ENS Lyon - Winter 2020

Lecture Notes - SV

Ralph Sarkis

March 31, 2020

These are lecture notes taken during the *Semantics and Verification* classes taught by Colin Riba in winter 2020.

Contents

Question 1. *What is **Semantics and Verification**?*

While it is easy enough to describe how a simple program is executed, it is virtually impossible to infer the precise behavior of a machine running a large piece of code. Even more so if the latter contains randomness, parallelism or other complex features now available in most devices. The field of semantics is concerned with circumventing the useless details of the machine implementation by giving formal mathematical meaning to programs. This lets us reason rigorously about their execution or any interesting properties that they have.

Verification is a terminology for methods that, given a program in an abstract language and a property usually in another language, automatically verify whether the program satisfies the property. A common object that is used to describe programs is a **transition system**[1], and in this class we will be interested in the so-called **linear time properties**[2] that they have.

[1] Somewhat similar to a labeled graph. The nodes represent states of the program and each state has some properties associated to it.

[2] Roughly, they are properties on the infinite sequences of transitions that can occur in the system.

Transition Systems

Definition 2. A **transition system** T is a tuple $(S, A, \to, I, \mathrm{AP}, L)$, where:

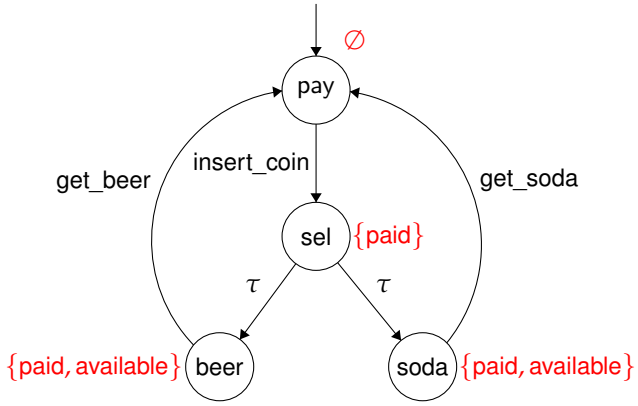
1. S is the set of **states** of the system (represented as nodes of the graph),
2. A is the set of **actions** (represented as labels for the edges of the graph),
3. $\to \subseteq S \times A \times S$ is the **transition relation**, we denote $s \xrightarrow{a} s'$ when $(s, a, s') \in \to$ (it translates to "When in state s, we can execute action a and end up in state s'"),
4. $I \subseteq S$ is the set of **initial states** (represented with an arrow pointing to them),
5. AP is the set of **atomic propositions**, and
6. $L : S \to 2^{AP}$ is the **state labeling**, telling which atomic propositions are true in each state of S (represented next to the states with a different color).

We give an illustrative example that is not so relevant, but it shows how we usually represent these types of machines.

Remark 3. While this definition might look similar to that of a DFA, there are a couple of important distinctions we can already make. First, this definition is highly non-deterministic as $\to$ is a relation and there might be several states related with the same transitions. Second, in general, a transition system is not necessarily finite.

Example 4 (Vending machine)**.** We will model a vending machine that waits for a user to pay by inserting a coin and then non-deterministically selects between giving beer or soda and waiting for another user to pay.[3]

[3] We will abbreviate the actions insert_coin, get_beer and get_soda respectively as ic, gb and gs.



The τ transitions are conventionally assumed to be non-deterministic from the point of view of the observer. The transition system depicted here has states $S = \{\mathsf{pay}, \mathsf{select}, \mathsf{soda}, \mathsf{beer}\}$, $A = \{\mathsf{ic}, \mathsf{gs}, \mathsf{gb}\}$[4], $I = \{\mathsf{pay}\}$, $\mathrm{AP} = \{\mathsf{paid}, \mathsf{available}\}$ and L assigns $\emptyset$ to pay, $\{\mathsf{paid}\}$ to sel and $\{\mathsf{paid}, \mathsf{available}\}$ to soda and beer.

[4] It is common practice to assume that τ is an action.

In the context of this course, and especially for this section, it is useful to have a way to generate transition systems. **Program graphs**, although they are designed to represent the evaluation of a program, can do exactly this.

Definition 5 (Program graph)**.** Given a (finite) set Vars of **variables** together with, for each variable $x \in \mathrm{Vars}$, a **domain** $\mathrm{Dom}(x)$[5], an **evaluation** is an element of $\mathrm{Eval}(\mathrm{Vars}) = \prod_{x \in \mathrm{Vars}} \mathrm{Dom}(x)$, that is, $\eta \in \mathrm{Eval}(\mathrm{Vars})$ assigns a value $\eta(x) \in \mathrm{Dom}(x)$ to each variable $x \in \mathrm{Vars}$.[6] We write Eval when the set of variables is clear from the context.

A **condition** is a propositional formula with atoms of the form $x \in D$ where $x \in \mathrm{Vars}$ and $D \subseteq \mathrm{Dom}(x)$ or $\top$ and $\bot$ to represent true and false values respectively. The set of such conditions denoted $\mathrm{Cond}(\mathrm{Vars})$ (or simply Cond) is of course closed under conjunctions, disjunctions and negations. Given a condition $g \in \mathrm{Cond}$ and a valuation $\eta \in \mathrm{Eval}$, we write $\eta \vDash g$ if g is true under the evaluation η.

A **program graph** over Vars has the form $PG = (\mathrm{Loc}, A, \mathrm{Effect}, \hookrightarrow, \mathrm{Loc}_0, g_0)$, where:

1. Loc is the (usually fininte) set of locations[7],
2. A is the set of actions,
3. $\mathrm{Effect} : A \times \mathrm{Eval} \to \mathrm{Eval}$ abstracts the effect that actions have on memory,
4. $\hookrightarrow \subseteq \mathrm{Loc} \times \mathrm{Cond} \times A \times \mathrm{Loc}$ which is a transition relation guarded by a condition[8],
5. $\mathrm{Loc}_0 \subseteq \mathrm{Loc}$ is the set of initial locations and $g_0 \in \mathrm{Cond}$ is the initial condition.

[5] Example of such domains are lists, machine integers, $\mathbb{Z}$, $\mathbb{R}$. Note that they can be infinite and even contain stuff that cannot be represented by a computer. This is because it is sometimes useful to abstract away these restrictions.

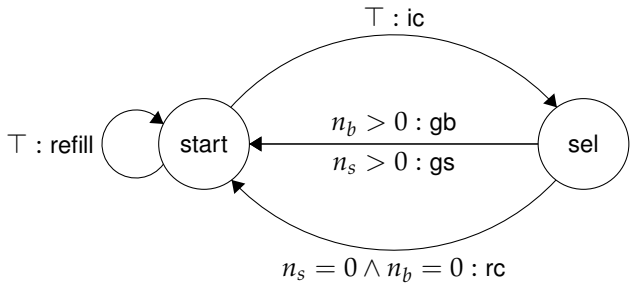
[6] In other words, an evaluation can be viewed as the state of the memory at a specific point in the program.

[7] They are an abstraction of the line number in the code, or of labels in assembly, another terminology is control point.

[8] We will denote $\ell \overset{g:a}{\hookrightarrow} \ell'$ when $(\ell, g, a, \ell') \in \hookrightarrow$. It roughly translates to "If we are in location ℓ and condition g is holds, then we can execute action a apply its effects and go to location ℓ'."

Example 6 (Vending machine (continued))**.** Let us extend Example **??** by giving the program graph for a vending machine with a similar behavior. The only difference is that there is a now a set amount of beers and sodas in the machine that can be refilled. When the user inserts a coin but there are no items left, the coin is returned.

Fix the maximum number of items $m \in \mathbb{N}$, let the amount of beers and sodas be variables in $n_b, n_s \in \text{Vars}$ with domain $\text{Dom}(n_b) = \text{Dom}(n_s) = \{0, \dots, max - 1\}$. There are two control points $\text{Loc}_0 = \mathsf{start}, \mathsf{sel} \in \text{Loc}$ and new actions to refill and return the coin ($A = \{\mathsf{ic}, \mathsf{gb}, \mathsf{gs}, \mathsf{refill}, \mathsf{rc}\}$). The initial condition is $g_0 = n_b = m - 1 \wedge n_s = m - 1$.



The effects are not represented in the diagram but a sensible Effect would satisfy: for any evaluation $\eta \in \text{Eval}$,

$$\begin{aligned} \text{Effect}(\eta, \mathsf{ic}) = \text{Effect}(\eta, \mathsf{rc}) &= \eta \\ \text{Effect}(\eta, \mathsf{gb}) &= \eta[n_b := n_b - 1] \\ \text{Effect}(\eta, \mathsf{gs}) &= \eta[n_s := n_s - 1] \\ \text{Effect}(\eta, \mathsf{refill}) &= \eta[n_b := 100, n_s := 100]. \end{aligned}$$

The crucial difference between program graphs and transition systems is that the former separate the control from the data. In other words, a program graph abstracts only the behavior the program while a transition system abstracts the behavior along with the memory of the program. This motivates that a transition system might be more appropriate for observing the evolution of a program graph along with the evaluation. The following definition makes this formal.

Definition 7 (TS of a PG)**.** Let us have a program graph PG with the same notation as in Definition **??**, the transition system of PG is $TS(PG) = (\text{Loc} \times \text{Eval}, A, \to, I, \text{AP}, L)$[9], where:

1. $\to$ is defined by the rule

$$\frac{\ell \overset{g:\alpha}{\hookrightarrow} \ell' \qquad n \vDash g}{(\ell, \eta) \overset{\alpha}{\to} (\ell', \text{Effect}(\eta, \alpha))},$$

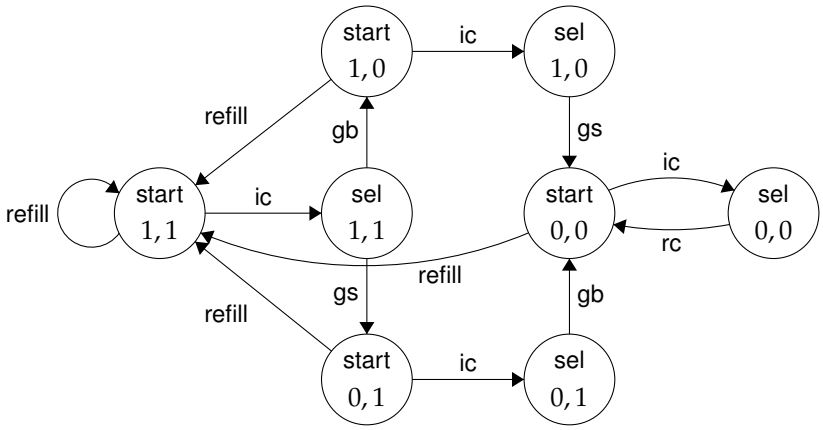
2. $I = \{(\ell, \eta) \mid \ell \in Loc_0, \eta \vDash g_0\}$,[10]
3. $\text{AP} = \text{Loc} + \text{Cond}(\text{Vars})$,[11]
4. $L(\ell, \eta) = \{\ell\} \cup \{g \mid \eta \vDash g\}$.

[9] Note that this can lead to a huge set of states because some variables can have huge domains.

[10] We require that the initial condition is satisfied with η in memory.

[11] In words, the atomic properties can say whether a state is in a certain location or whether it satisfies a condition of Cond. In practice, we use a smaller subset of AP that contains only properties relevant to the particular application.

Example 8 (Vending machine (still))**.** Assuming that the $m = 2$, we can draw the transition graph for the vending machine in Example **??** (we omit the state labeling as it is clear what properties hold at each state). We denote the evaluations as a pairs of values (n_s, n_b).



Notice that increasing m, even by only one, would make the transition graph way more complex.

To end this section presenting the basics of transition systems, we describe three ways of combining them.

The first one is similar to taking the product of two DFA[12], but we allow the case where some system can do an action that the other cannot. [13]

[12] Deterministic finite automata.

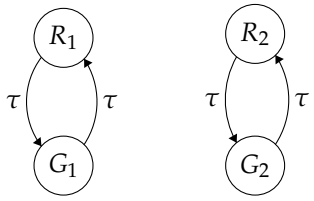
[13] In DFA terminology, it amounts to taking the products of two automata on different alphabets. When the new machine sees a letter that only one of the original DFA recognizes, it makes a transition only according to that DFA. When it sees a letter that both DFA recognize, it non-deterministically choose what transition to make. There is a slight caveat because actions are not consumed by a transition system, so after doing a common action on one of the system, the new machine can still do that action on the other system.

Definition 9 (Interleaving of TSs)**.** Given two transition systems $T_i = (S_i, A_i, \to_i, I_i, \mathrm{AP}_i, L_i)$ for $i = 1, 2$, their interleaving composition denoted $T_1 \;|||\; T_2$ is $(S_1 \times S_2, A_1 \cup A_2, \to, I_1 \times I_2, \mathrm{AP}_1 \cup \mathrm{AP}_2, L)$, where $\to$ is defined by

$$\frac{s_1 \xrightarrow{\alpha}_1 s_1'}{(s_1, s_2) \xrightarrow{\alpha} (s_1', s_2)} \qquad \frac{s_2 \xrightarrow{\alpha}_2 s_2'}{(s_1, s_2) \xrightarrow{\alpha} (s_1, s_2')},$$

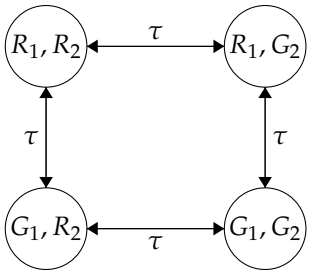
and $L(s_1, s_2) = L_1(s_1) \cup L_2(s_2)$.

Example 10 (Traffic lights)**.** Suppose we have two traffic lights with that can switch from red to green and vice-versa non-deterministically, they are represented by the following graphs.



Taking their interleaving yields the following transition system.[14]

[14] We left out the state labeling as it is a trivial construction.



Unfortunately, such a simple way of composing transition systems does not allow shared memory between the systems. For this reason, when we want to take this possibility into account, it is preferred to do a composition of program graphs.

Definition 11 (Interleaving of PGs)**.** Given two program graphs $G_i = (\mathrm{Loc}_i, A_i, \mathrm{Effect}_i, \hookrightarrow_i, \mathrm{Loc}_{i,0}, g_{i,0})$ over Vars_i for $i = 1, 2$, their interleaving is the program graph[15] over $\mathrm{Vars}_1 \cup \mathrm{Vars}_2$ denoted $G_1 \,|||\, G_2 = (\mathrm{Loc}_1 \times \mathrm{Loc}_2, A_1 + A_2, \mathrm{Effect}, \hookrightarrow, \mathrm{Loc}_{1,0} \times Loc_{2,0}, g_{1,0} \wedge g_{2,0})$, where $\hookrightarrow$ is defined by the rule

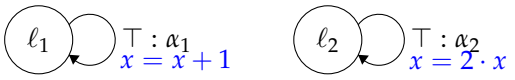
$$\frac{\ell_1 \overset{g:\alpha}{\hookrightarrow}_1 \ell_1'}{(\ell_1, \ell_2) \overset{g:\alpha}{\hookrightarrow} (\ell_1', \ell_2)} \qquad \frac{\ell_2 \overset{g:\alpha}{\hookrightarrow}_2 \ell_2'}{(\ell_1, \ell_2) \overset{\alpha}{\hookrightarrow} (\ell_1, \ell_2')},$$

and $\mathrm{Effect}(\alpha, \eta) = \mathrm{Effect}_i(\alpha, \eta)$ for $\alpha \in A_i$.[16]

[15] Observe that the variables are not necessarily disjoint, hence we must consider actions of A_1 and A_2 as disjoint, otherwise there would be an ambiguity in the choice of what effect to apply.

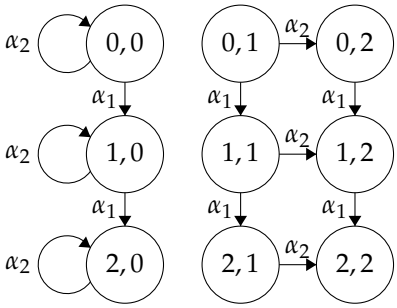
[16] The evaluation η is an element of $\mathrm{Eval}(\mathrm{Vars}_1 \cup \mathrm{Vars}_2)$, so we implicitly adapted Effect_i in the obvious way (i.e.: it does not modify variables outside Vars_i).

Example 12. Let us illustrate this construction on two simple program graphs manipulating the same variable x with $\mathrm{Dom}(x) = \mathbb{N}$, here are their representations with effects in blue.



Interleaving the program graphs is simple enough (see Figure **??**), but what is more interesting is comparing the transition systems we obtain when do the operations $TS(G_1) \,|||\, TS(G_2)$ and $TS(G_1 \,|||\, G_2)$. Since the domain of x is infinite, both these transition systems have infinitely many states.

For the former, observe that interleaving $TS(G_1)$ and $TS(G_2)$ (represented in Figure **??**) will lead to the dissociation of the variable x into two independent copies. It leads to a system (partially represented below) which is irrelevant for the purpose of analyzing the behavior of both programs when run concurrently.



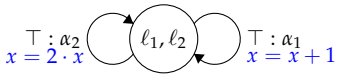


Figure 1: Interleaving of program graphs in Example **??**.

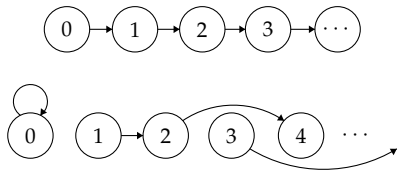
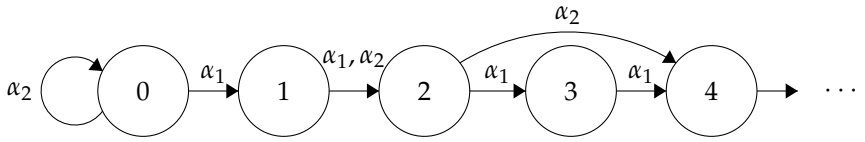


Figure 2: Part of $TS(G_1)$ and $TS(G_2)$ from Example **??**. (the label of the nodes is the value of x at that state)

For the latter, we actually obtain an interesting system (depicted below) because interleaving the program graphs first ensures that α_1 and α_2 act on the same x.



This example shows the relevance of program graphs when we care about concurrent data. While there are many more possibilities to compose transition systems, we introduce one last definition that illustrates how we can deal with concurrent control without using program graphs.

Definition 13 (Parallel Composition of TSs)**.** Let $T_i = (S_i, A_i, \to_i, I_i, \mathrm{AP}_i, L_i)$ for $i = 1, 2$ be two transition systems and $H \subseteq A_1 \cap A_2$,[17] their **parallel composition** (or **handshaking**) is $T_1 \parallel_H T_2 = (S_1 \times S_2, A_1 \cup A_2, \to, I_1 \times I_2, \mathrm{AP}_1 \cup \mathrm{AP}_2, L)$, where $\to$ is defined by the rules

$$\frac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad \alpha \notin H}{(s_1, s_2) \xrightarrow{\alpha} (s_1', s_2)} \qquad \frac{s_2 \xrightarrow{\alpha}_2 s_2' \qquad \alpha \notin H}{(s_1, s_2) \xrightarrow{\alpha} (s_1, s_2')}$$

$$\frac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad s_2 \xrightarrow{\alpha}_2 s_2' \qquad \alpha \in H}{(s_1, s_2) \xrightarrow{\alpha} (s_1', s_2')},$$

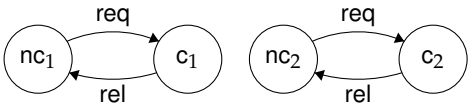
and $L(s_1, s_2) = L_1(s_1) \cup L_2(s_2)$. One should view the actions in H as *synchronized* actions that both systems have to do at the same time.[18]

[17] In general, we suppose $\tau \notin H$ and we write $T_1 \parallel T_2$ when $H = (A_1 \cap A_2) \setminus \{\tau\}$.

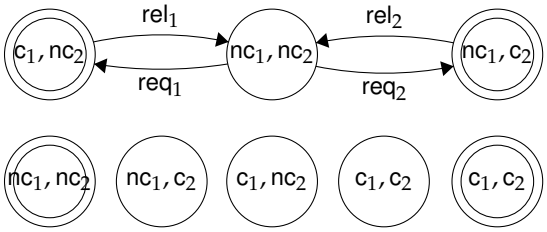
[18] Note that this definition is a generalization of the interleaving composition as $T_1 \parallel_\varnothing T_2 = T_1 \mid\mid\mid T_2$.

Example 14. Given two transition systems T_1 and T_2 that have a non-critical state denoted nc_i and a critical state c_i and can jump from one to the other using actions req and rel as depicted below.[19]

[19]



We leave it as an exercise to show that in the plain interleaving composition of T_1 and T_2, both processes can reach their critical state at the same time. However, if we add an arbiter system A that can unlock or lock the critical section, we can disallow this behavior: with A is as in Figure **??**, $A \parallel_{\mathsf{rel},\mathsf{req}} (T_1 \mid\mid\mid T_2)$ is as follows.



We have constructed a system where (c_1, c_2) cannot be reached. This kind of property is called a **safety property** and it is **finitary** because it does not mention

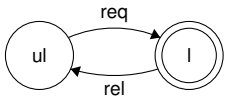


Figure 3: Arbiter system in Example **??**.

an infinite execution of the program. In the next section, we will talk about **linear time properties** which are infinitary. In the context of this example, a reasonable linear time property could require that in any execution, both c_1 and c_2 are visited infinitely many times, ensuring fairness of the arbiter.

Before leaving this section, we show a simple result that illustrates how to deal with transition systems in a more theoretical fashion.

Proposition 15 (Associativity of $\|$)**.** *Let $T_i = (S_i, A_i, \to_i, I_i, AP_i, L_i)$ for $i = 1, 2, 3$ be transition systems, then*[20]

$$T := (T_1 \parallel T_2) \parallel T_3 = T_1 \parallel (T_2 \parallel T_3) =: T'.$$

[20] Recall that $\|$ with no subscript is the parallel composition synchronizing all common actions (except τ).

Proof. It is easy to see that the states, actions, initial states, atomic propositions and state labelings of T and T' will be the same because they are constructed with $\times$ and $\cup$ which are associative operations. Let us denote $\to$ and $\Rightarrow$ for the transition relations of T and T' respectively. We have to show that for any $s_1, s_1' \in S_1$, $s_2, s_2' \in S_2$, $s_3, s_3' \in S_3$ and $\alpha \in A_1 \cup A_2 \cup A_3$,

$$(s_1, s_2, s_3) \xrightarrow{\alpha} (s_1', s_2', s_3') \Leftrightarrow (s_1, s_2, s_3) \overset{\alpha}{\Rightarrow} (s_1', s_2', s_3').$$

We proceed by case analysis on the nature of α.

Case 1: The action α belongs to exactly one of the systems, say $\alpha \in A_1$,[21] then we have the following inferences:

[21] The other cases are similar.

$$\dfrac{\dfrac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad \alpha \notin A_1 \cap A_2}{(s_1, s_2) \xrightarrow{\alpha}_{1\|2} (s_1', s_2)} \qquad \alpha \notin (A_1 \cup A_2) \cap A_3}{(s_1, s_2, s_3) \xrightarrow{\alpha} (s_1', s_2, s_3)}$$

$$\dfrac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad \alpha \notin A_1 \cap (A_2 \cup A_3)}{(s_1, s_2, s_3) \overset{\alpha}{\Rightarrow} (s_1', s_2, s_3)}$$

Case 2: The action α belongs to exactly two of the systems, say $\alpha \in A_1 \cap A_3$, then we have the following inferences:

$$\dfrac{\dfrac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad \alpha \notin A_1 \cap A_2}{(s_1, s_2) \xrightarrow{\alpha}_{1\|2} (s_1', s_2)} \qquad s_3 \xrightarrow{\alpha}_3 s_3' \qquad \alpha \in (A_1 \cup A_2) \cap A_3}{(s_1, s_2, s_3) \xrightarrow{\alpha} (s_1', s_2, s_3')}$$

$$\dfrac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad \dfrac{s_3 \xrightarrow{\alpha}_3 s_3' \qquad \alpha \notin A_2 \cap A_3}{(s_2, s_3) \xrightarrow{\alpha}_{2\|3} (s_2, s_3')} \qquad \alpha \in A_1 \cap (A_2 \cup A_3)}{(s_1, s_2, s_3) \overset{\alpha}{\Rightarrow} (s_1', s_2, s_3')}$$

Case 3: The action α belongs to all of the systems, then we have the following inferences:

$$\frac{\dfrac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad s_2 \xrightarrow{\alpha}_2 s_2' \qquad \alpha \in A_1 \cap A_2}{(s_1, s_2) \xrightarrow{\alpha}_{1\|2} (s_1', s_2')} \qquad s_3 \xrightarrow{\alpha}_3 s_3' \qquad \alpha \in (A_1 \cup A_2) \cap A_3}{(s_1, s_2, s_3) \xrightarrow{\alpha} (s_1', s_2', s_3')}$$

$$\frac{s_1 \xrightarrow{\alpha}_1 s_1' \qquad \dfrac{s_2 \xrightarrow{\alpha}_2 s_2' \qquad s_3 \xrightarrow{\alpha}_3 s_3' \qquad \alpha \in A_2 \cap A_3}{(s_2, s_3) \xrightarrow{\alpha}_{2\|3} (s_2', s_3')} \qquad \alpha \in A_1 \cap (A_2 \cup A_3)}{(s_1, s_2, s_3) \xRightarrow{\alpha} (s_1', s_2', s_3')}$$

□

Linear Time Properties

Definition 16 (Linear Time Property)**.** A **linear time property** (LTP) over atomic propositions AP is a set of ω-words $P \subseteq (2^{\text{AP}})^\omega$.[22]

Example 17. Recall the transition system T_{VM} depicted in Example **??** (and shown again in Figure **??** with AP $= \{\mathsf{paid}, \mathsf{available}\}$. Here are four examples of LTP in this context.

First, the property that any state with an available drink is preceded by a state where the user has paid can be written formally as[23]

$$P_1 = \{\sigma \in (2^{\text{AP}})^\omega : \forall i, \mathsf{available} \in \sigma(i) \implies i > 0 \wedge \exists j < i, \mathsf{paid} \in \sigma(j)\}.$$

Intuitively, the system seems to behave according to P_1, so one might expect that T_{VM} "satisfies"this property in some sense. Definition **??** will formalize this intuition.

The property that the number of states where a user has paid is at least as large as the number of states where a drink is available is written:

$$P_2 = \left\{\sigma \in (2^{\text{AP}})^\omega : |\{i \mid \mathsf{available} \in \sigma(i)\}| \leq |\{i \mid \mathsf{paid} \in \sigma(i)\}|\right\}.$$

In general, LTPs similar to P_1 and P_2 are hard to work with because they are not finitary in the sense that, to verify them, one has no choice but to look at an infinite amount of symbols.

We will see that some properties which might look infinitary are easier to automatically verify because they have a finite representation. For instance, the property that there is an infinite number of states is written:[24]

$$P_3 = \{\sigma \in (2^{\text{AP}})^\omega \mid \exists^\infty i, \mathsf{paid} \in \sigma(i)\}.$$

This last LTP illustrates how $\forall^\infty$ can be used:

$$P_4 = \left\{\sigma \in (2^{\text{AP}})^\omega \mid [\forall^\infty i, \mathsf{paid} \in \sigma(i)] \implies [\exists^\infty i, \mathsf{available} \in \sigma(i)]\right\}.$$

Definition 18 (Path)**.** A (finite or infinite) **path** in a transition system $T = (S, A, \to, I, \text{AP}, L)$ is a (finite or infinite) sequence of states $\pi = (s_i)_{i \leq n} \subseteq S$ with $n \leq \omega$ and such that $\forall i, i+1 < n \implies \exists \alpha \in A, s_i \xrightarrow{\alpha} s_{i+1}$. We say that a path π is **initial** if $s_0 \in I$.

[22] We use ω as the cardinality of $\mathbb{N}$, thus an ω-word on an alphabet Σ is an element of Σ^ω, i.e.: an infinite sequence of symbols in Σ. Although some of the results about LTPs can be shown with general alphabets, we will remain in the case of $\Sigma = 2^{\text{AP}}$ for clarity.

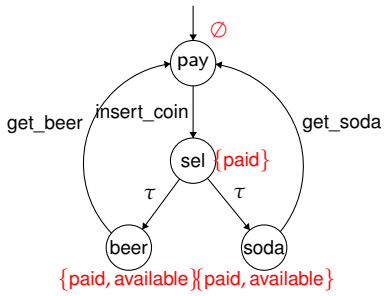


Figure 4: Representation of T_{VM}

[23] Note that indices used for ω-words (like i and j for P_1) are quantified over all ω unless stated otherwise.

[24] The notation $\exists^\infty$ is a shorthand of $\forall N, \exists i \geq N$. Its less intuitive dual, "always true after some point", is denoted $\forall^\infty := \exists N, \forall i \geq N$.

Definition 19 (Trace). The **trace** of a path $\pi = (s_i)_{i<n}$ is the sequence $L(\pi) := (L(s_i))_{i<n}$. The **set of traces** of a transition system T, denoted $\mathrm{Tr}(T)$, is

$$\mathrm{Tr}(T) = \{L(\pi) \mid \pi \text{ is an initial path in } T\}.$$

Also, $\mathrm{Tr}^\omega(T)$ denotes the set of infinite traces and $\mathrm{Tr}_{\mathrm{fin}}(T)$ the set of finite traces.

Definition 20. We say that a transition system T **satisfies a linear time property** P if $\mathrm{Tr}^\omega(T) \subseteq P$. We denote this by write $T \approx P$.[25]

[25] We use this notation instead of the more usual $\models$ because this definition is not the perfect notion of satisfaction. Informally, this comes from the fact branchings are a feature internal to transition systems but not to LTPs. When we cover modal logics, we will see how to fix this definition.

Example 21. Let us show that all the properties in Example **??** are satisfied by T_{VM}.

1. Clearly, $T_{\mathrm{VM}} \approx P_1$ because any path in T goes through sel before going through either beer or soda.
2. Since for any i such that available $\in L(\pi_i)$, we also have paid $\in L(\pi)$ it follows trivially that $T_{\mathrm{VM}} \approx P_2$.
3. The structure of T_{VM} is very simple and we can observe that for any π and any $N \in \mathbb{N}$, paid $\in L(\pi_{N+2})$,[26] thus $T_{\mathrm{VM}} \approx P_3$.
4. Note that for any infinite path in T_{VM} goes infinitely many times through pay, thus it is not possible that at some point, any state in the path has paid in its labeling. We conclude that $T_{\mathrm{VM}} \approx P_4$.

[26] In words, starting in any state, doing two transition always leads to a state where the user has paid.

Proposition 22. *Let T and T' be two transition systems over AP, then*

$$\mathrm{Tr}^\omega(T) \subseteq \mathrm{Tr}^\omega(T') \Leftrightarrow \left[\forall P \subseteq (2^{AP})^\omega, (T' \approx P \implies T \approx P)\right].$$

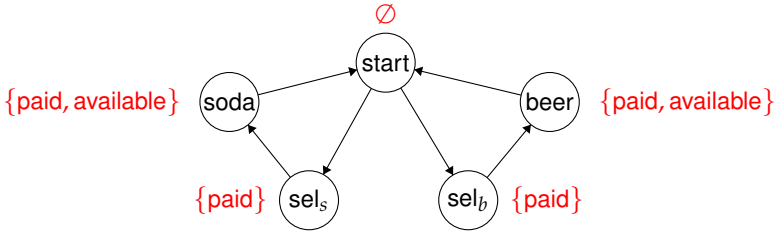
Proof. ($\Rightarrow$) Follows trivially from the definitions. Indeed, for any $P \subseteq (2^{\mathrm{AP}})^\omega$,

$$T' \approx P \overset{\text{def}}{\Leftrightarrow} \mathrm{Tr}^\omega(T') \subseteq P \overset{\text{hyp}}{\implies} \mathrm{Tr}^\omega(T) \subseteq P \overset{\text{def}}{\Leftrightarrow} T \approx P.$$

($\Leftarrow$) Consider $P = \mathrm{Tr}^\omega(T')$. We know that $T' \approx P$, so, by our hypothesis, $T \approx P$, that is $\mathrm{Tr}^\omega(T) \subseteq \mathrm{Tr}^\omega(T')$.[27] □

[27] Although this proof is quite trivial, it illustrates the importance of the fact that infinite traces of a transition system form an LTP.

Example 23. Let T'_{VM} be the transition depicted below (we omit the actions as they are irrelevant for this example).



The traces of T'_{VM} are the same as the traces of T_{VM}. In particular, we have $\mathrm{Tr}^\omega(T_{\mathrm{VM}}) = \mathrm{Tr}^\omega(T'_{\mathrm{VM}})$, so Proposition **??** says either system satisfies LTPs that the other satisfies.

We have already mentioned that some LTPs are harder to verify than others, now we will introduce different families of linear time properties are nicer than most. There are many such families, but we chose three which are simple to define and have both historical and theoretical importance.[28]

[28]

Invariants and Safety Properties

Definition 24 (Invariant)**.** A linear time property $P \subseteq (2^{\mathrm{AP}})^\omega$ is an **invariant** if there exists a propositional formula ϕ over AP such that $P = \{\sigma \mid \forall i, \sigma(i) \vDash \phi\}$.[29]

[29] In words, all the paths of a system satisfying P goes through states that satisfy some property ϕ.

Example 25. Recall Example **??** where we ensured mutual exclusion. In the last system $T = A \,||_{\mathsf{rel,req}}\, (T_1 \,|||\, T_2)$, the states where both T_1 and T_2 are in their critical sections are not reachable. Thus, a formula of the form $\phi = \neg(\mathsf{c}_1 \wedge \mathsf{c}_2)$ is satisfied at any state in a path in T. The property of mutual exclusion is thus an invariant.

Definition 26 (Safety Property)**.** A linear time property $P \subseteq (2^{\mathrm{AP}})^\omega$ is a **safety property** if there is a set of finite words $P_{\mathrm{bad}} \subseteq (2^{\mathrm{AP}})^*$ such that[30]

$$P = \left\{\sigma \in (2^{\mathrm{AP}})^\omega \mid \forall i \in \mathbb{N}, \sigma(0)\cdots\sigma(i) \notin P_{\mathrm{bad}}\right\}.$$

[30] Intuitively, a safety property is one that always fails in finite time. That is, if $L(\pi) \notin P$, then there exists a finite point in the execution of π where we can decide that the trace of π is not in P.

Example 27. In Example **??**, P_1 and P_2 for T_{VM} are safety properties. For P_1, we know that σ fails to be in P when available appears before paid, thus we can write[31]

$$P_{1,\mathrm{bad}} = \emptyset^* \cdot (\{\mathsf{available}\} + \{\mathsf{paid}, \mathsf{available}\})\,.$$

[31] We use the regular expression notation, where a^* means any finite sequence of a, $\mathsf{a}\cdot\mathsf{b}$ means a followed by b and $\mathsf{a}+\mathsf{b}$ means an a or a b.

Before getting dirty with these family of properties, we show two very simple statements.

Proposition 28. *An LTP P is a safety property if and only if for any $\sigma \in P^c$,[32] there exists $i \in \mathbb{N}$ such that $\sigma(0)\cdots\sigma(i)\cdot(2^{AP})^\omega \cap P = \emptyset$.*

[32] The complement of an LTP P on AP is $P^c := (2^{\mathrm{AP}})^\omega \setminus P$.

Proof. ($\Rightarrow$) Since P is a safety property, it is induced by some P_{bad}. If $\sigma \in P^c$, then we infer from the definition that there exists $i \in \mathbb{N}, \sigma(0)\cdots\sigma(i) \in P_{\mathrm{bad}}$. Now, any word in $\sigma(0)\cdots\sigma(i)\cdot(2^{\mathrm{AP}})^\omega$ has a finite prefix in P_{bad}, namely $\sigma(0)\cdots\sigma(i)$, so it cannot be in P. This direction follows.

($\Leftarrow$) Using the axiom of choice, for any $\sigma \in P^c$, we can choose i_σ such that $\sigma(0)\cdots\sigma(i_\sigma)\cdot(2^{\mathrm{AP}})^\omega \cap P = \emptyset$. Thus, if we let $P_{\mathrm{bad}} = \{\sigma(0)\cdots\sigma(i_\sigma) \mid \sigma \in P^c\}$, we can easily see that no word in P has a finite prefix in P_{bad}[33] and any word in P^c has a finite prefix in P_{bad}. Therefore, P is the safety property induced by P_{bad}. □

[33] If for $i \in \mathbb{N}$ and $\sigma \in P$, $\sigma(0)\cdots\sigma(i) \in P_{\mathrm{bad}}$, it contradicts the fact that $\sigma(0)\cdots\sigma(i_\sigma)\cdot(2^{\mathrm{AP}})^\omega \cap P = \emptyset$.

Proposition 29. *Any invariant LTP is a safety property.*

Proof. It suffices to let P_{bad} be the set of finite words with one character not satisfying the invariant. We leave the details as an exercise. □

Definition 30. Given $\sigma \in (2^{\mathrm{AP}})^\omega$, a **finite prefix** of σ is $\hat{\sigma} \in (2^{\mathrm{AP}})^*$ such that $\hat{\sigma} = \sigma(0)\ldots\sigma(n)$ for $n \in \mathbb{N}$. We write $\subseteq$ for the relation "is a finite prefix of".

Definition 31. A state of a transition system is **terminal** if it has no outward transition.[34]

[34] Formally, $s \in S$ is terminal if for any $\alpha \in A$ and any $s' \in s$, $s \overset{\alpha}{\not\to} s'$.

Proposition 32. *Let T be a transition system with no terminal states and $P \subseteq (2^{AP})^\omega$ be a safety property induced by P_{bad}, then*

$$T \approx P \Leftrightarrow \mathrm{Tr}_{fin}(T) \cap P_{bad} = \emptyset.$$

Proof. ($\Leftarrow$) Let $L(\pi)$ be an infinite trace of T. Since any of its finite prefix is in $\mathrm{Tr}_{\mathrm{fin}}(T)$, it cannot coincide with a word in P_{bad}. Hence, $L(\pi) \in P$.

($\Rightarrow$) Suppose there exists $\hat{\sigma} \in \mathrm{Tr}_{\mathrm{fin}}(T) \cap P_{\mathrm{bad}}$, we have $\hat{\sigma} = L(\pi)$ for a finite path π, but since there are no terminal state, we can always add states to π and obtain an infinite path π' such that $\hat{\sigma} \subseteq L(\pi')$. This means $L(\pi') \notin P$, but it contradicts our assumption that $T \approx P$. □

Corollary 33. [35] *Let T and T' be transition systems over AP with no terminal states, then*

$$\mathrm{Tr}_{fin}(T) \subseteq \mathrm{Tr}_{fin}(T') \Leftrightarrow \left[\forall \text{ safety } P \subseteq (2^{AP})^{\omega}, (T' \approx P \implies T \approx P)\right].$$

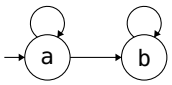
[35] This result is essentially a characterization similar to Proposition **??** that applies to safety properties. But now, instead of comparing all the traces, we only have to compare the finite traces.

Proof. ($\Rightarrow$) Follows trivially from the last proposition.

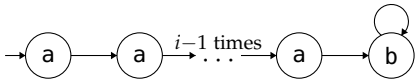
($\Leftarrow$) Consider the safety property induced by $P_{\mathrm{bad}} = (2^{\mathrm{AP}})^* \setminus \mathrm{Tr}_{\mathrm{fin}}(T')$. It is clear that $T' \approx P$ by the last proposition, thus our assumption gives us $T \approx P$ or equivalently $\mathrm{Tr}_{\mathrm{fin}}(T) \cap (2^{\mathrm{AP}})^* \setminus \mathrm{Tr}_{\mathrm{fin}}(T') = \emptyset$, it follows that $\mathrm{Tr}_{\mathrm{fin}}(T) \subseteq \mathrm{Tr}_{\mathrm{fin}}(T')$. □

Since, finite traces can be arbitrarily large, it is natural to ask whether comparing finite traces of two systems suffices to compare all the LTPs that they satisfy. This is almost the right intuition, but as usual, infinity breaks our intuition as shown in the following example.

Example 34. Let T be the transition system depicted below where the state labeling is written inside the states and states' and actions' names are omitted.



We have $\mathrm{Tr}_{\mathrm{fin}}(T) = \mathsf{a}^*\mathsf{b}^*$ and $\mathrm{Tr}^{\omega}(T) = \mathsf{a}^*\mathsf{b}^{\omega} + \mathsf{a}^{\omega}$. Now consider for any $i \in \mathbb{N}$, the system T_i as depicted below (with the same conventions as for T).



The finite traces of T_i will be words recognized by $\mathsf{a} + \cdots + \mathsf{a}^i \cup \mathsf{a}^i\mathsf{b}^*$ and its infinite traces will be recognized by $\mathsf{a}^i\mathsf{b}^{\omega}$.

Now, let T' be the union[36] for $i \in \mathbb{N}$ of the T_i's, then we have $\mathrm{Tr}_{\mathrm{fin}}(T') = \mathsf{a}^*\mathsf{b}^* = \mathrm{Tr}_{\mathrm{fin}}(T)$, but $\mathrm{Tr}^{\omega}(T') = \mathsf{a}^*\mathsf{b}^{\omega} \neq \mathrm{Tr}^{\omega}(T)$.

[36] Informally, it is like putting all systems next to each other with no interaction between them. All initial states of the T_i's are still initial states, so the paths in T' are just the union of the paths in the T_i's.

The following definition introduces a sufficient condition to get rid of such counterexamples. As expected, it is a finiteness property.

Definition 35 (Finitely Branching)**.** A transition system T is said to be **finitely branching** if I is finite and for any $s \in S$, $\{s' \in S \mid \exists \alpha \in A, s \xrightarrow{\alpha} s'\}$ is finite.[37]

[37] In later parts of the course, we will study logics that will care about what actions are used. In these cases, finitely branching will require that there is a finite number of distinct actions that can be done at s.

Proposition 36. *Two finitely branching T and T' with no terminal states and on AP satisfy*

$$\mathrm{Tr}^{\omega}(T) \subseteq \mathrm{Tr}^{\omega}(T') \Leftrightarrow \mathrm{Tr}_{fin}(T) \subseteq \mathrm{Tr}_{fin}(T').$$

Proof. ($\Rightarrow$) Since the systems have no terminal states, any finite trace in T corresponds to a path π in T that can be extended to an infinite path π' so that $L(\pi')$ is in $\mathrm{Tr}^\omega(T)$ and thus in $\mathrm{Tr}^\omega(T')$. Now, $L(\pi')$ must correspond to a path π'' in T' and truncating it to the size of π shows that $L(\pi) = L(\pi''|_{i\leq|\pi|})$ is also in $\mathrm{Tr}_{\mathrm{fin}}(T')$.

($\Leftarrow$)[38] Let $\sigma \in \mathrm{Tr}^\omega(T)$, for any $n \in \mathbb{N}$, $\sigma_n := \sigma(0)\cdots\sigma(n) \subseteq \sigma$ is in $\mathrm{Tr}_{\mathrm{fin}}(T) \subseteq \mathrm{Tr}_{\mathrm{fin}}(T')$, so in particular, it is the finite trace of an initial path, say π_n, in T'.

To construct an initial path π in T' that satisfies $L(\pi) = \sigma$, we will build $(s_i)_{i\in\mathbb{N}}$ by induction on i with $s_0 \in I$ and the following invariant: There are infinitely many π_n's such that $\forall k \leq i, \pi_n(k) = s_k$.

First, since I' is finite and all paths π_n satisfy $\pi_n(0) \in I'$, there is at least one $s_0 \in I'$ such that there are infinitely many π_n with $\pi_n(0) = s_0$.

Second, suppose s is defined up to $i-1$ and there are infinitely many π_n's satisfying $P_{i-1} := \forall k \leq i-1, \pi_n(k) = s_k$. Then, since there are finitely many $s \in S'$ such that $s_{i-1} \xrightarrow{\alpha} s$ for some $\alpha \in A'$, we can pick one such s_i such that there are still infinitely many of the π_n satisfying P_{i-1} that satisfy $P_i := \forall k \leq i, \pi_n(k) = s_k$.

By the induction principle, this defines a path $\pi = (s_i)_{i\in\mathbb{N}}$ that

1. is initial because $s_0 \in I'$,
2. is in T' because every finite subpath is in T', and
3. satisfies $L(\pi) = \sigma$ because $L(\pi_n) = \sigma_n$ for all π_n.

We conclude that $\sigma \in \mathrm{Tr}^\omega(T')$. □

[38] In class, this direction was proved as a corollary of the more general König's lemma which states that any finitely branching infinite tree has an infinite path. I chose to integrate the proof of the lemma into the proof of the proposition to avoid introducing more definitions than needed.

Corollary 37. *Two transition systems on AP with no terminal states satisfy the same LTPs if and only if they satisfy the same safety properties.*

Proof. The proof follows from these equivalences that use the previous results:

$$\begin{aligned}\left[\forall P \in (2^{\mathrm{AP}})^\omega, T \approx P \Leftrightarrow T' \approx P\right] &\Leftrightarrow \mathrm{Tr}^\omega(T) = \mathrm{Tr}^\omega(T') \\ &\Leftrightarrow \mathrm{Tr}_{\mathrm{fin}}(T) = \mathrm{Tr}_{\mathrm{fin}}(T') \\ &\Leftrightarrow \left[\forall \text{ safety } P \in (2^{\mathrm{AP}})^\omega, T \approx P \Leftrightarrow T' \approx P\right]\end{aligned}$$

□

We will end this section with a bit more terminology and practice with results on invariants and safety properties.

Definition 38 (Closure)**.** Let P be an LTP, we denote the set of finite prefixes of P by

$$\mathrm{pref}(P) = \{\hat{\sigma} \in (2^{\mathrm{AP}})^* \mid \exists \sigma \in P, \hat{\sigma} \subseteq \sigma\}.$$

The **closure** of P is the set of LTPs that have all their finite prefixes in P, that is,

$$\mathrm{cl}(P) = \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \mathrm{pref}(\sigma) \subseteq \mathrm{pref}(P)\}.$$

Proposition 39. *An LTP P is a safety property if and only if $cl(P) = P$.*

Proof. ($\Rightarrow$) Note that $P \subseteq \mathrm{cl}(P)$ is trivially true for any P.[39] Now, suppose that $\sigma \in \mathrm{cl}(P)$, for any finite prefix $\hat{\sigma} \subseteq \sigma$, $\hat{\sigma} \in \mathrm{pref}(P)$, so there exists σ' with $\hat{\sigma} \subseteq \sigma'$. In other words, $\hat{\sigma} \cdot (2^{\mathrm{AP}})^\omega \cap P \neq \emptyset$ and we conclude that $\sigma \in P$ by the contrapositive of Proposition **??**.

($\Leftarrow$) Let $\sigma \in P^c$, in particular $\sigma \notin \mathrm{cl}(P)$, so there is a finite prefix $\hat{\sigma} \subseteq \sigma$ that is not the prefix of any word in P. In mathematical terms, this means $\hat{\sigma} \cdot (2^{\mathrm{AP}})^\omega \cap P = \emptyset$. Since σ was arbitrary, P is a safety property by Proposition **??**. □

[39] One way to see this is:

$$\mathrm{pref}(P) = \bigcup_{\sigma \in P} \mathrm{pref}(\sigma).$$

Proposition 40. *Let P and Q be safety properties, then $P \cup Q$ and $P \cap Q$ are also safety properties.*

Proof. For the union, since a word is in $P \cup Q$ if it has no finite prefix in one of P_{bad} and Q_{bad}, it follows that $P \cup Q$ is the safety property induced by $(P \cup Q)_{\mathrm{bad}} := P_{\mathrm{bad}} \cap Q_{\mathrm{bad}}$.

For the intersection, it follows from Proposition **??** and

$$\mathrm{cl}(P) \cap \mathrm{cl}(Q) = \bigcup_{\sigma \in P} \mathrm{pref}(\sigma) \bigcap \bigcup_{\sigma \in Q} \mathrm{pref}(\sigma) = \bigcup_{\sigma \in P \cap Q} \mathrm{pref}(\sigma) = \mathrm{cl}(P \cap Q).$$

□

Liveness Properties

Definition 41 (Liveness)**.** A LTP $P \subseteq (2^{\mathrm{AP}})^\omega$ is a **liveness** property if for any $\hat{\sigma} \in (2^{\mathrm{AP}})^*$, there exists $\sigma \in (2^{\mathrm{AP}})^\omega$ such that $\hat{\sigma} \subseteq \sigma$ and $\sigma \in P$.

Example 42. The system T_{VM} from Example **??** satisfies the property

$$P = \{\sigma \mid \exists^\infty i, \mathit{available} \in \sigma(i) \implies \exists^\infty i, \mathit{paid} \in \sigma(i)\},$$

because both sides of the implication are true for any $\sigma \in \mathrm{Tr}^\omega(T_{\mathrm{VM}})$.[40] P is a liveness property because any finite word can be completed with $(\{\mathsf{available}\}\{\mathsf{paid}\})^*$.

[40]

Proposition 43. *An LTP P is a liveness property if and only if $\mathit{pref}(P) = (2^{AP})^*$.*

Proof. ($\Rightarrow$) Suppose there exists $\sigma \in (2^{\mathrm{AP}})^* \setminus \mathrm{pref}(P)$, then σ could not be extended into a word of σ, contradicting the liveness of P.

($\Leftarrow$) Any finite word is in $\mathrm{pref}(P)$, thus it can be extended in a word of P. We conclude that P is a liveness property. □

Corollary 44. *Let P and Q be liveness properties on AP, then $P \cup Q$ is also a liveness property.*[41]

[41] It follows from Proposition **??** because $\mathrm{pref}(P \cup Q) = \mathrm{pref}(P) \cup \mathrm{pref}(Q)$.

Example 45. Unlike for safety properties, the intersection of two liveness properties is not always a liveness property. Consider the following properties:[42]

$$P = \{\sigma \in \{\mathtt{0},\mathtt{1}\}^\omega \mid \forall^\infty i, \sigma(i) = \mathtt{1}\}$$
$$Q = \{\sigma \in \{\mathtt{0},\mathtt{1}\}^\omega \mid \forall^\infty i, \sigma(i) = \mathtt{0}\}.$$

[42] P and Q respectively contain all ω-words that are eventually all `1`s and all `0`s.

They are both clearly liveness as any finite word can be completed with either 1^ω or 0^ω and belong to P or Q respectively. However, their intersection is clearly empty and $\emptyset$ is not a liveness property.

Proposition 46. *The property $\top := (2^{AP})^\omega$ is the only LTP that is a liveness and safety property.*

Proof. Since any finite word can be completed into an ω-word, $\top$ is liveness. It is also safety induced by $P_{\text{bad}} = \emptyset$.

Let P be a safety property induced by P_{bad}, then any $x \in P_{\text{bad}}$ cannot be extended into a infinite path in P by definition. Therefore, if P is safety and liveness, P_{bad} must be empty and $P = \top$. □

Theorem 47 (Decomposition)**.** *Any LTP P can be decomposed in a liveness property and safety property. $P = P_{safe} \cap P_{live}$.*

In order to prove this theorem, we will introduce two very different approaches that lead to very elegant proofs. Thus, the two next sections will feel ad hoc at first, but they are very much used in current research in semantics, so they are worth covering.

Topological Spaces

Preliminaries

Not much theory is needed, but we present it here for completeness.

Definition 48. A **topological space** is a pair $(X, \Omega X)$, where X is a set and $\Omega X \subseteq 2^X$ is a set that is closed under arbitrary unions and finite intersections[43] whose elements are called **open sets** of X.

Complements of open sets (denoted U^c) are called **closed sets**. Observe that both the empty set and the whole space are open and closed (sometimes referred to as **clopen**) because

$$\emptyset = \bigcup_{U \in \emptyset} U \text{ and } X = \bigcap_{U \in \emptyset} U.$$

All the following terminology and results are basic tools use in topology that will end up helping us prove the decomposition theorem. Fix a topological space $(X, \Omega X)$.

Lemma 49. *Let $(C_i)_{i \in I}$ be a family of closed sets of X, then $\cap_{i \in I} C_i$ is closed and if I is finite, $\cup_{i \in I} C_i$ is also closed.*[44]

Proof. Both statements follow trivially from DeMorgan's laws and the fact that the complement of a closed set is open and vice-versa. For the first one, DeMorgan's laws yield

$$\bigcap_{i \in I} C_i = \left(\bigcup_{i \in I} C_i^c \right)^c,$$

[43] For any family of open sets $\{U_i\}_{i \in I}$,

$$\bigcup_{i \in I} U_i \in \Omega X,$$

and if I is finite,

$$\bigcup_{i \in I} U_i \in \Omega X.$$

[44] Observe that this are statements dual to the axioms of Definition **??**. In fact, it is sometimes more convenient to define a topological space by giving its closed sets, and it is equivalent.

and the LHS is the complement of a union of opens, so it is closed. For the second one, DeMorgan's laws yield

$$\bigcup_{i\in I} C_i = \left(\bigcap_{i\in I} C_i^c\right)^c,$$

and the LHS is the complement of a finite intersection of opens, so it is closed. □

Lemma 50. *A subset $A \subseteq X$ is open if and only if for any $x \in A$, there exists an open $U \subseteq A$ such that $x \in A$.*

Proof. ($\Rightarrow$) For any $x \in A$, set $U = A$.

($\Leftarrow$) For each $x \in X$, pick an open $U_x \subseteq A$ such that $x \in A$, then we claim $A = \cup_{x\in A} U_x$ which is open[45]. The $\subseteq$ inclusion follows because each $x \in A$ has a set U_x in the union that contains x. The $\supseteq$ inclusion follows because each term of the union is a subset of A by assumption. □

[45] Arbitrary unions of opens are open.

Lemma 51. *A subset $A \subseteq X$ is closed if and only if for any $x \notin A$, there exists an open U such that, $x \in U$ and $U \cap A = \emptyset$.*[46]

[46] This result is simply a restatement of the last one by setting $A = A^c$.

Definition 52. Given $A \subseteq X$, the **closure** of A is

$$\overline{A} := \bigcap \{C \text{ closed} \mid C \supseteq A\}.$$

It is very easy to show that $\overline{A}$ is the smallest closed set containing A.[47] Then, it follows that A is closed if and only if $\overline{A} = A$.

[47] $\overline{A}$ is closed because it is an intersection of closed sets and any closed sets containing A also contains $\overline{A}$ by definition.

Here are more easy results on the closure of a subset.

Lemma 53. *Given $A, B \subseteq X$ then the following statements hold:*

1. $A \subseteq B \implies \overline{A} \subseteq \overline{B}$.
2. $A \subseteq \overline{A}$
3. $\overline{\overline{A}} = \overline{A}$.
4. $\overline{\emptyset} = \emptyset$
5. $\overline{A \cup B} = \overline{A} \cup \overline{B}$.

Proof of Lemma **??**.
1. By definition, $\overline{B}$ contains B, thus A, but $\overline{B}$ is closed, so it must contain $\overline{A}$.
2. By definition.
3. $\overline{A}$ is closed, so its closure is itself.
4. 3 applied to $\emptyset$.
5. $\subseteq$ follows because the LHS is the smallest closed set containing $A \cup B$ and the RHS is closed and contains $A \cup B$. $\supseteq$ follows because the LHS is a closed set containing A and B, it contains $\overline{A}$ and $\overline{B}$.

□

Definition 54. A subset $A \subseteq X$ is said to be **dense** (in X) if any non-empty open set intersects A non-trivially, that is, $\forall U \neq \emptyset \in \Omega X, A \cap U \neq \emptyset$.

Theorem 55 (Decomposition)**.** *Let $A \subseteq X$, then $A = \overline{A} \cap (A \cup \overline{A}^c)$, where $\overline{A}$ is closed and $A \cup \overline{A}^c$ is dense.*[48]

[48] This results says that any set can be decomposed into a closed and a dense set. Note the similarity with Theorem **??**, we will see that the latter is a corollary of this basic result in topology.

Proof. The equality is trivial and $\overline{A}$ is closed by definition. It is left to show that $A \cup \overline{A}^c$ is dense.

Let $U \neq \emptyset$ be an open set. If U intersects A, we are done. Otherwise, we have the following equivalences:

$$U \cap A = \emptyset \Leftrightarrow A \subseteq U^c \Leftrightarrow \overline{A} \subseteq U^c \Leftrightarrow U \subseteq \overline{A}^c,$$

where the second $\implies$ holds because U^c is closed. We conclude $U \cap (A \cup \overline{A}^c) \neq \emptyset$. □

Lemma 56. *A subset $A \subseteq X$ is dense if and only if $\overline{A} = X$.*

Proof. ($\Rightarrow$) Since $\overline{A}^c$ is open but it intersects trivially a dense set A, it must be empty, thus $\overline{A}$ is the whole space.

($\Leftarrow$) Let U be an open set such that $U \cap A = \emptyset$, then A is contained in the closed set U^c, but this implies $\overline{A} \subseteq U^c$,[49] thus U is empty. □

[49] Recall that the closure of A is the smallest closed set containing A.

Definition 57. Let $A \subseteq X$, the **interior** of A is

$$A^{\circ} := \bigcup \{U \in \Omega X \mid U \subseteq A\}.$$

It is obvious that A° is the largest open subset of A and thus that A is open if and only if $A = A^{\circ}$.[50]

[50] It also follows that $A \subseteq B \implies A^{\circ} \subseteq B^{\circ}$ and that $A^{\circ\circ} = A^{\circ}$.

Finally, we end this these preliminaries with a result on how to specify a topology.

Definition 58 (Base)**.** Let X be a set, a **base** B is a set $B \subseteq 2^X$ such that $X = \cup_{U \in B} U$ and any finite intersection of sets in B can be written as a union of sets in B.

Lemma 59. *Let X and $B \subseteq 2^X$, then ΩX be the set of all unions of sets in B, $(X, \Omega X)$ is a topology. We say that ΩX is the topology **generated** by B.*

Proof. We know that unions of opens are open and finite intersections of sets in B are open. It remains to show that finite intersections of unions of sets in B are also open. Let $U = \cup_{i \in I} U_i$ and $V = \cup_{j \in J} V_j$ with $U_i \in B$ and $V_j \in B$, then by distributivity, we obtain

$$U \cap V = \cup_{i \in I} U_i \bigcap \cup_{j \in J} V_j = \bigcup_{i \in I, j \in J} U_i \cap V_j,$$

so $U \cap V$ is open (being a union of opens). The lemma then follows by induction. □

In practice, instead of generating a topology from a base B, we start with any family $B_0 \subseteq 2^X$ and consider its closure under finite intersections B, so that it satisfies the axioms of a base. Such a B_0 is often called a **subbase** for the topology generated by B.

Although we could indefinitely extend this digression in topology, we will stop now that we have the essentials and see how this relates to our investigation on linear time properties.

ω-words

Definition 60 (Extensions)**.** Given a non-empty set A and a finite word $u \in A^*$, we denote the extensions of u by[51]

$$\mathrm{ext}_A(u) := u \cdot A^{\omega} = \{\sigma \in A^{\omega} \mid u \subseteq \sigma\}.$$

[51] We write $\mathrm{ext}(u)$ when the alphabet is clear from context. Note that $A^{\omega} = \mathrm{ext}_A(\varepsilon)$.

We generalize this notation to sets of finite words $W \subseteq A^*$ in the natural way:

$$\mathrm{ext}(W) = \{\mathrm{ext}(u) \mid u \in W\}.$$

Proposition 61. *Let A be a non-empty set, the set*[52]

$$\Omega A = \{\text{ext}(W) \mid W \subseteq A^*\}$$

is a topology for A^ω.

Proof. Let $\{U_i\}_{i\in I}$ be a family of opens, for any $i \in I$, there exists $W_i \subseteq A^*$ such that $U_i = \text{ext}(W_i)$. Then,[53]

$$\cup_{i\in I} U_i = \cup_{i\in I} \text{ext}(W_i) = \text{ext}\left(\cup_{i\in I} W_i\right),$$

so we conclude that ΩA is closed under unions.

Furthermore, we observe that[54]

$$\text{ext}(u) \cap \text{ext}(v) = \begin{cases} \text{ext}(u) & v \subseteq u \\ \text{ext}(v) & u \subseteq v \\ \emptyset & \text{o/w} \end{cases}.$$

Thus, let $W_1, W_2 \subseteq A^*$, we have

$$\begin{aligned} \text{ext}(W_1) \cap \text{ext}(W_2) &= \cup_{u\in W_1} \text{ext}(u) \bigcap \cup_{v\in W_2} \text{ext}(v) \\ &= \bigcup_{u\in W_1, v\in W_2} \text{ext}(u) \cap \text{ext}(v) \\ &= \bigcup \{\text{ext}(u) \mid u \in W_1, \exists v \in W_2, v \subseteq U\} \\ &\quad \cup \bigcup \{\text{ext}(v) \mid v \in W_2, \exists u \in W_1, u \subseteq v\} \\ &= \text{ext}(W_1 \Cap W_2), \end{aligned}$$

where $W_1 \Cap W_2$ is the set of words in one of the W_i's that have a prefix in the other, that is,

$$W_1 \Cap W_2 = \{u \in A^* \mid \exists i \neq j, \exists v \in W_j, u \in W_i, v \subseteq u\}.$$

We conclude that ΩA is also closed under finite intersection, so it is a topology for A^ω. □

From now on, unless otherwise said, we assume that the topology on A^ω is generated by ΩA given above.

Remark 62. A set $P \subseteq A^\omega$ is open if and only if there exists $W \in A^*$ such that $P = \text{ext}(W) = \cup_{u\in W} \text{ext}(u)$. In particular, if $\sigma \in P$, then there exists $\hat{\sigma} \subseteq \sigma$ such that $\text{ext}(\hat{\sigma}) \subseteq P$.[55]

If we stare at this remark long enough, we can recover the intuition behind safety LTPs and in particular, the equivalent definitions seen in Proposition **??**. Indeed, the tools we have developed lead to a nice characterization of safety and liveness properties.

Lemma 63. *An LTP $P \subseteq (2^{AP})^\omega$ is a safety property if and only if it is closed.*[56]

[52] Notice that this topology is generated by the subbase with containing $\text{ext}(u)$ for any $u \in A^*$ as $\text{ext}(W)$ is the union of such sets when $W \subseteq A^*$.

[53] The last equality holds because an ω-word extends a word in one of the W_i's if and only if it extends the same word in the union of the W_i's.

[54] Indeed, if $u \subseteq v$, then any ω-word that extends v also extends u, so $\text{ext}(v) \subseteq \text{ext}(u)$. The argument is symmetric for $v \subseteq u$ and if v and u are not comparable, then they do not agree at some index and no ω-word can have two distinct symbols at this index.

[55] In other words, we will know that $\sigma \in P$ after observing it for a finite amount of time because we know all extensions of $\hat{\sigma}$ are in P.

[56] In the usual topology on ω-words.

Proof. ($\Leftarrow$) We have just said that if σ is in an open (in this case P^c), then there exists $\hat{\sigma} \subseteq \sigma$ such that $\text{ext}(\hat{\sigma}) \subseteq P^c$, i.e. $\text{ext}(\hat{\sigma}) \cap P = \emptyset$ as required for safety properties.

($\Rightarrow$) Let P be induced by P_{bad}, any extension σ of a word in P_{bad} is not in P. In other words $P^c = \text{ext}(P_{\text{bad}})$ which is open, thus P is closed. □

Lemma 64. *An LTP $P \subseteq (2^{AP})^\omega$ is a liveness property if and only if it is dense.*

Proof. ($\Rightarrow$) Let $U = \emptyset$ be open, then $U = \cup_{u \in W} \text{ext}(u)$, where W cannot be empty. Hence, since for any $u \in W$, $\text{ext}(u) \cap P \neq \emptyset$[57], $P \cap U \neq \emptyset$ and this direction follows.

($\Leftarrow$) Let P be dense, then for any u, $\text{ext}(u)$ is open, so $P \cap \text{ext}(u) \neq \emptyset$ which means we can extend u to be in P. This direction follows. □

[57] Because any finite word can be extended (by liveness).

Corollary 65. *Theorem* **??**.[58]

[58] Indeed, any set can be written as the intersection of a closed set and a dense set by Theorem **??**, which are respectively safety and liveness properties.

As we mentioned before, we have not gone through all this theory only to prove this theorem, these concepts will come back in this class as well as in a more advanced study of semantics. However, before going back to the main point of this class, we keep our promise of giving two proofs of the decomposition theorem. Therefore, in the next section, we will revisit this characterization through the point of view of lattice theory.

Posets and Complete Lattices

Preliminaries

Definition 66. A **poset** (short for partially ordered set) is a pair $(A, \leq)$ where A is a set and $\leq \subseteq A \times A$ is a reflexive, transitive and antisymmetric binary relation.

A prototypical example of a poset which is paradigmatic in this course is the powerset with binary relation being inclusion :$(2^X, \subseteq)$. The restriction of this poset to the opens ΩX of a topological space also yields an important poset: $(\Omega X, \subseteq)$.

Definition 67. A function $f : (A, \leq_A) \to (B, \leq_B)$ between posets is **monotone** (or **order-preserving**) if for any $a, a' \in A$, $a \leq a' \implies f(a) \leq f(a')$.

Example 68. The closure in a topological space X is a monotone function from $(2^X, \subseteq)$ to itself because $A \subseteq B$ implies $\overline{A} \subseteq \overline{B}$.

Definition 69. The **dual** of a poset $(A, \leq)$ is denoted $(A, \leq)^{\text{op}} := (A, \geq)$, where for any $a, a' \in A$, $a' \geq a \Leftrightarrow a \leq a'$.[59]

[59] This definition lets us avoid many symmetric arguments.

Definition 70. Let $(A, \leq)$ be a poset and $S \subseteq A$, then $a \in A$ is an **upper bound** of S if $\forall s \in S, s \leq a$. Moreover, $a \in A$ is the **supremum** of S, denoted $\vee S$, if it is the least upper bound, that is, a is an upper bound of S and for any upper bound a' of S, $a \leq a'$.

Dually, $a \in A$ is **a lower bound** (resp. **infimum**) of S if and only if it is an upper bound (resp. supremum) of S in $(A, \leq)^{\text{op}}$.

Proposition 71. *Infimums and supremums are unique when they exist.*[60]

[60] By antisymmetry.

Definition 72. A **complete lattice** is the data $(L, \wedge, \vee, \leq)$ where $(L, \leq)$ is a poset, and $\wedge, \vee : (2^L, \subseteq) \to (L, \leq)$ are respectively infimum (or **meet**) and the supremum (or **join**) as defined above.[61] Observe that L has a smallest element that we denote $\bot := \vee\emptyset$ and a largest element $\top := \wedge\emptyset$.

[61] Thus, all supremums and infimums exist in $(L, \leq)$.

Lemma 74. *Let $(L, \leq)$ be a poset, then the following are equivalent:*

(i) *$(L, \wedge, \vee, \leq)$ is a complete lattice.*

(ii) *Any $S \subseteq L$ has a supremum.*

(iii) *Any $S \subseteq L$ has an infimum.*

Proof. (i) $\implies$ (ii), (i) $\implies$ (iii) and (ii) + (iii) $\implies$ (i) are all trivial. Also, by using duality, we only need to prove (ii) $\implies$ (iii). For that, it suffices to note that for any $S \subseteq L$, $\wedge S = \bigvee\{a \in L \mid \forall s \in S, a \leq s\}$ is a suitable definition of the infimum.

Defined that way, $\wedge S$ is a lower bound of S because if $s < \wedge S$, then $s < a$ for some lower bound a of S[62], in particular $s \notin S$. Additionally, since we are taking the supremum over all lower bounds of S, no lower bound of S can be greater and we conclude that $\wedge S$ is indeed the infimum of S. □

[62] Because $\wedge S$ was the least upper bound for lower bounds of S.

Example 73. Again, the powerset with inclusion order is a good example, the join of a family of subsets is their union and the meet is their intersection.

Example 75. As a corollary, we obtain that the open sets of a topological space form a complete lattice. The supremums are given by unions which are open for any arbitrary families of open sets. However, while the finite infimums are given by intersection and infinite infimums exist by the previous lemma, they are not necessarily intersections.[63]

For instance, consider the topology on A^ω with $A = \{\mathtt{a}, \mathtt{b}\}$ and $P = \cap_{n \in \mathbb{N}} \mathrm{ext}(a^n)$. All the elements in the intersection are open by definition, but P is not open because $a^\omega \in P$ and there does not exists $a^n \subseteq a^\omega$ with $\mathrm{ext}(a^n) \subseteq P$. However, the interior of $P = \{a^\omega\}$ which is $\emptyset$ is open.

[63] In fact, the formula given above, states that the infimum of a family of opens is the **interior** of its intersection.

Definition 76. Let $(A, \leq)$ be a poset, a **closure operator** on A is a map $c : A \to A$ that is monotone, expansive and idempotent.[64] We say that $a \in A$ is **closed** if $a = c(a)$, the set of **closed elements** of A is denoted A_c.

[64] That is, $\forall a, a' \in A$,

$$a < a' \implies c(a) \leq c(a')$$
$$a \leq c(a)$$
$$c(a) = c(c(a)).$$

A typical example is the closure operator in topological spaces as we have said, but it satisfies more properties that are not usually satisfied by closure operators. These are properties 4 and 5 in Lemma **??** and if a closure operator satisfies these properties, it is a **Kuratowski closure operator**.[65]

[65] In fact, any Kuratowski closure operator $c : 2^X \to 2^X$ yields a topology on X (by defining the closed sets instead of the opens).

Lemma 77. *Let $(L, \wedge, \vee, \leq)$ be a complete lattice and $c : L \to L$ a closure operator on L, then $(L_c, \leq)$ is also a complete lattice where infimums are taken as in L and supremums are the closure of the supremums taken in L.*[66]

[66] In particular, c has greatest fixed points and least fixed points. In fact, only the fact that c is monotone is enough to show the existence of the greatest and least fixed points. We will see that they play an important role in our study of transition systems.

Proof. The fact that $\leq$ is also a partial order on L_c is trivial. Also, if S only has closed elements, then $\wedge S$ is also closed because $c(\wedge S) \leq c(s) = s$ for any $s \in S$ by monotonicity, but $\wedge S \leq c(\wedge S)$ by expansiveness. We can conclude because $c(\wedge S)$ is a lower bound greater than $\wedge S$, so they must be equal.

Now, we will show that for $S \subseteq L_c$, $c(\vee S)$ is the supremum of S in L_c. It is clearly an upper bound because $\vee S \leq c(\vee S)$. It is the least because if u is an upper bound of S that is closed, then the fact that $u \geq \vee S$ implies $u = c(u) \geq c(\vee S)$. □

Lemma 78. *Let $(L, \wedge, \vee, \leq)$ be a complete lattice and $c : L \to L$ a closure operator, then for any $a \in L$, $c(a) = \bigwedge\{c(b) \in L_c \mid a \leq c(b)\}$.*

Proof. Since $a \leq c(b)$ implies $c(a) \leq c(c(b)) = c(b)$, $c(a)$ is a lower bound fo this set. It is the infimum. because it belongs to this set, thus any $d > c(a)$ is not a lower bound. □

Definition 79. Given two posets $(A, \leq)$ and $(B, \preceq)$, a **Galois connection** is a pair of functions $g : A \to B$ and $f : B \to A$ such that for any $a \in A$ and $b \in B$,

$$g(a) \preceq b \Leftrightarrow a \leq f(b).$$

For such a pair, we write $g \dashv f : A \to B$.

Lemma 80. *Let $g \dashv f : A \to B$ be a Galois connection, then g and f are monotone.*

Proof. Assume towards a contradiction that $a < a'$ and $g(a) \succ g(a')$, then because $g(a') \preceq g(a')$, we infer that $a' \leq f(g(a'))$ and thus, by transitivity, $a \leq f(g(a'))$. However, this contradicts the fact that $g(a) \not\preceq g(a')$ (using the $\Leftarrow$ of the Galois connection). We conclude that g is monotone.

A symmetric argument works to show that f is monotone. □

Example 81.

Lemma 82. *Let $g \dashv f : A \to B$ be a Galois connection, then $f \circ g : A \to A$ is a closure operator.*

Proof. Because f and g are monotone, $f \circ g$ is clearly monotone. Also, for any $a \in A$, $g(a) \preceq g(a)$ implying $a \leq f(g(a))$, so $f \circ g$ is expansive.

Now, in order to prove $f \circ g$ is idempotent, it is enough to show that[67]

$$f(g(a)) \geq f(g(f(g(a)))).$$

Observe that since $f(b) \leq f(b)$ for any $b \in B$, we have $g(f(b)) \leq b$, thus in particular, with $b = g(a)$, we have $g(f(g(a))) \leq g(a)$. Applying f which is monotone yields the desired inequality. □

[67] The $\leq$ inequality follows by expansiveness.

Prefixes and Closure

For this section, we fix a non-empty set A. Recall the operations of prefixes and closure defined in Definition **??**, we will study them with the view point of lattice theory.[68] First, let us generalize the definition of pref and give a slightly different definition of closure:

$$\text{pref} : \mathcal{P}(A^\omega) \to \mathcal{P}(A^*) = P \mapsto \{\hat{\sigma} \in A^* \mid \exists \sigma \in P, \hat{\sigma} \subseteq \sigma\}$$

[68] The motivation behind this is the results of Proposition **??** and **??** that characterize safety and liveness properties in terms of this operations.

$$\underline{\mathrm{cl}} : \mathcal{P}(A^*) \to \mathcal{P}(A^\omega) = W \mapsto \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \mathrm{pref}(\sigma) \subseteq W\}.$$

Observe that $\mathrm{cl} = \underline{\mathrm{cl}} \circ \mathrm{pref} : \mathcal{P}(A^\omega) \to \mathcal{P}(A^\omega)$[69] and in fact, we can prove that cl is a closure operator with the following lemma that states $\mathrm{pref} \dashv \underline{\mathrm{cl}}$ is a Galois connection.

[69] $\mathrm{cl}(P) = \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \mathrm{pref}(\sigma) \subseteq \mathrm{pref}(P)\}$

Lemma 83. *For any $P \in \mathcal{P}(A^\omega)$ and $W \in \mathcal{P}(A^*)$, then*

$$\mathit{pref}(P) \subseteq W \Leftrightarrow P \subseteq \underline{\mathit{cl}}(W).$$

Proof. ($\Rightarrow$) For any $\sigma \in P$, we have $\mathrm{pref}(\sigma) \subseteq W$, thus $\sigma \in \underline{\mathrm{cl}}(W)$.

($\Leftarrow$) Because any $\sigma \in P$ is also in $\underline{\mathrm{cl}}(W)$, we infer that $\mathrm{pref}(\sigma) \subseteq W$. Now, since $\mathrm{pref}(P) = \cup_{\sigma \in P} \mathrm{pref}(\sigma)$ and all terms are subsets of W, we conclude $\mathrm{pref}(P) \subseteq W$. □

Moreover, one can show that cl coincides with the closure operator in the topological space $\mathcal{P}(A^\omega)$.

Proposition 84. *For any $P \subseteq A^w$, $\overline{P} = cl(P)$.*

Proof. First, we show that $\mathrm{cl}(P)$ is closed.[70] It is closed because if $\sigma \notin \mathrm{cl}(P)$, then there exists $\hat{\sigma}$ such that $\hat{\sigma}$ is not a prefix of any word in P. Therefore, $\mathrm{ext}(\hat{\sigma})$ is an open set containing σ that does not intersect $\mathrm{cl}(P)$. By Lemma **??**, $\mathrm{cl}(P)$ is closed.

Second, we show that $\mathrm{cl}(P) \subseteq \overline{P}$. Suppose that there exists $\sigma \in A^\omega$ that is in $\mathrm{cl}(P)$, but not in $\overline{P}$. By Lemma **??** again, we have an open set U containing σ and not intersecting $\overline{P}$. Without loss of generality, $U = \mathrm{ext}(\hat{\sigma})$ for some prefix $\hat{\sigma} \subseteq \sigma$.[71] However, because $\sigma \in \mathrm{cl}(P)$, $\hat{\sigma}$ is the prefix of some word in P contradicting the fact that $\mathrm{ext}(\hat{\sigma})$ does not intersect P. □

[70] It is obvious that it contains P.

[71] Indeed, we already know open sets have the form $\mathrm{ext}(W)$ for $W \subseteq A^*$. Thus, $U = \mathrm{ext}(W) = \cup_{i \in I} \mathrm{ext}(w_i)$, and if $\sigma \in U$ we can choose one i with $\sigma \in \mathrm{ext}(w_i)$, which means w_i is the desired prefix.

Corollary 85. *Let $P \subseteq (2^{AP})^\omega$, then*

1. *P is a safety property if and only if $cl(P) = P$.*
2. *P is a liveness property if and only if $cl(P) = (2^{AP})^\omega$ if and only if $pref(P) = (2^{AP})^*$.*

Before ending this section, let us spend a bit more time expanding on the properties of Galois connections.

Proposition 86. *Let $g \dashv f : A \to B$ be a Galois connection where A and B are complete lattices, then g preserves supremums and f preserves infimums.*[72]

[72] For any $S \subseteq A$, $g(\vee S) = \vee g(S)$ and for any $T \subseteq B$, $f(\wedge T) = \wedge f(T)$.

Proof. Let $S \subseteq A$, we claim that $g(\vee S)$ is the supremum of $g(S)$. By monotonicity, it is an upper bound, and suppose $g(\vee S) \succ \vee g(S)$, then $\vee S > f(g(\vee S))$, which contradicts the expansiveness of $f \circ g$. The claim follows.

Let $T \subseteq B$, we claim that $f(\wedge S)$ is the infimum of $f(S)$. By monotonicity, it is a lower bound, and suppose $f(\wedge S) \prec \wedge f(S)$, then $\wedge S$ □

In fact, a kind of converse of this result holds.

Proposition 87. *Let A and B be complete lattices, then if $g : A \to B$ preserves all supremums, then there exists $f : B \to A$ such that $g \dashv f$ is a Galois connection.*[73]

[73] By duality (considering the opposite orders), if $f : B \to A$ preserves all infimums, then there is a Galois connection $g \dashv f$.

Proof. We define $f = b \mapsto \bigvee\{a \in A \mid g(a) \leq b\}$ and we have to show that $g(a) \leq b \Leftrightarrow a \leq f(b)$. The $\Rightarrow$ direction is trivial because clearly a is in the set that $f(b)$ is the upper bound of, so $a \leq f(b)$.

For $\Leftarrow$, note that g preserving supremums implies g is monotone.[74] Thus, if $a \leq f(b)$, we have

$$g(a) \leq g(f(b)) = \bigvee\{g(a) \mid g(a) \leq b\} \leq b.$$

□

[74] Assume $a \leq a'$, then $a \vee a' = a'$, so $g(a) \vee g(a') = g(a')$, because g preserves supremums. Therefore, $g(a) \leq g(a')$.

Observable Properties

Continuous Functions

Let $f : Y \to X$, the inverse image function is

$$f^{-1} : 2^X \to 2^Y = A \mapsto \{y \in Y \mid f(y) \in A\}.$$

Let us show a basic, but fundamental result.

Lemma 88. *For any $f : Y \to X$, f^{-1} preserves arbitrary unions and intersection. Thus, it also preserves complements.*

Proof. □

Definition 89. Let $(X, \Omega X)$ and $(Y, \Omega Y)$ be topological spaces, a function $f : Y \to X$ is **continuous** if $f^{-1}(\Omega X) \subseteq \Omega Y$, that is, the preimage of any open set in X is open in Y.

Lemma 90. *A function $f : A^\omega \to B^\omega$ is continuous if and only if*

$$\forall \alpha \in A^\omega, \forall n \in \mathbb{N}, \exists k \in \mathbb{N}, \forall \beta \in A^\omega,$$
$$\alpha(0) \cdots \alpha(k) = \beta(0) \cdots \beta(k) \implies f(\alpha)(0) \cdots f(\alpha)(n) = f(\beta)(0) \cdots f(\beta)(n).$$

Proof. ($\Rightarrow$) An other formulation of the implication is that $f^{-1}(\text{ext}(x)) \supseteq \text{ext}(y)$ with $y := \alpha(0) \cdots \alpha(k)$ and $x := f(\alpha)(0) \cdots f(\alpha)(n)$. Now, note that $f^{-1}(\text{ext}(x))$ contains α, and since it is open, Remark **??** tells us that there exists $\hat{\sigma} \subseteq \sigma$ with $\text{ext}(\hat{\sigma}) \subseteq f^{-1}(\text{ext}(x))$. Letting $k = |\hat{\sigma}|$ yields the desired y.

($\Leftarrow$) Since f^{-1} preserves unions, open sets are all of the form $\text{ext}(W) = \cup_{x \in W} \text{ext}(x)$ and arbitrary unions of opens are open, it is enough to show that $f^{-1}(\text{ext}(x))$ is open for any $x \in A^*$. Let $\alpha \in f^{-1}(\text{ext}(x))$ and $n = |x|$, the hypothesis tells us that there exists $y = \alpha(0) \cdots \alpha(k)$, such that $f^{-1}(\text{ext}(x)) \supseteq \text{ext}(y)$. Since $\alpha \in \text{ext}(y)$ and $\text{ext}(y)$ is open, we conclude by Lemma **??** that $f^{-1}(\text{ext}(x))$ is open. □

In other words, in order to determine a finite prefix of the image of α, we only need to observe a finite prefix of α.

Remark 91. This property is also important when we talk about "computable" functions on streams, they are always continuous. Consequently, a "decidable" property must have a continuous characteristic function.[75]

[75] In the codomain, the booleans are equipped with the discrete topology $(\{0,1\}, \{\emptyset, \{0\}, \{1\}, \{0,1\}\})$.

In light of this remark, given $P \subseteq A^\omega$ with a continuous characteristic function χ_P, we infer that $\chi_P^{-1}(1) = P$ and $\chi_P^{-1}(0) = P^c$ are open, or equivalently P is clopen.[76] Notice now that the subbase we used to form the topology on ω-words, namely, all the extensions of finite words, is very nice as it only contains clopen sets.

[76] We will see that clopens are important because they form a Boolean algebra (which we will see formally later) as $\emptyset$, X, $A \cup B$, $A \cap B$ and A^c are clopen whenever A and B are clopen.

Proposition 92. *Let A be a non-empty set and $u \in A^*$, then* $\mathrm{ext}(u)$ *is clopen.*

Proof. We proceed by induction on the length of u. If $u = \varepsilon$, then $\mathrm{ext}(\varepsilon) = A^\omega$ which is the whole space, so it is clopen.

Now, let $u \in A^*$ with $\mathrm{ext}(u)$ clopen, then for any $a \in A$, we have already seen that $\mathrm{ext}(u \cdot a)$ is open. To see that it is closed, observe that

$$A^\omega \setminus \mathrm{ext}(u \cdot a) = (A^\omega \setminus \mathrm{ext}(u)) \bigcup \cup_{b \neq a \in A} \mathrm{ext}(u \cdot b),$$

which is open because arbitrary unions of opens are open. □

Corollary 93. *If $S : W \subseteq A^*$ is finite, then* $\mathrm{ext}(W)$ *is clopen.*[77]

[77] Follows because finite unions of closed sets are closed.

However, the converse is not always true as shown in this example.

Example 94. Let $A = \mathbb{N}$ and consider $P = \cup_{n>0} \mathrm{ext}(n)$, it is open because each term in the union is open, and it is closed because it is the complement of $\mathrm{ext}(0)$. However, there is no finite set of prefixes W such that $P = \mathrm{ext}(W)$.

To finish this section, we will show that the only defect to obtaining the converse is the fact that A is not finite. In order to do this, we will need to linger on in the realm of topology.

Compactness

Definition 95. Let $(X, \Omega X)$, $A \subseteq X$, we say that a family $\{U_i\}_{i \in I}$ is an **open cover** of A if all U_i's are open are they cover A, i.e.: $A \subseteq \cup_{i \in I} U_i$. If $J \subseteq I$ is such that $A \subseteq \cup_{j \in J}$, we say that $\{U_j\}_{j \in J}$ is a **subcover**. It is a **finite subcover** if J is finite.

Definition 96. Let $(X, \Omega X)$ be a topological space, then $A \subseteq X$ is said to be **compact** if any cover of A has a finite subcover.[78]

[78] We say that X is a compact topological space if it is a compact subset of itself.

Fact 97. *If A is infinite, then A^ω is not compact.*

Proof. Consider $A^\omega = \cup_{a \in A} \mathrm{ext}(a)$, if A is infinite, then we cannot find a finite subcover. □

Proposition 98. *Let $A \neq \emptyset$ be finite, then A^ω is compact.*[79]

[79] This result is equivalent to the axiom of choice.

Proof. Let $\{U_i\}_{i \in I}$ be an open cover of A. For any $i \in I$, let $V_i \subseteq A^*$ be such that $U_i = \mathrm{ext}(V_i)$ and define $V = \cup_{i \in I} V_i \subseteq A^*$. Then, for $n \in \mathbb{N}$, we inductively define $W_n \subseteq A^n$ as follows.

For $n = 0$, we let $W_0 = \{\varepsilon\}$ if $\varepsilon \in V$, otherwise W_0 is empty. Suppose all W_i's are defined up to n, W_{n+1} is defined such that for any $u \in A^{n+1}$, $u \in W_{n+1}$ if and only if $u \in V$ and u has no prefix in the previously defined W_i's.

Let $W = \cup_{n\in\mathbb{N}} W_n$, it satisfies two properties. Clearly, it is prefix free[80] and any word in V has a prefix in W, because either this word was added to one of the W_i's, or it was not added because one of its prefix was already added. We conclude that $A^\omega = \text{ext}(W)$ because $\text{ext}(W) \supseteq \text{ext}(V)$. Moreover, we claim that W is finite and this finishes the proof.[81]

Assume towards a contradiction that W is infinite, then $T = \{x \in A^* \mid \exists w \in W, x \subseteq w\}$ is infinite as it contains W. Moreover, T can be seen as a subtree of the tree on A^* where u is a parent of v if and only if $v = u \cdot a$ for $a \in A^*$. Since that tree is finitely branching, T is too and by Konig's lemma, T contains an infinite path. That is, there exists $\sigma \in A^\omega = \text{ext}(W)$ such that for any $n \in \mathbb{N}$, $\sigma(0)\cdots\sigma(n)$ is a prefix of some $w_n \in W$. However, this contradicts the fact that W is prefix free because □

[80] Namely, if $u \in W$, then no prefix of u is in W.

[81] Because we can pick one of the V_i's containing w for each word $w \in W$ and this forms a finite subcover of A^ω.

Lemma 99. *If $(X, \Omega X)$ is a compact topological space and $C \subseteq X$ is closed, then C is compact.*

Proof. Let $\{U_i\}_{i\in I}$ be an open cover of C, then adding C^c to this family yields an open cover of X. Thus it has a finite subcover, which after removing C^c yields a finite subcover of $\{U_i\}_{i\in I}$. □

Hausdorff Spaces

Definition 100. A topological space $(X, \Omega X)$ is **Hausdorff** (or T_2, or separated) if for any $x \neq y \in X$, there exists $U, V \in \Omega X$ such that $x \in U$, $y \in V$ and $U \cap V = \emptyset$.

Example 101. The space A^ω with the usual topology is always Hausdorff because if $\alpha \neq \beta \in A^\omega$, then there is a finite index i where they disagree. Therefore, with $x = \alpha(0)\cdots\alpha(i)$ and $y = \beta(0)\cdots\beta(i)$, $\text{ext}(x)$ and $\text{ext}(y)$ are the desired separating sets.

Proposition 102. *Let $(X, \Omega X)$ be a Hausdorff space and $C \subseteq X$ be compact, then C is closed.*

Proof. Let $x \notin C$, for any $y \in C$, x and y are separated as in the Hausdorff definition by sets U_y and V_y, where $y \in V_y$. Note that $\{V_y\}_{y\in C}$ is an open cover of C and by compactness, there is a finite set I such that $\{V_{y_i}\}_{i\in I}$ still covers C. But, now $\cap_{i\in I} U_{y_i}$ is a finite intersection of opens that contains x and that cannot contain any point in C.[82] Thus, it is an open set disjoint from C that contains x. The proposition follows by Lemma **??**. □

[82] For each $i \in I$, U_{y_i} intersects V_{y_i} trivially, hence the intersection of all U_{y_i} cannot intersect any V_{y_i}. The claim follows since the latter cover C.

Corollary 103. *In A^ω the closed sets are exactly the compact sets, thus the clopen sets are exactly the open and compact sets.*[83]

[83] By Lemma **??** and Proposition **??**.

Corollary 104. *If $A \neq \emptyset$ is finite, then $P \subseteq A^\omega$ is clopen if and only if $P = \text{ext}(W)$ with $W \subseteq A^*$ finite.*

Proof. ($\Leftarrow$) Corollary **??**.

($\Rightarrow$) Since P is open and compact, the open cover $\{\text{ext}(w)\}_{w\in W}$ has a finite subcover $W_f \subseteq W$ with $P = \cup_{w\in W_f} \text{ext}(w) = \text{ext}(W_f)$. □

Linear Temporal Logic (LTL)

Linear Modal Logic (LML)

This logic is really simple and based on the concept of observable properties we have recently characterized through topological concepts. We will first describe the syntax and semantics of LML.

In this section, we fix an infinite countable set of variables $\mathcal{X} = \{X, Y, Z, \dots\}$ and a set of atomic propositions AP.

Definition 105. The **formulas** in LML (over AP) are given by the following grammar:[84]

$$\phi, \psi ::== \top \mid \bot \mid X \in \mathcal{X} \mid a \in \mathrm{AP} \mid \phi \wedge \psi \mid \phi \vee \psi \mid \neg\phi \mid \bigcirc\phi.$$

[84] All the usual connectives have the same semantics as before and the new connective $\bigcirc$ is the *linear* part of this logic. We read $\bigcirc\phi$ as "next phi", its semantics will become clearer when we define its interpretation.

Definition 106. 1. A **valuation** of a subset $V \subseteq X$ is a function $\rho : V \to \mathcal{P}\left((2^{\mathrm{AP}})^\omega\right)$.

2. A **formula with parameters** is a pair (ϕ, ρ), where ρ is a valuation on V such that V contains all free variables of ϕ.

The interpretation of a formula ϕ with parameters ρ is an LTP $[\![\phi]\!]_\rho \in \mathcal{P}\left((2^{\mathrm{AP}})^\omega\right)$ defined inductively as follows:[85]

$$[\![\top]\!]_\rho = (2^{\mathrm{AP}})^\omega \qquad [\![\bot]\!]_\rho = \emptyset$$
$$[\![X]\!]_\rho = \rho(X) \qquad [\![a]\!]_\rho = \left\{\sigma \in (2^{\mathrm{AP}})^\omega \mid a \in \sigma(0)\right\}$$
$$[\![\phi \wedge \psi]\!]_\rho = [\![\phi]\!]_\rho \cap [\![\psi]\!]_\rho \qquad [\![\phi \vee \psi]\!]_\rho = [\![\phi]\!]_\rho \cup [\![\psi]\!]_\rho$$
$$[\![\neg\phi]\!]_\rho = (2^{\mathrm{AP}})^\omega \setminus [\![\phi]\!]_\rho \qquad [\![\bigcirc\phi]\!]_\rho = \left\{\sigma \in (2^{\mathrm{AP}})^\omega \mid \sigma\restriction 1 \in [\![\phi]\!]_\rho\right\}.$$

[85] The intuition behind these will be made clearer when we talk about satisfaction of a formula.

We use the notation $\sigma \restriction i$ as a shorthand for $\sigma \circ \mathrm{succ}^i$, that is $\sigma\restriction i(k) = \sigma(k+i)$ for any $k \in \mathbb{N}$.

We also define some syntactic sugar for implications and equivalences, namely,

$$\phi \to \psi := \neg\phi \vee \psi \quad \text{and} \quad \phi \leftrightarrow \psi = (\phi \to \psi) \wedge (\psi \to \phi).$$

Definition 107. We say that $\sigma \in (2^{\mathrm{AP}})^\omega$ **satisfies** (ϕ, ρ) if $\sigma \in [\![\phi]\!]_\rho$.

Let us give two lemmas that are proven with a simple structural induction and that will help us make other proofs more clear.

Lemma 108. *Let $\rho, \rho' : V \to \mathcal{P}\left((2^{AP})^\omega\right)$ be such that $\rho(X) = \rho'(X)$ for any free variable X in ϕ, then $[\![\phi]\!]_\rho = [\![\phi]\!]_{\rho'}$.*[86]

[86] Proved by a simple structural induction on the formulas.

Lemma 109. *Let ϕ and ψ be formulas with parameters ρ and $X \in \mathcal{X}$ not in ψ, then*[87]

$$[\![\phi]\!]_{\rho[[\![\psi]\!]_\rho / X]} = [\![\phi[\psi/X]]\!]_\rho.$$

[87] For parameters ρ, $X \in \mathcal{X}$ and $A \subseteq (2^{\mathrm{AP}})^\omega$, the parameter $\rho[A/X]$ acts as ρ on any variable except for X were $\rho[A/X](X) = A$.

For a formulas ϕ and ψ and $X \in \mathcal{X}$, $\phi[\psi/X]$ is the formula ϕ where all free occurrences of X have been replaced by ψ.

If ϕ has no free variable, we say that it is **closed**. Moreover, we will write $[\![\phi]\!] = [\![\phi]\!]_\rho$ because by the previous lemma, the interpretation of ϕ is independent of the choice of a valuation. For satisfaction, we denote $\sigma \Vdash \phi$, that is, when $\sigma \in [\![\phi]\!]$.

The notion of satisfaction lets us give a nice intuition for the interpretation of formulas. For instance, any word should satisfy the formula $\top$ and we indeed have for any word $\sigma \in (2^{\mathrm{AP}})^\omega$, $\sigma \in [\![\top]\!]$, that is $\sigma \Vdash \bot$. Similarly, no word can satisfy $\bot$. The following list gives the reasoning for the other connectives. For any word σ:[88]

[88] Recall that we do not have to deal with variables $X \in \mathcal{X}$ as we assumed the formula was closed.

- An ω-word satisfies an atomic proposition whenever at the first step, the proposition is true, i.e.: $\sigma \Vdash a \Leftrightarrow a \in \sigma(0)$.
- An ω-word satisfies a conjunction of formulas if it satisfies both formulas, i.e.: $\sigma \Vdash \phi \wedge \psi \Leftrightarrow \sigma \Vdash \phi$ and $\sigma \Vdash \psi$.
- An ω-word satisfies a disjunction of formulas if it satisfies either formula, i.e.: $\sigma \Vdash \phi \vee \psi \Leftrightarrow \sigma \Vdash \phi$ or $\sigma \Vdash \psi$.
- An ω-word satisfies the negation of a formula if it does not satisfy that formula, i.e.: $\sigma \Vdash \neg\phi \Leftrightarrow \sigma \nVdash \phi$.
- An ω-word satisfies the *next* of a formula if it satisfies that formula at the next time step, i.e.: $\sigma \Vdash \bigcirc\phi \Leftrightarrow \sigma\restriction 1 \Vdash \phi$.

Remark 110. The intuition for $\rightarrow$ and $\leftrightarrow$ are missing from this list, but they can easily be recovered from their definition. However, we note that we can use the classical definition of implication (as opposed to the intuitionistic one) because satisfaction of a formula is the belonging to an element of the Boolean algebra on $\mathcal{P}\left((2^{\mathrm{AP}})^\omega\right)$, where classical logic can always be done.

Definition 111. Given ϕ and ψ with all their variables in V, we say that ϕ and ψ are **logically equivalent**, denoted $\phi \equiv \psi$, if for any valuation ρ on V, $[\![\phi]\!]_\rho = [\![\psi]\!]_\rho$.

Example 112. We have the following for any formulas ϕ and ψ.[89]

$$\bigcirc(\phi \wedge \psi) \equiv \bigcirc\phi \wedge \bigcirc\psi \qquad \bigcirc\top \equiv \top$$
$$\bigcirc(\phi \vee \psi) \equiv \bigcirc\phi \vee \bigcirc\psi \qquad \bigcirc\bot \equiv \bot$$
$$\bigcirc\neg\phi \equiv \neg\bigcirc\phi$$

[89] These equivalences can all be easily proven when looking at the definition of the interpretation, or the intuition we have given above.

Remark 113. These formulas are only the important ones using the connective $\bigcirc$, but all the other equivalences proved in classical logic such as DeMorgan's laws, distributivity, etc. can also be shown.

Let us look at how LML relates to observable properties.

Proposition 114. *If ϕ is a closed LML formula, then $[\![\phi]\!] \in \mathcal{P}\left((2^{AP})^\omega\right)$ is clopen.*

Proof. We proceed by structural induction on ϕ. The $\top$ and $\bot$ case are trivial and since clopen sets are closed under binary union and intersection and under complements, the connectives $\wedge$, $\vee$ and $\neg$ are also taken care of.

Case a**:** We have $[\![a]\!] = \bigcup\{\mathrm{ext}(A) \mid A \in 2^{\mathrm{AP}}, a \in A\}$ If AP is finite, then we are done because this is a finite union of clopen set. Otherwise, we need to show that this union is closed. Let σ be such that $a \notin \sigma(0)$, $\mathrm{ext}(\sigma(0))$ is an open set containing σ that does not intersect $[\![a]\!]$.[90]

[90] The case then follows from Lemma **??**.

Case $\bigcirc\phi$**:** By induction hypothesis, $[\![\phi]\!]$ is clopen, thus it is equal to $\mathrm{ext}(W)$ for a finite $W \subseteq (2^{\mathrm{AP}})^*$[91]. Moreover, unrolling the definition of $[\![\cdot]\!]$, we find

[91] By Corollary **??**.

$$[\![\bigcirc\phi]\!] = \bigcup_{A \in 2^{\mathrm{AP}}} \left(\bigcup_{w \in W} \mathrm{ext}(A \cdot w) \right).$$

Hence, if AP is finite, this is a finite union of clopen sets and we are done. Otherwise, we need to show this union is closed. Let $\sigma \notin [\![\bigcirc\phi]\!]$, we have $\sigma\restriction 1 \notin [\![\phi]\!]$. Since $[\![\phi]\!]$ is closed, there exists $u \in (2^{\mathrm{AP}})^*$ such that $\sigma\restriction 1 \in \mathrm{ext}(u)$ and $\mathrm{ext}(u) \cap [\![\phi]\!] = \emptyset$,[92] then $\mathrm{ext}(\sigma(0) \cdot u)$ contains σ and does not intersect $[\![\bigcirc\phi]\!]$. □

[92] We can assume that the open set separating $\sigma\restriction 1$ from $[\![\phi]\!]$ is the extension of a single word u, because if it is $\mathrm{ext}(W)$ for $W \subseteq (2^{\mathrm{AP}})^*$, then we can pick one $u \in W$ such that $\sigma\restriction 1 \in \mathrm{ext}(u)$.

Proposition 115. *If AP is finite, then for any observable property (clopen), there is a closed LML formula ϕ such that $[\![\phi]\!] = P$.*

Proof. When AP is finite, we have seen that there is a finite $U \subseteq (2^{\mathrm{AP}})^*$ such that $P = \mathrm{ext}(U)$. We will show that any $u \in (2^{\mathrm{AP}})^*$, $\mathrm{ext}(u)$ is **definable** by a formula in LML.[93] The result then follows from the fact that P is a finite union of such sets and we can define finite unions with disjunctions of formulas.

First, for any $A \in 2^{\mathrm{AP}}$, we define

$$\phi_A = \left(\bigwedge_{a \in A} a\right) \wedge \left(\bigwedge_{a \notin A} \neg a\right).$$

We can see that $\sigma \in [\![\phi_A]\!]$ must satisfy $a \in \sigma(0) \Leftrightarrow a \in A$, that is $\sigma(0) = A$. We conclude that $\mathrm{ext}(A) = [\![\phi_A]\!]$, so the extensions of single-letter words are definable.

Second, let $u = A_n \cdots A_1 \in (2^{\mathrm{AP}})^*$, we proceed by induction to show that for any $k \leq n$, $A_k \cdots A_1$ is definable by ϕ_k. The base case $k = 0$ is trivial because $\mathrm{ext}(\varepsilon) = [\![\top]\!]$, so $\phi_0 = \top$. Next, suppose we have $[\![\phi_k]\!] = \mathrm{ext}(A_k \cdots A_1)$, then we define $\phi_{k+1} = \bigcirc\phi_k \wedge \phi_{A_{k+1}}$ since we can easily verify that[94]

$$\mathrm{ext}(A_{k+1} \cdot A_k \cdots A_1) = [\![\bigcirc\phi_k \wedge \phi_{A_{k+1}}]\!].$$

□

[93] An LTP is defined by ϕ if $[\![\phi]\!] = P$.

[94] Indeed, an ω-word σ that satisfies this formula must have $\sigma(0) = A_{k+1}$ as argued above and at the next step, it must satisfy ϕ_k. Namely, by induction hypothesis, $\sigma\restriction 1 \in \mathrm{ext}(A_k \cdots A_1)$.

In the following example, we show that the proposition does not always hold when AP is infinite.

Example 116. Let $\mathrm{AP} = \mathbb{N}$ and $A = 2\mathbb{N} \in 2^{\mathrm{AP}}$ (the even numbers). We know that $\mathrm{ext}(A)$ is observable, but there are no formula ϕ with $[\![\phi]\!] = \mathrm{ext}(A)$. Indeed, assume towards a contradiction that such a ϕ exists. Without loss of generality, ϕ has no $\bigcirc$ connective[95], hence we can write ϕ in DNF as

$$\phi = \bigvee_{i \in I} \bigwedge_{j \in J_i} \lambda_{i,j},$$

where the $\lambda_{i,j}$'s are $n \in \mathbb{N}$ or its negation $\neg n$. If $\sigma \Vdash \phi$, then σ must satisfy one of the term in the disjunction, wlog it is the first, so $\sigma \Vdash \bigwedge_{j \in J_1} \lambda_{1,j}$. However, if we let n be the greatest odd number in the $\lambda_{1,j}$'s, the LTP $(\sigma(0) \cup n + 2) \cdot \sigma\restriction 1$ still satisfies the same term, and so ϕ as well. This contradicts the fact that $[\![\phi]\!] = \mathrm{ext}(A)$ as $n + 2 \in \sigma(0)$ is odd.

[95] Indeed, $\mathrm{ext}(A)$ only restricts the first step of the LTPs it contains, so looking at the next steps with $\bigcirc$ is useless. More formally, $\sigma \in \mathrm{ext}(A)$ if and only if $\sigma(0) \cdot \tau \in \mathrm{ext}(A)$ for all $\tau \in (2^{\mathrm{AP}})^\omega$, and if ϕ contains a non-trivial $\bigcirc$ (it is not $\bigcirc\top$), then ϕ will not accept all extensions of $\sigma(0)$.

Note that ϕ is a finite formula, so we can expect it will not be effective to describe something infinitary such as having all even numbers in the first step.

LML with Fixed Points

LML's expressiveness is poor as it only describes *nice* safety properties (the observable ones). In particular, the only liveness property it describes is trivial.[96] In order to remedy that, we will add two modalities: "eventually" ($\Diamond\phi$) and "always" ($\Box\phi$).

The interpretation of these new connectives is[97]

$$[\![\Diamond\phi]\!]_\rho := \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \exists i \in \mathbb{N}, \sigma\restriction i \in [\![\phi]\!]_\rho\}$$

and

$$[\![\Box\phi]\!]_\rho := \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \forall i \in \mathbb{N}, \sigma\restriction i \in [\![\phi]\!]_\rho\}.$$

[96] Recall Proposition **??**.

[97] An ω-word σ satisfies $\Diamond\phi$ if it satisies ϕ at some step.
It satisfies $\Box\phi$ if it satisfies ϕ at all steps.

Example 117. We give a few simple formulas and give the intuition behind their interpretation. Fix $a \in \mathrm{AP}$:

- The property defined by $\Diamond a$ contains all ω-words for which a is true at some point in the word:

$$[\![\Diamond a]\!] = \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \exists i \in \mathbb{N}, a \in \sigma(i)\}.$$

- The property defined by $\Box a$ contains all ω-words for which a is true all the time:

$$[\![\Diamond a]\!] = \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \forall i \in \mathbb{N}, a \in \sigma(i)\}.$$

- The property defined by $\Box\Diamond a$ contains all ω-words for which at any step, a will be true at some point, or equivalently, a is true infinitely many times:

$$[\![\Box\Diamond a]\!] = \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \exists^\infty i, a \in \sigma(i)\}.$$

- The property defined by $\Diamond\Box a$ contains all ω-words for which at some step, a starts to be true forever:

$$[\![\Diamond\Box a]\!] = \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \forall^\infty i, a \in \sigma(i)\}.$$

One can show that the $[\![\Diamond a]\!]$ is an open liveness property, $[\![\Box a]\!]$ is a safety property and both $[\![\Box\Diamond a]\!]$ and $[\![\Diamond\Box a]\!]$ are liveness properties that are neither open nor closed.

Lemma 118. *We have the following logical equivalences:*

$$\Diamond\phi \equiv \neg\Box\neg\phi \qquad \Box\phi \equiv \neg\Diamond\neg\phi$$
$$\Diamond\phi \equiv \phi \vee \bigcirc\Diamond\phi \qquad \Box\phi \equiv \phi \wedge \bigcirc\Box\phi$$

Proof. □

In light of this, one could think of $\Diamond\phi$ as the infinitary property $\bigvee_{n\in\mathbb{N}} \bigcirc^n \phi$. However, syntactically, it is more complex to deal with infinite formulas. In order to use this intuition while avoiding such difficulties, we need to extend the logic with fixed points.[98]

Lemma 119. *Let ϕ be a formula with parameters ρ, $\phi_\Diamond(X) = \phi \vee \bigcirc X$ and $\phi_\Box(X) = \phi \wedge \bigcirc X$, where $X \in \mathcal{X}$ is not in ϕ. Then:*[99]

1. *$[\![\Diamond\phi]\!]_\rho$ is the least fixed point of the function*

$$[\![\phi_\Diamond]\!]_\rho(X) : \mathcal{P}\left((2^{AP})^\omega\right) \to \mathcal{P}\left((2^{AP})^\omega\right) = A \mapsto [\![\phi_\Diamond(X)]\!]_{\rho[A/X]}.$$

2. *$[\![\Box\phi]\!]_\rho$ is the greatest fixed point of the function*

$$[\![\phi_\Box]\!]_\rho(X) : \mathcal{P}\left((2^{AP})^\omega\right) \to \mathcal{P}\left((2^{AP})^\omega\right) = A \mapsto [\![\phi_\Box(X)]\!]_{\rho[A/X]}.$$

[98] Recall that a fixed point x for a function f satisfies $x = f(x)$.

[99] For an LML formula ϕ with parameters ρ and $X \in \mathcal{X}$, we have the following notation:

$$[\![\phi]\!]_\rho(X) = A \mapsto [\![\phi]\!]_{\rho[A/X]}.$$

Proof. 1. First, we can show $[\![\Diamond\phi]\!]_\rho$ is a fixed point because[100]

$$[\![\phi_\Diamond]\!]_\rho([\![\Diamond\phi]\!]_\rho) = [\![\phi \vee \bigcirc\Diamond\phi]\!]_\rho = [\![\Diamond\phi]\!]_\rho.$$

Let P be another fixed point and $\sigma \in (2^{\mathrm{AP}})^\omega$, we claim that if there exists $i \in \mathbb{N}$ such that $\sigma{\restriction}i \in P$, then $\sigma \in P$. Indeed, because $[\![\phi_\Diamond]\!]_\rho(P) \subseteq P$, we infer that

$$\{\sigma \in (2^{\mathrm{AP}})^\omega \mid \sigma{\restriction}1 \in P\} = [\![\bigcirc X]\!]_{\rho[P/X]} \subseteq P.$$

Our claim follows.[101] Moreover, for any $\sigma \in [\![\Diamond\phi]\!]_\rho$, there exists $i \in \mathbb{N}$ such that $\sigma{\restriction}i \in [\![\phi]\!]_\rho \subseteq [\![\phi_\Diamond]\!]_\rho(P) \subseteq P$. Hence, we conclude that $[\![\Diamond\phi]\!]_\rho \subseteq P$.

2. First, similarly to above, $[\![\Box\phi]\!]_\rho$ is a fixed point because

$$[\![\phi_\Box]\!]_\rho([\![\Box\phi]\!]_\rho) = [\![\phi \vee \bigcirc\Box\phi]\!]_\rho = [\![\Box\phi]\!]_\rho.$$

Let P be another fixed point and $\sigma \in P$, we will show that $\sigma{\restriction}i \in [\![\phi]\!]_\rho$ for any $i \in \mathbb{N}$.[102] We proceed by induction on i. Since

$$[\![\phi]\!]_\rho \supseteq [\![\phi]\!]_{\rho[P/X]} \cap [\![\bigcirc X]\!]_{\rho[P/X]} = [\![\phi_\Box]\!]_\rho(P),$$

we have $\sigma \in [\![\phi]\!]_\rho$, covering the base case. Also, we have

$$P \subseteq [\![\bigcirc X]\!]_{\rho[P/X]} = \{\sigma \in (2^{\mathrm{AP}})^\omega \mid \sigma{\restriction}1 \in P\}.$$

In particular $\sigma{\restriction}i \in P \implies \sigma{\restriction}(i+1) \in P$, finishing the induction because we have just proven that any word in P is in $[\![\phi]\!]_\rho$.

□

This result shows how the modal connectives $\Diamond$ and $\Box$ can be described by fixed points of functions defined using only connectives in LML. However, we have skimped on the small detail that the greatest or least fixed points of a function between posets might not always exist.[103] We have to introduce a bit of terminology and two results in order to show that the fixed points in Lemma **??** actually exist.

Definition 120. Let $f : (L, \leq) \to (L, \leq)$, a **pre-fixpoint** of L is an element $a \in L$ such that $f(a) \leq a$. A **post-fixpoint** is an element $a \in L$ such that $a \leq f(a)$. A **fixpoint** (or fixed point) of f is a pre- and post-fixpoint.

Theorem 121 (Knaester-Tarski)**.** [104] *Let $(L, \wedge, \vee, \leq)$ be a complete lattice and $f : L \to L$ be monotone, then*

1. *The least fixpoint of f is $\mu f := \bigwedge\{a \in L \mid f(a) \leq a\}$.*
2. *The greatest fixpoint of f is $\nu f := \bigvee\{a \in L \mid a \leq f(a)\}$.*

Proof. 1. Any fixpoint of f is in particular a pre-fixpoint, thus μf, being a lower bound of pre-fixpoints, is smaller than all fixpoints. Moreover, because for any pre-fixpoint $a \in L$, $f(\mu f) \leq f(a) \leq a$, $f(\nu f)$ is also a lower bound of the pre-fixpoints, so $f(\mu f) \leq \mu f$. Then, $f(f(\mu f)) \leq f(\mu f)$, so $f(\mu f)$ is a pre-fixpoint and $\mu f \leq f(\mu f)$. We conclude that μf is a fixpoint.

[100] The first equality is Lemma **??** (adapted to work with the $\Diamond$ connective) and the second is Lemma **??**.

[101] We have the following implications:

$$\sigma{\restriction}i \in P \Rightarrow \sigma{\restriction}(i-1) \in P \Rightarrow \cdots \Rightarrow \sigma \in P.$$

[102] It then follows that $P \subseteq [\![\Box\phi]\!]_\rho$.

[103] Consider the function $f : (\mathbb{N}, \leq) \to (\mathbb{N}, \leq)$ defined by

$$n \mapsto \begin{cases} n & n \equiv 1 \pmod 2 \\ 0 & n \equiv 0 \pmod 2 \end{cases}.$$

Every odd number is a fixed point, so f clearly has no greatest fixed points.

[104] This is actually a weaker version of the Knaester-Tarski theorem which states that the fixed points of f form a complete lattice.

2. Any fixpoint of f is in particular a post-fixpoint, thus νf, being an upper bound of post-fixpoints, is bigger than all fixpoints. Moreover, because for any post-fixpoint $a \in L$, $a \leq f(a) \leq f(\nu f)$, $f(\nu f)$ is an upper bound of the post-fixpoints, so $\nu f \leq f(\nu f)$. Then, $f(\nu f) \leq f(f(\nu f))$, so $f(\nu f)$ is a post-fixpoint and $f(\nu f) \leq \nu f$. We conclude that νf is a fixpoint.

□

With this result, it is left to show that the functions $[\![\phi_\Diamond]\!]_\rho(X)$ and $[\![\phi_\Box]\!]_\rho(X)$ are monotone (or antimonotone as the greatest fixed point is the least fixed point in the dual poset).

Definition 122. Let ϕ be a formula in LML and $X \in \mathcal{X}$, we define the relations X Pos ϕ and X Neg ϕ inductively as follows:[105]

$$\frac{}{X \text{ Pos } X} \qquad \frac{X \neq Y}{X \text{ Pos } Y} \qquad \frac{a \in \mathrm{AP}}{X \text{ Pos } a} \qquad \frac{}{X \text{ Pos } \top} \qquad \frac{}{X \text{ Pos } \bot} \qquad \frac{X \text{ Pos } \phi}{X \text{ Pos } \bigcirc\phi}$$

$$\frac{X \text{ Pos } \phi \qquad X \text{ Pos } \psi}{X \text{ Pos } \phi \vee \psi} \qquad \frac{X \text{ Pos } \phi \qquad X \text{ Pos } \psi}{X \text{ Pos } \phi \wedge \psi} \qquad \frac{X \text{ Neg } \phi}{X \text{ Pos } \neg\phi}$$

$$\frac{X \neq Y}{X \text{ Neg } Y} \qquad \frac{a \in \mathrm{AP}}{X \text{ Neg } a} \qquad \frac{}{X \text{ Neg } \top} \qquad \frac{}{X \text{ Neg } \bot} \qquad \frac{X \text{ Neg } \phi}{X \text{ Neg } \bigcirc\phi}$$

$$\frac{X \text{ Neg } \phi \qquad X \text{ Neg } \psi}{X \text{ Neg } \phi \vee \psi} \qquad \frac{X \text{ Neg } \phi \qquad X \text{ Neg } \psi}{X \text{ Neg } \phi \wedge \psi} \qquad \frac{X \text{ Neg } \phi}{X \text{ Neg } \neg\phi}$$

[105] The relations are read as "X **positive** in ϕ" and "X **negative** in ϕ" respectively. They intuitively correspond to the fact that X always appears under an even (resp. odd) number of negations in ϕ. If X does not appear in ϕ, then X Pos ϕ and X Neg ϕ.

Example 123. Both the formulas $\phi_\Diamond(X)$ and $\phi_\Box(X)$ we defined in Lemma **??** have X positive in them.

Lemma 124. *Let (ϕ, ρ) be a formula with parameters and $X \in \mathcal{X}$.*

1. *If X is positive in ϕ, then $[\![\phi]\!]_\rho(X)$ is monotone.*
2. *If X is negative in ϕ, then $[\![\phi]\!]_\rho(X)$ is antimonotone.*

Proof. □

Lemma 125. *Let ϕ be a formula with parameters ρ, $X \in \mathcal{X}$ be positive in ϕ and $\psi = \neg\phi[\neg X/X]$, then*[106]

$$\nu[\![\phi]\!]_\rho(X) = \left(\mu[\![\psi]\!]_\rho(X)\right)^c \quad \textit{and dually} \quad \mu[\![\phi]\!]_\rho(X) = \left(\nu[\![\psi]\!]_\rho(X)\right)^c.$$

Proof. First, by definition, we have $[\![\psi]\!]_\rho(A) = \left([\![\phi]\!]_\rho\left(A^c\right)\right)^c$. Moreover, we can write the following derivation[107]

$$\begin{aligned}
(2^{\mathrm{AP}})^\omega \setminus \mu\left([\![\psi]\!]_\rho(X)\right) &= (2^{\mathrm{AP}})^\omega \setminus \bigcap \left\{A \subseteq (2^{\mathrm{AP}})^\omega \mid [\![\psi]\!]_\rho(A) \subseteq A\right\} \\
&= \bigcup \left\{A^c \mid A \subseteq (2^{\mathrm{AP}})^\omega, [\![\psi]\!]_\rho(A) \subseteq A\right\} \\
&= \bigcup \left\{A \subseteq (2^{\mathrm{AP}})^\omega \mid [\![\psi]\!]_\rho(A^c) \subseteq A^c\right\} \\
&= \bigcup \left\{A \subseteq (2^{\mathrm{AP}})^\omega \mid \left([\![\phi]\!]_\rho(A)\right)^c \subseteq A^c\right\} \\
&= \bigcup \left\{A \subseteq (2^{\mathrm{AP}})^\omega \mid A \subseteq [\![\phi]\!]_\rho(A^c)\right\}
\end{aligned}$$

[106] Recall that νf and μf are respectively the least and greatest fixpoints of the function f.

[107] The first equality is the formula for least fixpoints in Theorem **??**. The second equality is an application of one of DeMorgan's laws. The third equality uses the first sentence of the proof, the fourth is just a property of inclusion and complements and the last is the formula now for greatest fixpoints.

$$= \nu[\![\phi]\!]_\rho(X).$$

□

Syntax and Semantics of LTL

Consider a formula θ with X Pos θ, we can write it (modulo logical equivalence) as

$$\theta \equiv \psi \vee \bigvee_{i \in I} (\phi_i \bigwedge_{j \in J_i} \bigcirc^{n_{i,j}} X),$$

where ψ and the ϕ_i's have no occurrence of X. If we assume that $n_{i,j} = 1$ for all $i \in I$, $j \in J_i$, then θ has a simpler form, that is,

$$\theta \equiv \psi \vee \bigvee_{i \in I} (\phi_i \wedge \bigcirc X) \equiv \psi \vee (\phi \wedge \bigcirc X),$$

where ψ and $\phi = \bigvee_{i \in I} \phi_i$ do not contain X. The idea behind LTL is to add fixed points as we did to obtain $\Box$ and $\Diamond$ but only for formulas θ of this form.

Formulas of LTL are recognized by the grammar:

$$\phi, \psi := \top | \bot | a \in \mathrm{AP} | X \in \mathcal{X} | \phi \wedge \psi | \phi \vee \psi | \neg\phi | \bigcirc \phi | \phi U \psi.$$

The interpretation of formulas in LTL are exactly the same as for LML formulas extended with the modal connective $\phi U \psi$ (read ϕ *until* ψ):[108]

$$[\![\phi U \psi]\!]_\rho = \left\{ \sigma \in (2^{\mathrm{AP}})^\omega \mid \exists i \in \mathbb{N}, \sigma\restriction i \in [\![\psi]\!]_\rho \text{ and } \forall j < i, \sigma\restriction j \in [\![\phi]\!]_\rho \right\}.$$

We also extend the notation $\Vdash$ to LTL formulas and we observe that $\sigma \Vdash \phi U \psi$ if and only if there exists $i \in \mathbb{N}$ such that $\sigma\restriction i \Vdash \psi$ and for any $j < i$, $\sigma\restriction j \Vdash \psi$.

[108] Intutively, there is a time i such that σ satisfies ϕ at every step before i and it satisfies ψ at i.

Fixed Points and Defined Modalities

Lemma 126. *Let ϕ and ψ be formulas with parameters ρ and $X \in \mathcal{X}$ not in ϕ and ψ, then $[\![\phi U \psi]\!]_\rho$ is the least fixed point of $[\![\theta]\!]_\rho(X)$ where $\theta = \psi \vee (\phi \wedge \bigcirc X)$.*

Proof. First, it is a fixed point because if $\sigma \Vdash \psi \vee (\phi \wedge \bigcirc(\phi U \psi))$, then either $\sigma \Vdash \psi$ and then $i = 0$ is a witness for $\sigma \Vdash \phi U \psi$, or $\sigma \Vdash \phi \wedge \bigcirc(\phi U \psi)$, and $i + 1$ is the desired witness where i is the witness of $\sigma\restriction 1 \Vdash \phi U \psi$.[109] □

[109] The formal details of this proof are left to the reader.

Thus, we have $[\![\phi U \psi]\!]_\rho = \mu[\![\theta]\!]_\rho(X)$. Moreover, to express greatest fixed points, we have seen that

$$\nu[\![\theta]\!]_\rho(X) = \left(\mu[\![\neg\theta[\neg X / X]]\!]_\rho(X)\right)^c.$$

After some simplifications[110], we can write

$$\mu[\![\neg\theta[\neg X / X]]\!]_\rho(X) = [\![\neg\psi U \neg(\psi \vee \phi)]\!]_\rho.$$

From now on, we will use the notation $\phi W \psi$ (read ϕ *weak until* ψ) as a shorthand:

$$\phi W \psi \equiv \neg\left(\neg\psi U \neg(\psi \vee \phi)\right),$$

and the next lemma follows from our derivations.

[110]

$$\begin{aligned} \neg\theta[\neg X / X] &\equiv \neg\left(\psi \vee (\phi \wedge \bigcirc \neg X)\right) \\ &\equiv \neg\psi \wedge \neg(\phi \wedge \neg \bigcirc X) \\ &\equiv \neg\psi \wedge (\neg\phi \vee \bigcirc X) \\ &\equiv (\neg\psi \wedge \neg\phi) \vee (\neg\psi \wedge \bigcirc X). \end{aligned}$$

Lemma 127. *The property $[\![\phi W \psi]\!]_\rho$ is the greatest fixed point of $[\![\theta]\!]_\rho(X)$ where $\theta = \psi \vee (\phi \wedge \bigcirc X)$.*

Recall now that the formulas that lead to $\Box$ and $\Diamond$ as fixpoints fit in our simple case and it is easy to see (intuitively and with the fixpoint definitions) that

$$\Diamond\phi \equiv \top U \phi \text{ and } \Box\phi \equiv \phi W \bot.$$

Therefore, by defining $\Diamond$ and $\Box$ with these equivalences within LTL, we can recover their original semantics.

Lemma 128. *For any LTL formula ϕ,*[111]

$$[\![\Diamond\phi]\!]_\rho := [\![\top U \phi]\!]_\rho = \{\sigma \in (2^{AP})^\omega \mid \exists i \in \mathbb{N}, \sigma\restriction i \in [\![\phi]\!]_\rho\}$$

$$[\![\Box\phi]\!]_\rho := [\![\phi W \bot]\!]_\rho = \{\sigma \in (2^{AP})^\omega \mid \forall i \in \mathbb{N}, \sigma\restriction i \in [\![\phi]\!]_\rho\}.$$

[111] The proofs are essentially an easy unrolling of the definitions of until and weak until.

In order to motivate the restrictions of fixed points to formulas where $\bigcirc$ only appears once at the leafs, we state the following fact without proof.

Proposition 129. *There are no closed LTL formula ϕ such that $[\![\phi]\!]$ is the greatest fixed point of $\theta(X) = a \wedge \bigcirc \bigcirc X$.*

The greatest fixpoint would be

$$\{\sigma \in (2^{\mathrm{AP}})^\omega \mid \forall i \in \mathbb{N}, a \in \sigma(2 \cdot i)\}.$$

Lemma 130. *For any ϕ and ψ, we have*

$$\begin{aligned} \phi W \psi &\equiv (\phi \cup \psi) \vee \Box\phi \\ \neg(\phi W \psi) &\equiv \neg\psi U \neg(\phi \vee \psi) \\ \neg(\phi U \psi) &\equiv \neg\psi W \neg(\phi \vee \psi) \end{aligned}$$

Remark 131. Using the last two equivalences, we obtain a systematic way to push all the negations to the leaves. Unfortunately, this process leads to an exponential blow up in the size of the formula. For this reason, there are variations on LTL that add a modal connective to make these DeMorgan's law have no blow up (it is called *release*).

Fixed Points and Continuity

Let us go back to studying LML formulas. We would like to get better ways to express fixed points than what is given by Knaester-Tarski. More precisely, let $f : L \to L$ be monotone on a complete lattice $(L, \wedge, \vee, \leq)$. Under certain conditions on f, we can show that

$$\mu f = \bigvee_{n \in \mathbb{N}} f^n(\bot) \quad \text{and} \quad \nu f = \bigwedge_{n \in \mathbb{N}} f^n(\top).$$

We introduce a bit more terminology before getting to this point.

Definition 132. Let $(A, \leq)$ be a poset, a subset $D \subseteq L$ is **directed** if it is non-empty and for any $a, b \in D$, there exists $c \in D$ such that $a, b \leq c$. A subset $D \subseteq L$ is **codirected** if it is directed in the opposite order.[112]

[112] Namely, it is non-empty and for any $a, b \in D$, there exists $c \in D$ such that $c \leq a, b$.

Lemma 133. *Let $f : (A, \leq) \to (B, \preceq)$ be a monotone function of posets. If $D \subseteq A$ is (co)directed, then $f(D)$ is also (co)directed.*[113]

[113] It follows trivially that when f is anti-monotone, the image of a directed set is codirected and vice-versa.

Proof. Let $d, d' \in f(D)$, there exists $a, b \in D$ such that $f(a) = d$ and $f(b) = d'$. Hence, if D is directed (resp. codirected), then there exists $c \in D$ such that $a, b \leq c$ (resp. $c \leq a, b$) and by monotonicity, $f(a), f(b) \leq f(c)$ (resp. $f(c) \leq f(a), f(b)$) with $f(c) \in f(D)$, so $f(D)$ is directed (resp. codirected). □

Definition 134. Let L and L' be two complete lattice,[114] a function $f : L \to L''$ is **continuous** (also Scott-continuous) if it is monotone and for any directed $D \subseteq L$, $f(\bigvee D) = \bigvee f(D)$. It is **cocontinuous** if it is monotnoe and for any codirected $C \subseteq L$, $f(\bigwedge C) = \bigwedge f(C)$.

[114] We will overload the notations $\leq$, $\wedge$, $\vee$, $\top$ and $\bot$ to mean the usual things in both L and L' while making it clear in which sets elements live.

Definition 135. A complete lattice L is a **frame** if for any $a \in L$ and $S \subseteq L$, $a \wedge \bigvee S = \bigvee\{a \wedge s \mid s \in S\}$.

Lemma 136. *Let $(D, \subseteq)$ be a directed poset, $(L, \leq)$ be a frame and $f, g : D \to L$ be monotone, then*

$$\left(\bigvee_{d \in D} f(d)\right) \vee \left(\bigvee_{d \in D} g(d)\right) = \bigvee_{d \in D} f(d) \vee g(d),$$

and

$$\left(\bigvee_{d \in D} f(d)\right) \wedge \left(\bigvee_{d \in D} g(d)\right) = \bigvee_{d \in D} f(d) \wedge g(d)$$

Proof. □

Corollary 137. *Let X be a set and $(D, \leq)$ a poset:*

- *If $(D, \leq)$ is directed and $f, g : D \to \mathcal{P}(X)$ are monotone, then we have the same thing as above but with unions and intersections as well as the duals.*

Proof. □

Proposition 138. *If $f : L \to L$ is monotone, then*

- *If f is continuous, then $\mu f = \bigvee_{n \in \mathbb{N}} f^n(\bot)$.*
- *If f is cocontinuous, then $\nu f = \bigwedge_{n \in \mathbb{N}} f^n(\top)$.*

Proof. □

Let ϕ be an LML formula with parameters ρ and $X \in \mathcal{X}$ such that X Pos ϕ, we can show that $[\![\phi]\!]_\rho(X)$ is continuous and cocontinuous. Therefore, we have

$$[\![\Diamond\phi]\!]_\rho = \cup_{n \in \mathbb{N}} [\![\bigcirc^n \phi]\!]_\rho,$$

and dually,

$$[\![\Box\phi]\!]_\rho = \cap_{n \in \mathbb{N}} [\![\bigcirc^n \phi]\!]_\rho.$$

ω-Regular Properties

We will introduce a notion similar to regular languages (the ones recognized by a DFA) that applies to linear time properties, namely, sets of ω-words.

Definition 139. Let Σ be a finite alphabet an **ω-regular expression** on Σ has the form

$$G = E_1 \cdot F_1^\omega + \cdots + E_k \cdot F_k^\omega,$$

where the E_i's and F_i's are regular expressions on Σ such that $\varepsilon \notin \mathcal{L}(F_i)$ for any i.

The language recognized by G is

$$\mathcal{L}_\omega(G) := \left\{\sigma \in \Sigma^\omega \mid \exists i \leq k, u \in \mathcal{L}(E_i), \{v_j\}_{j\in\mathbb{N}} \subseteq \mathcal{L}(F_i), \sigma = uv_0 \cdots v_n \cdots\right\}.$$

A language $L \subseteq \Sigma^\omega$ is ω-regular if $L = \mathcal{L}_\omega(G)$ for an ω-regular expression G.

Example 140. Let $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

- The language L_Π that satisfies $\sigma \in L_\Pi \Leftrightarrow \exists^\infty t, \sigma(t) = \mathtt{a}$ is given by $\mathcal{L}_\omega((\mathtt{b}*\mathtt{a})^\omega)$.
- The language L_Σ that satisifes $\sigma \in L_\Pi \Leftrightarrow \forall^\infty t, \sigma(t) = \mathtt{b}$ is given by $\mathcal{L}_\omega(\Sigma^* \cdot \mathtt{b}^\omega)$.

Remark 141. The set of ω-regular languages on a fixed alphabet are clearly closed under finite union because $\mathcal{L}_\omega(G) \cup \mathcal{L}_\omega(G') = \mathcal{L}_\omega(G + G')$. We will also see that they are closed under finite intersection and complementation. We will also see that any closed LTL formula defines an ω-regular language.