# Coinductive unwinding of security-relevant hyperproperties

Dimiter Milushev and Dave Clarke

IBBT-DistriNet, KU Leuven, Heverlee, Belgium

**Abstract.** Unwinding relations have been widely used to prove that finite systems are secure with respect to a variety of noninterference policies. The latter are prominent instances of security-relevant hyperproperties. As hyperproperties are defined on potentially infinite systems, a new mathematical development is needed in order to (re)use unwinding relations for generic verification of security-relevant hyperproperties. In this paper we propose a framework for coinductive unwinding of security relevant hyperproperties. To illustrate the usefulness of the framework, we show that Mantel's Basic Security Predicates (BSPs), the noninterference policies they compose, as well as their respective unwinding relations, have a meaningful coinductive reinterpretation. We prove that in a number of cases the coinductive variants of the unwinding relations imply the respective coinductive variants of the BSPs. Moreover, the latter can be used to compose high-level security-relevant hyperproperties for both finite and infinite systems. A number of the unwinding theorems also hold as expected. In conclusion, the proposed framework and results are useful both theoretically in the study of hyperproperties and in practice for verification of hyperproperties on potentially infinite systems.

## 1 Introduction

*Unwinding* is a well-known technique used to prove that systems are secure with respect to a variety of noninterference policies, which essentially regulate the flow of information within a system. The original term and idea of unwinding date back to the work of Goguen and Meseguer [3]. As they describe it, unwinding is the process of translating a security policy, first, into local constraints on the transition system, inductively guaranteeing that the policy is satisfied, and second, into a finite set of lemmas such that any system that satisfies the lemmas is guaranteed to satisfy the policy. The idea is intuitively appealing because the connection between the transitions of the system and the higher level policy, often expressed as difficult to check relations on execution traces, is given by an *unwinding theorem*.

There is a substantial amount of work on unwinding of information flow policies [3,5,15,13,10,16]. Each of these results are developed for a specific definition(s) and hence they lack modularity. This is unfortunate, as it results in the need to reprove many similar results. In an attempt to remedy this, Mantel [7,8] introduced a *modular* framework in which most well-known information flow

policies can be composed from a set of *basic security predicates* (BSPs). A major advantage of Mantel's framework is precisely its modularity: BSPs common to different definitions need to be verified only once per system; the same holds for unwinding relations. Interestingly, some BSPs are equivalent to, and can be constructed as, conjunctions of unwinding relations, whereas other BSPs are over-approximated by conjunctions of unwinding relations. Mantel's unwinding relations are noteworthy for at least two major reasons. First, because they can be *arbitrary* relations rather than equivalence relations, as are typically found in the literature. And second, because they can be specified *locally*, on states of the system, as opposed to the more traditional, global, trace-based unwinding relations. In addition to the local unwinding relations for his BSPs, Mantel presented *unwinding theorems* for most known possibilistic security policies.

Later on, Clarkson and Schneider introduced the notion of *hyperproperties* [2] in order to formalize security policies. A *hyperproperty* is a set of sets of (infinite) execution traces, or alternatively a property on trace sets or a second-order predicate over execution traces. Hyperproperties generalize properties and are expressive enough to capture most interesting security policies on systems, including notions of noninterference, but also many other policies. The notion of a hyperproperty is intuitively appealing, because it represents the set of systems permitted by some policy. Unfortunately, a generic verification methodology for hyperproperties does not exist. In this work, we make a step towards such a methodology based on unwinding.

The problem of directly using Mantel's framework for verification of hyperproperties, in particular for security-relevant ones, is that the framework is geared towards reasoning about only terminating behaviors (finite systems). In order to enable reasoning about infinite systems, particularly about systems having confidential events occurring infinitely often, we illustrate the need of and propose a new mathematical development. We present a coinductive reinterpretation of Mantel's unwinding relations, BSPs and security-relevant hyperproperties. The security relevant hyperproperties are different than the respective policies studied by Mantel, required by the fact that systems are possibly infinite. This results in different definitions of the BSPs and the respective unwinding relations. Another consequence is that security policies have different semantics on finite systems compared to the ones presented by Mantel. Further, we show that the respective variants of a number of the unwinding theorems hold as expected. Our contribution opens the door to verification of nontrivial, potentially infinite systems w.r.t. security relevant hyperproperties. Moreover, it reuses key ideas from Mantel's framework, which is both well-established and conceptually appealing. Finally, it sheds light on the significance of *incremental hyperproperties*, recently proposed by Milushev and Clarke [12].

The rest of the paper is structured as follows. Section 2 provides some background material and motivation. Section 3 introduces the proposed coinductive variants of some well-known holistic security hyperproperties and their respective BSPs. Section 4 presents the proposed coinductive variants of the unwinding relations of selected BSPs. Section 5 presents the coinductive versions of three

types of theorems: firstly, theorems connecting unwinding relations and BSPs, secondly, ones connecting BSPs and holistic hyperproperties and thirdly, a version of Mantel's unwinding theorems, which can be used directly for verification. In Section 6 we discuss our main contributions and compare them with related work. Finally, we conclude by summarizing our results and sharing some ideas for future work. The proofs and more examples can be found in the accompanying technical report [11].

## 2 Background and Motivation

### 2.1 Background

Let $A$ be a fixed alphabet of abstract observations, sometimes called *events*. A *string* is a finite sequence of elements of $A$. The set of all strings over $A$ is denoted $A^*$. A *stream* of $A$'s is an infinite sequence of elements of $A$. The set of all streams over $A$ is $A^\omega = \{\sigma \mid \sigma : \{0, 1, 2, \ldots\} \to A\}$. A stream $\sigma$ can be specified in terms of its first element $\sigma(0)$ and its stream derivative $\sigma'$, given by $\sigma'(n) = \sigma(n+1)$; these operators are also known as *head* and *tail*. A *trace* is a finite or infinite sequence of elements of $A$. The set of all traces over $A$ is denoted $A^\infty = A^* \cup A^\omega$. Let 2 be any two element set, for instance $2 = \{true, false\}$. A *system* is a set of traces. The set of all systems is $\mathsf{Sys} = 2^{A^\infty}$, the set of infinite systems is $\mathsf{Sys}_\omega = 2^{A^\omega}$.

**Properties vs. Hyperproperties.** Clarkson and Schneider present a theory of policies based on properties and hyperproperties [2]. A *property* is a set of traces. The set of all properties is $\mathsf{Prop} = 2^{A^\infty}$. A *hyperproperty* is a set of sets of traces or equivalently a set of properties. The set of all hyperproperties is $\mathsf{HP} = 2^{2^{A^\infty}} = 2^{\mathsf{Prop}} = 2^{\mathsf{Sys}}$. Note that our definition, unlike the original one, does not require all traces to be infinite; as a result termination-sensitive definitions can be expressed in a more natural fashion.

**Partial Automata.** We model systems as partial automata [14]. A *partial automaton* with input alphabet $A$ and a start state is defined *coalgebraically* as a 4-tuple $\langle S, o, t, s_0 \rangle$, where set $S$ is the possibly infinite state space of the automaton, the observation function $o : S \to 2$ says whether a state is accepting or not, the function $t : S \to (1 + S)^A$ gives the transition structure and $s_0$ is the initial state. Notation $S^A$ stands for the set of functions with signature $A \to S$; $1 + S$ is notation used for the set $\{\bot\} \cup S$: whenever the function $t(s)$ is undefined, it is the constant function mapping $A$ to $\bot$; if $t(s)$ is defined for some $a \in A$, then $t(s)(a) = s'$ gives the next state.

Let $A^* \cdot \delta = \{w \cdot \delta \mid w \in A^*\}$ be the set of finitely deadlocked words. Note that $\delta \notin A$ is a special symbol to signify divergence. The collection of all languages acceptable by partial automata is $A_\delta^\infty = A^* \cup (A^* \cdot \delta) \cup A^\omega$. For words $w \in A^*$ and sets $L \subseteq A_\delta^\infty$, define the *w-derivative* of $L$ to be $L_w = \{v \in A_\delta^\infty \mid w \cdot v \in L\}$. Finally, let $t|_Z$ be the projection of a string $t$ to elements from some set $Z$.

**Trees.** Following our recent work [12], we model system behavior as potentially infinite trees instead of as sets of traces. A *tree* is obtained from a language (or set of traces) by continuously taking derivatives with respect to elements of $A$. The start state of the system corresponds to the root of the tree and vertices correspond to sets of traces (languages).

We define a particular pair of functions $\langle o, t\rangle$ allowing us to switch perspective from seeing a system as a set of traces to seeing it as a partial automaton. It should be noted that such a pair of functions induces a unique tree, giving the behavior of the automaton and the set of traces (see [14] and [12] for details). Let $C \in \mathsf{Sys}$, $a \in A$ and $\sigma \in A^\infty$. First, define an auxiliary function $test : \mathsf{Sys} \to (A \to 2)$ as follows:

$$test_a(C) \;\widehat{=}\; \exists \sigma \in C \,.\, \sigma(0) = a.$$

Now, we can define the functions $o$ and $t$ as follows:

$$o(C) \;\widehat{=}\; \epsilon \in C \qquad t(C)(a) \;\widehat{=}\; \begin{cases} \{\sigma' \mid \sigma(0) = a\} & \text{if } test_a(C) \\ \bot & \text{if } \neg test_a(C). \end{cases}$$

**Auxiliary definitions.** We can straightforwardly extend *test* to an obvious inductive definition of predicate $test^* : \mathsf{Sys} \to (A^* \to 2)$ on words. For $a \in A$, $w \in A^*$ and $\epsilon$ the empty trace:

$$\frac{o(X)}{{test^*}_\epsilon(X)} \qquad \frac{test_a(X) \qquad {test^*}_w(X_a)}{{test^*}_{a\cdot w}(X)}$$

We also need a coinductive definition of trace inclusion in trees:

$$\frac{o(S)}{\epsilon \in S}\ {\scriptstyle coind} \qquad \frac{test_a(S) \qquad w \in S_a}{a \cdot w \in S}\ {\scriptstyle coind}$$

Note that coinductive definitions are indicated as *coind* on the right side of their respective inference rules.

**Incremental Hyperproperties as Coinductive Predicates on Trees.** Our recent work [12] introduced and formalized the notion of *incremental hyperproperties*. Such a hyperproperty is the greatest fixed point of a monotone function over $\mathsf{Sys}^n$, given in a fragment of Least Fixed Point Logic called $\mathcal{IL}$. More informally, incremental hyperproperties are *coinductive tree predicates* or alternatively coinductively defined relations on the state space of the system. The original notion of hyperproperties [2] is also formalized and called *holistic hyperproperties* [12].

### 2.2 Motivation

Many typical security-relevant policies (for instance all the ones presented in [8]) reason about finite only traces. As a result, such definitions are termination insensitive: informally, this implies that all diverging computations are considered

to be "the same". This is clearly not satisfactory when reasoning about potentially infinite systems, such as servers, embedded systems, operating systems etc. Moreover, the typical termination insensitive definitions allow leaks through *covert channels*. For instance, consider the simple system $S_1 = \{(hl)^\omega, l^*\}$, where $A = L \cup H$, $L = \{l\}$ and $H = \{h\}$ (*low* and *high* events). Clearly, termination insensitive definitions (such as $NF_o$ introduced in Section 3.1) distinguish this system as trivially *secure*: as only one of the traces is in $A^*$ and it has low events only. However, the system is intuitively not secure as termination is low-observable. In general, theoretical machinery is needed in order to reason about potentially infinite behaviors allowed by system specifications. Such machinery, at least for reasoning about security-relevant hyperproperties in general, is currently lacking. This is the main motivation of this work.

## 3 Coinductive Interpretation of Security-Relevant Hyperproperties

Security-relevant policies (notably notions of noninterference) have traditionally been defined using a model of finite traces, as well as inductively defined relations on those traces using existential and universal quantification. In order to be able to reason about potentially infinite behavior, the above mentioned relations have to be lifted to potentially infinite traces. Thus, a coinductive (re)interpretation of the well-known notions of noninterference is needed in order to reason about the same policies on (potentially) nonterminating systems. The reason is that when computations do not terminate, there is no longer a well-ordering and hence inductive definitions are ill-formed.

The coinductive interpretation of well-known, holistic, security hyperproperties requires coinductively defined predicates on traces and sometimes functions (often treated as relations). We start off by giving the needed definitions. Note that set $Z \subseteq A$ used below is assumed to be *non-empty*.

**Definitions.** Coinductively define predicate $no_Z : A^\infty \to 2$ (parameterized by set $Z$), which states that there are no events from set $Z$ in a trace:

$$\frac{}{no_Z(\epsilon)}\ {\scriptstyle coind} \qquad \frac{a \in A \setminus Z \qquad no_Z(x)}{no_Z(a \cdot x)}\ {\scriptstyle coind}$$

Note that $no_Z(t)$ is the coinductive version of predicate $t|_Z = \epsilon$. Next, inductively define $w \rightsquigarrow_Z a \cdot w'$ ($w$ $Z$-reveals $a$ with tail $w'$):

$$\frac{}{\epsilon \rightsquigarrow_Z \epsilon} \qquad \frac{a \in Z}{a \cdot w \rightsquigarrow_Z a \cdot w} \qquad \frac{b \in A \setminus Z \qquad w \rightsquigarrow_Z a \cdot w'}{b \cdot w \rightsquigarrow_Z a \cdot w'}$$

We also need a coinductive relation $ev_Z$, relating any trace to its projection onto $Z$. Technically, the relation is a partial function, filtering out events from set $Z$, and will be defined and used as one, mainly to keep the connection to $t|_Z$.

$$\frac{}{ev_Z(\epsilon) = \epsilon}\ {\scriptstyle coind} \qquad \frac{w \rightsquigarrow_Z a \cdot w' \qquad ev_Z(w') = u}{ev_Z(w) = a \cdot u}\ {\scriptstyle coind}$$

Finally, coinductively define *weak bisimulation* (parameterized by set $Z$) $\approx_Z : A^\infty \times A^\infty \to 2$ as follows:

$$\frac{}{\epsilon \approx_Z \epsilon}\ {\scriptstyle coind} \qquad \frac{w \rightsquigarrow_Z a \cdot w' \qquad u \rightsquigarrow_Z a \cdot u' \qquad w' \approx_Z u'}{w \approx_Z u}\ {\scriptstyle coind}$$

Note that the inductive/coinductive part of the definition avoids the potential fairness problem where $\tau^\omega$ is equivalent to any trace.

Next, we present definitions adopted from Mantel's MAKS framework [8]. For a partition of alphabet $A$ as $A = A_v \cup A_n \cup A_c$, define a *view* to be a tuple $V = (A_v, A_n, A_c)$ corresponding to *visible*, *neither* visible *nor* confidential (i.e. neutral) and *confidential* events. Let $\mathcal{H}$ denote the view $(L, \emptyset, H)$ where $H$ and $L$ are the sets of *high* and *low* events, and the set of neutral events is empty. Let sets $I$ and $O$ represent inputs and outputs such that $I \subseteq A$, $O \subseteq A$ and $I \cap O = \emptyset$. Let $\mathcal{HI}$ denote the view $(L, H \setminus HI, HI)$, where $HI$ is the set of high inputs, i.e. $H \cap I$. Let the set of all views be $\mathcal{V}$ and $\rho$ be a function from *views* to subsets of $A$, i.e. $\rho : \mathcal{V} \to 2^A$. An event is defined to be $\rho$-admissible in a tree $T$ after a possible finite trace $\beta$ for some view $V = (A_v, A_n, A_c)$ if $Adm^\rho_V(T, \beta, e)$ holds, where $Adm^\rho_V(T, \beta, e)$ is defined:

$$Adm^\rho_V(T, \beta, e) \ \widehat{=}\ \exists \gamma \in A^*.(\gamma \cdot e \in T \wedge \gamma \approx_{\rho(V)} \beta).$$

Intuitively, $\rho$ can give a finer grained distinction of events than a view. For instance, one policy might be defined as follows: given some view $(A_v, A_n, A_c)$ and from observing events in $A_v$ one should not be able to deduce occurrence/nonoccurrence of $\rho$-admissible events in $A_c$ (or some subset of it).

### 3.1 Coinductive view on some well-known holistic security-relevant hyperproperties

The definitions presented next are well-known from the literature, but we present their respective variants on potentially infinite systems (as hyperproperties).

**Noninference** This policy will be called $NF_o$ (original $NF$) and is originally defined on finite systems as follows [17]:

$$NF_o(X) \ \widehat{=}\ \forall x \in X.\ x|_L \in X.$$

The coinductive variant of *noninference* is called $NF$ and given using $ev_Z$:

$$NF(X) \ \widehat{=}\ \forall x \in X.\ ev_L(x) \in X.$$

**Generalized Noninference** This policy is originally proposed by Zakinthinos and Lee [17] and given as follows:

$$GNF_o(X) \ \widehat{=}\ \forall x_0 \in X\ \exists x_1 \in X.\ (x_1|_{HI} = \epsilon \wedge\ x_1|_L = x_0|_L).$$

Our coinductive interpretation of *generalized noninference* $GNF$ is given here:

$$GNF(X) \ \widehat{=}\ \forall x_0 \in X\ \exists x_1 \in X.\ (no_{HI}(x_1) \wedge x_1 \approx_L x_0).$$

For the following policies, we only give the coinductive definitions.

**Generalized Noninterference** Using coinductive relations on traces, we define *generalized noninterference GNI* as a hyperproperty:

$$GNI(X) \;\widehat{=}\; \forall x_1 \in A^* \; \forall x_2, x_3 \in A^\infty \; . \; [(x_1 \cdot x_2 \in X \wedge x_3 \approx_{A \setminus HI} x_2) \rightarrow \\ \exists x_4 \in A^\infty . \; (x_1 \cdot x_4 \in X \wedge x_4 \approx_{L \cup HI} x_3)].$$

Note that the equality of projections is replaced by our coinductively defined $\approx_Z$ relation.

**Perfect Security Property** Finally, we present our coinductive variant of the *perfect security property PSP* (proposed by Zakinthinos and Lee [17]):

$$PSP(X) \;\widehat{=}\; (\forall x \in X . \, ev_L(x) \in X) \wedge (\forall \alpha \in A^\infty \forall \beta \in A^* . \\ [(\beta \cdot \alpha \in X \wedge no_H(\alpha)) \rightarrow \forall h \in H . \; (\beta \cdot h \in X \rightarrow \beta \cdot h \cdot \alpha \in X)]).$$

We have presented only a few of the information flow definitions in order to illustrate what is needed to represent them as hyperproperties. It is relatively straightforward to convert any of the ones not presented here. Nevertheless, the examples are enough to cover a number of important BSPs and unwinding relations, as well as to raise some interesting questions, which will be presented in the following sections. Moreover, the examples suggest a possible technique to adapt Mantel's unwinding relations to reason about security-relevant hyperproperties and a connection with incremental hyperproperties, namely that an $H'$-simulation [12], implying that an incremental hyperproperty $H'$ holds, is a(n) (conjunction of) unwinding relation(s). This is further elaborated in Section 4.

### 3.2 Coinductive view on BSPs

Mantel introduces the MAKS framework [8,7], which can represent most well-known possibilistic security policies as conjunctions of Basic Security Predicates. Mantel classifies his BSPs in two dimensions. In the *first dimension* fall BSPs that essentially hide the *occurrence* of $A_c$-events, whereas in the *second dimension* are BSPs that hide the *non-occurrence* of $A_c$-events. Mantel's BSPs and security policies are defined on finite traces only. In this section we present a coinductive perspective on a number of the BSPs, parameterized by a *security view* (see Section 3). Although we have changed the definitions, we have kept their original names. We start with some BSPs from the first dimension.

**Removal of events.** Predicate $R_V(T)$ requires for any trace $\sigma \in T$ the existence of another trace $\gamma$ which has no events from $A_c$ and which has the same $A_v$-events (essentially allowing "corrections" of $A_n$-events). Our definition is:

$$R_V(T) \;\widehat{=}\; \forall \sigma \in T \; \exists \gamma \in T \; . \; (no_{A_c}(\gamma) \; \wedge \sigma \approx_{A_v} \gamma).$$

Note that we have replaced the relations on traces in the original work with coinductive ones, similarly to the modifications of the definitions from Section 3.1. Interestingly, such a straightforward modification will not be possible for the rest of the BSP definitions we explore.

**Stepwise deletion of events.** The original definition [8] changes any trace $\sigma$ in a candidate set $T$ by deleting the *last* occurrence of a confidential event and requires that the resulting trace can be corrected (by possibly inserting/deleting events in $A_n$ if it is not empty) resulting in a possible trace $\gamma$ in $T$. If we *naively* convert Mantel's definition to potentially infinite traces, we get the following:

$$DN_V(T) \;\widehat{=}\; \forall \alpha \in A^\infty \; \forall \beta \in A^* \; \forall c \in A_c \; . \; [(\beta \cdot c \cdot \alpha \in T \wedge no_{A_c}(\alpha)) \rightarrow \\ \exists \alpha' \in A^\infty, \beta' \in A^* \; . \; (\beta' \cdot \alpha' \in T \wedge \alpha' \approx_{A_v} \alpha \wedge no_{A_c}(\alpha') \wedge \beta' \approx_{A_v \cup A_c} \beta)].$$

This definition would work as expected on finite traces. Unfortunately, it is not well-suited for infinite traces. To illustrate this, consider the following example:

*Example 1.* Let $V = (A_v, A_n, A_c)$ be a view such that $A_v = \{l_1, l_2\}$, $A_n = \emptyset$ and $A_c = \{h_1, h_2\}$. Consider system $S_1 = \{(l_1 h_1 h_2 l_2)^\omega\}$. Intuitively system $S_1$ is not secure, as every time $l_2$ is observed it is clear that $h_1$ and $h_2$ must have occurred. Unfortunately, the definition of $DN_V$ does not capture this intuition, as system $S_1$ is trivially secure w.r.t. the definition. The reason for this problem is that *confidential events appear infinitely often*, thus there is no suffix $t$ for which $no_{A_c}(t)$ holds.

Potentially infinite traces are allowed in many useful systems (operating systems, reactive and embedded systems etc.) and oftentimes there is no last confidential event, as confidential events might occur infinitely often. Thus the definition needs to be changed. The following definition fixes the problem:

$$D_V(T) \;\widehat{=}\; \forall \alpha \in A^\infty \; \forall \beta \in A^* \; \forall c \in A_c \; . \; [\beta \cdot c \cdot \alpha \in T \rightarrow \\ \exists \alpha' \in A^\infty, \beta' \in A^* . \; (\beta' \cdot \alpha' \in T \wedge \beta' \approx_{A_v \cup A_c} \beta \wedge \alpha' \approx_{A_v \cup A_c} \alpha)].$$

This definition deletes *any* occurrence of a confidential event in a trace and then perturbs the resulting trace. Unfortunately, on finite traces the definition is not semantically equivalent to the original one. To see this consider the following:

*Example 2.* Let $V = (A_v, A_n, A_c)$ be a view such that $A_v = \{l_1, l_2\}$, $A_n = \emptyset$ and $A_c = \{h_1, h_2\}$. Consider system $S_2 = \{l_1 h_1 h_2 l_2, l_1 h_1 l_2, l_1 l_2\}$. It is easy to check that $D_V(S_2)$ does not hold (because $l_1 h_2 l_2$ has to be in $S_2$, but it is not). Nevertheless $S_2$ is secure w.r.t. Mantel's original definition, as well as w.r.t. our naive definition $DN_V$.

It should be noted that the definition $D_V$ (proposed here and used throughout the work) is stronger (it requires more possible traces and hence higher uncertainty for the attacker) than $DN_V$. In other words, $D_V(X) \rightarrow DN_V(X)$. Moreover, $D_V$ properly rejects systems exhibiting the pattern of $S_1$ as insecure; to see one reason why, note that $l_1 h_2 l_2 (l_1 h_1 h_2 l_2)^\omega \notin S_1$.

**Backwards strict deletion.** The next BSP is called *BSD*. The intuitive idea is that the occurrence of an $A_c$-event should not be deducible. The difference with $D_V$ is that the part of the trace that has already occurred ($\beta$) cannot be changed. Our coinductive definition of $BSD_V(T)$ is given as:

$$BSD_V(T) \;\widehat{=}\; \forall \alpha \in A^\infty\, \forall \beta \in A^*\, \forall c \in A_c\, .\, [\beta \cdot c \cdot \alpha \in T \rightarrow \exists \alpha' \in A^\infty .\, (\beta \cdot \alpha' \in T \wedge \alpha' \approx_{A_v \cup A_c} \alpha)].$$

Note that a similar modification as to $D_V$ is needed here and the reason again is to enable tackling systems which do not have a last confidential event.

Although the BSP definitions have changed, the following theorem establishes a connection between the BSPs, familiar from Mantel's work.

**Theorem 1.** *Let $V = (A_v, A_n, A_c)$ be a view and $T$ be a set of traces. Then the following implications hold: $BSD_V(T) \rightarrow D_V(T)$ and $D_V(T) \rightarrow R_V(T)$.*

**Strict Deletion.** Our version of this BSP is again different than Mantel's: as in the previous definition, it does not search for the last $A_c$-event, hence it works on infinite traces. Our coinductive version of $SD_V(T)$ is given next:

$$SD_V(T) \;\widehat{=}\; \forall \alpha \in A^\infty\, \forall \beta \in A^*\, \forall c \in A_c\, .\, [\beta \cdot c \cdot \alpha \in T \rightarrow \beta \cdot \alpha \in T].$$

The rest of the presented BSPs are from the second dimension, hiding the *non-occurrence* of $A_c$-events.

**Backwards strict insertion.** This BSP is in a sense dual to $BSD_V$— instead of deleting an $A_c$-event, it requires the possible insertion of such an event. Of course, we have again modified the definition to a coinductive one and it does not search for the last $A_c$-event, hence it works on infinite traces. The same also holds for all the following definitions. Our coinductive version of $BSI_V(T)$ is:

$$BSI_V(T) \;\widehat{=}\; \forall \alpha \in A^\infty\, \forall \beta \in A^*\, \forall c \in A_c\, .\, [\beta \cdot \alpha \in T \rightarrow \exists \alpha' \in A^\infty .\, (\beta \cdot c \cdot \alpha' \in T \wedge \alpha' \approx_{A_v \cup A_c} \alpha)].$$

**Backwards strict insertion of admissible events.** This BSP is similar to $BSI_V$, but it hides the non-occurrence of *admissible* events only. Our coinductive version of $BSIA^\rho_V(T)$ is given as follows:

$$BSIA^\rho_V(T) \;\widehat{=}\; \forall \alpha \in A^\infty\, \forall \beta \in A^*\, \forall c \in A_c\, . \; [(\beta \cdot \alpha \in T \wedge Adm^\rho_V(T, \beta, c)) \rightarrow \exists \alpha' \in A^\infty .\, (\beta \cdot c \cdot \alpha' \in T \wedge \alpha' \approx_{A_v \cup A_c} \alpha)].$$

**Strict Insertion.** This BSP requires the possibility to insert any $A_c$-event at any place in a stream, it is strict because neither the past nor the future part of the trace may be changed. Our coinductive version of $SI_V(T)$ is given as follows:

$$SI_V(T) \;\widehat{=}\; \forall \alpha \in A^\infty\, \forall \beta \in A^*\, \forall c \in A_c\, .\, [\beta \cdot \alpha \in T \rightarrow \beta \cdot c \cdot \alpha \in T].$$

**Strict Insertion of $\rho$-admissible events.** This BSP requires the possibility to insert any $\rho$-admissible $A_c$-event at any place (where admissible) in a stream. Our coinductive version of $SIA^{\rho}_{V}(T)$ is given as follows:

$$SIA^{\rho}_{V}(T) \;\widehat{=}\; \forall\alpha \in A^{\infty}\; \forall\beta \in A^{*}\; \forall c \in A_c\;. \\ [(\beta \cdot \alpha \in T \wedge Adm^{\rho}_{V}(T,\beta,c)) \rightarrow \beta \cdot c \cdot \alpha \in T].$$

**Insertion of $\rho$-admissible events.** This BSP is similar to $SIA^{\rho}_{V}$, except that the definition is not strict (perturbations of the front and back parts of the trace are possible). Our coinductive version of $IA^{\rho}_{V}(T)$ is given as follows:

$$IA^{\rho}_{V}(T) \;\widehat{=}\; \forall\alpha \in A^{\infty}\; \forall\beta \in A^{*}\; \forall c \in A_c. \\ [(\beta \cdot \alpha \in T \wedge Adm^{\rho}_{V}(T,\beta,c)) \rightarrow \\ \exists\alpha' \in A^{\infty}.\exists\beta' \in A^{*}.\; (\beta' \cdot c \cdot \alpha' \in T \wedge \alpha' \approx_{A_v \cup A_c} \alpha \wedge \beta' \approx_{A_v \cup A_c} \beta)].$$

## 4 Coinductive interpretation of BSP unwinding relations

Instead of verifying BSPs directly or via global unwinding conditions, Mantel proposes the use of *local* unwinding conditions [7]. Essentially, the existence of an *unwinding relation(s)* satisfying a set of unwinding conditions has to be shown in order to prove that a number of BSPs hold and hence a particular policy is respected. In this section, we present a coinductive reinterpretation of Mantel's unwinding relations, which is needed in order for them to be suitable for non-terminating systems. We also show that the relations are instances of our *$H'$-simulation* relations [12].

The coinductively defined unwinding relations are presented next. The first relation is historically called *output-step consistency* and denoted $osc_V$. Defined coinductively, an *$osc_V$-simulation* is a relation $R$ such that for all $X, Y \in \mathsf{Sys}$ if $(X, Y) \in R$, then

$$o(X) \leftrightarrow o(Y) \bigwedge \forall a \in A \setminus A_c.\; (test_a(X) \rightarrow \\ \exists\sigma \in (A \setminus A_c)^{*}.(test^{*}{}_{\sigma}(Y) \wedge \sigma \approx_{A_v} a \wedge (X_a, Y_\sigma) \in R)).$$

An $osc_V$-simulation relation $R$ is denoted $osc_V(R)$. We will often overload notation and state $osc_V(X, Y, R)$ iff $osc_V(R)$ and $(X, Y) \in R$.

Next, define an *$lrf_V$-simulation* as follows: a relation $R$ such that for all $X, Y \in \mathsf{Sys}$ if $(X, Y) \in R$, then

$$o(X) \leftrightarrow o(Y) \bigwedge \forall a \in A_c.\; (test_a(X) \rightarrow (X_a, Y) \in R).$$

Next, define an *$lrb_V$-simulation* relation as follows: a relation $R$ such that for all $X, Y \in \mathsf{Sys}$ if $(X, Y) \in R$, then

$$o(X) \leftrightarrow o(Y) \bigwedge \forall a \in A_c.\; (test_a(Y) \wedge (X, Y_a) \in R).$$

Next, define $En^{\rho}_{V}(X, s, a)$, saying whether event $a$ is enabled in state $s$ of system $X$ w.r.t. a set of admissible events given by function $\rho$ (see Section 3):

$$En^{\rho}_{V}(X, s, a) \;\widehat{=}\; \exists \beta, \gamma \in A^*.[{test^*}_{\beta}(X) \;\wedge\; s = X_{\beta} \;\wedge\; o(s) \;\wedge\; \gamma \approx_{\rho(V)} \beta \;\wedge\; {test^*}_{\gamma}(X) \wedge o(X_{\gamma}) \wedge test_a(X_{\gamma}) \wedge o(X_{\gamma \cdot a})].$$

Finally, define an *$lrbe^{\rho}_{V}$-simulation* relation as follows: a relation $R$ such that if $(X, Y) \in R$, then

$$o(X) \leftrightarrow o(Y) \bigwedge \forall a \in A_c.\ (En^{\rho}_{V}(T, Y, a) \rightarrow (test_a(Y) \wedge (X, Y_a) \in R)).$$

Next, we show that the relations defined in this section are indeed $H'$-simulations. First, recall that incremental hyperproperties are coinductive predicates on trees [12]. Formally, an *$H'$-simulation* is an $n$-ary relation $R$ such that $R \subseteq \Psi_{H'}(R)$. An $H'$-simulation corresponds to a monotone operator $\Psi_{H'}$ whose greatest fixed point is the coinductive predicate $H'$. Hence showing the existence of such a relation is sufficient to show that $H'$ holds [12]. Because of the way the relations are defined, it is obvious that $R \subseteq \Psi_{H'}(R)$ holds; informally, $\Psi_{H'}$ is the "step" of the relation. Thus, the relations are indeed *$H'$-simulation* relations.

## 5 Coinductive interpretation of the theory

We have taken a coinductive perspective on Mantel's unwinding relations [8]. The high-level goal is to properly incorporate the unwinding relations in our framework in order to facilitate the verification of security-relevant hyperproperties. To show that we have succeeded in this, we present three types of theorems, very similar to the ones initially introduced by Mantel in his framework: firstly, theorems connecting unwinding conditions and BSPs, secondly, ones connecting BSPs and holistic hyperproperties and finally, a version of Mantel's unwinding theorems.

The fact that we can prove these theorems implies that our definitions of unwinding relations, BSPs and holistic hyperproperties are reasonable and, more importantly, that our framework is suitable for the verification of a number of security-relevant policies via unwinding.

### 5.1 Unwinding conditions to BSPs theorems

The following two lemmas prove the intuition of $osc_V$-simulation relations: states related by such a relation are indistinguishable to the $A_v$ part of the view.

**Lemma 1.** *Let $R \subseteq \mathsf{Sys} \times \mathsf{Sys}$ be an arbitrary relation and $T, S$ arbitrary systems. If $osc_V(T, S, R)$ holds for some $T, S \in \mathsf{Sys}$ then we have*

$$\forall \alpha_1 \in (A \setminus A_c)^*.({test^*}_{\alpha_1}(T) \;\rightarrow\; \exists \alpha_2 \in (A \setminus A_c)^*.\ ({test^*}_{\alpha_2}(S) \wedge \alpha_1 \approx_{A_v} \alpha_2 \wedge (T_{\alpha_1}, S_{\alpha_2}) \in R)).$$

**Lemma 2.** *For all $T, S \in \mathsf{Sys}$ if there exists $R \subseteq \mathsf{Sys} \times \mathsf{Sys}$ s.t. $osc_V(T, S, R)$ holds, then the following is valid:*

$$\forall \alpha_1 \in (A \setminus A_c)^\infty.(\alpha_1 \in T \ \rightarrow \exists \alpha_2 \in (A \setminus A_c)^\infty.(\alpha_2 \in S \wedge \alpha_1 \approx_{A_v} \alpha_2)).$$

The next result gives logically sufficient conditions (conjunctions of unwinding relations) for a number of BSPs. This is not surprising (Mantel presents a similar result), but it is nevertheless important, because the definitions have changed.

**Theorem 2.** *Let $R \subseteq \mathsf{Sys} \times \mathsf{Sys}$ be an arbitrary relation and $T$ an arbitrary system. The following implications are valid:*

1. $lrf_V(T, T, R) \wedge osc_V(T, T, R) \rightarrow BSD_V(T)$
2. $lrf_V(T, T, R) \wedge osc_V(T, T, R) \rightarrow D_V(T)$
3. $lrf_V(T, T, R) \wedge osc_V(T, T, R) \rightarrow R_V(T)$
4. $lrbe^\rho_V(T, T, R) \wedge osc_V(T, T, R) \rightarrow BSIA^\rho_V(T)$
5. $lrb_V(T, T, R) \wedge osc_V(T, T, R) \rightarrow BSI_V(T)$.

The following theorem gives a conditional completeness result (when $A_n = \emptyset$) for some BSPs. Further results on this have been left out due to space limitations.

**Theorem 3.** *Consider a view $(A_v, A_n, A_c)$ s.t. $A_n = \emptyset$. The following are valid:*

1. *$BSD_V(T)$ implies there exists a relation $R \subseteq \mathsf{Sys} \times \mathsf{Sys}$ s.t. $lrf_V(T, T, R)$ and $osc_V(T, T, R)$ hold.*
2. *$BSIA^\rho_V(T)$ implies there exists a relation $R \subseteq \mathsf{Sys} \times \mathsf{Sys}$ s.t. $lrbe^\rho_V(T, T, R)$ and $osc_V(T, T, R)$ hold.*

### 5.2 Coinductive version of BSPs to holistic hyperproperties theorems

This section presents useful results, relating BSPs and the holistic, security-relevant hyperproperties, introduced in Section 3.1. First, recall that $\mathcal{HI} = (L, H \setminus HI, HI)$. The instantiation of $BSD_V$ with view $\mathcal{HI}$ is given as follows:

$$\begin{aligned} BSD_{\mathcal{HI}}(T) \ \widehat{=} \ \forall \alpha \in A^\infty \ \forall \beta \in A^* \ \forall c \in HI \ . \ [(\beta \cdot c \cdot \alpha \in T \rightarrow \\ \exists \alpha' \in A^\infty. \ (\beta \cdot \alpha' \in T \wedge \alpha' \approx_{L \cup HI} \alpha)]. \end{aligned}$$

The instantiation of $BSI_V$ with view $\mathcal{HI}$ is given as follows:

$$\begin{aligned} BSI_{\mathcal{HI}}(T) \ \widehat{=} \ \forall \alpha \in A^\infty \ \forall \beta \in A^* \ \forall c \in HI \ . \ [(\beta \cdot \alpha \in T \rightarrow \\ \exists \alpha' \in A^\infty. \ (\beta \cdot c \cdot \alpha' \in T \wedge \alpha' \approx_{L \cup HI} \alpha)]. \end{aligned}$$

The following result establishes the connection between certain BSPs and *GNI*.

**Theorem 4.** *For all $T \in \mathsf{Sys}$ we have $BSD_{\mathcal{HI}}(T) \wedge BSI_{\mathcal{HI}}(T)$ iff $GNI(T)$.*

The next theorem establishes that the holistic hyperproperty noninference is equivalent to the BSP removal of events, instantiated with view $\mathcal{H}$.

**Theorem 5.** *For all* $T \in \mathsf{Sys}$ *we have* $R_{\mathcal{H}}(T)$ *iff* $NF(T)$.

The following result claims that the holistic hyperproperty *GNF* is equivalent to the BSP removal of events, this time instantiated with view $\mathcal{HI}$.

**Theorem 6.** *For all* $T \in \mathsf{Sys}$ *we have* $R_{\mathcal{HI}}(T)$ *iff* $GNF(T)$.

Finally, we have proven Theorem 7, representing *PSP* as a conjunction of BSPs.

**Theorem 7.** *For all* $T \in \mathsf{Sys}$ *we have* $BSD_{\mathcal{H}}(T) \wedge BSIA^{\rho_A}_{\mathcal{H}}(T)$ *iff* $PSP(T)$.

### 5.3 Coinductive version of Mantel's unwinding theorems

Finally, we present the coinductive unwinding theorems for a number of known security-relevant hyperproperties. These unwinding theorems allow the specification and verification of the high-level policy by reasoning about the local states of the candidate system. Interestingly, there is a completeness result only for the definition of *PSP*. Similar theorems have been shown before, but for different unwinding relation, BSP and hyperproperty definitions and different models (for instance see the MAKS framework [8]).

**Noninference *NF*** We have proven an unwinding theorem for *NF*, giving logically sufficient conditions. Because we only have an implication, there may be secure systems for which the needed unwinding relation does not exist.

**Theorem 8 (Unwinding of *NF*).** *If there exists a relation* $R \subseteq \mathsf{Sys} \times \mathsf{Sys}$ *such that* $lrf_{\mathcal{H}}(T,T,R) \wedge osc_{\mathcal{H}}(T,T,R)$*, we have that* $NF(T)$ *holds.*

**Generalized Noninference *GNF*** Further, we have proven an unwinding theorem for *GNF*, again giving logically sufficient conditions. Unfortunately, such conditions are again not necessary.

**Theorem 9 (Unwinding of *GNF*).** *If there exists a relation* $R \subseteq \mathsf{Sys} \times \mathsf{Sys}$ *such that* $lrf_{\mathcal{HI}}(T,T,R) \wedge osc_{\mathcal{HI}}(T,T,R)$*, we have that* $GNF(T)$ *holds.*

**Generalized Noninterference *GNI*** We have also been able to prove an unwinding theorem, giving logically sufficient conditions for *GNI*.

**Theorem 10 (Unwinding of *GNI*).** *If there exist relations* $R, Q \subseteq \mathsf{Sys} \times \mathsf{Sys}$ *such that* $lrf_{\mathcal{HI}}(T,T,R) \wedge osc_{\mathcal{HI}}(T,T,R)$*, as well as* $lrb_{\mathcal{HI}}(T,T,Q) \wedge osc_{\mathcal{HI}}(T,T,Q)$*, we have that* $GNI(T)$ *holds.*

**Perfect Security Property *PSP*** The *Perfect Security Property* is special, as there are necessary and sufficient conditions. We have been able to show this in our framework as well. First, let $\rho_A(A_v, A_n, A_c) = A_v \cup A_n \cup A_c = A$.

**Theorem 11 (Unwinding of *PSP*).** *There exist relations* $R, Q \subseteq \mathsf{Sys} \times \mathsf{Sys}$ *such that* $lrf_{\mathcal{H}}(T,T,R) \wedge osc_{\mathcal{H}}(T,T,R)$*, as well as* $lrbe^{\rho_A}_{\mathcal{H}}(T,T,Q) \wedge osc_{\mathcal{H}}(T,T,Q)$*,* iff $PSP(T)$ *holds.*

This theorem gives unwinding relations for *PSP*. Moreover (unlike for the other definitions), for *PSP* we know that if no relations $R, Q \subseteq \mathsf{Sys} \times \mathsf{Sys}$ such that $lrf_{\mathcal{H}}(T,T,R) \wedge osc_{\mathcal{H}}(T,T,R)$ exist, the candidate system $T$ is not secure.

## 6 Discussion

We have presented a new mathematical development enabling the application of unwinding relations for the verification of security-relevant hyperproperties.

First, a novel *coinductive perspective* was taken on the security-relevant hyperproperties themselves, by adapting their definitions to allow reasoning about nonterminating behavior. Such a modification is important not only from a theoretical point of view, but also in practice. As a motivating example, consider systems with nonterminating behavior such that confidential events in all traces occur infinitely often (this is a liveness property). In such situations it is impossible to declare a system secure by examining only finite prefixes. A typical policy, for instance given by Mantel's deletion of events [8] or a naive coinductive interpretation of the latter ($DN_V$ from Section 3.2), would not be able to properly reason about such systems. Such policies would simply accept systems having only infinite behavior as being trivially secure (e.g. $S = \{(lhl)^\omega, (hll)^\omega, (llh)^\omega\}$). Intuitively, this is not desirable. As systems, exhibiting infinite behaviors and having no last confidential event, are abundant in practice (databases, operating systems, reactive and embedded systems), they are important for both specification and verification. Since hyperproperties are generic system specifications, it is natural to address the above-mentioned problems by giving them a coinductive semantics and use coinduction as a reasoning tool for such systems.

The only related paper that explores nonterminating behaviors and identifies the need for a coinductive interpretation of noninterference for potentially nonterminating systems is by Bohannon et al. [1]. They introduce the notion of *reactive noninterference* and explore variants suitable for reactive systems. The main similarity with our work is that they use coinductive and inductive/coinductive definitions in order to define relations on streams. They convert their high-level, holistic definition into a relation (called *ID-bisimulation*) on program states; they show that their ID-bisimulation implies the high level, holistic policy. Their ID-bisimulation is essentially an incremental hyperproperty.

Second, we have demonstrated the potential of a modular framework for coinductive reasoning about hyperproperties. This is achieved by combining the framework from our previous work [12] with a coinductive reinterpretation of Mantel's BSPs and unwinding relations. It should be noted that our proposed

coinductive variants of the BSPs are not equivalent to Mantel's on finite systems, nevertheless their conjunctions still imply the desired high level policies.

Third, we present a coinductive reinterpretation of Mantel's unwinding relations and argue that they are instances of our $H'$-simulations [12]. More precisely, an $H'$-simulation is a (conjunction of) unwinding relation(s). This realizes a connection between unwinding relations and incremental hyperproperties: incremental security hyperproperties can be seen as conjunctions of coinductively-defined unwinding relations, or alternatively as (conjunctions of) our $H'$-simulations [12], implying the respective high level policies. This is obvious if we consider some of the incrementalizable classes of hyperproperties [12], particularly SHH and OHH. Moreover, other interesting security-relevant hyperproperties, such as separability [17], forward correctability [6], nondeducibility of outputs [4] etc., having known representations as conjunctions of unwinding relations, can benefit from the techniques presented here.

Finally, we have presented a number of *unwinding theorems* for our coinductive reinterpretation of well-known security-relevant hyperproperties.

## 7 Conclusion

We have proposed a framework suitable for coinductive reasoning about hyperproperties in general and illustrated its usefulness by exploring a new coinductive reinterpretation of known noninterference policies as hyperproperties. The framework is modular, as it permits expressing a number of security-relevant hyperproperties as conjunctions of variants of Mantel's BSPs. We have demonstrated the usefulness of coinductive unwinding relations for reasoning about hyperproperties. In particular, we have presented unwinding theorems for generalized noninterference [9], noninference [17], generalized noninference [17] and the perfect security property [17]. Moreover, we have proven results connecting unwinding relations and BSPs, relating different BSPs and relating BSPs and holistic hyperproperties.

To the best of our knowledge, the results are novel in several ways. First we further develop our recently proposed framework for reasoning about hyperproperties [12] and establish a connection with the most relevant (in our opinion) work on verification via unwinding [7]. We also identify and illustrate the potential of unwinding relations (which turn out to be instances of our $H'$-simulations) for generic verification of hyperproperties. Further, we argue that coinductively defined hyperproperties are important not only from a theoretical standpoint, but also in practice, due to the abundance of nontrivial reactive systems. Finally, the results shed light on the significance of incremental hyperproperties.

In the future, we envision extending the work in two main directions: formally characterizing the class of security-relevant incremental hyperproperties and applying the framework for reasoning about reactive system security.

**Acknowledgements.** We thank Dominique Devriese for valuable comments on a draft of this paper and Tatyana Doktorova for helpful suggestions on the presentation. We also thank the anonymous reviewers for the constructive feedback.

## References


1. Aaron Bohannon, Benjamin C. Pierce, Vilhelm Sjöberg, Stephanie Weirich, and Steve Zdancewic. Reactive noninterference. In *Proceedings of the 16th ACM conference on Computer and communications security*, CCS '09, pages 79–90, New York, NY, USA, 2009. ACM.
2. Michael R. Clarkson and Fred B. Schneider. Hyperproperties. In *CSF '08: Proceedings of the 2008 21st IEEE Computer Security Foundations Symposium*, pages 51–65, Washington, DC, USA, 2008. IEEE Computer Society.
3. Joseph A. Goguen and José Meseguer. Unwinding and Inference Control. *IEEE Symposium on Security and Privacy*, pages 75–86, 1984.
4. Joshua D. Guttman and Mark E. Nadel. What Needs Securing? In *Proceedings of the IEEE Computer Security Foundations Workshop*, pages 34–57, 1988.
5. J. Thomas Haigh and William D. Young. Extending the Noninterference Version of MLS for SAT. *IEEE Transactions on Software Engineering*, 13(2):141–150, February 1987.
6. Dale M. Johnson and Javier F. Thayer. Security and the Composition of Machines. In *Proceedings of the IEEE Computer Security Foundations Workshop*, pages 72–89, 1988.
7. Heiko Mantel. Possibilistic Definitions of Security - An Assembly Kit. In *Proceedings of the 13th IEEE Workshop on Computer Security Foundations*, pages 185–199, Washington, DC, USA, 2000. IEEE Computer Society.
8. Heiko Mantel. *A Uniform Framework for the Formal Specification and Verification of Information Flow Security*. PhD thesis, Universität des Saarlandes, Saarbrücken, Germany, July 2003.
9. Daryl McCullough. Specifications for Multi-Level Security and a Hook-Up. *IEEE Symposium on Security and Privacy*, pages 161–166, 1987.
10. Jonathan Millen. Unwinding Forward Correctability. In *In Proceedings of the Computer Security Foundations Workshop*, pages 2–10. IEEE, 1994.
11. Dimiter Milushev and Dave Clarke. Coinductive unwinding of security-relevant hyperproperties: extended version. Technical Report CW 623, Katholieke Universiteit Leuven, August 2012.
12. Dimiter Milushev and Dave Clarke. Towards Incrementalization of Holistic Hyperproperties. In *Proceedings of the First International Conference on Principles of Security and Trust*, volume 7215, pages 329–348. Springer, March 2012.
13. John Rushby. Noninterference, Transitivity and Channel-Control Security Policies. Technical Report CSL-92-02, SRI International.
14. Jan J. M. M. Rutten. Automata and Coinduction (An Exercise in Coalgebra). In Davide Sangiorgi and Robert de Simone, editors, *CONCUR*, volume 1466 of *Lecture Notes in Computer Science*, pages 194–218. Springer, 1998.
15. P. Y. A. Ryan. A CSP formulation of non-interference and unwinding. *Cipher: IEEE Computer Society Technical Committee Newsletter on Security Privacy*, pages 19–30, March 1991.
16. Peter Y. A. Ryan and Steve A. Schneider. Process Algebra and Non-Interference. *Journal of Computer Security*, 9(1/2):75–103, 2001.
17. A. Zakinthinos and E. S. Lee. A general theory of security properties. In *Proceedings of the 1997 IEEE Symposium on Security and Privacy*, SP '97, pages 94–102, Washington, DC, USA, 1997. IEEE Computer Society.