FORMALIZATION AND VERIFICATION OF PROPERTY SPECIFICATION PATTERNS

by

Dmitriy Bryndin

A THESIS

Submitted
to Michigan State University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

COMPUTER SCIENCE

2010

ABSTRACT

FORMALIZATION AND VERIFICATION OF PROPERTY SPECIFICATION PATTERNS

by

Dmitriy Bryndin

Finite-state verification (FSV) techniques are intended for proving properties of software systems. Although significant progress has been made in the last decade automating FSV techniques, the adoption of these techniques by software developers is low. The Specification Pattern System (SPS) is intended to assist users in creating such specifications. It identifies common specification patterns and indicates how to translate the patterns into a variety of different specification languages. However, the patterns in the SPS are defined informally and their translations are not verified. This work discusses the informal nature of these definitions, proposes a formalization for them and provides formal proofs for the translation of patterns to Linear Temporal Logic.

ACKNOWLEDGEMENTS

I want to thank my committee members: Dr. Laura Dillon, Dr. Eric Torng and Dr. William F. Punch. This thesis could not have been written without their support. I really appreciate their time. Especially, I want to thank Laurie Dillon for being patient and guiding me all these years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Property specifications are intended to be used in software development to describe different parts of a system behavior. They can help to detect design flaws in early stages of development, serve as a reference for programmers in later stages of software development, and be used for the verification of an implementation. Being developed by humans, the initial specifications are never formal [12]. Developers usually represent initial specifications graphically or using the natural language [13]. Property specifications often stay in this form for the rest of the development process. While still useful, these informal specifications are often ambiguous and inconsistent with the actual system's behavior and the cost of the related errors is high, as they are usually detected on the later stages of development.

When property specifications are formalized, developers obtain precise specifications, which can be formally analyzed for consistency, completeness, and other desirable (or absence of undesirable) properties. Tools automating such analysis typically require some special types of formalisms, such as temporal logic [4]. The use of these formalisms requires expertise and significantly limits adoption of FSV techniques by the developers.

1.1 Specification Pattern System

M.Dwyer et al. in [6, 7] proposed an approach that helps developers in mapping informal property specifications to the formalisms accepted by a variety of automated verification tools. Similarly to the idea of Design Patterns [9], M.Dwyer et al. came up with a set of parameterized patterns that are independent of the formalisms used in the verification tools. These patterns were obtained from a survey of commonly occurring properties that users verify with the finite-state verification tools, such as SPIN [11], SMV [16], CWB-NC, INCA [5] and FLAVERS [8].

Finite-state verification tools model a system execution by a (possibly event driven) finite-state machine. A property pattern restricts some aspect of system behavior, namely the occurrence of some state/event or the order in which multiple state/events occur in the execution. For example, to say that a deadlock never occurs, we use the "Absence" pattern, or to say that a thread has to eventually release memory that has been dynamically allocated to it, we use the "Response" pattern. These patterns are described in greater details in Section 2.1.

Each pattern is associated with a scope that describes a sequence of states/events over which the restrictions imposed by the pattern apply. A scope can define the entire execution of a system, a part between the moment when a thread is created and the moment when it is terminated, etc. Scopes are described in Section 2.2.

The crucial part of the work done by M.Dwyer et al. in [1] is a translation of pairs of patterns and scopes to the following specification formalisms, which are used by different verification tools: Linear Temporal Logic [14], Computation Tree Logic [3], Quantified Regular Expressions [19] and INCA Queries [5]. Other researchers have developed mappings for: Action Computation Tree Logic [18], Graphical Interval Logic [20] and Regular Alternation-Free Mu-Calculus [15].

1.2 Linear Temporal Logic

Linear Temporal Logic (LTL) [14] is commonly used in software verification to specify and reason about behaviors that are modeled as linear state sequences, with states denoting finite sets of propositions (the propositions that are true). There are several model checkers available that support LTL: SPIN [11], Java PathFinder [24], NuSMV [2] and others.

Denote a set of atomic propositions by $\mathtt{A}$. A *state* gives an interpretation to propositions in $\mathtt{A}$. A *state formula* is a formula in ordinary first-order logic over the propositions in $\mathtt{A}$. We use $s \Vdash P$ to say that P holds on S or that s is a P-state. LTL extends ordinary predicate logic with a set of temporal operators presented in Table 1.1.

$\Box P$	Henceforth P
$\Diamond P$	Eventually P
$P \mathcal{U} Q$	P Until Q
$P \mathcal{W} Q$	P Waiting-for Q
$\bigcirc P$	Next P

Table 1.1: LTL, Future Temporal Operators.

An *LTL formula* is defined as

- each proposition in $\mathtt{A}$ is an LTL formula
- if P and Q are LTL formulas, $\neg P$, $P \wedge Q$, $P \vee Q$, $P \rightarrow Q$, $P \leftrightarrow Q$, $\Box P$, $\Diamond P$, $P \mathcal{U} Q$, $P \mathcal{W} Q$, $\bigcirc P$ are also LTL formulas.

In LTL formulas we use $\rightarrow$ to denote implication and $\leftrightarrow$ to denote equivalence, using the more common $\Longrightarrow$ and $\Longleftrightarrow$ as meta notation.

LTL formulas are interpreted over a *model*, which is an infinite sequence of states, $\sigma : \ s_0, s_1, \ldots$. We write $(\sigma, i) \vDash P$ to say that P holds at a position $i \geq 0$ in σ, for a given model σ and an LTL formula P. For a state formula P, $(\sigma, i) \vDash P \iff s_i \Vdash P$.

For the LTL formulas P and Q:

- $(\sigma, i) \vDash \neg P \iff \neg (\sigma, i) \vDash P$.
- $(\sigma, i) \vDash P \vee Q \iff (\sigma, i) \vDash P \vee (\sigma, i) \vDash Q$, with $\wedge, \rightarrow$ and $\leftrightarrow$ defined similarly.
- $(\sigma, i) \vDash \Box P \iff \forall k \geq i \bullet (\sigma, k) \vDash P$.
- $(\sigma, i) \vDash \Diamond P \iff \exists k \geq i \bullet (\sigma, k) \vDash P$.
- $(\sigma, i) \vDash P \mathcal{U} Q \iff \exists k \geq i \bullet ((\sigma, k) \vDash Q \ \wedge \ \forall j : i \leq j < k \bullet (\sigma, j) \vDash P)$.
- $(\sigma, i) \vDash P \mathcal{W} Q \iff (\sigma, i) \vDash P \mathcal{U} Q \ \vee \ (\sigma, i) \vDash \Box P$.
- $(\sigma, i) \vDash \bigcirc P \iff (\sigma, i + 1) \vDash P$.

1.3 Structure of this work

In this work we check for inconsistencies between the informal definitions of patterns and scopes in the Specification Pattern System proposed in [6, 7] and their translations to LTL, presented in [1]. We give a formal interpretation for these definitions, removing all inconsistencies and ambiguities. Finally, the largest part of this work provides proofs of the equivalence between the pattern/scope combinations and their translations to LTL. Because this work only considers the translation to LTL, we consider only the state-based definitions presented in [1].

Chapter 2 provides the definitions of patterns and scopes, as they are given in [6, 7] and highlights several problems related to the informal nature of these definitions.

In Chapter 3, we formalize the original informal definitions of scopes and patterns and discuss their translations to LTL.

Appendix A is the reference for the original LTL formulas in [1].

Appendix B contains all proofs for the translations to LTL.

Appendix C provides counterexamples for the cases when our proposed LTL formulas are not equivalent to the original formulas in [1].

Chapter 2

INFORMAL DEFINITIONS

2.1 Informal Definitions of Patterns

Figure 2.1 illustrates the hierarchy of patterns. Figure 2.2 provides the explanations for the two major groups of patterns, classified by the system behaviors. Figures 2.3 and 2.4 describe the intent of all patterns. The contents of all figures are reproduced from [1].

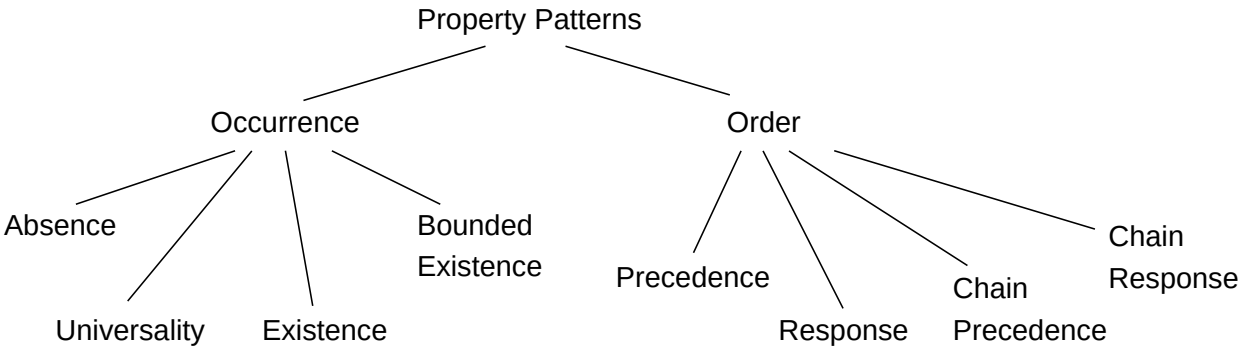

Figure 2.1: Classification of patterns in terms of system behaviors, as it appears in [1]

Occurrence Patterns talk about the occurrence of a given event/state during system execution.
Order Patterns talk about relative order in which multiple events/states occur during system execution.

Figure 2.2: Classification of patterns, as it appears in [1]

2.2 Informal Definitions of Scopes

Figure 2.5 illustrates the definitions of scopes, provided in Figure 2.6. The content of both figures is reproduced exactly as in [1].

Absence
To describe a portion of a system's execution that is free of certain events or states. Also known as Never.
Universality
To describe a portion of a system's execution which contains only states that have a desired property. Also known as Henceforth and Always.
Existence
To describe a portion of a system's execution that contains an instance of certain events or states. Also known as Eventually
Bounded Existence
To describe a portion of a system's execution that contains at most a specified number of instances of a designated state transition or event.

Figure 2.3: Intent of occurrence patterns, as it appears in [1]

Precedence
To describe relationships between a pair of events/states where the occurrence of the first is a necessary pre-condition for an occurrence of the second. We say that an occurrence of the second is enabled by an occurrence of the first.
Response
To describe cause-effect relationships between a pair of events/states. An occurrence of the first, the cause, must be followed by an occurrence of the second, the effect. Also known as Follows and Leads-to.
Chain patterns are used to express requirements related to complex combinations of individual state/event relationships. These include precedence/response relationships consisting of sequences of individual states/events. We call these chain patterns.
Chain Precedence
This is a scalable pattern. We describe the 1 cause - 2 effect version here.
To describe a relationship between an event/state P and a sequence of events/states (S, T) in which the occurrence of S followed by T within the scope must be preceded by an occurrence of the the sequence P within the same scope. In state-based formalisms, the beginning of the enabled sequence (S, T) may be satisfied by the same state as the enabling condition (i.e., P and S may be true in the same state).
Chain Response
This is a scalable pattern. We describe the intent of the 1 stimulus - 2 response version here.
To describe a relationship between a stimulus event (P) and a sequence of two response events (S,T) in which the occurrence of the stimulus event must be followed by an occurrence of the sequence of response events within the scope. In state-based formalisms, the states satisfying the response must be distinct (i.e., S and T must be true in different states to count as a response), but the response may be satisfied by the same state as the stimulus (i.e., P and S may be true in the same state).

Figure 2.4: Intent of order patterns, as it appears in [1]

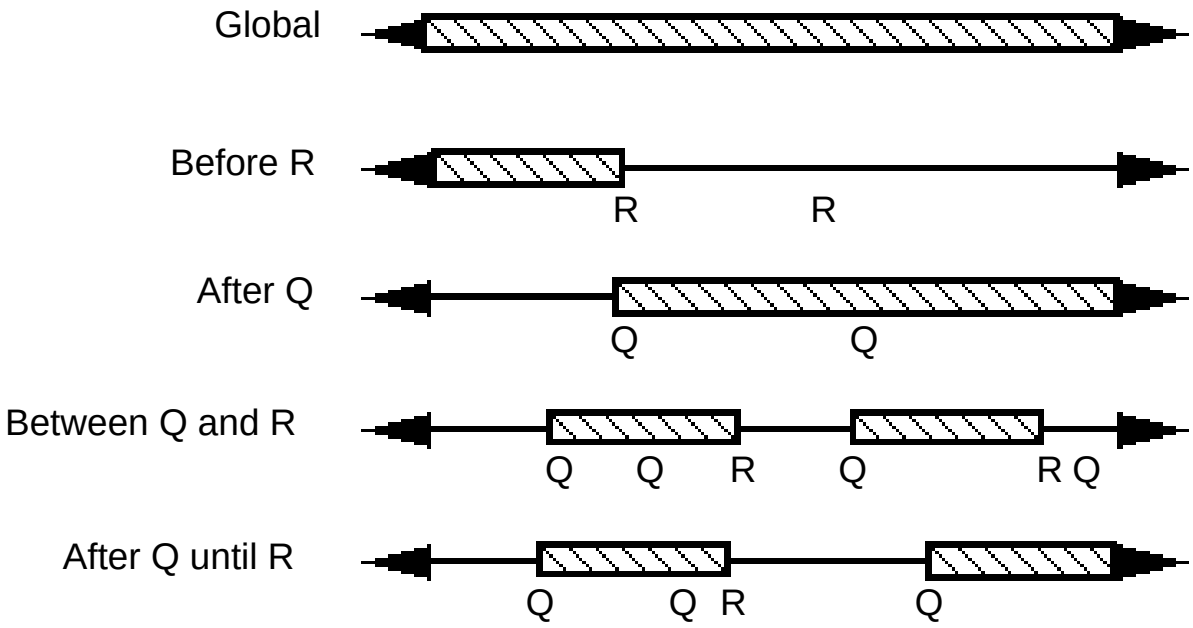

Figure 2.5: The illustration of scopes, as it appears in [1]

Each pattern has a scope, which is the extent of the program execution over which the pattern must hold. There are five basic kinds of scopes: global (the entire program execution), before (the execution up to a given state/event), after (the execution after a given state/event), between (any part of the execution from one given state/event to another given state/event) and after-until (like between but the designated part of the execution continues even if the second state/event does not occur). The scope is determined by specifying a starting and an ending state/event for the pattern: the scope consists of all states/events beginning with the starting state/event and up to but not including the ending state/event.

We note that a scope itself should be interpreted as optional; if the scope delimiters are not present in an execution then the specification will be true.

Figure 2.6: The definitions of scopes, as they appear in [1]

2.3 Problems with the Interpretation

The sections 2.1 and 2.2 provide informal definitions of patterns and scopes, as they are presented in [1]. But are these definitions good? According to [12], good definitions must be consistent and unambiguous. Unfortunately, informal definitions tend to give rise to

ambiguity and inconsistency. We illustrate some ambiguities in the informal definitions provided for the pattern system and use the translations to LTL presented in [1] to motivate more precise definitions, which we then formalize in Chapter 3.

We start from the definitions of scopes, listed in Section 2.2.

2.3.1 Multiple delimiters in "Before" and "After" scopes

The first question arises from the "Before R" scope. From it's definition, the given state R is the right delimiter for this scope. It is not immediately clear which R-state is the delimiter for this scope, in case there are several R-states present in the execution. The illustration in Figure 2.5 suggests that the scope goes up to the first R-state. The constructions in the corresponding LTL formulas

$$\ldots \mathcal{U} R \quad \text{or} \quad \ldots \mathcal{W} R$$

suggest that this interpretation is correct.

Therefore, we refine the informal definition of the "Before R" scope to

Before: from the beginning of the execution up to the first occurrence of a given state.

We treat the case of multiple Q-states in "After Q" scope similarly. The illustration in Figure 2.5 suggests that the first occurrence of the Q-state serves as the left delimiter. Indeed, the constructions

$\Box\,(Q \rightarrow \ldots)$,	for Absence, Universality, Response and Chain Response
$\Box\,(\neg Q) \vee \Diamond (Q \wedge \ldots)$,	for Existence and Precedence
$\Diamond\, Q \rightarrow (\neg Q) \mathcal{U} (Q \wedge \ldots)$	for Bounded Existence
$\Box\,(\neg Q) \vee (\neg Q) \mathcal{U} (Q \wedge \ldots)$	for Precedence Chain

in the LTL formulas are consistent with the interpretation that the restrictions on the system execution start to apply from the first occurrence of a Q-state (inclusive). Therefore, we refine the informal definition of the "After Q" scope to

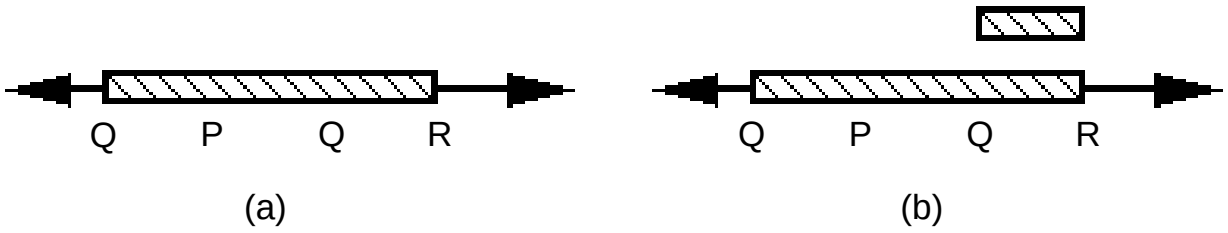

Figure 2.7: Possible interpretations of "Between Q and R" scopes.

After: from the first occurrence of a given state until the end of the execution.

2.3.2 Multiple delimiters in "Between" and "After-Until" scopes

The "After-Until" scope is an unambiguously extended version of "Between" scope and, therefore, everything that we clarify for the "Before" scope also applies to the "After-Until" scope.

Consider the "Between Q and R" scope. The illustration in Figure 2.5 suggests that this scope consists of "maximal" intervals, where each interval ends with an R-state and starts with a Q-state that is the farthermost from this R-state, while not including other R-states. In the case of several R-states after a Q-state, the constructions

$$\Box(Q \wedge \ldots \rightarrow \ldots \mathcal{U} R) \quad \text{or} \quad \Box\,(Q \rightarrow (\neg R)\mathcal{W}(\ldots \wedge \neg R)$$

of LTL formulas are consistent with saying that the restrictions apply only until the closest R.

The case of several Q-states is slightly different. For all patterns except "Existence", the constructions of LTL formulas

$$\Box(Q \wedge \ldots \rightarrow \ldots)$$

agree with the illustration.

$$\text{Existence of } P\text{, Between } Q \text{ and } R\text{:} \qquad \Box\,(Q \wedge \neg R \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R)) \tag{2.1}$$

Consider the example, shown in Figure 2.7a. The informal definition of the "Existence" pattern holds on this "maximal" interval, however, the corresponding LTL formula (2.1) fails.

The construction

$$\Box(Q \wedge \ldots \rightarrow \ldots)$$

implies that each Q-state starts the interval, as it is shown in Figure 2.7b, and for the "Existence" pattern it is not a "maximal", but a "minimal" interval. In this case the informal definition is ambiguous, and the provided illustration fails to clarify it. Instead we propose the following definitions

Between Q **and** R scope consists of all intervals that start with a Q-state and extends to the next R-state.

After Q **Until** R extends "Between" scope with all suffixes that begin with a Q-state and have no subsequent R-states.

2.3.3 Empty intervals

According to the definitions of scopes, the left delimiter is included in and the right delimiter is excluded from the scope. In the case the delimiters are not present, the specification is vacuously *True*. The natural question here is how we treat empty intervals in the scopes. The "Global" scope is never empty. The "After Q" scope is not empty if there is a Q-state in the execution.

Consider the scope "Before R", when the initial state in the execution is an R-state. The initial state is not included in the scope and, therefore, the scope consists of an empty interval. The specification is not vacuously true, as the delimiter R is actually present. With the "Before R" scope, the LTL formulas for all patterns except the "Existence" has the form

$$\ldots \mathcal{U} R$$

which is *True*. However, the LTL for the "Existence" pattern

$$(\neg R)\mathcal{W}(P \wedge \neg R)$$

is *False*.

For the "Between Q and R" and "After Q until R" scopes, it is possible to have an empty interval when a state is both a Q-state and an R-state. The LTL formulas for these scopes have the forms

$\Box\,(Q \rightarrow \ldots \mathcal{U} R)$	for the Bounded Existence and Chain patterns
$\Box\,(Q \wedge \neg R \rightarrow \ldots)$	for the rest of the patterns

It is easy to see that all of them are *True* on the $(Q \wedge R)$ empty interval.

The last two examples show some ambiguity in the interpretation of an empty interval for the "Existence" pattern with the "Before" and the "Between" scopes.

If we treat the empty interval as a part of a scope, we expect the "Existence" pattern to fail. This interpretation is consistent with all scopes except "Between" and "After-Until" scopes for the "Existence" pattern. If we assume that an empty interval is not a part of a scope, then having no other states in the scope results the "Existence" pattern being vacuously true, by the informal definition of scopes. This interpretation is consistent with all scopes, except the "Before" scope for the "Existence" pattern.

As there is no reasonable argument against any of these interpretations of scopes, we propose a formalization for each of them.

Chapter 3

PROPOSED FORMALIZATION

3.1 Formal Definitions of Strong Scopes

We use commas between predicates in the meaning of conjunctions, to increase the readability of formulas. We assume placeholders P, Q, R and S are state formulas.

A *scope* $\mathbb{S}$ is relative to a given model of execution $\sigma : s_0, s_1, s_2, \ldots$. We represent a scope as a set of *intervals*, where an interval is a nonempty sequence of consecutive numbers, corresponding to indices of the states in the model σ.

Globally:

The scope consists of one interval listing all the indexes of the states in the model

$$\mathbb{S}_G = \big\{[0, \infty)\big\} \tag{3.1}$$

Before R:

The scope contains one interval at most. This interval includes the indices of all states before (but not including) the first R-state. If an R-state is absent or in position 0, the scope is empty

$$\mathbb{S}_{B_R} = \big\{[0, i) \mid s_0 \Vdash \neg R,\, i = \min(\{k > 0 \mid s_k \Vdash R\})\big\} \tag{3.2}$$

After Q:

The scope contains one interval at most. This interval corresponds to all states occurring after (and including) the first Q-state. If no Q-state exists, the scope is empty

$$\mathbb{S}_{A_Q} = \big\{[i, \infty) \mid i = \min(\{k \geq 0 \mid s_k \Vdash Q\})\big\} \tag{3.3}$$

Between Q and R:

This scope may consist of multiple intervals. Each interval starts with an index of a

Q-state that is not an R-state, (inclusive) and extends to the index of the next R-state (not inclusive). A Q-state that is also an R-state is not an interval.

$$\mathbb{S}_{BW_{QR}} = \left\{[i,j) \mid i \geq 0, s_i \Vdash (Q \wedge \neg R), j = \min(\{k > i \mid s_k \Vdash R\})\right\} \tag{3.4}$$

After Q until R:

This scope extends "Between Q and R" with all suffixes that begin with a Q-state that is not an R-state (inclusive) and have no subsequent R-states.

$$\mathbb{S}_{AU_{QR}} = \mathbb{S}_{BW_{QR}} \cup \left\{[i,\infty) \mid i \geq 0, s_i \Vdash Q, (\forall j \geq i \bullet\ s_j \Vdash \neg R)\right\} \tag{3.5}$$

3.2 Formal Definitions of Weak Scopes

The scopes defined in Section 3.1 contain no empty intervals by construction, in this section we weaken this restriction. Depending on a model the weak scope $\mathbb{S}^W$ is either a scope $\mathbb{S}$ from Section 3.1, or the scope $\mathbb{S}$ extended with the empty interval.

There are only two cases when a scope $\mathbb{S}^W$ may contain an empty interval:

- initial state is an R-state in "Before R" scope, i.e. $s_0 \Vdash R$
- there is a $(Q \wedge R)$-state in "Between Q and R" or "After Q until R" scopes, i.e. $\exists n \geq 0 \bullet s_n \Vdash Q \wedge R$

Here we provide weak definitions only for "Before R", "Between Q and R" and "After Q until R" scopes. The rest of the scopes can't possibly contain an empty interval and, therefore, their weaken versions coincide with the versions defined in Section 3.1.

Weak Before R:

The scope contains one interval at most. This interval includes the indices of all states before (but not including) the first R-state. If an R-state is absent, the scope is empty. If R-state is in position 0, the scope consists of an empty interval.

$$\mathbb{S}^W_{B_R} = \left\{[0, i) \mid i = \min(\{k \geq 0 \mid s_k \models R\})\right\} \tag{3.6}$$

Weak Between Q and R:

This scope may consist of multiple intervals. Each interval starts with an index of a Q-state (inclusive) and extends to the index of the next R-state (not inclusive). A Q-state that is also an R-state is an empty interval.

$$\mathbb{S}^W_{BW_{QR}} = \left\{[i, j) \mid s_i \models Q, j = \min(\{k \geq i \mid s_k \models R\})\right\} \tag{3.7}$$

Weak After Q until R:

This scope extends "Weak Between Q and R" with all suffixes that begin with a Q-state that is not an R-state (inclusive) and have no subsequent R-states.

$$\mathbb{S}^W_{AU_{QR}} = \mathbb{S}_{AU_{QR}} \cup \mathbb{S}^W_{BW_{QR}} \tag{3.8}$$

When we switch from $\mathbb{S}$ to the weak version of scope $\mathbb{S}^W$, LTL formulas for the constraints that use a universal quantification over the elements of the intervals, does not change. However, the use of an existential quantifier in "Existence" and "Strong Existence" will fail the constraint on an empty interval.

3.3 Formal Definitions of Patterns

According to the informal definition of scopes, when there is no scope, all patterns listed in Section 2.1 are *True*. However, similarly to an existence being *False* on an empty set, one may require the "Existence" pattern to be *False*, when the scope is absent (i.e. empty). We introduce the "Strong Existence" pattern with exactly that additional requirement.

Absence:

There is no P-state in the scope.

$$\forall I \in \mathbb{S}, \forall n \in I \bullet s_n \models \neg P \tag{3.9}$$

Existence:

Each interval contains a P-state.

$$\forall I \in \mathbb{S}, \exists n \in I \bullet s_n \Vdash P \tag{3.10}$$

Strong Existence:

The scope is nonempty and each of it's intervals contains a P-state.

$$\mathbb{S} \neq \varnothing,\ \forall I \in \mathbb{S}, \exists n \in I \bullet s_n \Vdash P \tag{3.11}$$

Universality:

Every state in the scope is a P-state.

$$\forall I \in \mathbb{S}, \forall n \in I \bullet s_n \Vdash P \tag{3.12}$$

k **- Bounded Existence:**

Given an interval I in the scope $\mathbb{S}$, we define $Max(I, P)$ as a set of maximal P-state subintervals of I.

$$Max(I,P) = \big\{[i,j) \mid [i,j) \subseteq I, \forall m \in [i,j) \bullet s_m \Vdash P, \\ (i-1 \notin I \vee s_{i-1} \Vdash \neg P), (j \notin I \vee s_j \Vdash \neg P)\big\} \tag{3.13}$$

No interval in the scope contains more than k maximal P-state subintervals.

$$\forall I \in \mathbb{S} \bullet \big|Max(I,P)\big| \leq k \tag{3.14}$$

Precedence:

If there is a P-state, either it is an S-state or there is an S-state before it in the same interval.

$$\forall I \in \mathbb{S}, \forall n \in I \bullet (s_n \Vdash P \implies \exists m \in I \bullet (m \leq n, s_m \Vdash S)) \tag{3.15}$$

Response:

If there is a P-state, either it is an S-state or there is an S-state after it in the same interval.

$$\forall I \in \mathbb{S}, \forall n \in I \bullet (s_n \Vdash P \implies \exists m \in I \bullet (m \geq n, s_m \Vdash S)) \tag{3.16}$$

Precedence Chain:

A subsequence of states satisfying $S_1, \ldots, S_k$ precedes each subsequence of states satisfying $P_1, \ldots, P_l$, if the later exists. It is possible for the S_k-state to coincide with the P_1-state. However, an S_i-state strictly precedes S_j-state for $i < j$; with the similar requirement for the P sequence.

$$\begin{aligned} &\forall I \in \mathbb{S},\ \forall n_1 \in I, \ldots, \forall n_l \in I : n_1 < \ldots < n_l \bullet \big(s_{n_1} \Vdash P_1, \ldots, s_{n_l} \Vdash P_l \implies \\ &\exists m_1 \in I, \ldots, \exists m_k \in I : m_1 < \ldots < m_k \leq n_1 \bullet (s_{m_1} \Vdash S_1, \ldots, s_{m_k} \Vdash S_k)\big) \end{aligned} \tag{3.17}$$

Response Chain:

A subsequence of states satisfying $S_1, \ldots, S_k$ follows each subsequence of states satisfying $P_1, \ldots, P_l$, if the later exists. It is possible for the P_l-state to coincide with the S_1-state. However, an S_i-state strictly precedes the S_j-state for $i < j$, with the similar requirement for the P sequence.

$$\begin{aligned} &\forall I \in \mathbb{S},\ \forall n_1 \in I, \ldots, \forall n_l \in I : n_1 < \ldots < n_l \bullet \big(s_{n_1} \Vdash P_1, \ldots, s_{n_l} \Vdash P_l \implies \\ &\exists m_1 \in I, \ldots, \exists m_k \in I : n_l \leq m_1 < \ldots < m_k \bullet (s_{m_1} \Vdash S_1, \ldots, s_{m_k} \Vdash S_k)\big) \end{aligned} \tag{3.18}$$

3.4 Pattern Mappings for LTL

Parenthesizes in the formulas below are set according to the notation introduced in [14]: temporal operators have higher binding power that the boolean ones. '$\rightarrow$' in LTL formulas is equivalent to '$\implies$' used everywhere else in this paper and stands for the implication.

To map the informal definitions of patterns and scopes into a precise formula in common formal specification languages, one has to guess the formula first and then verify that this

formula is consistent with the definitions. Based on the formal definitions introduced in Sections 3.1, 3.2 and 3.3, we were able to formally derive corresponding LTL formulas, thereby ensuring that the translations are consistent with our definitions. This is a major improvement over the way the formulas in [1] were created. The formulas that we derived are not necessarily identical to the formulas in [1]. In cases where the formula that we derived differs from the translation in [1], we provide a proof that the formulas are equivalent or a counterexample that shows they are not equivalent.

Due to the overall complexity of manual derivation of these formulas, we decided to limit the extent of this work to the "Absence", "Existence", "Strong Existence" and "Universality" patterns with both Strong and Weak versions of all scopes, and the "Precedence" and "Response" patterns with the "Globally" scope. By doing so, we cover 89% of the 555 pattern/scope combinations mentioned in the survey in [7].

Our LTL formulas in Table 3.1 are proven to be equivalent to the corresponding original formulas by M.Dwyer et al. [1]. The proofs are listed in the Appendix B.

We simplified the LTL formulas for the "Absence" and "Universality" patterns with both "Between" and "After-Until" scopes by removing the redundant $\neg R$ term from the antecedent of the implication. We also fixed a typographical error in the "Existence" pattern with "After" scope.

In Section 2.3.3 we described the inconsistent treatment of empty intervals in the original work by M.Dwyer et al. To resolve this problem we proposed two possible formalizations: strong scopes and weak scopes (Sections 3.1 and 3.2 respectively). We show the corresponding LTL formulas for these versions of scopes in Table 3.2. It is enough to list only the formulas related to existential patterns, as there is no essential difference between strong and weak scopes for the other patterns and these formulas are listed in Table 3.1.

For both cases, if we ignore empty intervals (strong scopes) or take them into account (weak scopes), we propose the LTL formulas. All formulas in Table 3.2 are proven to be consistent with our definitions of strong and weak scopes in Appendix B. We highlight in

Property Pattern	New	Original (if different)
Absence of P		
Globally	$\Box(\neg P)$	
Before R	$\Diamond R \rightarrow (\neg P)\mathcal{U}R$	
After Q	$\Box(Q \rightarrow \Box(\neg P))$	
Between Q and R	$\Box(Q \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R)$	$\Box(Q \wedge \neg R \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R)$
After Q until R	$\Box(Q \rightarrow (\neg P)\mathcal{W}R)$	$\Box(Q \wedge \neg R \rightarrow (\neg P)\mathcal{W}R)$
Existence of P		
Globally	$\Diamond P$	
Before R	see Table 3.2	$(\neg R)\mathcal{W}(P \wedge \neg R)$
After Q	$\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P)$	$\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P))$ typo
Between Q and R	see Table 3.2	$\Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R))$
After Q until R	see Table 3.2	$\Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{U}(P \wedge \neg R))$
Universality of P		
Globally	$\Box P$	
Before R	$\Diamond R \rightarrow P\mathcal{U}R$	
After Q	$\Box(Q \rightarrow \Box P)$	
Between Q and R	$\Box(Q \wedge \Diamond R \rightarrow P\mathcal{U}R)$	$\Box(Q \wedge \neg R \wedge \Diamond R \rightarrow P\mathcal{U}R)$
After Q until R	$\Box(Q \rightarrow P\mathcal{W}R)$	$\Box(Q \wedge \neg R \rightarrow P\mathcal{W}R)$
S **precedes** P		
Globally	$(\neg P)\mathcal{W}S$	
S **responds to** P		
Globally	$\Box(P \rightarrow \Diamond S)$	

Table 3.1: Proposed LTL formulas vs. original formulas by M.Dwyer et al.

bold the formulas that are equivalent to the the original formulas by M.Dwyer et al. As it was stated in Section 2.3.3, original formulas for the "Existence" pattern with the "Between" and "After-Until" scopes ignore empty intervals, while the formula for the "Before" scope does not.

While the proposed pattern "Strong Existence" is not equivalent to the "Existence" pattern by M.Dwyer et al., we think some users may find this alternative interpretation to be valuable.

Property Pattern	LTL formula for: Strong scopes	Weak scopes
Existence of P		
Globally	$\boldsymbol{\Diamond P}$	$\boldsymbol{\Diamond P}$
Before R	$R \vee \neg(\neg P)\mathcal{U}R$	$\boldsymbol{\neg(\neg P)\mathcal{U}R}$
After Q	$\boldsymbol{\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P)}$	$\boldsymbol{\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P)}$
Between Q and R	$\boldsymbol{\Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{U}R)}$	$\Box(Q \rightarrow \neg(\neg P)\mathcal{U}R)$
After Q until R	$\boldsymbol{\Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R)}$	$\Box(Q \rightarrow \neg(\neg P)\mathcal{W}R)$
Strong Existence of P		
Globally	$\boldsymbol{\Diamond P}$	$\boldsymbol{\Diamond P}$
Before R	$\Diamond R \wedge \neg(\neg P)\mathcal{U}R$	$\Diamond R \wedge \neg(\neg P)\mathcal{U}R$
After Q	$\Diamond(Q \wedge \Diamond P)$	$\Diamond(Q \wedge \Diamond P)$
Between Q and R	$\Diamond(Q \wedge \neg R \wedge \Diamond R) \wedge$ $\Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{U}R)$	$\Diamond(Q \wedge \Diamond R) \wedge$ $\Box(Q \rightarrow \neg(\neg P)\mathcal{U}R)$
After Q until R	$\Diamond(Q \wedge \neg R) \wedge$ $\Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R)$	$\Diamond(Q \wedge (\neg R \vee \Diamond R)) \wedge$ $\Box(Q \rightarrow \neg(\neg P)\mathcal{W}R)$

Table 3.2: Proposed LTL formulas for Weak and Strong scopes. Bold formulas are equivalent to original formulas of M.Dwyer et al.

Chapter 4

RELATED WORK

This work is based on the specification pattern system (SPS) described in Dwyer et al. [6, 7] and the translations of patterns to LTL in [1].

Several approaches were proposed to formalize and extend SPS, and to verify the translations to different formalisms. PROPEL [23] uses the disciplined natural language (DNL) and the finite-state automaton (FSA) notations to formalize most common patterns in SPS. This approach assumes only the event-based formalism and translates patterns only between DNL and FSA.

Prospec [17] extends SPS with 12 classes of composite propositions (CP), allowing the use of multiple propositions in a pattern or as a delimiter of the scope. A tool interactively guide the user during the specification process and translates the specification to future interval logic or LTL. However [17] does not provide any details about the correctness of the generated LTL formulas.

Salamah et al. [21] extend ideas of Prospect. The paper considers 4 patterns out of the 8, defined by Dwyer et al. and CPs from Prospec, translates CP classes to LTL, extends LTL with an additional conjunction operator and translates 4 patterns and 5 scopes into the extended LTL. In [22], Salamah formally proves correctness of the formulas within the "Global" scope and tests the formulas within the "Before R" scope. He elaborates on the CP extension, uses non-standard extension of LTL for the translation and does not verify most of the formulas.

Garcia and Roach [10] developed the Property Testing Tool (Protest), which automatically generates and tests LTL formulas representing specifications. They did some additional testing of the formulas in [21], but their tests covered only a small subset of properties and CP classes.

Chapter 5

CONCLUSIONS

The use of patterns is a way for experts to share their knowledge. Like design patterns, specification patterns prepared by an expert speed up the process of writing specifications and widen the use of formal methods in software development. The SPS developed by Dwyer et al. is well known and easily accessible through a web site [1]. While being prepared and reviewed by experts, this collection is not guaranteed to be correct. The definitions of patterns and scopes have to be precise and their translation to other formalisms have to be verified. As we showed in Section 2.3 the original definitions contain some ambiguity and their manner of interpretation does not appear to always be consistent. It makes little sense to verify the formal LTL formulas obtained from the informal definitions. We proposed a formalization of patterns and scopes that is easy to read, closely resembles the original definitions, and is precise. These formal definitions were used to formally derive the corresponding LTL formulas. We verified the formulas provided in [1] by showing their equivalence to the formulas we derived. Because of the complexity of the manual proofs, we considered only the most popular combinations of patterns and scopes. "Absence", "Existence" and "Universality" patterns are verified with all five scopes; "Precedence" and "Response" are verified with "Global" scope only. From 555 patterns/scopes listed in the survey [7] we covered 89% of them. We provide the additional pattern, "Strong Existence", which is *False* on the empty scope. The corresponding LTL formulas for this pattern with all five scopes were derived and also verified.

APPENDICES

Appendix A

ORIGINAL LTL FORMULAS

Parenthesizes in the formulas below are set according to the notation introduced in [14]: temporal operators have higher binding power that the boolean ones.

Property Pattern	LTL formulas by M.Dwyer et al.
Absence of P	
Globally	$\Box(\neg P)$
Before R	$\Diamond R \rightarrow (\neg P)\mathcal{U}R$
After Q	$\Box(Q \rightarrow \Box(\neg P))$
Between Q and R	$\Box(Q \wedge \neg R \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R)$
After Q until R	$\Box(Q \wedge \neg R \rightarrow (\neg P)\mathcal{W}R)$
Existence of P	
Globally	$\Diamond P$
Before R	$(\neg R)\mathcal{W}(P \wedge \neg R)$
After Q	$\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P))$ typo
Between Q and R	$\Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R))$
After Q until R	$\Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{U}(P \wedge \neg R))$
Universality of P	
Globally	$\Box P$
Before R	$\Diamond R \rightarrow P\mathcal{U}R$
After Q	$\Box(Q \rightarrow \Box P)$
Between Q and R	$\Box(Q \wedge \neg R \wedge \Diamond R \rightarrow P\mathcal{U}R)$
After Q until R	$\Box(Q \wedge \neg R \rightarrow P\mathcal{W}R)$
S **precedes** P	
Globally	$(\neg P)\mathcal{W}S$
Before R	$\Diamond R \rightarrow (\neg P)\mathcal{W}(S \vee R)$
After Q	$(\neg Q)\mathcal{W}(Q \wedge (\neg P)\mathcal{W}S)$
Between Q and R	$\Box((Q \wedge \neg R \wedge \Diamond R) \rightarrow (\neg P)\mathcal{W}(S \vee R))$
After Q until R	$\Box(Q \wedge \neg R \rightarrow (\neg P)\mathcal{W}(S \vee R))$
S **responds to** P	
Globally	$\Box(P \rightarrow \Diamond S)$
Before R	$\Diamond R \rightarrow (P \rightarrow (\neg R)\mathcal{U}(S \wedge \neg R))\mathcal{U}R$
After Q	$(\neg Q)\mathcal{W}(Q \wedge \Box(P \rightarrow \Diamond S))$
Between Q and R	$\Box((Q \wedge \neg R \wedge \Diamond R) \rightarrow$ $(P \rightarrow (\neg R)\mathcal{U}(S \wedge \neg R))\mathcal{U}R)$
After Q until R	$\Box(Q \wedge \neg R \rightarrow$ $((P \rightarrow (\neg R)\mathcal{U}(S \wedge \neg R)))\mathcal{W}R)$

Table A.1: LTL formulas presented in [1]

Appendix B

PROOFS OF EQUIVALENCE

Before starting the proofs of equivalences, we define and prove two auxiliary claims.

Claim 1.

$$\neg(\neg P)\mathcal{U}R \iff (\neg R)\mathcal{W}(P \wedge \neg R) \tag{B.1}$$

Proof of the Claim 1. We want to show

$$\neg(\neg P)\mathcal{U}R \iff \Box(\neg R) \vee (\neg R)\mathcal{U}(P \wedge \neg R)$$

Suppose there is no R-state in our model. It is easy to see that both LTLs are *True*.

For the rest of the proof we assume an R-state exists and show

$$\neg(\neg P)\mathcal{U}R \iff (\neg R)\mathcal{U}(P \wedge \neg R)$$

If $\neg(\neg P)\mathcal{U}R$ is *True*, $(\neg P)\mathcal{U}R$ is *False*. As an R-state exists by our assumption, there is a P-state before the first R-state. Therefore $(\neg R)\mathcal{U}(P \wedge \neg R)$ is *True*.

If $(\neg R)\mathcal{U}(P \wedge \neg R)$ is *True*, there exists a P-state before any R-state. Then $(\neg P)\mathcal{U}R$ is *False* and $\neg(\neg P)\mathcal{U}R$ is *True*. We proved the iff relation in (B.1). ■

Claim 2.

$$\neg(\neg P)\mathcal{W}R \iff (\neg R)\mathcal{U}(P \wedge \neg R) \tag{B.2}$$

Proof of the Claim 2.

$$\neg(\neg P)\mathcal{W}R = \neg(\Box(\neg P) \vee (\neg P)\mathcal{U}R) = \Diamond P \wedge \neg(\neg P)\mathcal{U}R$$

Suppose there is no R-state in our model. Both $\neg(\neg P)\mathcal{W}R$ and $(\neg R)\mathcal{U}(P \wedge \neg R)$ reduce to $\Diamond P$.

For the rest of the proof we assume there is an R-state. In the proof of Claim 1 we've shown $(\neg R)\mathcal{U}(P \wedge \neg R) \iff \neg(\neg P)\mathcal{U}R$.

If $(\neg R)\mathcal{U}(P \wedge \neg R)$ is *False*, $\Diamond P \wedge \neg(\neg P)\mathcal{U}R$ is also *False*.

If $\Diamond P \wedge \neg(\neg P)\mathcal{U}R$ is *False*, either $\Diamond P$, $\neg(\neg P)\mathcal{U}R$ or both are *False*. It is easy to see that $(\neg R)\mathcal{U}(P \wedge \neg R)$ is *False* in all these cases. This proves (B.2). ■

B.0.1 Absence

B.0.1.1 Globally

$$\Box(\neg P) \;\approx\; \text{Absence of } P, \text{ Globally}$$

Proof.

$$\text{Absence of } P, \text{ Globally} \;\equiv\; \forall I \in \mathbb{S}_G, \forall n \in I \bullet s_n \Vdash \neg P, \text{ where } \mathbb{S}_G = \{[0, \infty)\}$$

$\mathbb{S}_G$ has only one interval, $[0, \infty)$, so the RHS of the equation above is

$$\forall n > 0 \bullet s_n \Vdash \neg P$$

. This is exactly the meaning of $\Box(\neg P)$ in first-order predicate logic. ■

B.0.1.2 Before R

Claim 3.

$$\forall i \geq 0 \bullet (s_0 \Vdash \neg R,\, i = \min(\{k > 0 \mid s_k \Vdash R\}) \implies \phi(i)) \;\equiv$$
$$\exists j \geq 0 \bullet s_j \Vdash R \implies \exists i \geq 0 \bullet (s_i \Vdash R,\, \phi(i)) \quad \text{(B.3)}$$

Proof.

$$
\begin{aligned}
&\forall i \geq 0 \bullet (s_0 \models \neg R,\, i = \min(\{k > 0 \mid s_k \models R\}) \implies \phi(i)) && \iff \\
&s_0 \models R \;\vee\; \forall i \geq 0 \bullet (i = \min(\{k > 0 \mid s_k \models R\}) \implies \phi(i)) && \iff \\
&s_0 \models R \;\vee\; \forall j \geq 0 \bullet s_j \models \neg R \;\vee\; \exists i > 0 \bullet (i = \min(\{k > 0 \mid s_k \models R\}),\, \phi(i)) && \iff \\
&\forall j \geq 0 \bullet s_j \models \neg R \;\vee\; \exists i \geq 0 \bullet (i = \min(\{k \geq 0 \mid s_k \models R\}),\, \phi(i)) && \iff \\
&\exists j \geq 0 \bullet s_j \models R \implies \exists i \geq 0 \bullet \big(i = \min(\{k \geq 0 \mid s_k \models R\}),\, \phi(i)) && \iff \\
&\exists j \geq 0 \bullet s_j \models R \implies \exists i \geq 0 \bullet (s_i \models R,\, \phi(i))
\end{aligned}
$$

■

$$\Diamond R \to (\neg P)\mathcal{U}R \;\approx\; \text{Absence of } P,\ \text{Before } R \tag{B.4}$$

Proof. According to (3.9)

$$\text{Absence of } P,\ \text{Before } R \;\equiv\; \forall I \in \mathbb{S}_{B_R}, \forall n \in I \bullet s_n \models \neg P$$

it follows from (3.2) that

$$
\begin{aligned}
&\forall I \in \mathbb{S}_{B_R}, \forall n \in I \bullet s_n \models \neg P \;\equiv \\
&\qquad \forall i \geq 0 \bullet \big(s_0 \models \neg R,\, i = \min(\{k > 0 \mid s_k \models R\}) \implies \forall n \in [0, i) \bullet s_n \models \neg P\big)
\end{aligned}
$$

by Claim 3, the RHS of the equivalence holds iff the following holds

$$\exists j \geq 0 \bullet s_j \models R \implies \exists i \geq 0 \bullet (s_i \models R,\ \forall n \in [0, i) \bullet s_n \models \neg P)$$

this is the definition of $\Diamond R \to (\neg P)\mathcal{U}R$. ■

B.0.1.3 After Q

$$\Box(Q \to \Box(\neg P)) \;\approx\; \text{Absence of } P,\ \text{After } Q \tag{B.5}$$

Proof. According to (3.9)

$$\text{Absence of } P,\ \text{After } Q \ \equiv\ \forall I \in \mathbb{S}_{A_Q}, \forall n \in I \bullet s_n \Vdash \neg P$$

it follows from (3.3) that

$$\forall I \in \mathbb{S}_{A_Q}, \forall n \in I \bullet s_n \Vdash \neg P \ \equiv$$
$$\forall i \geq 0 \bullet \big(i = \min(\{k \geq 0 \mid s_k \Vdash Q\}) \implies \forall n \in [i, \infty) \bullet s_n \Vdash \neg P\big)$$

the RHS of the equivalence holds iff the following holds

$$\forall i \geq 0 \bullet (s_i \Vdash Q \implies \forall n \in [i, \infty) \bullet s_n \Vdash \neg P)$$

this is the definition of $\Box(Q \rightarrow \Box(\neg P))$. ■

B.0.1.4 Between Q and R

$$\Box(Q \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R) \ \approx\ \text{Absence of } P, \text{Between } Q \text{ and } R \tag{B.6}$$

Proof. According to (3.9)

$$\text{Absence of } P,\ \text{Between } Q \text{ and } R \ \equiv\ \forall I \in \mathbb{S}_{BW_{QR}}, \forall n \in I \bullet s_n \Vdash \neg P$$

it follows from (3.4) that

$$\forall I \in \mathbb{S}_{BW_{QR}}, \forall n \in I \bullet s_n \Vdash \neg P \ \equiv$$
$$\forall i \geq 0, \forall j \geq 0 \bullet \big(i \geq 0, s_i \Vdash (Q \wedge \neg R), j = \min(\{k > i \mid s_k \Vdash R\}) \implies$$
$$\forall n \in [i, j) \bullet s_n \Vdash \neg P\big)$$

the RHS of the equivalence holds iff the following holds

$$\forall i \geq 0, \forall j \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R), j = \min(\{k > i \mid s_k \Vdash R\}) \implies \forall n \in [i,j) \bullet s_n \Vdash \neg P\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash \neg(Q \wedge \neg R) \vee \forall j \geq 0 \bullet (j = \min(\{k > i \mid s_k \Vdash R\}) \implies \forall n \in [i,j) \bullet s_n \Vdash \neg P)\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash \neg Q \vee s_i \Vdash R \vee \forall m > i \bullet s_m \Vdash \neg R \vee \exists j \geq 0 \bullet (j = \min(\{k > i \mid s_k \Vdash R\}), \forall n \in [i,j) \bullet s_n \Vdash \neg P)\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash \neg Q \vee s_i \Vdash R \vee \forall m \geq i \bullet s_m \Vdash \neg R \vee \exists j \geq 0 \bullet (j = \min(\{k > i \mid s_k \Vdash R\}), \forall n \in [i,j) \bullet s_n \Vdash \neg P)\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash \neg Q \vee \forall m \geq i \bullet s_m \Vdash \neg R \vee \exists j \geq 0 \bullet (j = \min(\{k \geq i \mid s_k \Vdash R\}), \forall n \in [i,j) \bullet s_n \Vdash \neg P)\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash Q, \exists m \geq i \bullet s_m \Vdash R \implies \exists j \geq 0 \bullet (j = \min(\{k \geq i \mid s_k \Vdash R\}), \forall n \in [i,j) \bullet s_n \Vdash \neg P)\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash Q, \exists m \geq i \bullet s_m \Vdash R \implies \exists j \geq 0 \bullet (s_j \Vdash R, \forall n \in [i,j) \bullet s_n \Vdash \neg P)\big)$$

this is the definition of $\Box(Q \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R)$. ■

B.0.1.5 After Q until R

$$\Box(Q \rightarrow (\neg P)\mathcal{W}R) \approx \text{Absence of } P, \text{ After } Q \text{ until } R \tag{B.7}$$

Proof. According to (3.9)

$$\text{Absence of } P, \text{ After } Q \text{ until } R \equiv \forall I \in \mathbb{S}_{AU_{QR}}, \forall n \in I \bullet s_n \Vdash \neg P$$

it follows from (3.5) that

$$\forall I \in \mathbb{S}_{AU_{QR}}, \forall n \in I \bullet s_n \Vdash \neg P \iff$$
$$(\forall I \in \mathbb{S}_{BW_{QR}}, \forall n \in I \bullet s_n \Vdash \neg P) \wedge (\forall I \in (\mathbb{S}_{AU_{QR}} - \mathbb{S}_{BW_{QR}}), \forall n \in I \bullet s_n \Vdash \neg P)$$

For the left part of the conjunction

$$\forall I \in \mathbb{S}_{BW_{QR}}, \forall n \in I \bullet s_n \Vdash \neg P \equiv$$
$$\forall i \geq 0, \forall j \geq 0 \bullet \big(i \geq 0, s_i \Vdash (Q \wedge \neg R), j = \min(\{k > i \mid s_k \Vdash R\}) \implies$$
$$\forall n \in [i, j) \bullet s_n \Vdash \neg P\big)$$

in (B.6) we have shown, it is $\Box(Q \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R)$.

In the right part of the conjunction

$$\forall I \in (\mathbb{S}_{AU_{QR}} - \mathbb{S}_{BW_{QR}}), \forall n \in I \bullet s_n \Vdash \neg P \equiv$$
$$\forall i \geq 0 \bullet (i \geq 0,\ s_i \Vdash Q,\ \forall j \geq i \bullet s_j \Vdash \neg R \implies \forall n \geq i \bullet s_n \Vdash \neg P)$$

the RHS of the equivalence holds iff the following holds

$$\forall i \geq 0 \bullet (s_i \Vdash Q,\ \forall j \geq i \bullet s_j \Vdash \neg R \implies \forall n \geq i \bullet s_n \Vdash \neg P)$$

this is the definition of $\Box(Q \wedge \Box(\neg R) \rightarrow \Box(\neg P))$.

Conjunction of these LTL

$$\Box\,(Q \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R) \wedge \Box(Q \wedge \Box(\neg R) \rightarrow \Box(\neg P)) \quad =$$
$$\Box\,\big((Q \wedge \Diamond R \rightarrow (\neg P)\mathcal{U}R) \wedge (Q \wedge \Box(\neg R) \rightarrow \Box(\neg P))\big) \quad =$$
$$\Box\,\big(\neg Q \vee (\Diamond R \vee \Box(\neg P)) \wedge (\Box(\neg R) \vee (\neg P)\mathcal{U}R)\big) \qquad \text{(B.8)}$$

Consider two cases: $\Diamond R$ =*True* or $\Diamond R$ =*False* for the formula

$$(\Diamond R \vee \Box(\neg P)) \wedge (\Box(\neg R) \vee (\neg P)\mathcal{U}R)$$

If $\Diamond R = True$, we have

$$True \wedge (\neg P) \mathcal{U} R \;=\; (\neg P) \mathcal{U} R$$

If $\Diamond R = False$,

$$\Box(\neg P) \wedge True \;=\; \Box(\neg P)$$

therefore,

$$(\Diamond R \vee \Box(\neg P)) \wedge (\Box(\neg R) \vee (\neg P) \mathcal{U} R) \;=\; (\neg P) \mathcal{U} R \vee \Box(\neg P) \tag{B.9}$$

finally, from (B.8) and (B.9) it follows

$$\Box(\neg Q \vee (\neg P) \mathcal{U} R \vee \Box(\neg P)) \;=\; \Box(Q \rightarrow (\neg P) \mathcal{W} R)$$

■

B.0.2 Existence with Strong Scopes

B.0.2.1 Globally

$$\Diamond P \;\approx\; \text{Existence of } P, \text{ Globally}$$

Proof.

$$\forall I \in \mathbb{S}_G, \exists n \in I \bullet s_n \Vdash P, \quad \text{where } \mathbb{S}_G = \left\{[0, \infty)\right\}$$

The scope "Globally", $\mathbb{S}_G$, contains only one interval $[0, \infty)$. For this case the definition of "Existence" transforms to

$$\exists n \in [0, \infty) \bullet s_n \Vdash P$$

This is exactly the meaning of $\Diamond P$. ■

B.0.2.2 Before R

$$R \vee \neg(\neg P) \mathcal{U} R \;\approx\; \text{Existence of } P, \text{ Before } R \tag{B.10}$$

Proof. According to (3.10)

$$\text{Existence of } P, \text{ Before } R \;\equiv\; \forall I \in \mathbb{S}_{B_R}, \exists n \in I \bullet s_n \models P$$

it follows from (3.2) that

$$\forall I \in \mathbb{S}_{B_R}, \exists n \in I \bullet s_n \models P \;\equiv$$
$$\forall i \geq 0 \bullet \big(s_0 \models \neg R,\, i = \min(\{k > 0 \mid s_k \models R\}) \implies \exists n \in [0, i) \bullet s_n \models P\big)$$

the RHS of the equivalence holds iff the following holds

$$\begin{aligned}
& s_0 \models R \vee \forall i \geq 0 \bullet \big(i = \min(\{k > 0 \mid s_k \models R\}) \implies \exists n \in [0, i) \bullet s_n \models P\big) && \iff \\
& s_0 \models R \vee \forall i \geq 0, \exists n \in [0, i) \bullet \big(i = \min(\{k > 0 \mid s_k \models R\}) \implies s_n \models P\big) && \iff \\
& s_0 \models R \vee \forall i \geq 0, \exists n \in [0, i) \bullet \big(i = \min(\{k \geq 0 \mid s_k \models R\}) \implies s_n \models P\big) && \iff \\
& s_0 \models R \vee \forall i \geq 0, \exists n \in [0, i) \bullet \big(s_i \models R \implies s_n \models P\big) && \iff \\
& s_0 \models R \vee \forall i \geq 0, \exists n \in [0, i) \bullet \big(s_i \models \neg R \vee s_n \models P\big) && \iff \\
& s_0 \models R \vee \neg\big(\exists i \geq 0, \forall n \in [0, i) \bullet (s_i \models R, s_n \models \neg P)\big)
\end{aligned}$$

this is the definition of $R \vee \neg(\neg P)\mathcal{U}R$. ∎

Applying Claim 1, we show

$$R \vee \neg(\neg P)\mathcal{U}R \iff R \vee (\neg R)\mathcal{W}(P \wedge \neg R)$$

B.0.2.3 After Q

$$\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P) \;\approx\; \text{Existence of } P, \text{ After } Q \tag{B.11}$$

Proof. According to (3.10)

$$\text{Existence of } P, \text{ After } Q \;\equiv\; \forall I \in \mathbb{S}_{A_Q}, \exists n \in I \bullet s_n \models P$$

it follows from (3.3) that

$$\forall I \in \mathbb{S}_{A_Q}, \exists n \in I \bullet s_n \Vdash P \;\equiv\; \forall i \geq 0 \bullet \big(i = \min(\{k \geq 0 \mid s_k \Vdash Q\}) \implies \exists n \in [i, \infty) \bullet s_n \Vdash P\big)$$

the RHS of the equivalence holds iff the following holds

$$(\forall j \geq 0 \bullet s_j \Vdash \neg Q) \vee (\exists i \geq 0 \bullet (i = \min(\{k \geq 0 \mid s_k \Vdash Q\}) \wedge \exists n \in [i, \infty) \bullet s_n \Vdash P)$$

$$(\forall j \geq 0 \bullet s_j \Vdash \neg Q) \vee (\exists i \geq 0 \bullet (s_i \Vdash Q \wedge \exists n \in [i, \infty) \bullet s_n \Vdash P)$$

this is the definition of $\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P)$. ■

B.0.2.4 Between Q and R

$$\Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R)) \;\approx\; \text{Existence of } P, \text{Between } Q \text{ and } R \tag{B.12}$$

Proof. According to (3.10)

$$\text{Existence of } P, \text{ Between } Q \text{ and } R \;\equiv\; \forall I \in \mathbb{S}_{BW_{QR}}, \exists n \in I \bullet s_n \Vdash P$$

it follows from (3.4) that

$$\forall I \in \mathbb{S}_{BW_{QR}}, \exists n \in I \bullet s_n \Vdash P \;\equiv\; \forall i \geq 0, \forall j \geq 0 \bullet \big((s_i \Vdash (Q \wedge \neg R),\ j = \min(\{k > i \mid s_k \Vdash R\})) \implies \exists n \in [i, j) \bullet s_n \Vdash P\big)$$

the RHS of the equivalence holds iff the following holds

$$\forall i \geq 0, \forall j > i, \exists n \in [i, j) \bullet \big((s_i \Vdash (Q \wedge \neg R), s_j \Vdash R) \implies s_n \Vdash P\big) \iff$$

$$\forall i \geq 0, \forall j \geq i, \exists n \in [i, j) \bullet \big((s_i \Vdash (Q \wedge \neg R), s_j \Vdash R) \implies s_n \Vdash P\big) \iff$$

$$\forall i \geq 0, \forall j \geq i, \exists n \in [i, j) \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies (s_j \Vdash \neg R \vee s_n \Vdash P)\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies \forall j \geq i, \exists n \in [i, j) \bullet (s_j \Vdash \neg R \vee s_n \Vdash P)\big) \iff$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies \neg(\exists j \geq i, \forall n \in [i, j) \bullet (s_j \Vdash R, s_n \Vdash \neg P))\big)$$

this is the definition of $\Box\big(Q \wedge \neg R \rightarrow \neg((\neg P)\mathcal{U}R)\big)$. ■

Applying Claim 1 we show equivalence

$$\Box\big(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{U}R\big) \iff \Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R))$$

B.0.2.5 After Q until R

$$\Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{U}(P \wedge \neg R)) \;\approx\; \text{Existence of } P, \text{ After } Q \text{ until } R \tag{B.13}$$

Proof. According to (3.10)

$$\text{Existence of } P, \text{ After } Q \text{ until } R \;\equiv\; \forall I \in \mathbb{S}_{AU_{QR}}, \exists n \in I \bullet s_n \Vdash P$$

it follows from (3.5) that

$$\forall I \in \mathbb{S}_{AU_{QR}}, \exists n \in I \bullet s_n \Vdash P \iff$$
$$(\forall I \in \mathbb{S}_{BW_{QR}}, \exists n \in I \bullet s_n \Vdash P) \wedge (\forall I \in (\mathbb{S}_{AU_{QR}} - \mathbb{S}_{BW_{QR}}), \exists n \in I \bullet s_n \Vdash P)$$

$$(\forall I \in \mathbb{S}_{BW_{QR}}, \exists n \in I \bullet s_n \Vdash P) \wedge (\forall I \in (\mathbb{S}_{AU_{QR}} - \mathbb{S}_{BW_{QR}}), \exists n \in I \bullet s_n \Vdash P) \;\equiv$$
$$\forall i \geq 0, \forall j \bullet \big((s_i \Vdash (Q \wedge \neg R),\ j = \min(\{k > i \mid s_k \Vdash R\})) \implies \exists n \in [i, j) \bullet s_n \Vdash P\big) \wedge$$
$$\forall i \geq 0 \bullet \big((s_i \Vdash Q,\ \forall j \geq i \bullet s_j \Vdash \neg R) \implies \exists n \geq i \bullet s_n \Vdash P\big)$$

the RHS of the equivalence holds iff the following holds

$$\forall i \geq 0, \forall j > i \bullet \big((s_i \Vdash (Q \wedge \neg R), s_j \Vdash R) \implies \exists n \in [i, j) \bullet s_n \Vdash P\big) \wedge$$
$$\forall i \geq 0 \bullet \big((s_i \Vdash (Q \wedge \neg R),\ \forall j > i \bullet s_j \Vdash \neg R) \implies \exists n \geq i \bullet s_n \Vdash P\big)$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies \forall j > i \bullet (s_j \Vdash \neg R \vee \exists n \in [i, j) \bullet s_n \Vdash P)\big) \wedge$$
$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies (\exists j > i \bullet s_j \Vdash R \vee \exists n \geq i \bullet s_n \Vdash P)\big)$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies$$
$$\forall j > i \bullet (s_j \Vdash \neg R \vee \exists n \in [i, j) \bullet s_n \Vdash P) \wedge$$
$$(\exists j > i \bullet s_j \Vdash R \vee \exists n \geq i \bullet s_n \Vdash P)\big)$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies$$
$$\neg\big(\neg(\forall j > i \bullet (s_j \Vdash \neg R \vee \exists n \in [i,j) \bullet s_n \Vdash P)) \vee$$
$$\neg(\exists j > i \bullet s_j \Vdash R \vee \exists n \geq i \bullet s_n \Vdash P)\big)\big)$$

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies$$
$$\neg\big(\exists j > i \bullet (s_j \Vdash R \wedge \forall n \in [i,j) \bullet s_n \Vdash \neg P) \vee$$
$$(\forall j > i \bullet s_j \Vdash \neg R \wedge \forall n \geq i \bullet s_n \Vdash \neg P)\big)\big) \quad \text{(B.14)}$$

assume $s_i \Vdash (Q \wedge \neg R)$ holds in (B.14), then we can switch to weak inequalities

$$\exists j > i \bullet (s_j \Vdash R \wedge \forall n \in [i,j) \bullet s_n \Vdash \neg P) \iff \exists j \geq i \bullet (s_j \Vdash R \wedge \forall n \in [i,j) \bullet s_n \Vdash \neg P)$$
$$\forall j > i \bullet s_j \Vdash \neg R \iff \forall j \geq i \bullet s_j \Vdash \neg R$$

therefore (B.14) holds iff

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies$$
$$\neg\big(\exists j \geq i \bullet (s_j \Vdash R \wedge \forall n \in [i,j) \bullet s_n \Vdash \neg P) \vee$$
$$(\forall j \geq i \bullet s_j \Vdash \neg R \wedge \forall n \geq i \bullet s_n \Vdash \neg P)\big)\big) \quad \text{(B.15)}$$

To show $\forall j \geq i \bullet s_j \Vdash \neg R$ is redundant, it is enough to consider the case

$$\exists j \geq i \bullet (s_j \Vdash R \wedge \forall n \in [i,j) \bullet s_n \Vdash \neg P) = False$$
$$\forall j \geq i \bullet s_j \Vdash \neg R = False$$

$\forall j \geq i \bullet s_j \Vdash \neg R$ is the negation of $\exists j \geq i \bullet s_j \Vdash R$, therefore

$$\forall n \in [i,j) \bullet s_n \Vdash \neg P = False$$

it follows

$$\forall n \geq i \bullet s_n \Vdash \neg P = False$$

and $\forall j \geq i \bullet s_j \Vdash \neg R$ is redundant.

$$\forall i \geq 0 \bullet \big(s_i \Vdash (Q \wedge \neg R) \implies \neg(\ \exists j \geq i \bullet (s_j \Vdash R \wedge \forall n \in [i,j) \bullet s_n \Vdash \neg P)\ \vee\ (\forall n \geq i \bullet s_n \Vdash \neg P)\ )\big) \quad \text{(B.16)}$$

finally, using the following

$$\begin{aligned} \Box(\neg P)\ \vee\ (\neg P)\mathcal{U}R &= (\neg P)\mathcal{W}R \\ \exists j \geq i \bullet (s_j \Vdash R \wedge \forall n \in [i,j) \bullet s_n \Vdash \neg P) &\equiv (\neg P)\mathcal{U}R \\ \forall n \geq i \bullet s_n \Vdash \neg P &\equiv \Box(\neg P) \end{aligned}$$

we show that (B.16) is equivalent to

$$\Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R) \quad \text{(B.17)}$$

■

Finally, we apply Claim 2 to show

$$\Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R) \iff \Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{U}(P \wedge \neg R))$$

B.0.3 Strong Existence with Strong Scopes

Comparing to "Existence", "Strong Existence" adds the requirement for a scope not to be empty. In the following proofs, we start from deriving the LTL formula for this requirement, then conjoining this derived formula with LTLs for "Existence", obtained in Section B.0.2.

B.0.3.1 Globally

$$\Diamond P \ \approx\ \text{Strong Existence of } P, \text{ Globally}$$

Proof. The scope $\mathbb{S}_G = \{[0,\infty)\}$ is not empty, thus "Strong Existence" is equivalent to "Existence" on this scope. ■

B.0.3.2 Before R

We prove the equivalence using the proposed formula

$$\Diamond R \wedge \neg(\neg P)\mathcal{U}R \;\approx\; \text{Strong Existence of } P, \text{ Before } R \tag{B.18}$$

Proof. According to (3.2)

$$\mathbb{S}_{B_R} \neq \varnothing \;\equiv\; \exists i \geq 0 \bullet \big(s_0 \models \neg R,\, i = \min(\{k > 0 \mid s_k \models R\})\big)$$

$$\begin{aligned}
&\exists i \geq 0 \bullet (s_0 \models \neg R,\, i = \min(\{k > 0 \mid s_k \models R\})) && \iff \\
&s_0 \models \neg R,\; \exists i \geq 0 \bullet (i = \min(\{k > 0 \mid s_k \models R\})) && \iff \\
&s_0 \models \neg R,\; \exists i \geq 0 \bullet s_i \models R
\end{aligned}$$

this is the definition of $\neg R \wedge \Diamond R$. Conjoining with LTL from (B.10), we get

$$\begin{aligned}
&(\neg R \wedge \Diamond R) \wedge (R \vee \neg(\neg P)\mathcal{U}R) && \iff \\
&\Diamond R \wedge (\neg R \wedge (R \vee \neg(\neg P)\mathcal{U}R)) && \iff \\
&\Diamond R \wedge ((\neg R \wedge R) \vee (\neg R \wedge \neg(\neg P)\mathcal{U}R) && \iff \\
&\Diamond R \wedge \neg R \wedge \neg(\neg P)\mathcal{U}R && \qquad (B.19)
\end{aligned}$$

The $\neg R$ term is redundant. To prove it, it is enough to show that (B.19) is *False*, if $\neg R$ is *False*:

R is *True* and $(\neg P)\mathcal{U}R$ is also *True*, thus, $\neg(\neg P)\mathcal{U}R$ is *False* and the formula is *False*.

We have shown the equivalence with the following LTL formula

$$\Diamond R \wedge \neg(\neg P)\mathcal{U}R \tag{B.20}$$

This finishes the proof of equivalence. ∎

We apply Claim 1 to show it is equivalent to

$$\Diamond R \wedge (\neg R)\mathcal{W}(P \wedge \neg R))$$

B.0.3.3 After Q

We prove the following equivalence

$$\Diamond(Q \wedge \Diamond P) \;\approx\; \text{Strong Existence of } P, \text{ After } Q \tag{B.21}$$

Proof. According to (3.3)

$$\mathbb{S}_{A_Q} \neq \varnothing \;\equiv\; \exists i \geq 0 \bullet (i = \min(\{k \geq 0 \mid s_k \Vdash Q\}))$$

$$\begin{aligned} &\exists i \geq 0 \bullet (i = \min(\{k \geq 0 \mid s_k \Vdash Q\})) && \Longleftrightarrow \\ &\exists i \geq 0 \bullet s_i \Vdash Q \end{aligned}$$

this is the definition of $\Diamond Q$. Conjoining with LTL from (B.11), we get

$$\begin{aligned} &\Diamond\, Q \wedge (\Box(\neg Q) \vee \Diamond(Q \wedge \Diamond P)) && \Longleftrightarrow \\ &(\Diamond Q \wedge \Box(\neg Q)) \vee (\Diamond Q \wedge \Diamond(Q \wedge \Diamond P)) && \Longleftrightarrow \\ &\Diamond\, Q \wedge \Diamond(Q \wedge \Diamond P) && \Longleftrightarrow \\ &\Diamond\, (Q \wedge \Diamond P) \end{aligned}$$

This finishes the proof of equivalence. ∎

B.0.3.4 Between Q and R

We propose the following formula

$$\Diamond\big(Q \wedge \neg R \wedge \Diamond R\big) \;\wedge\; \Box\big(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{U}R\big) \;\approx$$

$$\text{Strong Existence of } P, \text{Between } Q \text{ and } R \tag{B.22}$$

Proof. According to (3.4)

$$\mathbb{S}_{BW_{QR}} \neq \varnothing \;\equiv\; \exists m \geq 0, \exists n \bullet \big(s_m \Vdash (Q \wedge \neg R),\, n = \min(\{k > m \mid s_k \Vdash R\})\big)$$

$$\exists m \geq 0, \exists n \bullet \big(s_m \Vdash\!\vDash (Q \wedge \neg R),\ n = \min(\{k > m \mid s_k \Vdash\!\vDash R\})\big) \iff$$

$$\exists m \geq 0, \exists n > m \bullet \big(s_m \Vdash\!\vDash (Q \wedge \neg R), s_n \Vdash\!\vDash R)\big) \iff$$

$$\exists m \geq 0, \exists n \geq m \bullet \big(s_m \Vdash\!\vDash (Q \wedge \neg R), s_n \Vdash\!\vDash R)\big)$$

this is exactly the definition of $\diamond(Q \wedge \neg R \wedge \diamond R)$. Conjoining with LTL from (B.12), we get

$$\diamond\big(Q \wedge \neg R \wedge \diamond R\big) \ \wedge\ \square\big(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{U}R\big)$$

This finishes the proof of equivalence. ■

We apply Claim 1 to show it is equivalent to

$$\diamond\big(Q \wedge \neg R \wedge \diamond R\big) \ \wedge\ \square\big(Q \wedge \neg R \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R)\big)$$

B.0.3.5 After Q until R

We propose the following formula

$$\diamond\,(Q \wedge \neg R) \ \wedge\ \square(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R) \ \approx$$

Strong Existence of P, After Q until R

Proof. According to (3.5)

$$\mathbb{S}_{AU_{QR}} \neq \varnothing \iff \mathbb{S}_{BW_{QR}} \neq \varnothing \ \vee\ (\mathbb{S}_{AU_{QR}} - \mathbb{S}_{BW_{QR}}) \neq \varnothing$$

$$\mathbb{S}_{BW_{QR}} \neq \varnothing \ \vee\ (\mathbb{S}_{AU_{QR}} - \mathbb{S}_{BW_{QR}}) \neq \varnothing \ \equiv$$

$$\exists i \geq 0, \exists j \bullet \big(s_i \Vdash\!\vDash (Q \wedge \neg R),\ j = \min(\{k > i \mid s_k \Vdash\!\vDash R\})\big) \vee$$

$$\exists i \geq 0, \forall j \geq i \bullet (s_i \Vdash\!\vDash Q, s_j \Vdash\!\vDash \neg R)$$

the RHS of the equivalence holds iff the following holds

$$\exists i \geq 0, \exists j > i \bullet (s_i \Vdash\!\vDash (Q \wedge \neg R), s_j \Vdash\!\vDash R) \ \vee\ \exists i \geq 0, \forall j \geq i \bullet (s_i \Vdash\!\vDash Q, s_j \Vdash\!\vDash \neg R)$$

$$\exists i \geq 0 \bullet \big(s_i \Vdash\!\vDash (Q \wedge \neg R), (\exists j > i \bullet s_j \Vdash\!\vDash R \ \vee\ \forall j > i \bullet s_j \Vdash\!\vDash \neg R)\big)$$

$$\exists i \geq 0 \bullet s_i \Vdash\!\vDash (Q \wedge \neg R)$$

this is the definition of

$$\Diamond(Q \wedge \neg R) \tag{B.23}$$

Conjoining with LTL from (B.13), we get

$$\Diamond(Q \wedge \neg R) \ \wedge \ \Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R)$$

This finishes the proof of equivalence. ■

We apply Claim 2 to show it is equivalent to

$$\Diamond(Q \wedge \neg R) \ \wedge \ \Box(Q \wedge \neg R \rightarrow (\neg R)\mathcal{U}(P \wedge \neg R))$$

B.0.4 Universality

"Universally P" is the same as "Absence of $\neg P$". Therefore we reuse LTLs and proofs from that section.

B.0.5 Existence with Weak Scopes, $\mathbb{S}^W$

B.0.5.1 (Before R)^W

$$\neg(\neg P)\mathcal{U}R \ \approx \ \text{Existence of } P, \ (\text{Before } R)^W \tag{B.24}$$

Proof. According to (3.10)

$$\text{Existence of } P, \ (\text{Before } R)^W \ \equiv \ \forall I \in \mathbb{S}^W_{B_R}, \exists n \in I \bullet s_n \models P$$

it follows from (3.6) that

$$\forall I \in \mathbb{S}^W_{B_R}, \exists n \in I \bullet s_n \models P \ \equiv$$
$$\forall i \geq 0 \bullet \big(i = \min(\{k \geq 0 \mid s_k \models R\}) \implies \exists n \in [0, i) \bullet s_n \models P\big)$$

the RHS of the equivalence holds iff the following holds

$$\begin{aligned}
&\forall i \geq 0, \exists n \in [0,i) \bullet (s_i \Vdash R \implies s_n \Vdash P) && \iff \\
&\forall i \geq 0, \exists n \in [0,i) \bullet (s_i \Vdash \neg R \vee s_n \Vdash P) && \iff \\
&\neg(\exists i \geq 0, \forall n \in [0,i) \bullet (s_i \Vdash R \wedge s_n \Vdash \neg P)
\end{aligned}$$

this is the definition of $\neg(\neg P)\mathcal{U}R$. ■

We apply Claim 1 to show it is equivalent to

$$(\neg R)\mathcal{W}(P \wedge \neg R)$$

B.0.5.2 (Between Q and R)^W

$$\Box\big(Q \to \neg(\neg P)\mathcal{U}R\big) \;\approx\; \text{Existence of } P, \text{ (Between } Q \text{ and } R)^W \tag{B.25}$$

Proof. According to (3.10)

$$\text{Existence of } P, \text{ (Between } Q \text{ and } R)^W \;\equiv\; \forall I \in \mathbb{S}^W_{BW_{QR}}, \exists n \in I \bullet s_n \Vdash P$$

it follows from (3.7) that

$$\begin{aligned}
&\forall I \in \mathbb{S}^W_{BW_{QR}}, \exists n \in I \bullet s_n \Vdash P \;\equiv \\
&\qquad \forall i \geq 0, \forall j \geq 0 \bullet \big(s_i \Vdash Q,\ j = \min(\{k \geq i \mid s_k \Vdash R\}) \implies \\
&\qquad\qquad \exists n \in [i,j) \bullet s_n \Vdash P\big)
\end{aligned}$$

the RHS of the equivalence holds iff the following holds

$$\begin{aligned}
&\forall i \geq 0, \forall j \geq i, \exists n \in [i,j) \bullet \big(s_i \Vdash Q, s_j \Vdash R \implies s_n \Vdash P\big) && \iff \\
&\forall i \geq 0, \forall j \geq i, \exists n \in [i,j) \bullet \big(s_i \Vdash Q \implies (s_j \Vdash \neg R \vee s_n \Vdash P)\big) && \iff \\
&\forall i \geq 0 \bullet \big(s_i \Vdash Q \implies \forall j \geq i, \exists n \in [i,j) \bullet (s_j \Vdash \neg R \vee s_n \Vdash P)\big) && \iff \\
&\forall i \geq 0 \bullet \big(s_i \Vdash Q \implies \neg(\exists j \geq i, \forall n \in [i,j) \bullet (s_j \Vdash R, s_n \Vdash \neg P))\big)
\end{aligned}$$

this is the definition of $\Box\big(Q \to \neg(\neg P)\mathcal{U}R\big)$. ■

We apply Claim 1 to show it is equivalent to

$$\Box(Q \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R))$$

B.0.5.3 (After Q until R)^W

$$\Box\big(Q \rightarrow \neg(\neg P)\mathcal{W}R\big) \;\approx\; \text{Existence of } P, \text{ (After } Q \text{ until } R)^W \tag{B.26}$$

Proof. According to (3.10)

$$\text{Existence of } P, \text{ (After } Q \text{ until } R)^W \;\equiv\; \forall I \in \mathbb{S}^W_{AU_{QR}}, \exists n \in I \bullet s_n \Vdash P$$

it follows from (3.8) that

$$\forall I \in \mathbb{S}^W_{AU_{QR}}, \exists n \in I \bullet s_n \Vdash P \iff$$
$$(\forall I \in \mathbb{S}_{AU_{QR}}, \exists n \in I \bullet s_n \Vdash P) \wedge (\forall I \in \mathbb{S}^W_{BW_{QR}}, \exists n \in I \bullet s_n \Vdash P)$$

in (B.13) and (B.28) we have derived the corresponding LTLs

$$\Box\big(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R\big) \wedge \Box\big(Q \rightarrow \neg(\neg P)\mathcal{U}R\big) \iff$$
$$\Box\big((Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R) \wedge (Q \rightarrow \neg(\neg P)\mathcal{U}R)\big) \iff$$

It is easy to see that $\neg R$ term is redundant

$$\Box\big((Q \rightarrow \neg(\neg P)\mathcal{W}R) \wedge (Q \rightarrow \neg(\neg P)\mathcal{U}R)\big) \iff$$
$$\Box\big((Q \rightarrow (\neg\,\Box\,(\neg P) \wedge \neg(\neg P)\mathcal{U}R)) \wedge (Q \rightarrow \neg(\neg P)\mathcal{U}R)\big) \iff$$
$$\Box\big(Q \rightarrow (\neg\,\Box\,(\neg P) \wedge \neg(\neg P)\mathcal{U}R)\big) \iff$$
$$\Box\big(Q \rightarrow \neg(\neg P)\mathcal{W}R\big)$$

This proves the equivalence. ∎

Finally, we apply Claim 2 to show

$$\Box\big(Q \rightarrow \neg(\neg P)\mathcal{W}R\big) \iff \Box(Q \rightarrow (\neg R)\mathcal{U}(P \wedge \neg R))$$

B.0.6 Strong Existence with Weak Scopes, $\mathbb{S}^W$

As in the Section B.0.3, it is enough to derive the LTL for the case of an empty scope and conjoin this LTL with the one derived in Section B.0.5.

B.0.6.1 (Before R)^W

$$\Diamond R \wedge \neg(\neg P)\mathcal{U}R \;\approx\; \text{Strong Existence of } P, \text{ (Before } R)^W \tag{B.27}$$

Proof. According to (3.6)

$$\mathbb{S}^W_{B_R} \neq \varnothing \;\equiv\; \exists i \geq 0 \bullet (i = \min(\{k \geq 0 \mid s_k \models R\}))$$

and the corresponding LTL is $\Diamond R$. Conjoining with (B.24)

$$\Diamond R \wedge \neg(\neg P)\mathcal{U}R$$

this is our final formula. ■

We apply Claim 1 to show it is equivalent to

$$\Diamond R \wedge (\neg R)\mathcal{W}(P \wedge \neg R)$$

B.0.6.2 (Between Q and R)^W

$$\Diamond(Q \wedge \Diamond R) \wedge \Box(Q \rightarrow \neg(\neg P)\mathcal{U}R) \;\approx\; \text{Existence of } P, \text{ (Between } Q \text{ and } R)^W \tag{B.28}$$

Proof. According to (3.7)

$$\mathbb{S}^W_{BW_{QR}} \neq \varnothing \;\equiv\; \exists i \geq 0, \exists j \geq 0 \bullet \big(s_i \models Q,\ j = \min(\{k \geq i \mid s_k \models R\})\big)$$

and the corresponding LTL is

$$\Diamond(Q \wedge \Diamond R) \tag{B.29}$$

Conjoining with (B.25)

$$\Diamond(Q \wedge \Diamond R) \wedge \Box(Q \rightarrow \neg(\neg P)\mathcal{U}R)$$

this is our final formula. ■

We apply Claim 1 to show it is equivalent to

$$\Diamond(Q \wedge \Diamond R) \wedge \Box(Q \rightarrow (\neg R)\mathcal{W}(P \wedge \neg R))$$

B.0.6.3 (After Q until R)^W

$$\Diamond\left(Q \wedge (\neg R \vee \Diamond R)\right) \wedge \Box\big(Q \rightarrow \neg(\neg P)\mathcal{W}R\big) \;\approx$$

Existence of P, (After Q until R)^W

Proof. According to (3.8)

$$\mathbb{S}^W_{AU_{QR}} \neq 0 \iff \mathbb{S}_{AU_{QR}} \neq 0 \vee \mathbb{S}^W_{BW_{QR}} \neq 0$$

it follows from (B.23) and (B.29)

$$\Diamond\left(Q \wedge \neg R\right) \vee \Diamond(Q \wedge \Diamond R) \qquad \iff \tag{B.30}$$

$$\Diamond\left((Q \wedge \neg R) \vee (Q \wedge \Diamond R)\right) \qquad \iff \tag{B.31}$$

$$\Diamond\left(Q \wedge (\neg R \vee \Diamond R)\right) \tag{B.32}$$

Conjoining with LTL from (B.26), we get

$$\Diamond(Q \wedge (\neg R \vee \Diamond R)) \wedge \Box\big(Q \rightarrow \neg(\neg P)\mathcal{W}R\big)$$

This finishes the proof of equivalence. ■

Finally, we apply Claim 2 to show it is equivalent

$$\Diamond(Q \wedge (\neg R \vee \Diamond R)) \wedge \Box(Q \rightarrow (\neg R)\mathcal{U}(P \wedge \neg R))$$

B.0.7 Precedence

B.0.8 Globally

$$(\neg P)\mathcal{W}S \;\approx\; S \text{ Precedes } P, \text{ Globally}$$

Proof. According to (3.1) and (3.15)

$$S \text{ Precedes } P, \text{ Globally} \quad \equiv \quad \forall n \in [0,\infty) \bullet (s_n \Vdash P \implies \exists m \in [0,n] \bullet s_m \Vdash S)$$

the LHS holds iff

$$\begin{aligned} &\forall n \geq 0 \bullet (s_n \Vdash \neg P \vee \exists m \in [0,n] \bullet s_m \Vdash S) && \iff \\ &\forall n \geq 0 \bullet (s_n \Vdash (\neg P \vee S) \vee \exists m \in [0,n) \bullet s_m \Vdash S) && \iff \\ &\neg\big(\exists n \geq 0 \bullet (s_n \Vdash (P \wedge \neg S),\ \forall m \in [0,n) \bullet s_m \Vdash \neg S)\big) \end{aligned}$$

this is the definition of $\neg((\neg S)\mathcal{U}(P \wedge \neg S))$. By (B.2) in Claim 2

$$\neg((\neg S)\mathcal{U}(P \wedge \neg S)) \iff (\neg P)\mathcal{W}S$$

this finishes the proof of equivalence. ■

B.0.9 Response

B.0.10 Globally

$$\Box(P \to \Diamond S) \quad \approx \quad S \text{ Responses to } P, \text{ Globally}$$

Proof. According to (3.1) and (3.16)

$$S \text{ Responses to } P, \text{ Globally} \quad \equiv \quad \forall n \in [0,\infty) \bullet (s_n \Vdash P \implies \exists m \geq n \bullet s_m \Vdash S)$$

this is the definition of $\Box(P \to \Diamond S)$. ■

Appendix C

PROOFS OF NON EQUIVALENCE

C.1 Existence

$$R \vee \neg(\neg P)\mathcal{U}R \not\equiv \text{Existence of } P, \text{ Before } R$$

Proof. Suppose the initial state of our model is an R-state ($s_0 \Vdash R$). The LTL formula on the LHS is *True*, however, the original LTL formula defined in [1], $(\neg R)\mathcal{W}(P \wedge \neg R)$, is *False* on this model. ■

$$\Box(Q \rightarrow \neg(\neg P)\mathcal{U}R) \not\equiv \text{Existence of } P, \text{ Between } Q \text{ and } R$$

Proof. Suppose the initial state of our model is a $(Q \wedge R)$-state and all other states are $(\neg Q)$-states, i.e. $(s_0 \Vdash Q \wedge R) \wedge (\forall i > 0 \bullet s_i \Vdash \neg Q)$. The LTL formula on the LHS is *False*, however, the original formula LTL defined in [1], $\Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{U}R)$, is *True*. ■

$$\Box(Q \rightarrow \neg(\neg P)\mathcal{W}R) \not\equiv \text{Existence of } P, \text{ After } Q \text{ until } R$$

Proof. We reuse the example form the previous proof. ■

C.2 Strong Existence

$$\Diamond R \wedge \neg(\neg P)\mathcal{U}R \not\equiv \text{Existence of } P, \text{ Before } R$$

Proof. Suppose our model is described by $\forall i \geq 0 \bullet s_i \Vdash (\neg R \wedge \neg P)$. The LTL formula on the LHS is *False*, however, the original LTL formula defined in [1], $(\neg R)\mathcal{W}(P \wedge \neg R)$, is *True* on this model. ■

$$\Diamond(Q \wedge \Diamond P) \not\models \text{Existence of } P, \text{ After } Q$$

$$\Diamond(Q \wedge \neg R \wedge \Diamond R) \wedge \Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{U}R) \not\models \text{Existence of } P, \text{ Between } Q \text{ and } R$$

$$\Diamond(Q \wedge \Diamond R) \wedge \Box(Q \rightarrow \neg(\neg P)\mathcal{U}R) \not\models \text{Existence of } P, \text{ Between } Q \text{ and } R$$

$$\Diamond(Q \wedge \neg R) \wedge \Box(Q \wedge \neg R \rightarrow \neg(\neg P)\mathcal{W}R) \not\models \text{Existence of } P, \text{ After } Q \text{ until } R$$

$$\Diamond(Q \wedge (\neg R \vee \Diamond R)) \wedge \Box(Q \rightarrow \neg(\neg P)\mathcal{W}R) \not\models \text{Existence of } P, \text{ After } Q \text{ until } R$$

Proof. For all five formulas we use the example, no Q-state in the model, $\forall i \geq 0 \bullet s_i \Vdash \neg Q$. The LTL formulas on the LHSs are *False*, however, the original LTL formulas defined in [1] are *True* on this model. ■

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Hamid Alavi, George Avrunin, James Corbett, Laura Dillon, Matt Dwyer, and Corina Pasareanu, *Property pattern mappings for ltl*, http://patterns.projects.cis.ksu.edu/documentation/patterns/ltl.shtml.

[2] A. Cimatti, E. Clarke, F. Giunchiglia, and M. Roveri, *Nusmv: a new symbolic model checker*, International Journal on Software Tools for Technology Transfer **2** (2000), 2000.

[3] E. M. Clarke, E. A. Emerson, and A. P. Sistla, *Automatic verification of finite-state concurrent systems using temporal logic specifications*, ACM Trans. Program. Lang. Syst. **8** (1986), 244–263.

[4] Rachel L. Cobleigh, George S. Avrunin, and Lori A. Clarke, *User guidance for creating precise and accessible property specifications*, Proceedings of the 14th ACM SIGSOFT international symposium on Foundations of software engineering (New York, NY, USA), SIGSOFT '06/FSE-14, ACM, 2006, pp. 208–218.

[5] James C. Corbett and George S. Avrunin, *Using integer programming to verify general safety and liveness properties*, Form. Methods Syst. Des. **6** (1995), 97–123.

[6] Matthew B. Dwyer, George S. Avrunin, and James C. Corbett, *Property specification patterns for finite-state verification*, Proceedings of the second workshop on Formal methods in software practice (New York, NY, USA), FMSP '98, ACM, 1998, pp. 7–15.

[7] Matthew B. Dwyer, George S. Avrunin, and James C. Corbett, *Patterns in property specifications for finite-state verification*, Proceedings of the 21st International Conference on Software Engineering ICSE'99, 1999.

[8] Matthew B. Dwyer, Lori A. Clarke, Jamieson M. Cobleigh, and Gleb Naumovich, *Flow analysis for verifying properties of concurrent software systems*, ACM Trans. Softw. Eng. Methodol. **13** (2004), 359–430.

[9] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, *Design patterns: elements of reusable object-oriented software*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1995.

[10] L. Garcia and S. Roach, *Model-checker-based testing of ltl specifications*, High Assurance Systems Engineering Symposium, 2007. HASE '07. 10th IEEE, 2007, pp. 417 –418.

[11] G.J. Holzmann, *The model checker spin*, Software Engineering, IEEE Transactions on **23** (1997), no. 5, 279 –295.

[12] Axel van Lamsweerde, *Formal specification: a roadmap*, Proceedings of the Conference on The Future of Software Engineering (New York, NY, USA), ICSE '00, ACM, 2000, pp. 147–159.

[13] Mich Luisa, Franch Mariangela, and Inverardi Pierluigi, *Market research for requirements analysis using linguistic tools*, Requir. Eng. **9** (2004), 40–56.

[14] Z. Manna and A. Pnueli, *Temporal verification of reactive systems: safety*, Temporal verification of reactive systems / Zohar Manna; Amir Pnueli, Springer, 1995.

[15] Radu Mateescu and Mihaela Sighireanu, *Efficient on-the-fly model-checking for regular alternation-free mu-calculus*, Sci. Comput. Program. **46** (2003), 255–281.

[16] Kenneth L. McMillan, *Symbolic model checking*, Kluwer Academic Publishers, Norwell, MA, USA, 1993.

[17] Oscar Mondragón, Ann Q. Gates, and Steven Roach, *Prospec: Support for elicitation and formal specification of software properties*, Electronic Notes in Theoretical Computer Science **89** (2003), no. 2, 67 – 88, RV '2003, Run-time Verification (Satellite Workshop of CAV '03).

[18] Rocco De Nicola and Frits W. Vaandrager, *Action versus state based logics for transition systems*, Proceedings of the LITP Spring School on Theoretical Computer Science: Semantics of Systems of Concurrent Processes (London, UK), Springer-Verlag, 1990, pp. 407–419.

[19] Kurt M. Olender and Leon J. Osterweil, *Cecil: A sequencing constraint language for automatic static analysis generation*, IEEE Trans. Softw. Eng. **16** (1990), 268–280.

[20] Y. S. Ramakrishna, P. M. Melliar-Smith, L. E. Moser, L. K. Dillon, and G. Kutty, *Interval logics and their decision procedures: part i: an interval logic*, Theor. Comput. Sci. **166** (1996), 1–47.

[21] Salamah Salamah, Ann Q. Gates, Vladik Kreinovich, and Steve Roach, *Verification of automatically generated pattern-based ltl specifications*, High-Assurance Systems Engineering, IEEE International Symposium on **0** (2007), 341–348.

[22] Salamah Ibrahim Salamah, *Generating linear temporal logic formulas for complex pattern-based specifications*, Ph.D. thesis, University of Texas, El Paso, Texas, 2007.

[23] Rachel L. Smith, George S. Avrunin, Lori A. Clarke, and Leon J. Osterweil, *Propel: an approach supporting property elucidation*, Proceedings of the 24th International Conference on Software Engineering (New York, NY, USA), ICSE '02, ACM, 2002, pp. 11–21.

[24] Willem Visser, Corina S. Păsăreanu, and Sarfraz Khurshid, *Test input generation with java pathfinder*, SIGSOFT Softw. Eng. Notes **29** (2004), 97–107.