POWVER

Technical Report 2020-01

Title: **Components in Probabilistic Systems: Suitable by Construction**

Authors: Christel Baier, Clemens Dubslaff, Holger Hermanns,
Michaela Klauck, Sascha Klüppelholz, Maximilian A. Köhl

Report Number: 2020-01

ERC Project: Power to the People. Verified.

ERC Project ID: 695614

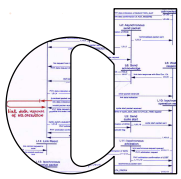
Funded Under: H2020-EU.1.1. – EXCELLENT SCIENCE

Host Institution: Universität des Saarlandes, Dependable Systems and Software
Saarland Informatics Campus

Published In: ISoLA 2020

Please cite this publication as follows:

Christel Baier, Clemens Dubslaff, Holger Hermanns, Michaela Klauck, Sascha Klüppelholz, Maximilian A. Köhl.
Components in Probabilistic Systems: Suitable by Construction.
Leveraging Applications of Formal Methods, Verification and Validation: Verification Principles - 9th International Symposium on Leveraging Applications of Formal Methods, ISoLA 2020, Rhodes, Greece, October 20-30, 2020, Proceedings, Part I. Lecture Notes in Computer Science 12476, Springer 2020, ISBN 978-3-030-61361-7: 240-261.



POWER TO THE PEOPLE.
VERIFIED.
erc

Components in Probabilistic Systems: Suitable by Construction*

Christel Baier[1], Clemens Dubslaff[1], Holger Hermanns[2,3], Michaela Klauck[2], Sascha Klüppelholz[1], and Maximilian A. Köhl[2]

[1] Technische Universität Dresden, Dresden, Germany
[2] Saarland University, Saarland Informatics Campus, Saarbrücken, Germany
[3] Institute of Intelligent Software, Guangzhou, China

Abstract. This paper focusses on the question when and to what extent a particular system component can be considered *suitable* to use in the context of the dynamics of a larger technical system. We introduce different notions of *suitability* that arise naturally in the context of probabilistic nondeterministic systems that interact through the exchange of messages in the style of input-output automata. Besides discussing algorithmic aspects for an analysis following our notions of suitability, we demonstrate practical usability of our concepts by means of experiments on a concrete use case.

1 Introduction

The structured composition of systems from smaller entities is a key technique across many engineering disciplines. For instance, in the field of architecture, it is well understood how the structural properties of construction stones translate into structural properties of walls and thus of houses. This concept also is extremely appealing for the engineering of *cyber-physical systems (CPSs)*, typically built up of components that interact and exchange information [30,3]. For CPSs, the compositional approach poses a number of challenges, stemming first and foremost from the notoriously complex dynamics of even simple CPSs placed in only partially controllable or partially known environments. But also the semantic heterogeneity of computational, physical, and human aspects for modelling the CPS, together with algorithmic and technical challenges in a model-based engineering process render the modelling and analysis of composite CPSs an exigent task.

The present paper contributes to the quest for methods and tools to construct, abstract, compose, and evaluate CPS models that summarise the crucial

* Authors are listed in alphabetical order. This work was partially supported by the DFG under the projects TRR 248 (see `https://perspicuous-computing.science`, project ID 389792660), EXC 2050/1 (CeTI, project ID 390696704, as part of Germany's Excellence Strategy), BA-1679/11-1, and BA-1679/12-1, the ERC Advanced Investigators Grant 695614 (POWVER), and the Key-Area Research and Development Program Grant 2018B010107004 of Guangdong Province.

aspects of components' quantitative behaviour, together with support for design-time evaluation of alternatives. Our long-term vision is a methodology to devise, verify, and compose *summaries of component characteristics*, and to provide means that enable the comparative analysis of such characteristics. To this end, we aim at deepening the known concepts of interfaces and service contracts in that they come with rigid semantic interpretations, and are supported by effective algorithmic analysis techniques. In doing so, we focus on component models that can exhibit probabilistic behaviour while engaging in interaction via inputs and outputs. The central notion that we study in this paper is *suitability*. We explore the spectrum of meaningful notions of suitability of a component with respect to a set of quantitative properties representing what is considered important in a specific context. In this, we concentrate on probabilistic aspects of suitability.

Concurrency, Composition, and Probability. The questions in how far component characteristics affect a larger context is entrenched with the question how the components interact, i.e., what the composition of components and contexts actually mean semantically. Process calculi like CSP [26] or CCS [34] are at the roots of generic and expressive ways to piece up larger systems from concurrent interacting components. Segala [38] lifted these ideas to the setting of probabilistic automata, nowadays the standard composition for Markov decision processes (MDPs) [37] also used in analysis tools such as Prism [29]. Earlier seminal work on probabilistic concurrency [24] has put in focus the importance of a generative/reactive view on probabilities. This echoes the separation of component activities into inputs and outputs, a central concept especially in the works on I/O automata [32]. In this modelling approach, component inputs are always enabled, meaning that no component can block the output of another component by not accepting it as input. This simple assumption is natural in many contexts: if in place, it is intuitively easy to add more components to an existing system, since none of them will block the behaviour already present. In a probabilistic setting with inputs and outputs, it is furthermore natural to associate to outputs a generative probabilistic effect: different outputs of a component can be generated according to a probability distribution (local to the output component), while inputs are reactive in the sense that for all inputs the component is able to react with a probabilistic effect. This idea was first worked out in probabilistic I/O automata [40], and later adapted to the setting of probabilistic automata [14,23,11].

Probabilistic Input-Output Systems (PIOSs). In this paper, we strive towards notions for the suitability of components to be composed with a larger context. To benefit from the compositional advantages detailed above, we work with a very expressive formalism for interacting probabilistic components and their composition based on the compositional framework of *interleaved probabilistic I/O system (IPIOA)* [23]. This formalism is a conservative extension of input-output automata [32] to the setting of discrete probabilities. We further enhance IPIOA slightly by a more flexible concept of observability, leading to the framework

of *partially observable PIOSs (PO-PIOSs)*. While the use of I/O formalisms as in PO-PIOSs is common for many compositional specification theories, input-enabledness is sometimes not natural for tightly interacting systems. However, our concept of suitability does not explicitly rely on the input-enabledness assumption in PO-PIOSs and can be adapted to other compositional MDP-based formalisms.

Notions of Suitability. Stepwise and with an increasing intricacy, we introduce several notions of suitability formalised for the setting of PO-PIOS. Our basic instance is provided through *threshold suitability* that determines whether each one of the given quantitative properties exceeds a given threshold. This notion has similarities to conjunctive multi-objective properties in MDPs [12,19,20,21]. Weighting quantitative properties for the CPS leads to a single quantitative measure of *suitability degree*, which then might be used to relate different components with respect to their suitability. That is, we call a component *more suitable* than another if executed in the same context CPS all possible executions achieve a higher suitability degree. For all of our notions, we present *universal* and *existential* versions, differing in the ability of the component investigated with respect to its ability to react on the other components of the CPS.

Suitability Evaluation. Algorithmically, the notions of suitability we introduce for PO-PIOS are closely related to threshold properties for IPIOA [23], and to verification problems on partially-observable MDPs [35,31,33,8], all of which are known to be undecidable already under mild assumptions. As we illustrate in this paper, this leaves little room for decidable suitability problems in the general case. Therefore, restricted classes of PO-PIOSs, properties, and schedulers have to be considered to establish decidable instances of our suitability problems. The problem instances for which we establish positive results comprise PO-PIOSs with full observability and restrictions on the nondeterminism that is present in the components. While these instances appear to be quite restricted at the first glance, our case study shows that they provide useful contributions to estimate suitability of components in CPSs.

Suitability in Action. Despite our definitions of suitability being a priori developed in a theoretical context, and despite the challenges in algorithmically capturing the concepts, we put them to a first practical litmus test. For this, we instantiated them in a concrete example context, known as the *Racetrack* case study across the automated planning community [9,10,36], here augmented with probabilistic noise [25]. Within this case study, a car that comprises multiple components such as an engine, tank, and a track with different types of ground, aims to reach a target position while meeting time, energy, and CO_2-emission constraints. We work on a feature-oriented model of the car where the model family consists of multiple car and environment configurations, e.g. differing in the engine variant, the tank size and the ground the car drives on. Specifically, we show that a more powerful engine is *existentially threshold suitable* on tarmac

but not on sand and that a less powerful engine is *more suitable* in terms of its *suitability degree* than a more powerful engine.

Contributions. In a nutshell, this paper (i) develops a spectrum of suitability notions for probabilistic components with inputs and outputs, (ii) provides results regarding decidability for the notions considered, and (iii) illustrates the notions and their effect in the context of a case study with vehicle components.

2 Partially Observable Probabilistic I/O Systems

This section discusses the basic concepts of the compositional framework of *probabilistic I/O automata* originally proposed by Giro et al. [23], enhanced with a notion of partial observability.

Markov Decision Processes. For a finite set S, we denote by $\mathit{Dist}(S)$ the set of all the probability distributions over the set S, i.e. functions $\mu\colon S \to [0,1]$ such that $\sum_{s\in S} \mu(s) = 1$. We write $\delta(s)$ for the *Dirac distribution* where $\delta(s)(s) = 1$.

Definition 1 (Markov Decision Process (MDP)). *A Markov Decision Process (MDP) is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, s_0)$ where $\mathcal{S}$ and $\mathcal{A}$ are sets of states and actions, respectively, $\mathcal{T} \subseteq \mathcal{S} \times \mathcal{A} \times \mathit{Dist}(\mathcal{S})$ is a transition probability relation, and $s_0 \in \mathcal{S}$ is an initial state.*

Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, s_0)$ be an MDP as above. We say that action $a \in \mathcal{A}$ is *applicable* in state $s \in \mathcal{S}$ if $(s, a, \mu) \in \mathcal{T}$ for some $\mu \in \mathit{Dist}(\mathcal{S})$. By $\mathcal{A}(s) \subseteq \mathcal{A}$ we denote the set of actions applicable in s. We assume w.l.o.g. that $\mathcal{A}(s)$ is nonempty for all $s \in \mathcal{S}$. Furthermore, we require that for all $(s, a, \mu), (s, a, \mu') \in \mathcal{T}$ we have $\mu = \mu'$. A *finite path* in $\mathcal{M}$ is an alternating sequence of states and transitions $\pi = s_0\, t_0\, s_1\, t_1\, \ldots\, t_{k-1}\, s_k$ where $s_1, \ldots, s_k \in \mathcal{S}$ and where for each index $i \in \{0, 1, \ldots, k-1\}$, $t_i = (s_i, a_i, \mu_i) \in \mathcal{T}$ such that $\mu_i(s_{i+1}) > 0$. We denote by $\mathit{Paths}(\mathcal{M})$ the set of all finite paths in $\mathcal{M}$. By $\mathit{last}(\pi)$ we denote the last state of π, i.e. $\mathit{last}(\pi) = s_k$. Infinite paths are defined accordingly, collected in a set $\mathit{IPaths}(\mathcal{M})$. A (randomised) *scheduler* for $\mathcal{M}$ is a function $\mathfrak{S}\colon \mathit{Paths}(\mathcal{M}) \to \mathit{Dist}(\mathcal{A})$ that resolves the nondeterminism in an execution of the MDP $\mathcal{M}$, i.e. for any path $\pi \in \mathit{Paths}(\mathcal{M})$ we have $\mathfrak{S}(\pi) \in \mathit{Dist}\big(\mathcal{A}(\mathit{last}(\pi))\big)$. $\mathfrak{S}$ is called *memoryless* in case for all paths $\pi_1, \pi_2 \in \mathit{Paths}(\mathcal{M})$ with $\mathit{last}(\pi_1) = \mathit{last}(\pi_2)$ we have $\mathfrak{S}(\pi_1) = \mathfrak{S}(\pi_2)$, and *deterministic* if all distributions in $\mathfrak{S}$ are Dirac. We define the probability measure $\Pr^{\mathfrak{S}}_{\mathcal{M}}$ on $\mathcal{M}$ with respect to a scheduler $\mathfrak{S}$ in the standard way, assigning a probability to measurable sets of paths in $\mathcal{M}$. Here, the fact that any scheduler resolves the nondeterminism in the given MDP towards a Markov chain [37] is exploited.

Observability in MDPs. A flexible notion of observation will allow us to map states and actions to *observables*.

Definition 2 (Observation Function). *An* observation function *for an MDP* $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, s_0)$ *over a set of atomic observables Obs is a function* $obs\colon (\mathcal{S} \cup \mathcal{A}) \to (Obs \cup \{\varepsilon\})$*, where* $obs(x)=\varepsilon$ *stands for unobservability of state or action* x*.*

We refer to observation function *obs* as *totally observable* in case $Obs = \mathcal{S} \cup \mathcal{A}$ and $obs(x) = x$ for all $x \in \mathcal{S} \cup \mathcal{A}$. For a transition $t = (s, a, \mu) \in \mathcal{T}$ we denote by $obs(t)$ the observation $obs(a)$ of the action of t. Observation functions *obs* are extended to functions from paths $\pi = s_0\, t_0\, s_1\, t_1\, \ldots\, t_{k-1}\, s_k$ to strings over the alphabet *Obs*, given by

$$obs(\pi) \quad = \quad obs(s_0)\, obs(t_0)\, obs(s_1)\, \ldots\, obs(t_{k-1})\, obs(s_k).$$

For an observation function *obs* as above, a function ρ defined on paths of $\mathcal{M}$ is said to be *obs-complying* if for all finite paths $\pi_1, \pi_2 \in Paths(\mathcal{M})$ we have that

$$obs(\pi_1) = obs(\pi_2) \text{ implies } \rho(\pi_1) = \rho(\pi_2).$$

Probabilistic I/O Systems. To introduce the PIOS framework [23], we first need to define reactive and generative structures for outputs and inputs, respectively: Given a set *Act* of action labels and a set *States* of states, a *generative output transition relation* G is a subset of $States \times Dist(Act \times States)$, and an *input reactive transition function* R is a function of the form $States \times Act \to Dist(States)$. Intuitively, executing a generative output transition $(s, \kappa) \in G$ available in some state s means choosing both an action a to output and a state s' with joint probability $\kappa(a, s)$. In a composed setting, action a will serve as an output broadcasted to other participants. Receiving input a while being in state t triggers a unique reaction $R(t, a)$ according to the input reactive transition function R, mapping to a distribution over successor states.

Definition 3 (Probabilistic Input/Output System (PIOS)). *A probabilistic I/O* component *is a tuple* $(States, Act, G, R, init)$*, where*

- *States is a finite set of states,*
- *Act is a finite set of action labels,*
- $G \subseteq States \times Dist(Act \times States)$ *is a generative output transition relation,*
- $R\colon States \times Act \to Dist(States)$ *is a reactive transition function, and*
- $init \in States$ *is an initial state.*

A probabilistic I/O system (PIOS) *is a finite vector* $\mathcal{P} = (\alpha_1, \ldots, \alpha_n)$ *of components* α_i*,* $i \in \{1, \ldots, n\}$*.*

Note that since $R\colon States \times Act \to Dist(States)$ is a total function, every component is input-deterministic and input-enabled. We use the indices of components also for their elements, e.g. refer to the states of α_i by $States_i$.

Definition 4 (MDP induced by PIOS). *Any PIOS* $\mathcal{P} = (\alpha_1, \ldots, \alpha_n)$ *gives rise to an MDP* $[\![\mathcal{P}]\!] = (\mathcal{S}, \mathcal{A}, \mathcal{T}, s_0)$ *as follows:*

- $\mathcal{S} = \bigtimes_{i=1}^{n} States_i$

- $\mathcal{A} = \mathit{Dist}(\bigcup_{i=1}^{n} \mathit{Act}_i)$
- $\mathcal{T} \subseteq \mathcal{S} \times \mathcal{A} \times \mathit{Dist}(\mathcal{S})$ *is the smallest set of transitions* $\big((s_1, \ldots, s_n), \kappa, \mu\big)$ *for which there is an* $i \in \{1, \ldots, n\}$ *and* $\kappa_i \in \mathit{Dist}(\mathit{Act}_i \times \mathit{States}_i)$ *such that*
  - $(s_i, \kappa_i) \in G_i$,
  - *for all* $a \in \mathit{Act}_i$ *we have* $\kappa(a) = \sum_{s \in \mathit{States}_i} \kappa_i(a, s)$, *and*
  - *for all* $(s'_1, \ldots, s'_n) \in \mathcal{S}$ *we have*

$$\mu(s'_1, \ldots, s'_n) \;=\; \sum_{a \in \mathit{Act}_i} \kappa_i(a, s'_i) \prod_{\substack{j=1 \\ j \neq i}}^{n} \mu_j^a(s'_j)$$

  where $\mu_j^a = \delta(s_j)$ *provided* $a \notin \mathit{Act}_j$ *and otherwise* $\mu_j^a = R_j(s_j, a)$.
- $s_0 = (\mathit{init}_1, \ldots, \mathit{init}_n)$

Remark 1. In the MDP defined above, output distributions appear as action labels of MDP transitions. This slightly differs from the semantics of PIOSs defined in [23], where the operational behaviour is specified through *compound transitions*, explicitly comprising generative and reactive transitions as well as the action label. ∎

Observability in PIOSs. In the following, we assume a fixed PIOS $\mathcal{P} = (\alpha_1, \ldots, \alpha_n)$ with the induced MDP semantics $[\![\mathcal{P}]\!] = (\mathcal{S}, \mathcal{A}, \mathcal{T}, s_0)$ as per Definition 4. Let α_i be a component of $\mathcal{P}$ and $\mathsf{s} = (s_1, \ldots, s_n)$ a global state of $\mathcal{P}$, i.e. a state in the MDP $[\![\mathcal{P}]\!]$. By $\mathsf{s}|_i = s_i$ we denote the *state projection* of s to the i-th local state. The set of atomic observables Obs_i collects the observations a component α_i of $\mathcal{P}$ can make on global states and actions. Suppose that for all $i \in \{1, \ldots, n\}$ we are given a *local observation function* obs_i, for which we require that global states of $\mathcal{P}$ with different local states for component α_i have different observables. Formally,

$$\text{if} \;\; \mathit{obs}_i(\mathsf{s}) = \mathit{obs}_i(\mathsf{s}') \;\; \text{then} \;\; \mathsf{s}|_i = \mathsf{s}'|_i.$$

We call obs_i *purely locally observable* in case $\mathit{Obs}_i = \mathit{States}_i \cup \mathit{Dist}(\mathit{Act}_i)$ where $\mathit{obs}_i(\mathsf{s}) = \mathsf{s}|_i$ for any $\mathsf{s} \in \mathcal{S}$ and for all $\mu \in \mathcal{A}$ we have $\mathit{obs}_i(\mu)(a) = \mu(a) / \sum_{a \in \mathit{Act}_i} \mu(a)$ for $a \in \mathit{Act}_i$ and $\mathit{obs}_i(\mu)(a) = \varepsilon$ for $a \notin \mathit{Act}_i$. Intuitively, a purely local observation function observes only the local state of component α_i and the normalised action distribution on its local actions.

Partially Observable PIOS. We define *observation profiles* $\mathfrak{O}$ for $\mathcal{P}$ as tuples

$$\mathfrak{O} \;=\; (\mathit{obs}_1, \ldots, \mathit{obs}_n, \mathit{obs})$$

where obs_i are local observation functions for each component α_i, $i \in \{1, \ldots, n\}$ as defined above, and obs is a global observation function. The tuple $\mathcal{Q} = \big(\mathcal{P}, \mathfrak{O}\big)$ is called *partially observable PIOS (PO-PIOS)*.

Strategies for Partially Observable PIOSs. Let $\big(\mathcal{P}, \mathfrak{O}\big)$ be a PO-PIOS with $\mathfrak{O} = (obs_1, \ldots, obs_n, obs_{intl})$. A *local strategy* for component α_i is a scheduler σ_i for $[\![\mathcal{P}]\!]$ where for all paths π in $[\![\mathcal{P}]\!]$ there is $(last(\pi)|_i, \mu) \in G_i$ such that $\sigma_i(\pi)(a) = \sum_{s \in States_i} \mu(a, s)$ for all $a \in Act_i$.

We also consider *interleaving strategies* for $\mathcal{P}$ as functions $\sigma_{intl} \colon Paths([\![\mathcal{P}]\!]) \to Dist(\{1, \ldots, n\})$ where for each path π in $[\![\mathcal{P}]\!]$ with $\sigma_{intl}(\pi)(i) > 0$ there is some $\mu \in Dist(States_i \times Act_i)$ such that $(last(\pi)|_i, \mu) \in G_i$. An interleaving strategy σ_{intl} is *deterministic* if all distributions of σ_{intl} are Dirac. Intuitively, an interleaving strategy selects the component to choose the next move, i.e. for a path $\pi \in Paths([\![\mathcal{P}]\!])$ and $i \in \{1, \ldots, n\}$ the component α_i is scheduled with probability $\sigma_{intl}(\pi)(i)$ to select and perform one of its generative output transitions.

Strategy Profiles. We restrict our attention to those schedulers for $[\![\mathcal{P}]\!]$ that arise by composing an interleaving strategy σ_{intl} for $\mathcal{P}$ and local strategies σ_i for component α_i for all $i \in \{1, \ldots, n\}$. To formalise the composition for PO-PIOSs, i.e. also take observability into account, we define *strategy profiles* to be tuples

$$\mathfrak{P} \;=\; (\sigma_1, \ldots, \sigma_n, \sigma_{intl})$$

where σ_i is an obs_i-complying strategy for component α_i for each $i = \{1, \ldots, n\}$ and σ_{intl} is an obs_{intl}-complying interleaving strategy.

Strategy profiles can be understood as a class of observation-based schedulers for $[\![\mathcal{P}]\!]$. The scheduler $\mathfrak{S}_{\mathfrak{P}} \colon Paths([\![\mathcal{P}]\!]) \to Dist(\mathcal{A})$ for $[\![\mathcal{P}]\!]$ induced by a strategy profile $\mathfrak{P}$ is a function that assigns to any finite path $\pi = \mathsf{s}_0\, \mathsf{t}_0\, \mathsf{s}_1 \ldots \mathsf{t}_{k-1}\, \mathsf{s}_k$ in $[\![\mathcal{P}]\!]$ an action $a \in Act$ with probability

$$\mathfrak{S}_{\mathfrak{P}}(\pi)(a) \;=\; \sum_{i=1}^{n} \sigma_{intl}(\pi)(i) \cdot \sigma_i(\pi)(a) \;\; .$$

We denote by $\Pr_{\mathcal{P}}^{\mathfrak{P}}$ the probability measure $\Pr_{[\![\mathcal{P}]\!]}^{\mathfrak{S}_{\mathfrak{P}}}$.

Remark 2 (On observability in [23]). For defining strategy profiles, we followed the approach of [23] by composing interleaving and local strategies, called "interleaving schedulers" and "output schedulers". The class of observation-based schedulers $\mathfrak{S}_{\mathfrak{P}}$ that arises from strategy profiles $\mathfrak{P}$ for PO-PIOS where the global observation function provides total observability and local observation functions are purely locally observable is similar however not equivalent to the class of *distributed schedulers*. The restricted class of *strongly distributed schedulers* that imposes constraints on the component distribution of interleaving schedulers corresponds to variants of $\mathfrak{S}_{\mathfrak{P}}$ where the global observation function is not totally observable. ∎

Remark 3 (Observability by the interleaving strategy). It appears reasonable to assume that interleaving strategies have access to the local information available to the components. Formally,

- if $\mathsf{s}, \mathsf{s}' \in \mathcal{S}$ such that $obs_{intl}(\mathsf{s}) = obs_{intl}(\mathsf{s}') \neq \varepsilon$ then $obs_i(\mathsf{s}) = obs_i(\mathsf{s}')$ for all $i \in \{1, \ldots, n\}$, and

– if $a, a' \in \mathcal{A}$ such that $obs_{intl}(a) = obs_{intl}(a') \neq \varepsilon$ then $obs_i(a) = obs_i(a')$ for all $i \in \{1, \ldots, n\}$. ∎

3 Notions of Suitability

We now turn our attention to the question of how far some component κ can be considered suitable to use in combination with a given system. For this, let us consider a fixed PO-PIOS $\mathcal{Q} = (\mathcal{P}, \mathfrak{O})$ where $\mathcal{P} = (\alpha_1, \ldots, \alpha_n)$ is a PIOS with observation profile $\mathfrak{O} = (obs_1, \ldots, obs_n, obs_{intl})$. Furthermore, we follow the convention of the last section, denoting by $\mathfrak{P}$ a not necessarily fixed strategy profile for $\mathcal{Q}$. For any fresh component κ not contained in $\mathcal{P}$ we assume furthermore an observability function obs_κ and denote by $\kappa \| \mathcal{Q}$ the PO-PIOS $\big((\kappa, \mathcal{P}), (obs_\kappa, \mathfrak{O})\big)$.

Properties and their Values. In what follows, suppose that we are given a set Φ of properties or quantitative measures (e.g. defined using some temporal logics). For the definition of suitability notions, the type and syntax of these properties is irrelevant as we shall take an abstract view and deal with *valuation functions* $val^{\mathfrak{P}} \colon \Phi \to \mathbb{R}$ for strategy profiles $\mathfrak{P}$ for $\mathcal{Q}$.

Example 1. We exemplify several variants for value functions:

(i) If ϕ is a (P)CTL-like state property, then $val^{\mathfrak{P}}(\phi)$ could be defined as Boolean value not directly depending on $\mathfrak{P}$, i.e. 1 ("true") if $s_0 \models \phi$ in $[\![\mathcal{P}]\!]$ and 0 ("false") otherwise. In case ϕ is a PCTL property, the semantics of the probability operator could be restricted to range over all strategy profiles only, rather than over arbitrary schedulers for the MDP $[\![\mathcal{P}]\!]$.

(ii) If ϕ is an LTL formula or more generally an ω-regular path property, then $val^{\mathfrak{P}}(\phi)$ could be $\Pr^{\mathfrak{P}}_{\mathcal{Q}}(\phi)$, the probability of the set of infinite paths that satisfy ϕ under the probability measure induced by $\mathfrak{S}_{\mathfrak{P}}$.

(iii) If ϕ is a random variable of type $\mathit{IPaths}([\![\mathcal{P}]\!]) \to \mathbb{R}$, then $val^{\mathfrak{P}}(\phi)$ could be the expectation of ϕ on $\mathfrak{S}_{\mathfrak{P}}$-paths in $[\![\mathcal{P}]\!]$. This, of course, requires a side constraint to ensure the existence of the expectation or a default value if the expectation does not exist. Examples for such random variables are the accumulated weight until reaching a target state set, or the mean payoff when weights are attached to the transitions of $[\![\mathcal{P}]\!]$. ∎

To ease the notations that follow, we suppose that high satisfaction values are desirable in the sense that the objective is to increase values $val^{\mathfrak{P}}(\phi)$ of properties $\phi \in \Phi$ whenever possible. Furthermore, when analysing multiple objectives, we might annotate the kind of valuation function on the property. For instance, we allow for a property set $\Phi = \{\mathsf{P}(ok \mathsf{U} goal), \mathsf{E}[cost](\Diamond goal)\}$ to describe that the LTL formula $ok \mathsf{U} goal$ and $\Diamond goal$ should be evaluated with respect to their probability $\Pr^{\mathfrak{P}}_{\mathcal{Q}}(ok \mathsf{U} goal)$ and expected costs $\mathrm{Exp}^{\mathfrak{P}}_{\mathcal{Q}}(\Diamond goal)$, respectively.

Remark 4. Note that if instead one aims at minimising objectives regarding a state or path property ϕ one can switch to its complement $\neg\phi$ and consider the

maximising objective instead. Likewise, in a weighted setting with accumulated, discounted, or instantaneous weights, weights can be multiplied by -1 turning the meaning of weights to costs to be paid rather than rewards to be earned. ∎

Remark 5 (Observation-compatible properties). It appears natural to assume that the properties fit with the observations, in the sense that if ϕ is a path property then ϕ does not distinguish between paths with identical observations. Formally, for $\pi_1, \pi_2 \in \mathit{IPaths}(\llbracket \mathcal{P} \rrbracket)$ with $\mathit{obs}_i(\pi_1) = \mathit{obs}_i(\pi_2)$ for $i \in \{1, \ldots, n\} \cup \{\mathit{intl}\}$ and $\pi_1 \models \phi$, then $\pi_2 \models \phi$. Similarly, if ϕ is a random variable formalising a reward to be earned along paths one might require that paths with the same observation have the same value under ϕ. ∎

3.1 Threshold Suitability

We are now in the position to propose formal criteria for a component β to be suitable in the context of other components. Suitability of β and $\mathcal{Q}$ is defined by imposing conditions on the PO-PIOS $\beta \| \mathcal{Q}$.

Definition 5 (Universal Threshold Suitability (∀TS)**).** *Let Φ be a set of properties with a valuation function for the PO-PIOS $\beta \| \mathcal{Q}$ and let $\vartheta = (\vartheta_\phi)_{\phi \in \Phi}$ be a real vector assigning a threshold value for each property $\phi \in \Phi$. β and $\mathcal{Q}$ are said to be* universally threshold-suitable *with respect to (Φ, ϑ) if for all strategy profiles $\mathfrak{P}$ for $\beta \| \mathcal{Q}$ and for each property $\phi \in \Phi$ we have*

$$\mathit{val}^{\mathfrak{P}}(\phi) \quad > \quad \vartheta_\phi.$$

In a nutshell, the definition says that $\beta \| \mathcal{Q}$ will meet all the criteria being part of $\mathit{val}^{\mathfrak{P}}(\cdot)$ regardless of what happens to the system, in terms of the strategy profiles imaginable. An alternative definition arises when β has the freedom to choose its strategy depending on the decisions of global control and other components.

Definition 6 (Existential Threshold Suitability (∃TS)**).** *Let Φ be a set of properties with a valuation function for the PO-PIOS $\beta \| \mathcal{Q}$ and let $\vartheta = (\vartheta_\phi)_{\phi \in \Phi}$ be a real vector assigning a threshold value for each property $\phi \in \Phi$. Then, β and $\mathcal{Q}$ are said to be* existentially threshold-suitable *with respect to (Φ, ϑ) if*

> *for all obs_i-complying strategies σ_i for α_i, $i \in \{1, \ldots, n\}$ and*
> *for all $\mathit{obs}_{\mathit{intl}}$-complying interleaving strategies σ_{intl}*
> *there exists an obs_β-complying strategy σ_β for β*

such that with $\mathfrak{P} = (\sigma_\beta, \sigma_1, \ldots, \sigma_n, \sigma_{\mathit{intl}})$ for each property $\phi \in \Phi$ we have

$$\mathit{val}^{\mathfrak{P}}(\phi) \quad > \quad \vartheta_\phi.$$

A practical example for threshold suitability are *Real Driving Emissions* (RDE) tests where it is required that the amount of emitted pollutants is below certain thresholds for all reasonable driver behaviours [28]. In terms of threshold suitability, a driver behaviour corresponds to a strategy and the system could

constrain nondeterministic choices to those that are reasonable as required. Universal threshold suitability then asks whether the emitted pollutants are below their respective thresholds for all possible RDE tests as required by the RDE regulation. In contrast, existential threshold suitability asks whether it is possible to pass an individual test by driving accordingly.

3.2 Degree of Suitability

To provide a more fine-grained mechanism to quantify how suitable components behave, we go beyond the simple discrimination discussed thus far, i.e. whether or not they are suitable. For this, we introduce measures of *degrees of suitability*, which rely on an aggregation function $f\colon \mathbb{R}^{\Phi} \to \mathbb{R} \cup \{\pm\infty\}$ for the potential satisfaction values of properties. Here, $\mathbb{R}^{\Phi}$ stands for the set of real-valued vectors $(v_\phi)_{\phi\in\Phi}$ over a set of properties Φ.

Example 2. Typical candidates for an aggregation function f are:

(i) Weighted sums $f(v) = \sum_{\phi\in\Phi} w_\phi \cdot v_\phi$ of the individual satisfaction values defined over vectors $v = (v_\phi)_{\phi\in\Phi}$ for a finite set of properties Φ. This corresponds to the switch to a composite valuation function

$$(\mathfrak{P}, \Phi) \;\mapsto\; \sum_{\phi\in\Phi} w_\phi \cdot val^{\mathfrak{P}}(\phi)$$

(ii) The valuation function of a (single) distinguished property $\psi \in \Phi$ under threshold conditions for the values for all other properties, and $-\infty$ otherwise. That is:

$$f(v) \;=\; \begin{cases} v_\psi & \text{if } v_\phi > \vartheta_\phi \text{ for all } \phi \in \Phi \setminus \{\psi\} \\ -\infty & \text{otherwise} \end{cases}$$

where ϑ_ϕ are thresholds as in Definition 5 or Definition 6.

(iii) Combinations of (i) and (ii).

In practical situations, the latter are all but uncommon. For instance, when consumer organisations like the Dutch *Consumentenbond* and the German *Stifung Warentest* [1] carry out safety tests of consumer products, it is very common to have some criteria where a certain threshold must be met in order to be considered eligible, and that the other criteria are weighted with percentages and mapped into a scalar of normed range. This principle is also behind the European car safety performance assessment programme *EuroNCAP* [2]. ∎

Definition 7 (Universal Degree of Suitability (∀DS)**).** *Let Φ be a set of properties with a valuation function for the PO-PIOS $\beta\|\mathcal{Q}$ and let $f\colon \mathbb{R}^{\Phi} \to \mathbb{R} \cup \{\pm\infty\}$ be an aggregation function. Then, the* universal degree of suitability *of β with respect to $\mathcal{Q}$ is defined as*

$$\inf_{\mathfrak{P}} \; f\Big(\big(val^{\mathfrak{P}}(\phi)\big)_{\phi\in\Phi} \Big)$$

where the infimum ranges over all strategy profiles $\mathfrak{P}$ for $\beta\|\mathcal{Q}$.

As in the case for threshold suitability, we also present an existential version of suitability degrees where component β has the freedom of choosing a strategy depending on the interleaving strategy and local strategies of other components.

Definition 8 (Existential Degree of Suitability ($\exists$DS)**).** *Let Φ be a set of properties with a valuation function for the PO-PIOS $\beta\|\mathcal{Q}$ and let $f\colon \mathbb{R}^{\Phi} \to \mathbb{R}\cup\{\pm\infty\}$ be an aggregation function. Then, the* existential degree of suitability *of β with respect to $\mathcal{Q}$ is defined as*

$$\sup_{\sigma_\beta} \inf_{\mathfrak{P}[\beta]} f\Big(\big(val^{\mathfrak{P}[\beta]}(\phi)\big)_{\phi\in\Phi}\Big)$$

where the supremum ranges over all obs_β-complying strategies σ_β for β and the infimum ranges over all strategy profiles $\mathfrak{P}[\beta] = (\sigma_\beta, \sigma_1, \ldots, \sigma_n, \sigma_{intl})$ for $\beta\|\mathcal{Q}$.

It is conceivable to combine both of the above notions in a weighted setting, but we do not spell out the details here. For instance, one may be interested in the average emissions in the best and the worst case.

3.3 Suitability Relations

We now consider two composite PO-PIOS $\beta\|\mathcal{Q}$ and $\gamma\|\mathcal{Q}$ and introduce formal notions that spell out in what sense β is *more suitable* than γ when running in the context of $\mathcal{Q}$ with respect to a given set Φ of properties with valuation functions $val^{\mathfrak{P}}\colon \Phi \to \mathbb{R}$ and aggregation functions $f\colon \mathbb{R}^{\Phi} \to \mathbb{R}\cup\{\pm\infty\}$ for both $\beta\|\mathcal{Q}$ and $\gamma\|\mathcal{Q}$. Although β and γ can have different observables, we suppose here that all the corresponding observation functions of $\beta\|\mathcal{Q}$ and $\gamma\|\mathcal{Q}$ coincide. Furthermore, we assume the following requirements for the observation functions of $\kappa\|\mathcal{Q}$, $\kappa \in \{\beta, \gamma\}$:

(Loc) We assume that the components of $\mathcal{Q}$ do not have information on the local states of κ in the sense that the observable of global state $(s_\kappa, s_1, \ldots, s_n)$ in $[\![\kappa\|\mathcal{Q}]\!]$ only depends on $(s_1, \ldots, s_n)$ but not on s_κ. Likewise, we suppose that actions in $Act_\kappa \setminus Act_i$ are invisible for all α_i, $i = 1, \ldots, n$.

(Intl) Global observation functions for $\kappa\|\mathcal{Q}$ do not have access to the local state of κ and cannot see the actions in $(Act_\beta \setminus Act_\gamma) \cup (Act_\gamma \setminus Act_\beta)$. Formally,
- $obs_{intl}(s_\kappa, s_1, \ldots, s_n) = obs_{intl}(s'_\kappa, s_1, \ldots, s_n)$ for all states $s_\kappa, s'_\kappa \in States_\kappa$ and $s_i \in States_i$ for $i = 1, \ldots, n$, and
- $obs_{intl}(a) = \varepsilon$ for each action $a \in (Act_\beta \setminus Act_\gamma) \cup (Act_\gamma \setminus Act_\beta)$.

Assumption **Loc** implies that if $\mathcal{P} = (\alpha_1, \ldots, \alpha_n)$ then any obs_i-complying strategy for α_i in $\beta\|\mathcal{P}$ is also an obs_i-complying strategy for α_i in $\gamma\|\mathcal{P}$, and vice versa. Note that here, we regard strategies as functions that take as input an observation sequence. Assumption **Intl** ensures that $\beta\|\mathcal{Q}$ and $\gamma\|\mathcal{Q}$ have the same interleaving strategies. While assumption **Loc** is a fairly natural and standard assumption in the partial information setting, assumption **Intl** appears technically rather strong. In an exemplary setting, **Loc** means that when testing the performance of two cars, we do not exploit that one of them offers the possibility

to turn on and off "boost mode" while the other one does not. **Intl** then corresponds to the idea that the behaviour considered relevant is observed from the outside, and does not refer to particularities of the components to be compared, such as a warning light only available in one of the cars.

Definition 9 (Universally More Suitable (∀MS)). *Let Φ be a set of properties with a valuation function for the PO-PIOS $\beta\|\mathcal{Q}$ and let $f\colon \mathbb{R}^{\Phi} \to \mathbb{R} \cup \{\pm\infty\}$ be an aggregation function. Under the assumptions **Loc** and **Intl**, β is said to be* universally more suitable *than γ if for all strategy profiles $\mathfrak{P}[\gamma] = (\sigma_\gamma, \sigma_1, \ldots, \sigma_n, \sigma_{intl})$ for $\gamma\|\mathcal{Q}$ and for all obs_β-complying strategies σ_β for β we have*

$$f\Big(\big(val^{\mathfrak{P}[\beta]}(\phi)\big)_{\phi\in\Phi}\Big) \;>\; f\Big(\big(val^{\mathfrak{P}[\gamma]}(\phi)\big)_{\phi\in\Phi}\Big)$$

where $\mathfrak{P}[\beta] = (\sigma_\beta, \sigma_1, \ldots, \sigma_n, \sigma_{intl})$.

Note that due to the assumption **Intl**, for any obs_β-complying strategy for β we have that $\mathfrak{P}[\beta]$ is indeed a strategy profile for $\beta\|\mathcal{Q}$. Intuitively, a component β is universally more suitable than γ if for all strategy profiles $\mathfrak{P}[\beta]$ for $\beta\|\mathcal{Q}$, we cannot find a local strategy σ_γ for γ that leads to a higher degree of suitability in $\gamma\|\mathcal{Q}$ when replacing σ_β in $\mathfrak{P}[\beta]$ by σ_γ.

Similar as for the notions of threshold suitability and the degrees of suitability, we also introduce an existential version of the "more suitable" relation that allows σ_β to react on behaviour imposed by σ_γ.

Definition 10 (Existentially More Suitable ∃MS). *Let Φ be a set of properties with a valuation function for the PO-PIOS $\beta\|\mathcal{Q}$ and let $f\colon \mathbb{R}^{\Phi} \to \mathbb{R} \cup \{\pm\infty\}$ be an aggregation function. Under the assumptions **Loc** and **Intl**, β is said to be* existentially more suitable *than γ if for all strategy profiles $\mathfrak{P}[\gamma] = (\sigma_\gamma, \sigma_1, \ldots, \sigma_n, \sigma_{intl})$ for $\gamma\|\mathcal{Q}$ there is an obs_β-complying strategy for β such that*

$$f\Big(\big(val^{\mathfrak{P}[\beta]}(\phi)\big)_{\phi\in\Phi}\Big) \;>\; f\Big(\big(val^{\mathfrak{P}[\gamma]}(\phi)\big)_{\phi\in\Phi}\Big)$$

where $\mathfrak{P}[\beta] = (\sigma_\beta, \sigma_1, \ldots, \sigma_n, \sigma_{intl})$.

To determine the ∀MS- and ∃MS-relations provided in Definitions 9 and 10, we have to evaluate aggregated valuations with respect to an observation-based scheduler for both, $[\![\beta\|\mathcal{Q}]\!]$ and $[\![\gamma\|\mathcal{Q}]\!]$. Since this might require more involved analysis techniques, an independent analysis of $[\![\beta\|\mathcal{Q}]\!]$ and $[\![\gamma\|\mathcal{Q}]\!]$ towards deriving a more strict notion of suitability is desirable.

Definition 11 (Strictly More Suitable (SMS)). *Let Φ be a set of properties with a valuation function for the PO-PIOS $\beta\|\mathcal{Q}$ and let $f\colon \mathbb{R}^{\Phi} \to \mathbb{R} \cup \{\pm\infty\}$ be an aggregation function. Then, β is said to be* strictly more suitable *than γ if*

$$\inf_{\mathfrak{P}[\beta]} f\Big(\big(val^{\mathfrak{P}[\beta]}(\phi)\big)_{\phi\in\Phi}\Big) \;>\; \sup_{\mathfrak{P}[\gamma]} f\Big(\big(val^{\mathfrak{P}[\gamma]}(\phi)\big)_{\phi\in\Phi}\Big)$$

where the infimum ranges over all strategy profiles $\mathfrak{P}[\beta]$ for $\beta\|\mathcal{Q}$ and the supremum ranges over all strategy profiles $\mathfrak{P}[\gamma]$ for $\gamma\|\mathcal{Q}$.

Note that if β is strictly more suitable than γ, then β is also universally and existentially more suitable than γ.

4 Suitability Analysis

We now turn to the algorithmic side of the definitions proposed. Assume we are given an input PO-PIOS $\mathcal{Q} = (\mathcal{P}, \mathfrak{O})$, two components β and γ, a set of properties Φ with a valuation function $val^{\mathfrak{P}} : \Phi \to \mathbb{R}$, and an aggregation function $f : \mathbb{R}^{\Phi} \to \mathbb{R} \cup \{\pm\infty\}$. Then we consider the following decision problems:

(a) For a threshold vector $\vartheta = (\vartheta_\phi)_{\phi \in \Phi}$ decide whether β and $\mathcal{Q}$ are threshold suitable with respect to (Φ, ϑ) as defined in Definitions 5 and 6.
(b) For a threshold $\vartheta \in \mathbb{R}$ decide whether the suitability degree of β with respect to $\mathcal{Q}$ exceeds ϑ for notions defined in Definitions 7 and 8.
(c) Decide whether β is more suitable than γ with respect to $\mathcal{Q}$ as defined in Definitions 9, 10, and 11.

In the sequel, we provide positive and negative answers for the above decision problems. Due to the lack of space, we moved full proofs to the appendix.

Theorem 1. *The problems (a)–(c) are undecidable for all valuation functions of Example 1 and all aggregation functions of Example 2.*

Due to the above theorem, one has to consider restrictions of strategy profiles, PO-PIOSs, and/or valuation functions in order to enable the analysis of suitability notions. A natural candidate for a restriction would be to only consider strategy profiles that are composed of strategies whose decisions can be represented as a finite-state machine. Existing results on IPIOAs [23] suggest that this direction is indeed worth to consider. In this paper, we do not a priori restrict the class of schedulers, but restrict the PO-PIOSs making up the system.

Threshold and Degree of Suitability Analysis. We arrive at a positive decidability result by restricting to total observation.

Proposition 1. *For all valuation functions of Example 1 and all aggregation functions of Example 2, problems (a) and (b) are decidable if all observation functions in the observation profile of $\beta \| \mathcal{Q}$ are totally observable.*

The above proposition relies on the fact that the class of observation-based schedulers $\mathfrak{S}_{\mathfrak{P}}$ for observation profiles consisting of totally observable observation functions in $\beta \| \mathcal{Q}$ coincides with the full class of schedulers for $[\![\beta \| \mathcal{Q}]\!]$. Thus, threshold suitability and deciding degree of suitability questions boil down to multi-objective analysis tasks for MDPs [12,19,21] in case of universal notions of suitability and $2\frac{1}{2}$-player games in case of existential notions of suitability [13].

More Suitable Relation Analysis. For problem (c), totally observable observation functions in observation profiles violate conditions **(Loc)** and **(Intl)**, such that we present different conditions to provide decidability.

Proposition 2. *For all valuation functions of Example 1 and all aggregation functions of Example 2, problems (a)–(c) are decidable if*

(i) all components in $\mathcal{Q}$ are not containing any generative input transition, and
(ii) the observation function for β, respectively γ, in the observation profile of $\beta \| \mathcal{Q}$, respectively $\gamma \| \mathcal{Q}$, is totally observable.

Due to (i), β and γ contain all generative input transitions and the interleaving strategies for $\beta \| \mathcal{Q}$ and $\gamma \| \mathcal{Q}$ agree in the sense that they are independent from the global state, always picking component β, respectively γ, to perform the next move. To this end, the only nondeterminism in the composite system stems from the components β or γ, respectively. In combination with condition (ii), solving problem (c) reduces to multi-objective analysis tasks for MDPs [12,19,21].

5 Racetrack – A Case Study

In this section, we explain and illustrate the applicability of the theoretical concepts discussed above by means of a simple scenario known as *Racetrack* [22]. For the fragment that can be reduced to standard methods for MDPs, we present initial experimental results obtained with PRISM [29]. The tooling as well as the obtained results are made available for download[4]. The computation of the results shown in this section took less than 40 minutes on a standard laptop.

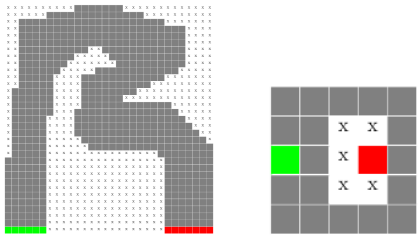


Fig. 1. Two example maps with start line in green, goals in red, and walls marked with `x`.

5.1 Racetrack Scenario

Originally, Racetrack is a pen and paper game [22], comprising a *vehicle* which has to manoeuvre through a given two-dimensional discrete *track* with a designated start and goal, walls on the boundaries, and barriers on the track. The vehicle starts with no initial velocity from a starting position, with the objective to reach the goal as fast as possible without crashing into a wall or barrier. We extend this setting with costs for time steps, fuel consumption and CO_2-emission yielding a trade-off between costs and reaching the goal fast. To this end, the *driver* modifies the current velocity vector by means of acceleration and steering actions. Apart from those nondeterministic actions, we extend our setting to a probabilistic environment such that actions may fail with a certain probability. We obtain a PIOS-based model with MDP semantics that allows, e.g. emulating slippery road conditions, where the driver's action may not induce the intended change in the velocity or direction. As a consequence, the vehicle will be unable to almost surely reach the goal, even when considering the best driver (namely a maximising scheduler for the underlying MDP). Stochastic variants of the racetrack scenario have traditionally served as benchmarks for MDP algorithms in

[4] `https://doi.org/10.5281/zenodo.3970766` [6]

the AI community [9,10,36] and lately also considered in the context of statistical model checking [25].

For our case study, we generalised the racetrack scenario by choosing a feature-oriented modelling approach [5,4] in the probabilistic variant introduced in [18,15]. To this end, features encapsulate the behavioural descriptions and characteristics for different road conditions (in the following: tarmac, sand, and ice), tank sizes (here: small, medium, large) with different fuel capacities, and engine variants, which are characterised by a maximal velocity $v_{\max}$ and maximal acceleration $a_{\max}$ (here as values from the set $\{1, 2, 3\}$). This feature model then gives rise to an entire family of PIOS (rather than just one) with three components: the *engine*, the *tank*, and the *map*. In our case we end up with $3^4 = 81$ family members, standing for separate models for each setting. The engine component controls the acceleration and thereby the speed of the car by generative input transitions corresponding to acceleration changes. A driver is in control of the car by selecting acceleration actions in x- and y-dimension. The tank updates its fuel level in reaction to the engine's acceleration decisions and gets trapped in a failure state once all fuel is entirely used up. Note that PIOS components have to be input-enabled and, hence, the tank has to be able to react to all acceleration decisions independent on whether there is enough fuel left for the required acceleration change. Finally, the map models the terrain as a grid with fixed road conditions and with starting cells, road cells, barrier cells and goal cells. Throughout this section we use a tiny map of size 5×5 as depicted on the right of Figure 1, which is included in the available artefacts. Depending on the drivers choices, i.e. in reaction to the engine's generative transitions, the map then updates the car's position on the track under the given road conditions. As the engine is the only generative component in this setting, our assumptions with regard to the case study are fulfilled and the system is completely determined by the driver's strategy for the engine. Following the decidability result of Proposition 2, this allows to use existing tooling for the analysis of MDPs.

5.2 A New Car

Imagine that we would like to purchase a new car which we primarily need to drive to the office every day. Hence, the map and in particular the possible routes to the office are fixed, while the road conditions may vary from day to day. Now, the car salesman asks us which tank and engine variant we would like to purchase. Obviously, we want to configure our new car such that it suits our needs and here our suitability notions come into play. To apply them, we first have to fix a context $\mathcal{Q}$ and decide on the component(s) for which we would like to analyse suitability. Assume that we already decided that we would like a medium sized tank, but we are still uncertain about the engine variant. Hence, we are interested in the suitability of engine variants. Notably, this scenario entails that the road conditions are part of the fixed context as well. However, we can still carry out the analysis for different contexts to cover threshold suitability. For instance, in case we are interested in whether a particular engine variant is threshold suitable for all road conditions.

Threshold Suitability. Threshold suitability allows us to define *minimal requirements* for our new car. Imagine that we would like the probability of reaching our office (without running out of fuel or crashing into walls or barriers) to be at least 0.55. At the same time, we want the expected number of time steps to be less than 20, the expected fuel consumption to be less than 39 and the expected CO_2-emission to be less than 35. Formally, these requirements manifest in the set of properties

$$\Phi \;=\; \{\;\; \mathsf{P}(\mathit{status\_ok}\,\mathsf{U}\,\mathit{office}),\; \mathsf{E}[\mathit{timesteps}](\Diamond\,\mathit{office}),\\ \mathsf{E}[\mathit{fuel}](\Diamond\,\mathit{office}),\; \mathsf{E}[\mathit{CO_2}](\Diamond\,\mathit{office})\;\;\}$$

and respective thresholds ϑ_ϕ for each property.

Threshold suitability allows us to decide whether a car with a particular engine variant β as characterised by a maximal velocity $v_{\max}$ and acceleration $a_{\max}$ fulfils these thresholds in context $\mathcal{Q}$ by considering $\beta \| \mathcal{Q}$. As ($\forall$TS) quantifies over all strategy profiles and β is nondeterministic with regard to the acceleration vector, it tells us whether the thresholds will be satisfied independent of the driver, i.e. it essentially assumes the worst possible driver. In contrast, ($\exists$TS) merely requires that there exists a strategy profile for which all thresholds are satisfied and thereby assumes the best possible driver. Intuitively ($\forall$TS) is not particularly helpful in our case as even with the best car, the worst possible driver can waste all fuel driving in circles, never reaching the office. The same phenomenon also applies to the other notions of universal suitability.

For our analysis we considered all engine variants with $a_{\max}, v_{\max} \in \{1, 2, 3\}$ on sand and on tarmac with a medium sized tank. For all variants we computed a multi-objective with a lower bound on reaching the goal without crashing and upper bounds on the expected fuel consumption, time steps and CO_2-emission. We refer to Section 5.3 for the technical details of the multi-objective analysis. From the analysis we can conclude that all engine variants with $a_{\max} = 1$ are existentially threshold suitable on sand, while all the others are not. On tarmac, however, all engine variants with $a_{\max} \in \{1, 2\}$ are existentially threshold suitable while all the others, i.e. with $a_{\max} = 3$, are not. If we would like to go off-road with our car we should thus purchase a car with an engine variant satisfying $a_{\max} = 1$. Otherwise, every engine variant with $a_{\max} \in \{1, 2\}$ is just fine. The full result, including the numbers for icy road conditions are included in the available artifacts.

Degree of Suitability. While threshold suitability is a purely qualitative notion, the degree of suitability provides a quantitative measure. Coming back to our example, multiple engine variants meet our minimal requirements as set by our thresholds, however, one of them may for instance be more fuel efficient than the others. Here suitability degrees come into play.

To apply ($\exists$DS) and ($\forall$DS) we first need to specify an aggregation function combining the values for the different properties into a single value depending on our requirements. Assume that it is more important for us to save time than it is to preserve fuel and that it is more important for us to preserve fuel than to emit

less CO_2. In this case, we may define an aggregation function f as a weighted sum giving weight -50 to the time it takes, -30 to the fuel consumption, and -20 to the CO_2 emissions with the set of properties being:

$$\Phi \;=\; \{\; \mathsf{E}[\mathit{timesteps}](\Diamond \mathit{office}),\; \mathsf{E}[\mathit{fuel}](\Diamond \mathit{office}),\; \mathsf{E}[\mathit{CO_2}](\Diamond \mathit{office}) \;\}.$$

Note that we weighted all properties with negative values as all these properties are subject to minimisation (cf. Remark 4). Analogously to threshold suitability, (∀DS) and (∃DS) provide a suitability degree assuming the worst, respectively best, driver behaviour.

For tarmac and the medium sized tank we determined the following suitability degrees: if $a_{\max} = 1$ then the suitability degree is -1450, if $a_{\max} = 2$ then the suitability degree is -1900, and if $a_{\max} = 3$ then the suitability degree is -2350. This is explained by the fact that an engine with a higher $a_{\max}$ is assumed to consume more fuel than a weaker engine. While all engine variants with $a_{\max} \in \{1, 2\}$ are existentially threshold suitable for tarmac, the engines with $a_{\max} = 1$ are more economical. Hence, we conclude that we should purchase a car with $a_{\max} = 1$. The technical details can again be found in Section 5.3.

More Suitable Relations. In addition to the already discussed notions of suitability, we defined *more suitable* relations that directly compare two variants. While one may use suitability degrees to compare two engine variants, this assumes the worst respectively best driver behaviour for both variants. Instead, the *more suitable* relations compare the worst strategy profile for one component with the best for the other (cf. Definition 9) or, as a more relaxed existential notion, the best component behaviour assuming the worst system behaviour with the best strategy profile for the other component (cf. Definition 10). We are not aware of tool support for these notions.

The strict variation (cf. Definition 11) is merely a comparison of the best degree for one component with the worst degree for the other. Specifically, the worst degree will always be $-\infty$ because the worst driver can just drive in a circle. Hence, while easier to analyse, this notion of suitability is too coarse for our example. The result would be that no engine variant is strictly dominating.

5.3 Implementation and Technical Aspects

We now present the technical details regarding the analysis for existential threshold suitability and degree of suitability as discussed in the previous section using standard methods for MDPs as provided by Prism.

Threshold Suitability. Using Prism's multi-objective engine [20] and manually translating the family of PIOS to their corresponding MDPs we were able to obtain experimental results for (∃TS) and using the following numerical multi-objective query:

```
multi(
```

```
    P>=PBound["ap_status_ok" U "ap_office"],
    R{"fuel"}<=FBound[C], R{"timesteps"}<=TBound[C],
    R{"CO2"}<=CBound[C]
  );
```

Note that in the above query, we used non-strict bounds on the valuation functions as opposed to our theoretical framework. This is due to the current tool support provided by Prism. Furthermore, encoding (∃TS) into a numerical multi-objective query required us to switch from the expected reachability rewards to total accumulated rewards, as expected rewards are not yet supported by the multi-objective engine. This change is reasonable, because the total accumulated rewards are all finite due to the fact that the number of time steps is bounded until the car can no longer move and one ends up in a trap state where no further reward is gained. Furthermore, the goal states, when the office is reached and the car stops, and the crashed states enjoy this property. Also the actual bounds used within the total reward properties can be scaled with a factor `PBound`. This is due to the fact that the multi-objective engine computes optimal weights for each property and the computed scheduler is in fact a randomised scheduler that balances out the individual objectives. Hence, the upper bounds for the total expected costs (fuel and CO_2) used within the multi-objective query were scaled down by multiplying with `PBound` and rounding.

While (∀TS) does not seem to be as important as (∃TS) in our case study, let us note that there is tool support by Prism to decide (∀TS) for our set of properties. For this, one can solve (∀TS) by considering a dual problem on multiple (∃TS) questions of single properties [21].

Degree of Suitability. To the best of our knowledge, there exists no tool support for aggregating and weighting properties over a particular scheduler and then searching for a scheduler which minimises respectively maximises this aggregation. But in case of probability and expectation properties, we can transform the model from a multi-reward into a single-reward model by pulling inwards the aggregation function, so that we arrive at weighted sums as rewards on edges. This is justified by the distributivity law, and results in the following transition reward structure:

```
reward "wsum" := (50 * c_timestep) + (30 * f_fuel_consumption)
                                   + (20 * f_co2_production)
```

Note that we switched here to positive weights, because Prism hardly supports negative rewards. Now, by computing the minimal expected reward for finally reaching the goal, we compute how *unsuitable* the system is in the best case. In the end, we have to invert the result in order to obtain the actual existential suitability degree as specified with the negative weights above. Please note that the expected reachability reward will be ∞ for all soils different from tarmac, as the probability of reaching the goal is strictly less than one.

Feature-oriented Analysis and Scalability. Using our feature-based modelling approach, the analysis for different contexts could be in principle carried out separately one-by-one per context or in a single run by means of an all-in-one analysis [16,4,39,15]. The latter relies on our family model that encodes all settings in a single model. It is well known (see, e.g. [17,16,39]) that all-in-one approaches can mitigate the exponential blowup of feature combinations in the number of features by exploiting similarities of behaviours within different settings using symbolic analysis techniques such as implemented in Prism's MTBDD engine. However, as the current implementation of Prism to analyse multi-objective properties does not fully support family models and symbolic engines, we had to follow a one-by-one analysis approach to compute results for different notions of suitability. The lack of such a support is also the reason why we used a comparably small case-study setup with the 5x5 map shown on the right of Figure 1. The map on the left of Figure 1 is an example of realistically sized map that is also considered in the automated planning community [9]. Here, the Prism family model contained $6 \cdot 81 = 486$ family members and led to a model with more than $1.1 \cdot 10^9$ states. As this model could not be explicitly represented in memory, we considered a symbolic representation with $1.4 \cdot 10^6$ MTBDD nodes.[5] Using Prism's MTBDD engine applied on the family model, an all-in-one analysis of single-objective threshold suitability was possible for this larger map, checking $(\forall \mathrm{TS})$ for $\Phi = \{\mathsf{P}(\mathit{status\_ok}\,\mathsf{U}\,\mathit{office})\}$ with $\vartheta_\Phi = 0.35$ in less than 14 minutes, equivalent to about 10 seconds per configuration. A corresponding one-by-one analysis required around 10 hours in total, i.e. in average more than 7 minutes per configuration. This comparison shows the potential of our feature-based modelling and analysis approach.

6 Concluding Remarks

This paper has introduced notions formalising the suitability of components in the context of probabilistic systems given as PO-PIOSs. We presented undecidability results for the general case of suitability notions and established decidability for restricted classes of PO-PIOSs that we used in our case study. Further positive results on suitability notions could be expected with respect to restricted classes of strategy profiles, e.g. where all strategies in a profile are finite-memory strategies [23].

Many facets of these suitability notions can be seen as future work. The definitions presented rely on strict comparisons in the case of threshold suitability and "more suitable" formalisations. Instead one may also consider relations that implement "at least as suitable", i.e. replace the strict comparison $>$ relation by $\geqslant$ in our formal definitions. For this, it is an open question whether threshold suitability is decidable for simple valuation functions. In addition, further kinds of valuation and aggregation functions could be investigated, e.g. by including energy-utility trade-offs into the measure of suitability or rely on conditional probabilities and expectations [7].

[5] Also exploiting variable-reordering techniques from [27] on the generated model.

On the evaluation and practical side, an implementation of the multi-objective engine of PRISM supporting family models would enable to exploit the benefits of our family-based approach towards an all-in-one suitability analysis.

References

1. Test-ablauf – So testet die Stiftung Warentest, https://www.test.de/unternehmen/testablauf-5017344-0/, accessed: 2020-06-30
2. The Official Site of The European New Car Assessment Programme, https://www.euroncap.com/en/, accessed: 2020-06-30
3. Alur, R.: Principles of Cyber-Physical Systems. The MIT Press (2015)
4. Apel, S., Batory, D., Kästner, C., Saake, G.: Feature-Oriented Software Product Lines. Concepts and Implementation. Springer (2013)
5. Apel, S., Kästner, C.: An overview of feature-oriented software development. Journal of Object Technology **8**, 49–84 (2009)
6. Baier, C., Dubslaff, C., Hermanns, H., Klauck, M., Klüppelholz, S., Köhl, M.A.: Tooling, Data and Results for "Components in Probabilistic Systems: Suitable by Construction" (2020), available at http://doi.org/10.5281/zenodo.3970766
7. Baier, C., Dubslaff, C., Klüppelholz, S.: Trade-off analysis meets probabilistic model checking. In: Proc. of the 23rd Conference on Computer Science Logic and the 29th Symposium on Logic In Computer Science (CSL-LICS). pp. 1:1–1:10. ACM (2014)
8. Baier, C., Größer, M., Bertrand, N.: Probabilistic ω-automata. Journal of the ACM **59**(1), 1:1–1:52 (2012)
9. Barto, A.G., Bradtke, S.J., Singh, S.P.: Learning to act using real-time dynamic programming. Artif. Intell. **72**(1-2), 81–138 (1995)
10. Bonet, B., Geffner, H.: Labeled RTDP: improving the convergence of real-time dynamic programming. In: ICAPS. pp. 12–21 (2003)
11. Canetti, R., Cheung, L., Kaynar, D.K., Liskov, M.D., Lynch, N.A., Pereira, O., Segala, R.: Task-structured probabilistic I/O automata. J. Comput. Syst. Sci. **94**, 63–97 (2018). https://doi.org/10.1016/j.jcss.2017.09.007, https://doi.org/10.1016/j.jcss.2017.09.007
12. Chatterjee, K., Majumdar, R., Henzinger, T.: Markov decision processes with multiple objectives. In: STACS (February 2006), http://chess.eecs.berkeley.edu/pubs/81.html
13. Chen, T., Forejt, V., Kwiatkowska, M.Z., Simaitis, A., Wiltsche, C.: On stochastic games with multiple objectives. In: Mathematical Foundations of Computer Science 2013 - 38th International Symposium, MFCS 2013, Klosterneuburg, Austria, August 26-30, 2013. Proceedings. pp. 266–277 (2013). https://doi.org/10.1007/978-3-642-40313-2_25
14. Cheung, L., Lynch, N.A., Segala, R., Vaandrager, F.W.: Switched PIOA: parallel composition via distributed scheduling. Theor. Comput. Sci. **365**(1-2), 83–108 (2006). https://doi.org/10.1016/j.tcs.2006.07.033, https://doi.org/10.1016/j.tcs.2006.07.033
15. Chrszon, P., Dubslaff, C., Klüppelholz, S., Baier, C.: Profeat: feature-oriented engineering for family-based probabilistic model checking. Formal Aspects of Computing **30**(1), 45–75 (2018)
16. Classen, A., Heymans, P., Schobbens, P.Y., Legay, A., Raskin, J.F.: Model checking lots of systems: Efficient verification of temporal properties in software product lines. In: Proceedings of ICSE'2010. pp. 335–344. ACM (2010)

17. Czarnecki, K., Eisenecker, U.W.: Generative Programming: Methods, Tools, and Applications. ACM Press/Addison-Wesley Publishing Co. (2000)
18. Dubslaff, C., Baier, C., Klüppelholz, S.: Probabilistic model checking for feature-oriented systems. Transactions on Aspect-Oriented Software Development **12**, 180–220 (2015). https://doi.org/10.1007/978-3-662-46734-3_5
19. Etessami, K., Kwiatkowska, M., Vardi, M., Yannakakis, M.: Multi-objective model checking of Markov decision processes. Logical Methods in Computer Science **4**(4) (2008)
20. Forejt, V., Kwiatkowska, M., Parker, D.: Pareto curves for probabilistic model checking. In: Chakraborty, S., Mukund, M. (eds.) Proceedings Automated Technology for Verification and Analysis (ATVA'12). pp. 317–332. Springer, Berlin, Heidelberg (2012)
21. Forejt, V., Kwiatkowska, M.Z., Norman, G., Parker, D., Qu, H.: Quantitative multi-objective verification for probabilistic systems. In: Tools and Algorithms for the Construction and Analysis of Systems - 17th International Conference, TACAS 2011, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2011, Saarbrücken, Germany, March 26-April 3, 2011. Proceedings. pp. 112–127 (2011). https://doi.org/10.1007/978-3-642-19835-9_11, https://doi.org/10.1007/978-3-642-19835-9_11
22. Gardner, M.: Mathematical games. Scientific American **229**, 118–121 (1973)
23. Giro, S., D'Argenio, P.R., Fioriti, L.M.F.: Distributed probabilistic input/output automata: Expressiveness, (un)decidability and algorithms. Theoretical Computer Science **538**, 84 – 102 (2014). https://doi.org/https://doi.org/10.1016/j.tcs.2013.07.017, quantitative Aspects of Programming Languages and Systems (2011-12)
24. van Glabbeek, R.J., Smolka, S.A., Steffen, B.: Reactive, generative and stratified models of probabilistic processes. Inf. Comput. **121**(1), 59–80 (1995). https://doi.org/10.1006/inco.1995.1123, https://doi.org/10.1006/inco.1995.1123
25. Gros, T.P., Hermanns, H., Hoffmann, J., Klauck, M., Steinmetz, M.: Deep statistical model checking. In: Gotsman, A., Sokolova, A. (eds.) Formal Techniques for Distributed Objects, Components, and Systems - 40th IFIP WG 6.1 International Conference, FORTE 2020, Held as Part of the 15th International Federated Conference on Distributed Computing Techniques, DisCoTec 2020, Valletta, Malta, June 15-19, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12136, pp. 96–114. Springer (2020). https://doi.org/10.1007/978-3-030-50086-3_6, https://doi.org/10.1007/978-3-030-50086-3_6
26. Hoare, C.A.R.: Communicating sequential processes. Commun. ACM **21**(8), 666–677 (Aug 1978). https://doi.org/10.1145/359576.359585, https://doi.org/10.1145/359576.359585
27. Klein, J., Baier, C., Chrszon, P., Daum, M., Dubslaff, C., Klüppelholz, S., Märcker, S., Müller, D.: Advances in probabilistic model checking with PRISM: variable reordering, quantiles and weak deterministic Büchi automata. International Journal on Software Tools for Technology Transfer **20**(2), 179–194 (2018)
28. Köhl, M.A., Hermanns, H., Biewer, S.: Efficient monitoring of real driving emissions. In: Colombo, C., Leucker, M. (eds.) Runtime Verification. pp. 299–315. Springer International Publishing, Cham (2018)
29. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: Verification of probabilistic real-time systems. In: Proceedings Computer Aided Verification (CAV'11). LNCS, vol. 6806, pp. 585–591. Springer (2011)

30. Lee, E.A.: Cyber physical systems: Design challenges. In: 2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC). pp. 363–369 (2008)
31. Lovejoy, W.S.: A survey of algorithmic methods for partially observable Markov decision processes. Annals of Operations Research **28**(1), 47–65 (1991)
32. Lynch, N., Tuttle, M.: An introduction to input/output automata. CWI-Quarterly **2**(3), 219–246 (1989)
33. Madani, O., Hanks, S., Condon, A.: On the undecidability of probabilistic planning and related stochastic optimization problems. Artificial Intelligence **147**(1-2), 5–34 (2003)
34. Milner, R.: Communication and concurrency. PHI Series in computer science, Prentice Hall (1989)
35. Papadimitriou, C., Tsitsiklis, J.: The complexity of Markov decision processes. Mathematics of Operations Research **12**(3), 441–450 (1987)
36. Pineda, L.E., Zilberstein, S.: Planning under uncertainty using reduced models: Revisiting determinization. In: ICAPS (2014)
37. Puterman, M.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons, Inc., New York, NY (1994)
38. Segala, R.: Modeling and verification of randomized distributed real-time systems. Ph.D. thesis, Massachusetts Institute of Technology (1995)
39. Thüm, T., Apel, S., Kästner, C., Schaefer, I., Saake, G.: A classification and survey of analysis strategies for software product lines. ACM Comput. Surv. **47**(1s), 6:1–6:45 (2014)
40. Wu, S., Smolka, S.A., Stark, E.W.: Composition and behaviors of probabilistic I/O automata. Theor. Comput. Sci. **176**(1-2), 1–38 (1997). https://doi.org/10.1016/S0304-3975(97)00056-X, https://doi.org/10.1016/S0304-3975(97)00056-X