Towards supervisory control of interactive Markov chains: controllability

Markovski, J.

Published: 01/01/2011

Document Version
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the author’s version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
SE Report: Nr. 2011-01

Towards Supervisory Control of Interactive Markov Chains:
Controllability

J. Markovski
Abstract

We propose a model-based systems engineering framework for supervisory control of stochastic discrete-event systems with unrestricted nondeterminism. We intend to develop the proposed framework in four phases outlined in this paper. Here, we study in detail the first step which comprises investigation of the underlying model and development of a corresponding notion of controllability. The model of choice is termed Interactive Markov Chains, which is a natural semantic model for stochastic variants of process calculi and Petri nets, and it requires a process-theoretic treatment of supervisory control theory. To this end, we define a new behavioral preorder, termed Markovian partial bisimulation, that captures the notion of controllability while preserving correct stochastic behavior. We provide a sound and ground-complete axiomatic characterization of the preorder and, based on it, we define two notion of controllability. The first notion conforms to the traditional way of reasoning about supervision and control requirements, whereas in the second proposal we abstract from the stochastic behavior of the system. For the latter, we intend to separate the concerns regarding synthesis of an optimal supervisor. The control requirements cater only for controllability, whereas we ensure that the stochastic behavior of the supervised plant meets the performance specification by extracting directive optimal supervisors.
1 Introduction

Development costs for control software rise due to the ever-increasing complexity of the machines and demands for better quality, performance, safety, and ease of use. Traditionally, the control requirements are formulated informally and manually translated into control software, followed by validation and rewriting of the code whenever necessary. Such iterative process is time-consuming as the requirements are often ambiguous. This issue gave rise to supervisory control theory [8,28], where high-level supervisory controllers are synthesized automatically based on formal models of hardware and control requirements.

The supervisory controller observes machine behavior by receiving signals from ongoing activities, upon which it sends back control signals about allowed activities. Assuming that the controller reacts sufficiently fast on machine input, this feedback loop is modeled as a pair of synchronizing processes [8,28]. The model of the machine, referred to as plant, is restricted by synchronization with the model of the controller, referred to as supervisor.

To structure the process of supervisor synthesis we employ a model-based systems engineering framework [25,30], depicted in Figure 1. Following the model-based methodology, domain engineers initially model the specification of the desired controlled system, contrived into a design by domain and software engineers together. The design defines the modeling level of abstraction and control architecture resulting in informal specifications of the plant, control, and performance requirements. Next, the plant and control requirements are modeled in parallel, serving as input to the automated supervisor synthesis tool.

The succeeding steps validate that the control is meaningful, i.e., desired functionalities of the controlled plant are preserved. This step involves (stochastic) verification of the supervised plant based on the model of the performance requirements, or validation by...
simulation. If validation fails, then the control requirements are remodeled, and sometimes a complete revision proves necessary. Finally, the control software is generated automatically, based on the validated models. We note that software engineers shift their focus from writing code to modeling.

We intend to enhance the model-based systems engineering framework of Figure 1 with stochastic capabilities to enable performance and reliability analysis as depicted in grey background. To support supervisory control of (nondeterministic) stochastic discrete-event systems, we will employ the process-theoretic model of Interactive Markov chains [13]. Process theories [3] are formalisms suitable for designing models of complex communicating systems.

Interactive Markov Chains uniquely couple labeled transition systems, a standard model which captures nondeterministic discrete-event behavior, with continuous-time Markov chains [16], the most prominent performance and reliability model. The extension is orthogonal, arbitrarily interleaving exponential delays with labeled transitions. It is arguably a natural semantic model [14] for stochastic process calculi [9] and (generalized) stochastic Petri nets [2]. It also supports separation of concerns by employing constraint-oriented specification of the performance aspects as separate parallel processes [14]. The process theory for Interactive Markov Chains that we will develop captures the central notion of controllability by means of a behavioral relation. Controllability defines the conditions under which a supervisor exists such that the control requirements are achievable by synchronizing it with the plant.

We plan to develop the proposed framework in four phases: (1) develop a process theory to capture the notion of controllability for Interactive Markov Chains, (2) develop a minimization procedure for the stochastic plant that preserves both controllability and stochastic behavior, (3) develop and implement a supervisor synthesis algorithm that satisfies the given control requirements and retains stochastic behavior, and (4) extract directive optimal supervisors that satisfy the performance specification. The framework will provide for convenient modeling of safety and performance requirements, supervisor synthesis for nondeterministic stochastic plants, and extraction of optimal directive supervisors. We will apply it in industrial case studies [23,32] dealing with energy-efficient and reliable supervision and coordination of distributed system components.

In this paper we study point (1), i.e., we develop a stochastic process theory geared towards supervisory control for Interactive Markov Chains. As a base for the behavioral relation that captures controllability for nondeterministic stochastic plants we introduce a stochastic variant of partial bisimulation preorder [4,29]. We give a sound and complete axiomatization that sets up the groundwork for defining the control problem. We discuss controllability with respect to both stochastic and time-abstractive control requirements.

In future work, the equivalence induced by the partial bisimulation preorder will naturally determine the minimization procedure of point (2) and support the development of the synthesis algorithm of point (3). The supervised plant with the performance specification will serve as input to point (4). The analysis encompassed within the framework involves elimination of labeled transitions by means of minimization procedures based on weak bisimulation relations [13] or lumping [24], followed by Markovian analysis [16] or stochastic model checking [6,19].

The proofs of the theorems in this paper follow the same lines as the proofs in [4] and employ standard process-theoretic techniques. We will publish an accompanying technical report that can be accessed at http://se.wtb.tue.nl.
2 Related Work

Supervisory control theory traditionally considers the language-theoretic domain [8, 28], despite early process-theoretically inclined approaches employing failure semantics [15, 18]. The use of refinement relations to relate the supervised plant, given as a desired control specification to be achieved, to the original plant was studied in [22, 26, 33]. A coalgebraic approach introduced partial bisimulation as a suitable behavioral relation that defines controllability [29]. It suggests that controllable events should be simulated, whereas uncontrollable events should be bisimulated. We adopted partial bisimulation to present a process-theoretic approach of supervisory control in a nondeterministic setting [4]. The main motivation of the approach is the elegance, conciseness, and efficient minimization algorithms that (bi)simulation-based relations support [3].

Regarding optimal control, conventional Markov processes were initially enhanced with control capabilities, by employing instant control actions that enable a choice between several possible future behaviors [16]. The control problem is to schedule the control actions such that some performance measure is optimized, typically solved by dynamic programming techniques [7]. Stochastic games problem variants that specify the control strategy using probabilistic extensions of temporal logics are emerging in the formal methods community as well [5].

On the other hand, in the supervisory control community discrete-event system were empowered with costs for disabling or taking transitions [31]. The optimal control problem in such a setting is to synthesize a supervisor that minimizes these costs. Extension with probabilities followed, leading to supervisory control of probabilistic languages [10, 21]. At this point, the supervisor can remain unquantified [17], or it can be (randomized) probabilistic, attempting to match the control specification [21, 27]. Extensions of traces with Markovian delays enabled computation of standard performance measures [20]. The optimal supervisory control problem is also tackled in the Petri net community, usually posed and solved as a linear programming problem.

Our proposal exploits the strengths of both approaches from above by employing traditional techniques to first synthesize a supervisor that will conform to the qualitative control requirements. Afterwards, we will extract a directive supervisor that will also second the quantitative performance requirements. This supervisor directs the plant by picking the specific activities that lead to optimal behavior. What will enable us to apply both techniques is the choice of the underlying process-theoretic model of Interactive Markov chains. We note however, that the (syntactic) manipulation of the Markovian transitions systems must be justified by showing that it preserves the stochastic compositional behavior, which is not an easy task [24]. Moreover, we need to cater for controllability following the guidelines of [4].

3 Interactive Markov Chains

Interactive Markov Chains are typically treated as extensions of labeled transition systems with Markovian transitions labeled by rates of the exponential distributions.

Definition 3.1. An Interactive Markov Chain is a tuple \( I = (\mathcal{S}, s_0, \mathcal{A}, \rightarrow, \mapsto, \downarrow) \), where \( \mathcal{S} \) is a set of states with an initial state \( s_0 \in \mathcal{S} \), \( \mathcal{A} \) is a set of action labels, \( \rightarrow \subseteq \mathcal{S} \times \mathcal{A} \times \mathcal{S} \) is a set of labeled transitions, \( \mapsto \subseteq \mathcal{S} \times \mathbb{N}^{>0} \mathcal{S} \) is a set of Markovian transitions, and \( \downarrow \subseteq \mathcal{S} \) is a successful termi-
We write \( p \xrightarrow{a} p' \) for \( (p, a, p') \in \rightarrow \) and \( p \xrightarrow{\lambda} p' \) for \( (p, \lambda, p') \in \rightarrow \) for \( p, p' \in S, a \in A \), and \( \lambda > 0 \). We also write \( p \xrightarrow{a} \) if there exists \( p' \in S \) such that \( p \xrightarrow{a} p' \) and \( p \xrightarrow{\lambda} p' \) if there exists \( p' \in S \) such that \( p \xrightarrow{\lambda} p' \). An Interactive Markov Chain becomes a labeled transition system if there does not exist \( p \in S \) and \( \lambda > 0 \) such that \( p \xrightarrow{\lambda} \). It becomes a conventional Markov chain if there does not exist \( p \in S \) and \( a \in A \) such that \( p \xrightarrow{a} \).

Labeled transitions are interpreted as standard delayable actions in process theories, i.e., an arbitrary amount of time can pass in a state comprising only outgoing labeled transitions, after which a nondeterministic choice is made on which one of the outgoing labeled transitions should be taken [3].

Intuitive interpretation of a Markovian transition \( p \xrightarrow{\lambda} p' \) is that there is a switch from state \( p \) to state \( p' \) within a time delay with duration \( d > 0 \) with probability \( 1 - e^{-\lambda d} \), i.e., the Markovian delays are distributed according to a negative exponential distribution which parameter labels the transition. By \( R(p, p') \) for \( p, p' \in S \) we denote the rate to transit from \( p \) to \( p' \), i.e., \( \sum \{ \lambda \mid p \xrightarrow{\lambda} p' \} \). By \( R(p, C) \) we denote the exit rate of \( p \in S \) to \( C \subseteq S \) given by \( \sum_{p' \in C} R(p, p') \).

If a given state \( p \) has multiple outgoing Markovian transitions, then there is probabilistic choice between these transitions, known as the race condition [13], and the probability of transiting to \( p' \) following a delay with duration \( d > 0 \) is given by \( \frac{R(p, p')}{\sum R(p, p')} (1 - e^{-R(p, S)d}) \). Roughly speaking, a discrete probabilistic choice is made on the winning rate to one of the candidate outgoing states, whereas the duration of the delay depends on the total exit rate of the origin state. In case a state has both labeled and Markovian transitions, then a nondeterministic choice is made on one of the labeled transitions after some arbitrary amount of time, provided that a Markovian transition has not been taken before with probability as described above.

The successful termination option predicate denotes states in which we consider the modeled process to be able to successfully terminate [3]. In supervisory control theory these states are referred to as marker states [8, 28].

Synchronization of Interactive Markov Chains induces a race condition between the Markovian transitions of the synchronizing states. It also merges labeled transitions in an lock-step manner, i.e., two synchronizing transitions are merged only if they have the same labels. Since labeled transitions can delay arbitrarily, Markovian delays can be interleaved consistently with the race condition, which is one of the greatest advantages of this model [13, 14].

**Example 3.2.** We depict Interactive Markov Chains as in Figure 2. The Interactive Markov Chain depicted in Figure 2a) has initial state labeled by 1 and it initially experiences a Markovian delay according to a negative exponential distribution with parameter \( \lambda \), or delay parameterized by \( \lambda \) for short, followed by a nondeterministic choice between two transitions labeled by \( b \) and \( c \), respectively. The process deadlocks if the transition labeled by \( c \) is chosen, whereas the transition labeled by \( b \) is followed by a transition labeled by \( a \) returning to the initial state. The Interactive Markov Chain depicted in Figure 2b) with initial state \( A \) at the beginning exhibits a race condition between two delays parameterized by \( \mu \) and \( \nu \), respectively. The probability of choosing the transition labeled by \( \mu \) is \( \mu/\mu+\nu \), whereas the probability of choosing the other transition is \( \nu/\mu+\nu \). The process continues with with two labeled transitions, which lead to successfully terminating states.
The result of the synchronization of the processes depicted in Figure 2a) and b) is the Interactive Markov Chain depicted in Figure 2c). The states are labeled by merging the labels of the synchronizing states from both processes. As the Interactive Markov Chains initially have Markovian transitions labeled by $\lambda$, $\mu$, and $\nu$, their synchronization has a race condition between these delays. Following Markovian memoryless semantics [16], if the Markovian delay parameterized by $\lambda$ expires first, then the delays parameterized by $\mu$ and $\nu$ are reset, and they sample again from their corresponding distributions in state \( 2A \). Similarly, if the winner is the Markovian delay parameterized by $\mu$ or $\nu$. We note that such interleaving of Markovian delays correctly captures the intuition that the passage of time following a race imposed of Markovian delays should behave as the maximal delay that lost the race [13]. In case the race between the delays parameterized by $\mu$ and $\nu$ is won by the delay parameterized by $\mu$ leading to state \( 2B \), we need to synchronize one of the transitions labeled by $b$ and $c$ with a transition labeled by $a$. Since no synchronization is possible the process deadlocks. In case the winning delay is parameterized by $\nu$, both transitions labeled by $c$ are synchronized in state \( 2C \) leading to state \( 4E \), which can successfully terminate, as both states \( 4 \) and \( E \) have successful termination options.

Suppose that the Interactive Markov Chain depicted in Figure 2a) represents a plant and that we want to execute the loop comprising the states 1, 2, and 3 once, and afterwards successfully terminate in 4. The supervisor that achieves this is depicted in Figure 2d). We note that the supervisor does not contain stochastic behavior since Markovian delays cannot be synchronized. They are interleaved with the labeled transitions, which are forced to synchronize and restrict the behavior of the plant. The supervised plant that is obtained by synchronization is depicted in Figure 2e).

Next, we introduce a process theory with semantics in terms of Interactive Markov Chains modulo a behavioral relation that captures the notion of controllability.
4 Process Theory $\text{BSPIMC}_{(\mathcal{A}, \mathcal{B})}$

We define a Basic Sequential Process theory for Interactive Markov Chains $\text{BSPIMC}_{(\mathcal{A}, \mathcal{B})}$ with full synchronization and a Markovian partial bisimilarity preorder, following the nomenclature of [3]. The theory is parameterized by a finite set of actions $\mathcal{A}$ and a bisimulation action set $\mathcal{B} \subseteq \mathcal{A}$, which plays a role in the behavioral relation.

4.1 Process Terms

The set of process terms $\mathcal{I}$ is induced by

$$P ::= 0 \mid 1 \mid a.P \mid \lambda.P \mid P + P \mid P | P$$

for $a \in \mathcal{A}$ and $\lambda > 0$. By $\mathcal{L}$ we will denote the process terms that do not contain the Markovian prefix. The constant process $0$ can only delay for an arbitrary amount of time after which it deadlocks, whereas $1$ delays with an option to successfully terminate. The process corresponding to $a.p$ delays for an arbitrary amount of time, executes the action $a$, and continues behaving as $p$. The process corresponding to $\lambda.p$ takes a sample from the negative exponential distribution parameterized by $\lambda$ (cf. Section 3) and delays with a duration determined by the sample after which it immediately continues to behave as $p$. The alternative composition $p + q$ behaves differently depending on the context (cf. Section 3). It induces a race condition if $p$ or $q$ comprise Markovian prefixes or, alternatively, it makes an arbitrarily delayed nondeterministic choice on an action, if $p$ or $q$ comprise action prefixes, provided that a Markovian delay has not expired. The synchronous parallel composition $p | q$ synchronizes all actions of $p$ and $q$ if possible, or it delays according to the race condition by interleaving any constituent Markovian prefixes, or it delays arbitrarily and deadlocks, otherwise. The binding power of the operators from strongest to weakest is: $\_a$, $\_\lambda$, $\_\_ | \_\_$, and $\_\_ + \_\_$. 

The semantics of process terms is given by Interactive Markov Chains, which states are taken from $\mathcal{I}$ and the initial state is the starting process term. The successful termination option predicate $\downarrow$, the labeled transition relation $\rightarrow$, and the Markovian transition relation $\mapsto \rightarrow$ are defined using structural operational semantics [3] in Figure 3.

![Figure 3: Structural operational semantics of $\text{BSPIMC}_{(\mathcal{A}, \mathcal{B})}$ terms](attachment:image.png)

Rule 1 states that the constant $1$ enables successful termination. Rules 2 and 3 state that the alternative composition has a termination option if at least one component has it. Rule 4 states that the synchronous parallel composition has a termination option.
only if both components have a termination option. Rule 5 states that action prefixes induce outgoing transitions with the same label. Rules 6 and 7 enable a nondeterministic choice between the outgoing transitions of the constituents of the alternative composition. Rule 8 states that in a synchronous parallel composition both components execute the same actions in lock-step. Rule 8 states that Markovian prefixes induces Markovian transitions with the same parameter. Rules 10 and 11 enable the race condition between Markovian transitions in the alternative composition, whereas rules 12 and 13 do the same for the synchronous parallel composition.

Next, we define the notion of Markovian partial bisimulation that represents the basis for the preorder that portrays controllability for Interactive Markov Chains.

4.2 Markovian Partial Bisimulation

The basic idea of Markovian partial bisimulation is that the "greater" process should simulate the "smaller", whereas the latter should simulate back only the actions in the bisimulation action set \( B \), which is the parameter of the theory. The Markovian transitions are handled using their rates as they are employed in the race condition. In the definition of the Markovian bisimulation, this is resolved by requiring that the relation is an equivalence [13]. Our behavioral relation, however, is not symmetric, so we will only require that the relation is reflexive and transitive and we will employ the induced equivalence to ensure that the exit rates of equivalent states coincide.

We introduce some preliminary notions used in the definition. Given a relation \( R \), we will write \( R^{-1} \triangleq \{(q,p) \mid (p,q) \in R\} \). We note that if \( R \) is reflexive and transitive, then it is not difficult to show that \( R^{-1} \) and \( R \cap R^{-1} \) are reflexive and transitive as well. Moreover, \( R \cap R^{-1} \) is symmetric, making it an equivalence. We employ this equivalence to ensure that the exiting Markovian rates to equivalence classes coincide as in the definition for Markovian bisimulation [13].

Definition 4.1. A reflexive and transitive relation \( R \subseteq I \times I \) is a Markovian partial bisimulation with respect to the bisimulation action set \( B \subseteq A \) if for all \( p,q \in I \) such that \((p,q) \in R\) it holds that:

1. if \( p \downarrow \), then \( q \downarrow \);
2. if \( p \xrightarrow{a} p' \) for some \( a \in A \), then there exists \( q' \in I \) such that \( q \xrightarrow{a} q' \) and \((p',q') \in R\);
3. if \( q \xrightarrow{b} q' \) for some \( b \in B \), then there exists \( p' \in I \) such that \( p \xrightarrow{b} p' \) and \((p',q') \in R\);
4. \( R(p,C) = R(q,C) \) for all \( C \in I/(R \cap R^{-1}) \).

We say that \( p \in I \) is partially bisimilar to \( q \in I \) with respect to the bisimulation action set \( B \subseteq A \) if for all \( p,q \in I \) such that \((p,q) \in R\) it holds that:

Definition 4.1 ensures that \( p \in I \) can be partially bisimulated by \( q \in I \) if the 1) termination options can be simulated, 2) the labeled transitions can also be simulated, whereas 3) the transitions labeled by action from the bisimulation action set are bisimulated, 4) the Markovian transitions must be related and, moreover, 5) the exit rates to equivalent processes must coincide. As an alternative to 4) we can require that every state of \( p \) is related to some state of \( q \). Note that \( \preceq_B \) is a preorder relation, making \( \leftrightarrow_B \) an equivalence relation for all \( B \subseteq A \) [29]. Also, note that if \( p \preceq_B q \), then \( p \preceq_C q \) for every \( C \subseteq B \).
If the processes do not comprise Markovian prefixes, then the relation coincides with partial bisimulation which additionally is required to be reflexive and transitive \[4\]. In that case, \( \cong_B \) coincides with strong similarity preorder and \( \leftrightarrow_B \) coincides with strong similarity equivalence \[3\], whereas both \( \preceq_A \) and \( \leftrightarrow_A \) turn into strong bisimilarity \[3\]. If the processes comprise only Markovian prefixes, then the relation corresponds to ordinary lumping \[13,24\]. If the processes comprise both action and Markovian prefixes, and if \( B = A \), then \( \leftrightarrow_A \) corresponds to strong Markovian bisimulation \[13\]. Thus, the orthogonal reduction of the Markovian partial bisimulation correspond to the established behavioral relations, which brings confidence that our behavioral relation is well-defined.

**Theorem 4.2.** Suppose \( p \preceq_B q \) with \( B \subseteq A \) and \( p, q \in I \). Then: (1) \( a.p \preceq_B a.q \); (2) \( \lambda . p \preceq_B \lambda . q \); (3) \( p + r \preceq_B q + r \) and \( r + p \preceq_B r + q \); and (4) \( p \mid r \preceq_B q \mid r \) and \( r \mid p \preceq_B r \mid q \), for all \( a \in A \), \( \lambda > 0 \), and \( r \in I \).

**Proof.** Suppose \( p \preceq q \). Then there exists a partial bisimulation relation \( R \) such that \( pRq \) as given by Definition 4.1. We define for each case a separate partial bisimulation relation \( R' \) based on \( R \). We only show one of the symmetrical cases for the alternative and parallel composition, as the other holds by symmetry of the operational rules.

- Define \( R' = \{(a.p, a.q), (a.p, a.p), (a.q, a.q)\} \cup R \). It should be clear that if \( R \) is reflexive and transitive, then \( R' \) is reflexive and transitive as well. The terms \( a.p \) and \( a.q \) cannot terminate and have the outgoing transitions \( a.p \xrightarrow{a} p \) and \( a.q \xrightarrow{a} q \) with \( (p, q) \in R \). The exit rates of \( a.p \) and \( a.q \) coincide as they are 0, making \( R' \) a Markovian partial bisimulation.

- Define \( R' = \{(\lambda.p, \lambda.q), (\lambda.p, \lambda.p), (\lambda.q, \lambda.q)\} \cup R \). It should be clear that if \( R \) is reflexive and transitive, then \( R' \) is reflexive and transitive as well. The terms \( \lambda.p \) and \( \lambda.q \) cannot terminate. The exit rates of \( \lambda.p \) and \( \lambda.q \) coincide as they are both \( \lambda \), making \( R' \) a Markovian partial bisimulation.

- Define \( R' = \{(p + r, q + r) \cup (r, r) \mid r \in I \} \cup R \). The relation \( R'' = \{(r, r) \mid r \in I \} \cup R \) is a Markovian partial bisimulation relation. Note that \( R' \) is reflexive and transitive, since \( R'' \) is the identity relation on \( I \) and \( R \) is transitive. The term \( p + r \) terminates due to a termination options of \( p \) or due to a termination option of \( r \). In the former situation \( q + r \) terminates due to a termination option of \( q \) and in the latter due to a termination option of \( r \). After the initial transition, be it an labeled or Markovian transition, the remaining terms are related by \( R'' \), if \( r \) is chosen, or by \( R \), if \( p \) and \( q \) are chosen. The exit rates of \( p + r \) and \( q + r \) coincide as the ones of \( p \) and \( q \) coincide.

- Define \( R' = \{(p \mid r, q \mid r) \mid (p, q) \in R, r \in I \} \cup \{(r, r) \mid r \in I \} \). Again, it should be clear that \( R' \) is reflexive and transitive. The terms \( p \mid r \) and \( q \mid r \) either have termination options due to coinciding termination of \( p, q \), and \( r \) or do not terminate. According to operational semantics \( p \mid r \xrightarrow{a} \) only if \( a \in A \). As \( (p, q) \) are related by \( R \), their exit rates coincide, implying the same for \( p \mid r \) and \( q \mid r \).

Theorem 4.2 states that \( \preceq_B \) is a precongruence, making \( \leftrightarrow_B \) a congruence for \( I \) and providing for substitution rules. We build the term model \( \mathbb{P}(\text{BSPIMC}_A(B)) \) \[3\], where

\[
\mathbb{P}(\text{BSPIMC}_A(B)) = (I, 0, 1, a.\_ for a \in A, \lambda.\_ for \lambda > 0, +, _, _|_).
\]
Axiomatization

We depict a sound and ground-complete axiomatization of the precongruence \( \preceq_B \) in Figure 4, whereas \( \leftrightarrow_B \) is not finitely axiomatizable [4]. The notation \( \sum_{i \in I} a_i \cdot p_i \) stands for the alternative composition of \( a_i \cdot p_i \in \mathcal{I} \) for some \( I \subseteq \mathbb{N} \), where \( \sum_{i \in I} a_i \cdot p_i \equiv 0 \) if \( I = \emptyset \). We can employ such a notation as the alternative composition is commutative and associative. For the sake of clarity and compactness of notation, we use \([+p]\) to denote an optional summand \( p \).

Axioms A1 and A2 express commutativity and associativity of the alternative composition, respectively. Axioms A3 and A4 state idempotence of the successful termination and the action prefix and they are characteristic for stochastic theories. Axiom A5 expresses that deadlock is a neutral element of the alternative composition. Axiom M resolves the race condition by adding Markovian rates to equivalent processes. Axioms P1 and P2 enable elimination of the successful termination option and action prefixed terms that do not have to be bisimulated, respectively. The synchronous parallel composition requires an expansion law [13] as it does not distribute with respect to the alternative composition due to the race condition. For example, \( (\lambda \cdot p + \nu \cdot q) \mid | \nu \cdot r + \mu \cdot q \mid \nu \cdot r \) for every \( p, q, r \in \mathcal{I} \) and \( \lambda, \mu, \nu > 0 \). The first process term induces a race between three stochastic delays parameterized by \( \lambda, \mu, \nu \), whereas the second induces a race between four stochastic delays parameterized by \( \lambda, \mu, \nu \), and \( \nu \), respectively. To apply the expansion law \( S \), one first needs to use axioms A1, A2, A3, A6, and M to obtain the required normal forms of the synchronizing processes [3]. We reiterate that the result of the synchronization has a successful termination option if both constituents exhibit it.

**Theorem 4.3.** The axioms of \( \text{BSPIMC}_\mathcal{A}(\mathcal{A}, \mathcal{B}) \), depicted in Figure 4, are sound and ground-complete for partial bisimilarity, i.e., \( p \preceq_B q \) is derivable if and only if \( p \preceq_B q \).

**Proof.** It should not be difficult to observe that axioms A1-A5 and M, form a sound and ground-complete axiomatization for strong Markovian bisimilarity [13], i.e., for \( \preceq_A \). Thus, from now on, we can assume that \( B \neq \mathcal{A} \).

The soundness of axioms P1 and P2 follows directly by application of the operational rules and Definition 4.1 for partial bisimilarity. It is sufficient to take \( R = \{(p, p+1)\} \cup \{(p, p) \mid p \in \mathcal{I}\} \) and \( R' = \{(q, a \cdot p + q)\} \cup \{(q, q) \mid q \in \mathcal{I}\} \) as partial bisimulations between the terms for axiom P1 and P2, respectively. Note that in both cases the exit rates are not altered. For axiom P1 it is clear that \( p + 1 \) terminates if \( p \) terminates, and they have the same outgoing labeled transitions. For axiom P2, if \( q \overset{c}{\rightarrow} q' \) for some \( q' \in \mathcal{I} \) and \( c \in \mathcal{A} \), \( a \cdot p + q \overset{\lambda}{\rightarrow} q' \) and \( (q', q') \in R \). Vice versa, the outgoing transitions labeled by \( b \in B \) of \( a \cdot p + q \) must originate from \( q \) as \( a \cdot p \) has only one outgoing transition labeled by \( a \notin B \).
Therefore, if \(a.p + q \xrightarrow{b} q'\) for some \(q' \in \mathcal{I}\) and \(b \in B\), then \(q \xrightarrow{b} q'\) and \((q', q'') \in R\).

In order to apply the expansion law \(S\), we need to transform the constituents of the parallel composition in a normal form applying axioms A1-A5 and \(M\) as outlined in, e.g., [1,3]. Note that the result of the expansion law is in normal form. By direct inspection of the operational rules we see that the processes have the same exit rates to the same processes and the same alternatives regarding labeled transitions. This makes both sides of the expansion law bisimilar. In order to show completeness, we reuse the normal forms and we transform every \(p, q \in \mathcal{I}\) to

\[
p' = B \sum_{i \in I} a_i p_i + \sum_{k \in K} \lambda_k r_k[+1] \\
qu' = B \sum_{j \in J} b_j q_j + \sum_{\ell \in L} \mu_\ell s_\ell[+1]
\]

where \(a_i, c_j \in A \setminus B\), \(\lambda_k, \mu_\ell > 0\), and \(p_i, q_j, r_k, s_\ell \in \mathcal{I}\), for \(i \in I\) and \(j \in J\). We denote the normal forms of \(p\) and \(q\) by \(p'\) and \(q'\), respectively. From \(p \leftrightarrow A\) \(p'\) it follows that \(p \leftrightarrow B\) \(p'\) as well. Analogously, we have \(q \leftrightarrow B\) \(q'\), so we can conclude that \(p' \preceq_B q'\) if and only if \(p \preceq_B q\). We eliminate idempotent summands and Markovian prefixes before equivalent terms in the normal forms by applying axioms A3, A4, and \(M\) modulo commutativity and associativity of the alternative composition. Now, the proof can be performed using induction on the total number of symbols, i.e., constants and action and Markovian prefixes, of the terms.

The base cases are \(p' = B 0 \leq_B 0 = B q'\) and \(p' = B 1 \leq_B 1 = B q'\), which hold directly by using the substitution rules in an empty context, and \(p' = B 0 \leq_B 1 = B q'\), which is obtained by applying \(0 \leq_B 0 + 1 = B 1\).

As \(p' \preceq_B q'\), there exists a partial bisimulation \(R\) such that \((p', q') \in R\). It is clear that if \(p'\) contains the optional summand 1, then \(q'\) contains it as well. If \(q'\) comprises the summand 1 and \(p'\) does not contain it, then we use axiom P1 to eliminate it. Suppose that \(p' \xrightarrow{a} p''\) for some \(a \in A\) and \(p'' \in \mathcal{I}\). Then, according to the operational rules there exists a summand \(a_kp_k\) of \(p''\) for some \(k \in I\) such that \(a_k = a\) and \(p_k = p''\). Analogously, by Definition 4.1 there exists a summand \(c_\ell q_\ell\) of \(q'\), such that \(c_\ell = a\) and \((p_\ell, q_\ell) \in R\) for some \(\ell \in J\). So, \(p_k \preceq_B q_\ell\) and, hence, by the induction hypothesis, \(p_k \preceq_B q_\ell\). Thus, there exists \(L \subseteq J\) such that for every \(i \in I\) there exists \(\ell \in L\) such that \(a_i, p_i \preceq_B c_\ell, q_\ell\). Vice versa, for every \(j \in J\) such that \(c_j \in B\) there exists \(i \in I\) such that \(a_i, p_i \preceq_B c_j, q_j\). Note that the same must hold in both directions for the Markovian prefixes, as there are no equivalent terms due to application of axiom \(M\).

Denote by \(K = L \cup \{j \mid c_j \in B, j \in J\}\). Now, we split \(q'\) to \(q' = q'' + q'''\) such that \(q''\) contains the summands that are prefixed by an action in \(B\) or that have an index in \(L\) and \(q'''\) comprises the remaining summands, i.e., \(q'' = \sum_{k \in K} c_k q_k\) and \(q''' = \sum_{\ell \in J \setminus K} c_\ell q_\ell\). Note that \(p''\) contains only summands prefixed by actions that are not in \(B\). Now, we have that \(p = B p' \preceq_B q'\). By applying Axiom P2 for the summands \(c_\ell q_\ell\) of \(q'''\) for \(\ell \in J \setminus K\) we obtain \(q' \preceq_B q''' + q''' = B q' = B q\), leading to \(p \preceq_B q\), which completes the proof.

We note that when the bisimulation action set \(B = \emptyset\), axiom P2 is valid for every possible prefix, effectively replacing axioms P1 and P2 with \(q \leq_B p + q\). Moreover, if the processes do not contain Markovian prefixes, then axiom \(M\) becomes inapplicable, reducing \(\text{BSPIMC}(\mathcal{A}, \emptyset)\) to the sound and ground-complete process theory with an expansion law for strong similarity preorder [3]. When \(B = \mathcal{A}\) axiom P2 becomes inapplicable and the remaining axioms (minus axiom P1) form a sound and ground-complete process theory for strong bisimilarity [3,12]. When the Markovian prefixes are present as well, we obtain the sound and ground-complete process theory for strong Markovian bisimulation [13].
4.4 Little Brother Terms

An important aspect of similarity-like equivalences, which plays an important role in their characterization are the so-called little brother terms [11]. Their characterization makes a minimization procedure for mutual partial bisimilarity possible, which is the cornerstone for plant aggregation that respects controllability. Two similar labeled transition systems that do not contain little brothers are actually strongly bisimilar [11], implying the same property for partially bisimilar terms.

**Definition 4.4.** Let \( p \xrightarrow{a} p' \) and \( p \xrightarrow{a} p'' \) for some \( a \in A \) and \( p, p', p'' \in \mathcal{I} \). If \( p' \preceq_B p'' \) holds, but \( p'' \preceq_B p' \) does not hold, then we say that \( p' \) is the little brother of \( p'' \). ■

The following equations show how to eliminate little brother terms.

**Theorem 4.5.** Suppose \( p \preceq_B q \preceq_B r \) for \( p, q, r \in \mathcal{I} \). Then:

\[
\begin{align*}
    a.p + a.q & \leftrightarrow_B a.q & \text{if } a \notin B & \quad \text{LB1}, \\
    b.p + b.q + b.r & \leftrightarrow_B b.p + b.r & \text{if } b \in B & \quad \text{LB2}. \\
\end{align*}
\]

Proof. We show that the equations are sound by showing the inequalities in both directions. For equation LB1 we have that \( a.p + a.q \leq a.p \) holds directly by axiom P2. For the other direction we calculate \( a.p = a.p + a.p \leq a.p + a.q \) using axiom A4 and the premise, respectively. For equation LB2 we have the following derivation, calculated by using axiom A4 and the premise, accordingly:

\[
\begin{align*}
    b.p + b.q + b.r & \leq b.p + b.r + b.r = b.p + b.r \\
    b.p + b.r = b.p + b.p + b.r & \leq b.p + b.q + b.r,
\end{align*}
\]

which completes the proof. ■

We note that LB1 is equivalent to the characteristic similarity relation \( a.(p + q) + a.q \leftrightarrow_B a.(p + q) \) when \( B = \emptyset \) [3]. Since the prefix action does not play a role in strong similarity, the relation there always holds. However, when the little brothers are prefixed by a bisimulation action \( b \in B \), the ‘littlest’ and ‘biggest’ brother must be preserved, as given by LB2. These equations will be employed when computing the quotient Interactive Markov Chain in the minimization procedure of point (2) (see Section 1), following the computation of the coarsest mutual partial bisimilarity.

5 Controllability of Interactive Markov Chains

We define controllability from a process-theoretic perspective in terms of the partial bisimilarity preorder. In the vein of [8, 28] we split \( A \) into a set of uncontrollable actions \( U \subseteq A \), and a set of controllable actions \( C = A \setminus U \). The former typically represent activities of the system at hand over which we do not have any control, like sensor measurements, whereas the latter can be disabled or enabled in order to achieve the behavior given by the control requirements, e.g., enabling or disabling actuators. First, we assume that both the plant and the control requirements are given as \( \text{BSPIMC}\langle A, B \rangle \) processes or, equivalently, in the form of Interactive Markov Chains. Subsequently, we propose a way to separate the concerns by abstracting from stochastic behavior in the control requirements and introducing performance specifications, as depicted in Figure 1, e.g., by means of stochastic temporal logics [5, 19].
5.1 Supervised Plant

We will use \( p \in \mathcal{I} \) to denote the plant and \( r \in \mathcal{I} \) for the control requirements. The supervised plant is given by \( p \upharpoonright s \) for a supervisor \( s \in \mathcal{L} \). Intuitively, the uncontrollable transitions of the plant should be bisimilar to those of the supervised plant, so that the reachable uncontrollable part of the former is indistinguishable from that of the latter. Note that the reachable uncontrollable part now contains the Markovian transitions as well, hence preserving the race condition that underpins the stochastic behavior. The controllable transitions of the supervised plant may only be simulated by the ones of the original plant, since some controllable transitions are suppressed by the supervisor. The stochastic behavior represented implicitly by the Markovian transitions and the underlying race condition is preserved employing lumping of the Markovian exit rates to equivalent states. Again, we emphasize that the supervisor does not contain any stochastic behavior as it should cater only for proper disabling of controllable transitions.

**Definition 5.1.** Let \( p \in \mathcal{I} \) be a plant and \( r \in \mathcal{I} \) control requirements. We say that \( s \in \mathcal{L} \) is a supervisor for \( p \) that satisfies \( r \) if \( p \upharpoonright s \leq_U p \) and \( p \upharpoonright s \leq_{\emptyset} r \).

As expected, Definition 5.1 ensures that no uncontrollable actions have been disabled in the supervised plant, by including them in the bisimulation action set. Moreover, it takes into account the nondeterministic behavior of the system. It suggests that the control requirements model the allowed behavior, independent of the plant. We opt for an ‘external’ specification in process-theoretic spirit and we require that the supervised plant has a behavior that is allowed, i.e., that can be simulated, by the control requirements. This setting is also a preparation for separation of concerns, where we will employ a time-free simulation that abstracts from the stochastic behavior of the plant to capture the relation between the supervised plant and the control requirements.

If we assume that the control requirements coincide with the desired supervised behavior, i.e., \( r \equiv_U p \upharpoonright s \), then we only require that \( r \leq_U p \), as \( r \leq_{\emptyset} r \) always holds, conforming to the original setting of [28]. Moreover, when \( p \) and \( r \) are deterministic labeled transition systems, this coincides with language controllability, which was the original purpose of partial bisimilarity in [29]. When Markovian prefixes are present as well, our work extends Markovian traces [20] to capture full nondeterminism.

We note that if we take the trivial case when both the plant and the control requirements coincide, then the corresponding conditions \( p \upharpoonright s \leq_U p \) and \( p \upharpoonright s \leq_{\emptyset} p \) will collapse to \( p \upharpoonright s \leq_U p \). We note that the equation suggests that we cannot directly establish that the plant and the supervised plant are bisimilar, even though we chose bisimilarity as an appropriate notion to capture nondeterminism. However, if the plant does not contain any redundant behavior in the form of little brothers, we have that \( p \upharpoonright s =_{\mathcal{L}} p \) implies \( p \upharpoonright s =_{\mathcal{L}} p \) [4]. This property of the notion of controllability given by Definition 5.1 is not preserved by any other stochastic extension of supervisory control theory that is based on the notion of controllability for nondeterministic systems introduced in [15,18,33], cf. [4] for a detailed discussion on the topic.

Definition 5.1 also admits nondeterministic supervisors in the vein of [33]. We can also observe that the restriction of the supervisor to labeled transition systems is a choice and not a necessity, as the supervised plant \( p \upharpoonright s \) is well-defined for \( s \in \mathcal{I} \) as well. We opt not to implement both nondeterministic and stochastic supervisors as they alter the nondeterministic and stochastic behavior of the plant, respectively, for which we cannot phantom a reasonable interpretation. We consider specifications that require the employment of nondeterministic supervisors as ill-defined. As discussed before in Section 3, the stochastic behavior of the supervisor has no significant contribution as Markovian transitions are interleaved. We may consider probabilistic supervisors in the...
\[
\begin{align*}
\pi_{tf}(0) &= A 0 & \text{TF1} \\
\pi_{tf}(1) &= A 1 & \text{TF2} \\
\pi_{tf}(a.p) &= A a.\pi_{tf}(p) & \text{TF3} \\
\pi_{tf}(\lambda.p) &= A \pi_{tf}(p) & \text{TF4} \\
\pi_{tf}(p + q) &= A \pi_{tf}(p) + \pi_{tf}(q) & \text{TF5} \\
\pi_{d}(0) &= A 0 & \text{D1} \\
\pi_{d}(a.p + a.q + r) &= A \pi_{d}(a.(p + q) + r) & \text{D2} \\
\pi_{d}\left(\sum_{i \in I} a_{i} p_{i} + 1\right) &= A \sum_{i \in I} a_{i} \pi_{d}(p_{i}) + 1 & \text{D3} \\
\text{if } a_{j} \neq a_{k} \text{ for all } j \neq k, j, k \in I \\
\text{where } [ + 1 ] \text{ is either present or not simultaneously.}
\end{align*}
\]

Figure 5: Axioms for the time-free and the determinized projection

According to Definition 5.1, the minimal possible supervised plant is the initial uncontrollable reach of the plant, i.e., the topmost subterm of \( p \) comprising only uncontrollable and Markovian prefixes. For example, the minimal supervised behavior of \( p \equiv u.\lambda.0 + c.u.0 + v.c.\mu.0 \), assuming that \( p \leftrightarrow r, u, v \in U, \lambda, \mu > 0 \), and \( c \in C \), is \( u.\lambda.0 + v.0 \). The maximal supervised behavior is the plant itself, i.e., every plant can accept itself as a control requirement.

### 5.2 Controllability

A usual suspect for a deterministic supervisor is the determinized version of the desired supervised behavior as it comprises all traces of the supervised behavior and, therefore, it does not disable any events that we wish to be present. As the supervisor does not contain stochastic behavior, the determinized process should abstract from the passage of time as well. We define a determined time-free projection \( \pi_{dtf}(p) \in \mathcal{L} \) of the process \( p \in \mathcal{I} \) as the minimal process that enables all possible time-free traces of \( p \in \mathcal{I} \). Recall that Markovian delays interleave in the synchronous parallel composition, so \( \pi_{dtf}(p) \) will not suppress any action or Markovian prefixes of \( p \), see Theorem 5.3 below.

The determined time-free projection \( \pi_{dtf}(p) \) is defined as a composition of two projections \( \pi_{tf} \) and \( \pi_{d} \), i.e., \( \pi_{dtf}(p) = \pi_{d}(\pi_{tf}(p)) \), where the former abstracts from the stochastic behavior and the latter merges the nondeterministic choice between equally-labeled transitions. The axioms that capture the behavior of the projections are depicted in Figure 5. We note that the behavioral relation is bisimilarity, since then the axioms will hold for every \( B \subseteq A \). Axioms TF1 and TF2 state that the constant processes 0 and 1 are time-free processes by definition. Axiom TF3 states that the time-free projection propagates through the action prefix. Axiom TF4 enables the abstraction of Markovian prefixes. Axiom TF5 lifts the nondeterministic choice disregarding the race condition in combination with axiom TF4. Axiom D1 states that the deadlock process is already deterministic as it has no outgoing transitions. Axiom D2 merges a nondeterministic choice over equally equally prefixed processes to a single prefix followed by the alternative composition of the original target processes. Axiom D3 expresses that the determined projection can be propagated only when all nondeterministic choices between the action prefixes have been eliminated. The determined projection does not affect the successful termination option.

**Definition 5.2.** We say that a process \( p \in \mathcal{I} \) is deterministic if \( p =_{A} \pi_{dtf}(p) \).
It should be clear from Definition 5.2 and the axioms in Figure 5 that \( \pi_{\text{dtf}}(p) \) is a deterministic process for every \( p \in \mathcal{I} \). The following theorem states two important properties of the determinized time-free projection.

**Theorem 5.3.** For all \( p, q \in \mathcal{I} \) it holds (1) \( p \mid \pi_{\text{dtf}}(p) =_A p \) and (2) if \( p \leq_B q \) then \( r \mid \pi_{\text{dtf}}(p) \leq_B r \mid \pi_{\text{dtf}}(q) \) for every \( r \in \mathcal{I} \) and \( B \subseteq A \).

**Proof.** To show (1), first we need to show that for every \( p \in \mathcal{I} \), we have that \( \pi_{\text{dtf}}(p) =_A \sum_{a \in E} a.\pi_{\text{dtf}}(p_a)[+1] \), where \( E \subseteq A \) and \( p_a \in \mathcal{I} \) for \( a \in E \). We will show this by means of structural induction. We have that \( \pi_{\text{dtf}}(0) =_A 0 \), \( \pi_{\text{dtf}}(1) =_A 1 \), and \( \pi_{\text{dtf}}(a.p) =_A a.\pi_{\text{dtf}}(p) \).

Now suppose that \( \pi_{\text{dtf}}(p) =_A \sum_{a \in E} a.\pi_{\text{dtf}}(p_a)[+1] \) for \( E \subseteq A \) and \( p_a \in \mathcal{I} \) for \( a \in E \) and \( \pi_{\text{dtf}}(q) =_A \sum_{b \in F} b.\pi_{\text{dtf}}(q_b)[+1] \) for \( F \subseteq A \) and \( q_b \in \mathcal{I} \) for \( b \in F \). Then \( \pi_{\text{dtf}}(\lambda.p) =_A \pi_{\text{dtf}}(p) \), which is satisfied by induction. For \( p + q \) we have that

\[
\pi_{\text{dtf}}(p + q) =_A \pi_{\text{dtf}} \left( \sum_{a \in E} a.\pi_{\text{dtf}}(p_a)[+1] + \sum_{b \in F} b.\pi_{\text{dtf}}(q_b) \right)[+1]
\]

where \( r_c = p_a \) if \( c = a \) and \( a \notin F \), \( r_c = q_b \) if \( c = b \) and \( b \notin E \), and \( r_c = p_a + q_b \) if \( c = a + b \) and \( a \in F \) and \( b \in E \) and the optional summand \([+1]\) exists if it exists in \( p \) or \( q \). Thus, we can represent \( \pi_{\text{dtf}}(p) \) as \( \sum_{a \in E} a.\pi_{\text{dtf}}(p_a)[+1] \) as defined above.

Next, we will show that \( p =_A p \mid \pi_{\text{dtf}}(p + q) \) for every \( p, q \in \mathcal{I} \) by total induction on the number of prefixes and constants in \( p \). For \( p = 0 \) and \( q \in \mathcal{I} \) we have that

\[
0 \mid \pi_{\text{dtf}}(0 + q) =_A 0 \mid \pi_{\text{dtf}}(q) =_A 0 \mid \left( \sum_{b \in F} b.\pi_{\text{dtf}}(q_b)[+1] \right) =_A 0.
\]

For \( p = 1 \) and \( q \in \mathcal{I} \) we have:

\[
1 \mid \pi_{\text{dtf}}(1 + q) =_A 1 \mid \left( \sum_{b \in F} b.\pi_{\text{dtf}}(q_b) + 1 \right) =_A 1.
\]

Now, suppose that the normal form of \( p \) is given by \( p =_A \sum_{i \in I} a_i.p_i + \sum_{k \in K} \lambda_k.r_k[+1] \) as above. Recall the form of the determinized time-free projection of \( p \) and \( p + q \) from above. Note that \( r_c \) comprises \( p_i \) for \( a_i = c \) and that \( p + q \) comprises \( r_k \), since \( \pi_{\text{tf}}(\lambda.p) =_A p \). Now, we have that

\[
p \mid \pi_{\text{dtf}}(p + q) =_A \left( \sum_{i \in I} a_i.p_i + \sum_{k \in K} \lambda_k.r_k[+1] \right) \mid \left( \sum_{c \in E \cup F} c.\pi_{\text{dtf}}(r_c)[+1] \right)
\]

where the optional summand \([+1]\) is present only if it is present in \( p \). Now, we have that (1) holds by putting \( q = 0 \).

15 Controllability of Interactive Markov Chains
To show (2), first note that since \( p \leq B q \), if \( p = B \sum_{a \in E} a.\pi_{dtf}(p_a)[1] + 1 \), then \( q = B \sum_{a \in F} b.\pi_{dtf}(q_b)[1] + 1 \) with \( E \subseteq F \) and \( E \cap B = F \cap B \). Now, we show the claim by structural induction.

If \( r \equiv 0 \) then \( 0 \mid \pi_{dtf}(p) = B 0 \) and \( 0 \mid \pi_{dtf}(q) = B 0 \).

If \( r \equiv 1 \) then \( 1 \mid \pi_{dtf}(p) = B 1 \), if \( p \) has the option summand 1, and \( 1 \mid \pi_{dtf}(p) = B 0 \), otherwise. If \( p \) has the optional summand 1, then \( q \) comprises it as well. Thus if \( 1 \mid \pi_{dtf}(p) = B 1 \), then \( 1 \mid \pi_{dtf}(q) = B 1 \) as well. Otherwise we can have that \( 1 \mid \pi_{dtf}q \) amounts to 0 or 1 with 0 \( = B 1 \).

Now, suppose that \( r \equiv a.s \) for some \( s \in I \) with \( a \in B \). If \( a \notin E \) then \( r \mid \pi_{dtf}(p) = B 0 \) since no synchronization with \( a \) can occur. Since \( E \cap B = F \cap B \), we have that \( r \mid \pi_{dtf}(q) = B 0 \) as well. Now, suppose that \( a \in E \). Then \( r \mid \pi_{dtf}(p) = B a.(s \mid p_a) \) and \( r \mid \pi_{dtf}(q) = B a.(s \mid q_a) \) with \( a.(s \mid p_a) \leq \Pi a.(s \mid q_a) \) by the induction premises and Theorem 4.2. Now, suppose that \( a \notin B \) and \( a \notin E \). Then, it is possible that \( a \in F \), but then we have that \( 0 \leq B a.(s \mid q_a) \).

If \( a \in E \), then \( a \in F \), leading to the same situation as above.

Next, suppose \( r \equiv \lambda.s \) for some \( \lambda > 0 \) and \( s \in I \). Then \( r \mid \pi_{dtf}(p) = B \lambda.(s \mid p_a) \) and \( r \mid \pi_{dtf}(q) = B \lambda.(s \mid q_a) \) using expansion law S, with \( \lambda.(s \mid p_a) \leq \Pi \lambda.(s \mid q_a) \).

Finally, let \( r \equiv r_1 + r_2 \) for some \( r_1, r_2 \in I \). Here, we exploit that for every \( p, q \in I \) and \( r \in L \) it holds that \( (p + q) \mid r = B p \mid r + q \mid r \) by simple inspection of the expansion law S in Figure 4. Thus, \( r \mid \pi_{dtf}(p) = B r_1 \mid \pi_{dtf}(p) + r_2 \mid \pi_{dtf}(p) \) and \( r \mid \pi_{dtf}(q) = B r_1 \mid \pi_{dtf}(q) + r_2 \mid \pi_{dtf}(q) \), implying the above using the induction hypothesis, which completes the proof.

Property (1) of Theorem 5.3 states that the synchronization of a process with its determined time-free projection does not restrict its behavior at all. If two processes are partially bisimilar, then their determined time-free versions are partially bisimilar as well, as stated by property (2). Note that the other direction does not hold in general.

Recall that we treat the control requirements as an external specification. Now, suppose that the desired supervised behavior is modeled as \( q \in I \). This desired behavior is achievable if there exists a supervisor \( s \in L \), such that \( p \mid s = U q \). Since Definition 5.1 requires that \( p \mid s \leq_U p \) and \( p \mid s \leq_B r \), we have that \( q \leq_U p \) and \( q \leq_B r \) are necessary conditions. As discussed above, a good candidate for the supervisor is \( s = \pi_{dtf}(q) \), since from \( q \leq_U p \) we have that \( q \mid \pi_{dtf}(q) \leq_U p \mid \pi_{dtf}(q) \), implying \( q \leq_U p \mid \pi_{dtf}(q) \) using property (1) of Theorem 5.3. Then, according to property (2) of Theorem 5.3 we have that \( \pi_{dtf}(q) \leq_U \pi_{dtf}(p) \), which implies that \( p \mid \pi_{dtf}(q) \leq_U p \mid \pi_{dtf}(p) = U p \). Next, we characterize when a desired behavior is controllable.

**Definition 5.4.** Process \( q \in I \) is controllable with respect to plant \( p \in I \) and control requirements \( r \in I \), if \( q \leq U p \), \( q \leq_B r \), and \( p \mid \pi_{dtf}(q) \leq U q \).

The following theorem states that if one process is controllable with respect to the plant and the control requirements, then there exists a (deterministic) supervisor that achieves the desired behavior when synchronized with the plant.

**Theorem 5.5.** Let \( q \in I \) be controllable with respect to a plant \( p \in I \) and control requirements \( r \in I \), then \( \pi_{dtf}(q) \) is a supervisor for \( p \) with respect to \( r \) and \( p \mid \pi_{dtf}(q) = U q \).

**Proof.** From Definition 5.4 we have that \( q \leq_U p \), \( q \leq_B r \), and \( p \mid \pi_{dtf}(q) \leq_U q \). For \( \pi_{dtf}(q) \) to be a supervisor for \( p \) with respect to \( r \) we need to show that \( p \mid \pi_{dtf}(q) \leq_U p \) and
5.3 Separation of Concerns

According to Definition 5.6 the time-abstracted simulation disregards the stochastic behavior behind the framework depicted in Figure 1 is that we wish to exploit standard supervisory control synthesis to come up with a least restrictive supervisor and, afterwards, to ensure that the performance specification is met. Thus, in the synthesis step we wish to treat stochastic behavior, i.e., Markovian transitions, as syntactic entities, towards, to ensure that the performance specification is met. Thus, in the synthesis step we wish to treat stochastic behavior, i.e., Markovian transitions, as syntactic entities, which are to be manipulated as stated by the Markovian partial bisimulation in order to preserve correct stochastic behavior. Therefore, we do not actually need any stochastic information in the control requirements, which should only ensure proper restriction of controllable events. Recall that the performance specification are specified separately, hence enabling separation of concerns. To this end, we will define the notion of time-abstracted simulation that captures the relation between the supervised plant and the control requirements.

We need some preliminary notions for the definition. By $p \xrightarrow{a} p'$ we denote the time-abstracted labeled transition relation, defined by $p \xrightarrow{\lambda_1} p_1 \xrightarrow{\lambda_2} \ldots \xrightarrow{\lambda_n} p_n \xrightarrow{a} p'$ for some $\lambda_1, \lambda_2, \ldots, \lambda_n > 0$ and $p_1, p_2, \ldots, p_n \in \mathcal{I}$ with $n \in \mathbb{N}$. By $p \xrightarrow{a}^*$ we denote the time-abstracted successful termination predicate, defined by $p \xrightarrow{\lambda_1} p_1 \xrightarrow{\lambda_2} \ldots \xrightarrow{\lambda_{n-1}} p_{n-1} \xrightarrow{\lambda_n} p_1$ for some $\lambda_1, \lambda_2, \ldots, \lambda_n > 0$ and $p_1, p_2, \ldots, p_n \in \mathcal{I}$ with $n \in \mathbb{N}$. We will use these transition relation and predicate to abstract from the Markovian transitions in the plant, so that we can establish similarity between the supervised plant and the control requirements in order to ensure that no uncontrollable transitions have been disabled.

**Definition 5.6.** A relation $R \subseteq \mathcal{I} \times \mathcal{L}$ is a time-abstracted simulation if for all $p \in \mathcal{I}$ and $q \in \mathcal{L}$ such that $(p, q) \in R$ it holds that:

1. if $p \xrightarrow{a}^*$, then $q \downarrow$;
2. if $p \xrightarrow{a} p'$ for some $a \in \mathcal{A}$, then there exists $q' \in \mathcal{L}$ such that $q \xrightarrow{a} q'$ and $(p', q') \in R$.

We say that $p \in \mathcal{I}$ can be simulated by $q \in \mathcal{L}$ while abstracting from timed behavior, notation $p \preceq_{ta} q$, if there exists a time-abstracted simulation $R$ with $(p, q) \in R$.

According to Definition 5.6 the time-abstracted simulation disregards the stochastic delays, while roughly treating the race condition as a nondeterministic choice between all labeled transitions that are reachable by the Markovian transitions that participate in the...
race. The time-abstracted simulation plays the role of the standard simulation in Definition 5.1 to give the relation between the supervised plant and the control requirements, taking into account only the ordering of events with respect to controllability.

It turns out that we can give an alternative characterization of the time-abstracted simulation in terms of the time-free projection and standard simulation.

**Theorem 5.7.** Let \( p \in \mathcal{I} \) and \( q \in \mathcal{L} \). Then, \( p \preceq_{ta} q \) if and only if \( \pi_{tf}(p) \preceq_{0} q \).

**Proof.** It is sufficient to show that \( p \downarrow^* \) and \( p \stackrel{a \cdot \pi}{\rightarrow} p' \) if and only if \( \pi_{tf}(p) \downarrow \) and \( \pi_{tf}(p) \rightarrow^a \pi_{tf}(p') \). By definition, \( p \downarrow^* \) if and only if there exist \( \lambda_1, \lambda_2, \ldots, \lambda_n > 0 \) and \( p_1, p_2, \ldots, p_n \in \mathcal{I} \) with \( n \in \mathbb{N} \) such that \( p \stackrel{\lambda_1}{\rightarrow} p_1 \stackrel{\lambda_2}{\rightarrow} \ldots \stackrel{\lambda_{n-1}}{\rightarrow} p_{n-1} \stackrel{\lambda_n}{\rightarrow} p_n \downarrow \). Reusing the results from the proof of Theorem 4.3 and following the operational semantics, we have that \( p = \emptyset \lambda_1.(\lambda_2.(\ldots(\lambda_n.(p_n + q_n)) + q_2) + q_1 \) for some \( q_1, \ldots, q_n \in \mathcal{I} \). Applying axioms TF4 and TF5 from Figure 5 we have that \( \pi_{tf}(p) = B \pi_{tf}(p_n) + \sum_{i=1}^{n} \pi_{tf}(p_i) \). Now, as \( p_n \downarrow \) we have that \( p_n = p' + 1 \) for some \( p' \in \mathcal{I} \), so \( \pi_{tf}(p) = B p' + 1 + \sum_{i=1}^{n} \pi_{tf}(p_i) \) directly implying \( \pi_{tf}(p) \downarrow \).

Similarly, for \( p \stackrel{a \cdot \pi}{\rightarrow} p' \) we have that \( p = \emptyset \lambda_1.(\lambda_2.(\ldots(\lambda_n.(a.p' + q') + q_n)) + q_2) + q_1 \), implying that \( \pi_{tf}(p) = B a.\pi_{tf}(p') + \pi_{tf}(q') + \sum_{i=1}^{n} \pi_{tf}(p_i) \) by using in addition axiom TF3 and further implying that \( \pi_{tf}(p) \rightarrow^a \pi_{tf}(p') \). Now, the claim above is obvious by comparing Definitions 4.1 and 5.6 having in mind that 3. and 4. are not applicable in Definition 4.1.

Theorem 5.7 enables us to replace the time-abstracted simulation with the time-free projection of the process and standard simulation. By applying the theorem, the definition of controllability that provides for separation of concerns employs a time-free projection of the plant.

**Definition 5.8.** Let \( p \in \mathcal{I} \) be a plant and \( r \in \mathcal{L} \) control requirements. We say that \( s \in \mathcal{L} \) is a supervisor for \( p \) that satisfies \( r \) if \( p \mid s \preceq_{ta} p \) and \( \pi_{tf}(p) \mid s \preceq_{0} r \).

Definition 5.8 will be the cornerstone for our framework as it defines the notion of controllability for Interactive Markov Chains that abstracts from the stochastic behavior in the control requirements. It enables separation of concerns as the stochastic behavior of the supervised plant is preserved with respect to the stochastic behavior of the original plant, so that one can safely proceed with performance and reliability analysis as proposed in the framework of Figure 1.

### 6 Conclusion

We proposed a process-theoretic approach to supervisory control theory of Interactive Markov Chains, an orthogonal extension of standard concurrency models with Markovian behavior. To this end, we develop a process theory based on the behavioral relation termed Markovian partial bisimulation. This relation captures the notion of controllability for nondeterministic discrete-event systems, while correctly preserving the stochastic behavior in the form of race condition between stochastic delays. We give two notions of controllability, one following a traditional approach to supervisory control and the other abstracting from stochastic behavior and catering only that no uncontrollable events are disabled.

Interactive Markov Chains are argued a natural semantic model for stochastic process...
calculi and Petri nets, making them a strong candidate for an underlying model for supervisory control of stochastic discrete-event systems with unrestricted nondeterministic. We cast our proposal as a model-based systems engineering framework and we outline the development of this framework in four phases: (1) identification of a suitable process-theoretic model and development of a corresponding notion of controllability, (2) a minimization process for the plant that respects controllability, (3) a supervisory control synthesis algorithm that tackles stochastic behavior syntactically, and (4) extraction of a directive supervisory that achieves the given performance specification. The framework will employ supervisory control algorithms to synthesize a supervisor, while abstracting from stochastic behavior, followed by an extraction of a directive supervisor that will guarantee some given performance specification. The framework enables separation of concerns and should provide for greater modeling expressivity and convenience.

In this paper, we studied in detail the first step in the development of the proposed framework. As future work, we will first develop a minimization procedure for the plant based on the mutual partial bisimilarity equivalence, and we will employ those results to efficiently synthesize a supervisor. Finally, we will develop extraction algorithms that will compute a directive supervisor that achieves the performance specification as well.

Acknowledgment

We thank Nikola Trčka for his useful comments and remarks on a preliminary draft of this paper.


