Conformance-Based Doping Detection for Cyber-Physical Systems

Rayna Dimitrova, Maciej Gazda, Mohammad Reza Mousavi, Sebastian Biewer, Holger Hermanns

Conformance-Based Doping Detection for Cyber-Physical Systems*

Rayna Dimitrova[1], Maciej Gazda[1], Mohammad Reza Mousavi[2], Sebastian Biewer[3], and Holger Hermanns[3]

[1] Department of Computer Science, University of Sheffield, Sheffield, UK
[2] School of Informatics, University of Leicester, Leicester, UK
[3] Saarland University - Computer Science, Saarland Informatics Campus, Saarbrücken, Germany

Abstract. We present a novel and generalised notion of doping cleanness for cyber-physical systems that allows for perturbing the inputs and observing the perturbed outputs both in the time– and value–domains. We instantiate our definition using existing notions of conformance for cyber-physical systems. We show that our generalised definitions are essential in a data-driven method for doping detection and apply our definitions to a case study concerning diesel emission tests.

1 Introduction

System doping, in our terminology, is an intentional intervention causing a change in the system's normal behaviour against the interests of the user or other stakeholders (such as the society at large). Examples of system doping are widespread and range from vendors' enforcing a monopoly on chargers and spare parts (by checking for and refusing third-party chargers and spare parts, respectively) to tampering with exhaust emission in order to detect and pass emission tests. Doping can be the result of embedding a piece of code or smuggling a piece of electronic circuit into the system and it can be caused by the original developers or by hackers. Software and system doping has been studied in the past couple of years and rigorous theories for it have been developed [8, 15, 9]. These theories were subsequently adopted in order to detect doping, or formally, to check system cleanness [32, 10] (corresponding to the absence of doping).

In the present paper, we extend the theory of doping to the setting of cyber-physical systems (CPS) by exploiting the notions of conformance testing for CPS [1, 17, 33]. The existing theories of software doping define doping in terms of drastic deviations in output as a result of minor deviations in input, where the term "deviation" refers to differences in validity of propositions or values of

* This work is partly supported by the ERC Grant 695614 (POWVER) by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) grant 389792660 as part of TRR 248, see https://perspicuous-computing.science, by the Saarbrücken Graduate School of Computer Science, and by the Sino-German CDZ project 1023 (CAP).

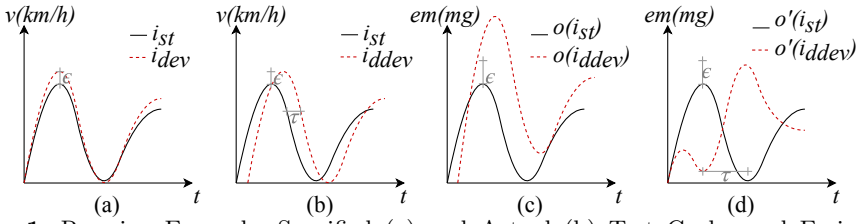

Fig. 1. Running Example: Specified (a) and Actual (b) Test Cycles and Emission Footprints obtained from Different (Fictitious) Vehicles (c) and (d).

variables. However, the current notions come short of properly dealing with the issues of retiming and delays, which are commonly present in the signals of CPS. We observe that this is an essential aspect of detecting doping for cyber-physical systems: often the traces to be tested for doping have subtly different timing behaviour, e.g., due to measurement and calibration errors or due to the slight deviations of human actors in acting upon the planned scenarios. The insufficient treatment of retiming and delays can both lead to false negatives, i.e., missing cases of doping, as well as false positives, i.e., reporting spurious doping cases.

To address these issues, we exploit the notion of conformance to devise a general theory of being clean from doping and instantiate that theory with some existing notions of conformance for hybrid systems. We show how these notions can account for retiming and lead to more precise notions of cleanness.

We illustrate the usefulness of our theory by empirical analysis of diesel engine exhaust emissions in the context of one of the official test cycles, the New European Driving Cycle (NEDC) [42]. In particular, we show that catering for retiming is essential in effectively exploiting the actual driving cycles for performing doping analysis. We thus demonstrate that our new theory remedies a major shortcoming in the existing notions from the literature. To facilitate the presentation, we use throughout the remainder of this paper the following simple running example, which is inspired by our case study.

Example 1. Fig. 1.(a) shows two test cycles (evolution of speed over time), designed to detect whether the exhaust emission control of a particular vehicle is doped. The test cycle i_{st}, depicted with a black solid line, is the standard one prescribed by the (fictitious) official regulation, while test cycle i_{dev}, depicted by a red dotted line, is a slight deviation thereof. If the exhaust emissions measured during the test cycle i_{dev} turn out to be significantly higher than the ones measured in test cycle i_{st}, then we can conclude that the exhaust emission system is potentially doped, since it appears tailored to the standard test cycle.

Fig. 1.(b) addresses a notorious problem of testing cars: a human tester is supposed to drive the car as just described, however, she can do this only up to a certain imprecision. Assume her driving of i_{dev} exhibits a slight time shift τ relative to the test cycle, as in i_{ddev}, while i_{st} is being driven as intended.

The result of a test is the emission footprint measured at the exhaust pipe of the car. Fig. 1.(c) and Fig. 1.(d) show two different possible test results (obtained from different cars) for the scenario in Fig. 1.(b). Intuitively, the footprints in

Fig. 1.(c) provide significant evidence for doping – a slightly different test cycle has resulted in significantly larger footprint. However, due to the time shift on the input side Fig. 1.(b) the point-wise difference of the two driven test-cycles has grown very large. As we show in the remainder of this paper, the existing theory of doping fails to detect such a clear evidence, due to the minor delay during the execution of the driving cycle. The emission footprint in Fig. 1.(d) is another (synthetic) example of a significant deviation which cannot be detected for the input in Fig. 1.(b) using existing theories; this latter footprint sheds some light on the intricate design decisions in the theory we develop in this paper.

The contributions for this paper can be summarized as follows:

- We define a *general notion of conformance* that can express different ways of comparing execution traces by allowing deviations both in value and in time.
- We define a general *notion of cleanness for hybrid systems*, and show that it subsumes the existing notion of robust cleanness [15].
- We demonstrate the usefulness of the proposed generic framework by applying it to *software doping tests* in the automotive domain, where we show that the new cleanness definition is able to flag a case of software doping that goes unnoticed when robust cleanness is used.

2 Related Work

The term "software doping" was coined around 2015 [30] in media uncovering the diesel exhaust emissions scandal. An informal problem formulation [8] pointed out the general phenomenon of intentionally added hidden software behaviour, which is not in the interest of the consumer. Shortly after, this observation has been complemented by a set of formal *cleanness* definitions [15] laying the theoretical foundations upon which formal methods to detect such software behaviour can be used. It is possible to detect missing functionality and undesired existing functionality. The definitions support both sequential programs and non-deterministic reactive programs. To check satisfaction of the definitions, it is necessary to compare two (or more) execution traces of the same system. Such properties are called *hyperproperties* [13] (whereas classical properties are *trace properties*). Tool support for analysing hyperproperties typically requires high computational effort [12, 25]. There exist several temporal logics for analysing satisfaction of trace properties of various kinds of systems, one of them being *Linear Temporal Logic* (LTL) [39] for systems producing outputs in discrete time steps and properties that do not consider the time passing between outputs. LTL has been extended to the logic HyperLTL, which can express hyperproperties by allowing explicit quantification of execution traces in front of an LTL formula [12]. Tools for model-checking boolean circuits, satisfiability and monitoring of HyperLTL specifications have been developed [6, 11, 25, 21–24, 29].

Signal Temporal Logic (STL) [36] is an extension of LTL that adds support for time constraints and real-valued signals. Tools exist that automatically try to falsify STL formulas [18, 7]. There has been an extension of STL to HyperSTL

in a similar fashion as it was done for HyperLTL [37]. The syntax of HyperSTL, however, is not able to express the cleanness definitions (for deterministic systems) in a way that allows (efficient) falsification. *Robust cleanness* is defined for distance functions on inputs and outputs [15]. When used with temporal logics the distance functions are restricted to those compatible with the logics. To be fully independent, robust cleanness analysis has been embedded into the theory of model-based testing [10] with input-output conformance [40, 41].

Notions of conformance for discrete event systems have been discussed for almost a century. The earliest work on this topic dates back to 1960's when researchers studied model-based testing of digital circuits using Finite State Machine models [31, 35]. Concurrency theory contributed ideas to this field, such as decoupling (i.e., removing the synchronised assumption between) inputs and outputs and observing failures to engage in a communication (and more specifically quiescence) [16, 40]. A theory of conformance testing for systems with continuous dynamics was developed by Michiel van Osch [38]; this theory did not gain much popularity in practice, partly because of its insufficient treatment of approximation (e.g., differences in values and retiming). Pappas and Girard [27, 28] proposed the use of Metric Bisimulation for conformance checking in dynamical systems and Pappas and Fainekos [20] developed a falsification framework for the same purpose. This ressearch led to two notions of conformance used in the present paper, namely hybrid conformance by Abbas and Fainekos [1] and Skorokhod conformance by Deshmukh, Majumdar, and Prabhu [17].

3 Preliminaries

Semantic domain. In this section, we provide definitions regarding semantic domain, conformance, and robust cleanness. We begin with the definition of our semantic domain, called generalised timed traces [26]. This definition subsumes both discrete-time state sequences and continuous-time trajectories. A generalised timed trace is a function with a discrete or continuous domain (called time domain) and a co-domain which is a metric space. Intuitively, a generalized timed trace maps each element of its time domain to a state. We require that the set of possible states is a metric space since we study conformance notions that compare traces based on the distance between the states of the traces.

Definition 1. *Let $(\mathcal{Y}, d_{\mathcal{Y}})$ be a metric space. A $\mathcal{Y}$-valued* generalised timed trace (GTT) *is a function $\mu : \mathcal{T} \to \mathcal{Y}$ such that $\mathcal{T} \subseteq \mathbb{R}_{\geq 0}$. We call $\mathcal{T}$ the* time domain *of μ, denoted $dom(\mu)$. $GTT(\mathcal{Y})$ is the set of all $\mathcal{Y}$-valued generalised timed traces.*

For a GTT $\mu : \mathcal{T} \to \mathcal{Y}$ and time $t_0 \in \mathcal{T}$, by $\mu[\ldots t_0]$ we denote the prefix of μ up to t_0, i.e., the restriction $\mu|_{t \in \mathcal{T} : t \leq t_0}$; likewise, by $\mu[t_s \ldots t_e]$, we shall denote the restriction $\mu|_{t \in \mathcal{T} : t_s \leq t \leq t_e}$

A hybrid system is a mapping from generalised (input) traces to sets of generalised (output) timed traces.

Definition 2. *A $\mathcal{Y}$-valued hybrid system is a function $H : GTT(\mathcal{Y}) \to \mathcal{P}(GTT(\mathcal{Y}))$ such that for all $\mu \in GTT(\mathcal{Y})$ and all $\mu' \in H(\mu)$ it holds that $dom(\mu') = dom(\mu)$. We define $\mathcal{H}(\mathcal{Y})$ to be the set of all $\mathcal{Y}$-valued hybrid systems.*

In addition, we distinguish deterministic hybrid systems whose output values range over singleton sets only. In what follows, we identify deterministic hybrid systems with functions of the type $GTT(\mathcal{Y}) \to GTT(\mathcal{Y})$.

For simplicity, we assume that the input and output domain are defined on the same metric spaces. The generalisation to different spaces is straightforward.

Conformance relations. Recently, a number of notions of conformance for cyber-physical systems have been proposed [3, 33]. It turns out that these notions, two of which are quoted below, can provide a rigorous basis for doping detection.

Note that throughout the paper, the variables τ and ϵ (with possible subscripts) always range over non-negative real numbers.

Definition 3. *We say that $\mathcal{Y}$-valued GTTs $\mu_1 : \mathcal{T}_1 \to \mathcal{Y}$ and $\mu_2 : \mathcal{T}_2 \to \mathcal{Y}$ are:*

- trace conformant *with tolerance threshold for signal value ϵ, notation $\mathsf{TraceConf}_\epsilon(\mu_1, \mu_2)$, if $\mathcal{T}_1 = \mathcal{T}_2$ and for all $t \in \mathcal{T}_1$, $d_\mathcal{Y}(\mu_1(t), \mu_2(t)) \leq \epsilon$*
- hybrid conformant *with thresholds τ and ϵ, denoted $\mathsf{HybridConf}_{\tau,\epsilon}(\mu_1, \mu_2)$, if:*
  - $\forall t_1 \in \mathcal{T}_1\, \exists t_2 \in \mathcal{T}_2 : |t_2 - t_1| \leq \tau \wedge d_\mathcal{Y}(\mu_2(t_2), \mu_1(t_1)) \leq \epsilon$
  - $\forall t_2 \in \mathcal{T}_2\, \exists t_1 \in \mathcal{T}_1 : |t_1 - t_2| \leq \tau \wedge d_\mathcal{Y}(\mu_1(t_1), \mu_2(t_2)) \leq \epsilon$
- Skorokhod conformant *with tolerance thresholds τ and ϵ, notation $\mathsf{SkorConf}_{\tau,\epsilon}(\mu_1, \mu_2)$, if $\mathcal{T}_1$ and $\mathcal{T}_2$ are intervals and there is a strictly increasing continuous bijection $r : \mathcal{T}_1 \to \mathcal{T}_2$ called retiming, such that:*
  - *for all $t \in \mathcal{T}_1$, $|r(t) - t| \leq \tau$, and*
  - *for all $t \in \mathcal{T}_1$, $d_\mathcal{Y}(\mu_1(t), \mu_2(r(t))) \leq \epsilon$.*

We show in the proposition below and also in our generalisation results in Section 4, that these notions are closely related. However, they also have some fundamental differences, that can be illustrated using the example in Fig. 1.

Example 2. Consider again the example shown in Fig. 1. We can see that in Fig. 1.(a) i_{st} and i_{dev} are trace conformant with value threshold ϵ, as they only exhibit point-wise deviations by values less than ϵ. In contrast, i_{st} and i_{ddev} in Fig. 1.(b) are not trace conformant, yet they are hybrid conformant with time and value margins τ and ϵ, respectively. The key difference is that the inputs depicted in Fig. 1.(b) are very different if compared point-wise, but if one allows for retiming, they are close enough in value after retiming.

The outputs $o'(i_{st})$ and $o'(i_{ddev})$ in Fig. 1.(d) illustrate the fundamental difference between hybrid and Skorokhod conformance: although the order of rising and falling signals are reversed in the two trajectories, they are still hybrid conformant, because hybrid conformance disregards the order. However, Skorokhod conformance requires an order-preserving retiming, and hence distinguishes these two trajectories. On the other hand, such retiming exists, e.g., for i_{st} and i_{ddev} in Fig. 1.(b), witnessing their Skorokhod conformance.

We shall use the following notation. We write $\mathsf{Conf}_1 \sqsubseteq \mathsf{Conf}_2$ whenever for all $\mu_1 : \mathcal{T}_1 \to \mathcal{Y}$ and $\mu_2 : \mathcal{T}_2 \to \mathcal{Y}$, we have $\mathsf{Conf}_1(\mu_1, \mu_2) \implies \mathsf{Conf}_2(\mu_1, \mu_2)$. We write $\mathsf{Conf}_1 \sqsubset \mathsf{Conf}_2$ whenever $\mathsf{Conf}_1 \sqsubseteq \mathsf{Conf}_2$ and $\neg\, \mathsf{Conf}_2 \sqsubseteq \mathsf{Conf}_1$.

Proposition 1. *For any $\tau, \epsilon \in \mathbb{R}_{\geq 0}$, the following relations hold:*

$$\mathit{TraceConf}_{\epsilon} \sqsubset \mathit{SkorConf}_{\tau,\epsilon} \sqsubset \mathit{HybridConf}_{\tau,\epsilon}$$

Robust cleanness. We shall now state the original definition of robust cleanness from [15], adapted to our framework of hybrid systems. It is based on Definition 7 and Proposition 19 from [15]; the phrasing below abstracts from the so-called parameters of interest and standard inputs. Moreover it is cast in the setting of generalised timed traces rather than discrete-step programs, and stated using trace conformance with different thresholds for inputs and outputs, κ_I and κ_O.

Intuitively, a hybrid system is robustly clean if for every pair of input prefixes on which no difference in the inputs exceeding κ_I has occurred so far (i.e., all sub-prefixes are trace conformant), the corresponding sets of output prefixes are also conformant with respect to κ_O. As we consider nondeterministic systems, Hausdorff distance is used to compare sets of outputs (see [15] for details).

Definition 4. *A hybrid system H is robustly clean, denoted $\mathit{RobustClean}(\kappa_I, \kappa_O)$, whenever:*
$\forall i_1, i_2 \in GTT(\mathcal{Y}) : \forall t \in dom(i_1) \cup dom(i_2) :$
$\big(\forall t' \leq t :\ \mathit{TraceConf}_{\kappa_I}(i_1[\ldots t'], i_2[\ldots t'])\ \implies$
$\quad\big((\forall o_1 \in H(i_1)\, \exists o_2 \in H(i_2) :\ \mathit{TraceConf}_{\kappa_O}(o_1[\ldots t], o_2[\ldots t])) \wedge$
$\quad\ (\forall o_2 \in H(i_2)\, \exists o_1 \in H(i_1) :\ \mathit{TraceConf}_{\kappa_O}(o_1[\ldots t], o_2[\ldots t]))\big)$

Note that in the above definition we do not require that $dom(i_1) = dom(i_2)$. In practice, robust cleanness is typically applied to pairs of traces that are both defined over $\mathbb{N}$. Here, however, for the sake of generality we impose no such restriction. In particular, when the time domains of two traces are different, for example disjoint, the predicate RobustClean will trivially evaluate to *true*.

Example 3. Consider the traces depicted in Fig. 1. The input prefixes i_{st} and i_{ddev} are given in Fig. 1.(b), and the corresponding pair of outputs is shown in Fig. 1.(c). The trace i_{st} results in output $o(i_{st})$ and i_{ddev} results in $o(i_{ddev})$. Suppose that $\epsilon < |i_{st}(t_0) - i_{ddev}(t_0)|$ and $\epsilon < |o(i_{st})(t_1) - o(i_{ddev})(t_1)|$ at some time t_1. Thus, the left-hand side of the implication in the Def. 4 instantiated with $\kappa_I = \kappa_O = \epsilon$ does not hold for any t'. Hence, regardless of the outputs, this pair of inputs satisfies the condition of $\mathsf{RobustClean}(\epsilon, \epsilon)$, and, if these are the only traces in a hybrid system H then we can conclude that H is $\mathsf{RobustClean}(\epsilon, \epsilon)$.

4 Conformance-Based Cleanness

We now define a general notion of conformance-based cleanness and provide two instantiations based on the conformance notions defined in the previous section.

The need for considering disturbance in time as well as in value is motivated by our running example from Fig. 1. One of the challenges in performing doping tests for cyber-physical systems is that in such systems timing is rarely perfectly precise, due to imprecision in measurements, or caused by the interaction with the physical world. As illustrated in Example 1, for instance, when checking for software doping in a car [10], the input to the system is the value of the car's speed over time, which is under the control of a driver, and can thus vary from one execution to the other, even if the driver is trying to execute the same input sequence. Clearly, those variations can be in value, as well as in time.

Example 4. Consider the test setup sketched in Fig. 1. There, i_{st} and i_{ddev}, depicted in Fig. 1.(b) define speed of a car as a function of time. These two input sequences follow a trajectory of values differing by a small margin ϵ (the difference in value allowed by the standard defining the doping tests), but also shifted by a small unit of time τ. Observe further that $|i_{st}(t_0) - i_{ddev}(t_0)| \gg \epsilon$. Thus, without allowing for deviations in time when comparing these input sequences, they will be considered sufficiently different, and as a result their respective exhaust emission outputs will fall out of the comparison when checking for doping according to Def. 4, even if the NO_x emission values in the corresponding outputs $H(i_{st}(t))$ and $H(i_{ddev}(t))$ are vastly different, as depicted in Fig. 1.(c). This results in a false negative, i.e., failing to detect a clearly doped system.

In the above example, we demonstrated that not accounting for timing disturbances when relating input trajectories can result in false negatives in doping detection. Dually, using the traditional comparison for output traces can result in false positives by requiring overly strict matching of outputs.

The above example motivates the need to account for timing deviations in trajectories. Intuitively, for input trajectories this relaxation results in considering more traces as conforming, and thus enforcing more comparisons when checking if a system is clean. For output trajectories this means relaxing the conformance requirement by considering two output sequences as conforming even if their values are not perfectly aligned in time. Furthermore, different types of timing deviations need to be considered in different scenarios, for example, depending on whether the order in which values occur is important or not.

Example 5. Consider the testing workflow from Example 1 and Fig. 1, where inputs i_{st} and i_{ddev} are passed to a car. In the second experiment, depicted in Fig. 1.(d), the car outputs $o'(i_{st})$ and $o'(i_{ddev})$, which are hybrid conformant for ϵ and τ. Hence this observation of the system is classified as clean under hybrid output conformance. However, the output $o'(i_{ddev})$ is clearly suspicious, as the values in $o'(i_{ddev})$ and $o'(i_{st})$ are reversed. This motivates considering conformance notions that require retimings to be order-preserving. Indeed, using Skorokhod conformance we can detect that the system is doped.

The above examples show that in order to be useful in a diverse set of applications, a software cleanness theory should allow for using a variety of conformance

notions. To this end, we next take a more general view on conformance notions, in order to be able to develop a generic conformance-based cleanness framework.

So far, we have defined three specific notions of conformance which either coincide, or are closely inspired by ones that have appeared in the literature. In order to define a general framework for cleanness, we also wish to treat notions of conformance in a more generic manner. To this end, we propose an abstract definition of conformance predicates. As conformance predicates admit variations in time, as well as in value, our definition is based on *retimings*, a device that will play a key role in the context of this work. In its general form a retiming is a pair of functions between two time domains. Intuitively, given two GTTs, a retiming will define a mapping from points in each of the traces to points in the other trace. Note that in general the mappings are not required to be injective; this way we can cater for notions of conformance allowing for the so-called local disorder phenomenon (in particular hybrid conformance – see Proposition 2).

Definition 5. *A* retiming *is a pair of functions between two time domains, i.e., a pair of the form* (r_1, r_2)*, where* $r_1 : \mathcal{T}_1 \to \mathcal{T}_2$ *and* $r_2 : \mathcal{T}_2 \to \mathcal{T}_1$*, with time domains* $\mathcal{T}_1, \mathcal{T}_2 \subseteq \mathbb{R}_{\geq 0}$*.Given two time domains* $\mathcal{T}_1$ *and* $\mathcal{T}_2$*, we denote the set of all retimings between* $\mathcal{T}_1$ *and* $\mathcal{T}_2$ *with* $\mathcal{RET}(\mathcal{T}_1, \mathcal{T}_2)$*.*

Retiming is explicitly present in the definition of Skorokhod conformance; there, each Skorokhod retiming is required to be a strictly increasing continuous bijection. We can express a Skorokhod retiming r as an instance of our definition as the pair (r, r^{-1}). In fact, one can also define hybrid conformance, as well as a whole class of conformance notions, using a suitable *family* of retimings.

A family of retimings Ret can be further constrained by τ to a subset Ret_τ of Ret containing only functions that shift time by at most τ time units. In order to use a family of retimings for concrete sequences μ_1 and μ_2, it is necessary to consider only functions that match the domains of the sequences. This leads to a generic notion of conformance associated with a given family of retimings Ret, a given time threshold τ and a given value threshold ϵ.

Definition 6. *Let* Ret *be a family of retimings, and let*

$$\begin{aligned}\mathsf{Ret}_\tau &\triangleq \{(r_1, r_2) \in \mathsf{Ret} \mid \forall t \in dom(r_i) : \ |r_i(t) - t| \leq \tau \ \ (i = 1, 2)\},\\ \mathsf{Ret}_\tau(\mathcal{T}_1, \mathcal{T}_2) &\triangleq \mathsf{Ret}_\tau \cap \mathcal{RET}(\mathcal{T}_1, \mathcal{T}_2).\end{aligned}$$

A conformance notion *with time threshold* τ *and value threshold* ϵ *induced by* Ret *is a predicate* $\mathsf{Conf}^{\mathsf{Ret}}_{\tau,\epsilon}$ *on pairs of GTTs such that, for* $\mu_1 : \mathcal{T}_1 \to \mathcal{Y}$*,* $\mu_2 : \mathcal{T}_2 \to \mathcal{Y}$*:*

$$\begin{aligned}\mathsf{Conf}^{\mathsf{Ret}}_{\tau,\epsilon}(\mu_1, \mu_2) \iff \exists (r_1, r_2) \in \mathsf{Ret}_\tau(\mathcal{T}_1, \mathcal{T}_2) : \forall t \in \mathcal{T}_1 : \ d_{\mathcal{Y}}(\mu_1(t), \mu_2 \circ r_1(t)) \leq \epsilon\\ \wedge \forall t \in \mathcal{T}_2 : \ d_{\mathcal{Y}}(\mu_2(t), \mu_1 \circ r_2(t)) \leq \epsilon.\end{aligned}$$

Using the above definition, we can easily express the specific notions of conformance defined in the previous section by selecting a suitable family of retimings.

Proposition 2. *The conformance predicates below coincide with the notions of conformance induced by the corresponding families of retimings:*

– *TraceConf*$_\epsilon$ *is induced by the family of retimings containing only identity functions:* $\mathsf{Ret}_{\mathsf{id}} = \{(\mathsf{id}, \mathsf{id}) \mid \mathsf{id} : \mathcal{T} \to \mathcal{T}$ *is the identity on some* $\mathcal{T} \subseteq \mathbb{R}_{\geq 0}\}$.
– *SkorConf*$_{\tau,\epsilon}$ *is induced by the family of retimings* $\mathsf{Ret} = \{(r, r^{-1}) \mid r$ *is a strictly increasing continuous bijection*$\}$.
– *HybridConf*$_{\tau,\epsilon}$ *is induced by pairs of arbitrary functions.*

Definition 6 also enables us to define other notions of conformance, such as, for instance a "shift conformance", which, intuitively, shifts all time points by a given constant $c \in \mathbb{R}$, i.e., $\mathsf{Ret}_c = \{(r, r^{-1}) \mid r(t) = t + c\}$.

Next, we define a generic notion of cleanness, parametrised by conformance predicates for the input and for the output traces. Instantiating these predicates with existing or new conformance notions, yields different conformance-based notions of cleanness that can capture a variety of cleanness specifications.

We now extend the notion of robust cleanness [15] to allow for "small" variations in time, in addition to the variations in value. To this end, the new notion makes use of two conformance predicates, one that postulates when two input traces should be considered close enough, and another one that specifies when two output traces are close enough.

Our starting point, the notion of robust cleanness in Definition 4, is based on comparison of matching prefixes of a pair of input traces and the corresponding prefixes of the associated output traces. As we now want to accommodate for distance in time, we (1) compare prefixes using a conformance relation, and (2) allow for variation in the length of the compared prefixes that is within the corresponding time-distance threshold. More precisely, when comparing two prefixes, we allow for discarding start and end segments of length at most τ.

This intuition is formalized by the predicate $\mathsf{PrefConf}$ for relaxed comparison of GTT prefixes using a notion of conformance Conf with tolerance threshold τ for time disturbance. We use cascaded notation to define $\mathsf{PrefConf}$ as a higher-order function taking Conf as its first argument. The predicate $\mathsf{PrefConf}$ compares two prefixes μ_1 and μ_2 by requiring that there exist traces $\mu_1[t_1^s \ldots t_1^e]$ and $\mu_2[t_2^s \ldots t_2^e]$ obtained from them, that are conformant with respect to Conf. These traces are obtained by possibly removing a sub-prefix of length at most τ, and/or removing extending with a suffix of length at most τ.

Definition 7. *Let* Conf *be a notion of conformance on GTTs with tolerance threshold* τ *for time disturbance. For any pair of GTTs* $\mu_1 : \mathcal{T}_1 \to \mathcal{Y}$, $\mu_2 : \mathcal{T}_2 \to \mathcal{Y}$, *and* $t \in \mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2$, *the predicate* $\mathsf{PrefConf}$ *is defined as:*

$$\begin{aligned}\mathsf{PrefConf}(\mu_1, \mu_2, t) \Longleftrightarrow\ & \exists t_1^s \in [0, \tau] \cap \mathcal{T}_1, \exists t_1^e \in [t - \tau, t + \tau] \cap \mathcal{T}_1, \\ & \exists t_2^s \in [0, \tau] \cap \mathcal{T}_2, \exists t_2^e \in [t - \tau, t + \tau] \cap \mathcal{T}_2: \\ & \mathsf{Conf}(\mu_1[t_1^s \ldots t_1^e], \mu_2[t_2^s \ldots t_2^e]).\end{aligned}$$

The predicate $\mathsf{PrefConf}$ provides a generic notion of prefix-conformance. By instantiating it with conformance relations Conf_I and Conf_O for input and output traces respectively, we define the notion of $(\mathsf{Conf}_I, \mathsf{Conf}_O)$-cleanness.

For deterministic systems $(\mathsf{Conf}_I, \mathsf{Conf}_O)$-cleanness requires that for all pairs of input prefixes for which all sub-prefixes are prefix-conformant w.r.t. Conf_I, the corresponding pair of output prefixes are prefix-conformant w.r.t. Conf_O.

Definition 8. *A deterministic system H is $(\mathsf{Conf}_I, \mathsf{Conf}_O)$-clean if*

$$\forall i_1, i_2 \in GTT(\mathcal{Y}) : \forall t \in dom(i_1) \cup dom(i_2) : \\ \big(\forall t' \leq t : \mathsf{PrefConf}_I(i_1, i_2, t')\big) \implies \mathsf{PrefConf}_O(H(i_1), H(i_2), t).$$

The above definition naturally generalises to nondeterministic hybrid systems, by comparing sets of possible output prefixes using Hausdorff distance as in [15].

Definition 9. *A system H is $(\mathsf{Conf}_I, \mathsf{Conf}_O)$-clean if*

$$\forall i_1, i_2 \in GTT(\mathcal{Y}) : \forall t \in dom(i_1) \cup dom(i_2) : \\ \big(\forall t' \leq t : \mathsf{PrefConf}_I(i_1, i_2, t')\big) \implies \\ \big((\forall o_1 \in H(i_1)\, \exists o_2 \in H(i_2) : \mathsf{PrefConf}_O(o_1, o_2, t)) \wedge \\ (\forall o_2 \in H(i_2)\, \exists o_1 \in H(i_1) : \mathsf{PrefConf}_O(o_1, o_2, t))\big).$$

Robust cleanness [15] can be now formulated as conformance-based cleanness, which establishes that $(\mathsf{Conf}_I, \mathsf{Conf}_O)$-cleanness is a generalisation. Using hybrid conformance, we define hybrid-conformance cleanness, and similarly, plugging in Skorokhod conformance, we define Skorokhod-conformance cleanness. Formally:

- A hybrid system H is robustly clean, denoted $\mathsf{RobustClean}(\kappa_I, \kappa_O)$, if and only if H is $(\mathsf{TraceConf}_{\kappa_I}, \mathsf{TraceConf}_{\kappa_O})$-clean.
- A hybrid system H is *hybrid-conformance clean with conformance thresholds* $(\tau_I, \epsilon_I, \tau_O, \epsilon_O)$, which we denote by $\mathsf{HybridClean}(\tau_I, \epsilon_I, \tau_O, \epsilon_O)$, if and only if H is $(\mathsf{HybridConf}_{\tau_I, \epsilon_I}, \mathsf{HybridConf}_{\tau_O, \epsilon_O})$-clean.
- A hybrid system H is *Skorokhod-conformance clean with conformance thresholds* $(\tau_I, \epsilon_I, \tau_O, \epsilon_O)$, denoted $\mathsf{SkorClean}(\tau_I, \epsilon_I, \tau_O, \epsilon_O)$, if and only if H is $(\mathsf{SkorConf}_{\tau_I, \epsilon_I}, \mathsf{SkorConf}_{\tau_O, \epsilon_O})$-clean.

We will now establish some key relations between the cleanness notions defined previously. We begin by lifting the implication between conformance relations to implication between cleanness notions defined using those relations.

Proposition 3. *Suppose that $\mathsf{Conf}_I^1 \sqsupseteq \mathsf{Conf}_I^2$ and $\mathsf{Conf}_O^1 \sqsubseteq \mathsf{Conf}_O^2$. Then for any system H: H is $(\mathsf{Conf}_I^1, \mathsf{Conf}_O^1)$-clean $\implies$ H is $(\mathsf{Conf}_I^2, \mathsf{Conf}_O^2)$-clean.*

The proposition above has two important corollaries. The first one explains the relationships between the original robust cleanness, and notions of cleanness based on Skorokhod conformance and hybrid conformance, in particular stating the conservative generalisation property for the latter notions. The second corollary compares cleanness notions with different conformance thresholds.

Corollary 1. *For all $\tau_I, \tau_O, \epsilon_I, \epsilon_O \in \mathbb{R}_{\geq 0}$, the following implications hold:*

1. $\mathsf{RobustClean}(\epsilon_I, \epsilon_O) \implies \mathsf{SkorClean}(0, \epsilon_I, \tau_O, \epsilon_O) \implies \mathsf{HybridClean}(0, \epsilon_I, \tau_O, \epsilon_O)$,
2. $\mathsf{HybridClean}(\tau_I, \epsilon_I, 0, \epsilon_O) \implies \mathsf{SkorClean}(\tau_I, \epsilon_I, 0, \epsilon_O) \implies \mathsf{RobustClean}(\epsilon_I, \epsilon_O)$.

Also, $\mathsf{RobustClean}(\epsilon_I, \epsilon_O) = \mathsf{SkorClean}(0, \epsilon_I, 0, \epsilon_O) = \mathsf{HybridClean}(0, \epsilon_I, 0, \epsilon_O)$ and hence SkorClean and HybridClean are conservative extensions of robust cleanness.

Corollary 2. *For all* $\epsilon_I, \epsilon'_I, \epsilon_O, \epsilon'_O, \tau_I, \tau'_I, \tau_O, \tau'_O$ *that satisfy the inequalities* $\epsilon'_I \leq \epsilon_I, \quad \tau'_I \leq \tau_I, \quad \epsilon'_O \geq \epsilon_O, \quad \tau'_O \geq \tau_O$ *the following implications hold:*

1. $\mathit{RobustClean}(\epsilon_I, \epsilon_O) \implies \mathit{RobustClean}(\epsilon'_I, \epsilon'_O)$;
2. $\mathit{HybridClean}(\epsilon_I, \tau_I, \epsilon_O, \tau_O) \implies \mathit{HybridClean}(\epsilon'_I, \tau'_I, \epsilon'_O, \tau'_O)$;
3. $\mathit{SkorClean}(\epsilon_I, \tau_I, \epsilon_O, \tau_O) \implies \mathit{SkorClean}(\epsilon'_I, \tau'_I, \epsilon'_O, \tau'_O)$.

Example 6. Consider the testing workflow in Fig. 1. The inputs passed to a car are i_{st} and i_{ddev}, depicted in Fig. 1.(b). One of the test results is presented in Fig. 1.(c), where i_{st} reveals output $o(i_{st})$ and i_{ddev} reveals $o(i_{ddev})$. We assume that $\epsilon < |i_{st}(t_0) - i_{ddev}(t_0)|$ and $\epsilon < |o(i_{st})(t_1) - o(i_{ddev})(t_1)|$ at some time t_1.

- For inputs i_{st} and i_{ddev}, any output is immediately deemed $\mathsf{RobustClean}(\epsilon, \epsilon)$, as the left-hand side of the implication in Def. 8 does not hold for any t'. Note, that for other inputs the car used for testing might not be $\mathsf{RobustClean}(\epsilon, \epsilon)$.
- As explained in Example 2, i_{st} and i_{ddev} are hybrid conformant for ϵ and τ, i.e., the predicate $\mathsf{PrefConf}_I$ on the left-hand side of the implication in Def. 8 holds. $\mathsf{PrefConf}_O$, however, fails at time t_1 for signals $o(i_{st})$ and $o(i_{ddev})$. Hence, the system tested in Fig. 1.(c) is not $\mathsf{HybridClean}(\epsilon, \tau, \epsilon, \tau)$.

We now discuss testing and falsification of conformance-based cleanness.

For systems with discrete time domains the existing methods for verifying [15] or testing [10] robust cleanness can be readily applied.

In the case of hybrid cleanness, existing methods for testing hybrid conformance, such as [2] and [4] can be extended to testing and falsification of hybrid cleanness of hybrid systems consisting of traces with finite time domains. Methods for checking Skorokhod conformance were presented in [17]. Due to the quantification over all time-points t' in our Definition 8 and Definition 9, it is not clear how to directly extend them to testing Skorokhod cleanness.

5 Case Study

In this section we evaluate the proposed notion of hybrid cleanness in the context of doping detection in relation to the recent Diesel Emissions Scandal.

Conducting software doping tests for cyber-physical systems has a range of applications. A prominent example is the body of recent work [10, 32, 34, 9, 15, 14, 8] that gives insights into the Diesel Emissions Scandal. This is a world-wide scandal where millions of diesel cars have been equipped with defeat devices reducing the effectiveness of emission cleaning systems during real-world usage – in contrast to the regulator-defined driving scenarios on a chassis dynamometer, where the amount of emitted pollutants are well below the applicable limits.

Assuming the existence of a contract that formalizes when software is considered to be doped, recent work demonstrates how doping tests can be generated automatically and how the characteristic challenges arising with these kinds of tests can be tackled [10]. A major challenge is the distortion of inputs that can occur during test execution. As doping tests have to be conducted on the final

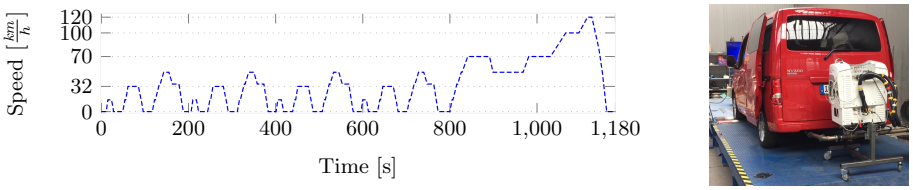

Fig. 2. Left: New European Driving Cycle (NEDC); Right: Test Setup with Nissan NV200 Evalia on a chassis dynamometer attached to a PEMS.

product, i.e., a vehicle such as a passenger car, a human driver has to provide the inputs to the car by driving it. It is far from trivial to provide the inputs exactly as defined by the test case. Official regulations, that define the approval process for new car models, precisely specify test cycles for which they allow tolerances in the input of up to 2 km/h (in car speed). But even driving a car within this tolerance requires a very experienced driver. To strengthen the position of consumers against manufacturers, it is necessary to allow manufacturer-independent methods to check the compliance of a car model with the applicable regulations, i.e., the absence of defeat devices. These methods are supposed to require a reasonable amount of effort, and training a driver over months so that she has enough experience to stay within the tolerance of 2 km/h is way beyond reasonable effort. This means that the responsibility for accounting for the driver's imprecision must be shifted to the techniques for checking for software doping.

In this section we give a short summary of recent doping tests with a diesel car and demonstrate how the theory developed in this paper addresses the above challenge. More precisely, it allows us to overcome the imprecise timing leading to minor input distortions, by appropriately accounting for the effect of retiming on the input value error. We further show how using our theory one of the tests reveals strong indications for a defeat device in the car under test – despite a very inexperienced driver conducting the test. This doping detection would not have been possible using the cleanness notions existing prior to this work.

Physical set-up of the experiment. Before a car model can be sold, it must meet the requirements defined in the official regulations. The type approval procedure requires the car to be placed on a chassis dynamometer. Cars have to follow certain standardized test cycles, each defined as a function from time to speed. One of the test cycles, involved in the diesel scandal, was the New European Driving Cycle (NEDC) [42] shown in Fig. 2. For the tests here, we consider the *speed* of the car as *input*, since this is the parameter defining a test cycle. The *total amount of* NO *and* NO_2 *(abbreviated as* NO_x*)* is the only *output* of interest. The car under test is a Nissan NV200 Evalia, with Renault 1.5 dci (110hp) diesel engine and approved w.r.t. regulation *Euro 6b*. The test set-up is shown in Fig. 2.

In order to perform a check for defeat devices using a cleanness test, we consider, in addition to the original NEDC, two manually synthesized tests. These test cycles, denoted PowerNEDC and SineNEDC were proposed in previous work [10] and are defined as follows. PowerNEDC is based on the

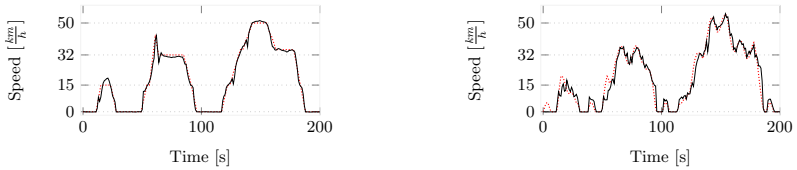

Fig. 3. Initial 200s of PowerNEDC (left) and SineNEDC (right) planned test cycles (red, dotted) and actually driven (black).

NEDC but slightly deviates from it by enforcing higher accelerations ($1.5\frac{m}{s^2}$ instead of $0.94\frac{m}{s^2}$) after 56s, 251s, 446s and 641s. The *maximum input deviation from* NEDC *is* $\kappa_I = 10km/h$. SineNEDC is defined as the NEDC superimposed by a sine curve, formally $\text{SineNEDC}(t) = \max\{0, \text{NEDC}(t) + 5\sin(0.5t)\}$, with a *maximum input deviation from* NEDC *of* $\kappa_I = 5km/h$.

These test cycles are defined by specifying the input value (the car's speed) in each second. Both test cases are shown by the red dotted lines in Fig. 3.

Conformance-based cleanness tests for NEDC. We have applied our theory of conformance-based cleanness to check for doping, i.e., the presence of a defeat device, in the car under test. For this, we have at our disposal the raw data obtained from three test drives: (1) Test drive dNEDC is the result of NEDC cycle driven by a human driver. It serves as the reference behaviour of the car, to which we will compare the executions of the other two test cycles. (2) Test drive dPowerNEDC is the trajectory that is produced as the result of a human driver driving PowerNEDC. (3) Test drive dSineNEDC is the trajectory that is produced as the result of a human driver driving SineNEDC.

The values of the actual sequences of inputs executed by driving the car are sampled in steps of 0.05s. As mentioned earlier, the human in the loop makes testing considerably more challenging. The maximum deviation of inputs compared to the test specification for NEDC is just below 10 km/h, for PowerNEDC is almost 12 km/h, and for SineNEDC it approaches 16 km/h. This shows that the perturbation introduced by the human driver is clearly noticeable. The amount of NO_x measured for dNEDC is $180mg/km$, for dPowerNEDC and dSineNEDC the measurements revealed $204mg/km$ and $584mg/km$, respectively.

In order to detect doping (by falsifying cleanness), the input sequences of dPowerNEDC and dSineNEDC have to be each compared to dNEDC, and if the input sequences in the corresponding pair are conforming, then the respective outputs (the total NO_x emission values) have to be checked for conformance.

As we desire for our doping tests to be as strict as possible, we identify hybrid conformance $\mathsf{HybridConf}_{\tau_I,\epsilon_I}$, i.e., the weakest of the conformance relations discussed in Section 3, as the most suitable conformance relation for the comparison of input traces. As the outputs are just single values, the choice of output conformance relation is immaterial in this case, so we take $\mathsf{HybridConf}_{0,\epsilon_O}$.

Formally, we consider the deterministic hybrid system H defined by the input GTTs dNEDC, dPowerNEDC, and dSineNEDC, and check whether H is $\mathsf{HybridClean}(\tau_I, \epsilon_I, 0, \epsilon_O)$-clean for given values of τ_I, ϵ_I and ϵ_O.

The driver's imprecision has a significant effect on the values in the input sequences and their timing. This can lead to dismissing pairs of sequences if they are incorrectly deemed too far apart, and thus missing some indications of doping. For instance, a too strict comparison of dSineNEDC to dNEDC will dismiss this pair of executions; however, the measured NO_x emission during the dSineNEDC drive is *three times more* than the one measured during dNEDC.

Testing $\mathsf{HybridClean}(\tau_I, \epsilon_I, 0, \epsilon_O)$ allows us to perform a realistic comparison by taking into account the two possible sources of driving errors: the over- or undershooting of the speed, and the timing offsets, where the driver accelerates or decelerates too fast or too slowly. In comparison, prior doping tests based on Robust Cleanness, considered only the former, i.e., the point-wise offset in speed. As we demonstrate, depending on the specified value threshold, there are cases when this is insufficient to identify doping. Indeed, looking into the official regulations, we can see that they allow for a timing variation of one second [42, 19]. Thus, essentially, the regulations allow for hybrid conformance with $\tau_I = 1$s.

Hybrid cleanness testing. In order to test $\mathsf{HybridClean}(\tau_I, \epsilon_I, 0, \epsilon_O)$ we have to examine the conformance relations $\mathsf{HybridConf}_{\tau_I,\epsilon_I}(\text{dNEDC}, \text{dPowerNEDC})$ and $\mathsf{HybridConf}_{\tau_I,\epsilon_I}(\text{dNEDC}, \text{dSineNEDC})$ between the corresponding input sequences. Recall that since the output of the system measured in each test is the total amount of NO_x emitted during the test, i.e., a single value for the whole execution, timing plays no role when quantifying the value error for the output.

In order to evaluate the power of using hybrid cleanness for detecting doping, we consider different values for ϵ_I and τ_I, and perform two types of analysis of the results of testing $\mathsf{HybridClean}(\tau_I, \epsilon_I, 0, \epsilon_O)$, which we describe below.

Effect of τ_I on the minimal ϵ_I for which inputs are conforming. First, we fix a maximum value that we allow for the time offset τ_I. For this τ_I we analyse our dataset to find the minimal ϵ_I such that for the combination τ_I and ϵ_I the input traces under consideration satisfy hybrid conformance. For $\tau_I = 0$ we get exactly the ϵ_I for which the two traces are trace conformant. Table 1 (left side) shows the computed ϵ_I values for $\tau_I = 0, 0.5, 1, 2, 5, 10$.

As expected, when we increase τ_I, the minimal ϵ_I decreases. At some point (at $\tau_I = 2$ for PowerNEDC and $\tau_I = 5$ for SineNEDC) the decrease in the value error reduces notably. This happens because the error is only partially caused by the incorrect timing of the driver.

From the values reported in Table 1 (left) we see that if, for example, we allow deviation for the input $\tau_I = 1$, as per the official regulation, and set $\epsilon_I = 15$, then we have that both $\mathsf{HybridConf}_{\tau_I,\epsilon_I}(\text{dNEDC}, \text{dPowerNEDC})$ and $\mathsf{HybridConf}_{\tau_I,\epsilon_I}(\text{dNEDC}, \text{dSineNEDC})$ are true, while, for $\tau_I = 0$ both are false. Thus, under hybrid conformance these pairs of traces will be considered in the cleanness test, while under trace conformance they will be dismissed.

Since the difference between the outputs measured during dSineNEDC and during dNEDC is vast, we establish that $\mathsf{HybridClean}(1, 15, 0, 180)$ does *not* hold.

Effect of ϵ_I on the minimal τ_I for which inputs are conforming. Second, we fix the maximum value error ϵ_I and examine what minimal τ_I results in a combination τ_I and ϵ_I for which the analysed data is hybrid conformant. For

	$\tau_I = 0$	$\tau_I = 0.5$	$\tau_I = 1$	$\tau_I = 2$	$\tau_I = 5$	$\tau_I = 10$	$\epsilon_I = \kappa_I$	$\epsilon_I = \kappa_I + 2$
PowerNEDC	$\epsilon_I = 15.88$	$\epsilon_I = 15.03$	$\epsilon_I = 12.41$	$\epsilon_I = 10.10$	$\epsilon_I = 10.07$	$\epsilon_I = 10.07$	$\tau_I = 67.35$	$\tau_I = 10.8$
SineNEDC	$\epsilon_I = 16.17$	$\epsilon_I = 15.46$	$\epsilon_I = 14.24$	$\epsilon_I = 12.91$	$\epsilon_I = 11.67$	$\epsilon_I = 11.37$	$\tau_I = 72.4$	$\tau_I = 4.05$

Table 1. Value thresholds for fixed τ_I (left) and time thresholds for fixed ϵ_I (right). Values are given as mg/km and time in seconds.

the synthesized test cases we study the error tolerance ϵ_I set to the respective input thresholds κ_I. As discussed above, this is 10km/h for PowerNEDC and 5km/h for SineNEDC. We also consider the scenario where the error tolerance allowed by the official regulation for the test cycle is added, that is, we also consider $\epsilon_I = \kappa_I + 2$km/h. The two rightmost columns of Table 1 show the necessary time shifts to achieve these value errors. As apparent, they reduce by approximately 84% and 94% when adding the error tolerance of 2 km/h.

These values for τ_I give us the minimal tolerance threshold for time, for which $\mathsf{HybridClean}(\tau_I, \epsilon_I, 0, 180)$ is violated in H for the given ϵ_I; the value of ϵ_O is fixed at 180mg/km according to the standard [10].

Evaluation and discussion. The analysis of the data shows that it is indeed necessary to not only consider a deviation of value, but to also allow for timing deviations, especially when the quality of the studied driving tests suffers from the human-caused input distortions. In terms of the theory established in this paper, this means that in scenarios like this one, employing HybridClean is more adequate than using prior notions such as RobustClean, and without this, the cases of doping we have detected would go unnoticed. Allowing a retiming of up to 10.8s (for PowerNEDC) and of 4.05s (for SineNEDC) makes both inputs conformant to the NEDC input, so we are able to detect the violation of SineNEDC for the hybrid cleanness for the specified desired value error tolerance. While these time deviations appear large given the test cycle timeline, they are acceptable when we recall that the tests are executed by human drivers.

If, on the other hand, we want to restrict the tolerance in time to one second, we are able to consider both tests for the hybrid cleanness for value error tolerance of 12.41km/h for PowerNEDC and 14.24km/h for SineNEDC.

This demonstrates how conformance-based cleanness notions like HybridClean allow us to some extent to account for human-caused errors related to timing.

Finally, while hybrid cleanness is arguably the appropriate notion for the case study considered here, our generic theory of conformance-based cleanness allows for using other conformance notions as appropriate for the CPS under test.

6 Conclusions

In this paper, we presented a theory of doping detection and cleanness based on the notions of conformance for cyber-physical systems. Our new notion accounts for possible "deviations" of the system output, upon "perturbing" its inputs, both in time and in values. Both notions of "deviation" and "perturbation" turn out to be expressible using a generic notion of retiming. We instantiate

our definition with specific notions of retiming from the conformance testing literature. We apply our notions to a case study from the automotive domain and demonstrate how our generalised notions are useful in using actual driving cycles for doping detection according to the New European Driving Cycle (NEDC) [42].

We intend to turn our theory into an automatic tool for doping detection, using hybrid systems models. We intend to use the HyConf tool [4] as the starting point and use our search-based testing implementation in HyConf [5] to automate the process of test-case generation and test-case selection. Once this process is automated, one can generate test-cases that can go beyond a specific standard and detect intelligent defeat devices that cheat the standards and the tests prescribed by them.

We also intend to organise widespread experiments regarding emission detection to put our theory into practice. Our experimental set-up involves instrumenting a large number of cars using low-cost equipments, constructing models of emission behaviour, and generating realistic driving scenarios that are more likely to detect doping.

References

1. Abbas, H., Mittelmann, H.D., Fainekos, G.E.: Formal property verification in a conformance testing framework. MEMOCODE 2014, pp. 155–164. IEEE (2014)
2. H. Abbas, B. Hoxha, G. Fainekos, J. V. Deshmukh, J. Kapinski and K. Ueda: WiP abstract: Conformance testing as falsification for cyber-physical systems, 2014 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS), Berlin, 2014, pp. 211-211
3. Aerts, A., Mousavi, M.R., Reniers, M.A.: Model-based testing of cyber-physical systems. In Cyber-Physical Systems: Foundations, Principles and Applications, chap. 19. Elsevier (2017)
4. Araujo, H., Carvalho, G., Mohaqeqi, M, Mousavi, M.R., Sampaio, A.: Sound conformance testing for cyber-physical systems: Theory and implementation. Sci. Comput. Program 2018; 162:35-54
5. Araujo, H., Carvalho, G., Mousavi, M.R., Sampaio, A.: Multi-objective Search for Effective Testing of Cyber-Physical Systems. SEFM 2019, vol. 11724 of LNCS, pp. 183–202, Springer, 2019.
6. Agrawal, S., Bonakdarpour, B.: Runtime Verification of k-Safety Hyperproperties in HyperLTL. CSF 2016. IEEE Computer Society, pp. 239–252
7. Annpureddy, Y., Liu, C., Fainekos, G.E., Sankaranarayanan, S.: S-taliro: A tool for temporal logic falsification for hybrid systems. TACAS 2011. LNCS, vol. 6605, pp. 254–257. Springer (2011)
8. Barthe, G., D'Argenio, P.R., Finkbeiner, B., Hermanns, H.: Facets of software doping. ISoLA 2016, Part II. LNCS, vol. 9953, pp. 601–608. Springer (2016)
9. Biewer, S., D'Argenio, P.R., Hermanns, H.: Cyber-physical doping tests. In: 3rd Workshop on Monitoring and Testing of Cyber-Physical Systems, MT@CPSWeek 201, pp. 18–19. IEEE (2018)
10. Biewer, S., D'Argenio, P.R., Hermanns, H.: Doping tests for cyber-physical systems. QEST 2019. LNCS, vol. 11785, pp. 313–331, Springer (2019)
11. Brett, B., Siddique, U., Bonakdarpour, B.: Rewriting-Based Runtime Verification for Alternation-Free HyperLTL. TACAS 2017. LNCS, vol. 10206, pp. 77–93. Springer (2017)

12. Clarkson, M.R., Finkbeiner, B., Koleini, M., Micinski, K.K., Rabe, M.N., Sánchez, C.: Temporal logics for hyperproperties. POST 2014. LNCS, vol. 8414, pp. 265–284. Springer (2014)
13. Clarkson, M.R., Schneider, F.B.: Hyperproperties. In: CSF'08. pp. 51–65 (2008)
14. Contag, M., Li, G., Pawlowski, A., Domke, F., Levchenko, K., Holz, T., Savage, S.: How they did it: An analysis of emission defeat devices in modern automobiles. SP 2017, pp. 231–250. IEEE Computer Society (2017)
15. D'Argenio, P.R., Barthe, G., Biewer, S., Finkbeiner, B., Hermanns, H.: Is your software on dope? - formal analysis of surreptitiously "enhanced" programs. ESOP 2017 LNCS vol. 10201, pp. 83–110. Springer (2017)
16. De Nicola, R., Hennessy, M.: Testing equivalences for processes. Theor. Comput. Sci. **34**, 83–133 (1984)
17. Deshmukh, J.V., Majumdar, R., Prabhu, V.S.: Quantifying conformance using the Skorokhod metric. Formal Methods in System Design **50**(2-3), 168–206 (2017)
18. Donzé, A.: Breach, A toolbox for verification and parameter synthesis of hybrid systems. CAV 2010 LNCS, vol. 6174, pp. 167–170. Springer (2010)
19. European Comission: Commission Regulation (EU) 2017/1151 (2017)
20. Fainekos, G.E., Pappas, G.J.: Robustness of temporal logic specifications for continuous-time signals. Theor. Comput. Sci. **410**(42), 4262–4291 (2009)
21. Finkbeiner, B., Hahn, C.: Deciding hyperproperties. In: Desharnais, J., Jagadeesan, R. (eds.) CONCUR 2016. LIPIcs, vol. 59, pp. 13:1–13:14. (2016)
22. Finkbeiner, B., Hahn, C., Stenger, M.: EAHyper: Satisfiability, implication, and equivalence checking of hyperproperties. CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part II. LNCS, vol. 10427, pp. 564–570. Springer (2017)
23. Finkbeiner, B., Hahn, C., Stenger, M., Tentrup, L.: Monitoring hyperproperties. RV 2017. LNCS, vol. 10548, pp. 190–207. Springer (2017)
24. Finkbeiner, B., Hahn, C., Stenger, M., Tentrup, L.: RVHyper: A runtime verification tool for temporal hyperproperties. TACAS 2018. LNCS vol. 10806, pp. 194–200. Springer (2018)
25. Finkbeiner, B., Rabe, M.N., Sánchez, C.: Algorithms for model checking HyperLTL and HyperCTL*.CAV 2015. LNCS, vol. 9206, pp. 30–48. Springer (2015)
26. Gazda, M., Mousavi, M.R.: Logical characterisation of hybrid conformance. To appear in ICALP 2020. (2020)
27. Girard, A., Julius, A.A., Pappas, G.J.: Approximate simulation relations for hybrid systems. Discrete Event Dynamic Systems **18**(2), 163–179 (2008)
28. Girard, A., Pappas, G.J.: Approximate bisimulation: A bridge between computer science and control theory. Eur. J. Control **17**(5-6), 568–578 (2011)
29. Hahn, C., Stenger, M., Tentrup, L.: Constraint-based monitoring of hyperproperties. TACAS 2019 Proceedings, Part II. LNCS, vol. 11428, pp. 115–131. Springer (2019)
30. Hapke, T., Hornung, P., Becker, J.: Schummeln auch in Europa. ARD/Norddeutscher Rundfunk, *https://www.tagesschau.de/wirtschaft/vw-schummelsoftware-101.html* (2015), Online; accessed: 2019-04-19
31. Hennie, F.C.: Fault detecting experiments for sequential circuits. In: 5th Annual Symposium on Switching Circuit Theory and Logical Design, Princeton, New Jersey, USA, November 11-13, 1964. pp. 95–110. IEEE Computer Society (1964)
32. Hermanns, H., Biewer, S., D'Argenio, P.R., Köhl, M.A.: Verification, testing, and runtime monitoring of automotive exhaust emissions.LPAR-22. EPiC Series in Computing, vol. 57, pp. 1–17. EasyChair (2018)

33. Khakpour, N., Mousavi, M.R.: Notions of conformance testing for cyber-physical systems: Overview and roadmap (invited paper). CONCUR 2015 LIPIcs, vol. 42, pp. 18–40. (2015)
34. Köhl, M.A., Hermanns, H., Biewer, S.: Efficient monitoring of real driving emissions. RV 2018. LNCS, vol. 11237, pp. 299–315. Springer (2018)
35. Lee, D., Yannakakis, M.: Principles and methods of testing finite-state machines - a survey. Proceedings of the IEEE **84**(8), 1089–1123 (1996)
36. Maler, O., Nickovic, D.: Monitoring temporal properties of continuous signals. FORMATS 2004 and FTRTFT 2004. LNCS, vol. 3253, pp. 152–166. Springer (2004)
37. Nguyen, L.V., Kapinski, J., Jin, X., Deshmukh, J.V., Johnson, T.T.: Hyperproperties of real-valued signals.MEMOCODE 2017. pp. 104–113. ACM (2017)
38. van Osch, M.: Hybrid input-output conformance and test generation. FATES 2006 and RV 2006. LNCS, vol. 4262, pp. 70–84. Springer (2006)
39. Pnueli, A.: The temporal logic of programs. In: 18th Annual Symposium on Foundations of Computer Science. pp. 46–57. IEEE Computer Society (1977)
40. Tretmans, J.: A formal Approach to conformance testing. Ph.D. thesis, University of Twente, The Netherlands (1992)
41. Tretmans, J.: Conformance testing with labelled transition systems: Implementation relations and test generation. Computer Networks and ISDN Systems **29**(1), 49–79 (1996)
42. United Nations: UN Vehicle Regulations - 1958 Agreement, Revision 2, Addendum 100, Regulation No. 101, Revision 3 — E/ECE/324/Rev.2/Add.100/Rev.3 (2013)