When Discrete Met Continuous: on the integration of discrete- and continuous-time metric temporal logics

Carlo A. Furia and Matteo Rossi

October 2005

Abstract

Real-time systems usually encompass parts that are best described by a continuous-time model, such as physical processes under control, together with other components that are more naturally formalized by a discrete-time model, such as digital computing modules. Describing such systems in a unified framework based on metric temporal logic requires to integrate formulas which are interpreted over discrete and continuous time.

In this paper, we tackle this problem with reference to the metric temporal logic TRIO, that admits both a discrete-time and a continuous-time semantics. We identify sufficient conditions for a TRIO specification to be invariant under change of time model from discrete to continuous and vice versa. These conditions basically involve the restriction to a proper subset of the TRIO language (which we call $\mathrm{TRIO}_{\mathrm{si}}$) and a requirement on the finite variability over time of the values of the basic items that constitute the formulas of the specification. A specification which is invariant can then be verified entirely under the simpler discrete-time model, with the results of the verification holding for the continuous-time model as well.

We believe that this approach is general enough to be easily extendible to other temporal logics of comparable expressive power.

Keywords: Formal methods, real-time, integration, metric temporal logic, discrete time, continuous time

Contents

1 Introduction and Motivation

The application of formal methods to the description and analysis of really large systems inevitably requires being able to model different parts of the system using disparate notations and languages. In fact, it is usually the case that different modules are naturally described using diverse formal techniques, each one tailored to the specific nature of that component. Indeed, the past decades have seen the birth and proliferation of a plethora of different formal languages and techniques, each one usually focused on the description of a certain kind of systems and hinged on a specific approach. This proliferation is a good thing, as it permits the user to choose the notation and methodology that is best suited for his/her needs and that matches his/her intuition. However, this is also inevitably a hurdle to the true scalability in the application of formal techniques, since we end up having heterogeneous descriptions of large systems, where different modules, described using distinct notations, have no definite global semantics when put together. Therefore, we need to find ways to *integrate* dissimilar models into a global description which can then be analyzed, so that heterogeneity of notation is no more a limit but only a tool to enforce the principles of modularization and separation of concerns.

A particularly relevant instance of the above general problem is encountered when describing real-time systems, which require a quantitative modeling of time. Commonly, such systems are *hybrid*, that is composed of some parts representing physical environmental processes and some others being digital computing modules. The former ones have to model physical quantities that vary continuously over time, whereas the latter ones are digital components that are updated periodically at every (discrete) clock tick. Hence, a natural way to model the physical processes is by assuming a *continuous-time* model, and using a formalism with a compliant semantics, whereas digital components would be best described using a *discrete-time* model, and by adopting a formalism in accordance. Thus, the need to integrate continuous-time formalisms with discrete-time formalisms, which is the object of the present paper.

In particular, let us consider the framework of descriptive (a.k.a. declarative) specifications based on (metric) temporal logic. Some temporal logic languages have semantics for both a continuous-time model and a discrete-time one: each formula of the language can be interpreted in one of the two classes of models. TRIO is an example of these logics [7, 12, 13, 4], the one we are considering in this paper; MTL [11] is another well-known instance. So, apparently, it is possible to pass from a discrete-time to a continuous-time interpretation of a formula in these logics, achieving integration. The discrete-time semantics and the continuous-time one, however, are unrelated in general, in that the same formula changes completely its models when passing from one semantics to another. On the contrary, integration requires different formulas to describe parts of the *same* system, thus referring to unique underlying models.

To this end, we introduce the notion of *sampling invariance* of a specification formula. Informally, we say that a temporal logic formula is sampling invariant when its discrete-time models coincide with the samplings of all its continuous-time models (modulo some additional technical requirements). The *sampling* of a continuous-time model is a discrete-time model obtained by observing the continuous-time model at periodic instants of time. We claim that this notion mirrors effectively the condition required for a specification formula to be equivalently interpreted over a continuous-time or a discrete-time model: thus by building specifications made of sampling invariant formulas, we can integrate different time models, as their global meaning is consistent. The justification for the notion of sampling invariance stems from how real systems are made. In fact, in a typical hybrid system the discrete-time part (e.g. a controller) is connected to the (probed) environment by a sampler, which communicates measurements of some physical quantities to the controller at some periodic time rate (see Figure 1). The discrete-time behaviors that the controller sees are samplings of the continuous-time behaviors that occur in the system under control. Our notion of sampling invariance captures this fact in relating a continuous-time formula to a discrete-time one, thus mirroring what happens in a real system.

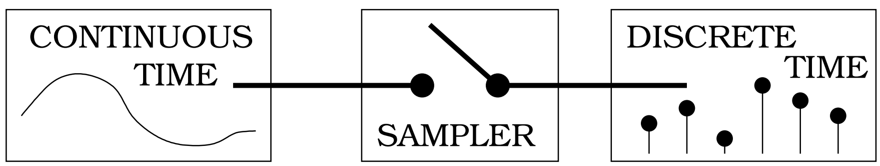

Figure 1: A hybrid system with a sampler.

Once we have a sampling invariant specification, we can integrate discrete-time and continuous-time parts, thus being able, among other things, to resort to verification in a discrete-time model, which can typically benefit from simpler and more automated approaches, while still being able to describe naturally physical processes in a continuous-time model.

Under this respect, Section 2 introduces $\mathrm{TRIO}_{\mathrm{si}}$, a suitable subset of TRIO for which sampling invariance can be achieved, and formally defines the sampling invariance requirement. Then, Section 3 proves that the above requirement is met with $\mathrm{TRIO}_{\mathrm{si}}$, according to some additional constraints. Finally, Section 4 compares our approach with related works, and Section 5 draws some conclusions.

2 Problem Statement and Definitions

In order to define precisely the sampling invariance condition, we have to introduce a metric temporal logic as well as its interpretations with continuous- and with discrete-time models. To this extent, Section 2.1 introduces $\mathrm{TRIO}_{\mathrm{si}}$, a subset of the TRIO language, as well as its discrete and continuous semantics. Then, Section 2.2 defines formally the concept of sampling invariance of a TRIO formula (Definition 1).

2.1 TRIO$_{\mathrm{si}}$: A Simple Metric Temporal Logic

The temporal logic language we are considering is TRIO. Full-fledged TRIO is a rich language with full first-order quantifications and arithmetic, augmented with modular constructs to build complex specifications [7, 13, 4]. However, for the purposes of the present work, we consider only a proper subset of it, which is strictly less expressive and which we call TRIO$_{\mathrm{si}}$ (for sampling invariant).

2.1.1 Syntax.

TRIO is based on a single modal operator named Dist. For a time-dependent formula A and a time distance t, $\mathrm{Dist}(A, t)$ is true iff A holds at a time which is t time instants apart from the current (implicit) one. TRIO$_{\mathrm{si}}$ is instead based on the two primitive temporal operators Until and Since, as well as the usual propositional connectives. For simplicity, we introduce TRIO$_{\mathrm{si}}$ as a purely propositional language, while leaving the discussion on how to extend it to be predicative to future work.

Let us define formally the syntax of TRIO$_{\mathrm{si}}$. Let Ξ be a set of time-dependent *conditions*. These are basically Boolean expressions obtained by functional combination of basic time-dependent items with constants. We are going to define them precisely later on (in Section 3), for now let us just assume that they are time-dependent formulas whose truth value is defined at any given time. Let us consider a set $\mathbb{S}$ of constants symbols. We denote *intervals* by expressions of the form $\langle l, u\rangle$, with l, u constants from $\mathbb{S}$, $\langle$ a left parenthesis from $\{(, [\}$, and $\rangle$ a right parenthesis from $\{),]\}$; let $\mathcal{I}$ be the set of all such intervals. Then, if $\xi, \xi_1, \xi_2 \in \Xi$, $I \in \mathcal{I}$, $\langle \in \{(, [\}$, and $\rangle \in \{),]\}$, well-formed formulas ϕ are defined recursively as follows.

$$\phi \quad ::= \quad \xi \mid \mathrm{Until}_{I\rangle}(\phi_1, \phi_2) \mid \mathrm{Since}_{I\langle}(\phi_1, \phi_2) \mid \neg\phi \mid \phi_1 \wedge \phi_2$$

From these basic operators, it is customary to define a number of *derived* operators: in Table 1 we define the most common ones.[1]

In the remainder, we will also need a notion of size of an interval, for both time models. To this end, we introduce here the $|\cdot|_{\mathbb{T}}$ operator, which is defined as follows, according to the time model. If the time domain is dense, then:

$$|\langle l, u\rangle|_{\mathbb{R}} = \begin{cases} u - l & \text{if } u \geq l \\ 0 & \text{if } u < l \end{cases}$$

[1] Implication $\Rightarrow$, disjunction $\vee$ and double implication $\Leftrightarrow$ are defined as usual. Moreover δ is a positive real constant that will be defined shortly. Finally, for simplicity we assume that $I = \langle l, u\rangle$ is such that $l, u \geq 0$ or $l, u \leq 0$ in the definition of the $\exists t \in I$ and $\forall t \in I$ operators.

Operator	Definition
$\text{Releases}_{I\rangle}(\phi_1, \phi_2)$	$\neg\text{Until}_{I\rangle}(\neg\phi_1, \neg\phi_2)$
$\text{Released}_{I\langle}(\phi_1, \phi_2)$	$\neg\text{Since}_{I\langle}(\neg\phi_1, \neg\phi_2)$
$\exists t \in I = \langle l, u\rangle : \text{Dist}(\phi, t)$	$\begin{cases} \text{Until}_{\langle l,u\rangle\rangle}(\text{true}, \phi) & \text{if } u \geq l \geq 0 \\ \text{Since}_{\langle -l,-u\rangle\rangle}(\text{true}, \phi) & \text{if } u \leq l \leq 0 \end{cases}$
$\forall t \in I = \langle l, u\rangle : \text{Dist}(\phi, t)$	$\begin{cases} \text{Releases}_{\langle l,u\rangle\rangle}(\text{false}, \phi) & \text{if } u \geq l \geq 0 \\ \text{Released}_{\langle -l,-u\rangle\rangle}(\text{false}, \phi) & \text{if } u \leq l \leq 0 \end{cases}$
$\text{Dist}(\phi, d)$	$\forall t \in [d, d] : \text{Dist}(\phi, t)$
$\text{Futr}(\phi, d)$	$d \geq 0 \wedge \text{Dist}(\phi, d)$
$\text{Past}(\phi, d)$	$d \geq 0 \wedge \text{Dist}(\phi, -d)$
$\text{SomF}(\phi)$	$\exists t \in (0, +\infty) : \text{Dist}(\phi, t)$
$\text{SomP}(\phi)$	$\exists t \in (-\infty, 0) : \text{Dist}(\phi, t)$
$\text{Som}(\phi)$	$\text{SomF}(\phi) \vee \phi \vee \text{SomP}(\phi)$
$\text{AlwF}(\phi)$	$\forall t \in (0, +\infty) : \text{Dist}(\phi, t)$
$\text{AlwP}(\phi)$	$\forall t \in (-\infty, 0) : \text{Dist}(\phi, t)$
$\text{Alw}(\phi)$	$\text{AlwF}(\phi) \wedge \phi \wedge \text{AlwP}(\phi)$
$\text{WithinF}(\phi, \tau)$	$\exists t \in (0, \tau) : \text{Dist}(\phi, t)$
$\text{WithinP}(\phi, \tau)$	$\exists t \in (-\tau, 0) : \text{Dist}(\phi, t)$
$\text{Within}(\phi, \tau)$	$\text{WithinF}(\phi, \tau) \vee \phi \vee \text{WithinP}(\phi, \tau)$
$\text{Lasts}(\phi, \tau)$	$\forall t \in (0, \tau) : \text{Dist}(\phi, t)$
$\text{Lasted}(\phi, \tau)$	$\forall t \in (-\tau, 0) : \text{Dist}(\phi, t)$
$\text{Until}(\phi_1, \phi_2)$	$\text{Until}_{(0,+\infty))}(\phi_1, \phi_2)$
$\text{Since}(\phi_1, \phi_2)$	$\text{Since}_{(0,+\infty)(}(\phi_1, \phi_2)$
$\text{NowOn}(\phi)$	$\begin{cases} \text{Lasts}(\phi, \delta) & \text{if the time domain is } \mathbb{R} \\ \text{Futr}(\phi, 1) & \text{if the time domain is } \mathbb{Z} \end{cases}$
$\text{UpToNow}(\phi)$	$\begin{cases} \text{Lasted}(\phi, \delta) & \text{if the time domain is } \mathbb{R} \\ \text{Past}(\phi, 1) & \text{if the time domain is } \mathbb{Z} \end{cases}$
$\text{Becomes}(\phi)$	$\text{UpToNow}(\neg\phi) \wedge \text{NowOn}(\phi)$

Table 1: TRIO derived temporal operators

If the time domain in instead discrete, we have the following definitions:

$$\begin{aligned} |(l, u\rangle|_{\mathbb{Z}} &= |[l+1, u\rangle|_{\mathbb{Z}} \\ |\langle l, u)|_{\mathbb{Z}} &= |\langle l, u-1]|_{\mathbb{Z}} \\ |[l, u]|_{\mathbb{Z}} &= \begin{cases} u - l + 1 & \text{if } u \geq l \\ 0 & \text{if } u < l \end{cases} \end{aligned}$$

2.1.2 Semantics.

In defining TRIO_{si} semantics we assume that constants symbols in $\mathbb{S}$ are interpreted naturally as numbers from the time domain $\mathbb{T}$ plus the symbols $\pm\infty$, which are treated as usual. Correspondingly, intervals $\mathcal{I}$ are interpreted as intervals of $\mathbb{T}$ which are closed/open to the left/right (as usual square brackets denote an included endpoint, and round brackets denote an excluded one).

Then, we define the semantics of a TRIO_{si} formula using as interpretations mappings from the time domain $\mathbb{T}$ to the domain D the basic items map their values to. Let $\mathcal{B}_{\mathbb{T}}$ be the set of all such mappings, which we call *behaviors*, and let $b \in \mathcal{B}_{\mathbb{T}}$ be any element from that set. If we denote by $\xi|_{b(t)}$ the truth value of the condition ξ at time $t \in \mathbb{T}$ according to behavior b, we can define the semantics of TRIO_{si} formulas as follows. We write $b \models_{\mathbb{T}} \phi$ to indicate that the behavior b is a model for formula ϕ under the time model $\mathbb{T}$. Thus, let us define the semantics for the generic time model $\mathbb{T}$.

$b(t) \models_{\mathbb{T}} \xi$	iff	$\xi|_{b(t)}$
$b(t) \models_{\mathbb{T}} \text{Until}_{I\rangle}(\phi_1, \phi_2)$	iff	there exists $d \in I$ such that $b(t+d) \models_{\mathbb{T}} \phi_2$ and, for all $u \in [0, d\rangle$ it is $b(t+u) \models_{\mathbb{T}} \phi_1$
$b(t) \models_{\mathbb{T}} \text{Since}_{I\langle}(\phi_1, \phi_2)$	iff	there exists $d \in I$ such that $b(t-d) \models_{\mathbb{T}} \phi_2$ and, for all $u \in \langle -d, 0]$ it is $b(t+u) \models_{\mathbb{T}} \phi_1$
$b(t) \models_{\mathbb{T}} \neg\phi$	iff	$b(t) \not\models_{\mathbb{T}} \phi$
$b(t) \models_{\mathbb{T}} \phi_1 \wedge \phi_2$	iff	$b(t) \models_{\mathbb{T}} \phi_1$ and $b(t) \models_{\mathbb{T}} \phi_2$
$b \models_{\mathbb{T}} \phi$	iff	for all $t \in \mathbb{T}$: $b(t) \models_{\mathbb{T}} \phi$

Thus, a TRIO_{si} formula ϕ constitutes the specification of a system, representing exactly all behaviors that are models of the formula. We denote by $[\![\phi]\!]_{\mathbb{T}}$ the set of all models of formula ϕ with time domain $\mathbb{T}$, i.e. $[\![\phi]\!]_{\mathbb{T}} \equiv \{b \in \mathcal{B}_{\mathbb{T}} | b \models_{\mathbb{T}} \phi\}$.

2.2 Sampled Behaviors and Sampling Invariance

We want to relate the models of a formula with a continuous-time domain to those of the same formula with a discrete-time domain. To do that, we introduce the notion of *sampling of a continuous-time behavior*. Basically, given a continuous-time behavior $b \in \mathcal{B}_{\mathbb{R}}$, we define its *sampling* as the discrete-time behavior $\sigma_{\delta,z}[b] \in \mathcal{B}_{\mathbb{Z}}$ that agrees with b at all integer time instants corresponding to multiples of a constant $\delta \in \mathbb{R}_{>0}$[2] from a basic offset $z \in \mathbb{R}$. We call δ the *sampling period* and z the *origin* of the sampling. More precisely, we have the following definition:

$$\forall k \in \mathbb{Z} : \quad \sigma_{\delta,z}[b](k) \equiv b(z + k\delta)$$

2.2.1 Sampling Invariance.

Now, given a TRIO_{si} formula ϕ we want to relate the set $[\![\phi]\!]_{\mathbb{Z}}$ of its discrete-time models to the set of behaviors obtained by sampling its continuous-time models $[\![\phi]\!]_{\mathbb{R}}$. Ideally, we would like that, for some sampling period δ and origin z, the following holds.

- For any continuous-time behavior $b \in \mathcal{B}_{\mathbb{R}}$ which is a model for ϕ, the sampling of b is a model for ϕ in discrete time, that is:

$$b \in [\![\phi]\!]_{\mathbb{R}} \quad \Rightarrow \quad \sigma_{\delta,z}[b] \in [\![\phi]\!]_{\mathbb{Z}}$$

[2]We use the symbol $\mathbb{R}_{>c}$, for some $c \in \mathbb{R}$, to denote the set $\{x \in \mathbb{R} \mid x > c\}$, and similar ones for integer numbers.

When this holds, we say that ϕ is *closed under sampling.*

- For any discrete-time behavior $b \in \mathcal{B}_{\mathbb{Z}}$ which is a model for ϕ, any continuous-time behavior, whose sampling coincides with b, is a model for ϕ in continuous time, that is:

$$b \in [\![\phi]\!]_{\mathbb{Z}} \quad \Rightarrow \quad \forall b' : (\sigma_{\delta,z}[b'] = b \;\Rightarrow\; b' \in [\![\phi]\!]_{\mathbb{R}})$$

When this holds, we say that ϕ is *closed under inverse sampling.*

When all the formulas of a language are closed both under sampling and under inverse sampling we say that the language is *sampling invariant.* We briefly note that it is possible to explore variations of the above definitions, for example by allowing to pick a sampling period and/or an origin which depend on the particular behavior being considered. We leave the discussion of the impact of these variations to future work.

2.2.2 On Continuous versus Discrete Semantics.

Actually, the above requirement is a too demanding one, as there are a number of facts indicating that it is not possible, in general, to achieve it, unless we adopt a logic language with such a limited expressiveness that it is of no practical use. In this subsection we briefly hint at some of these facts: notice that the exposition is deliberately loose, as we only want to motivate the developments that will follow.

Cardinality. The first fact is a simple cardinality argument, that reminds us that the mappings from $\mathbb{R}$ are "many more" than the mappings from $\mathbb{Z}$. In fact, for any domain D of size $d = |D|$, the cardinality of the set of all mappings $\mathbb{Z} \to D$ is $d^{\aleph_0}$, whereas the set of all mappings $\mathbb{R} \to D$ has the much larger cardinality $d^{\mathcal{C}} = d^{2^{\aleph_0}}$.

Time Units and Sampling Period. Another issue that needs to be dealt with is the problem of time units. In fact, when passing from a dense (and continuous) interpretation to a discrete one, we are implicitly assigning a different meaning to time units with respect to the sampling period. Consider, for instance, the $\mathrm{TRIO_{si}}$ formula $\mathrm{Lasts}(A, 1/2))$, where A is a primitive condition. In a discrete-time setting, we would like the formula to mean: for all (integer) time distances that fall in an interval between the two time instants corresponding to the sampling instants closer to 0 and 1/2, etc. Hence, we should actually adopt the formula $\mathrm{Lasts}(A, 1/2\delta)$ as discrete-time counterpart, dividing every *time constant* by the sampling period.

Items Velocity and Change Detection. A basic intuition that should be developed about the above idea of sampling is that, whenever the values of the basic items of our specification — whose evolution over time is described by behaviors — change "too fast" with respect to the chosen sampling period

δ, it is impossible to guarantee that those changes are "detected" at sampling instants, so that the properties of the behavior are preserved in the discrete-time setting.

Consider, for instance, the example of Figure 2. There, a Boolean-valued

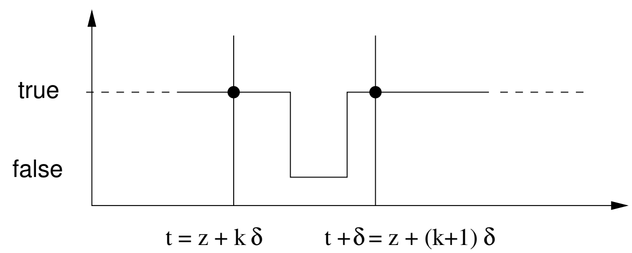

Figure 2: Change detection failure.

item changes its values twice (from true to false and then back to true) within the interval, of length δ, between two adjacent sampling instants. Therefore, any continuous-time formula predicating about the value of the item within those instants may be true in continuous time and false for the sampling of the behavior, which only "sees" two consecutive true values. Instead, we would like that the "rate of change" of the items values is slow-paced enough that every change in the value of an item is detected at some sampling instant, before it changes again.

2.2.3 Constrained Behaviors and Adaptation.

Under the above respect, we introduce three orders of mechanisms in order to obtain a sampling invariant language.

Normal syntactic form. We introduce a subset of $\mathrm{TRIO_{si}}$ in which sampling invariant formulas must be expressed. This means that we assume that every formula is in a suitable normal form. We will show that every $\mathrm{TRIO_{si}}$ formula can be put in normal form, but this may require to introduce auxiliary basic items.

Semantic constraint. We restrict the dynamics of the basic items in the specification to ensure that it is sufficiently "slow", with respect to the chosen sampling period, so that no change of the truth value of a formula can happen between two sampling instants without being "detected" at the next (or previous) sampling instant.

Adaptation function. We define an *adaptation function* which defines how the time bounds used in the formulas must be changed (by dividing or multiplying them by the sampling period, and by choosing suitable integer approximations) in order to pass from the interpretation under a time domain to another time domain while achieving sampling invariance. The adaptation function is in practice a simple translation function that implements the changes required to have compliant formulas.

In the remainder of this section, we are going to define precisely the above mechanisms.

Normal Form. TRIO_{si} formulas in *normal form* are written according to the following grammar.

$$\begin{array}{rcl} \phi & ::= & \xi \mid \phi_1 \vee \phi_2 \mid \phi_1 \wedge \phi_2 \mid \text{Until}_{I\rangle}(\xi_1, \xi_2) \mid \text{Since}_{I\langle}(\xi_1, \xi_2) \\ & & \text{Releases}_{I\rangle}(\xi_1, \xi_2) \mid \text{Released}_{I\langle}(\xi_1, \xi_2) \end{array}$$

Notice that the normal form prohibits nesting of temporal operators and negations of formulas. Moreover, we have an additional syntactic restriction which applies to discrete-time formulas only (that is, to formulas that are meant to be interpreted in discrete time): we require that the Until and Since operators always use an included boundary, while the Releases and Released operators always use an excluded boundary. In other words, we consider only discrete-time formulas using the temporal operators $\text{Until}_{I]}(\cdot)$, $\text{Since}_{I[}(\cdot)$, $\text{Releases}_{I)}(\cdot)$, and $\text{Released}_{I(}(\cdot)$.

Despite the above restrictions, any TRIO_{si} formula can be expressed with this TRIO_{si} language fragment, with the additional requirement that we are allowed to introduce new auxiliary basic items in the specification, as we are going to show shortly. First, for the ease of exposition of the following proofs, let us introduce explicitly the semantics for the operators of the normal form which are not basic TRIO_{si} operators, namely Releases, Released, and $\vee$.

$b(t) \models_{\mathbb{T}} \text{Releases}_{I\rangle}(\xi_1, \xi_2)$	iff	for all $d \in I$ it is either $b(t+d) \models_{\mathbb{T}} \xi_2$ or there exists $u \in [0, d\rangle$ such that $b(t+u) \models_{\mathbb{T}} \xi_1$
$b(t) \models_{\mathbb{T}} \text{Released}_{I\langle}(\xi_1, \xi_2)$	iff	for all $d \in I$ it is either $b(t-d) \models_{\mathbb{T}} \xi_2$ or there exists $u \in \langle -d, 0]$ such that $b(t+u) \models_{\mathbb{T}} \xi_1$
$b(t) \models_{\mathbb{T}} \phi_1 \vee \phi_2$	iff	$b(t) \models_{\mathbb{T}} \phi_1$ or $b(t) \models_{\mathbb{T}} \phi_2$

Now we show how any TRIO_{si} formula can be expressed using just the above operators.

Elimination of negations. In order to eliminate negations in a formula, we just push them inward to boolean conditions. Since conditions are closed under complement (as it will be clear in their definition, see Section 3), and we have the dual of each basic operator, we can always eliminate negations from formulas. More explicitly, we perform the substitutions:

$$\begin{array}{lcl} \neg(\neg\phi) & \longrightarrow & \phi \\ \neg(\phi_1 \wedge \phi_2) & \longrightarrow & \neg\phi_1 \vee \neg\phi_2 \\ \neg\text{Until}_{I\rangle}(\phi_1, \phi_2) & \longrightarrow & \text{Releases}_{I\rangle}(\neg\phi_1, \neg\phi_2) \\ \neg\text{Since}_{I\langle}(\phi_1, \phi_2) & \longrightarrow & \text{Released}_{I\langle}(\neg\phi_1, \neg\phi_2) \end{array}$$

recursively, until all negations are eliminated or on conditions.

Elimination of nesting. Nesting can be eliminated from TRIO_{si} by introducing auxiliary basic items (which are primitive conditions). More explicitly, if ξ is a primitive condition, ξ' is an auxiliary item (not used anywhere else in the specification), ϕ is a generic formula, and TempOper is any temporal operator (i.e., Until, Since, Releases, or Released), we perform the substitutions:

$$\begin{array}{lcl} \text{TempOper}_{I\rangle}(\phi, \xi) & \longrightarrow & (\xi' \Leftrightarrow \phi) \ \wedge \ \text{TempOper}_{I\rangle}(\xi', \xi) \\ \text{TempOper}_{I\rangle}(\xi, \phi) & \longrightarrow & (\xi' \Leftrightarrow \phi) \ \wedge \ \text{TempOper}_{I\rangle}(\xi, \xi') \end{array}$$

recursively, until nestings of temporal operators are eliminated. It is simple to check, by direct application of the definition of the operators, that the above substitutions preserve the semantics of the formula. Exactly, one can check that if $\xi' \Leftrightarrow \phi$ then $\text{TempOper}_{I\rangle}(\phi, \xi) \equiv \text{TempOper}_{I\rangle}(\xi', \xi)$ and $\text{TempOper}_{I\rangle}(\xi, \phi) \equiv \text{TempOper}_{I\rangle}(\xi, \xi')$.

Finally, one has obviously to eliminate the double implications by introducing the corresponding definitions: $\xi' \Leftrightarrow \phi \equiv (\xi' \wedge \phi) \vee (\neg\xi' \wedge \neg\phi)$; the negations which are introduced in the process are then eliminated as explained in the above step.

Elimination of $\text{Until}_{)}$ **and** $\text{Releases}_{]}$ **in discrete time.** Let us first note the following equivalences. If ξ_1, ξ_2 are any conditions, and ξ' is an auxiliary item (not used anywhere else in the specification), then:

$$\begin{array}{lcl} (\xi' \Leftrightarrow \text{NowOn}(\xi_2)) & \Rightarrow & \big(\text{Until}_{\langle l,u\rangle)}(\xi_1, \xi_2) \equiv \text{Until}_{\langle l-1,u-1\rangle]}(\xi_1, \xi')\big) \\ (\xi' \Leftrightarrow \text{UpToNow}(\xi_2)) & \Rightarrow & \big(\text{Since}_{\langle l,u\rangle(}(\xi_1, \xi_2) \equiv \text{Since}_{\langle l-1,u-1\rangle[}(\xi_1, \xi')\big) \\ (\xi' \Leftrightarrow \text{UpToNow}(\xi_2)) & \Rightarrow & \big(\text{Releases}_{\langle l,u\rangle]}(\xi_1, \xi_2) \equiv \text{Releases}_{\langle l+1,u+1\rangle)}(\xi_1, \xi')\big) \\ (\xi' \Leftrightarrow \text{NowOn}(\xi_2)) & \Rightarrow & \big(\text{Released}_{\langle l,u\langle[}(\xi_1, \xi_2) \equiv \text{Released}_{\langle l+1,u+1\langle(}(\xi_1, \xi')\big) \end{array}$$

Let us sketch the proof of the above equivalences, for a generic discrete-time behavior $b \in \mathcal{B}_{\mathbb{Z}}$.

- *Until and Since.* Assume that $b \models_{\mathbb{Z}} \xi' \Leftrightarrow \text{NowOn}(\xi_2)$. Then, if $b(k) \models_{\mathbb{Z}} \text{Until}_{\langle l,u\rangle)}(\xi_1, \xi_2)$ for some instant k, then there exists a $d \in \langle l, u\rangle$ such that ξ_2 is true at $k + d$ and ξ_1 is true in the whole interval $[0, d)$ from k, which in discrete time is equivalent to $[0, d - 1]$. Thus, for $d' = d - 1$, we have that ξ' is true at $k + d'$, since ξ_2 is true at $k + d = k + d' + 1$. Notice that $d' \in \langle l - 1, u - 1\rangle$. All in all we have that $b(k) \models_{\mathbb{Z}} \text{Until}_{\langle l-1,u-1\rangle]}(\xi_1, \xi')$. For the converse, assume that $b(k) \models_{\mathbb{Z}} \text{Until}_{\langle l-1,u-1\rangle]}(\xi_1, \xi')$. So, there exists a $d \in \langle l - 1, u - 1\rangle$ such that ξ' holds at $k + d$ and ξ_1 holds in the whole interval $[0, d]$ from k. Therefore, ξ_2 holds at $k + d + 1$, and $d + 1 \in \langle l, u\rangle$. Hence, for $d' = d + 1$ we have that $b(k) \models_{\mathbb{Z}} \text{Until}_{I\rangle}(\xi_1, \xi_2)$ holds.

  The proof for the Since is all similar, but for the past direction of time.

- *Releases and Released.* The equivalences follow from the above equivalence about the Until operator, exploiting the fact that the Releases is its dual, as well as the easily verifiable equivalences of $\neg\text{NowOn}(\xi)$ with $\text{NowOn}(\neg\xi)$, and of $\xi' \Leftrightarrow \text{NowOn}(\xi'')$ with $\xi'' \Leftrightarrow \text{UpToNow}(\xi')$.

All similarly for the Released, but derived from the property of the Since.

Therefore, we perform the following substitutions to put a formula into normal form.

$$
\begin{array}{lcl}
\mathrm{Until}_{\langle l,u\rangle)}(\xi_1,\xi_2) & \longrightarrow & (\xi' \Leftrightarrow \mathrm{NowOn}(\xi_2)) \;\wedge\; \mathrm{Until}_{\langle l-1,u-1\rangle]}(\xi_1,\xi') \\
\mathrm{Since}_{\langle l,u\rangle(}(\xi_1,\xi_2) & \longrightarrow & (\xi' \Leftrightarrow \mathrm{UpToNow}(\xi_2)) \;\wedge\; \mathrm{Since}_{\langle l-1,u-1\rangle[}(\xi_1,\xi') \\
\mathrm{Releases}_{\langle l,u\rangle]}(\xi_1,\xi_2) & \longrightarrow & (\xi' \Leftrightarrow \mathrm{UpToNow}(\xi_2)) \;\wedge\; \mathrm{Releases}_{\langle l+1,u+1\rangle)}(\xi_1,\xi') \\
\mathrm{Released}_{\langle l,u\rangle[}(\xi_1,\xi_2) & \longrightarrow & (\xi' \Leftrightarrow \mathrm{NowOn}(\xi_2)) \;\wedge\; \mathrm{Releases}_{\langle l+1,u+1\rangle(}(\xi_1,\xi')
\end{array}
$$

Semantic Constraint. The precise form of the semantic constraint to be introduced depends on the kind of items we are dealing with in our specification, and namely whether they take values to discrete or dense domains. We are going to discuss separately the two cases in Section 3. In a nutshell, we constrain the dynamics of the basic items and/or conditions of the specification by requiring that at least δ time units (where δ is the chosen sampling period) elapse between each pair of instants of time at which the items change their values. This restriction — which mandates a "maximum rate of change" of the items' values — assures that every change in the truth value of a formula is propagated until either the previous or the next sampling instant, before being possibly reversed by another change. Therefore, there is no information loss by observing the system just at the sampling instants, where by *information* we (informally) mean the truth value of a specification formula (which is the only description of the system that we consider).

In practice, the semantic constraint is expressed by an additional $\mathrm{TRIO_{si}}$ formula χ, which in general depends on the particular specification we have written. Thus, whenever we write a specification ϕ, we are actually considering the (more restrictive) specification $\phi \wedge \chi$, which only has models with the desired qualities. We can see the semantic constraint χ as a price that we have to pay in order to have a sampling invariant specification. Clearly, whenever the specification already entails the semantic constraint, sampling invariance comes simply at no price. In future work, we will discuss how the semantic constraint impacts on the description of real systems, as well as on the limitations it imposes on the properties that a system can have.

Adaptation Function. We define an *adaptation* function $\eta_\delta^{\mathbb{R}}\{\cdot\}$ which translates formulas by scaling the time constants, appearing as endpoints of the intervals, to their values divided by the sampling period δ. We also define its "inverse" $\eta_\delta^{\mathbb{Z}}\{\cdot\}$, which translates valid discrete-time formulas to valid continuous-time ones, by scaling back the endpoints. Notice that, in discrete time, it suffices to define the adaptation rule for closed intervals. Thus, for any formula respecting the all of the above syntactic restrictions, adaptation is defined formally inductively on the structure of the formulas of TRIO_{si} as follows.

$$
\begin{array}{lcl}
\eta_\delta^{\mathbb{R}}\{\xi\} & \equiv & \xi \\
\eta_\delta^{\mathbb{R}}\left\{\text{Until}_{\langle l,u\rangle\rangle}(\phi_1,\phi_2)\right\} & \equiv & \text{Until}_{[\lfloor l/\delta\rfloor,\lceil u/\delta\rceil])}\left(\eta_\delta^{\mathbb{R}}\{\phi_1\},\eta_\delta^{\mathbb{R}}\{\phi_2\}\right) \\
\eta_\delta^{\mathbb{R}}\left\{\text{Since}_{\langle l,u\rangle\langle}(\phi_1,\phi_2)\right\} & \equiv & \text{Since}_{[\lfloor l/\delta\rfloor,\lceil u/\delta\rceil](}\left(\eta_\delta^{\mathbb{R}}\{\phi_1\},\eta_\delta^{\mathbb{R}}\{\phi_2\}\right) \\
\eta_\delta^{\mathbb{R}}\left\{\text{Releases}_{\langle l,u\rangle\rangle}(\phi_1,\phi_2)\right\} & \equiv & \text{Releases}_{\langle l',u'\rangle]}\left(\eta_\delta^{\mathbb{R}}\{\phi_1\},\eta_\delta^{\mathbb{R}}\{\phi_2\}\right) \\
 & & \text{where } l' = \begin{cases} \lfloor l/\delta\rfloor & \text{if } \langle \text{ is } (\\ \lceil l/\delta\rceil & \text{if } \langle \text{ is } [\end{cases} \\
 & & \text{and } u' = \begin{cases} \lceil u/\delta\rceil & \text{if } \rangle \text{ is }) \\ \lfloor u/\delta\rfloor & \text{if } \rangle \text{ is }] \end{cases} \\
\eta_\delta^{\mathbb{R}}\left\{\text{Released}_{\langle l,u\rangle\langle}(\phi_1,\phi_2)\right\} & \equiv & \text{Released}_{\langle l',u'\rangle[}\left(\eta_\delta^{\mathbb{R}}\{\phi_1\},\eta_\delta^{\mathbb{R}}\{\phi_2\}\right) \\
 & & \text{where } l' = \begin{cases} \lfloor l/\delta\rfloor & \text{if } \langle \text{ is } (\\ \lceil l/\delta\rceil & \text{if } \langle \text{ is } [\end{cases} \\
 & & \text{and } u' = \begin{cases} \lceil u/\delta\rceil & \text{if } \rangle \text{ is }) \\ \lfloor u/\delta\rfloor & \text{if } \rangle \text{ is }] \end{cases} \\
\eta_\delta^{\mathbb{R}}\{\phi_1 \wedge \phi_2\} & \equiv & \eta_\delta^{\mathbb{R}}\{\phi_1\} \wedge \eta_\delta^{\mathbb{R}}\{\phi_2\} \\
\eta_\delta^{\mathbb{R}}\{\phi_1 \vee \phi_2\} & \equiv & \eta_\delta^{\mathbb{R}}\{\phi_1\} \vee \eta_\delta^{\mathbb{R}}\{\phi_2\}
\end{array}
$$

Notice that in the discrete-to-continuous adaptation of the Until and Since operators, the adapted interval $\langle (l-1)\delta, (u-1)\delta\rangle$ can indifferently be taken to include or exclude its endpoints.

$$
\begin{array}{lcl}
\eta_\delta^{\mathbb{Z}}\{\xi\} & \equiv & \xi \\
\eta_\delta^{\mathbb{Z}}\left\{\text{Until}_{[l,u]]}(\phi_1,\phi_2)\right\} & \equiv & \text{Until}_{\langle (l-1)\delta,(u+1)\delta\rangle]}\left(\eta_\delta^{\mathbb{Z}}\{\phi_1\},\eta_\delta^{\mathbb{Z}}\{\phi_2\}\right) \\
\eta_\delta^{\mathbb{Z}}\left\{\text{Since}_{[l,u][}(\phi_1,\phi_2)\right\} & \equiv & \text{Since}_{\langle (l-1)\delta,(u+1)\delta\rangle[}\left(\eta_\delta^{\mathbb{Z}}\{\phi_1\},\eta_\delta^{\mathbb{Z}}\{\phi_2\}\right) \\
\eta_\delta^{\mathbb{Z}}\left\{\text{Releases}_{[l,u])}(\phi_1,\phi_2)\right\} & \equiv & \text{Releases}_{[(l+1)\delta,(u-1)\delta])}\left(\eta_\delta^{\mathbb{Z}}\{\phi_1\},\eta_\delta^{\mathbb{Z}}\{\phi_2\}\right) \\
\eta_\delta^{\mathbb{Z}}\left\{\text{Released}_{[l,u](}(\phi_1,\phi_2)\right\} & \equiv & \text{Released}_{[(l+1)\delta,(u-1)\delta](}\left(\eta_\delta^{\mathbb{Z}}\{\phi_1\},\eta_\delta^{\mathbb{Z}}\{\phi_2\}\right) \\
\eta_\delta^{\mathbb{Z}}\{\phi_1 \wedge \phi_2\} & \equiv & \eta_\delta^{\mathbb{Z}}\{\phi_1\} \wedge \eta_\delta^{\mathbb{Z}}\{\phi_2\} \\
\eta_\delta^{\mathbb{Z}}\{\phi_1 \vee \phi_2\} & \equiv & \eta_\delta^{\mathbb{Z}}\{\phi_1\} \vee \eta_\delta^{\mathbb{Z}}\{\phi_2\}
\end{array}
$$

2.2.4 Sampling Invariance: A Definition.

Finally, according to the above ideas, we can formulate a definition of sampling invariance which is the one we are actually going to use in the remainder.

Definition 1 (Sampling Invariance). Given a formula ϕ, a behavior constraint formula χ, two adaptation functions $\eta_\delta^{\mathbb{R}}\{\cdot\}$ and $\eta_\delta^{\mathbb{Z}}\{\cdot\}$, a sampling period δ, and an origin z, we say that:

- ϕ is *closed under sampling* iff for any continuous-time behavior $b \in \mathcal{B}_{\mathbb{R}}$:

$$b \in [\![\phi \wedge \chi]\!]_{\mathbb{R}} \quad \Rightarrow \quad \sigma_{\delta,z}[b] \in [\![\eta_\delta^{\mathbb{R}}\{\phi\}]\!]_{\mathbb{Z}}$$

- ϕ is *closed under inverse sampling* iff for any discrete-time behavior $b \in \mathcal{B}_{\mathbb{Z}}$:

$$b \in [\![\phi]\!]_{\mathbb{Z}} \quad \Rightarrow \quad \forall b' \in [\![\chi]\!]_{\mathbb{R}} : \left(\sigma_{\delta,z}[b'] = b \ \Rightarrow\ b' \in [\![\eta_\delta^{\mathbb{Z}}\{\phi\} \wedge \chi]\!]_{\mathbb{R}}\right)$$

- A language is *sampling invariant* iff all the formulas of the language are closed under sampling (when interpreted in the continuous-time domain) and are closed under inverse sampling (when interpreted in the discrete-time domain).

3 Sampling Invariant Specifications

In this section we formulate a sufficient condition for the sampling invariance of a TRIO_{si} specification formula ϕ. This condition involves the definition of a suitable constraint formula χ on the continuous-time behaviors, as well as a definition of the forms of the conditions ξ that can appear in TRIO_{si} formulas. More precisely, we distinguish two cases, whether we are dealing with time-dependent items which take values onto a discrete set, or with time-dependent items which take values onto a dense set.

Let us introduce this idea with more precision. We assume that every specification is built out of primitive time-dependent items from a (finite) set Ψ. For example one such item may represent — by a Boolean value — the state of a light bulb (on or off), another one may instead represent — by a real value — the measure of the temperature in a room, etc. So, every time-dependent item ψ_i in Ψ is a mapping from time $\mathbb{T}$ to a suitable domain D_i, for $i = 1, \ldots, n$, where n is the number of primitive time-dependent items. Therefore, every behavior $b \in \mathcal{B}_{\mathbb{T}}$ represents the evolution over time of the values of *all* primitive items; in other words, every behavior $b \in \mathcal{B}_{\mathbb{T}}$ is a mapping from time $\mathbb{T}$ to the domain $D \equiv D_1 \times \cdots \times D_n$. Note that we do not consider the definition of more complicated time-dependent items, such as time-dependent functions, which are part of standard TRIO. These would not be difficult to introduce, but we leave them out of the present work, in favor of a simpler exposition. The next two subsections will draw sufficient conditions for sampling invariance in the two cases of when every D_i is a discrete set (Section 3.1) and when some D_i is a dense set (Section 3.2).

3.1 Discrete-valued Items

Let us consider the case in which all time-dependent items in Ψ have discrete codomains, i.e. ψ_i is a time-dependent item taking values to the discrete set D_i, for all $i = 1, \ldots, n$. Thus, behaviors are mappings $\mathbb{T} \to D$ where $D \equiv D_1 \times \cdots \times D_n$. We also assume that a total order is defined on each D_i. Although this assumption could be avoided, we introduce it for uniformity of presentation; still, one can easily extend the present exposition to unordered sets as well.

Conditions. Let Λ be a set of constants from the sets D_i's, and Γ be a set of functions[3] having domains in subsets of D and codomains in some D_i's. Then, if $\lambda \in \Lambda$, $f, f_1, f_2 \in \Gamma$, and $\bowtie \in \{=, <, \leq, >, \geq\}$, conditions ξ can be defined recursively as follows, assuming type compatibility is respected:

[3]These are of course time-independent functions, in TRIO terms.

$$\xi \ ::= \ f(\psi_1, \ldots, \psi_n) \bowtie \lambda \mid f_1(\psi_1, \ldots, \psi_n) \bowtie f_2(\psi_1, \ldots, \psi_n) \mid \neg\xi \mid \xi_1 \wedge \xi_2$$

Abbreviations and notational conventions are introduced in the obvious ways.

Constraint on Behaviors. The constraint on behaviors is expressed by Formula $\chi_\circ$ and basically requires that the values of the items in Ψ vary over time in such a way that changes happen at most every δ time units, i.e. the value of each item is held for δ, at least. This can be expressed formally with the following TRIO_{si}[4] formula.[5]

$$\chi_\circ \triangleq \ \ \forall v \in D: \ (\langle \psi_1, \ldots, \psi_n \rangle = v \ \Rightarrow \ \text{WithinP}_{\text{ii}}(\text{Lasts}_{\text{ii}}(\langle \psi_1, \ldots, \psi_n \rangle = v, \delta)\,, \delta))$$

We notice that, in particular, $\chi_\circ$ rules out any Zeno behavior [6] of the basic items of a specification. More precisely, we notice the following fact that we will use in the following.

Lemma 2 (Items Change Points). *For any behavior obeying the constraint $\chi_\circ$, if for some time $t \in \mathbb{R}$ it is $b(t) \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle = v$ for some v, then there exists $c_n \geq t$ and $c_p \leq t$ such that:*

- $b(c_n) \models_{\mathbb{R}} \text{Becomes}(\langle \psi_1, \ldots, \psi_n \rangle \neq v)$ *or* $b(t) \models_{\mathbb{R}} \text{AlwF}(\langle \psi_1, \ldots, \psi_n \rangle = v)$*;*
- $b(c_p) \models_{\mathbb{R}} \text{Becomes}(\langle \psi_1, \ldots, \psi_n \rangle = v)$ *or* $b(t) \models_{\mathbb{R}} \text{AlwP}(\langle \psi_1, \ldots, \psi_n \rangle = v)$
- *for all* $t' \in (c_p, c_n)$ *it is* $b(t') \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle = v$*;*
- $c_n - c_p \geq \delta$*.*

Moreover, the same property holds for any condition ξ*.*

Proof. Since $\chi_\circ$ holds for b, then there exists a $p \in [0, \delta]$ such that for all $u \in [-p, -p+\delta]$ it is $b(t+u) \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle = v$. In particular, it is $b(t-p) \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle = v$ and $b(t - p + \delta) \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle = v$. Since $\chi_\circ$ implies the non-Zenoness of the items values, then [6] there exist a $d_p, d_n \geq 0$ such that:

- either for all $u' \in [0, d_n)$ it is $b(t-p+\delta+u') \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle = v$, and there exists an $\epsilon_n > 0$ such that for all $u'' \in (0, \epsilon_n)$ it is $b(t-p+\delta+d_n+u'') \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle \neq v$, or $\langle \psi_1, \ldots, \psi_n \rangle = v$ always in the future;
- either for all $u' \in [0, d_p)$ it is $b(t - p - u') \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle = v$, and there exists an $\epsilon_p > 0$ such that for all $u'' \in (0, \epsilon_n)$ it is $b(t - p - d_n - u'') \models_{\mathbb{R}} \langle \psi_1, \ldots, \psi_n \rangle \neq v$, or $\langle \psi_1, \ldots, \psi_n \rangle = v$ always in the past.

[4] Actually, TRIO_{si} as we defined it above is purely propositional, while $\chi_\circ$ uses a universal quantification on non-temporal variables. However, later it will be shown why this modification is a natural and acceptable generalization which is orthogonal to sampling invariance.

[5] $\text{WithinP}_{\text{ii}}(\phi, \tau) \triangleq \exists u \in [-\tau, 0] : \text{Dist}(\phi, u)$, and $\text{Lasts}_{\text{ii}}(\phi, \tau) \triangleq \forall u \in [0, \tau] : \text{Dist}(\phi, u)$.

In other words, either there is a point where the items change (some of) their values, or they keep their current value indefinitely.

If the latter is the case, then the lemma follows immediately. Otherwise, notice that $(-p+\delta+d_n)-(-p) = \delta+d_n \geq \delta$, therefore we have $b(t-p+\delta+d_n) \models_{\mathbb{R}} \text{UpToNow}(\langle \psi_1, \ldots, \psi_n \rangle = v)$. Similarly, notice that $-((-p-d_p)-(-p+\delta)) = d_p + \delta \geq \delta$, therefore $b(t-p-d_p) \models_{\mathbb{R}} \text{NowOn}(\langle \psi_1, \ldots, \psi_n \rangle = v)$.

Finally, by considering condition $\chi_\circ$ again for the instants $t-p+\delta+d_n+\epsilon'$ and $t-p-d_p-\epsilon'$ "to the limit" as ϵ goes to 0, then we realize that it must also be $b(t-p+\delta+d_n) \models_{\mathbb{R}} \text{NowOn}(\langle \psi_1, \ldots, \psi_n \rangle \neq v)$ and $b(t-p-d_p) \models_{\mathbb{R}} \text{UpToNow}(\langle \psi_1, \ldots, \psi_n \rangle \neq v)$.

Therefore, for $c_n = t-p+\delta+d_n$ and $c_p = t-p-d_p$ the lemma holds.

Finally, since the value of a condition changes only if some values change their values, the lemma holds immediately for conditions as well. $\square$

A Sufficient Condition for Sampling Invariance. We finally prove a sufficient condition for sampling invariance of TRIO_{si} formulas in the following.

Theorem 3 (Sampling Invariance for Discrete-valued Items). *Normal-form $\mathit{TRIO}_{\text{si}}$ is sampling invariant, for items mapping to a* discrete *set D, with respect to the behavior constraint $\chi_\circ$, the adaptation functions $\eta_\delta^{\mathbb{R}}\{\cdot\}$ and $\eta_\delta^{\mathbb{Z}}\{\cdot\}$, for any sampling period δ and origin z.*

Proof. The proof is split into two main parts: first we show that any TRIO_{si} formula ϕ, interpreted in the continuous-time domain, is closed under sampling; then we show that any formula ϕ, interpreted in the discrete-time domain, is closed under inverse sampling. We assume formulas written in the normal form presented above.

For ease of exposition, let us also introduce the following abbreviations: for a real number r, let us denote by $\Omega(r)$ the sampling instant $z + \lfloor (r-z)/\delta \rfloor \delta$, which is immediately before r, and by $O(r)$ the sampling instant $z + \lceil (r-z)/\delta \rceil \delta$ which is immediately after r. Moreover, we also denote by $\omega(r)$ and $o(r)$ the distances between r and its previous and next sampling instant, respectively, that is $\omega(r) = r - \Omega(r)$ and $o(r) = O(r) - r$. Obviously $\omega(r), o(r) \geq 0$.

(Closure under sampling).[6] Let b be a continuous-time behavior in $[\![\chi_\circ]\!]_{\mathbb{R}}$ and let $\phi' = \eta_\delta^{\mathbb{R}}\{\phi\}$. Then, let b' be the sampling $\sigma_{\delta,z}[b]$ of behavior b with the given origin and sampling period.

Now, for a generic sampling instant $t = z + k\delta$, we show that $b(t) \models_{\mathbb{R}} \phi$ implies $b'(k) \models_{\mathbb{Z}} \phi'$, by induction on the structure of ϕ.

- $\phi = \xi$.
  By definition of sampling of a behavior, the truth value of ξ at t and k is the same, i.e. $\xi|_{b(t)} = \xi|_{b(z+k\delta)} = \xi|_{b'(k)}$.

[6] This proof exploits some properties of the floor and ceiling functions. We refer the reader to [8] for a thorough treatment of these functions.

- $\phi = \text{Until}_{\langle l,u\rangle\rangle}(\xi_1, \xi_2)$.
  Notice that ϕ' is $\text{Until}_{[l',u'])}(\xi_1, \xi_2)$, with $l' = \lfloor l/\delta \rfloor$ and $u' = \lceil u/\delta \rceil$.

  Let d be a time instant in $\langle l, u\rangle$ such that $b(t+d) \models_{\mathbb{R}} \xi_2$ and, for all time instants $e \in [0, d\rangle$ it is $b(t+e) \models_{\mathbb{R}} \xi_1$. Condition $\chi_\circ$ at time $t+d$ implies that there exists a $p \in [0, \delta]$ such that for all $f \in [-p, -p+\delta]$ it is $b(t+d+f) \models_{\mathbb{R}} \xi_2$. In other words, ξ_2 holds on the closed real interval $I = [t+d-p, t+d-p+\delta]$. Since I has size δ, its intersection with the sampling points (which are δ time units apart) must be non-empty. In particular, it is either $p \geq \omega(t+d)$ or $-p+\delta \geq o(t+d)$: otherwise it would be $\delta = p + (-p+\delta) < \omega(t+d) + o(t+d) = O(t+d) - \Omega(t+d) = \delta(\lceil (t+d-z)/\delta \rceil - \lfloor (t+d-z)/\delta \rfloor) \leq \delta$, a contradiction (where we exploited the property: $\lceil r \rceil - \lfloor r \rfloor \leq 1$ for any real r).

  So, let t' be the sampling instant:

  $$t' = \begin{cases} \Omega(t+d) & \text{if } p \geq \omega(t+d) \\ O(t+d) & \text{otherwise} \end{cases}$$

  It is not difficult to check that $(t'-t)/\delta \in [l', u']$. In fact:

  - if $p \geq \omega(t+d)$, then $t'-t = \Omega(t+d)-t = \delta(\lfloor (k\delta+d)/\delta \rfloor - k) = \delta \lfloor d/\delta \rfloor$. Recall that $d \in \langle l, u\rangle$, and then *a fortiori* $d \in [l, u] \supseteq \langle l, u\rangle$. So $d/\delta \in [l/\delta, u/\delta]$, and $(t'-t)/\delta = \lfloor d/\delta \rfloor \in [\lfloor l/\delta \rfloor, \lfloor u/\delta \rfloor] \subseteq [l', u']$.
  - if $p < \omega(t+d)$, then $t'-t = O(t+d)-t = \delta(\lceil (k\delta+d)/\delta \rceil - k) = \delta \lceil d/\delta \rceil$. Recall that $d \in \langle l, u\rangle$, and then *a fortiori* $d \in [l, u] \supseteq \langle l, u\rangle$. So $d/\delta \in [l/\delta, u/\delta]$, and $(t'-t)/\delta = \lceil d/\delta \rceil \in [\lceil l/\delta \rceil, \lceil u/\delta \rceil] \subseteq [l', u']$.

  So far, we have shown that $b(t') \models_{\mathbb{R}} \xi_2$. By inductive hypothesis, it follows that for $d' = (t'-t)/\delta$ it is $b'(k+d') \models_{\mathbb{Z}} \xi_2$. Since $d' \in [l', u']$, to conclude this branch of the proof we have to show that for all $e' \in [0, d'-1]$ it is $b'(k+e') \models_{\mathbb{Z}} \xi_1$. To this end, we just have to realize that $\delta(d'-1) < d$. In fact, we have shown above that $d' \leq \lceil d/\delta \rceil < d/\delta + 1$, since $\lceil r \rceil < r+1$ for any real number r. So obviously $\delta(d'-1) < \delta(d/\delta) = d$. Since for all $e \in [0, d\rangle$ we have $b(t+e) \models_{\mathbb{R}} \xi_1$, and since $[0, \delta(d'-1)] \subset [0, d\rangle$, then *a fortiori* for all $e \in [0, \delta(d'-1)]$ it is $b(t+e) \models_{\mathbb{R}} \xi_1$. Finally, by inductive hypothesis, it follows that for all integers $e' \in [0, d'-1] = [0, d')$ it is $b'(k+e') \models_{\mathbb{Z}} \xi_1$, which lets us conclude that $b'(k) \models_{\mathbb{Z}} \phi'$.

- $\phi = \text{Since}_{\langle l,u\rangle\langle}(\xi_1, \xi_2)$.
  Notice that ϕ' is $\text{Since}_{[l',u'](}(\xi_1, \xi_2)$, with $l' = \lfloor l/\delta \rfloor$ and $u' = \lceil u/\delta \rceil$.

  Let d be a time instant in $\langle l, u\rangle$ such that $b(t-d) \models_{\mathbb{R}} \xi_2$ and, for all time instants $e \in \langle -d, 0]$ it is $b(t+e) \models_{\mathbb{R}} \xi_1$. Condition $\chi_\circ$ at time $t-d$ implies that there exists a $p \in [0, \delta]$ such that for all $f \in [-p, -p+\delta]$ it is $b(t-d+f) \models_{\mathbb{R}} \xi_2$. In other words, ξ_2 holds on the closed real interval $I = [t-d-p, t-d-p+\delta]$. Since I has size δ, its intersection with the sampling points (which are δ time units apart) must be non-empty. In particular, it is either $p \geq \omega(t-d)$ or $-p+\delta \geq o(t-d)$: otherwise it would be $\delta = p + (-p + \delta) < \omega(t - d) + o(t - d) = O(t - d) - \Omega(t - d) = \delta(\lceil (t-d-z)/\delta \rceil - \lfloor (t-d-z)/\delta \rfloor) \leq \delta$, a contradiction (where we exploited the property: $\lceil r \rceil - \lfloor r \rfloor \leq 1$ for any real r).

So, let t' be the sampling instant:

$$t' = \begin{cases} \Omega(t-d) & \text{if } p \geq \omega(t-d) \\ O(t-d) & \text{otherwise} \end{cases}$$

It is not difficult to check that $(t - t')/\delta \in [l', u']$. In fact:

- if $p \geq \omega(t - d)$, then $t - t' = t - \Omega(t - d) = \delta(k - \lfloor (k\delta - d)/\delta \rfloor) = \delta(-\lfloor -d/\delta \rfloor) = \delta \lceil d/\delta \rceil$, since $\lfloor -r \rfloor = -\lceil r \rceil$ for any real r. Recall that $d \in \langle l, u \rangle$, and then *a fortiori* $d \in [l, u] \supseteq \langle l, u \rangle$. So $d/\delta \in [l/\delta, u/\delta]$, and $(t - t')/\delta = \lceil d/\delta \rceil \in [\lceil l/\delta \rceil, \lceil u/\delta \rceil] \subseteq [l', u']$.
- if $p < \omega(t + s)$, then $t - t' = t - O(t - d) = \delta(k - \lceil (k\delta - d)/\delta \rceil) = \delta(-\lceil -d/\delta \rceil) = \delta \lfloor d/\delta \rfloor$, since $\lceil -r \rceil = -\lfloor r \rfloor$ for any real r. Recall that $d \in \langle l, u \rangle$, and then *a fortiori* $d \in [l, u] \supseteq \langle l, u \rangle$. So $d/\delta \in [l/\delta, u/\delta]$, and $(t - t')/\delta = \lfloor d/\delta \rfloor \in [\lfloor l/\delta \rfloor, \lfloor u/\delta \rfloor] \subseteq [l', u']$.

So far, we have shown that $b(t') \models_{\mathbb{R}} \xi_2$. By inductive hypothesis, it follows that for $d' = (t - t')/\delta$ it is $b'(k - d') \models_{\mathbb{Z}} \xi_2$. Since $d' \in [l', u']$, to conclude this branch of the proof we have to show that for all $e' \in [-(d' - 1), 0]$ it is $b'(k + e') \models_{\mathbb{Z}} \xi_1$. To this end, we just have to realize that $\delta(d' - 1) < d$. In fact, we have shown above that $d' \leq \lceil d/\delta \rceil < d/\delta + 1$, since $\lceil r \rceil < r + 1$ for any real number r. So obviously $\delta(d' - 1) < \delta(d/\delta) = d$. Since for all $e \in \langle -d, 0]$ we have $b(t+e) \models_{\mathbb{R}} \xi_1$, and since $[-\delta(d' - 1), 0] \subset \langle -d, 0]$, then *a fortiori* for all $e \in [-\delta(d'-1), 0]$ it is $b(t+e) \models_{\mathbb{R}} \xi_1$. Finally, by inductive hypothesis, it follows that for all integers $e' \in [-(d' - 1), 0] = (-d', 0]$ it is $b'(k + e') \models_{\mathbb{Z}} \xi_1$, which lets us conclude that $b'(k) \models_{\mathbb{Z}} \phi'$.

- $\phi = \text{Releases}_{\langle l,u \rangle \rangle}(\xi_1, \xi_2)$.
  Notice that ϕ' is $\text{Releases}_{\langle l',u' \rangle]}(\xi_1, \xi_2)$, where l', u' depend on the kind of interval $I = \langle l, u \rangle$ is.

  Let d' be a generic integer in $\langle l', u' \rangle$. We have to show that either $b'(k + d') \models_{\mathbb{Z}} \xi_2$ or there exists a $e' \in [0, d']$ such that $b'(k + e') \models_{\mathbb{Z}} \xi_1$.

  Now, we claim that $\langle l', u' \rangle \subseteq \langle l/\delta, u/\delta \rangle$. To show this we discuss the four possible cases for the interval $I' = \langle l', u' \rangle$.

  - $I = [l, u]$, so $I' = [l', u']$, where $l' = \lceil l/\delta \rceil$ and $u' = \lfloor u/\delta \rfloor$.
    Thus, $[l', u'] \subseteq [l/\delta, u/\delta]$, since $\lfloor r \rfloor \leq r$ and $\lceil r \rceil \geq r$ for any real number r.
  - $I = [l, u)$, so $I' = [l', u')$, where $l' = \lceil l/\delta \rceil$ and $u' = \lceil u/\delta \rceil$.
    Thus, $[l', u') \subseteq [l/\delta, u/\delta)$, since $[l', u') = [\lceil l/\delta \rceil, \lceil u/\delta \rceil - 1] \subseteq [l/\delta, u/\delta)$, noting that $\lceil r \rceil \geq r$, and that $\lceil r \rceil - 1 < r$, for any real r.

- $I = (l, u]$, so $I' = (l', u']$, where $l' = \lfloor l/\delta \rfloor$ and $u' = \lfloor u/\delta \rfloor$.
  Thus, $(l', u'] \subseteq (l/\delta, u/\delta]$, since $(l', u'] = [\lfloor l/\delta \rfloor + 1, \lfloor u/\delta \rfloor] \subseteq (l/\delta, u/\delta]$, noting that $\lfloor r \rfloor \leq r$, and that $\lfloor r \rfloor + 1 > r$, for any real r.
- $I = (l, u)$, so $I' = (l', u')$, where $l' = \lfloor l/\delta \rfloor$ and $u' = \lceil u/\delta \rceil$.
  Thus, $(l', u') \subseteq (l/\delta, u/\delta)$, since $(l', u') = [\lfloor l/\delta \rfloor + 1, \lceil u/\delta \rceil - 1] \subset (l/\delta, u/\delta)$, noting that $\lfloor r \rfloor + 1 > r$, and that $\lceil r \rceil - 1 < r$, for any real r.

All in all, by hypothesis it is either $b(t + \delta d') \models_{\mathbb{R}} \xi_2$ or there exists a $e \in [0, \delta d'\rangle$ such that $b(t + e) \models_{\mathbb{R}} \xi_1$.

In the former case, by inductive hypothesis we have that $b'(k + d') \models_{\mathbb{Z}} \xi_2$, which fulfills the goal. In the latter case, we can write that $b(t) \models_{\mathbb{R}} \mathrm{Until}_{[0,\delta d'\rangle\rangle}(\mathrm{true}, \xi_1)$. Thus, by inductive hypothesis, we infer that $b'(k) \models_{\mathbb{Z}} \mathrm{Until}_{[0,d'\rangle\rangle}(\mathrm{true}, \xi_1)$. Therefore, from the definition of the Until operator, we have that there exists a $e' \in [0, d'\rangle \subseteq [0, d']$ such that $b'(k + e') \models_{\mathbb{Z}} \xi_1$, as required.

- $\phi = \mathrm{Released}_{\langle l,u \rangle \langle}(\xi_1, \xi_2)$.
  Notice that ϕ' is $\mathrm{Released}_{\langle l',u' \rangle [}(\xi_1, \xi_2)$, where l', u' depend on the kind of interval $I = \langle l, u \rangle$ is.

  Let d' be a generic integer in $\langle l', u' \rangle$. We have to show that either $b'(k - d') \models_{\mathbb{Z}} \xi_2$ or there exists a $e' \in [-d', 0]$ such that $b'(k + e') \models_{\mathbb{Z}} \xi_1$.

  Now, we claim that $\langle l', u' \rangle \subseteq \langle l/\delta, u/\delta \rangle$. In fact, we showed this fact in the above proof for the Releases operator (notice that the bounds l', u' are the same for the Releases and Released operators).

  All in all, by hypothesis it is either $b(t - \delta d') \models_{\mathbb{R}} \xi_2$ or there exists a $e \in \langle -\delta d', 0]$ such that $b(t + e) \models_{\mathbb{R}} \xi_1$.

  In the former case, by inductive hypothesis we have that $b'(k - d') \models_{\mathbb{Z}} \xi_2$, which fulfills the goal. In the latter case, we can write that $b(t) \models_{\mathbb{R}} \mathrm{Since}_{[0,\delta d'\rangle \langle}(\mathrm{true}, \xi_1)$. Thus, by inductive hypothesis, we infer that $b'(k) \models_{\mathbb{Z}} \mathrm{Since}_{[0,d'\rangle (}(\mathrm{true}, \xi_1)$. Therefore, from the definition of the Since operator, we have that there exists a $e'' \in [0, d'\rangle \subseteq [0, d']$ such that $b'(k - e'') \models_{\mathbb{Z}} \xi_1$, or equivalently, by substituting e' for $-e''$, there exists a $e' \in [-d', 0]$ such that $b(k + e') \models_{\mathbb{Z}} \xi_1$, as required.

- $\phi = \phi_1 \wedge \phi_2$.
  Since $b(t) \models_{\mathbb{R}} \phi_i$ implies $b'(k) \models_{\mathbb{Z}} \phi'_i$ for $i = 1, 2$, then also $b(t) \models_{\mathbb{R}} \phi_1 \wedge \phi_2$ implies $b'(k) \models_{\mathbb{Z}} \phi'_1 \wedge \phi'_2$.

- $\phi = \phi_1 \vee \phi_2$.
  Since $b(t) \models_{\mathbb{R}} \phi_i$ implies $b'(k) \models_{\mathbb{Z}} \phi'_i$ for $i = 1$ or $i = 2$, then also $b(t) \models_{\mathbb{R}} \phi_1 \vee \phi_2$ implies $b'(k) \models_{\mathbb{Z}} \phi'_1 \vee \phi'_2$.

Since the above holds for a generic sampling instant t, we have shown that if $b \models_{\mathbb{R}} \phi$ then $\sigma_{\delta,z}[b] \models_{\mathbb{Z}} \eta^{\mathbb{R}}_{\delta}\{\phi\}$ for any behavior $b \in [\![\chi_\circ]\!]_{\mathbb{R}}$. Therefore, we have shown that any $\mathrm{TRIO}_{\mathrm{si}}$ formula ϕ, interpreted in the continuous-time domain, is closed under sampling

(Closure under inverse sampling.) Let b be a discrete-time behavior, and let $\phi' = \eta_\delta^{\mathbb{Z}}\{\phi\}$. Then, let b' be a continuous-time behavior such that $b' \in [\![\chi_\circ]\!]_{\mathbb{R}}$ and $b = \sigma_{\delta,z}[b']$ for the given sampling period δ and origin z. In the remainder, for a generic sampling instant $t = z + k\delta$, we first show that if $b(k) \models_{\mathbb{Z}} \phi$ then $b'(t) \models_{\mathbb{R}} \phi'$.

Let us also introduce the following terminology. For any formula $\phi' = \eta_\delta^{\mathbb{Z}}\{\phi\}$ which holds at a sampling point $t = z + k\delta$:

- if for all $s \in (-\delta, 0)$ it is $b'(t+s) \models_{\mathbb{R}} \phi'$, we say that ϕ' "shifts to the left";
- if for all $s \in (0, \delta)$ it is $b'(t+s) \models_{\mathbb{R}} \phi'$, we say that ϕ' "shifts to the right";
- if there exists a $c \in (0, \delta)$ such that for all $t' \in [t, t+c)$ it is $b'(t') \models_{\mathbb{R}} \phi'$ and for all $t'' \in (t+c, t+c+\epsilon)$ it is $b'(t'') \models_{\mathbb{R}} \neg\phi'$, we say that ϕ' "turns false at c".
- if there exists a $c \in (-\delta, 0)$ such that for all $t' \in (t+c, t]$ it is $b'(t') \models_{\mathbb{R}} \phi'$ and for all $t'' \in (t+c-\epsilon, t+c)$ it is $b'(t'') \models_{\mathbb{R}} \neg\phi'$, we say that ϕ' "turned false at c".

In the following proof, we will also show that, for any $\phi' = \eta_\delta^{\mathbb{Z}}\{\phi\}$:

- either ϕ' shifts to the right, or there exists a $c \in (0, \delta)$ and a condition ξ such that ϕ' and ξ turn false at c;
- either ϕ' shifts to the left, or there exists a $c \in (-\delta, 0)$ such that ϕ' and ξ turned false at c.

We call the point c the "change point" of ϕ'. Notice that, whenever the above properties hold, ϕ' becomes false *together with some condition* ξ becoming false, at any change point. Therefore, in such cases, ϕ' becoming false is ultimately a consequence of some basic item changing its value.

Let us now proceed by induction on the structure of the formula ϕ.

- $\phi = \xi$.
  By definition of sampling of a behavior, the truth value of ξ at k and t is the same, i.e. $\xi|_{b(k)} = \xi|_{b'(z+k\delta)} = \xi|_{b'(t)}$.

  By Lemma 2, there exist c_n, c_p such that the condition ξ changes its value at these points, or ξ is indefinitely true in the past and/or future. If the latter is the case, then ξ obviously shifts to the left/right respectively. Thus, let us assume that the former is the case.

  Let us consider c_p first, and show that either ξ shifts to the left, or there exists a $c \in (-\delta, 0)$ such that ξ turned false at c. Indeed, if $c_p - t \in (-\delta, 0)$, then ξ is true for all $t' \in (c_p, t] = (t+c, t]$, and for $\epsilon = \delta$, we have that for all $t'' \in (c_p - \epsilon, c_p)$ it is $b'(t'') \models_{\mathbb{R}} \neg\xi$, therefore ξ turned false at $t - c_p$. Otherwise, ξ shifts to the left.
  Similarly, by considering c_n, we show that either ξ shifts to the right, or there exists a $c \in (0, \delta)$ such that ξ turns false at c. Indeed, if $c_n - t \in (0, \delta)$,

then ξ is true for all $t' \in [t, c_n) = [t, t+c)$, and for $\epsilon = \delta$, we have that for all $t'' \in (c_n, c_n + \epsilon)$ it is $b'(t'') \models_{\mathbb{R}} \neg\xi$, therefore ξ turns false at $c_n - t$. Otherwise, ξ shifts to the right.

- $\phi = \text{Until}_{[l,u]]}(\xi_1, \xi_2)$.
  Notice that $\phi' = \text{Until}_{[l',u']]}(\xi_1, \xi_2)$, with $l' = (l-1)\delta$ and $u' = (u+1)\delta$.

  First, let us show that $b'(t) \models_{\mathbb{R}} \text{Until}_{[l\delta,u\delta]}(\xi_1, \xi_2)$. Notice that this implies $b'(t) \models_{\mathbb{R}} \phi'$, as $[l\delta, u\delta] \subset [(l-1)\delta, (u+1)\delta]$.
  Let $d \in [l, u]$ be the integer time instant such that $b(k+d) \models_{\mathbb{Z}} \xi_2$, which exists by hypothesis. Let $d' = d\delta$, and notice that $d' \in [l\delta, u\delta]$. Then, by inductive hypothesis, it is $b'(t+d') \models_{\mathbb{R}} \xi_2$.
  By hypothesis we also know that for all integers $e \in [0, d]$ it is $b(k+e) \models_{\mathbb{Z}} \xi_1$. Thus, by inductive hypothesis, we have that for all δ-multiples $e' \in [0, d']$ it is $b'(t+e') \models_{\mathbb{R}} \xi_1$. Finally, because of the condition $\chi_\circ$, it must also be that, for all *real* values $e'' \in [0, d']$, it is $b'(t+e'') \models_{\mathbb{R}} \xi_1$. Therefore, we have shown that $b'(t) \models_{\mathbb{R}} \text{Until}_{[l\delta,u\delta]]}(\xi_1, \xi_2)$.

  Now, let us show that Until shifts to the right, that is for all $s \in (0, \delta)$ it is $b'(t+s) \models_{\mathbb{R}} \phi'$.
  Let s be a generic time instant in $(0, \delta)$. Let us consider $c = d' - s$, and notice that $c > d' - \delta \geq l\delta - \delta = l'$, and $c < d' \leq u\delta < u'$, thus $c \in [l', u']$. Since $t + s + c = t + d'$, we have shown above that $b'((t+s)+c) = b'(t+d') \models_{\mathbb{R}} \xi_2$.
  Moreover, $[s, s+c] \subset [0, d']$, thus *a fortiori* for all $f \in [0, c]$ it is $b'((t+s)+f) \models_{\mathbb{R}} \xi_1$, from what we have shown above. This shows that $b'(t+s) \models_{\mathbb{R}} \phi'$ for any $s \in (0, \delta)$.

  Finally, let us show that either ϕ' also shifts to the left, or there exist a $c \in (-\delta, 0)$ and a condition ξ such that ϕ' and ξ turned false at c.
  To this end, take any $f \in (-\delta, 0)$, and let $\xi = \xi_1$. For $d'' = d' - f$ we have that $d'' \in [l\delta, (u+1)\delta] \subset [(l-1)\delta, (u+1)\delta]$ and $b'(t+f+d'') \models_{\mathbb{R}} \xi_2$.
  Now, let us consider ξ_1. By inductive hypothesis, either there exists a $c \in (-\delta, 0)$ such that ξ_1 turned false at c, or ξ_1 shifts to the left. In the latter case, ϕ' shifts to the left as well. In the former case, it is easy to realize that for all $t' \in (t+c, t]$ we have $b'(t') \models_{\mathbb{R}} \phi' \wedge \xi_1$ and ϕ' turned false at c, since "right before" $t+c$ the Until must be false, since ξ_1 is false there.

- $\phi = \text{Since}_{[l,u][}(\xi_1, \xi_2)$.
  Notice that $\phi' = \text{Since}_{[l',u'][}(\xi_1, \xi_2)$, with $l' = (l-1)\delta$ and $u' = (u+1)\delta$.

  First, let us show that $b'(t) \models_{\mathbb{R}} \text{Since}_{[l\delta,u\delta][}(\xi_1, \xi_2)$. Notice that this implies $b'(t) \models_{\mathbb{R}} \phi'$, as $[l\delta, u\delta] \subset [(l-1)\delta, (u+1)\delta]$.
  Let $d \in [l, u]$ be the integer time instant such that $b(k-d) \models_{\mathbb{Z}} \xi_2$, which exists by hypothesis. Let $d' = d\delta$, and notice that $d' \in [l\delta, u\delta]$. Then, by inductive hypothesis, it is $b'(t-d') \models_{\mathbb{R}} \xi_2$.
  By hypothesis we also know that for all integers $e \in [-d, 0]$ it is $b(k+e) \models_{\mathbb{Z}} \xi_1$. Thus, by inductive hypothesis, we have that for all δ-multiples

$e' \in [-d', 0]$ it is $b'(t+e') \models_{\mathbb{R}} \xi_1$. Finally, because of the condition $\chi_\circ$, it must also be that, for all *real* values $e'' \in [-d', 0]$, it is $b'(t+e'') \models_{\mathbb{R}} \xi_1$. Therefore, we have shown that $b'(t) \models_{\mathbb{R}} \mathrm{Since}_{[l\delta, u\delta][}(\xi_1, \xi_2)$.

Now, let us show that Since shifts to the left, that is for all $s \in (-\delta, 0)$ it is $b'(t+s) \models_{\mathbb{R}} \phi'$.
Let s be a generic time instant in $(-\delta, 0)$. Let us consider $c = d' + s$, and notice that $c > d' - \delta \geq l\delta - \delta = l'$, and $c < d' \leq u\delta < u'$, thus $c \in [l', u']$. Since $t - (c - s) = t - d'$, we have shown above that $b'((t+s) - c) = b'(t - d') \models_{\mathbb{R}} \xi_2$.
Moreover, $[s - c, s] \subset [-d', 0]$, thus *a fortiori* for all $f \in [-c, 0]$ it is $b'((t+s)+f) \models_{\mathbb{R}} \xi_1$, from what we have shown above. This shows that $b'(t+s) \models_{\mathbb{R}} \phi'$ for any $s \in (-\delta, 0)$.

Finally, let us show that either ϕ' also shifts to the right, or there exist a $c \in (0, \delta)$ and a condition ξ such that ϕ' and ξ turn false at c.
To this end, take any $f \in (0, \delta)$, and let $\xi = \xi_1$. For $d'' = d' + f$ we have that $d'' \in [l\delta, (u+1)\delta] \subset [(l-1)\delta, (u+1)\delta]$ and $b'((t+f) - d'') = b'(t - d') \models_{\mathbb{R}} \xi_2$.
Now, let us consider ξ_1. By inductive hypothesis, either there exists a $c \in (-\delta, 0)$ such that ξ_1 turns false at c, or ξ_1 shifts to the right. In the latter case, ϕ' shifts to the right as well. In the former case, it is easy to realize that for all $t' \in [t, t+c)$ we have $b'(t') \models_{\mathbb{R}} \phi' \wedge \xi_1$ and ϕ' turns false at c, since "right after" $t + c$ the Since must be false, since ξ_1 is false there.

- $\phi = \mathrm{Releases}_{[l,u])}(\xi_1, \xi_2)$.
  Notice that $\phi' = \mathrm{Releases}_{[l',u'])}(\xi_1, \xi_2)$, with $l' = (l+1)\delta$ and $u' = (u-1)\delta$.

  First, let us show that $b'(t) \models_{\mathbb{R}} \mathrm{Releases}_{[l\delta,u\delta])}(\xi_1, \xi_2)$. Notice that this implies $b'(t) \models_{\mathbb{R}} \phi'$, as $[l\delta, u\delta] \supset [l', u']$.
  Let d' be a generic instant in $[l\delta, u\delta]$: we have to show that either $b'(t + d') \models_{\mathbb{R}} \xi_2$ or there exists a $e' \in [0, d')$ such that $b'(t+e') \models_{\mathbb{R}} \xi_1$. We distinguish two cases, whether $t + d'$ is a sampling instant or not.

  - If $t+d'$ is a sampling instant, then $d = d'/\delta$ is an integer, and $d \in [l, u]$ by hypothesis.
    Therefore, by hypothesis it is either $b(k+d) \models_{\mathbb{Z}} \xi_2$ or there exists an integer $e \in [0, d-1]$ such that $b(k+e) \models_{\mathbb{Z}} \xi_1$.
    In the former case, by inductive hypothesis we have that $b'(t+d') \models_{\mathbb{R}} \xi_2$, which satisfies the definition of the Releases operator for d'.
    Otherwise, by inductive hypothesis we have that $b'(t+e') \models_{\mathbb{R}} \xi_1$, for $e' = e\delta$. Since $e' \in [0, d' - \delta] \subset [0, d')$, we have that the definition of the Releases operator is satisfied for d' in this case as well.

  - If $t + d'$ is not a sampling instant, let us consider the distances $p' = d' - \omega(t+d')$ and $n' = d' + o(t+d')$; these are δ-multiples by definition of $\omega(\cdot)$ and $o(\cdot)$. Notice that $p' > d' - \delta \geq l\delta - \delta = (l-1)\delta$, and

$n' < d' + \delta \leq u\delta + \delta = (u+1)\delta$. Therefore, the two integers $p = p'/\delta$ and $n = n'/\delta$ are $\geq l$ and $\leq u$ respectively, that is $p, n \in [l, u]$.
Thus, by hypothesis we have that:

* $b(k+p) \models_{\mathbb{Z}} \xi_2$ or there exists a $e_p \in [0, p-1]$ such that $b(k + e_p) \models_{\mathbb{Z}} \xi_1$; and that:
* $b(k+n) \models_{\mathbb{Z}} \xi_2$ or there exists a $e_n \in [0, n-1]$ such that $b(k + e_n) \models_{\mathbb{Z}} \xi_1$.

Therefore, we distinguish the following three cases (covering all the possibilities):

* $b(k+p) \models_{\mathbb{Z}} \xi_2$ and $b(k+n) \models_{\mathbb{Z}} \xi_2$;
* there exists a $e_p \in [0, p-1]$ such that $b(k+e_p) \models_{\mathbb{Z}} \xi_1$;
* there exists a $e_n \in [0, n-1]$ such that $b(k+e_n) \models_{\mathbb{Z}} \xi_1$.

Let us first consider the case: $b(k+p) \models_{\mathbb{Z}} \xi_2$ and $b(k+n) \models_{\mathbb{Z}} \xi_2$. Notice that $n = p+1$, so ξ_2 is true on two adjacent time instants. By inductive hypothesis, we have that $b'(t+p') \models_{\mathbb{R}} \xi_2$ and $b'(t+n') \models_{\mathbb{R}} \xi_2$. Moreover, thanks to the constraint $\chi_\circ$ which holds for b by hypothesis, ξ_2 must also be true for all time instants in the interval $[t+p', t+n'] = [t+p', t+p'+\delta]$. In particular, since $d' \in [p', p'+\delta]$, then $b'(t+d') \models_{\mathbb{R}} \xi_2$. That is, the definition of the Releases is satisfied for d'.
Now, for the other two cases, notice that $(p-1)\delta \leq (n-1)\delta < (d'+\delta) - \delta = d'$; hence $e_p, e_n \in [0, n-1]$ and $\delta e_p, \delta e_n \in [0, (n-1)\delta] \subset [0, d')$. Therefore, if there exists a $e_p \in [0, p-1]$ such that $b(k+e_p) \models_{\mathbb{Z}} \xi_1$, or if there exists a $e_n \in [0, n-1]$ such that $b(k+e_n) \models_{\mathbb{Z}} \xi_1$, then there exists an $e' = \delta e_p$ or $e' = \delta e_n$, respectively, such that $e' \in [0, d')$ and $b'(t+e') \models_{\mathbb{R}} \xi_1$. That is, the definition of the Releases is satisfied for d' in both these cases.

All in all, since d' is generic, we have shown that $b'(t) \models_{\mathbb{R}} \text{Releases}_{[l\delta, u\delta])}(\xi_1, \xi_2)$, and thus *a fortiori* $b'(t) \models_{\mathbb{R}} \phi'$.

Now, let us show that Releases shifts to the left, that is for all $s \in (-\delta, 0)$ it is $b'(t+s) \models_{\mathbb{R}} \phi'$.
Let s be any time instant in $(-\delta, 0)$, and consider a generic $d \in [l', u']$. Since $s+d \in [l\delta, u\delta]$, we have shown above that it is either $b'(t+(s+d)) \models_{\mathbb{R}} \xi_2$ or there exists a $e' \in [0, s+d)$ such that $b'(t+e') \models_{\mathbb{R}} \xi_1$.
In the former case, we have shown that $b'((t+s)+d) \models_{\mathbb{R}} \xi_2$. Otherwise, let $e'' = e' - s$; then $e'' \in [-s, d) \subset [0, d)$, as $s < 0$. Therefore, we have that there exists a $e'' \in [0, d)$ such that $b'(t+e') = b'((t+s)+e'') \models_{\mathbb{R}} \xi_1$.
All in all, we have shown that $b'(t+s) \models_{\mathbb{R}} \phi'$ for a generic $s \in (-\delta, 0)$.

Finally, let us show that either ϕ' also shifts to the right, or there exist a $c \in (0, \delta)$ and a condition ξ such that ϕ' and ξ turn false at c.
To this end, take generic $f \in (0, \delta)$ and $d'' \in [(l+1)\delta, (u-1)\delta]$, and let $\xi = \xi_1$. Notice that $d'' + f \in [l\delta, u\delta]$, therefore by hypothesis it is either

$b'(t{+}d''{+}f) \models_{\mathbb{R}} \xi_2$ or there exists a $c \in [0, d''{+}f)$ such that $b'(t{+}c) \models_{\mathbb{R}} \xi_1$. Now let us consider two cases:

- For all f and d'' as above, it is either $b'((t+f)+d'') \models_{\mathbb{R}} \xi_2$ or there exists a $c \in [f, d''{+}f)$ such that $b'(t{+}c) = b'((t{+}f){+}(c{-}f)) \models_{\mathbb{R}} \xi_1$. Since $c - f \in [0, d'')$, this shows that $b'(t + f) \models_{\mathbb{R}} \phi'$, and thus ϕ' shifts to the right since f is generic.
- Otherwise, assume that the previous case is false, that is there is some f and d'' as above such that $b'((t+f)+d'') \models_{\mathbb{R}} \neg\xi_2$ and for all $c' \in [f, d''{+}f)$ it is $b'(t{+}c') = b'((t{+}f){+}(c'{-}f)) \models_{\mathbb{R}} \neg\xi_1$. Therefore, it must be that there exists a $c \in [0, f)$ such that $b'(t + c) \models_{\mathbb{R}} \xi_1$.
  Now, if we consider Lemma 2, and because of the non-Zeno behavior of the basic items values, a point $t + v$ at which ξ_1 becomes false must exist for some v, that is for some v it must be $b'(t + v) \models_{\mathbb{R}}$ Becomes$(\neg\xi_1)$. Then, ξ_1 is true in the whole interval $[t, t + v)$, since $v = c < f < \delta$ in this branch of the proof, and false afterwards, and in particular in the interval $(t + v, t + v + \delta)$. Consequently, ϕ' is also true in the whole interval $[t, t + v)$, and it becomes false "right after" $t + v$, as a consequence of ξ_1 turning false.

A little reasoning should convince us that the two cases do indeed cover all the possibilities. Then, since f is generic, we have shown that either ϕ' shifts to the right, or there exists a $c \in (0, \delta)$ such that ϕ' and ξ_1 turn false at c.

- $\phi = \text{Released}_{[l,u](}(\xi_1, \xi_2)$.
  Notice that $\phi' = \text{Released}_{[l',u'](}(\xi_1, \xi_2)$, with $l' = (l{+}1)\delta$ and $u' = (u{-}1)\delta$.

  First, let us show that $b'(t) \models_{\mathbb{R}} \text{Released}_{[l\delta,u\delta](}(\xi_1, \xi_2)$. Notice that this implies $b'(t) \models_{\mathbb{R}} \phi'$, as $[l\delta, u\delta] \supset [l', u']$.
  Let d' be a generic instant in $[l\delta, u\delta]$: we have to show that either $b'(t - d') \models_{\mathbb{R}} \xi_2$ or there exists a $e' \in (-d', 0]$ such that $b'(t + e') \models_{\mathbb{R}} \xi_1$. We distinguish two cases, whether $t - d'$ is a sampling instant or not.

  - If $t{-}d'$ is a sampling instant, then $d = d'/\delta$ is an integer, and $d \in [l, u]$ by hypothesis.
    Therefore, by hypothesis it is either $b(k - d) \models_{\mathbb{Z}} \xi_2$ or there exists an integer $e \in [-(d - 1), 0]$ such that $b(k + e) \models_{\mathbb{Z}} \xi_1$.
    In the former case, by inductive hypothesis we have that $b'(t{-}d') \models_{\mathbb{R}} \xi_2$, which satisfies the definition of the Released operator for d'.
    Otherwise, by inductive hypothesis we have that $b'(t + e') \models_{\mathbb{R}} \xi_1$, for $e' = e\delta$. Since $e' \in [-(d' - \delta), 0] \subset (-d', 0]$, we have that the definition of the Released operator is satisfied for d' in this case as well.

  - If $t - d'$ is not a sampling instant, let us consider the distances $p' = d' - o(t{-}d')$ and $n' = d' + \omega(t{-}d')$; these are δ-multiples by definition

of $\omega()$ and $o()$. Notice that $p' > d' - \delta, \geq l\delta - \delta = (l-1)\delta$, and $n' < d' + \delta \leq u\delta + \delta = (u+1)\delta$. Therefore, the two integers $p = p'/\delta$ and $n = n'/\delta$ are $\geq l$ and $\leq u$ respectively, that is $p, n \in [l, u]$.
Thus, by hypothesis we have that:

* $b(k-p) \models_{\mathbb{Z}} \xi_2$ or there exists a $e_p \in [-(p-1), 0]$ such that $b(k + e_p) \models_{\mathbb{Z}} \xi_1$; and that:
* $b(k-n) \models_{\mathbb{Z}} \xi_2$ or there exists a $e_n \in [-(n-1), 0]$ such that $b(k + e_n) \models_{\mathbb{Z}} \xi_1$.

Therefore, we distinguish the following three cases (covering all the possibilities):

* $b(k-p) \models_{\mathbb{Z}} \xi_2$ and $b(k-n) \models_{\mathbb{Z}} \xi_2$;
* there exists a $e_p \in [-(p-1), 0]$ such that $b(k+e_p) \models_{\mathbb{Z}} \xi_1$;
* there exists a $e_n \in [-(n-1), 0]$ such that $b(k+e_n) \models_{\mathbb{Z}} \xi_1$.

Let us first consider the case: $b(k-p) \models_{\mathbb{Z}} \xi_2$ and $b(k-n) \models_{\mathbb{Z}} \xi_2$. Notice that $-n = -p-1$, so ξ_2 is true on two adjacent time instants. By inductive hypothesis, we have that $b'(t-p') \models_{\mathbb{R}} \xi_2$ and $b'(t-n') \models_{\mathbb{R}} \xi_2$. Moreover, thanks to the constraint $\chi_\circ$ which holds for b by hypothesis, ξ_2 must also be true for all time instants in the interval $[t-n', t-p'] = [t-n', t-(n'-\delta)]$. In particular, since $d' \in [n'-\delta, n']$, then $b'(t-d') \models_{\mathbb{R}} \xi_2$. That is, the definition of the Released is satisfied for d'.
Now, for the other two cases, notice that $-(p-1)\delta \geq -(n-1)\delta > -d' + \delta - \delta = -d'$; hence $e_p, e_n \in [-(n-1), 0]$ and $\delta e_p, \delta e_n \in [-(n-1)\delta, 0] \subset (-d', 0]$. Therefore, if there exists a $e_p \in [-(p-1), 0]$ such that $b(k+e_p) \models_{\mathbb{Z}} \xi_1$, or if there exists a $e_n \in [-(n-1), 0]$ such that $b(k+e_n) \models_{\mathbb{Z}} \xi_1$, then there exists an $e' = \delta e_p$ or $e' = \delta e_n$, respectively, such that $e' \in (-d', 0]$ and $b'(t+e') \models_{\mathbb{R}} \xi_1$. That is, the definition of the Released is satisfied for d' in both these cases.

All in all, since d' is generic, we have shown that $b'(t) \models_{\mathbb{R}} \text{Released}_{[l\delta, u\delta](}(\xi_1, \xi_2)$, and thus *a fortiori* $b'(t) \models_{\mathbb{R}} \phi'$.

Now, let us show that Released shifts to the right, that is for all $s \in (0, \delta)$ it is $b'(t+s) \models_{\mathbb{R}} \phi'$.
Let s be any time instant in $(0, \delta)$, and consider a generic $d \in [l', u']$. Since $d - s \in [l\delta, u\delta]$, we have shown above that it is either $b'(t-(d-s)) \models_{\mathbb{R}} \xi_2$ or there exists a $e' \in (-(d-s), 0]$ such that $b'(t+e') \models_{\mathbb{R}} \xi_1$.
In the former case, we have shown that $b'((t+s)-d) \models_{\mathbb{R}} \xi_2$. Otherwise, let $e'' = e' - s$; then $e'' \in (-d, -s] \subset (-d, 0]$, as $s > 0$. Therefore, we have that there exists a $e'' \in (-d, 0]$ such that $b'(t+e') = b'((t+s)+e'') \models_{\mathbb{R}} \xi_1$.
All in all, we have shown that $b'(t+s) \models_{\mathbb{R}} \phi'$ for a generic $s \in (0, \delta)$.

Finally, let us show that either ϕ' also shifts to the left, or there exist a $c \in (-\delta, 0)$ and a condition ξ such that ϕ' and ξ turned false at c.
To this end, take generic $f \in (-\delta, 0)$ and $d'' \in [(l+1)\delta, (u-1)\delta]$, and let $\xi = \xi_1$. Notice that $d'' - f \in [l\delta, u\delta]$, therefore by hypothesis it is

either $b'(t-(d''-f)) \models_{\mathbb{R}} \xi_2$ or there exists a $c \in (d''+f, 0]$ such that $b'(t+c) \models_{\mathbb{R}} \xi_1$.
Now let us consider two cases:

- For all f and d'' as above, it is either $b'((t+f)-d'') \models_{\mathbb{R}} \xi_2$ or there exists a $c \in (-(d''-f), f]$ such that $b'(t+c) = b'((t+f)+(c-f)) \models_{\mathbb{R}} \xi_1$.
  Since $c-f \in (-d'', 0]$, this shows that $b'(t+f) \models_{\mathbb{R}} \phi'$, and thus ϕ' shifts to the left since f is generic.
- Otherwise, assume that the previous case is false, that is there is some f and d'' as above such that $b'((t+f)-d'') \models_{\mathbb{R}} \neg\xi_2$ and for all $c' \in (-(d''-f), f]$ it is $b'(t+c') \models_{\mathbb{R}} \neg\xi_1$. Therefore, it must be that there exists a $c \in (f, 0]$ such that $b'(t+c) \models_{\mathbb{R}} \xi_1$.
  Now, if we consider Lemma 2, and because of the non-Zeno behavior of the basic items values, a point $t+v$ at which ξ_1 becomes false must exist for some $v < 0$, that is for some $v < 0$ it must be $b'(t+v) \models_{\mathbb{R}}$ Becomes(ξ_1). Then, ξ_1 is true in the whole interval $(t+v, t]$, since $-v = -c < -f < \delta$ in this branch of the proof, and false beforehand, and in particular in the interval $(t+v, t+v-\delta)$. Consequently, ϕ' is also true in the whole interval $(t+v, t]$, and it became false "right before" $t+v$, as a consequence of ξ_1 turned false.

A little reasoning should convince us that the two cases do indeed cover all the possibilities. Then, since f is generic, we have shown that either ϕ' shifts to the left, or there exists a $c \in (-\delta, 0)$ such that ϕ' and ξ_1 turned false at c.

- $\phi = \phi_1 \wedge \phi_2$.
  Since $b(k) \models_{\mathbb{Z}} \phi_i$ implies $b'(t) \models_{\mathbb{R}} \phi'_i$ for $i = 1, 2$ by inductive hypothesis, then also $b(k) \models_{\mathbb{Z}} \phi_1 \wedge \phi_2$ implies $b'(t) \models_{\mathbb{R}} \phi'_1 \wedge \phi'_2$.

  Now, by inductive hypothesis, for both $i = 1, 2$: either ϕ'_i shifts to the right, or there exists a $c_i \in (0, \delta)$ and some condition ξ_i such that ϕ'_i and ξ_i turn to false at c_i. Thus, if both ϕ'_1 and ϕ'_2 shift to the right, then $\phi' = \phi'_1 \wedge \phi'_2$ also shifts to the right; otherwise there exists a $c \in (0, \delta) = \min_{i=1,2} c_i$ such that $\phi'_1 \wedge \phi'_2$ turns to false at c.
  Similarly, by inductive hypothesis for both $i = 1, 2$: either ϕ'_i shifts to the left, or there exists a $c_i \in (-\delta, 0)$ and some condition ξ_i such that ϕ'_i and ξ_i turned to false at c_i. Thus, if both ϕ'_1 and ϕ'_2 shift to the left, then $\phi' = \phi'_1 \wedge \phi'_2$ also shifts to the left; otherwise there exists a $c \in (-\delta, 0) = \max_{i=1,2} c_i$ such that $\phi'_1 \wedge \phi'_2$ turned false at c.

- $\phi = \phi_1 \vee \phi_2$.
  Since $b(k) \models_{\mathbb{Z}} \phi_i$ implies $b'(t) \models_{\mathbb{R}} \phi'_i$ for $i = 1$ or $i = 2$ by inductive hypothesis, then also $b(k) \models_{\mathbb{Z}} \phi_1 \vee \phi_2$ implies $b'(t) \models_{\mathbb{R}} \phi'_1 \vee \phi'_2$.

  Now, we realize that if $b'(t) \models_{\mathbb{R}} \phi'_i$ for some $i = 1$ or $i = 2$, then by inductive hypothesis either ϕ'_i shifts to the right or there exists a $c \in (0, \delta)$

and a condition ξ_i such that ϕ'_i and ξ_i turn false at c. Thus, *a fortiori* either ϕ' shifts to the right, or there exist a $c \in (0, \delta)$ and a condition $\xi = \xi_i$ such that ϕ' and ξ turn false at c.
Similarly, by inductive hypothesis either ϕ'_i shifts to the left or there exists a $c \in (-\delta, 0)$ and a condition ξ_i such that ϕ'_i and ξ_i turned false at c. Thus, *a fortiori* either ϕ' shifts to the left, or there exist a $c \in (-\delta, 0)$ and a condition ξ such that ϕ' and $\xi = \xi_i$ turned false at c.

(Filling in the gaps between sampling points.) So far, we have proved that, for any formula ϕ, if $b \models_{\mathbb{Z}} \phi$ then for all $t = z + k\delta$ for *integer* k's, it is $b'(t) \models_{\mathbb{R}} \phi'$. Moreover, we have proved some results about shifting of temporal operators that we are going to use in the remainder.

Now, in order to finish the proof by showing that $b'(t) \models_{\mathbb{R}} \phi'$ for all $t \in \mathbb{R}$, that is $b' \models_{\mathbb{R}} \phi'$, we have to demonstrate that any formula ϕ' which is true at all sampling points cannot become false between any two of them. More precisely, we prove that whenever $\phi' = \eta^{\mathbb{Z}}_{\delta}\{\phi\}$ is true at two adjacent sampling points $t_p = z + k\delta$ and $t_n = t_p + \delta = z + (k+1)\delta$, then for all $t' \in [t_p, t_n]$ it is $b'(t') \models_{\mathbb{R}} \phi'$.

If ϕ' at t_p shifts to the right, or ϕ' at t_n shifts to the left, then the conclusion follows trivially.

Thus, let us assume that ϕ' does not shift to the right at t_p and does not shift to the left at t_n. Therefore, we proved above that:

- there exist a $c_p \in (0, \delta)$ and a condition ξ_p such that ϕ' and ξ_p turn false at c_p; and
- there exist a $c_n \in (-\delta, 0)$ and a condition ξ_n such that ϕ' and ξ_n turned false at c_n.

Now notice that $|(t_n + c_n) - (t_p + c_p)| = |\delta + c_n - c_p| < \delta$. Therefore, because of condition $\chi_\circ$, the two changing points must actually coincide, otherwise there would be two changing points (corresponding to two changing points for the value of the conditions ξ_p and ξ_n, and thus to the changing points of some basic items) within δ time units. But the two formulas above contradict each other if the two changing points coincide. Thus, this means that ϕ' actually *never* becomes false in the interval $[t_p, t_n]$, which is what we had to prove.

Since t_p, t_n are actually generic, we have proved that $b \models_{\mathbb{R}} \phi'$, that is any TRIO_{si} formula ϕ is closed under inverse sampling.

Since any formula of the TRIO_{si} language is both closed under sampling and closed under inverse sampling, then TRIO_{si} is sampling invariant. □

3.2 Dense-valued Items

Let us now consider the case in which some time-dependent items in Ψ have dense codomains, i.e. ψ_i has a dense codomain D_i, for some $i = 1, \ldots, n$. Without loss of generality, we can even assume that *all* D_i's are dense sets, since we can handle separately the discrete sets as seen in the previous subsection (we do not discuss the details of that as it is straightforward). Similarly as for discrete-valued items, we assume that each D_i is a totally ordered set, with no practical loss of generality.

Conditions. Let Λ be a set of n-tuples of the form $\langle \langle^1 l_1, u_1\rangle^1, \ldots, \langle^n l_n, u_n\rangle^n \rangle$, with $l_i, u_i \in D_i$, $l_i \leq u_i$, $\langle^i \in \{(, [\}$, and $\rangle^i \in \{),]\}$, for all $i = 1, \ldots, n$. Moreover, for any $\lambda \in \Lambda$, we denote by $\lambda|_i$ its projection to the ith component, i.e. $\lambda|_i \equiv \langle^i l_i, u_i\rangle^i$. Then, if $\lambda \in \Lambda$, conditions ξ are defined simply as follows:

$$\xi \ ::= \ \langle \psi_1, \ldots, \psi_n \rangle \in \lambda \mid \neg \xi \mid \xi_1 \wedge \xi_2$$

For a behavior $b : \mathbb{T} \to D$, we define $\langle \psi_1, \ldots, \psi_n \rangle|_{b(t)} \equiv \forall i = 1, \ldots, n : \psi_i(t) \in \langle^i l_i, u_i\rangle^i$, i.e. all items are in their respective ranges at time t. Also in this case, it is simple to introduce abbreviations and notational conventions.

Constraint on Behaviors. The constraint on behaviors is now dependent on the actual conditions ξ that we have introduced in our specification formula ϕ. This is different than the constraint $\chi_\circ$ for discrete-valued items, which was independent of ϕ. Now, if we let $\widetilde{\Xi}$ be the set of conditions that appear in ϕ, we have to require that the value of every item in Ψ is such that changes with respect to the conditions in $\widetilde{\Xi}$ happen at most every δ time units. In other words, if any condition $\widetilde{\xi} \in \widetilde{\Xi}$ is true (resp. false) at some time, then it stays true (resp. false) in an interval of length δ (at least). This is expressed by the following "pseudo"-TRIO$_{\mathrm{si}}$ formula, where the higher-order quantification $\forall \widetilde{\xi} \in \widetilde{\Xi}$ is to be meant as a shorthand for the explicit enumeration of all the conditions in (the finite set) $\widetilde{\Xi}$.

$$\chi_\bullet^\phi \triangleq \forall \widetilde{\xi} \in \widetilde{\Xi} : \ \mathrm{WithinP_{ii}}\Big(\mathrm{Lasts_{ii}}\Big(\widetilde{\xi}, \delta\Big), \delta\Big) \vee \mathrm{WithinP_{ii}}\Big(\mathrm{Lasts_{ii}}\Big(\neg\widetilde{\xi}, \delta\Big), \delta\Big)$$

Theorem 4 (Sampling Invariance for Dense-valued Items). *Normal-form TRIO$_{\mathrm{si}}$ is sampling invariant, for items mapping to a dense set D, with respect to the behavior constraint $\chi_\bullet^\phi$, the adaptation functions $\eta_\delta^{\mathbb{R}}\{\cdot\}$ and $\eta_\delta^{\mathbb{Z}}\{\cdot\}$, for any sampling period δ and origin z.*

Proof. Similarly as in Theorem 3, for dense-valued items the truth value of any condition ξ cannot change between any two consecutive sampling instants, exactly because of the behavior constraint $\chi_\bullet^\phi$. In fact, $\chi_\bullet^\phi$ requires that if the value of any item ψ_i is in (resp. out of) an interval $\lambda|_i$, it stays in (resp. out of) $\lambda|_i$ until the next sampling instant. Therefore, thanks to this observation, the proof is exactly as in Theorem 3. □

3.3 How to Avoid Degenerate Intervals

So far, we made no assumptions on the size of the intervals involved in the formulas. In particular, it may happen that the adaptation function changes an interval so that it becomes degenerate, that is empty. Although the above proofs still hold, it may be objected that a degenerate formula makes little sense, since it is either trivially true or trivially false. While leaving a full discussion of the impact of this issue to future work, in this section we present sufficient conditions on the size of the un-adapted intervals that guarantee that adaptation gives non-degenerate adapted intervals.

Table 2 lists the requirements on unadapted intervals that guarantee that the adapted intervals are non-empty (i.e., of size greater than 0, but allowing the endpoints to coincide for intervals of $\mathbb{R}$).

Operator (in $\mathbb{R}$)	Interval	Requirement
$\text{Until}_{I\rangle}(\cdot,\cdot)$	$I = \langle l, u\rangle$	$\lvert I\rvert_{\mathbb{R}} > 0$
$\text{Since}_{I\langle}(\cdot,\cdot)$	$I = \langle l, u\rangle$	$\lvert I\rvert_{\mathbb{R}} > 0$
$\text{Releases}_{I\rangle}(\cdot,\cdot)$	$I = \langle l, u\rangle \neq (l, u)$	$\lvert I\rvert_{\mathbb{R}} \geq \delta$
$\text{Released}_{I\langle}(\cdot,\cdot)$	$I = \langle l, u\rangle \neq (l, u)$	$\lvert I\rvert_{\mathbb{R}} \geq \delta$
$\text{Releases}_{I\rangle}(\cdot,\cdot)$	$I = (l, u)$	$\lvert I\rvert_{\mathbb{R}} \geq 2\delta$
$\text{Released}_{I\langle}(\cdot,\cdot)$	$I = (l, u)$	$\lvert I\rvert_{\mathbb{R}} \geq 2\delta$
Operator (in $\mathbb{Z}$)	Interval	Requirement
$\text{Until}_{I\rangle}(\cdot,\cdot)$	$I = [l, u]$	$\lvert I\rvert_{\mathbb{Z}} \geq 1$
$\text{Since}_{I\langle}(\cdot,\cdot)$	$I = [l, u]$	$\lvert I\rvert_{\mathbb{Z}} \geq 1$
$\text{Releases}_{I\rangle}(\cdot,\cdot)$	$I = [l, u]$	$\lvert I\rvert_{\mathbb{Z}} \geq 3$
$\text{Released}_{I\langle}(\cdot,\cdot)$	$I = [l, u]$	$\lvert I\rvert_{\mathbb{Z}} \geq 3$

Table 2: Requirements for non-degenerate adapted intervals

Now, we prove that those requirements do guarantee the non-emptiness of adapted intervals.

Adapting from continuous time to discrete time.

- *Until and Since.* Let $l' = \lfloor l/\delta \rfloor$ and $u' = \lceil u/\delta \rceil$. Thus, $u' - l' + 1 \geq u/\delta - l/\delta + 1 \geq 1$ by definition of floor and ceilings, that is $\lvert[l', u']\rvert_{\mathbb{Z}} > 0$.
- *Releases and Released.* Let $I' = \langle l', u'\rangle$ be the adapted interval. We distinguish the following four cases.
  - $I' = [l', u']$, with $l' = \lceil l/\delta \rceil$ and $u' = \lfloor u/\delta \rfloor$. Thus, $\lfloor u/\delta \rfloor - \lceil l/\delta \rceil + 1 \geq \lfloor u/\delta \rfloor - \lfloor l/\delta \rfloor - 1 + 1 = \lfloor u/\delta \rfloor - \lfloor l/\delta \rfloor \geq \lfloor u/\delta - l/\delta \rfloor \geq 1$, since $\lvert[u, l]\rvert_{\mathbb{R}} \geq \delta$ by hypothesis, and $\lceil r \rceil \leq r+1$ and $\lfloor r_1 \rfloor - \lfloor r_2 \rfloor \geq \lfloor r_1 - r_2 \rfloor$ for any real numbers $r, r_1 \geq r_2$. That is, $\lvert[l', u']\rvert_{\mathbb{Z}} \geq 1$.
  - $I' = [l', u')$, with $l' = \lceil l/\delta \rceil$ and $u' = \lceil u/\delta \rceil$. Thus, $\lvert[l', u')\rvert_{\mathbb{Z}} = \lvert[l', u' - 1]\rvert_{\mathbb{Z}} = u' - l'$, and then $\lceil u/\delta \rceil - \lceil l/\delta \rceil \geq u/\delta - \lceil l/\delta \rceil > u/\delta - (l/\delta + 1) \geq 0$, since $\lvert[l, u)\rvert_{\mathbb{R}} \geq \delta$ by hypothesis, $\lceil r \rceil < r + 1$ for any real r, and we are considering integer numbers (hence > 0 implies ≥ 1). That is, $\lvert[l', u')\rvert_{\mathbb{Z}} \geq 1$.

- $I' = (l', u']$, with $l' = \lfloor l/\delta \rfloor$ and $u' = \lfloor u/\delta \rfloor$. Thus, $|(l', u']|_{\mathbb{Z}} = |[l'+1, u']|_{\mathbb{Z}} = u' - l'$, and then $\lfloor u/\delta \rfloor - \lfloor l/\delta \rfloor \geq \lfloor u/\delta \rfloor - l/\delta > (u/\delta - 1) - l/\delta \geq 0$, since $|(l, u]|_{\mathbb{R}} \geq \delta$ by hypothesis, $\lfloor r \rfloor > r - 1$ for any real r, and we are considering integer numbers (hence > 0 implies ≥ 1). That is, $|(l', u']|_{\mathbb{Z}} \geq 1$.
- $I' = (l', u')$, with $l' = \lfloor l/\delta \rfloor$ and $u' = \lceil u/\delta \rceil$. Thus, $|(l', u')|_{\mathbb{Z}} = |[l'+1, u'-1]|_{\mathbb{Z}} = u' - l' - 1$, and then $\lceil u/\delta \rceil - \lfloor l/\delta \rfloor - 1 \geq u/\delta - l/\delta - 1 \geq 1$, since $|(l, u)|_{\mathbb{R}} \geq 2\delta$ by hypothesis, and $\lfloor r \rfloor \leq r$ and $\lceil r \rceil \geq r$ for any real number r. That is, $|(l', u')|_{\mathbb{Z}} \geq 1$.

Adapting from discrete time to continuous time.

- *Until and Since.* Let $I' = [l', u']$, with $l' = (l-1)\delta$ and $u' = (u+1)\delta$. Thus, $u' - l' = \delta(u - l + 2) = \delta((u - l + 1) + 1) \geq 2\delta > 0$, since $|[l, u]|_{\mathbb{Z}} = u - l + 1 \geq 1$ by hypothesis.
- *Releases and Released.* Let $I' = [l', u']$, with $l' = (l+1)\delta$ and $u' = (u-1)\delta$. Thus, $u' - l' = \delta(u - l - 2) = \delta((u - l + 1) - 3) \geq 0$, since $|[l, u]|_{\mathbb{Z}} = u - l + 1 \geq 3$ by hypothesis.

4 Related Works

The problem of formally describing in a uniform manner large systems that encompass both discrete-time and continuous-time modules is not a recent one, but has been prominently brought up by the growing interest in *hybrid systems*. Hybrid systems constitute a special class of systems that combine both digital and analog components, whose mathematical models are especially focused on problems of distributed embedded control [2].

Alur et al. present in [1] a particularly successful model of hybrid systems, based on automata. In hybrid automata, phases of continuous-time evolution, guided by simple differential constraints, alternate with phases of discrete-time evolution, where switching within a discrete set of states occurs. Such models are particularly effective for analysis based on automated techniques derived from model checking [1]. Conversely, the approach presented in this paper is different in several respects to that of hybrid automata. First of all, we do not focus on analysis and synthesis of *control* problems, but on more general analysis problems that are typically encountered in the earlier phases of development of a system (esp. requirements and specification phases). In fact, we consider descriptive models, i.e. entirely based on temporal logic formulas, which are are well suited for very high-level descriptions of a system. Moreover, our idea of integration stresses separation of concerns in the development of a specification, as the modules describing different parts of the systems, that obey very different models, can be developed independently and then joined together to have a global description of the system. Thus, no *ad hoc* formalism has to be introduced, since the analysis can exploit the ideas and formalisms of temporal logics.

A straightforward way to compose continuous-time and discrete-time models is to integrate discrete models into continuous ones, by introducing some suitable conventions. This is the approach followed by Fidge in [5], with reference to timed refinement calculus. The overall simplicity of the approach is probably its main strength; nonetheless we notice that integrating everything into a continuous-time setting has some disadvantages in terms of verification complexity, as continuous time usually introduces some peculiar difficulties that render verification less automatizable. On the contrary, our approach wants to achieve *equivalence* between discrete-time and continuous-time descriptions, so that one can resort to the simpler discrete time when verifying properties, while still being able to describe naturally physical systems using a full continuous-time model.

The problem of integrating different temporal (and modal) logics is studied by Chen and Liu in [3] in a very abstract setting. Their solution is based on the introduction of a very general semantic structure, called *resource cumulator*, that can accommodate several concrete semantics and their respective modalities. The framework is then based on *refinement* rules that permit to translate from one logic language to another one, while mapping appropriately the corresponding resource cumulators. The approach seems indeed general and abstract, even if we believe that one may desire more intuitiveness and simplicity of use from that. Moreover, it focuses on refinement among semantically different temporal logic specifications, while our framework focuses on the different problem of integrating different semantics at the same level of abstraction, thus stressing modularity and reuse of specifications.

A framework which is instead less abstract and more similar, at least in principle, to ours is the one presented by Hung and Giang in [10]. There, the authors define a *sampling semantics* for Duration Calculus (DC), which is a formal way to relate the models of a formula to its discrete observations, approximating DC's integrals of state variables by sums. As expected, for general DC formulas the standard semantics and the sampling semantics differ. Hence, the authors derive some inference rules that outline what is the relation between the two semantic models under certain assumptions. In particular, the main aim is to formulate sufficient conditions that guarantee the soundness of *refinement* rules, that permit to move from a continuous-time description to a sampled one, thus moving from specification to implementation. This is the main overall difference with respect to the contribution of the present paper: while we deal with integration *of* continuous and discrete, which then coexist in the same formal model, [10] stresses refinement *from* continuous *to* discrete towards implementation.

The paper by Henzinger et al. [9] is one of the first works studying explicitly the relations between discrete- and continuous-time models, with particular reference to the verification problem. In particular, [9] introduces the idea of *digitizability* which is a sufficient condition for the equivalence of the verification problem of the same property under two different time models. Digitizability is very close in meaning to sampling invariance when considering a fixed sampling period of unit length: in other words a digitized behavior is one obtained by observing a behavior at integer time instants only. However, notice that [9] only considers *discrete trace* models, that is those where behaviors are modeled by discrete sequences of observations. Then, it distinguishes between analog-clock models, i.e. those where a real number is attached to every observation, representing the time at which the observation is recorded, and digital-clock models, i.e. those where the observation time is represented by an integer value. Analog-clock models do not describe truly continuous processes; as a consequence, equivalence between analog- and digital-clock models requires weaker conditions than those we introduced in the present paper. On the contrary, the present work was motivated by the need to model truly continuous physical processes that require a continuous time sort, and analyzed some consequences of this more demanding assumption.

5 Conclusions

This paper introduced the idea of sampling invariance, which is a natural and physically motivated way to relate the continuous-time and discrete-time behaviors of a modular system. Sampling invariance means that a temporal logic formula can be interpreted with a continuous-time or discrete-time model consistently and in a uniform manner. Therefore, writing sampling invariant formulas permits to achieve integration of discrete-time and continuous-time formalisms in the same specification, thus being able to verify the system in the simpler discrete-time models, while still describing naturally physical processes with the true continuity permitted by continuous-time models.

We demonstrated how to achieve sampling invariance with $\mathrm{TRIO}_{\mathrm{si}}$, a subset of the TRIO language. The approach involved the introduction of some suitable constraints on the possible behaviors of a system, which limit the dynamics in an appropriate way, as well as simple translation rules that adapt the assumed meaning of time units in the two time models. We believe that the approach is general enough to permit its extension to other metric temporal logics of comparable expressive power.

References

[1] Rajeev Alur, Thomas A. Henzinger, Gerardo Lafferriere, and George J. Pappas. Discrete abstractions of hybrid systems. In Antsaklis [2], pages 971–984.

[2] Panos J. Antsaklis, editor. *Special issue on hybrid systems: theory and applications*, volume 88 of *Proceedings of the IEEE*, 2000.

[3] Yifeng Chen and Zhiming Liu. Integrating temporal logics. In Eerke A. Boiten, John Derrick, and Graeme Smith, editors, *Proceedings of the 4th International Conference on Integrated Formal Methods (IFM'04)*, volume 2999 of *Lecture Notes in Computer Science*, pages 402–420, 2004.

[4] Emanuele Ciapessoni, Alberto Coen-Porisini, Ernani Crivelli, Dino Mandrioli, Piergiorgio. Mirandola, and Angelo Morzenti. From formal models to formally-based methods: an industrial experience. *ACM Transactions on Software Engineering and Methodology*, 8(1):79–113, 1999.

[5] Colin J. Fidge. Modelling discrete behaviour in a continuous-time formalism. In Keijiro Araki, Andy Galloway, and Kenji Taguchi, editors, *Proceedings of 1st International Conference on Integrated Formal Methods (IFM'99)*, pages 170–188. Springer-Verlag, 1999.

[6] Angelo Gargantini and Angelo Morzenti. Automated deductive requirement analysis of critical systems. *ACM Transactions on Software Engineering and Methodology*, 10(3):255–307, 2001.

[7] Carlo Ghezzi, Dino Mandrioli, and Angelo Morzenti. TRIO: A logic language for executable specifications of real-time systems. *The Journal of Systems and Software*, 12(2):107–123, 1990.

[8] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A foundation for computer science*. Addison-Wesley, 2nd edition, 1994.

[9] Thomas A. Henzinger, Zohar Manna, and Amir Pnueli. What good are digital clocks? In Werner Kuich, editor, *Proceedings of the 19th International Colloquium on Automata, Languages and Programming (ICALP'92)*, volume 623 of *Lecture Notes in Computer Science*, pages 545–558. Springer-Verlag, 1992.

[10] Dang Van Hung and Phan Hong Giang. Sampling semantics of duration calculus. In Bengt Jonsson and Joachim Parrow, editors, *Proceedings of the 4th International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems (FTRTFT'96)*, volume 1135 of *Lecture Notes in Computer Science*, pages 188–207. Springer-Verlag, 1996.

[11] Ron Koymans. Specifying real-time properties with metric temporal logic. *Real-Time Systems*, 2(4):255–299, 1990.

[12] Angelo Morzenti, Dino Mandrioli, and Carlo Ghezzi. A model parametric real-time logic. *ACM Transactions on Programming Languages and Systems*, 14(4):521–573, 1992.

[13] Angelo Morzenti and Pierluigi San Pietro. Object-oriented logical specification of time-critical systems. *ACM Transactions on Software Engineering and Methodology*, 3(1):56–98, 1994.