When are prime formulae characteristic?*

L. Aceto, D. Della Monica, I. Fábregas, and A. Ingólfsdóttir

ICE-TCS, School of Computer Science, Reykjavik University, Iceland
{luca|dariodm|fabregas|annai}@ru.is

Abstract. In the setting of the modal logic that characterizes modal refinement over modal transition systems, Boudol and Larsen showed that the formulae for which model checking can be reduced to preorder checking, that is, the characteristic formulae, are exactly the consistent and prime ones. This paper presents general, sufficient conditions guaranteeing that characteristic formulae are exactly the consistent and prime ones. It is shown that the given conditions apply to the logics characterizing all the semantics in van Glabbeek's branching-time spectrum.

1 Introduction

Model checking and equivalence/preorder checking are the two main approaches to the computer-aided verification of reactive systems [3, 6]. In model checking, one typically describes the behaviour of a computing system using a state-transition model, such as a labelled transition system [11], and specifications of properties systems should exhibit are expressed using some modal or temporal logic. In this approach, system verification amounts to checking whether a system is a model of the formulae describing a given specification. When using equivalence/preorder checking instead, systems and their specifications are both expressed in the same state-machine-based formalism. In this approach, checking whether a system correctly implements its specification amounts to verifying whether the state machines describing them are related by some suitable notion of behavioural equivalence/preorder. (See [8, 9] for taxonomic studies of the plethora of behavioural relations that have been considered in the field of concurrency theory.)

A bridge between model checking and equivalence/preorder checking is provided by the notion of *characteristic formula* [10, 13]. Intuitively, a characteristic formula provides a complete logical characterization of the behaviour of a process modulo some notion of behavioural equivalence or preorder. At least for finite labelled transition systems, such formulae can be used to reduce equivalence/preorder checking to model checking effectively, and, as argued in [7], this

* Research supported by the project 001-ABEL-CM-2013 within the NILS Science and Sustainability Programme, the Spanish project STRONGSOFT TIN2012-39391-C04-04, and the projects *Nominal SOS* (project nr. 141558-051) and *Decidability and Expressiveness for Interval Temporal Logics* (project nr. 130802-051) of the Icelandic Research Fund.

approach has better complexity than known algorithms for preorder checking. A natural question to ask is for what kinds of logical specifications model checking can be reduced to establishing a behavioural relation between an implementation and a labelled transition system that suitably encodes the specification. To the best of our knowledge, this question was first addressed by Boudol and Larsen, who showed in [5] that, in the context of the modal logic that characterizes modal refinement over modal transition systems, the formulae that are "graphically representable" (that is, the ones that are characteristic for some process) are exactly the consistent and prime ones. (A formula is *prime* if whenever it implies a disjunction of two formulae, it implies one of the disjuncts.) A similar result is given in [2] in the setting of covariant-contravariant simulation. Moreover, each formula in the logics considered in [2, 5] can be "graphically represented" by a (possibly empty) finite set of processes.

To our mind, those are very pleasing results that show the very close connection between logical and behavioural approaches to verification in two specific settings. But, how general are they? Do similar results hold for the plethora of other process semantics and their modal characterizations studied in the literature? And, if so, are there general sufficient conditions guaranteeing that characteristic formulae are exactly the consistent and prime ones? The aim of this article is to provide answers to those questions.

From a methodological perspective, we follow a purely logical approach towards the characterization of process semantics, which allows us to work in an abstract and very general setting (described in Section 2): instead of investigating each behavioural semantics separately, we define a process semantics as the preorder induced by some logic, i.e. a process p is smaller than a process q if the set of logical properties of p is strictly included in that of q. By investigating preorders defined in this way, we can identify common properties for all logically characterized preorders, and thus we are able to give a general recipe to logically characterize processes by means of consistent and prime formulae (*characterization by primality*). The first piece of our characterization by primality result consists in showing that characteristic formulae are always consistent and prime (Theorem 1). This result was already proven for specific semantics [2, 5], and we generalise it here to every logically characterized preorder. The converse is not true in general. Therefore our main technical contribution is to provide sufficiently general conditions guaranteeing that consistent and prime formulae are characteristic formulae for some process.

In Section 3, we introduce the notion of *decomposable logic* and show that, for such logics, consistent and prime formulae are characteristic for some process (Theorem 2). (Intuitively, a logic is decomposable if, for each formula, the set of processes satisfying it includes the set of processes satisfying a characteristic formula and the logic is sufficiently expressive to witness this inclusion.) We then proceed to identify features that make a logic decomposable, thus paving the way to showing the decomposability of a number of logical formalisms (Section 3.1). In particular, we prove that if the set of formulae satisfied by each process can be finitely characterized in a suitable technical sense (see Definition 4), then,

under some mild assumptions, the logic is decomposable (Corollary 2). Moreover, such finitely characterized logics can express the characteristic formula for each process (Proposition 6(ii)).

In order to show the applicability of our general framework, we use it in Sections 4–5 to show that, for a variety of logical characterizations of process semantics, characteristic formulae are exactly the consistent and prime ones. In particular, this applies to all the semantics in van Glabbeek's branching-time spectrum. In all these cases, there is a perfect match between the behavioural and logical view of processes: not only do the logics characterize processes up to the chosen notion of behavioural relation, but processes represent all the consistent and prime formulae in the logics.

Proofs of most of the technical results can be found in [1].

2 Process semantics defined logically

We assume that $\mathcal{L}$ is a language interpreted over a non-empty set P, which we refer to as a set of processes. Thus, $\mathcal{L}$ is equipped with a semantic function $[\![\cdot]\!]_{\mathcal{L}} : \mathcal{L} \to \mathcal{P}(P)$ (where $\mathcal{P}(P)$ denotes the powerset of P), and we say that $p \in P$ satisfies $\phi \in \mathcal{L}$ whenever $p \in [\![\phi]\!]_{\mathcal{L}}$. For all $p, q \in P$, we define the following notions:

- $\mathcal{L}(p) = \{\phi \in \mathcal{L} \mid p \in [\![\phi]\!]_{\mathcal{L}}\}$: the set of formulae in $\mathcal{L}$ that p satisfies; we assume $\mathcal{L}(p) \neq \emptyset$, for each $p \in P$;
- $p^{\uparrow_{\mathcal{L}}} = \{p' \in P \mid \mathcal{L}(p) \subseteq \mathcal{L}(p')\}$: the *upwards closure* of p (with respect to $\mathcal{L}$);
- p and q are *logically equivalent* if $\mathcal{L}(p) = \mathcal{L}(q)$;
- p and q are *incomparable* (with respect to $\mathcal{L}$) iff neither $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ nor $\mathcal{L}(q) \subseteq \mathcal{L}(p)$ holds.

We say that a formula $\phi \in \mathcal{L}$ is *consistent* iff $[\![\phi]\!]_{\mathcal{L}} \neq \emptyset$. Formulae $\phi, \psi \in \mathcal{L}$ are said to be *logically equivalent* (or simply *equivalent*) iff $[\![\phi]\!]_{\mathcal{L}} = [\![\psi]\!]_{\mathcal{L}}$. When it is clear from the context, we omit the logic $\mathcal{L}$ in the subscript (and in the text). For example, we write $[\![\phi]\!]$ and $p^{\uparrow}$ instead of $[\![\phi]\!]_{\mathcal{L}}$ and $p^{\uparrow_{\mathcal{L}}}$. We note that $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ defines a preorder between processes, which we refer to as the *logical preorder* characterized by $\mathcal{L}$. We say that a preorder over P is *logically characterized* or simply *logical* if it is characterized by some logic $\mathcal{L}$.

For a subset $S \subseteq P$ we say that:

- S is *upwards closed* iff $p^{\uparrow} \subseteq S$ for all $p \in S$;
- $p \in S$ is *minimal* in S iff for each $q \in S$, if $\mathcal{L}(q) \subseteq \mathcal{L}(p)$ then $\mathcal{L}(q) = \mathcal{L}(p)$;
- $p \in S$ is a *least element* in S iff $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ for each $q \in S$.

Clearly, if p is a least element in a set S, then p is also minimal in S. Notice that, if a set S contains a least element, then it is the unique minimal element in S, up to logical equivalence.

2.1 Characteristic and prime formulae

We introduce here the crucial notion of *characteristic formula* for a process [3, 10, 13] and the one of *prime formula* [2, 5], in the setting of logical preorders over

processes. Our aim in this study is to investigate when these notions coincide, thus providing a characterization of logically defined processes by means of prime formulae, which sometimes we will refer to as *characterization by primality*. To begin with, in this section we study such a connection in a very general setting. As it turns out, for logically characterized preorders, the property of being characteristic always implies primality (Theorem 1). The main focus of this paper becomes therefore to investigate under what conditions a consistent and prime formula is characteristic for some process in a logical preorder (Section 3).

Definition 1 (Characteristic formula). *A formula $\phi \in \mathcal{L}$ is* characteristic *for $p \in P$ iff for all $q \in P$ it holds that $q \in [\![\phi]\!]$ if and only if $\mathcal{L}(p) \subseteq \mathcal{L}(q)$.*

The following simple properties related to characteristic formulae will be useful in what follows.

Proposition 1. *The following properties hold for all $p, q \in P$ and $\phi \in \mathcal{L}$:*
- *(i) ϕ is characteristic for p if and only if $[\![\phi]\!] = p^{\uparrow}$;*
- *(ii) a characteristic formula for p, if it exists, is unique up to logical equivalence (and can therefore be referred to as $\chi(p)$);*
- *(iii) if the characteristic formulae for p and q, namely $\chi(p)$ and $\chi(q)$, exist then $[\![\chi(p)]\!] \subseteq [\![\chi(q)]\!]$ if and only if $\mathcal{L}(q) \subseteq \mathcal{L}(p)$.*

Next we state two useful properties.

Proposition 2. *The following properties hold: (i) for each $\phi \in \mathcal{L}$, $[\![\phi]\!]$ is upwards closed, and (ii) if $p \in [\![\phi]\!] \subseteq [\![\chi(p)]\!]$, then $[\![\phi]\!] = [\![\chi(p)]\!]$.*

We now define what it means for a formula to be prime.

Definition 2 (Prime formula). *We say that $\phi \in \mathcal{L}$ is* prime *iff for each non-empty, finite subset of formulae $\Psi \subseteq \mathcal{L}$ it holds that $[\![\phi]\!] \subseteq \bigcup_{\psi \in \Psi} [\![\psi]\!]$ implies $[\![\phi]\!] \subseteq [\![\psi]\!]$ for some $\psi \in \Psi$.*

Observe that our definition is a semantic version of the one given in [5]. This serves our purpose to keep the discussion as abstract as possible. In this perspective, we want to abstract (at least at this point of the discussion) from the syntactic details of the logical formalism, while the classic definition tacitly applies only to languages that feature at least the Boolean connective $\vee$.

We provide here the first piece of our characterization by primality, by showing that the property of being characteristic implies primality without any extra assumption on the language $\mathcal{L}$ or its interpretation.

Theorem 1. *Let $\phi \in \mathcal{L}$. If ϕ is a characteristic formula for some $p \in P$, then ϕ is prime and consistent.*

Proof. The formula ϕ is obviously consistent because $p \in [\![\chi(p)]\!] = [\![\phi]\!]$. Towards proving that $\chi(p)$ is prime, we assume that $[\![\chi(p)]\!] \subseteq \bigcup_{i \in I} [\![\psi_i]\!]$, where I is finite and non-empty. By our assumption, since $p \in [\![\chi(p)]\!]$, then for some $i \in I$, $p \in [\![\psi_i]\!]$ holds. As, by Proposition 2(i), $[\![\psi_i]\!]$ is upwards closed, using Proposition 1(i) we can conclude that $[\![\chi(p)]\!] = p^{\uparrow} \subseteq [\![\psi_i]\!]$ as we wanted to prove. □

Notice that the converse is not true in general, that is, there exist formulae that are consistent and prime but not characteristic. To see this, let $P = \mathbb{Q}$, $\mathcal{L} = \mathbb{R}$ and $[\![\phi]\!] = \{p \in \mathbb{Q} \mid \phi \leq p\}$. Clearly, all formulae are consistent. Then, $\mathcal{L}(p) = \{\phi \in \mathbb{R} \mid \phi \leq p\}$ which implies that $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ iff $p \leq q$ iff $q \in [\![p]\!]$. This means that, for each $p \in \mathbb{Q}$, $\phi = p$ is characteristic for p and therefore the characteristic formula is well-defined for all $p \in P$. Furthermore $[\![\phi]\!] \cup [\![\psi]\!] = \{p \in \mathbb{Q} \mid \min\{\phi, \psi\} \leq p\}$ for all $\phi, \psi \in \mathcal{L}$, which implies that all formulae are prime. On the other hand $\phi = \sqrt{2} \notin \mathbb{Q}$ cannot be characteristic for any process as $[\![\sqrt{2}]\!]$ does not have a least element.

3 Characterization by primality for logical preorders

In this section we introduce sufficient conditions under which the converse of Theorem 1 is also true for logical preorders, that is, conditions guaranteeing that every consistent, prime formula is characteristic.

As a first step, we introduce the notion of *decomposable* logic. We show that if a logic is decomposable, then we have a logical characterization of processes by primality. Some of the results involve the Boolean connectives $\wedge$ and $\vee$, whose intended semantics is the standard one.

Definition 3 (Decomposability). *We say that a formula $\phi \in \mathcal{L}$ is* decomposable *iff $[\![\phi]\!] = [\![\chi(p)]\!] \cup [\![\psi_p]\!]$ for some $p \in P$ and $\psi_p \in \mathcal{L}$, with $p \notin [\![\psi_p]\!]$. We say that $\mathcal{L}$ is* decomposable *iff all consistent formulae $\phi \in \mathcal{L}$ are either decomposable or characteristic for some $p \in P$.*

The following theorem allows us to reduce the problem of relating the notions of prime and characteristic formulae in a given logic to the problem of establishing the decomposability property for that logic. This provides us with a very general setting towards characterization by primality.

Theorem 2. *If $\mathcal{L}$ is decomposable then every formula that is consistent and prime is also characteristic for some $p \in P$.*

3.1 Paths to decomposability

The aim of this section is to identify features that make a logic decomposable, thus paving the way towards showing the decomposability of a number of logical formalisms in the next sections. First, we observe that if a characteristic formula $\chi(p)$ exists for every $p \in P$, then what we are left to do is to define, for each $\phi \in \mathcal{L}$, a formula ψ_p, for some $p \in P$, with the properties mentioned in Definition 3, as captured by the following proposition.

Proposition 3. *Let $\mathcal{L}$ be a logic such that (i) $\chi(p)$ exists for each $p \in P$, and (ii) for each consistent formula ϕ there exist $p \in [\![\phi]\!]$ and $\psi_p \in \mathcal{L}$ such that $p \notin [\![\psi_p]\!]$ and $[\![\phi]\!] \setminus [\![\chi(p)]\!] \subseteq [\![\psi_p]\!] \subseteq [\![\phi]\!]$. Then $\mathcal{L}$ is decomposable.*

Clearly, when dealing with formalisms featuring at least the Boolean operators $\neg$ and $\wedge$, as it is the case with the logic for the bisimulation semantics in Section 5, such a formula ψ_p is easily defined as $\neg\chi(p) \wedge \phi$. This is stated in the following corollary.

Corollary 1. *Let $\mathcal{L}$ be a logic that features at least the Boolean connective $\wedge$ and such that, for each $p \in P$, the formula $\chi(p)$ exists and there is some formula $\bar{\chi}(p) \in \mathcal{L}$ where $[\![\bar{\chi}(p)]\!] = [\![\chi(p)]\!]^c$ (the complement of $[\![\chi(p)]\!]$). Then $\mathcal{L}$ is decomposable.*

The situation is more complicated when it comes to the other logics for the semantics in the branching-time spectrum (which we consider in Section 5) as negation is in general not expressible in these logics, not even for characteristic formulae. Therefore, instead we will prove a slightly stronger statement than the one in Corollary 1 by identifying a weaker condition than the existence of a negation of the characteristic formulae (that we assume to exist) that also leads to decomposability of the logic. This is described in the following proposition.

Proposition 4. *Let $\phi \in \mathcal{L}$, p be a minimal element in $[\![\phi]\!]$ such that $\chi(p)$ exists in $\mathcal{L}$, and let $\bar{\chi}(p)$ be a formula in $\mathcal{L}$ such that $\{q \in P \mid \mathcal{L}(q) \not\subseteq \mathcal{L}(p)\} \subseteq [\![\bar{\chi}(p)]\!]$. Then, $[\![\phi]\!] \setminus [\![\chi(p)]\!] \subseteq [\![\bar{\chi}(p)]\!]$ holds.*

In the next proposition, we build on the above result, and establish some conditions, which are met by the logics we consider in Section 5 (apart for the one for bisimulation semantics), and which immediately lead to decomposability.

Proposition 5. *Let $\mathcal{L}$ be a logic that features at least the Boolean connective $\wedge$ and such that:*

(i) $\chi(p)$ exists for each $p \in P$,
(ii) for each consistent ϕ, the set $[\![\phi]\!]$ has a minimal element, and
(iii) for each $p \in P$, there exists a formula $\bar{\chi}(p)$ such that $p \notin [\![\bar{\chi}(p)]\!]$ and $\{q \in P \mid \mathcal{L}(q) \not\subseteq \mathcal{L}(p)\} \subseteq [\![\bar{\chi}(p)]\!]$.

Then, $\mathcal{L}$ is decomposable.

In order to apply the above result to prove decomposability for a logic $\mathcal{L}$, we now develop a general framework ensuring conditions (i) and (ii) in Proposition 5. To this end, we exhibit a finite characterization of the (possibly) infinite set $\mathcal{L}(p)$ of true facts associated with every $p \in P$. (In order to ensure condition (iii) of the proposition, we will actually construct the formula $\bar{\chi}(p)$ in each of the languages considered in Section 5.)

Definition 4 (Characterization). *We say that the logic $\mathcal{L}$ is* characterized *by a function $\mathcal{B} : P \to \mathcal{P}(\mathcal{L})$ iff for each $p \in P$, $\mathcal{B}(p) \subseteq \mathcal{L}(p)$ and for each $\phi \in \mathcal{L}(p)$ there exists a non-empty $\Psi \subseteq \mathcal{B}(p)$ such that $\bigcap_{\psi \in \Psi} [\![\psi]\!] \subseteq [\![\phi]\!]$. We say that $\mathcal{L}$ is* finitely characterized *by $\mathcal{B}$ iff $\mathcal{L}$ is characterized by a function $\mathcal{B}$ such that $\mathcal{B}(p)$ is finite for each $p \in P$. Finally, we say that $\mathcal{B}$ is monotonic iff $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ implies $\mathcal{B}(p) \subseteq \mathcal{B}(q)$ for all $p, q \in P$.*

In what follows, we show that if a logic $\mathcal{L}$ features at least the Boolean connective $\wedge$ and it is finitely characterized by $\mathcal{B}$, for some monotonic $\mathcal{B}$, then it fulfils conditions (i) and (ii) in Proposition 5.

Proposition 6. *The following statements hold.*

(*i*) *If $\mathcal{L}$ is characterized by $\mathcal{B}$, then for each $p, q \in P$, $\mathcal{B}(p) \subseteq \mathcal{B}(q)$ implies $\mathcal{L}(p) \subseteq \mathcal{L}(q)$.*

(*ii*) *If $\mathcal{L}$ features at least the Boolean connective $\wedge$ and is finitely characterized by $\mathcal{B}$, then each $p \in P$ has a characteristic formula in $\mathcal{L}$ given by $\chi(p) = \bigwedge_{\phi \in \mathcal{B}(p)} \phi$.*

(*iii*) *If $\mathcal{L}$ is finitely characterized by $\mathcal{B}$, for some monotonic $\mathcal{B}$, then for each consistent $\phi \in \mathcal{L}$, the set $[\![\phi]\!]$ has a minimal element.*

It is worth pointing out that the Boolean connective $\wedge$ plays a minor role in (the proof of) Proposition 6(ii). Indeed, it is applied to formulae in $\mathcal{B}(p)$ only. Thus, such a result can be used also to deal with logics that allow for a limited use of such a connective, such as the logics for trace equivalence and other linear-time semantics [9].

Finally, we can summarize the results in this section in the following corollary.

Corollary 2. *Let $\mathcal{L}$ be a logic that features at least the Boolean connective $\wedge$ and such that:*

(*i*) *$\mathcal{L}$ is finitely characterized by $\mathcal{B}$, for some monotonic $\mathcal{B}$, and*

(*ii*) *for each $\chi(p)$, there exists a formula $\bar{\chi}(p)$ such that either*
- *$[\![\bar{\chi}(p)]\!] = [\![\chi(p)]\!]^c$, or*
- *$p \notin [\![\bar{\chi}(p)]\!]$ and $\{q \in P \mid \mathcal{L}(q) \not\subseteq \mathcal{L}(p)\} \subseteq [\![\bar{\chi}(p)]\!]$.*

Then, $\mathcal{L}$ is decomposable.

In the remainder of the paper, we will present some applications of our general results.

4 Application to finitely many processes

As a first application, we investigate the case when the set P is finite and the logic $\mathcal{L}$ features at least the Boolean connectives $\wedge$ and $\vee$. Note that although P itself is finite, it can contain processes with infinite behaviours, e.g., when $p \in P$ represents a labelled transition system with loops. If P is finite, so is $\mathcal{L}$, up to logical equivalence. Let $\mathcal{L}^{fin}$ be a set of representatives of the equivalence classes of $\mathcal{L}$ modulo logical equivalence, and define $\mathcal{B}^{fin}(p) = \mathcal{L}^{fin}(p) = \mathcal{L}(p) \cap \mathcal{L}^{fin}$, for each $p \in P$. It is easy to see that $\mathcal{L}$ is finitely characterized by $\mathcal{B}^{fin}$, according to Definition 4. Moreover, $\mathcal{B}^{fin}$ is clearly monotonic. Thus, by Proposition 6(ii), $\chi(p)$ is well-defined for each p as $\bigwedge_{\psi \in \mathcal{B}^{fin}(p)} \psi$.

In order to show that $\mathcal{L}$ is decomposable, let us consider a consistent formula $\phi \in \mathcal{L}^{fin}$, and let p be minimal in $[\![\phi]\!]$ (the existence of such a p is guaranteed by the finiteness of P). Now, either $[\![\phi]\!] = [\![\chi(p)]\!]$ (in this case we are done), or $[\![\phi]\!] \setminus [\![\chi(p)]\!] \neq \emptyset$, and thus, the set $S = \{q \in P \mid q \in [\![\phi]\!], \mathcal{L}^{fin}(p) \neq \mathcal{L}^{fin}(q)\}$ is not empty. In this second case it is easy to see that $\psi_p = \bigvee_{q \in S} \chi(q)$ fulfils the requirements of Definition 3. This can be summarized in the following theorem.

Semantic relation	Definition
simulation (S)	$p \lesssim_{\mathsf{S}} q \Leftrightarrow$ for all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ such that $p' \lesssim_{\mathsf{S}} q'$;
complete simulation (CS)	$p \lesssim_{\mathsf{CS}} q \Leftrightarrow$ for all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ such that $p' \lesssim_{\mathsf{CS}} q'$, and $I(p) = \emptyset$ iff $I(q) = \emptyset$;
ready simulation (RS)	$p \lesssim_{\mathsf{RS}} q \Leftrightarrow$ for all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ such that $p' \lesssim_{\mathsf{RS}} q'$, and $I(p) = I(q)$;
trace simulation (TS)	$p \lesssim_{\mathsf{TS}} q \Leftrightarrow$ for all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ such that $p' \lesssim_{\mathsf{TS}} q'$, and $traces(p) = traces(q)$;
2-nested simulation ($\mathsf{2S}$)	$p \lesssim_{\mathsf{2S}} q \Leftrightarrow$ for all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ such that $p' \lesssim_{\mathsf{2S}} q'$, and $q \lesssim_{\mathsf{S}} p$;
bisimulation (BS)	$p \lesssim_{\mathsf{BS}} q \Leftrightarrow$ for all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ such that $p' \lesssim_{\mathsf{BS}} q'$, and for all $q \xrightarrow{a} q'$ there exists $p \xrightarrow{a} p'$ such that $p' \lesssim_{\mathsf{BS}} q'$.

Table 1. Semantic relations in van Glabbeek's branching-time spectrum.

Theorem 3 (Characterization by primality). *Let $\mathcal{L}$ be a logic interpreted over a finite set P that features at least the Boolean connectives $\wedge$ and $\vee$. Then, each formula $\phi \in \mathcal{L}$ is consistent and prime if and only if ϕ is characteristic for some $p \in P$.*

5 Application to semantics in van Glabbeek's spectrum

Our next task is to apply the result described in Corollary 2 to the semantics in the branching-time spectrum, over finite trees and with finite set of actions. All those semantics have been shown to be characterized by specific logics and therefore inherit all the properties of logically defined preorders. We reason about characterization by primality (Theorem 5) by showing that each logic is finitely characterized by some monotonic $\mathcal{B}$, and by building, for each characteristic formula $\chi(p)$, a formula $\bar{\chi}(p)$ with the properties specified in Proposition 5(iii).

The logics we focus on are the ones for the semantics in *van Glabbeek's branching-time spectrum* [8, 9], namely *simulation* (S), *complete simulation* (CS), *ready simulation* (RS), *trace simulation* (TS), *2-nested simulation* ($\mathsf{2S}$), and *bisimulation* (BS). Their syntax and semantics are briefly described in what follows. For a comprehensive overview, we refer the reader to the corresponding literature. In the rest of this section $\mathsf{spectrum}$ denotes the set $\{\mathsf{S}, \mathsf{CS}, \mathsf{RS}, \mathsf{TS}, \mathsf{2S}, \mathsf{BS}\}$ and we let $X \in \mathsf{spectrum}$.

Syntax for processes. The set of processes P over a finite set of actions Act is given by the following grammar:

$$p ::= \mathsf{0} \mid ap \mid p + p,$$

where $a \in Act$. Given a process p, we say that p can perform the action a and evolve into p', denoted $p \xrightarrow{a} p'$, iff (i) $p = ap'$ or (ii) $p = p_1 + p_2$ and either $p_1 \xrightarrow{a} p'$ or $p_2 \xrightarrow{a} p'$ holds. Note that every process denotes a finite loop-free labelled transition system.

We define the set of *initials* of p, denoted $I(p)$, as the set $\{a \in Act \mid p \xrightarrow{a} p' \text{ for some } p' \in P\}$. We write $p \xrightarrow{a}$ if $a \in I(p)$, and we write $p \not\xrightarrow{a}$ if $a \notin I(p)$. We define $traces(p)$ as follows:

$traces(p) = \{\varepsilon\} \cup \{a\tau \mid \exists a \in Act\ \exists p' \in P\ .\ p \xrightarrow{a} p' \text{ and } \tau \in traces(p')\}$.

For each trace $\tau = a_1 \dots a_n$, we write $p \xrightarrow{\tau} p'$ for $p \xrightarrow{a_1} p_1 \xrightarrow{a_2} p_2 \dots p_{n-1} \xrightarrow{a_n} p'$. Finally, for each $p \in P$, $dep(p)$ is the length of the longest trace in $traces(p)$.

Behavioural preorders. The semantics of processes is expressed by preorder relations, which, intuitively, classify processes according to their possible behaviours. Roughly speaking, a process *follows* another in the preorder (or it is *above* it) if it exhibits at least the same behaviours as the latter. The semantic relations in van Glabbeek's branching-time spectrum are defined as follows.

Definition 5. *For each $p, q \in P$ and each $X \in \mathsf{spectrum}$, $\lesssim_X$ is the largest relation satisfying the corresponding condition in Table 1.*

It is well-known that $\lesssim_{\mathsf{BS}} \subsetneq \lesssim_{\mathsf{2S}} \subsetneq \lesssim_{\mathsf{TS}} \subsetneq \lesssim_{\mathsf{RS}} \subsetneq \lesssim_{\mathsf{CS}} \subsetneq \lesssim_{\mathsf{S}}$ [8, 9].

Syntax for logics. Table 2 provides the definition of the syntax of the logics that capture exactly the above mentioned process semantic relations. We treat formulae of the form $\mathbf{0}$ and $[a]\psi$ as syntactic shorthand for $\bigwedge_{a \in Act}[a]\mathbf{ff}$ and $\neg\langle a\rangle\neg\psi$, respectively. The languages of the different logics yield the following chain of strict inclusions: $\mathcal{L}_{\mathsf{S}} \subsetneq \mathcal{L}_{\mathsf{CS}} \subsetneq \mathcal{L}_{\mathsf{RS}} \subsetneq \mathcal{L}_{\mathsf{TS}} \subsetneq \mathcal{L}_{\mathsf{2S}} \subsetneq \mathcal{L}_{\mathsf{BS}}$, corresponding to formalisms with strictly increasing expressive power. Notice that, as it will become clear after the definition of the satisfaction relation below, some of the languages present some redundancy, in the sense that they could be replaced with smaller ones, without any loss in expressiveness. For instance, a disjunction is expressible in $\mathcal{L}_{\mathsf{BS}}$ using conjunction and negation, and suitably replacing $\mathbf{tt}$ with $\mathbf{ff}$ and vice versa. We followed this approach because we find it helpful to have syntactically larger languages corresponding to more expressive semantics.

Roughly speaking, each language consists of an "existential" and a "universal" sub-language, as highlighted by the definitions in the second and the fourth column of Table 2 ($\phi_X ::= \phi_X^{\exists} \mid \phi_X^{\forall}$ for each $X \in \mathsf{spectrum}$ apart from simulation). The "existential" sub-language (formulae derivable from the non-terminal $\phi_X^{\exists}$) is common to all the logics and so is its definition (bottom line of Table 2). The "universal" sub-language (formulae derivable from the non-terminal $\phi_X^{\forall}$) is what actually distinguishes the several languages: its definition is provided for each logic in the corresponding row of Table 2 (see [1] for further explanations and expanded definitions).

Satisfaction relation. We give here the semantics of the logics, by describing the satisfaction relation for the most expressive one, namely $\mathcal{L}_{\mathsf{BS}}$, corresponding to bisimulation semantics. The semantics for the other logics can be obtained by considering the corresponding subset of clauses.

- $p \in [\![\mathbf{tt}]\!]$ and $p \notin [\![\mathbf{ff}]\!]$, for every $p \in P$,
- $p \in [\![\phi_1 \wedge \phi_2]\!]$ iff $p \in [\![\phi_1]\!]$ and $p \in [\![\phi_2]\!]$,
- $p \in [\![\phi_1 \vee \phi_2]\!]$ iff $p \in [\![\phi_1]\!]$ or $p \in [\![\phi_2]\!]$,
- $p \in [\![\langle a\rangle\phi]\!]$ iff $p' \in [\![\phi]\!]$ for some $p' \in P$ such that $p \xrightarrow{a} p'$,
- $p \in [\![\neg\phi]\!]$ iff $p \notin [\![\phi]\!]$.

Semantics	Syntax	Semantics	Syntax
S	$\phi_{\mathsf{S}} ::= \phi^{\exists}_{\mathsf{S}}$	TS	$\phi_{\mathsf{TS}} ::= \phi^{\exists}_{\mathsf{TS}} \mid \phi^{\forall}_{\mathsf{TS}}$ $\phi^{\forall}_{\mathsf{TS}} ::= \mathbf{ff} \mid [a]\phi^{\forall}_{\mathsf{TS}}$
CS	$\phi_{\mathsf{CS}} ::= \phi^{\exists}_{\mathsf{CS}} \mid \phi^{\forall}_{\mathsf{CS}}$ $\phi^{\forall}_{\mathsf{CS}} ::= \mathbf{0}$	2S	$\phi_{\mathsf{2S}} ::= \phi^{\exists}_{\mathsf{2S}} \mid \phi^{\forall}_{\mathsf{2S}}$ $\phi^{\forall}_{\mathsf{2S}} ::= \neg\phi_{\mathsf{S}}$
RS	$\phi_{\mathsf{RS}} ::= \phi^{\exists}_{\mathsf{RS}} \mid \phi^{\forall}_{\mathsf{RS}}$ $\phi^{\forall}_{\mathsf{RS}} ::= [a]\mathbf{ff}$	BS	$\phi_{\mathsf{BS}} ::= \phi^{\exists}_{\mathsf{BS}} \mid \phi^{\forall}_{\mathsf{BS}}$ $\phi^{\forall}_{\mathsf{BS}} ::= \neg\phi_{\mathsf{BS}}$
$\phi^{\exists}_X ::= \mathbf{tt} \mid \mathbf{ff} \mid \phi_X \wedge \phi_X \mid \phi_X \vee \phi_X \mid \langle a\rangle\phi_X \qquad \forall X \in \{\mathsf{S}, \mathsf{CS}, \mathsf{RS}, \mathsf{TS}, \mathsf{2S}, \mathsf{BS}\}$			

Table 2. Syntax of the logics in van Glabbeek's branching-time spectrum. For every $X \in \{\mathsf{S}, \mathsf{CS}, \mathsf{RS}, \mathsf{TS}, \mathsf{2S}, \mathsf{BS}\}$, the language of $\mathcal{L}_X$ is generated by the grammar rooted in the non-terminal ϕ_X.

The following well-known theorem states the relationship between logics and process semantics that allows us to use our general results about logically characterized semantics.

Theorem 4 (Logical characterization [8, 9]). *For each $X \in$ spectrum and for all $p, q \in P$, $p \lesssim_X q$ iff $\mathcal{L}_X(p) \subseteq \mathcal{L}_X(q)$.*

We observe that all the logics we consider feature the Boolean connective $\wedge$, as required by one of the assumptions of Corollary 2. In what follows, we show that every logic meets also the other conditions of the corollary, that is, it is finitely characterized by some monotonic $\mathcal{B}$, and for each $\chi(p)$ there exists a formula $\bar{\chi}(p)$ such that $p \notin [\![\bar{\chi}(p)]\!]$ and $\{q \in P \mid \mathcal{L}(q) \not\subseteq \mathcal{L}(p)\} \subseteq [\![\bar{\chi}(p)]\!]$. We provide here proof details for the illustrative case of *ready simulation* (RS) [4, 9, 12] only, and refer the interested reader to [1] for further details. We recall the "expanded" syntax of the corresponding logic $\mathcal{L}_{\mathsf{RS}}$:

$$\phi_{\mathsf{RS}} ::= \mathbf{tt} \mid \mathbf{ff} \mid \phi_{\mathsf{RS}} \wedge \phi_{\mathsf{RS}} \mid \phi_{\mathsf{RS}} \vee \phi_{\mathsf{RS}} \mid \langle a\rangle\phi_{\mathsf{RS}} \mid [a]\mathbf{ff}.$$

5.1 Finite characterization

We prove here that logics in van Glabbeek's branching-time spectrum are finitely characterized by some monotonic $\mathcal{B}$ (condition (i) in Corollary 2).

Lemma 1. *Let $X \in$ spectrum. $\mathcal{L}_X$ is finitely characterized by $\mathcal{B}$, for some monotonic $\mathcal{B}$.*

Proof. We detail the case of ready simulation only. For this relation, the function $\mathcal{B}$ is defined as $\mathcal{B}^+(p) \cup \mathcal{B}^-(p)$, where

- $\mathcal{B}^+(p) = \{\mathbf{tt}\} \cup \{\langle a\rangle\varphi \mid \varphi = \bigwedge_{\psi\in\Psi} \psi, \Psi \subseteq \mathcal{B}(p') \text{ and } p \xrightarrow{a} p'\}$, and
- $\mathcal{B}^-(p) = \{[a]\mathbf{ff} \mid p \in [\![[a]\mathbf{ff}]\!], a \in Act\}$.

We have to show that, for each $p \in P$, (i) $\mathcal{B}(p) \subseteq \mathcal{L}(p)$, (ii) $\mathcal{B}(p)$ is finite, (iii) for each $\phi \in \mathcal{L}(p)$ there exists a non-empty Ψ such that $\Psi \subseteq \mathcal{B}(p)$ and $\bigcap_{\psi\in\Psi} [\![\psi]\!] \subseteq [\![\phi]\!]$, and (iv) for each $q \in P$, if $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ then $\mathcal{B}(p) \subseteq \mathcal{B}(q)$.

Here we only deal with properties (iii) and (iv). The proof of property (*iii*) is by induction on the structure of formulae. The cases $\phi = \mathbf{tt}$, $\phi = [a]\mathbf{ff}$, $\phi = \varphi_1 \vee \varphi_2$, and $\phi = \varphi_1 \wedge \varphi_2$ are simple, and are omitted. Assume that $\phi = \langle a \rangle \varphi$. By definition we have that $\varphi \in \mathcal{L}(p')$ for some $p \xrightarrow{a} p'$. By the inductive hypothesis, there exist formulae $\psi_1, \ldots, \psi_n \in \mathcal{B}(p')$ (with $n \geq 1$) such that $\bigcap_{i \in \{1,\ldots n\}} [\![\psi_i]\!] \subseteq [\![\varphi]\!]$. We define $\psi = \langle a \rangle \bigwedge_i \psi_i$. Clearly, ψ belongs to $\mathcal{B}^+(p)$ (by construction) and $[\![\psi]\!] \subseteq [\![\phi]\!]$ (because $\bigcap_{i \in \{1,\ldots n\}} [\![\psi_i]\!] \subseteq [\![\varphi]\!]$).

Finally, we show that $\mathcal{B}(p)$ is monotonic (property (*iv*)). Consider $p, q \in P$, with $\mathcal{L}(p) \subseteq \mathcal{L}(q)$. We want to show that $\phi \in \mathcal{B}(p)$ implies $\phi \in \mathcal{B}(q)$, for each ϕ. Firstly, we observe that, by $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ and Theorem 4, $p \lesssim_{\mathsf{RS}} q$ holds. Thus, we have that $I(p) = I(q)$ and, for each $a \in Act$ and $p' \in P$ with $p \xrightarrow{a} p'$, there exists $q' \in P$ such that $q \xrightarrow{a} q'$, and $p' \lesssim_{\mathsf{RS}} q'$. Since $I(p) = I(q)$, clearly $\mathcal{B}^-(p) = \mathcal{B}^-(q)$. In order to show that $\mathcal{B}^+(p) \subseteq \mathcal{B}^+(q)$, we proceed by induction on the depth of p. If $I(p) = \emptyset$, then $I(q) = \emptyset$ as well. Thus, we have that $\mathcal{B}^+(p) = \mathcal{B}^+(q) = \{\mathbf{tt}\}$, and the thesis follows. Otherwise ($I(p) \neq \emptyset$), let us consider a formula $\phi = \langle a \rangle \varphi \in \mathcal{B}^+(p)$ (the case when $\psi = \mathbf{tt}$ is trivial). By definition of $\mathcal{B}^+$, there exist $p' \in P$, with $p \xrightarrow{a} p'$, and $\Psi \subseteq \mathcal{B}(p')$ such that $\varphi = \bigwedge_{\psi \in \Psi} \psi$. This implies the existence of $q' \in P$ such that $q \xrightarrow{a} q'$ and $p' \lesssim_{\mathsf{RS}} q'$ (and therefore $\mathcal{L}(p') \subseteq \mathcal{L}(q')$). By the inductive hypothesis, $\mathcal{B}(p') \subseteq \mathcal{B}(q')$ holds as well, which means that $\Psi \subseteq \mathcal{B}(q')$. Hence, we have that $\langle a \rangle \varphi \in \mathcal{B}^+(q)$. □

5.2 Existence of $\bar{\chi}(\cdot)$

In what follows, we show that it is possible to build, for each $\chi(p)$, a formula $\bar{\chi}(p)$, with the properties described in Corollary 2(ii).

Lemma 2. *Let $X \in$ spectrum. For each $\chi(p) \in \mathcal{L}_X$ there exists a formula in $\mathcal{L}_X$, denoted $\bar{\chi}(p)$, such that (i) $p \notin [\![\bar{\chi}(p)]\!]$ and (ii) $\{p' \in P \mid p' \not\lesssim p\} \subseteq [\![\bar{\chi}(p)]\!]$.*

Proof (sketch). We deal with the case of ready simulation only. The formula $\bar{\chi}(p)$ is defined as follows: $\bar{\chi}(p) = \bigvee_{a \notin I(p)} \langle a \rangle \mathbf{tt} \vee \bigvee_{a \in I(p)} [a]\mathbf{ff} \vee \bigvee_{a \in I(p)} \langle a \rangle \bigwedge_{p \xrightarrow{a} p'} \bar{\chi}(p')$. Both properties can be easily proved by induction on the depth of p. □

Finally, the following theorem states our main result of this section.

Theorem 5 (Characterization by primality). *Let $X \in$ spectrum and let $\phi \in \mathcal{L}_X$. Then, ϕ is consistent and prime if and only if ϕ is characteristic for some $p \in P$.*

6 Conclusions

In this paper, we have provided general sufficient conditions guaranteeing that formulae for which model checking can be reduced to equivalence/preorder checking are exactly the consistent and prime ones. We have applied our framework to show that characteristic formulae are exactly the consistent and prime ones when the set of processes is finite, as well as for all the semantics in van Glabbeek's

branching-time spectrum. Our results indicate that the "characterization by primality result" first proved by Boudol and Larsen [5] in the context of the modal logic that characterizes modal refinement over modal transition systems holds in a wide variety of settings in concurrency theory. We feel, therefore, that this study reinforces the view that there is a very close connection between the behavioural and logical view of processes: not only do the logics characterize processes up to the chosen notion of behavioural relation, but processes characterize all the prime and consistent formulae.

In this paper, we have presented applications of our general framework to branching-time semantics. However, ongoing, preliminary investigations indicate that our framework can also be applied to obtain characterization by primality results for the logics for the linear-time semantics in van Glabbeek's spectrum [9]. By way of example, we mention here that we have already obtained such characterizations for trace, complete trace, failures, readiness, possible futures, and impossible futures semantics. We leave the study of further applications and of possible generalizations of our results for future work.

References

1. Aceto, L., Della Monica, D., Fábregas, I., Ingólfsdóttir, A.: When are prime formulae characteristic? (extended version), http://www.icetcs.ru.is/dario/techrep/lc15.pdf
2. Aceto, L., Fábregas, I., de Frutos Escrig, D., Ingólfsdóttir, A., Palomino, M.: Graphical representation of covariant-contravariant modal formulas. In: Proc. of the 18th EXPRESS. EPTCS, vol. 64, pp. 1–15 (2011)
3. Aceto, L., Ingólfsdóttir, A., Larsen, K.G., Srba, J.: Reactive Systems: Modelling, Specification and Verification. Cambridge University Press (2007)
4. Bloom, B., Istrail, S., Meyer, A.R.: Bisimulation can't be traced. Journal of the ACM 42(1), 232–268 (1995)
5. Boudol, G., Larsen, K.G.: Graphical versus logical specifications. Theoretical Computer Science 106(1), 3–20 (1992)
6. Clarke, E.M., Grumberg, O., Peled, D.A.: Model Checking. MIT Press (1999)
7. Cleaveland, R., Steffen, B.: Computing behavioural relations, logically. In: Proc. of the 18th ICALP. LNCS, vol. 510, pp. 127–138. Springer (1991)
8. de Frutos-Escrig, D., Gregorio-Rodríguez, C., Palomino, M., Romero-Hernández, D.: Unifying the linear time-branching time spectrum of process semantics. Logical Methods in Computer Science 9(2) (2013)
9. van Glabbeek, R.J.: The linear time-branching time spectrum I: The semantics of concrete, sequential processes. In: Handbook of Process Algebra, pp. 3–99. Elsevier (2001)
10. Graf, S., Sifakis, J.: A modal characterization of observational congruence on finite terms of CCS. Information and Control 68(1-3), 125–145 (1986)
11. Keller, R.M.: Formal verification of parallel programs. Commun. ACM 19(7), 371–384 (1976)
12. Larsen, K.G., Skou, A.: Bisimulation through probabilistic testing. Information and Computation 94(1), 1–28 (1991)
13. Steffen, B., Ingólfsdóttir, A.: Characteristic formulae for processes with divergence. Information and Computation 110(1), 149–163 (1994)