the modal $\mu$-calculus, notation: $\mu ML$.

The syntactic combinations $\mu x$ and $\nu x$ are called the least and greatest fixpoint operators, respectively. We use the symbol $\mu$ to denote either $\mu$ or $\nu$. A fixpoint formula of the form $\mu x. \varphi$ is called a $\mu$-formula, while $\nu$-formulas are the ones of the form $\nu x. \varphi$.

The concepts of subformula and proper subformula are extended from basic modal logic to the modal $\mu$-calculus in the obvious way.

Definition 2.3 We add the inductive clause

$$Sfor(\eta x. \varphi) := \{\eta x. \varphi\} \cup Sfor(\varphi), \quad \text{where } \eta \in \{\mu, \nu\}$$

to the definition of the collection $Sfor(\xi)$ of the collection of subformulas of $\varphi \in \mu ML_D$ (cf. Definition ??). We write $\varphi \preceq \psi$ if $\varphi$ is a subformula of $\psi$, using $<$ for the proper subformula relation.

Definition 2.4 The size $|\varphi|$ of $\varphi$ as the size of this set $Sfor(\varphi)$, that is, $|\varphi|$ is the number of subformulas of $\varphi$.

- Discuss alternative notions of size

Syntactically, the fixpoint operators are very similar to the quantifiers of first-order logic in the way they bind variables.
Definition 2.5 Fix a formula \( \varphi \). The sets \( FV(\varphi) \) and \( BV(\varphi) \) of free and bound variables of \( \varphi \) are defined by the following induction on \( \varphi \):

\[
\begin{align*}
FV(\bot) & := \emptyset & BV(\bot) & := \emptyset \\
FV(\top) & := \emptyset & BV(\top) & := \emptyset \\
FV(p) & := \{p\} & BV(p) & := \emptyset \\
FV(\neg p) & := \{p\} & BV(\neg p) & := \emptyset \\
FV(\varphi \lor \psi) & := FV(\varphi) \cup FV(\psi) & BV(\varphi \lor \psi) & := BV(\varphi) \cup BV(\psi) \\
FV(\varphi \land \psi) & := FV(\varphi) \cup FV(\psi) & BV(\varphi \land \psi) & := BV(\varphi) \cup BV(\psi) \\
FV(\Box_d \varphi) & := FV(\varphi) & BV(\Box_d \varphi) & := BV(\varphi) \\
FV(\Diamond_d \varphi) & := FV(\varphi) & BV(\Diamond_d \varphi) & := BV(\varphi) \\
FV(\eta x. \varphi) & := FV(\varphi) \setminus \{x\} & BV(\eta x. \varphi) & := BV(\varphi) \cup \{x\}
\end{align*}
\]

For a finite set of propositional variables \( P \), we let \( \mu ML_D(P) \) denote the set of \( \mu ML_D \)-formulas \( \varphi \) of which all free variables belong to \( P \).

Formulas like \( x \lor x.(p \land \Diamond \mu x. \Box x) \) may be well formed, but in practice they are very hard to read and to work with. In the sequel we will almost exclusively work with formulas in which every bound variable uniquely determines a fixpoint operator binding it, and in which there is no overlap between free and bound variables.

Definition 2.6 A formula \( \varphi \in \mu ML_D \) is clean if no two distinct (occurrences of) fixed point operators in \( \varphi \) bind the same variable, and no variable has both free and bound occurrences in \( \varphi \). If \( x \) is a bound variable of the clean formula \( \varphi \), we let \( \varphi_x = \eta x. \delta_x \) denote the unique subformula of \( \varphi \) where \( x \) is bound by the fixpoint operator \( \eta_x \).

Convention 2.7 As a notational convention, we will use the letters \( p, q, r, \ldots \) and \( x, y, x, \ldots \) to denote, respectively, the free and the bound propositional variables of a \( \mu ML_D \)-formula. This convention can be no more than a guideline, since the division between bound and free variables will never be the same for a formula and its subformulas.

An important role in the theory of the modal \( \mu \)-calculus is played by a certain order on the bound variables of a formula. The idea behind this ‘dependency order’ is that if \( x \leq y \), the meaning of \( \delta_x \) is (in principle) dependent on the meaning of \( y \), because \( y \) may occur freely in \( \delta_x \); observe that this can only happen if \( \delta_x \preceq \delta_y \).

Definition 2.8 Given a clean formula \( \varphi \), we define a dependency order on the set \( BV(\varphi) \), saying that \( y \) ranks higher than \( x \), notation: \( x \preceq y \) iff \( \delta_x \preceq \delta_y \).

We finish our sequence of syntactic definitions with the notion of guardedness, which will become important later on.

Definition 2.9 A variable \( x \) is guarded in a \( \mu ML_D \)-formula \( \varphi \) if every occurrence of \( x \) in \( \varphi \) is in the scope of a modal operator. A formula \( \xi \in \mu ML_D \) is guarded if for every subformula of \( \xi \) of the form \( \eta x. \delta_x \), \( x \) is guarded in \( \delta_x \).
2.2 Game semantics

For a definition of the evaluation game of the modal $\mu$-calculus, fix a clean formula $\xi$ and an LTS $S$. Basically, the game $E(\xi, S)$ for $\xi$ a fixpoint formula is defined in the same way as for plain modal logic formulas.

Definition 2.10 Given a clean modal $\mu$-calculus formula $\xi$ and a transition system $S$, we define the evaluation game $E(\xi, S)$ as a board game with players $\exists$ and $\forall$ moving a token around positions of the form $(\varphi, s) \in S_{for}(\xi) \times S$. The rules, determining the admissible moves from a given position, together with the player who is supposed to make this move, are given in Table 3.

As before, $E(\xi, S)@((\xi, s))$ denotes the instantiation of this game where the starting position is fixed as $(\xi, s)$.

One might expect that the main difference with the evaluation game for basic modal formulas would involve the new formula constructors of the $\mu$-calculus: the fixpoint operators. Perhaps surprisingly, the fixpoint operators themselves are dealt with in the most straightforward way possible: the successor of a position of the form $(\tau x.\cdot, s)$ is simply obtained as the pair $(\delta, s)$. Since this next position is thus uniquely determined, the position $(\tau x.\cdot, s)$ will not be assigned to either of the players.

The crucial difference lies in the treatment of the bound variables of a fixpoint formula $\xi$. Previously, all positions of the form $(p, s)$ would be final positions of the game, immediately determining the winner of the match, and this is still the case here if $p$ is a free variable. However, at a position $(x, s)$ with $x$ bound, the fixpoint variable $x$ gets unfolded; this means that the new position is given as $(\delta x, s)$, where $\eta x.\cdot x$ is the unique subformula of $\xi$ where $x$ is bound. Note that for this to be well defined, we need $\xi$ to be clean. The disjointness of $FV(\xi)$ and $BV(\xi)$ ensures that it is always clear whether a variable is to be unfolded or not, and the fact that bound variables are bound by unique occurrences of fixpoint operators guarantees that $\delta x$ is uniquely determined. Finally, since in this case the next position is also completely determined by the current one, positions of the form $(x, s)$ with $x$ bound are assigned to neither of the players.

Example 2.11 Let $S = (S, R, V)$ be the Kripke model based on the set $S = \{0, 1, 2\}$, with $R = \{(0, 1), (1, 1), (1, 2), (2, 2)\}$, and $V$ given by $V(p) = \{2\}$. Now let $\xi$ be the formula $\eta x. p \lor \Box x$, and consider the game $E(\xi, S)$ initialized at $(\xi, 0)$.

The second position of any match of this game will be $(p \lor \Box x, 0)$ belonging to $\exists$. Assuming that she wants to win, she chooses the disjunct $\Box x$ since otherwise $p$ being false at 0 would mean an immediate loss for her. Now the position $(\Box x, 0)$ belongs to $\forall$ and he will make the only move allowed to him, choosing $(x, 1)$ as the next position. Here an automatic move is made, unfolding the variable $x$, and thus changing the position to $(p \lor \Box x, 1)$. And as before, $\exists$ will choose the right disjunct: $(\Box x, 1)$.

At $(\Box x, 1)$, $\forall$ does have a choice. Choosing $(x, 2)$, however, would mean that $\exists$ wins the match since $p$ being true at 2 enables her to finally choose the first disjunct of the formula $p \lor \Box x$. So $\forall$ chooses $(x, 1)$, a position already visited by the match before.
The modal $\mu$-calculus

<table>
<thead>
<tr>
<th>Position</th>
<th>Player</th>
<th>Admissible moves</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(\varphi_1 \lor \varphi_2, s)$</td>
<td>$\exists$</td>
<td>${((\varphi_1, s), (\varphi_2, s))}$</td>
</tr>
<tr>
<td>$(\varphi_1 \land \varphi_2, s)$</td>
<td>$\forall$</td>
<td>${((\varphi_1, s), (\varphi_2, s))}$</td>
</tr>
<tr>
<td>$(\Diamond d \varphi, s)$</td>
<td>$\exists$</td>
<td>${((\varphi, t)</td>
</tr>
<tr>
<td>$(\Box q \varphi, s)$</td>
<td>$\forall$</td>
<td>${((\varphi, t)</td>
</tr>
<tr>
<td>$(\bot, s)$</td>
<td>$\exists$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$(\top, s)$</td>
<td>$\forall$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$(p, s)$, with $p \in \text{FV}(\xi)$ and $s \in V(p)$</td>
<td>$\forall$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$(p, s)$, with $p \notin \text{FV}(\xi)$ and $s \notin V(p)$</td>
<td>$\exists$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$(\neg p, s)$, with $p \in \text{FV}(\xi)$ and $s \in V(p)$</td>
<td>$\exists$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$(\neg p, s)$, with $p \notin \text{FV}(\xi)$ and $s \notin V(p)$</td>
<td>$\forall$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$(\eta \cdot ! x. \delta_x, s)$</td>
<td>$-$</td>
<td>${ \delta_x, s }$</td>
</tr>
<tr>
<td>$(x, s)$, with $x \in \text{BV}(\xi)$</td>
<td>$-$</td>
<td>${ \delta_x, s }$</td>
</tr>
</tbody>
</table>

Table 3: Evaluation game for modal fixpoint logic

This means that these strategies force the match to be infinite, with the variable $x$ unfolding infinitely often at positions of the form $(x, 1)$, and the match taking the following form:

$$(\xi, 0)(p \lor \Box x, 0)(\Box x, 0)(x, 1)(p \lor \Box x, 1)(\Box x, 1)(x, 1)(p \lor \Box x, 1)\ldots$$

So who is declared to be the winner of this match? This is where the difference between the two fixpoint operators shows up. In case $\eta = \mu$, the above infinite match is lost by $\exists$ since the fixpoint variable that is unfolded infinitely often is a $\mu$-variable, and $\mu$-variables are to be unfolded only finitely often. In case $\eta = \nu$, the variable unfolded infinitely often is a $\nu$-variable, and this is unproblematic: $\exists$ wins the match.

The above example shows the principle of unfolding at work. Its effect is that matches may now be of infinite length since formulas are no longer deconstructed at every move of the game. Nevertheless, as we will see, it will still be very useful to declare a winner of such an infinite game. Here we arrive at one of the key ideas underlying the semantics of fixpoint formulas, which in a slogan can be formulated as follows:

$\nu$ means unfolding, $\mu$ means finite unfolding.

Giving a more detailed interpretation to this slogan, in case of a unique variable that is unfolded infinitely often during a match $\Sigma$, we will declare $\exists$ to be the winner of $\Sigma$ if this variable is a $\nu$-variable, and $\forall$ in case we are dealing with a $\mu$-variable. But what happens in case that various variables are unfolded infinitely often? As we shall see, in these cases there is always a unique such variable that ranks higher than any other such variable.

**Definition 2.12** Let $\xi$ be a clean $\mu\text{MLQ}$-formula, and $S$ a labelled transition system. A match of the game $E(\xi, S)$ is a (finite or infinite) sequence of positions

$$\Sigma = (\varphi_i, s_i)_{i < \kappa}$$
(where \( \kappa \) is either a natural number or \( \omega \)) which is in accordance with the rules of the evaluation game — that is, \( \Sigma \) is a path through the game graph given by the admissibility relation of Table 3. A full match is either an infinite match, or a finite match in which the player responsible for the last position got stuck. In practice we will always refer to full matches simply as matches. A match that is not full is called partial.

Given an infinite match \( \Downarrow \), we let \( \text{Unf}^\infty(\Downarrow) \subseteq \text{BV}(\xi) \) denote the set of variables that are unfolded infinitely often during \( \Downarrow \).

**Proposition 2.13** Let \( \Downarrow \) be a clean \( \mu \)-MLD-formula, and \( S \) a labelled transition system. Then for any infinite match \( \Sigma \) of the game \( E(\Downarrow, S) \), the set \( \text{Unf}^\infty(\Sigma) \) has a highest ranking member, in terms of the dependency order of Definition 2.8.

**Proof.** Since \( \Downarrow \) is an infinite match, the set \( \text{Unf}^\infty(\Sigma) \) is not empty. We claim that it is in fact directed (with respect to the ranking order). That is, for any \( x \) and \( y \) in \( \text{Unf}^\infty(\Sigma) \) there is a variable \( z \in \text{Unf}^\infty(\Sigma) \) such that \( x \leq_\xi z \) and \( y \leq_\xi z \).

For suppose otherwise. Then in particular, \( \varphi_x = \eta_x x.\delta_x \) and \( \varphi_y = \eta_y y.\delta_y \) are not subformulas of one another. However, \( \Sigma \) passes through both \( \varphi_x \) and \( \varphi_y \) infinitely often. Now the only way it can move from \( \varphi_x \) to \( \varphi_y \) is by unfolding some variable \( z \) such that both \( \varphi_x \) and \( \varphi_y \) are subformulas of \( \varphi_z \), that is, \( x \leq_\xi z \) and \( y \leq_\xi z \). Since this happens infinitely often, one of these variables \( z \) must belong to \( \text{Unf}^\infty(\Sigma) \), as required.

But if \( \text{Unf}^\infty(\Sigma) \) is directed, being finite it must have a maximum. That is, there is indeed a highest variable in \( \text{BV}(\xi) \) that gets unfolded infinitely often during \( \Sigma \). QED

Given this result, there is now a natural formulation of the winning conditions for infinite matches of evaluation games.

**Definition 2.14** Let \( \Downarrow \) be a clean \( \mu \text{-MLD} \)-formula. The winning conditions of the game \( E(\Downarrow, S) \) are given in Table 4.

*Table 4: Winning conditions of \( E(\Downarrow, S) \)*

<table>
<thead>
<tr>
<th>( \Sigma ) is finite</th>
<th>( \exists ) wins ( \Sigma )</th>
<th>( \forall ) wins ( \Sigma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Downarrow ) got stuck</td>
<td>( \exists ) got stuck</td>
<td>( \forall ) got stuck</td>
</tr>
</tbody>
</table>

We can now formulate the game-theoretic semantics of the modal \( \mu \)-calculus as follows.

**Definition 2.15** Let \( \xi \) be a clean formula of the modal \( \mu \)-calculus, and let \( S \) be a transition system of the appropriate type. Then we say that \( \xi \) is (game-theoretically) satisfied at \( s \), notation: \( S, s \models g \xi \) if \( (s, \xi) \in \text{Win}_S(E(\xi, S)) \).

**Remark 2.16** As mentioned we have kept this introduction to evaluation games for fixpoint formulas rather informal, referring to Chapter 5 for a more rigorous discussion of infinite
games. Nevertheless, we want to mention already here that evaluation games, on the ground of being so-called *parity games*, have two very useful properties that make them attractive to work with. To start with, every evaluation game is determined in the sense that every position is winning for exactly one of the two players. And second, one may show that winning strategies for either player of an evaluation game, can always be assumed to be positional, that is, do not depend on moves made earlier in the match, but only on the current position. Combining this, evaluation games enjoy *positional determinacy*; that is, every position \((\varphi, s)\) is winning for exactly one of the two players, and each player \(\Pi \in \{\exists, \forall\}\) has a positional strategy \(f_\Pi\) which is winning for the game \(E(\xi, S)@((\varphi, s)\) for every position \((\varphi, s)\) that is winning for \(\Pi\).

**Remark 2.17** Observe that we have defined the game-theoretic semantics for *clean* formula only. It is certainly possible to extend this definition to arbitrary fixpoint formulas; a straightforward approach would be to involve the construction tree of a non-clean formula \(\xi\), and redefine a *position* of the evaluation game \(E(\xi, S)\) to be a pair, consisting of a node in this construction tree and a point in the Kripke structure. Alternatively, one may work with a clean alphabetical variant of the formula \(\xi\); once we have given the algebraic semantics for arbitrary formulas, it is not hard to show that in that semantics, alphabetic variants are equivalent.

### 2.3 Examples

**Example 2.18** As a first example, consider the formulas \(\eta x.p \lor x\), and fix a Kripke model \(S\). Observe that any match of the evaluation game \(E(\eta x.p \lor x, S)\) starting at position \((\eta x.p \lor x, s)\) immediately proceeds to position \((p \lor x, s)\), after which \(\exists\) can make a choice. In case \(\eta\) is the least fixpoint operator, \(\eta = \mu\), we claim that \(S, s \models \mu x.p \_x\) if and only if \(s \in V(p)\).

For the direction from right to left, assume that \(s \in V(p)\). Now, if \(\exists\) chooses the disjunct \(p\) at the position \((s, p \lor x)\), she wins the match because \(\forall\) will get stuck at \((s, p)\). Hence \(s \in \text{Win}_\exists(E(\mu x.p \lor x, S))\).

On the other hand, if \(s \notin V(p)\), then \(\exists\) will lose if she chooses the disjunct \(p\) at position \((s, p \lor x)\). So she must choose the disjunct \(x\) which then unfolds to \(p \lor x\) so that \(\exists\) is back at the position \((s, p \lor x)\). Thus if \(\exists\) does not want to get stuck, her only way to survive is to keep playing the position \((s, x)\), thus causing the match to be infinite. But such a match is won by \(\forall\) since the only variable that gets unfolded infinitely often is a \(\mu\)-variable. Hence in this case we see that \(s \notin \text{Win}_\exists(E(\nu x.p \lor x, S))\).

If on the other hand we consider the case where \(\eta = \nu\), then \(\exists\) can win any match:

\[S, s \models_\nu \nu x.p \lor x.\]

It is easy to see that now, the strategy of always choosing the disjunct \(x\) at a position of the form \((s, p \lor x)\) is winning. For, it forces all games to be infinite, and since the only fixpoint variable that gets ever unfolded here is a \(\nu\)-variable, all infinite matches are won by \(\exists\).

Concluding, we see that \(\mu x.p \lor x\) is equivalent to the formula \(p\), and \(\nu x.p \lor x\), to the formula \(\top\).
Example 2.19 Now we turn to the formulas $\mu x.\Box x$ and $\nu x.\Diamond x$. First consider how a match for any of these formulas proceeds. The first two positions of such a match will be of the form $(\eta x.\circ x, s)(\circ x, s)$, at which point it is $\exists$’s turn to make a move. Now she either is stuck (in case the state $s$ has no successor) or else the next two positions are $(x, t)(\circ x, t)$ for some successor $t$ of $s$, chosen by $\exists$. Continuing this analysis, we see that there are two possibilities for a match of the game $\mathcal{E}(\eta x.\circ x, \mathbb{S})$:

1. the match is an infinite sequence of positions

$$(\eta x.\circ x, s_0)(\circ x, s_0)(x, s_1)(\circ x, s_1)(x, s_2)\ldots$$

   corresponding to an infinite path $s_0R s_1 R s_2 R\ldots$ through $\mathbb{S}$.

2. the match is a finite sequence of positions

$$(\eta x.\circ x, s_0)(\circ x, s_0)(x, s_1)(\circ x, s_1)\ldots(\circ x, s_k)$$

   corresponding to a finite path $s_0R s_1 R\ldots s_k$ through $\mathbb{S}$, where $s_k$ has no successors.

Note too that in either case it is only $\exists$ who has turns, and that her strategy corresponds to choosing a path through $\mathbb{S}$. From this it is easy to derive that

- $\mu x.\Box x$ is equivalent to the formula $\perp$,
- $\mathbb{S}, s \models_r \nu x.\Diamond x$ iff there is an infinite path starting at $s$.

Example 2.20 Let $\xi$ be the following formula:

$$\xi = \nu x.\mu y. (p \land \Box x) \lor (\neg p \land \Diamond y)$$

Then we claim that for any LTS $\mathbb{S}$, and any state $s$ in $\mathbb{S}$:

$$\mathbb{S}, s \models_g \xi \text{ iff } \text{there is some path from } s \text{ on which } p \text{ is true infinitely often.} \quad (6)$$

To see this, first suppose that there is a path $\Sigma = s_0s_1s_2\ldots$ as described in the right hand side of (6) and suppose that $\exists$ plays according to the following strategy:

(a) at a position $(\alpha_p \lor \alpha_{\neg p}, t)$, choose $(\alpha_p, t)$ if $\mathbb{S}, t \models_g p$ and choose $(\alpha_{\neg p}, t)$ otherwise;

(b) at a position $(\Diamond \varphi, t)$, distinguish cases:
   - if the match so far has followed the path, with $t = s_k$, choose $(\varphi, s_{k+1})$;
   - otherwise, choose an arbitrary successor (if possible).
We claim that this is a winning strategy for $\exists$ in the evaluation game initialized at $(\xi, s)$. Indeed, since $\exists$ always chooses the propositionally safe disjunct of $\alpha_p \lor \lnot \alpha_{\neg p}$, she forces $\forall$, when faced with a position of the form $(\alpha_{\neg p}, t) = (\lnot p \land \Diamond z, t)$ to always choose the diamond conjunct $\Diamond z$, or lose immediately. In this way she guarantees to always get to positions of the form $(\Diamond z, s_i)$, and thus she can force the match to last infinitely long, following the infinite path $\Sigma$. But why does she actually win this match? The point is that, whenever she chooses $\alpha_p$, three positions later, $x$ will be unfolded, and likewise with $\lnot \alpha_{\neg p}$ and $y$. Thus, $p$ being true infinitely often on $\Sigma$ means that the $\nu$-variable $x$ gets unfolded infinitely often. And so, even though the $\mu$-variable $y$ might get unfolded infinitely often as well, she wins the match since $x$ ranks higher than $y$ anyway.

For the other direction, assume that $\mathbf{S}, s \models \neg \xi$ so that $\exists$ has a winning strategy in the game $\mathcal{E}(\xi, \mathbf{S})$ initialized at $(\xi, s)$. It should be clear that any winning strategy must follow (a) above. So whenever $\forall$ faces a position $(p \land \Diamond z, t)$, $p$ will be true, and likewise with positions $(\lnot p \land \Diamond z, t)$. Now consider a match in which $\forall$ plays propositionally sound, that is, always chooses the diamond conjunct of these positions. This match must be infinite since both players will stay alive forever: $\forall$ because he can always choose a diamond conjunct, and $\exists$ because we assumed her strategy to be winning. But a second consequence of $\exists$ playing a winning strategy, is that it cannot happen that $y$ is unfolded infinitely often, while $x$ is not. So $x$ is unfolded infinitely often, and as before, $x$ only gets unfolded right after the match passed a world where $p$ is true. Thus the path chosen by $\exists$ must contain infinitely many states where $p$ holds.

2.4 Bounded tree model property

Given the game-theoretic characterization of the semantics, it is rather straightforward to prove that formulas of the modal $\mu$-calculus are bisimulation invariant. From this it is immediate that the modal $\mu$-calculus has the tree model property. But in fact, we can use the game semantics to do better than this, proving that every satisfiable modal fixpoint formula is satisfied in a tree of which the branching degree is bounded by the size of the formula.

**Theorem 2.21 (Bisimulation Invariance)** Let $\xi$ be a modal fixpoint formula with $\text{FV}(\xi) \subseteq P$, and let $\mathbf{S}$ and $\mathbf{S}'$ be two labelled transition systems with points $s$ and $s'$, respectively. If $\mathbf{S}, s \triangleq_{\mathbf{P}} \mathbf{S}', s'$, then

$$\mathbf{S}, s \models \neg \xi \iff \mathbf{S}', s' \models \xi.$$  

**Proof.** Assume that $s \triangleq_{\mathbf{P}} s'$ and that $\mathbf{S}, s \models \neg \xi$, with $\text{FV}(\xi) \subseteq P$. We will show that $\mathbf{S}', s' \models \xi$. By Positional Determinacy we may assume that $\exists$ has a positional winning strategy $f$ in the evaluation game $\mathcal{E} := \mathcal{E}(\xi, \mathbf{S})$ initialized at $(\xi, s)$. We need to provide her with a winning strategy in the game $\mathcal{E}' := \mathcal{E}(\xi, \mathbf{S}')@((\xi, s'))$. She obtains her strategy $f'$ in $\mathcal{E}'$ from playing a shadow match of $\mathcal{E}$, using the bisimilarity relation to guide her choices.

To see how this works, let’s simply start with comparing the initial position $(\xi, s')$ of $\mathcal{E}'$ with its counterpart $(\xi, s)$ of $\mathcal{E}$. (Form now on we will write $s \triangleq s'$ instead of $s \triangleq_{\mathbf{P}} s'$).

In case $\xi$ is an atomic formula, then it is easy to see that both $(\xi, s)$ and $(\xi, s')$ are final positions. Also, since $f$ is assumed to be winning, $\xi$ must be true at $s$, and so it must hold at $s'$ as well. Hence, $\exists$ wins the match.
If $\xi$ is not atomic, we distinguish cases. First suppose that $\xi = \xi_1 \lor \xi_2$. If $f$ tells $\exists$ to choose disjunct $\xi_1$ at $(\xi, s)$, then she chooses the same disjunct $\xi_1$ at position $(\xi, s')$. If $\xi = \xi_1 \land \xi_2$, it is $\forall$ who moves. Suppose in $E'$ he chooses $\xi_i$, making $(\xi_i, s')$ the next position. We now consider in $E$ the same move of $\forall$, so that the next position in the shadow match is $(\xi_i, s)$.

A third possibility is that $\xi = \Diamond \psi$. In order to make her move at $(\xi, s')$, $\exists$ first looks at $(\xi, s)$. Since $f$ is a winning strategy, it indeed picks a successor $t$ of $s$. Then because $s \equiv s'$, there is a successor $t'$ of $s'$ such that $t \equiv t'$. This $t'$ is $\exists$’s move in $E$, so that $(\psi, t)$ and $(\psi, t')$ are the next positions in $E$ and $E'$, respectively.

Finally, if $\xi = \Box \psi$, we are dealing again with positions for $\forall$. Suppose in $E'$ he chooses the successor $t'$ of $s'$, so that the next position is $(\psi, t')$. (In case $s'$ has no successors, $\forall$ immediately loses, so that there is nothing left to prove.) Now again we turn to the shadow match; by bisimilarity of $s$ and $s'$ there is a successor $t$ of $s$ such that $t \equiv t'$. So we may assume that $\forall$ moves the game token of $E$ to position $(\psi, t)$.

The crucial observation is that if $\exists$ does not win immediately, then at least she can guarantee that the next positions in $E$ and $E'$ are of the form $(\varphi, u)$ and $(\varphi, u')$ respectively, with $u \equiv u'$, and such that the move in $E$ is consistent with $f$.

Continuing in this fashion, $\exists$ is able to maintain the condition that for any match

$$\beta' = (\varphi_0, s'_0)(\varphi_1, s'_1) \cdots (\varphi_n, s'_n)$$

played thus far, there is an $f$-guided shadow match

$$\beta = (\varphi_0, s_0)(\varphi_1, s_1) \cdots (\varphi_n, s_n)$$

in $E$, and such that $Z : s_i \equiv s'_i$ for all $i \leq n$. (A full proof of this would proceed by induction on $n$.)

It is not hard to see why this suffices to prove the theorem; for infinite matches, the key observation is that the two sequences of formulas, in the $E'$-match and its $E$-shadow, respectively, are exactly the same.

As an immediate corollary, we obtain the tree model property for the modal $\mu$-calculus.

**Theorem 2.22 (Tree Model Property)** Let $\xi$ be a modal fixpoint formula. If $\xi$ is satisfiable, then it is satisfiable at the root of a tree model.

**Proof.** For simplicity, we confine ourselves to the basic modal language. Suppose that $\xi$ is satisfiable at state $s$ of the Kripke model $S$. Then by bisimulation invariance, $\xi$ is satisfiable at the root of the unravelling $\tilde{S}_s$ of $S$ around $s$, cf. Definition 1.22. This unravelling clearly is a tree model.

QED

For the next theorem, recall that the size of a formula is simply defined as the number of its subformulas.

**Theorem 2.23 (Bounded Tree Model Property)** Let $\xi$ be a modal fixpoint formula. If $\xi$ is satisfiable, then it is satisfiable at the root of a tree, of which the branching degree is bounded by the size $|\xi|$ of the formula.
Proof. Suppose that \( \xi \) is satisfiable. By the Bisimulation Invariance Theorem it follows that \( \xi \) is satisfiable at the root \( r \) of some tree model \( T = (T, R, V) \). So \( \exists \) has a winning strategy \( f \) in the game \( E := E(\xi, T) \otimes (\xi, r) \). By the Positional Determinacy of evaluation game, we may assume that this strategy is positional — this will simplify our argument a bit. We may thus represent this strategy as a map \( f \) that, among other things, maps positions of the form \((s, \Diamond \varphi)\) to positions of the form \((t, \varphi)\) with \( Rst \).

We will prune the tree \( T \), keeping only the nodes that \( \exists \) needs in order to win the match. Formally, define subsets \( (T_n)_{n \in \omega} \) as follows:

\[
T_0 := \{r\},
\]
\[
T_{n+1} := T_n \cup \{s \mid (\varphi, s) = f(\Diamond \varphi, t) \text{ for some } t \in T_n \text{ and } \Diamond \varphi \sqsubseteq \xi\},
\]
\[
T_\omega := \bigcup_{n \in \omega} T_n.
\]

Let \( T_\omega \) be the subtree of \( T \) based on \( T_\omega \) (\( T_\omega \) is in general not a generated submodel of \( T \)). From the construction it is obvious that the branching degree of \( T_\omega \) is bounded by the size of \( \xi \), because \( \xi \) has at most \( |\xi| \) diamond subformulas.

We claim that \( T_\omega, r \Vdash g \xi \). To see why this is so, let \( E' := E(\xi, T_\omega) \) be the evaluation game played on the pruned tree. It suffices to show that the strategy \( f' \), defined as the restriction of \( f \) to positions of the game \( E' \), is winning for \( \exists \) in the game starting at \( (\xi, r) \). Consider an arbitrary \( E' \)-match \( \Sigma = (\xi, r)(\varphi_1, t_1) \ldots \) which is consistent with \( f' \). The key observation of the proof is that \( \Sigma \) is also a match of \( E \otimes (\xi, r) \), that is consistent with \( f \). To see this, simply observe that all moves of \( \forall \) in \( \Sigma \) could have been made in the game on \( T \) as well, whereas by construction, all \( f' \) moves of \( \exists \) in \( E' \) are \( f \) moves in \( E \).

Now by assumption, \( f \) is a winning strategy for \( \exists \) in \( E \), so she wins \( \Sigma \) in \( E \). But then \( \Sigma \) is winning as such, i.e., no matter whether we see it as a match in \( E \) or in \( E' \). In other words, \( \Sigma \) is also winning as an \( E' \)-match. And since \( \Sigma \) was an arbitrary \( E' \) match starting at \( (\xi, r) \), this shows that \( f' \) is a winning strategy, as required.

QED

Notes

The modal \( \mu \)-calculus was introduced by D. Kozen [13]. Its game-theoretical semantics goes back to at least Emerson & Jutla [9] (who use alternating automata as an intermediate step). As far as we are aware, the bisimulation invariance theorem, with the associated tree model property, is a folklore result. The bounded tree model property is due to Kozen & Parikh [15].

Exercises

Exercise 2.1 Express in words the meaning of the following \( \mu \)-calculus formula:

\[
\nu x. \mu y. (q \land \Box x) \lor (\neg p \lor \Box y).
\]

Exercise 2.2 (defining modal \( \mu \)-formulas) Give a modal \( \mu \)-formula \( \varphi(p, q) \) such that for all transition systems \( S \), and all states \( s_0 \) in \( S \):

\[
S, s_0 \Vdash \varphi(p, q) \quad \text{iff} \quad \text{there is a path } s_0 R s_1 \ldots R s_n (n \geq 0) \text{ such that } S, s_n \Vdash p \text{ and } S, s_i \Vdash q \text{ for all } i \text{ with } 0 \leq i < n.
\]
Exercise 2.3 (characterizing winning strategies)
A board is a structure $B = (B_0, B_1, E)$ such that $B_0 \cap B_1 = \emptyset$ and $E \subseteq B^2$, where $B = B_0 \cup B_1$ is a set of objects called positions. A match on $B$ consists of the players 0 and 1 moving a token from one position to another, following the edge relation $E$. Player $i$ is supposed to move the token when it is situated on a position in $B_i$.

Suppose in addition that $B$ is also partitioned into green and red positions, $B = G \uplus R$.

We will use a modal language to describe this structure, with the modalities being interpreted by the edge relation $E$, the proposition letter $p_0$ and $r$ referring to the positions belonging to player 0, and the red positions, respectively. That is, $V(p_0) = B_0$ and $V(r) = R$.

(a) Consider the game where player 0 wins as soon as the token reaches a green position. (That is, all infinite matches are won by player 1. Finite matches in which a player gets stuck are are won by his/her opponent, and in addition 0 wins finite matches ending in a green position.) Show that the formula $\mu x. \neg r \cdot (p_0 \leq 3 x) \cdot (\neg p_0 \leq 2 x)$ characterizes the winning positions for player 0 in this game, in the sense that for any position $b \in B$, we have

$B, V, b \models \varphi$ iff player 0 has a w.s. in the game starting at position $b$.

(b) Now consider the game where player 0 wins if she manages to reach a green position infinitely often. (More precisely, infinite matches are won by 0 iff a green position is reached infinitely often; finite matches are lost by a player is he/she gets stuck.) Give a formula $\varphi_b$ that characterizes the winning positions in this game.

Exercise 2.4 (characterizing fairness) Let $D = \{a, b\}$ be the set of atomic actions, and consider the following formula $\xi$, with subformulas as indicated:

$$\xi = \nu x. \nu y. \nu z. a_1 x \cdot (a_2 \downarrow \lor b y) \cdot a_3 z$$

Fix an LTS $S = (S, R_a, R_b, V)$. We say that the transition $a$ is enabled at state $s$ of $S$ if $S, s \models \diamond a \top$.

Show that $\xi$ expresses some kind of fairness condition, i.e., $S, s \models \xi$ iff there is no path starting at $s$ on which $a$ is enabled infinitely often, but executed only finitely often.

Exercise 2.5 We write $\models \models \psi$ to denote that $\psi$ is a local consequence of $\varphi$, that is, if for all pointed Kripke models $(S, s)$ it holds that $S, s \models \varphi$ implies $S, s \models \psi$.

(a) Show that $\mu x. \nu y. \alpha(x, y) \models \nu y. \mu x. \alpha(x, y)$, for all formulas $\alpha$.

(b) Show that $\mu x. \nu y. \alpha(x, y) \equiv \nu y. \mu x. \alpha(x, y)$, for all formulas $\alpha$.

(c) Show that $\mu x. (x \lor \gamma(x)) \land \delta(x) \models \mu x. \gamma(x) \land \delta(x)$, for all formulas $\gamma, \delta$.

Exercise 2.6 Show that the least and greatest fixpoint operators do not add expressive power to classical propositional logic, or, in other words, that the modality-free fragment of the modal $\mu$-calculus is expressively equivalent to classical propositional logic.