1 Tutorial Problems

1.1 LTL Equivalences and Counterexamples

a) **Prove or disprove** \(\Diamond(\phi \lor \psi) \equiv \Diamond\phi \lor \Diamond\psi\).

**If it is not true, provide a counterexample.**

Fixing some set of atomic predicates \(AP\), by definition, we have \(\Diamond(\phi \lor \psi) \equiv \Diamond\phi \lor \Diamond\psi\) if and only if \(\{\sigma \in (2^{AP})^\omega \mid \sigma \vDash (\phi \lor \psi)\} = \{\sigma \in (2^{AP})^\omega \mid \sigma \vDash (\Diamond\phi \lor \Diamond\psi)\} \).

Let \(\sigma \in (2^{AP})^\omega\). By extensionality, we can prove this LTL equivalence if we can prove

\[
\sigma \vDash (\phi \lor \psi) \iff \sigma \vDash (\Diamond\phi \lor \Diamond\psi)
\]

The reverse direction is true:

Suppose \(\sigma \vDash (\Diamond\phi \lor \Diamond\psi)\).

Case 1: \(\sigma \vDash \Diamond\phi\).

Then for all \(i\), \(\sigma[i..] \vDash \phi\).

Let \(j \in \mathbb{N}\) be arbitrary. By specialization, \(\sigma[j..] \vDash \phi\). And hence \(\sigma[j..] \vDash \phi \lor \psi\).

By generalization of \(j\), \(\sigma \vDash (\phi \lor \psi)\)

Case 2: symmetric argument of case 1.

However, when you try to prove the forward direction, you get stuck.

Suppose \(\sigma \vDash (\Diamond(\phi \lor \psi))\). Then for all \(i\), \(\sigma[i..] \vDash \phi \lor \psi\). I.e., at least one of \(\phi\) or \(\psi\) is true at each time step. But in order to prove the conclusion, we need to know that a specific disjunct is true at all time steps. This is not the same thing!

As a counterexample, take \(AP = \{a, b\}, \phi = a, \psi = b, \) and \(\sigma = (\{a\}\{b\})^\omega\).

I.e., \(\sigma[i..] \vDash a\) when \(i\) is even, and \(\sigma[i..] \vDash b\) when \(i\) is odd. Clearly, \(\sigma \vDash (\Diamond a \lor \Diamond b)\).

We don’t have \(\sigma \vDash \Diamond a\), since (for example), \(\sigma[1..] \nvdash a\).

We also don’t have \(\sigma \vDash \Diamond b\), since (for example), \(\sigma[0..] \nvdash b\).

So \(\sigma \nvdash \Diamond a \lor \Diamond b\).
b) **Prove or disprove** \( \phi U \psi \equiv \psi \lor (\phi \land \Box (\phi U \psi)) \).

**If it is not true, provide a counterexample.**

Fixing some set of atomic predicates \( AP \), by definition, we have \( \phi U \psi \equiv \psi \lor (\phi \land \Box (\phi U \psi)) \) if and only if \( \{ \sigma \in (2^{AP})^\omega \mid \sigma \models \phi U \psi \} = \{ \sigma \in (2^{AP})^\omega \mid \sigma \models \psi \lor (\phi \land \Box (\phi U \psi)) \} \).

Let \( \sigma \in (2^{AP})^\omega \). By extensionality, we can prove this LTL equivalence if we can prove

\[
\sigma \models \phi U \psi \iff \sigma \models \psi \lor (\phi \land \Box (\phi U \psi))
\]

This one is true.

**Forward Direction.**

Suppose \( \sigma \models \phi U \psi \).

Then there exists some \( k \in \mathbb{N} \) such that \( \sigma[k..] \models \psi \) and for all \( i < k \), \( \sigma[i..] \models \phi \).

Let’s write these premises down.

A) \( \sigma[k..] \models \psi \)
B) \( \forall i < k, \sigma[i..] \models \phi \)

We proceed by case analysis on \( k \).

Case 1: \( k = 0 \). And so \( \sigma \models \psi \), by premise A.

But then \( \sigma \models \psi \lor (\phi \land \Box (\phi U \psi)) \), so we are done.

Case 2: \( k = k' + 1 \) for some \( k' \in \mathbb{N} \).

You can probably imagine that we’re going to prove the right disjunct.

I.e. we need to prove \( \sigma \models \phi \land \Box (\phi U \psi) \).

We begin with the left conjunct.

Note that our premise B becomes \( \forall i < k' + 1, \sigma[i..] \models \phi \).

In particular, this means that we have \( \sigma[0..] \models \phi \), since \( \forall m \in \mathbb{N}, 0 < m + 1 \). So we have \( \sigma \models \phi \).

To see how we prove the right conjunct, consider its semantics.

\[
\sigma \models \Box (\phi U \psi) \iff \sigma[1..] \models \phi U \psi \\
\iff \exists m \in \mathbb{N}, \sigma[1..][m..] \models \psi \land \forall j < m, \sigma[1..][j..] \models \phi.
\]

What should we use as the existential witness \( m \)? It’s \( k' \). I.e. one less than our original index \( k \).

So we need to prove first \( \sigma[1..][k'].. \models \psi \) – but this just means \( \sigma[k' + 1..] \models \psi \), which is our premise A.

Similarly the other proof obligation is \( \forall j < k', \sigma[1..][j..] \models \phi \). Or, \( \forall j < k', \sigma[j + 1..] \models \phi \).

So, let \( j \in \mathbb{N} \) such that \( j < k' \). Then, \( j + 1 < k' + 1 \). By specialization and Modus Ponens on premise B, we have \( \sigma[j + 1..] \models \phi \). So we are done.

**Reverse Direction.**

Let \( \sigma \in (2^{AP})^\omega \) and suppose \( \sigma \models \psi \lor (\phi \land \Box (\phi U \psi)) \).

By the semantics of \( (\lor) \), we have either \( \sigma \models \psi \) or \( \sigma \models \phi \land \Box (\phi U \psi) \).
We proceed by case analysis on this disjunction.

Case 1. Assume $\sigma \models \psi$. We need to prove

$$\sigma \models \phi \mathcal{U} \psi \iff \exists k \in \mathbb{N}, \sigma[k..] \models \psi \land \forall i < k, \sigma[i..] \models \phi$$

To prove this, we use $k = 0$ as the witness, and our assumption of $\phi \vdash \psi$ gives us the left conjunct. There are no $i \in \mathbb{N}$ such that $i < 0$, so the right conjunct is vacuously true.

Case 2. Assume $\sigma \models \phi \land \bigcirc (\phi \mathcal{U} \psi)$. The rest of the proof is very similar in spirit to the forward direction:

1. $\sigma \models \phi \land \bigcirc (\phi \mathcal{U} \psi)$ (Assumption)
2. $\sigma \models \bigcirc (\phi \mathcal{U} \psi)$ (1, right conjunct)
3. $\sigma[1..] \models \phi \mathcal{U} \psi$ (2, Def. $\models \bigcirc$)
4. $\exists k \in \mathbb{N}, \sigma[1..][k..] \models \psi \land \forall i < k, \sigma[1..][i..] \models \phi$ (3, Def. $\models \mathcal{U}$)
5. $\exists k \in \mathbb{N}, \sigma[k+1..] \models \psi \land \forall i < k, \sigma[i+1..] \models \phi$ (4, algebra of $\omega$-words)
6. $\forall i < k+1, \sigma[i..] \models \phi$ (5, right conjunct, algebra of $\omega$-words)
7. Let $k' = k + 1$
8. $\forall i < k', \sigma[i..] \models \phi$ (6,7, substitution)
9. $\sigma[k'..] \models \psi$ (5, left conjunct, 7, substitution)
10. $\sigma[k'..] \models \psi \land \forall i < k', \sigma[i..] \models \phi$ (8, 9, Def $\models \land$)
11. $\exists k', \sigma[k'..] \models \psi \land \forall i < k', \sigma[i..] \models \phi$ (10, construction)
12. $\sigma \models \phi \mathcal{U} \psi$ (11, Def $\models \mathcal{U}$)
c) **Prove or disprove** □□(φ ∨ ¬ψ) ≡ ¬◊(¬φ ∧ ψ).

First, let’s do some rewriting. Let σ ∈ (2^{AP})ω.

\[ \sigma \models ¬◊(¬φ ∧ ψ) \iff \sigma \not\models ◊(¬φ ∧ ψ) \]
\[ \iff \exists k, σ[k..] \not\models ¬φ ∧ ψ \]
\[ \iff \forall k, σ[k..] \not\models ¬φ ∧ ϕ \]
\[ \iff \forall k, σ[k..] \not\models ¬(φ ∨ ¬ψ) \]
\[ \iff \forall k, σ[k..] \models ¬¬(φ ∨ ¬ψ) \]
\[ \iff \sigma \models □(φ ∨ ¬ψ) \]

**Forward Direction.**

Assume ∀i, ∀j, σ[i..][j..] ⊨ φ ∨ ¬ψ.

We need to show that ∀k, σ[k..] ⊨ φ ∨ ¬ψ

Let k be arbitrary.

By specialization of our assumption, ∀j, σ[k..][j..] ⊨ φ ∨ ¬ψ.

Specializing again with j = 0, we have σ[k..][0..] = σ[k + 0..] = σ[k..] ⊨ φ ∨ ¬ψ. By generalization and the semantics of □, we have σ ⊨ □(φ ∨ ¬ψ).

**Reverse Direction.**

Assume that ∀k, σ[k..] ⊨ φ ∨ ¬ψ.

We need to show that ∀i, ∀j, σ[i..][j..] ⊨ φ ∨ ¬ψ.

Let i, j both be arbitrary. By specialization of our assumption with k = i + j, we have σ[i + j..] ⊨ φ ∨ ¬ψ. Equivalently, this means we have σ[i..][j..] ⊨ φ ∨ ¬ψ. By generalization twice and the semantics of □, we have σ ⊨ □□(φ ∨ ¬ψ).
1.2 LTL Formalization

**Client-Server:** There are three clients, and the server has a priority system. If Client 1 issues a request, Clients 2 and 3 will not receive answers until Client 1 is answered. Likewise, if Client 2 sends a request, Client 3 will not receive an answer until client 2 is answered.

\[ \square(req_1 \implies \neg(ans_2 \lor ans_3) \cup ans_1) \land \square(req_2 \implies \neg ans_3 \cup ans_2) \]

**Client-Server:** There are two clients. It is guaranteed that both clients will issue exactly one request. The server will answer the clients in the reverse order in which the requests were received. Assume only one event can happen in a given time-step.

\[ \Diamond(req_1 \land \square \Diamond(req_2 \land \square \Diamond(ans_2 \land \square \Diamond(ans_1)))) \lor \\
\Diamond(req_2 \land \square \Diamond(req_1 \land \square \Diamond(ans_1 \land \square \Diamond(ans_2)))) \]

**Traffic-light:** Uh, oh! The traffic light has been damaged! When this happens, it has been programmed to flash green, then yellow, then alternate between red and yellow forever. Each colour lasts one time-step.

\[ g \land \Diamond y \land \square(y \implies \Diamond r \land r \implies \Diamond y) \]