

Markov Automata: Deciding weak bisimulation by means of non-naïvely vanishing states

Johann Schuster^a, Markus Siegle^a

^a*University of the Federal Armed Forces Munich*

Abstract

This paper develops a decision algorithm for weak bisimulation on Markov Automata (MA). For this purpose, different notions of vanishing states (a concept originating from the area of Generalised Stochastic Petri Nets) are defined. In particular, non-naïvely vanishing states are shown to be essential for relating the concepts of (state-based) naïve weak bisimulation and (distribution-based) weak bisimulation. The bisimulation algorithm presented here follows the partition-refinement scheme and has exponential time complexity.

Keywords: Markov automata, weak bisimulation, vanishing state, elimination

1. Introduction

Markov Automata (MA) are a powerful formalism for modelling systems with nondeterminism, probability and continuous time. The weak bisimulation relation for MA [1, 2] is not a relation on the set of states, but rather a relation on the set of subdistributions over states. This is the reason, why it is not obvious how to develop an algorithm for deciding distribution-based weak bisimulation for MA, and this is exactly the topic of the present paper.

Our approach carries over some intuition from the area of Generalised Stochastic Petri Net [3] to the MA setting. There, vanishing markings are eliminated in order to minimise the number of reachable markings and to enable the subsequent steps of numerical analysis. A basic example of a GSPN is given in Fig. 1. It consists of the places p_1 to p_4 , an exponentially distributed transition t_1 and the immediate transitions t_2 and t_3 . We assume that the weights of the immediate transitions have already been transformed

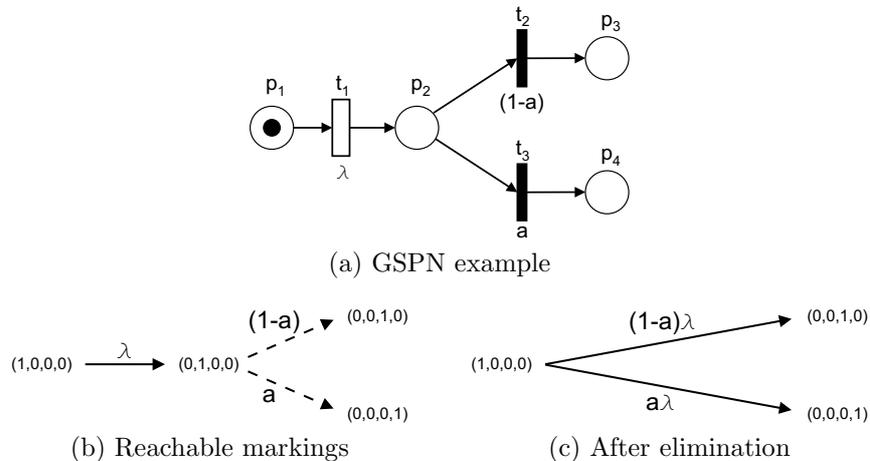


Figure 1: Example of a GSPN

to probabilities. The resulting Labelled Transition System – including both exponential and probabilistic transitions – of the reachable markings is shown in Fig. 1b: the solid arc defines an exponential transition with rate λ , the dashed arcs denote the immediate transitions driven by probabilities $(1 - a)$ and (a) . After elimination of marking $(0, 1, 0, 0)$ we obtain the transition system in Fig. 1c. In GSPN terminology, the state corresponding to marking $(0, 1, 0, 0)$ is called “vanishing”, whereas in this paper, where we develop a more detailed classification of vanishing states, it will be denoted *trivially vanishing*.

With this intuition, we are able to define vanishing states in the nondeterministic context of MA. We provide a topological characterisation of a special kind of states that is equivalent to a “real” distribution, i.e. a distribution consisting of at least two different classes with respect to some equivalence relation. It will turn out that this characterisation, that we call *non-naïvely vanishing* (*nn-vanishing*), is sufficient for calculating state-minimal normal forms of Markov Automata. With the aid of this characterisation we are able to give a decision algorithm for weak MA bisimulation.

In contrast to distribution-based weak bisimulation, decision algorithms for naïve weak bisimulation on MA have been known for some time. Since naïve weak bisimulation on MA [1] corresponds to weak probabilistic bisimulation on Probabilistic Automata (PA), naïve weak MA bisimulation is known to be decidable since 2002 [4]. There, an exponential time algorithm was pre-

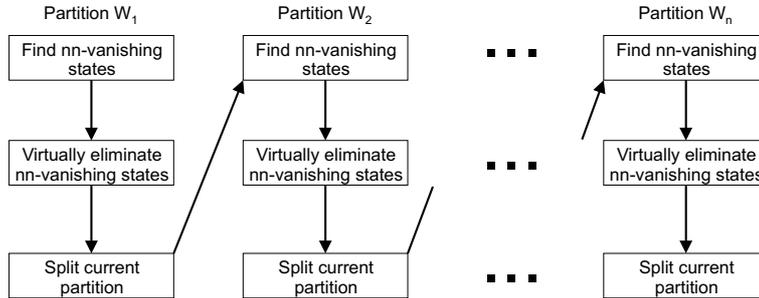


Figure 2: Proposed algorithm

sented. In 2012 a polynomial time algorithm has been presented for deciding naïve weak MA bisimulation [5].

Our algorithm is built upon the algorithm in [4] which is a partition refinement algorithm. The main difference is that for every partition of the state space we first identify nn-vanishing classes of states. Before we split the current partition, we “virtually” eliminate all states that belong to nn-vanishing classes, i.e. we only consider restricted probability distributions where the probability of every nn-vanishing state is equal to zero. On this “reduced” transition system, we run the algorithm of [4] (on *all* states, considering also nn-vanishing states which are sufficiently identified by the not nn-vanishing states they can reach). This basic scheme of the algorithm is depicted in Fig. 2.

Our algorithm has exponential time complexity (the result of [5] does not seem to be applicable to the weak case, as the naïve weak bisimulation problems after speculative eliminations have an exponential number of transitions in contrast to the original weak bisimulation problem).

By its generality, using only minor changes it can also be applied to the case of the MA bisimulation recently defined in [6].

More or less at the same time to our approach [7], a different approach for a decision algorithm has been given in [8]. A comparison of the two approaches is given in Sec. 8.

This paper is a major rework of our report [7]. While the algorithm is completely the same, the correctness proofs of the previous paper relied heavily on results of [1, 2]. The present paper provides a new line of argumentation and new proofs, independent of Thm. 2 of [1, 2].

The paper is organised as follows: In Sec. 2 we present the necessary pre-

liminaries and recall a mapping from MA to PA from [1]. Sec. 3 recapitulates some facts on weak and naïve weak bisimulation for MA. In Sec. 4 we define different notions of vanishing states and use them to relate weak bisimulation and naïve weak bisimulation. Sec. 5 discusses properties of vanishing states and provides the main theorems. Sec. 6 describes our decision algorithm for weak MA bisimulation that heavily relies on Sec. 4 and Sec. 5. In Sec. 7 we briefly discuss the applicability of our concepts and our decision algorithm to the weak bisimulation published in [6]. In Sec. 8 we compare our concepts to a recently published alternative approach [8] for deciding weak MA bisimulation. Finally, Sec. 9 concludes the paper.

2. Preliminaries

This section introduces some common notations on distributions, defines Markov Automata following [1, 2] and recalls the mapping from Markov Automata to probabilistic automata which is used in Sec. 3 to define weak bisimulation for Markov Automata.

2.1. Probability (Sub-)Distributions

First we define the notion of discrete subdistribution and related terms and notations: A mapping $\mu : S \rightarrow [0, 1]$ is called (discrete) subdistribution, if $\sum_{s \in S} \mu(s) \leq 1$. As usual we write $\mu(S')$ for $\sum_{s \in S'} \mu(s)$. The *support* of μ is defined as $Supp(\mu) := \{s \in S \mid \mu(s) > 0\}$. The empty subdistribution μ_\emptyset is defined by $Supp(\mu_\emptyset) = \emptyset$. The *size* of μ is defined as $|\mu| := \mu(S)$. A subdistribution μ is called *distribution* if $|\mu| = 1$. The sets $Dist(S)$ and $SubDist(S)$ denote distributions and subdistributions defined over the set S . Let $\Delta_s \in Dist(S)$ denote the *Dirac* distribution on s , i.e. $\Delta_s(s) = 1$. For two subdistributions μ, μ' the sum $\mu'' := \mu \oplus \mu'$ is defined as $\mu''(s) := \mu(s) + \mu'(s)$ (as long as $|\mu''| \leq 1$). As long as $c \cdot |\mu| \leq 1$, we denote by $c\mu$ the subdistribution defined by $(c\mu)(s) := c \cdot \mu(s)$. For a subdistribution μ and a state $s \in Supp(\mu)$ we define $\mu - s$ by

$$(\mu - s)(t) = \begin{cases} \mu(t) & \text{for } t \neq s \\ 0 & \text{for } t = s \end{cases}$$

Occasionally, we will also need the lifting of relations to distributions:

Definition 1 (Lifting of equivalence relations to distributions). *An equivalence relation $R \subseteq S \times S$ is lifted to $Dist(S) \times Dist(S)$ in the following way:*

For $\mu, \gamma \in \text{Dist}(S)$ we write $\mu \equiv_R \gamma$ (or simply, by abuse of notation, $\mu R \gamma$) if and only if for each equivalence class $C \in S/R : \mu(C) = \gamma(C)$.

2.2. Markov and Probabilistic Automata

The definition of Markov Automata we use is the one from [1, 2].

Definition 2 (Markov Automata [1]). *A Markov automaton MA is a tuple $(S, \text{Act}, \rightarrow, \twoheadrightarrow, s_0)$, where*

- S is a nonempty finite set of states,
- Act is a set of actions containing the internal action τ ,
- $\rightarrow \subseteq S \times \text{Act} \times \text{Dist}(S)$ a set of action-labelled probabilistic transitions,
- $\twoheadrightarrow \subseteq S \times \mathbb{R}_{\geq 0} \times S$ a set of Markovian timed transitions and
- $s_0 \in S$ the initial state.

A state in a MA is called *stable* if it has no emanating τ transitions, otherwise it is called *unstable*. A stable state s will be denoted by $s\downarrow$.

In order to make our decision algorithm feasible we assume in the following that, in contrast to the original definition from [1, 2], *all* sets in Definition 2 are finite. This means that there are finitely many states, finitely many actions and finitely many transitions.

For simplicity we define probabilistic automata (PA) in terms of MA.

Definition 3. *A probabilistic automaton (PA) is a MA $P = (S, \text{Act}, \rightarrow, \emptyset, s_0)$. We also write $P = (S, \text{Act}, \rightarrow, s_0)$ if the context is clear.*

This definition corresponds to a *simple* probabilistic automaton in the sense of Segala [9].

For the mapping from MA to PA introduced in [1] we need to define the probability distribution on successor states. In contrast to [1, 2, 10], our definition of successor distribution also takes care of the case $\text{rate}(s) = 0$.

Definition 4 (modified¹ version of Definition 3 in [1]). Let $M = (S, Act, \rightarrow, \twoheadrightarrow, s_0)$ be a MA. Define

$$rate(s, s') := \sum_{(s, \lambda, s') \in \twoheadrightarrow} \lambda$$

and $rate(s) := \sum_{s' \in S} rate(s, s')$ which is called the exit rate of state s . The probability distributions P_s are defined in the following way:

$$P_s := \begin{cases} s' \mapsto \frac{rate(s, s')}{rate(s)} & \text{for } rate(s) \neq 0 \\ \Delta_s & \text{otherwise} \end{cases}$$

2.3. A mapping from MA to PA

The remarkable idea of [1] is to define bisimulations on MA using a mapping from MA to PA. The basic ingredient is a set of special actions, denoted by $\chi(\cdot)$, that cover timed behaviour. In the setting of [1, 2] countable action sets are mapped to uncountable action sets by definition, as for every real number a new action name is introduced. In order to keep the action set finite to retain algorithmic tractability, we redefine Act^x in the context of a fixed MA:

Definition 5. Let $M = (S, Act, \rightarrow, \twoheadrightarrow, s_0)$ be a MA. Assume $\forall r \in \mathbb{R}_{\geq 0} \chi(r) \notin Act$ and define $\mathcal{RT} := \{rate(s) | s \in S\}$ (which is finite). Then we define $Act^{\chi(\mathcal{RT})} = \{\chi(r) | r \in \mathcal{RT}\}$ and $Act^x := Act \cup Act^{\chi(\mathcal{RT})}$.

There is a mapping from MA to PA (adapted from [1]) where we use Definition 4:

Definition 6. Let $M = (S, Act, \rightarrow, \twoheadrightarrow, s_0)$ be a MA. Define the transitions \rightarrow as follows: For $s \in S$ define

$$s \xrightarrow{\alpha} \mu \text{ if } \begin{cases} \alpha \in Act \text{ and } s \xrightarrow{\alpha} \mu \\ s \downarrow, \alpha = \chi(rate(s)) \in Act^{\chi(\mathcal{RT})} \text{ and } \mu = P_s \end{cases}$$

Then the mapping $\mathcal{PA} : MA \rightarrow PA$ is defined by $M \mapsto (S, Act^x, \rightarrow, \emptyset, s_0)$.

¹The original definition from [1, 2, 10] is problematic, as for $rate(s) = 0$ the fraction $\frac{0}{0}$ is not defined (this case is treated separately in our definition), and for infinite sets \twoheadrightarrow the exit rate may not converge (this case is not problematic for us, as we deal with *finite* sets). Both issues have no impact on the decision algorithm presented here, but the first issue has an impact on the compositionality of MA bisimulation in general. For a detailed explanation of why compositionality is lost with the original definitions of [1, 2, 10] we refer to Appendix A of [7].

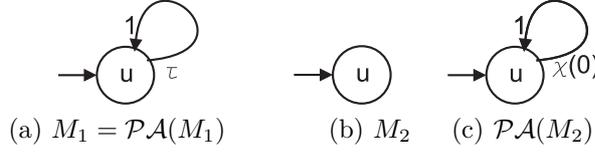


Figure 3: MA to PA transformations

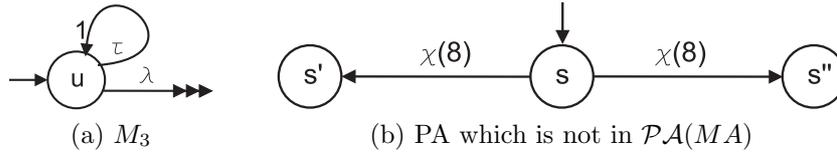


Figure 4: MA to PA transformations - continued

Note that every timed transition is part of a special $\chi(\cdot)$ action. So the set of actions is increased by the mapping $\mathcal{PA}(\cdot)$, but no timed transition remains in the image. For more details on the procedure we refer to [1].

Example 1. Two special cases are given in Fig. 3. M_1 (Fig. 3a) has a τ loop, so no timed transition (i.e. $\chi(\cdot)$) exists after the transformation, i.e. $\mathcal{PA}(M_1) = M_1$. In the example M_2 (Fig. 3b) u is a stable state and therefore the transformation leads to a χ transition with exit rate 0 (Fig. 3c).

Lemma 1. The mapping $\mathcal{PA} : MA \rightarrow PA$ is neither injective nor surjective.

Proof. It is not surjective, as a Markovian race condition is always converted to a deterministic χ transition. That means for example the PA in Fig. 4b is not in $\mathcal{PA}(MA)$. It is also not injective, as M_3 in Fig. 4a is not equal to M_1 in Fig. 3a, but $\mathcal{PA}(M_1) = \mathcal{PA}(M_3) = M_1$. \square

2.4. Weak transitions

In the following we use the definitions and terminology of [11], but we leave out the definitions for labelled transition systems. Given a transition $tr = (s, a, \mu)$, we denote s by $source(tr)$ and μ by μ_{tr} . Consider a PA $P = (S, Act, T, \emptyset, s_0)$ (with transition relation T). An execution fragment of P is a finite or infinite sequence $\alpha = q_0 a_1 q_1 a_2 q_2 \dots$ of alternating states and actions, starting with a state and, if the sequence is finite, ending in a state, where each $(q_i, a_{i+1}, \mu_{i+1}) \in T$ and $\mu_{i+1}(q_{i+1}) > 0$. State q_0 , the first state

of α , is denoted by $fstate(\alpha)$. If α is a finite sequence, then the last state of α is denoted by $lstate(\alpha)$. An *execution* of P is an execution fragment of P where $q_0 = s_0$. We let $frags(P)$ denote the set of execution fragments of P and $frags^*(P)$ the set of finite execution fragments of P . Similarly, we let $execs(P)$ denote the set of executions of P and $execs^*(P)$ the set of finite executions. Execution fragment α is a *prefix* of execution fragment α' , denoted $\alpha \leq \alpha'$, if sequence α is a prefix of sequence α' .

The *trace* of an execution fragment α , written $trace(\alpha)$, is the sequence of actions obtained by restricting α to the set of external actions, i.e. $Act \setminus \{\tau\}$. For a set E of executions of a PA P , $traces(E)$ is the set of traces of the executions in E . We say that β is a trace of a PA P if there is an execution α of P with $trace(\alpha) = \beta$. Let $traces(P)$ denote the set of traces of P .

A *scheduler* for a PA P is a function $\sigma : frags^*(P) \rightarrow \text{SubDist}(T)$ such that $tr \in \text{supp}(\sigma(\alpha))$ implies that $source(tr) = lstate(\alpha)$. This means that the image $\sigma(\alpha)$ is a *discrete* subdistribution over transitions. The defect of the subdistribution, i.e. $1 - |\sigma(\alpha)|$ is used for stopping in the current state. A scheduler σ is said to be *deterministic* if for each finite execution fragment α either $\sigma(\alpha)(T) = 0$ or $\sigma(\alpha) = \Delta_{tr}$ (Dirac measure for tr) for some $tr \in T$. In other words, a deterministic scheduler is the entity that resolves nondeterminism in a probabilistic automaton by choosing randomly either to stop or to perform one of the transitions that are enabled from the current state. A scheduler is called *memoryless* if it depends only on the last state of its argument, that is, for each pair α_1, α_2 of finite execution fragments, if $lstate(\alpha_1) = lstate(\alpha_2)$, then $\sigma(\alpha_1) = \sigma(\alpha_2)$. A scheduler is called *determinate* if its choice depends only on the current trace and on the last state of its argument, that is, for each pair α_1, α_2 of finite execution fragments, if $trace(\alpha_1) = trace(\alpha_2)$ and $lstate(\alpha_1) = lstate(\alpha_2)$, then $\sigma(\alpha_1) = \sigma(\alpha_2)$. Following [4] we call a deterministic determinate scheduler a *Dirac determinate* scheduler.

A scheduler σ and a discrete initial probability measure $\mu_0 \in \text{Dist}(S)$ induce a measure ϵ on the sigma-field generated by cones of execution fragments as follows. If α is a finite execution fragment, then the *cone* of α is defined by $C_\alpha = \{\alpha' \in frags(P) | \alpha \leq \alpha'\}$. The measure ϵ of a cone C_α is defined recursively: If $\alpha = s$ for some $s \in S$ we define $\epsilon(C_\alpha) = \mu_0(s)$. If α is of the form $\alpha'a's'$, $\epsilon(C_\alpha)$ is defined by the equation

$$\epsilon(C_\alpha) = \epsilon(C_{\alpha'}) \cdot \sum_{tr \in T(a')} \sigma(\alpha')(tr) \mu_{tr}(s'),$$

where $T(a')$ denotes the set of transitions of T that are labelled by a' . Standard measure theoretical arguments ensure that ϵ is well defined. We call the measure ϵ a probabilistic execution fragment of P , and we say that ϵ is generated by σ and μ_0 .

Consider a probabilistic execution fragment ϵ of a PA P , with first state s , i.e. $\mu_0 = \Delta_s$, that assigns probability 1 to the set of all finite execution fragments α with trace $trace(\alpha) = \beta$ for some $\beta \in (Act \setminus \{\tau\})^*$. Let μ be the discrete measure defined by $\mu(s') = \epsilon(\{\alpha \mid lstate(\alpha) = s'\})$. Then $s \xrightarrow{\beta}_C \mu$ is a *weak combined transition* of P . We call ϵ a *representation* of $s \xrightarrow{\beta}_C \mu$. If $s \xrightarrow{\beta}_C \mu$ is induced by a deterministic scheduler, we also write $s \xrightarrow{\beta} \mu$. In case $trace(\alpha)$ is empty we write $s \xrightarrow{\tau}_C \mu$.

Let $\{s \xrightarrow{a} \mu_i\}_{i \in I}$ be a collection of transitions of a PA P , and let $\{c_i\}_{i \in I}$ be a collection of probabilities such that $\sum_{i \in I} c_i = 1$. Then the triple $(s, a, \sum_{i \in I} c_i \mu_i)$ is called a (*strong*) *combined transition* of P and we write $s \xrightarrow{a}_C \sum_{i \in I} c_i \mu_i$. We say that there is a *hyper-transition* from $\mu \xrightarrow{a}_C \mu'$, if there exists a family of weak combined transitions $\{s \xrightarrow{a}_C \mu_s\}_{s \in Supp(\mu)}$ such that $\mu' = \sum_{s \in Supp(\mu)} \mu(s) \cdot \mu_s$.

3. Relating naïve weak & weak bisimulation

Remember that for a MA M its transitions have been defined by means of $\mathcal{PA}(M)$, so in the following it is safe to assume that all Markov Automata are represented by their PA images. All calculations will be made in this context.

Note that, in contrast to the transition tree notation of [1, 2, 12], we do not need the notation $\xrightarrow{\hat{\alpha}}$ (which includes the possibility of zero steps in the case $\alpha = \tau$) as our definition of $\xrightarrow{\alpha}$ also includes this case².

The relation defined in the following is called “weak probabilistic bisimulation” [4] in the context of PA:

Definition 7 (Naïve weak bisimulation in the spirit of [13]). *An equivalence relation \mathcal{R} on the set of states S of a MA $M = (S, Act, \xrightarrow{\cdot}, \xrightarrow{\cdot}, s_0)$ is called naïve weak bisimulation if and only if $x \mathcal{R} y$ implies for all $\alpha \in Act^*$: $(x \xrightarrow{\alpha} \mu)$ implies $(y \xrightarrow{\alpha}_C \mu')$ with $\mu \equiv_R \mu'$ (note that the transitions are regarded in $\mathcal{PA}(M)$). If x and y are contained in a naïve weak bisimulation relation, we*

²Definition 10 in [1, 2] erroneously uses α instead of $\hat{\alpha}$.

write $x \approx_{\text{naïve}} y$. Two MA are called naïvely weakly bisimilar if their initial states are related by a naïve weak bisimulation relation on the direct sum of their states.

We would like to mention that modulo naïve weak bisimulation it is possible to omit τ -loops in the image $\mathcal{PA}(\cdot)$. The property whether a state is stable or unstable can still be recovered by looking for the presence (or absence) of χ transitions.

The authors of [1, 2] argued that the (state-based) notion of naïve weak bisimulation is too fine. Therefore they defined the coarser notion of (distribution-based) weak bisimulation:

Definition 8 (Weak bisimulation [1, 2]). *A relation \mathcal{R} on sub-distributions over the set of states S of a MA $M = (S, \text{Act}, \rightarrow, \twoheadrightarrow, s_0)$ is called weak bisimulation if for all $(\mu_1, \mu_2) \in \mathcal{R}$ it holds that (transitions regarded in $\mathcal{PA}(M)$)*

A.) $|\mu_1| = |\mu_2|$

B.) $\forall t \in \text{Supp}(\mu_1), \forall \alpha \in \text{Act}^x : \exists \mu_2^g, \mu_2^s : \mu_2 \Rightarrow_C \mu_2^g \oplus \mu_2^s$ such that

(i) $(\mu_1(t) \cdot \Delta_t) \mathcal{R} \mu_2^g$ and $(\mu_1 - t) \mathcal{R} \mu_2^s$

(ii) whenever $(t \xrightarrow{\alpha} \mu'_1)$, then $(\exists \mu'_2 : \mu_2^g \xrightarrow{\alpha}_C \mu'_2)$ and $(\mu_1(t) \cdot \mu'_1) \mathcal{R} \mu'_2$

C.) a symmetric condition with μ_1 and μ_2 interchanged (roles of left-hand side and right-hand side also interchanged)

Two distributions μ, γ are called weakly bisimilar (with respect to some MA M), written $\mu \approx \gamma$, if the pair (μ, γ) is contained in a weak bisimulation relation (with respect to M). Two states are called weakly bisimilar if their corresponding Dirac distributions are weakly bisimilar. We write $s \approx_{\Delta} t$ for $\Delta_s \approx \Delta_t$. Two MA are called weakly bisimilar if their initial states are weakly bisimilar in the direct sum of the MA.

Thm. 1 in [1, 2] shows that \approx is an equivalence relation (and therefore also \approx_{Δ}).

The following statement is a corollary of Thm. 2 in [1, 2]. As we show in Appendix A that Lemma 16 in [2] (and therefore also Thm. 2 in [1, 2]) must be considered unproven, we give an independent proof of the following statement.

Lemma 2 (Corollary of Thm. 2 in [1, 2]). *If two MA are naïvely weakly bisimilar, they are also weakly bisimilar.*

Proof. We provide a direct proof of this statement. Let $P = (S, Act^X, \rightarrow, \emptyset, s_0)$, $P' = (S', Act^X, \rightarrow', \emptyset, s'_0)$. We directly construct a weak bisimulation relation R' out of a given naïve weak bisimulation relation $R \subseteq S \times S'$. $R' = \{(\mu_1, \mu_2) \mid \mu_1 \in \text{SubDist}(S), \mu_2 \in \text{SubDist}(S'), \mu_1 \equiv_R \mu_2\}$. We show that this is indeed a weak bisimulation. To simplify matters, we assume that we work on the quotient with respect to $\approx_{\text{naïve}}$ of the direct sum of S and S' . Now we may verify the condition for weak bisimulation. Assume that $(\mu_1, \mu_2) \in R'$. By $\mu_1 \equiv_R \mu_2$ it is clear that $|\mu_1| = |\mu_2|$. Now choose an arbitrary $s \in \text{Supp}(\mu_1)$ and let $\mu_1(s) = c$. Again by $\mu_1 \equiv_R \mu_2$ we find also $s \in \text{Supp}(\mu_2)$ and we may choose $\mu_2^g = c\Delta_s$ (the fact that we may consider the same state s in both $\text{Supp}(\mu_1)$ and $\text{Supp}(\mu_2)$ holds only for quotients – in the general case we must find bisimilar states with the same probability mass c). Of course also the remaining parts $\mu_1 - s$ and μ_2^s (where $\mu_2 = \mu_2^g \oplus \mu_2^s$) must be in relation, as $\mu_1 \equiv_R \mu_2$. Finally it is clear that whenever $s \xrightarrow{a} \mu_1'$, then also $\mu_2^g \xrightarrow{a}_C \mu''$ with $(c \cdot \mu_1', \mu'') \in R'$ by naïve weak bisimilarity and the construction of R' . The same holds with the roles of μ_1 and μ_2 interchanged, so the claim is shown. \square

A precalculation of the elimination procedure presented later in this paper is the “rescaling” procedure.

Remark 1. *Combined transitions \Rightarrow_C can be used to rescale loops. A basic example is given in Fig. 5. A Dirac determinate scheduler choosing the transition $s \xrightarrow{\tau} \frac{1}{3}\Delta_x \oplus \frac{1}{3}\Delta_s \oplus \frac{1}{3}\Delta_y$ with probability one and stopping in states x and y leads to the weak transition $s \xrightarrow{\tau} \frac{1}{2}\Delta_x \oplus \frac{1}{2}\Delta_y$. To mimic the τ transition of s , t has to perform a transition combined of $\frac{2}{3}$ times $t \rightarrow \frac{1}{2}\Delta_x \oplus \frac{1}{2}\Delta_y$ and $\frac{1}{3}$ times $t \Rightarrow \Delta_t$. Without using combined transitions t could not mimic this transition of s .*

For many of the proofs in this paper we are only interested in properties “up to an equivalence relation R ”, which motivates the following definition. In the quotient automaton, no two distinct states s, t exist with sRt .

Definition 9 (Quotient automaton). *Let $P = (S, Act, T, \emptyset, s_0)$ be a PA and R an equivalence relation over S . The equivalence class of a state s is denoted*

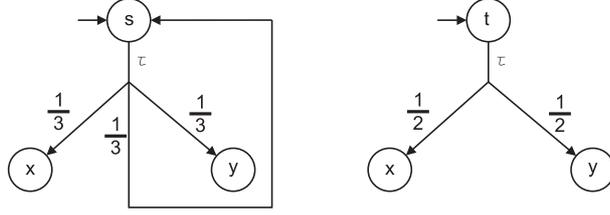


Figure 5: Resolving a τ loop

by $[s]_R$ (when the context is clear we write $[s]$ instead of $[s]_R$). We write P/R to denote the quotient automaton of P with respect to R , that is

$$P/R = (S/R, Act, T/R, [s_0]_R)$$

with $T/R \subseteq S/R \times Act \times \text{Dist}(S/R)$ such that $([s]_R, a, \mu) \in T/R$ if and only if there exists a state $s' \in [s]_R$ such that $(s', a, \mu') \in T$ and $\forall [t]_R \in S/R : \mu([t]_R) = \sum_{t' \in [t]_R} \mu'(t')$. We call an automaton a quotient with respect to R or, if R is clear, just a quotient, if it holds that P and P/R coincide (up to a renaming of the set of states).

Definition 10 (Reachable states). Let $P = (S, Act, T, \emptyset, s_0)$ be a PA, $S' \subseteq S$ its set of reachable states, i.e. those states that can be reached with non-zero probability by a scheduler starting from s_0 . Let $T' := T|_{S' \times Act \times \text{Dist}(S')}$ be the restriction of the transition relation to S' . We define $r(P) := (S', Act, T', s_0)$ and call it the reachable fragment of P .

As bisimulation only focuses on the reachable fragment of the state space, we assume from now on that by *quotient* we always mean the reachable part of the quotient, i.e. $r(P/R)$.

4. Vanishing states and vanishing representations

We now introduce a notion of vanishing states in the context of MA.

Definition 11. Given a PA $P = (S, Act, T, \emptyset, s_0)$, and $t \notin S$, we define the following renamings:

- for each $v \in S$,

$$v[t/s] = \begin{cases} t & \text{if } v = s, \\ v & \text{otherwise.} \end{cases}$$

The set of all renamed states is denoted by $S[t/s]$;

- for each $\nu \in \text{Dist}(S)$ and $v \in S[t/s]$,

$$\nu[t/s](v) = \begin{cases} \nu(s) & \text{if } v = t, \\ \nu(v) & \text{otherwise;} \end{cases}$$

- for each $(v, a, \nu) \in T$, $(v, a, \nu)[t/s] = (v[t/s], a, \nu[t/s])$. The set of all renamed transitions is denoted by $T[t/s]$.

Definition 12 (Emanating Internal Weak Combined Transitions). Given a PA $P = (S, \text{Act}, T, \emptyset, s_0)$ and a state $s \in S$, we denote by $\mathfrak{C}(s)$ the set $\{(s, \tau, \nu) \mid s \xrightarrow{\tau}_C \nu\}$ of internal weak combined transitions emanating from s . Further, we denote by $\mathfrak{S}(s)$ the set $\{(s, a, \nu) \mid s \xrightarrow{a} \nu, a \in \text{Act}\}$ of strong transitions emanating from s .

Definition 13 (local change of transitions). Let $P = (S, \text{Act}, T, \emptyset, s_0)$ be a PA and $s \in S$.

For any $\mathfrak{T} \subseteq \mathfrak{C}(s)$ we define the PA

$$P(\mathfrak{T}) = (S, \text{Act}, (T \setminus \mathfrak{S}(s) \cup \mathfrak{T}), \emptyset, s_0).$$

If $\mathfrak{T} = \{(s, \tau, \nu)\}$ we also write $P_{(s, \nu)}$ – or simply P' if the context is clear – instead of $P(\{(s, \tau, \nu)\})$.

Definition 14 (vanishing states). Let $P = (S, \text{Act}, T, \emptyset, s_0)$ be a PA. Let $s \in S$ be unstable and $\mathfrak{C}(s)$ as in Def. 12. State s is called

trivially vanishing if $\mathfrak{S}(s) = \{(s, \tau, \nu)\}$ for some $\nu \in \text{Dist}(S)$.

vanishing if there exists $(s, \tau, \nu) \in \mathfrak{C}(s)$ such that $s \approx_{\Delta} t$ when comparing $P[t/s]$ and $P_{(s, \nu)}$ for $t \notin S$. In this case $P_{(s, \nu)}$ – or (s, ν) , for short – is called a vanishing representation of s .

non-naïvely vanishing or nn-vanishing, for short, if it is vanishing and there is a vanishing representation $P_{(s, \nu)}$ such that there exists $t \in \text{Supp}(\nu)$ such that $s \not\approx_{\Delta} t$.

naïvely vanishing if it is vanishing, but not nn-vanishing, i.e. for all vanishing representations $P_{(s, \nu)}$ it holds that for all $t \in \text{Supp}(\nu) : s \approx_{\Delta} t$.

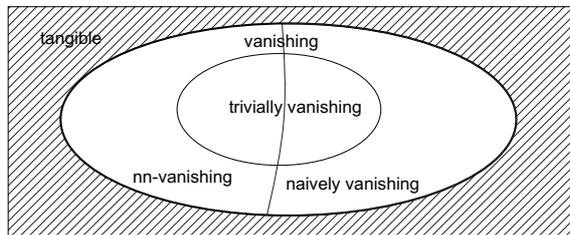


Figure 6: Partition of state space into vanishing and tangible states

A state that is not vanishing is called tangible. A state that is not nn-vanishing is called nn-tangible³.

Trivially vanishing states correspond to vanishing markings in GSPNs [14], provided that they are well-defined, i.e. there is no non-determinism [15]. Vanishing states extend this idea to the presence of non-determinism: for a vanishing state, emanating non-deterministic transitions can be bisimilarly reduced to a single deterministic transition. Non-naïvely vanishing states can be transformed to a distribution, such that the equivalence class with respect to \approx_{Δ} changes. It will turn out that this is essentially the difference to state-based bisimulations.

According to the definition, the set of states can be partitioned in two ways: vanishing vs. tangible, or nn-vanishing vs. nn-tangible. This classification of states is illustrated in Fig. 6 and Fig. 7. By definition, any vanishing representation of a naïvely vanishing state cannot change the equivalence class with respect to \approx_{Δ} . For that reason, naïvely vanishing states do not have to be eliminated in order to reduce the problem to naïve bisimulation. We will show later that the only obstacle are nn-vanishing states. The next example shows basic representatives of the different types of vanishing states.

Example 2. Assume that $p \in (0, 1)$. State E in Fig. 8a is trivially vanishing since it only has an emanating τ transition, and it is naïvely vanishing, as it holds that $E \approx_{\Delta} C \approx_{\Delta} D$. A non-trivially and naïvely vanishing state E is given in Fig. 8b. Note that all non- τ transitions emanating from E may be omitted as they can be mimicked by appropriate weak transitions.

³Note that in [7] we used the term “tangible” for describing “nn-tangible”. For a more readable notation we distinguish now between tangible and nn-tangible.

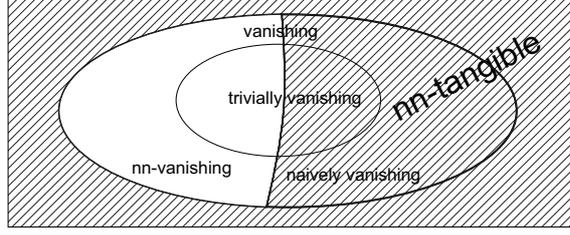


Figure 7: Partition of state space into nn-vanishing and nn-tangible states

The automaton in Fig. 8a is the corresponding vanishing representation, so naïvety follows as in this case. E is naïvely vanishing as it turns out to be in the same class as C and D modulo weak bisimulation. For the next example, first note that in Fig. 8c C and D cannot be weakly bisimilar (because C can only perform the c to A , while D can additionally perform the d to B). As E is trivially vanishing we notice that it is also nn-vanishing, because E moves to the distribution $p\Delta_C \oplus (1-p)\Delta_D$, where $C \not\approx D$. Moreover, since in Fig. 8c D is not vanishing (and E is nn-vanishing), we have that $E \not\approx D$. In the last example in Fig. 8d we see that E is not trivially vanishing as there is more than one emanating transition. Still it is easy to verify that the automaton in Fig. 8c is a vanishing representation, as the Dirac determinate scheduler choosing the transition $E \rightarrow p\Delta_C \oplus (1-p)\Delta_D$ with probability 1, $D \rightarrow \Delta_E$ with probability 1, and stopping in all other states realises the transition $E \xrightarrow{\tau} \Delta_C$.

Definition 15 (Elimination of vanishing states). Let $P = (S, Act, \rightarrow, \emptyset, s_0)$ be a PA. Let $s \in S$ be a vanishing state and let $s \xrightarrow{\tau_{van}} \nu$ be the only transition emanating from s in the vanishing representation $P' = P_{(s,\nu)} = (S, Act, \rightarrow_{van}, \emptyset, s_0)$. The elimination of s is defined by two steps:

1. Rescaling (cf. Remark 1):

$$\rightarrow_{res} = \begin{cases} \rightarrow_{van} \setminus \{(s, \tau, \nu)\} & \text{if } \nu = \Delta_s \\ (\rightarrow_{van} \setminus \{(s, \tau, \nu)\}) \uplus \{(s, \tau, \frac{1}{1-\nu(s)}(\nu - s))\} & \text{otherwise} \end{cases}$$

2. Elimination (only performed if after rescaling a transition $s \xrightarrow{\tau_{res}} \nu_{res}$

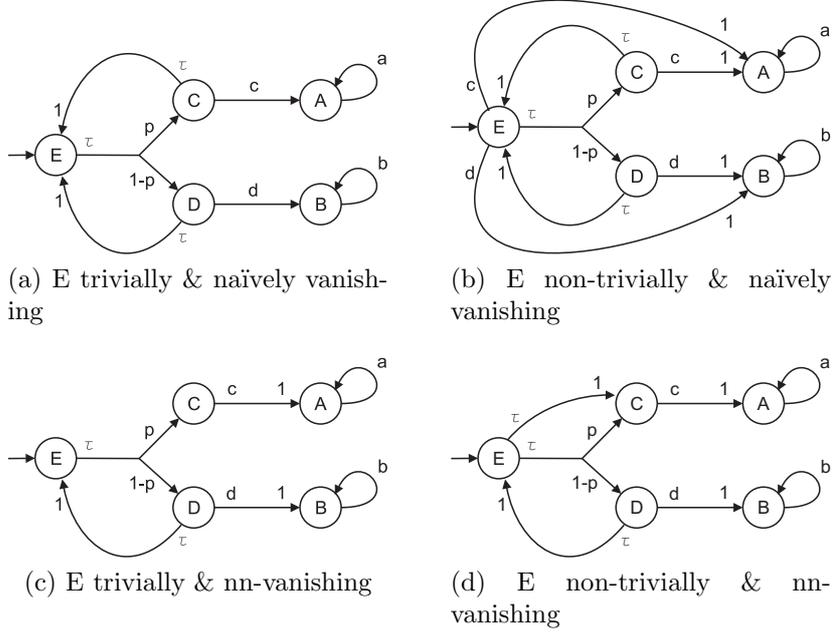


Figure 8: Examples of vanishing states

remains):

$$P^{\hat{s}} = \begin{cases} (S \setminus \{s\}, Act, \rightarrow_{el}, \emptyset, s_0) & \text{if } s \neq s_0 \\ ((S \setminus \{s_0\}) \uplus \{s_0^\circ\}, Act, \rightarrow_{el} \uplus \{(s_0^\circ, \tau, \nu_{res})\}, \emptyset, s_0^\circ) & \text{if } s = s_0 \text{ and} \\ & \exists t \rightarrow_{res} \gamma : s_0 \in \text{Supp}(\gamma) \\ P' & \text{otherwise} \end{cases}$$

where $\rightarrow_{el} := \{(t, \alpha, \mu') \mid t \xrightarrow{\alpha}_{res} \mu, t \in S \setminus \{s\}, \mu' := \mu_{s \rightarrow \nu_{res}}\}$. Here $\mu_{s \rightarrow \nu_{res}}$ denotes the replacement of every occurrence of s by the corresponding distribution ν_{res} :

Without loss of generality let μ be of the form $\mu := c_s \Delta_s \oplus (\oplus_{i \in I, s_i \neq s} c_i \Delta_{s_i})$ and ν be of the form $\nu = \oplus_{j \in J, s_j \neq s} d_j \Delta_{s_j}$. Then we define $\mu_{s \rightarrow \nu} := c_s (\oplus_{j \in J, s_j \neq s} d_j \Delta_{s_j}) \oplus (\oplus_{i \in I, s_i \neq s} c_i \Delta_{s_i})$.

We omit the (s, τ, Δ_s) transition from the set of transitions for the purpose of minimality of the resulting PA. One could also just add the loop case to the case where P' is not changed. Note that even when loops are removed, all information about the MA may be safely recovered. Such a state without

loop is a deadlock in PA and no longer vanishing according to our definition (note that there cannot be any other competing transition, as we then could not get the vanishing representation with the τ loop). Looking back to the MA setting it is clear that s must be an unstable state as it does not have the $\chi(0)$ transition. We will see later when we describe the decision algorithm that we basically treat *all* nn-vanishing states as if they were starting states, i.e. we consider them as transient copies.

Example 3. *To explain Definition 15 we give the following examples. The first example is the most common one (cf. Fig. 9a): The vanishing state s is neither the initial state nor does it have a probability-one-self-loop. Therefore the elimination is straightforward: Redirect all incoming arcs according to the vanishing representation (cf. Fig. 9b). The next example is the probability-one-self-loop case (cf. Fig. 9c): It does not matter whether the vanishing state s is the initial state or not, the self-loop is removed by the rescaling operation (cf. 9d). In the third example we have a vanishing initial state with incoming transition(s) (cf. Fig. 9e). We add a copy s_0^o of the initial state and eliminate the old initial state s_0 (cf. Fig. 9f). Note also that when s is a vanishing initial state but it has no incoming transitions, then nothing is changed (without a figure).*

Lemma 3 (Elimination does not destroy weak bisimilarity). *For every vanishing state s it holds that $P \approx P'^s$*

Proof. By definition $P \approx P'$. With the same arguments as in [1] (proof of Thm. 7) it follows that $P' \approx P'^s$. So by transitivity of \approx the claim follows. \square

The following lemma helps to understand the difference between naïvely vanishing and nn-vanishing states.

Lemma 4. *For every vanishing representation (s, τ, μ) that renders state s nn-vanishing there must be at least two distinct states $t_1, t_2 \in \text{Supp}(\mu)$ such that $t_1 \not\approx_{\Delta} s$, $t_2 \not\approx_{\Delta} s$ and $t_1 \not\approx_{\Delta} t_2$.*

Proof. Let (s, τ, μ) be the vanishing representation and assume that there is only *one* state $t \in \text{Supp}(\mu)$ such that $t \not\approx_{\Delta} s$. Without loss of generality, we may work on the quotient with respect to \approx_{Δ} . Further we may assume that μ is rescaled, i.e. $s \notin \text{Supp}(\mu)$ ⁴. Therefore we may assume that $\mu = \Delta_t$.

⁴For otherwise pretend that s is the initial state and eliminate it, i.e. replace it by a transient copy. By Lemma 3 it is clear that bisimilarity is not lost by this operation.

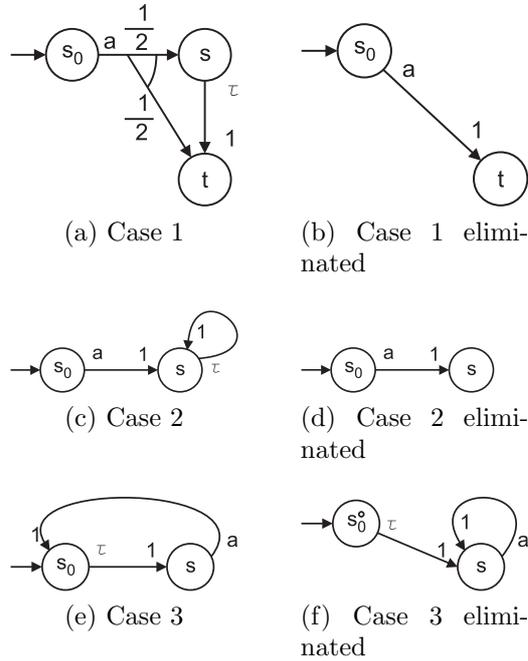


Figure 9: Different cases of eliminations

Thus we obtain the vanishing representation $s \xrightarrow{\tau} \Delta_t$ from which it would follow that $s \approx_{\Delta} t$, which is a contradiction. \square

Lemma 5. *It is not always possible to eliminate all naïvely vanishing states, but by elimination it is always possible to reach an automaton without naïvely vanishing states.*

Proof. A trivial example is given in Fig. 10. Here both s and s' are naïvely vanishing. But only either s or s' may be eliminated (it is easy to see that after the first elimination, the other state is no longer vanishing but has become tangible). \square

Lemma 6. *All nn-vanishing states can be eliminated (except for a nn-vanishing initial state which can only be made transient).*

Proof. We show that the situation described in the proof of Lemma 5, cannot occur for nn-vanishing states. I.e. we show that the elimination of a nn-vanishing state s may not cause another nn-vanishing state s' to lose its

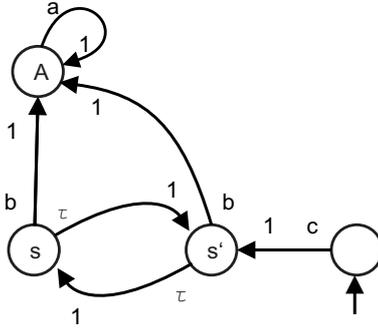


Figure 10: Two naïvely vanishing states

property of being nn-vanishing. Again we work on the quotient with respect to \approx_Δ . Assume that s is nn-vanishing with vanishing representation (s, τ, μ) . Let s' be another nn-vanishing state with vanishing representation (s', τ, γ) . In the case $s \notin \text{Supp}(\gamma)$ the elimination of s will not affect s' . In the case $s \in \text{Supp}(\gamma)$ the elimination of s will cause s to be replaced by μ in the vanishing representation of s' . We denote the resulting vanishing representation of s' as (s', τ, γ') ⁵. Since according to Lemma 4 we know that $\text{Supp}(\mu)$ still contains at least two states t_1 and t_2 such that $t_1 \not\approx_\Delta t_2$, we conclude that γ' contains at least one state (t_1 or t_2) that is not in the \approx_Δ relation with s' . Thus, after the elimination of state s , state s' is still nn-vanishing. \square

Directly from the proof of Lemma 6 we may deduce:

Corollary 1. *The property of a state s being nn-vanishing is not destroyed by other states being eliminated.*

Example 4. *This example shows that it is not enough to consider only strong emanating transitions when searching for vanishing representations such that non-bisimilar states are reached. Assume that $p \in (0, 1)$. We start with the automaton in Fig. 11a. Clearly both states E and F are trivially vanishing. Considering only strong transitions we see that E is nn-vanishing (vanishing representation $P_{(E, p\Delta_C \oplus (1-p)\Delta_D)}$), while F would be erroneously detected as naïvely vanishing with its vanishing representation $P_{(F, \Delta_E)}$. After elimination of E we obtain the automaton in Fig. 11b. As elimination leads to bisimilar results (Lemma 3) we see – after possibly rescaling the transition emanating*

⁵The resulting distribution γ' is still bisimilar to the original one because of Lemma 4.

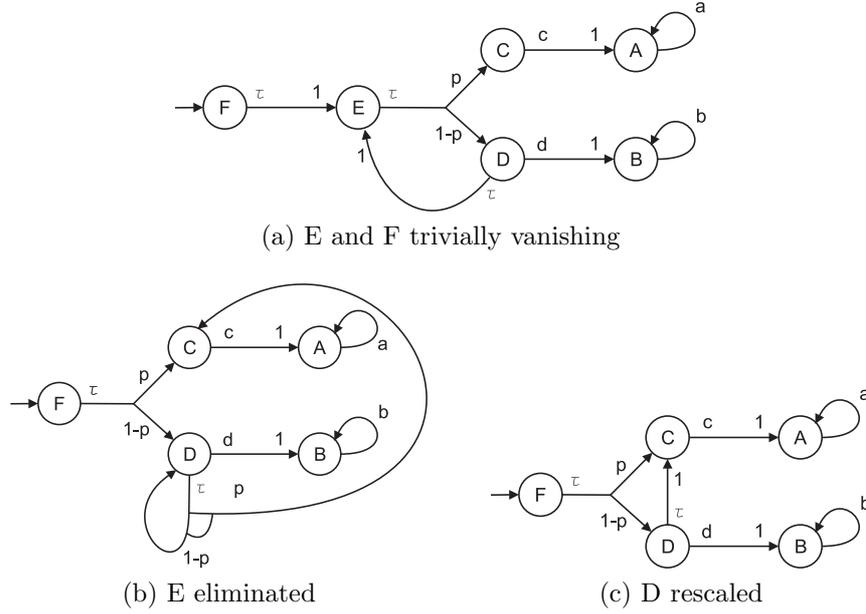


Figure 11: Examples of nn-vanishing states

from D leading to Fig. 11c – that also F must be nn-vanishing with vanishing representation $P_{(F,p\Delta_C\oplus(1-p)\Delta_D)}$.

We saw that all nn-vanishing states of an automaton may be eliminated, but not necessarily all naïvely vanishing states. That is why we introduce $m \leq n$ in the following definition.

Definition 16. Let $P = (S, Act, \rightarrow, \emptyset, s_0)$. Let $S^v = \{s_1^v, \dots, s_n^v\}$ be the set of vanishing states. Denote by \widehat{P} the complete elimination of P , i.e. $\widehat{P} := (\dots (P^{s_{i_1}^v})^{s_{i_2}^v} \dots)^{s_{i_m}^v}$, $m \leq n$, $i_j \in \{1, \dots, n\}$ for all $j \in \{1, \dots, m\}$, such that \widehat{P} contains no more vanishing states. Let \widehat{P}^* denote the elimination of all nn-vanishing states.

Using this definition, we can state the following important lemma:

Lemma 7 (Complete elimination and bisimilarity). For two PA P_1 and P_2 it holds: $P_1 \approx P_2 \Leftrightarrow \widehat{P}_1 \approx \widehat{P}_2 \Leftrightarrow \widehat{P}_1^* \approx \widehat{P}_2^*$

Proof. By Lemma 3 we know that elimination preserves weak bisimilarity.

The following diagrams show this by the right arrows.

$$\begin{array}{ccccc}
 P_1 & \xrightarrow{\approx} & \widehat{P}_1^* & \xrightarrow{\approx} & \widehat{P}_1 \\
 \downarrow & & \downarrow & & \downarrow \\
 P_2 & \xrightarrow{\approx} & \widehat{P}_2^* & \xrightarrow{\approx} & \widehat{P}_2
 \end{array}$$

As soon as one of the down arrows is a weak bisimulation, by transitivity of weak bisimulation we immediately get that the other two arrows are also weak bisimulations. \square

Remark 2. *In every example from [1], elimination leads to isomorphic automata (assuming that we replace vanishing initial states by their vanishing representation).*

It is clear by Lemma 2 that $\widehat{P}_1 \approx_{\text{naïve}} \widehat{P}_2 \Rightarrow \widehat{P}_1 \approx \widehat{P}_2$ and therefore (by Lemma 7) $P_1 \approx P_2$. Now we try to understand why it is also the case that $P_1 \approx P_2 \Rightarrow \widehat{P}_1 \approx_{\text{naïve}} \widehat{P}_2$.

5. Canonical vanishing representations and properties of vanishing states

In this section we prove that every nn-vanishing state s has a vanishing representation (s, τ, μ) where $\text{Supp}(\mu)$ only consists of nn-tangible states and the weak transition $s \xrightarrow{\tau} \mu$ is driven by a Dirac determinate scheduler (i.e. it is not a combined transition).

Lemma 8 (Vanishing representations don't need nn-vanishing states). *Every nn-vanishing state s has a vanishing representation (s, τ, μ') where $\text{Supp}(\mu')$ does not contain any nn-vanishing state (i.e. only nn-tangible states are in $\text{Supp}(\mu')$).*

Proof. Let s be nn-vanishing with vanishing representation (s, τ, μ) . We assume that $s \notin \text{Supp}(\mu)$, otherwise rescale. If $\text{Supp}(\mu)$ does not contain any nn-vanishing state we are done. Otherwise we perform “successive” eliminations (and possibly rescalings) of the nn-vanishing states in μ until all nn-vanishing states are eliminated. In detail: Replace all nn-vanishing states in $\text{Supp}(\mu)$ by their vanishing representations, leading to a new vanishing representation (s, τ, μ') of state s . If $s \in \text{Supp}(\mu')$ then rescale. In case

$Supp(\mu')$ contains only nn-tangible states we are finished. Otherwise set $\mu := \mu'$ and perform another round of elimination / rescaling. The finally resulting vanishing representation (s, τ, μ') has only nn-tangible states. The procedure terminates according to Lemma 6. \square

Theorem 1 (nn-vanishing states correspond to “real” distributions). *Let $P = (S, Act, T, \emptyset, s_0)$ be a PA. A state $s \in S$ is nn-vanishing iff there exists a distribution μ such that $\Delta_s \approx \mu$ but $\exists t \in Supp(\mu)$ such that $s \not\approx_{\Delta} t$.*

Proof. \Rightarrow Assume that s is nn-vanishing, then we may use the vanishing representation. Let (s, τ, μ') be the vanishing representation of s . Then we have both $\Delta_s \approx_{\Delta} \mu'$ (which follows directly from the definition of weak bisimilarity) and $\exists t \in Supp(\mu')$ such that $s \not\approx_{\Delta} t$. Therefore we can use $\mu = \mu'$ to satisfy the right hand side of the Theorem.

\Leftarrow Without loss of generality we may work on quotients with respect to \approx_{Δ} . Assume that on the quotient it holds that $\mu = \bigoplus_{i \in \{0, \dots, n\}} d_i \Delta_{t_i}$ for some $n \in \mathbb{N}$ ($t_i \neq t_j$ whenever $i \neq j$) and without loss of generality we assume that $t^0 \not\approx_{\Delta} s$. Firstly, we show that there must be a weak combined transition $s \xrightarrow{\tau}_C \gamma$ with $\exists x \in Supp(\gamma) : x \not\approx_{\Delta} s$ and $\Delta_s \approx \gamma$: From $\Delta_s \approx \mu$ we get by [2, Lemma 11] a transition $s \xrightarrow{\tau}_C \gamma = \bigoplus_{i \in \{0, \dots, n\}} \gamma_i$ such that for all $i \in \{0, \dots, n\}$ we get $d_i \Delta_{t_i} \approx \gamma_i$. First note that by [2, Lemma 9] we get $\Delta_s \approx \gamma$, as from $d_i \Delta_{t_i} \approx \gamma_i$ it follows that $\mu = \bigoplus_{i \in \{0, \dots, n\}} d_i \Delta_{t_i} \approx \bigoplus_{i \in \{0, \dots, n\}} \gamma_i = \gamma$ and we have $\Delta_s \approx \mu$ as a precondition. So also $\Delta_s \approx \gamma$. For (s, τ, γ) being a vanishing representation and s being nn-vanishing, it remains to show that there exists some $x \in Supp(\gamma)$ such that $s \not\approx_{\Delta} x$. Assume that this is not the case, i.e. $\forall x \in Supp(\gamma) : x \approx_{\Delta} s$. Now assume that $\gamma_i = \bigoplus_{j \in J_i} b_{ij} \Delta_{x_{ij}}$ ($x_{ij} \neq x_{ik}$ whenever $j \neq k$). But then we have

$$d_0 \Delta_s \not\approx d_0 \Delta_{t^0} \approx \underbrace{\bigoplus_{j \in J_0} b_{0j} \Delta_{x_{0j}}}_{\gamma_0} \approx d_0 \Delta_s$$

which is a contradiction (the rightmost \approx follows from $\sum_{j \in J_0} b_{0j} = d_0$ and that for all x_{0j} we have $x_{0j} \approx_{\Delta} s$, therefore $\gamma_0 \approx d_0 \Delta_s$). So we conclude that there must be a $x \in Supp(\gamma)$ such that $x \not\approx_{\Delta} s$. Therefore we have a transition $s \xrightarrow{\tau}_C \gamma$ with $\Delta_s \approx \gamma$ and $\exists x \in Supp(\gamma) : s \not\approx_{\Delta} x$.

Secondly, we have to show that (s, γ) is a vanishing representation, i.e. that $s \approx_{\Delta} t$ when comparing $P[t/s]$ and $P_{(s, \nu)}$. So we have to show that when only using the transition (s, τ, γ) all other transitions emanating from t can be mimicked and thus omitted. Now let $t \xrightarrow{\alpha} \rho$ be an arbitrary transition

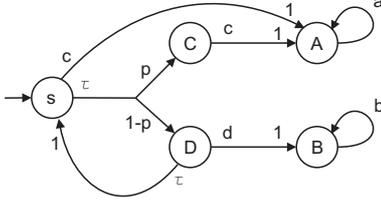


Figure 12: Example for the substitution of transitions

from $P[t/s]$. By $\Delta_t \approx \gamma$ we get directly from Definition 8 a hypertransition $\gamma \xrightarrow{a}_C \rho'$ such that $\rho \approx \rho'$. It remains to show that $\gamma \xrightarrow{a}_C \rho'$ is possible in $P_{(s,\nu)}$ (i.e. without using the transition $t \xrightarrow{a} \rho$). If there was a transition $\gamma \xrightarrow{\tau}_C \Delta_t$, this would be a contradiction to the existence of $\exists x \in \text{Supp}(\gamma) : x \not\approx_\Delta t$ because all states in $\text{Supp}(\gamma)$ would be bisimilar to t due to the loop $t \xrightarrow{\tau}_C \gamma \xrightarrow{\tau}_C \Delta_t$. Therefore we know that such a transition does not exist⁶. That means we may successively substitute every occurrence of $t \xrightarrow{a} \rho$ by the transition $t \xrightarrow{\tau}_C \gamma \xrightarrow{a}_C \rho'$. Note that such a substitution clearly does not change bisimilarity, as the substituted distribution is bisimilar (by [2, Lemma 9] distributions containing those different sub-distributions will still be bisimilar). The successive substitutions make the probability of choosing $t \xrightarrow{a} \rho$ tend to zero, i.e. this transition is not needed. As our example transition was general (and not equal to $t \Rightarrow \gamma$), we see that state t must be nn-vanishing. \square

Example 5. A basic example of this kind of substitution used in the proof is given in Fig. 12. Let $(s, p\Delta_C \oplus (1-p)\Delta_D)$ be the candidate for the vanishing representation. We have to show that $s \xrightarrow{c} \Delta_A$ is superfluous. According to the construction of the previous proof we get a hypertransition $p\Delta_C \oplus (1-p)\Delta_D \xrightarrow{c}_C \Delta_A$. We assume that this hypertransition is driven by the transitions $\Delta_C \xrightarrow{c} \Delta_A$ and $\Delta_D \xrightarrow{\tau} \Delta_s \xrightarrow{c} \Delta_A$. This leads us to the sequence of transitions $s \xrightarrow{\tau} (p\Delta_C \oplus (1-p)\Delta_D) \xrightarrow{\tau} (p\Delta_C \oplus (1-p)\Delta_s) \xrightarrow{c} \Delta_A$, where $s \xrightarrow{c} \Delta_A$ is taken with probability $(1-p) < 1$, as assumed. So in the next substitution step, we may use $s \xrightarrow{\tau} (p\Delta_C \oplus (1-p)\Delta_D) \xrightarrow{\tau} (p\Delta_C \oplus (1-p)\Delta_s) \xrightarrow{\tau} (p(2-p)\Delta_C \oplus (1-p)^2\Delta_s) \xrightarrow{c} \Delta_A$, that is we utilise $s \xrightarrow{c} \Delta_A$ only with probability $(1-p)^2$. Taking this to infinity means that the transition $s \xrightarrow{c} \Delta_A$ is indeed redundant, so $(s, p\Delta_C \oplus (1-p)\Delta_D)$ is really a vanishing

⁶So the probability for returning to s is strictly smaller than one.

representation.

The following corollary states that all states within an equivalence class with respect to \approx_Δ are either nn-vanishing or not.

Corollary 2. *Let $P = (S, Act, \rightarrow, \emptyset, s_0)$ be a PA. If $s \in S$ is nn-vanishing and $s \approx_\Delta t$ then t is also nn-vanishing.*

Proof. It follows from Thm. 1 that there exists a distribution μ such that $\Delta_s \approx \mu$ but $\exists x \in \text{Supp}(\mu)$ with $s \not\approx_\Delta x$. The assumption $s \approx_\Delta t$ yields, by transitivity of \approx (cf. Thm. 1 in [1, 2]), that $\Delta_t \approx \mu$. Still $t \not\approx_\Delta x$, for otherwise $s \approx_\Delta t \approx_\Delta x$ which would be a contradiction to the nn-vanishing property of s . So, again by Thm. 1, we find that t must be nn-vanishing, too. \square

The following theorem states that the nn-vanishing states are the “obstacle” between “weak bisimulation” and “naïve weak bisimulation”. This theorem renders Thm. 2 in [1, 2] more precisely.

Theorem 2. *It holds that $P_1 \approx P_2 \Leftrightarrow \widehat{P}_1^* \approx_{\text{naïve}} \widehat{P}_2^*$.*

Proof. \Leftarrow is immediate by Lemma 2 and Lemma 7.

\Rightarrow From Lemma 7 we already know $P_1 \approx P_2 \Leftrightarrow \widehat{P}_1^* \approx \widehat{P}_2^*$. So it remains to show that $\widehat{P}_1^* \approx \widehat{P}_2^*$ is already a naïve weak bisimulation. By the definition of weak bisimilarity (Definition 8) it follows that whenever $s \approx_\Delta t$ then for every $s \xrightarrow{a} \mu$ we find $t \xrightarrow{a}_C \gamma$ with $\mu \approx \gamma$ (and vice versa). By Lemma 6 and Definition 15 it is clear that neither \widehat{P}_1^* nor \widehat{P}_2^* contain any nn-vanishing states (the only exception are, if present, nn-vanishing initial states, which then must be transient). We have to show that $\mu \approx \gamma$ already coincide on classes, i.e. the weak bisimulation is already naïve. Assume that we split μ according to its support: $\mu = \oplus_{i \in I} c_i \Delta_{s_i}$, then with Lemma 11 of [2] we get a hypertransition $\gamma \Rightarrow_C \gamma' = \oplus_{i \in I} c_i \gamma'_i$ with $\Delta_{s_i} \approx \gamma'_i$. By assumption s_i cannot be nn-vanishing (only the initial state could be, but $s_i \neq s_0$ as it is a target state of some transition and s_0 is transient). Now it is clear that for all states $x \in \text{Supp}(\gamma'_i)$ it must hold that $x \approx_\Delta s_i$ – for otherwise by Thm. 1 s_i would be nn-vanishing, which is a contradiction. Summing up, we have $t \xrightarrow{a}_C \gamma'$ and $\mu \equiv_{\approx_\Delta} \gamma'$. We can use the same argumentation for $t \xrightarrow{a} \gamma$ to find $s \xrightarrow{a}_C \mu'$ with $\gamma \equiv_{\approx_\Delta} \mu'$ and we conclude that \approx_Δ is already a naïve weak bisimulation relation. \square

The following lemma will be used in the proof of Lemma 10, which shows that it suffices to consider Dirac determinate schedulers when trying to identify nn-vanishing states. Note that, since Lemma 9 assumes that the PA at hand is a quotient with respect to \approx_Δ , it cannot contain any naïvely vanishing states.

Lemma 9. *Let $P = (S, Act, \rightarrow, \emptyset, s_0)$ a PA that is a quotient with respect to \approx_Δ . Let s be a nn-vanishing state. For a vanishing representation (s, τ, μ) where $Supp(\mu)$ contains only nn-tangible states it holds that μ is uniquely defined, i.e. there is no other vanishing representation (s, τ, γ) with $\mu \neq \gamma$ where $Supp(\gamma)$ contains only nn-tangible states. (Such a vanishing representation (s, τ, μ) is called canonical.)*

Proof. Let $P' = (S, Act, \rightarrow, \emptyset, s_0)$ be a quotient with respect to \approx_Δ , $s \in S$ nn-vanishing and assume that there exist different vanishing representations (s, τ, μ) and (s, τ, γ) , where $Supp(\mu)$ and $Supp(\gamma)$ contain only nn-tangible states. We now pretend that s is the starting state, i.e. consider the automaton $P = (S, Act, \rightarrow, \emptyset, s)$. According to Thm. 2 it must hold that $\widehat{P}_{(s,\mu)}^* \approx_{\text{naïve}} \widehat{P}_{(s,\gamma)}^*$. It is easy to see that (s, τ, μ) and (s, τ, γ) are not changed by the elimination procedure⁷. As μ and γ do not coincide and there are no other emanating transitions of state s , Thm. 1 of [16] tells us that the corresponding Normal Form cannot have an emanating τ transition from s . A contradiction to the nn-vanishing property of s . \square

The use of combined transitions $\mathfrak{C}(s)$ for vanishing representations is not necessary, as the following lemma shows:

Lemma 10 (Considering Dirac determinate schedulers is sufficient to find nn-vanishing states). *Every nn-vanishing state s has a vanishing representation (s, τ, μ) where $s \xrightarrow{\tau} \mu$ is driven by a Dirac determinate scheduler.*

Proof. Let $P^0 = (S, Act, \rightarrow, \emptyset, s_0)$ be a PA. In the following we assume that we work on quotients with respect to \approx_Δ . We will show that any vanishing representation (s, τ, μ) of nn-vanishing state s can be transformed to a

⁷The transitions corresponding to the vanishing representations must already be rescaled as s is nn-vanishing while all states in the support of γ and μ are nn-tangible, so it holds that $s \notin Supp(\mu)$ and $s \notin Supp(\gamma)$. No state in the support of these distributions will have been eliminated.

vanishing representation (s, τ, μ') where $s \xrightarrow{\tau}_C \mu'$ starts with a strong non-combined transition. Let $\{(s, \tau, \nu) \mid s \xrightarrow{\tau} \nu\} = \{(s, \tau, \nu_1), \dots, (s, \tau, \nu_n)\}$. Assume now that we have a vanishing representation (s, τ, μ) where the first strong step of $s \xrightarrow{\tau}_C \nu \xrightarrow{\tau}_C \mu$ leads to a non-trivial combination $\nu = \sum_{i \in I} c_i \nu_i$ where $I \subseteq \{1, \dots, n\}$ and $0 < c_i < 1$ for all $i \in I$. Now we pretend for the moment that s is the starting state, i.e. we consider the automaton $P = (S, Act, \rightarrow, \emptyset, s)$. Fix for every nn-vanishing state t a vanishing representation (t, τ, μ_t) with support in nn-tangible states (which is unique according to Lemma 9). Now we perform two different kinds of eliminations:

1. \widehat{P}^* , i.e. the usual elimination from Def. 15
2. $\widehat{P}_{(s)}^*$, which means we make s transient by elimination but keep all τ -transitions emanating from s , i.e. we do not move to its vanishing representation. (All nn-vanishing states apart from s are eliminated as usual.)

By construction, all transitions in \widehat{P}^* and $\widehat{P}_{(s)}^*$ lead to distributions whose support consists only of nn-tangible states (this will be denoted by the superscript “nn-tang”). As the common starting point for reaching eliminations \widehat{P}^* and $\widehat{P}_{(s)}^*$ is the automaton P , we get by Thm 2 that $\widehat{P}^* \approx_{\text{naive}} \widehat{P}_{(s)}^*$, so the resulting transition in $\widehat{P}_{(s)}^*$ emanating from s leading to nn-tangible states must still be a vanishing representation. In \widehat{P}^* the transitions $s \xrightarrow{\tau} \nu_i$ are transformed to strong non-combined transitions $s \xrightarrow{\tau} \nu_i^{\text{nn-tang}}$. In $\widehat{P}_{(s)}^*$ the transition $s \xrightarrow{\tau}_C \nu$ is transformed to a strong (possibly combined) transition $s \xrightarrow{\tau}_C \nu^{\text{nn-tang}}$. Now we need the following reduction argument: If there are i, j such that $\nu_i^{\text{nn-tang}} = \nu_j^{\text{nn-tang}}$, then we can substitute ν_i by ν_j in $\nu = \sum_{i \in I} c_i \nu_i$ without losing the property of (s, τ, ν) being the first step of a vanishing representation (by Thm. 1). (As an example we have in Fig. 13 the distributions $\nu_1 = \frac{2}{3}\Delta_x \oplus \frac{1}{3}\Delta_C$ and $\nu_2 = \frac{1}{3}\Delta_A \oplus \frac{2}{3}\Delta_y$, which leads to $\nu_1^{\text{nn-tang}} = \nu_2^{\text{nn-tang}} = \frac{1}{3}\Delta_A \oplus \frac{1}{3}\Delta_B \oplus \frac{1}{3}\Delta_C$ — therefore we may substitute ν_1 by ν_2 in $\nu = \sum_{i \in I} c_i \nu_i$). If after this reduction argument, the vanishing representation has a strong transition as first step, we are done. Otherwise we may assume that $\nu = \sum_{i \in I^*} c_i^* \nu_i$ with $I^* \subseteq I$, where $\nu_i^{\text{nn-tang}} \neq \nu_j^{\text{nn-tang}}$ for $i \neq j$. As for every nn-vanishing state t a *unique* vanishing representation (t, τ, μ_t) was chosen, it must hold that $\nu^{\text{nn-tang}} = \sum_{i \in I^*} c_i^* \nu_i^{\text{nn-tang}}$ (with the same coefficients c_i^* as above). As s is transient in \widehat{P}^* and $\widehat{P}_{(s)}^*$ by definition, all transitions emanating from s in these eliminated automata must be

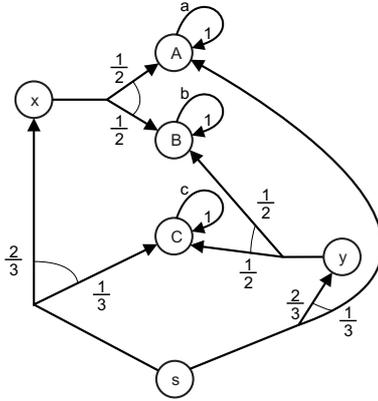


Figure 13: Coinciding $\nu_i^{\text{nn-tang}}$ and $\nu_j^{\text{nn-tang}}$

rescaled. Now we may apply Thm. 1 of [16] to see that the corresponding normal form (whose set of transitions must be from the intersection of the transition sets of \widehat{P}^* and $\widehat{P}_{(s)}^*$) cannot have an emanating τ transition from s ⁸. This is a contradiction to the nn-vanishing property of s . This means that the first step of a transition leading to a vanishing representation must be non-combined, i.e. some $c_i^* = 1$ in the sum above. As the same argument holds for the other nn-vanishing states, we see that with Lemma 8 a vanishing representation (s, τ, ρ) for s with some $t \in \text{Supp}(\rho) : s \not\approx_{\Delta} t$ can be reached from s by Dirac Determinate schedulers. \square

It can be shown that as long as there are no probability-one τ -loops, the elimination procedure is unique. At the presence of such loops the elimination procedure is unique up to isomorphism [17].

Corollary 3. *It holds that $P_1 \approx P_2 \Leftrightarrow \widehat{P}_1 \approx_{\text{naïve}} \widehat{P}_2$.*

Proof. \Leftarrow is immediate by Lemma 2 and Lemma 7.

\Rightarrow Using Thm. 2 we know that after elimination of nn-vanishing states \approx_{Δ} is already a naïve weak bisimulation, so it only remains to show that $\widehat{P}_1^* \approx_{\text{naïve}} \widehat{P}_2^* \Leftrightarrow \widehat{P}_1 \approx_{\text{naïve}} \widehat{P}_2$. This is true since eliminating any naïvely vanishing states from \widehat{P}_1^* or \widehat{P}_2^* does not change the behaviour with respect to \approx_{Δ} , so it is still a naïve weak bisimulation. \square

⁸The only possible vanishing representation on the quotient would be (s, τ, Δ_s) , which clearly doesn't satisfy the nn-vanishing property.

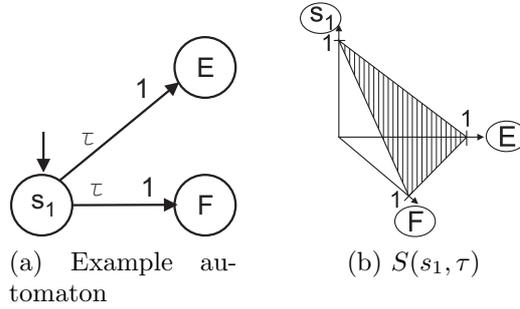


Figure 14: Example for reachable distributions

6. A partition refinement algorithm

With Lemma 10 we can find nn-tangible states and Th. 2 reduces the problem to naïve weak bisimulation. For the description of the partition refinement algorithm below we need the convex sets $S(x, a) \subseteq \mathbb{R}^n$ introduced by [4]. For details on how to calculate those sets we refer to [4].

Example 6. For the MA given in Fig. 14a the set $S(s_1, \tau)$ is given by the shaded triangle in Fig. 14b. This triangle encodes all distributions that are reachable via a (weak) combined τ transition starting from s_1 .

Remark 3. It was shown in [4] that each convex set is the convex hull (CHull) of distributions (i.e. points in \mathbb{R}^n) generated by Dirac determinate schedulers. It is shown there that the extremal points (i.e. generators) of the convex hull can be found by a linear program and that the complexity of calculating the sets $S(s, a)$ is exponential for the weak case. This is one of the reasons why our algorithm also has exponential complexity (see Sec. 6.2).

Given a set $S(s, a) \subseteq \mathbb{R}^n$, it may be restricted, which is an essential ingredient of our decision algorithm.

Definition 17 (Restriction of convex sets). We define

$$S(s, a)|_{x_i=0} := \{(x_1, \dots, x_n) \in S(s, a) | x_i = 0\}$$

Multiple restrictions can also be realised. Especially, we define a restriction to a set of “nn-tangible” states $S(s, a)|_{S_{nn-tangible}}$ by requiring that $x_i = 0$ for all nn-vanishing states x_i .

Note that the restriction of convex sets is still convex by definition.

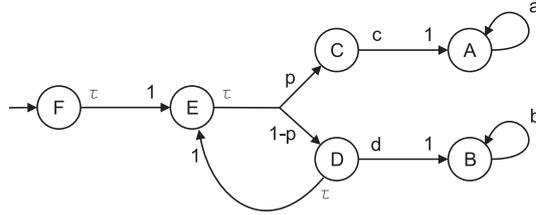
Lemma 11. *Restriction of $S(s, a)$ to the set of nn-tangible states corresponds to virtual elimination of nn-vanishing states.*

Proof. In the elimination procedure we redirect transitions according to the vanishing representation. The vanishing state is no longer reached and can be removed from the state space (or alternatively: its probability can be restricted to zero). All other transitions that can be realised from a nn-vanishing state s using other transitions than the one belonging to the vanishing representation can be weakly emulated by using the vanishing representation as a first step. The only exception to keep in mind is when considering the sets $S(s, a)$ where s itself is nn-vanishing. There normally Δ_s must be in $S(s, \tau)$ (as of course always $s \xrightarrow{\tau} \Delta_s$). As s is nn-vanishing, there must also be a vanishing representation consisting of nn-tangible states which uniquely identifies s (cf. Lemma 8 and Lemma 9). Note that it holds that $S(s, \tau) = CHull(\Delta_s, S(s, \tau)|_{prob(s)=0})$ and therefore $S(s, \tau)$ is uniquely determined by the set $S(s, \tau)|_{prob(s)=0}$. Similar considerations apply also for the case $a \neq \tau$: nn-vanishing states are identified by their vanishing representations consisting of nn-tangible states. \square

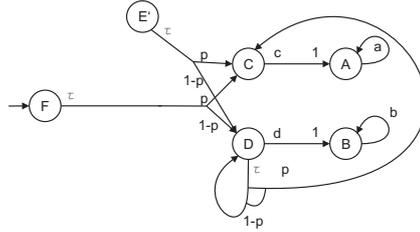
The last missing part for setting up the algorithm is to show how it can be fitted to a partition refinement algorithm. This can be seen in Fig. 15. The underlying idea is to treat every nn-vanishing state in the elimination as if it were an initial state (i.e. eliminate it, but leave a transient copy in the transition system). Note, however, that the algorithm does not perform a real elimination, but only a “virtual” elimination (by considering the restricted $S(s, a)$ -sets). But now it is clear by Thm. 2 that the bisimulation problems we have to solve are naïve weak bisimulations that can be treated by the Segala/Cattani algorithm. For example, in Fig. 15 one trivially sees that E and F are naïvely weakly bisimilar. But after a splitting occurred, we have to verify if all nn-vanishing states are still nn-vanishing with respect to the new partition. This justifies the iterative scheme sketched in Fig. 2.

So the proposed algorithm for deciding whether two MA are weakly bisimilar looks as follows:

1. Start with the initial partition $\mathcal{W} = \{S_1 \uplus S_2\}$.
2. For all states s and actions a calculate the convex sets $S(s, a)$ (cf. [4]) and for every (Dirac determinate) weak transition (s, τ, ν) calculate



(a) E and F nn-vanishing



(b) E eliminated (treating it as if it were a starting state)

Figure 15: Rendering nn-vanishing states transient

$S_\nu(s, a)$ ($S_\nu(s, a)$ denotes the convex set calculated for the PA $(P_1 \uplus P_2)_{(s, \nu)}$, that is the direct sum of automata P_1 and P_2 where we move to the vanishing representation (s, ν)).

3. Set $S_{nn-tangible} = \emptyset$.
4. For all states s that are not in $S_{nn-tangible}$
 - Check whether s can leave its equivalence class in \mathcal{W} by a (Dirac determinate) weak transition (s, τ, ν) such that (modulo \mathcal{W}) $S(s, a)|_{S_{nn-tangible}} = S_\nu(s, a)|_{S_{nn-tangible}}$ for all $a \in Act$. This is a *vanishing representation of nn-vanishing state s with respect to \mathcal{W}* .
 - If no vanishing representation with respect to the current partition \mathcal{W} can be found, then s must be nn-tangible with respect to \mathcal{W} . Add state s to $S_{nn-tangible}$.
 - Cross-check all other states if they also become nn-tangible as an effect of s being nn-tangible.
5. Find a new splitter (in the sense of [4]) with respect to the current partition and the current set of nn-tangible states, i.e. a tuple (C, a, \mathcal{W})

which indicates that class C needs to be refined w.r.t. a weak a transition.

6. Refine the partition according to the splitter and start next round at step 3.

The algorithm is depicted in Alg. 1. By $DiracDet(s, \tau)$ we mean all distributions ν induced by $s \Rightarrow \nu$ by means of a Dirac determinate scheduler. It remains to define the ComputeInfo algorithm, FindWeakSplit algorithm and the Refine algorithm. As the Refine algorithm is standard, we omit it from this paper. The routine ComputeInfo just calculates the convex sets $S(s, a)$ according to Remark 3. The routine FindWeakSplit given in Alg. 2 looks very much like the one given in [4] but it only “sees” nn-tangible states that are provided as an additional parameter to the routine.

Lemma 12. *The algorithm in Alg. 1 calculates the coarsest partition with respect to \approx_{Δ} .*

Proof. The claim follows by the correctness of the naïve weak bisimulation algorithm given in [4]. The only special case to consider is when a nn-vanishing and a nn-tangible state are detected in the *same* class. This case is not problematic due to the following reasoning: the nn-vanishing state can leave its class towards an equivalent distribution (i.e. a distribution weakly bisimilar to the Dirac distribution on the nn-vanishing state) which consists of at least two other classes (cf. Lemma 4). In contrast, the nn-tangible state either cannot leave its class at all, or it can leave its class but thereby losing bisimilarity. As the classes are refined and never merged, by the above reasoning nn-vanishing and nn-tangible states cannot be bisimilar and may always be split. Therefore the special case that an nn-tangible state s has Δ_s in $S(s, \tau)|_{S_{nn-tangible}}$, whereas – due to restriction – for an nn-vanishing state t , Δ_t is not in $S(s, \tau)|_{S_{nn-tangible}}$ is not problematic as it allows for separating nn-vanishing from nn-tangible states. Similar considerations apply to the case $a = \tau$, which we do not discuss explicitly.

Once the algorithm terminates, there is no class containing both nn-vanishing and nn-tangible states. When restricting to the nn-tangible fraction of successor states (which corresponds by Lemma 11 to elimination of nn-vanishing states), the states within one class are naïvely weakly bisimilar (this follows from the algorithm given in [4]). Furthermore, our algorithm calculates for each nn-vanishing state the canonical vanishing representation

consisting of nn-tangible states only (cf. Lemma 8). Summing up, this means (by Thm. 2) weak bisimilarity when considering both nn-tangible and nn-vanishing states (i.e. without restrictions). \square

Algorithm 1 DecideWeakBisim

Require: Two MA as PA $P_1 = (S_1, Act_1, \rightarrow_1, \emptyset, s_0)$, $P_2 = (S_2, Act_2, \rightarrow_2, \emptyset, t_0)$

- 1: $S = S_1 \uplus S_2$, $\mathcal{W} = \{S\}$, $Act = Act_1 \cup Act_2$
- 2: **for** $s \in S$, $a \in Act$, $\nu \in DiracDet(s, \tau)$ **do**
- 3: $S(s, a) = ComputeInfo(s, a)$ on $(P_1 \uplus P_2)$
- 4: $S_\nu(s, a) = ComputeInfo(s, a)$ on $(P_1 \uplus P_2)_{(s, \nu)}$
- 5: **end for**
- 6: **while** \mathcal{W} changes **do**
- 7: $S_{nn-tangible} := \emptyset$
- 8: **while** $S_{nn-tangible}$ changes **do**
- 9: **for** $s \in S \setminus S_{nn-tangible}$ **do**
- 10: **for** $\nu \in DiracDet(s, \tau)$ where $\exists x \in Supp(\nu) : [x]_{\mathcal{W}} \neq [s]_{\mathcal{W}}$ **do**
- 11: **if** $\forall a \in Act : (S(s, a)|_{S_{nn-tangible}})/\mathcal{W} = (S_\nu(s, a)|_{S_{nn-tangible}})/\mathcal{W}$ **then**
- 12: vanishing representation found **break**
- 13: **end if**
- 14: **end for**
- 15: **if** no vanishing representation found **then**
- 16: $S_{nn-tangible} := S_{nn-tangible} \cup \{s\}$
- 17: **end if**
- 18: **end for**
- 19: **end while**
- 20: $(C, a, \mathcal{W}) = FindWeakSplit(S_{nn-tangible}, \mathcal{W}, S, Act, S(\cdot, \cdot))$
- 21: $\mathcal{W} = Refine(C, a, \mathcal{W})$
- 22: **end while**
- 23: $P_1 \approx P_2$ iff $[s_0]_{\mathcal{W}} = [t_0]_{\mathcal{W}}$

Remark 4. *The algorithm detects nn-vanishing states and finds their vanishing representations (regarding the most recent partition \mathcal{W}). Therefore, at the end of the algorithm, we will be able to really (i.e. not virtually) eliminate all nn-vanishing states and reach a form where only nn-tangible states*

Algorithm 2 FindWeakSplit (Find weak bisimulation splitter)

Require: nn-tangible states $S_{nn-tangible}$, partition \mathcal{W} , states S , actions Act ,
Info $S(\cdot, \cdot)$
1: **for** $C_i \in \mathcal{W}$, $s, t \in C_i$, $a \in Act$ **do**
2: **if** $(S(s, a)|_{S_{nn-tangible}})/\mathcal{W} \neq (S(t, a)|_{S_{nn-tangible}})/\mathcal{W}$ **then**
3: **return** (C_i, a, \mathcal{W})
4: **end if**
5: **end for**

are present (with the only exception of a nn-vanishing initial state, which can only be rendered transient) and where for every equivalence class only one state is used. This can be regarded as a kind of normal form.

6.1. Example

Suppose we are given the MA in Fig. 16a (there already transformed to a PA P) where $p, q \in (0, 1)$. This automaton can be seen as a condensed form of two separate automata (starting with s_1 and t_1 , thus these are indicated as initial states), where states A and B have been identified (to keep things short – if there were two copies of A and B : one for the left and one for the right automaton, they would be grouped in the course of the algorithm). We want to show that $s_1 \approx_{\Delta} t_1$.

We assume that $p = q = \frac{1}{2}$, as the pictures are easier to draw in that case, but we would like to stress that the same arguments work for all other choices (as long as p and q are not equal to 0 or 1).

Remark 5. *In the following graphical representations of the convex sets, we add dots for every result of a Dirac determinate scheduler (according to [4]) whenever we draw the convex sets of reachable distributions as subsets of \mathbb{R}^n . Dots that are not extremal points may safely be omitted, as they can be reached as convex combinations of the extremal points.*

First round: Start with the partition $\mathcal{W}_0 = \{\{s_1, s_2, t_1, A, B\}\}$ (cf. Fig. 16b). Observe that in the loop from line 9 to line 18 we can never find a vanishing representation of a nn-vanishing state, as no state may leave its equivalence class with some probability greater than zero. Therefore we get $S_{nn-tangible} = \{s_1, s_2, t_1, A, B\}$.

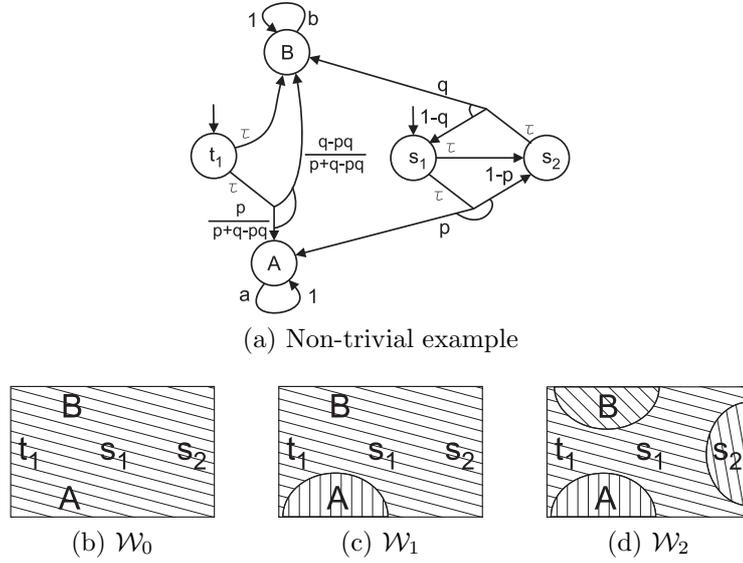


Figure 16: Example and partitions during algorithm run

Now we have to find a splitter with respect to $(S_{nn-tangible}, \mathcal{W}_0)$. Suppose that we check the sets $S(\cdot, b)/\mathcal{W}_0$. Here we see that:

$$S(x, b)/\mathcal{W}_0 = \begin{cases} \begin{array}{c} 0 \\ \text{---} \\ 1 \end{array} \bullet \begin{array}{c} s_1 \ B \\ t_1 \ A \\ s_2 \end{array} & \text{for } x \in \{B, s_1, s_2, t_1\} \\ \emptyset & \text{otherwise} \end{cases}$$

So we have found a splitter. Refining according to $(\{s_1, s_2, t_1, A, B\}, b, \mathcal{W}_0)$ leads to $\mathcal{W}_1 = \{\{s_1, s_2, t_1, B\}, \{A\}\}$ (cf. Fig. 16c).

Second round: We first have to detect the nn-vanishing states with respect to the current partition. We calculate $S(x, \tau)$ for every state, verify if it is possible to reach another equivalence class and see whether one single τ transition suffices. The values of $S(x, \tau)$ are given in Tab. 1. Firstly notice that both A and B cannot be nn-vanishing, as they have no possibility of leaving their equivalence classes. Notice also, that even if s_2 is trivially vanishing, as it has only one single emanating τ transition, we cannot detect it as nn-vanishing (the only vanishing representation that leaves the class would be $P_{(s_2, \frac{1}{3}\Delta_A \oplus \frac{2}{3}\Delta_B)}$, but $S(s_2, b)/\mathcal{W}_1 \neq S_{\frac{1}{3}\Delta_A \oplus \frac{2}{3}\Delta_B}(s_2, b)/\mathcal{W}_1$). Regarding s_1 we see that we cannot omit transition $s_1 \xrightarrow{\tau} \frac{1}{2}\Delta_A \oplus \frac{1}{2}\Delta_{s_2}$, as $S_{(\Delta_{s_2})}(s_1, \tau)/\mathcal{W}_1 = S(B, \tau)/\mathcal{W}_1 \neq S(s_1, \tau)$. But notice also that $s_1 \xrightarrow{\tau} \Delta_{s_2}$ cannot be omitted,

x	$S(x, \tau)/\mathcal{W}_1$	x	$S(x, \tau)/\mathcal{W}_1$	x	$S(x, \tau)/\mathcal{W}_1$
s_1		t_1		s_2	
B		A			

Table 1: $S(x, \tau)$

as $S(s_1, b)/\mathcal{W}_1 = S(B, \tau)/\mathcal{W}_1$, but $S_{(\frac{1}{2}\Delta_A \oplus \frac{1}{2}\Delta_{s_2})}(s_1, b)/\mathcal{W}_1 = \emptyset$. So we see that s_1 cannot be nn-vanishing. With the same argument we see that also t_1 cannot be nn-vanishing. Therefore we get $S_{nn-tangible} = \{s_1, s_2, t_1, A, B\}$.

Now we look for splitters with respect to $(\mathcal{W}_1, S_{nn-tangible})$. Looking at $C = \{s_1, s_2, t_1, B\}$ we see in routine FindWeakSplit that we can use a splitter (C, τ, \mathcal{W}_1) and get the partition $\mathcal{W}_2 = \{\{s_1, t_1\}, \{s_2\}, \{A\}, \{B\}\}$ (cf. Fig. 16d, note that $S(s_1, \tau)/\mathcal{W}_1 = S(t_1, \tau)/\mathcal{W}_1$, as $(\frac{1}{2}, \frac{1}{2})$ is *not* a generator of the convex set).

Third round: We first have to detect nn-vanishing states. It is clear that s_2 must be nn-vanishing as it can leave its class and only has a single outgoing τ transition. With the same arguments as above we see that both s_1 and t_1 must be nn-tangible. So we get $S_{nn-tangible} = \{s_1, t_1, A, B\}$.

Now we again can look for splitters, but have to consider the restriction to $S_{nn-tangible}$. Notice that with coordinates $[s_1] = [t_1]$, $[s_2]$, $[A]$, $[B]$ we have

$$S(s_1, \tau)/\mathcal{W}_2 = CHull\left(\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ \frac{1}{2} \\ \frac{1}{2} \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \frac{2}{3} \\ \frac{1}{3} \end{pmatrix}\right).$$

We want to calculate the restriction $S(s_1, \tau)|_{S_{nn-tangible}}/\mathcal{W}_2$. Let us for the moment ignore the vertex $[s_1] \in S(s_1, \tau)/\mathcal{W}_2$. Then we get the picture in Fig. 17a for $S(s_1, \tau)|_{s_1=0}/\mathcal{W}_2$. We see that the restriction of this set to $S_{nn-tangible}$ gives only the line from $(0, 0, 1)$ to $(0, \frac{2}{3}, \frac{1}{3})$ (cf. Fig. 17b), therefore

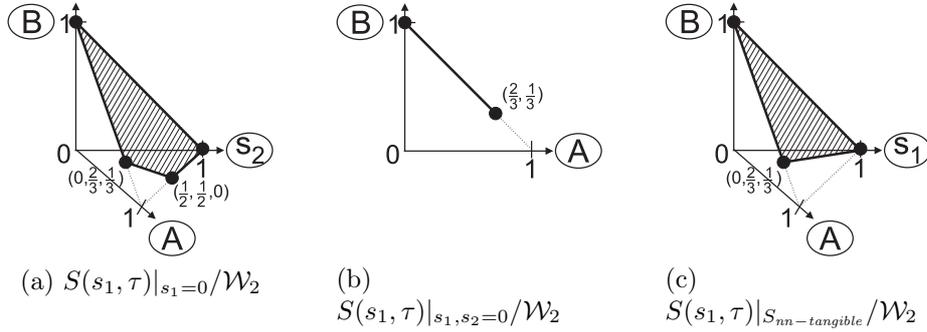


Figure 17: Convex sets

we conclude that $S(s_1, \tau)|_{S_{nn-tangible}}/\mathcal{W}_2$ is the set given in Fig. 17c. We get the same set for $S(t_1, \tau)|_{S_{nn-tangible}}/\mathcal{W}_2$ (here, no nn-vanishing state has to be ignored). Looking at all other sets $S(\cdot, \cdot)$ we find no other splitter, so \mathcal{W}_2 cannot be refined.

With the partition \mathcal{W}_2 and the set of stable states $S_{nn-tangible}$ we have reached our fixed point, the algorithm terminates and we see that s_1 and t_1 are still in the same partition, so they are weakly bisimilar.

Remark 6 (Optimisations). *A few optimisations can be performed:*

- *In every round states without other τ transitions than the loop will always be detected as stable, so this set can be separated as a preprocessing step.*
- *All states with only one single outgoing τ transitions and without non- τ transitions can safely be eliminated in advance (cf. Lemma 3).*

6.2. Complexity considerations

Our algorithm needs to compute the sets $S(s, a)$ and $S_\nu(s, a)$ for all $s \in S$, $a \in Act$ and all distributions ν that can be reached from s by a τ transition driven by a Dirac determinate scheduler. The sets have to be considered with respect to certain partitions. According to Sec. 7 in [4] the problem of finding one of the above sets is already exponential. Further the Dirac determinate schedulers needed to find nn-vanishing states are also exponentially many, as pointed out in Example 1 of [4]. The restriction operation to nn-tangible states is negligible, as the generating points of the above sets, that have non-zero probabilities for nn-tangible states, can simply be omitted to describe the restricted sets.

7. Relation to Deng-Hennesy bisimulation

Recently an alternative distribution-based bisimulation \approx_{bis} for Markov Automata has been defined [6]. One key property of $\mu \approx_{bis} \gamma$ is that whenever $\mu \xrightarrow{a}_C \oplus_{i \in I} p_i \mu_i$ then also $\gamma \xrightarrow{a}_C \oplus_{i \in I} p_i \gamma_i$ where $\mu_i \approx_{bis} \gamma_i$ for all $i \in I$ and vice versa. This assumption can be directly fed into the proof of our Thm. 2 (instead of having to use Lemma 11 from [2]). The reduction to Dirac determinate schedulers works similarly as for the bisimulation \approx , as we (as well as [16]) only use standard arguments for PA which also apply to the Deng-Hennesy setting. Therefore we conclude that our approach is also capable of deciding \approx_{bis} .

8. Related work

Recently, in [8] an alternative approach has been presented to solve the weak bisimulation problem for MA. We now sum up the analogies and differences, omitting the proofs. In the approach of [8], MEC contractedness plays a crucial role:

Definition 18 (Maximal End Components, Definitions 6 and 7 in [8]). *Given a PA $P = (S, Act, T, s_0)$, a maximal end component (mec) is a maximal set $C \subseteq S$ such that for each $s, t \in C : s \xrightarrow{\tau}_C \Delta_t$ and $t \xrightarrow{\tau}_C \Delta_s$. A PA $P = (S, Act, T, s_0)$ is called mec-contracted, if for each pair of states $(s, t) \in S \times S$ it holds that $(s \xrightarrow{\tau}_C \Delta_t \text{ and } t \xrightarrow{\tau}_C \Delta_s) \Rightarrow s = t$.*

Definition 19 (Behaviourally pivotal state [8]). *We call a state s behaviourally pivotal, if $s \xrightarrow{\tau} \mu$ implies that s and μ are not observation equivalent, i.e. $\Delta_s \not\approx \mu$. It is not behaviourally pivotal if there exists (at least) one transition $s \xrightarrow{\tau} \mu$ such that $\Delta_s \approx \mu$.*

Lemma 13 (Relating vanishing states to the definitions of [8]). *On MEC-contracted PA, “vanishing” corresponds to “not behaviourally pivotal” and “tangible” corresponds to “behaviourally pivotal”.*

Note that MEC-contractedness is crucial for this coincidence. Omitting this precondition, the definitions are different: Even if “not behaviourally pivotal” is defined for arbitrary PA, the definition only makes sense for MEC-contracted PA. Look at the PA given in Fig. 18. Of course s and s' are in the same class with respect to \approx_{Δ} . Therefore both s and s' are not behaviourally pivotal, but it does not make sense to think about ignoring

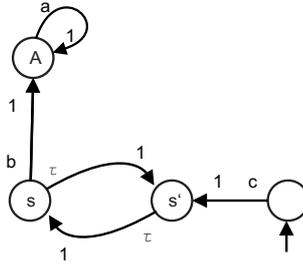


Figure 18: Not behaviourally pivotal vs. vanishing states

both of them. In the context of vanishing states we see that s is tangible while s' is trivially vanishing. The state s' can be safely ignored, that is: eliminated. Note that for MEC-contracted PA this problem doesn't arise. Our approach has a finer granularity: The set of vanishing states is split into the nn-vanishing and naïvely vanishing states. We show in our approach that classes of nn-tangible states (where also naïvely vanishing states belong to) cannot “vanish”, whereas classes of nn-vanishing states can “vanish” without losing weak bisimilarity. Lacking this fundamental difference makes the approach of [8] unnecessarily complicated.

Looking at preserving transitions defined in [8], the picture is similar.

Definition 20 (Preserving Transitions (adapted from [8])). *Let B be an equivalence relation on S . A set P of τ -transitions in T is called preserving with respect to B if for all $(s, \tau, \gamma) \in P$ it holds that whenever $s \xrightarrow{\alpha}_C \mu$ then there exist μ', γ' such that $\mu \xrightarrow{\tau|_P}_C \mu'$ and $\gamma \xrightarrow{\alpha}_C \gamma'$ and $\mu' \equiv_B \gamma'$. Here $|_P$ means that we only use transitions from the set P .*

Lemma 14 (Relating vanishing representations to the definitions of [8]). *On a MEC-contracted PA, let s be a vanishing (i.e. not behaviourally pivotal) state. Then a “vanishing representation” $s \xrightarrow{\tau} \mu$ corresponds to a “preserving transition”.*

Note that it is important that we use *strong* transitions, as preserving transitions are defined as a subset of the set of transitions T (no weak transitions allowed). It can be shown that every vanishing state has such a vanishing representation⁹. Similar to the definition of not behaviourally piv-

⁹When searching for nn-vanishing states, it is not enough to consider only these strong transitions, as example 4 shows.

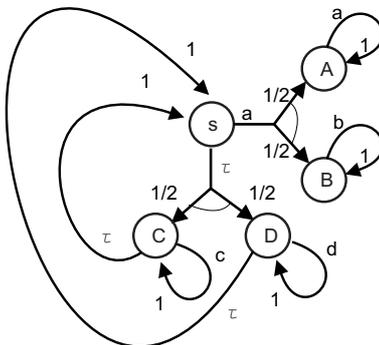


Figure 19: A not contracted example

otal states, also the definition of preserving transitions only makes sense for MEC-contracted PA: The restriction to the set P of preserving transitions is not required for the τ parts of the weak a transition from γ to γ' in Definition 20. Therefore this definition would render the set of all τ transitions in Fig. 19 as “preserving”, but still the a transition from s can clearly not be left out, as from $\gamma = \frac{1}{2}\Delta_C \oplus \frac{1}{2}\Delta_D$ still the a transition from s must be used in order to mimick the transition to $\frac{1}{2}\Delta_A \oplus \frac{1}{2}\Delta_B$. In the context of vanishing states it is clear that there is no vanishing representation for s , as then clearly the a transition would get lost.

We collect the following important differences and coincidences from our approach to the approach of [8]:

- The results of [8] only apply to MEC-contracted PA/MA, while our approach does not preassume MEC-contracted MAs, it is a general approach.
- The concept of *preserving* transitions – which remains rather unspecific in [8] and has to be tackled by a brute force attack over all possible subsets – is nicely explained by our concept of *vanishing representations* consisting of strong transitions. Especially we have shown that it is enough to consider those sets of preserving transitions where each not behaviourally pivotal state has only one emanating preserving transition (this is a direct consequence of our Lemma 10).
- The definitions of [8] only characterise “vanishing” states, but do not distinguish between nn-vanishing and naïvely vanishing states. Therefore that work is lacking the main result similar to our Thm. 1 (nn-vanishing states are the missing part when switching from state-based to distribution-based bisimulations) and Thm. 2 (after elimination of

nn-vanishing states, a weak bisimulation will be a naïve weak bisimulation).

- With our theory we can easily explain the so-called “pitfalls” described in [8], Example 6 and 7. We show that these are no pitfalls at all in the context of nn-vanishing states in Appendix B.
- Regarding the complexity, even if not explicitly mentioned in [8] (but as a consequence of the broken “strong challenger characterisation”), all Dirac determinate schedulers have to be considered for deciding weak bisimilarity between two states. In other words this means that also there the sets $S(s, a)$ are constructed. Therefore, the approach of [8] lies in the same complexity class as our approach.

9. Conclusion

We have shown that weak and naïve weak bisimulation for MA are closely related by an appropriate formulation of elimination and that the two notations coincide, when no non-naïvely vanishing states are present. We have presented an algorithm for deciding weak MA bisimilarity that, as a by-product, finds non-naïvely vanishing states and their corresponding vanishing representations. This can also be used to define normal forms for MA. Even with the magnificent results of [5] it remains an open question whether weak MA bisimulation can be decided in polynomial time.

Acknowledgements: Cordial thanks to Andrea Turrini for giving a beautiful and more readable reformulation of our original definition of nn-vanishing states [7] and some interesting discussions on Markov Automata and bisimulations. We would also like to thank the anonymous reviewers of Information and Computation who indicated problems in the proof of the main Theorem, which finally uncovered a problem in Lemma 16 of [2] and lead to our new proofs of the main Theorems that are independent of [2].

Deutsche Forschungsgemeinschaft (DFG) supported this work under grant SI 710/7-1, and we also acknowledge support by the DFG/NWO Bilateral Research Programme ROCKS.

References

- [1] C. Eisentraut, H. Hermanns, L. Zhang, On Probabilistic Automata in Continuous Time, in: Proceedings of the 2010 25th Annual IEEE Sym-

posium on Logic in Computer Science, LICS '10, IEEE Computer Society, Washington, DC, USA, 2010, pp. 342–351.

- [2] C. Eisentraut, H. Hermanns, L. Zhang, On Probabilistic Automata in Continuous Time, http://www.avacs.org/fileadmin/Publikationen/Open/avacs_technical_report_062.pdf, Reports of SFB/TR 14 AVACS 62 (2010).
- [3] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, G. Franceschinis, Modelling with Generalized Stochastic Petri Nets, Wiley Series in Parallel Computing, 1995.
- [4] S. Cattani, R. Segala, Decision Algorithms for Probabilistic Bisimulation, in: L. Brim, P. Jancar, M. Kretínský, A. Kucera (Eds.), CONCUR, Vol. 2421 of Lecture Notes in Computer Science, Springer, 2002, pp. 371–385.
- [5] H. Hermanns, A. Turrini, Deciding Probabilistic Automata Weak Bisimulation in Polynomial Time, in: D. D’Souza, T. Kavitha, J. Radhakrishnan (Eds.), IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2012), Vol. 18 of Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2012, pp. 435–447. doi:<http://dx.doi.org/10.4230/LIPIcs.FSTTCS.2012.435>. URL <http://drops.dagstuhl.de/opus/volltexte/2012/3879>
- [6] Y. Deng, M. Hennessy, On the semantics of Markov automata, in: Proceedings of the 38th international conference on Automata, languages and programming - Volume Part II, ICALP’11, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 307–318.
- [7] J. Schuster, M. Siegle, Markov Automata: Deciding Weak Bisimulation by means of “non-naïvely” Vanishing States, <http://arxiv.org/abs/1205.6192>, Revised version with appendix on compositionality (initial version May 2012) (2013).
- [8] C. Eisentraut, H. Hermanns, J. Krämer, A. Turrini, L. Zhang, Deciding Bisimilarities on Distributions, in: 10th International Conference on Quantitative Evaluation of SysTems (QEST 2013), 2013, pp. 72–88.

- [9] R. Segala, Modeling and Verification of Randomized Distributed Real-Time Systems, Ph.D. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology (1995).
- [10] H. Hatefi, H. Hermanns, Model Checking Algorithms for Markov Automata, ECEASST 53.
- [11] N. A. Lynch, R. Segala, F. W. Vaandrager, Observing Branching Structure through Probabilistic Contexts, *SIAM J. Comput.* 37 (4) (2007) 977–1013.
- [12] C. Eisentraut, H. Hermanns, L. Zhang, Concurrency and Composition in a Stochastic World, in: P. Gastin, F. Laroussinie (Eds.), *CONCUR 2010 - Concurrency Theory*, Vol. 6269 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2010, pp. 21–39.
- [13] R. Segala, N. A. Lynch, Probabilistic Simulations for Probabilistic Processes, *Nord. J. Comput.* 2 (2) (1995) 250–273.
- [14] M. Ajmone Marsan, S. Donatelli, F. Neri, GSPN models of Markovian multiserver multiqueue systems, *Performance Evaluation* 11 (1990) 227–240.
- [15] G. Ciardo, R. Zijal, Well-defined stochastic Petri nets, in: *Proceedings of the 4th International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, MASCOTS '96*, IEEE Computer Society, Washington, DC, USA, 1996, pp. 274–280.
- [16] C. Eisentraut, H. Hermanns, J. Schuster, A. Turrini, L. Zhang, The Quest for Minimal Quotients for Probabilistic Automata, in: *Tools and Algorithms for the Construction and Analysis of Systems - 19th International Conference, TACAS 2013*, Vol. 7795 of LNCS, Springer, 2013, pp. 16–31.
- [17] J. Schuster, Towards faster numerical solution of Continuous Time Markov Chains stored by symbolic data structures, Ph.D. thesis, Universität der Bundeswehr München (2011).

Appendix A. “Continuous” vs. “nn-vanishing” states

Thm. 2 in [1, 2] relies on Lemma 16 of [2]. There, the concept of “continuous” states is introduced and used for proving a key property of weak bisimulation. However, this appendix points out a counterexample, thus [2, Lemma 16] and therefore Thm. 2 in [2, 1] have to be considered as yet unproven. But in this appendix we also show that with the notion of nn-vanishing states it is possible to prove [2, Lemma 16], thus that the lemma and Thm. 2 in [2, 1] are now known to be indeed correct.

Definition 21 (Continuous state [2]). $P = (S, Act, T, \emptyset, s_0)$ be a PA. A state $s \in S$ that has a transition $s \rightarrow \nu$ where $\Delta_s \approx \nu$, but $\exists t \in Supp(\nu)$ such that $s \not\approx_{\Delta} t$ is called a continuous state.

In order to compare the concepts of “continuous” and “nn-vanishing” states, it is convenient to have an alternative characterisation of nn-vanishing states. By Thm. 1 and Lemma 10 we see that we could alternatively define nn-vanishing states in the following way:

Definition 22 (nn-vanishing state – alternative definition to Definition 14). $P = (S, Act, T, \emptyset, s_0)$ be a PA. A state $s \in S$ that has a (non-combined!) weak transition $s \Rightarrow \nu$ where $\Delta_s \approx \nu$ but $\exists t \in Supp(\nu)$ such that $s \not\approx_{\Delta} t$ is called a nn-vanishing state.

So we see that the set of continuous states is in general smaller than the set of nn-vanishing states (strong transition $s \xrightarrow{\tau} \nu$ vs. weak transition $s \xRightarrow{\tau} \nu$). With this knowledge, we can give a simple example that renders the proof of [2, Lemma 16] wrong.

Example 7 (Counterexample: Weak transitions not considered). *The proof of [2, Lemma 16] consists of two steps. The first step constructs canonical transitions that resolve “continuous” states $\mu \Rightarrow \mu^*$ and $\gamma \Rightarrow \gamma^*$ (where $\mu^* \approx \mu$ and $\gamma^* \approx \gamma$) using Dirac determinate schedulers. In the second step it is shown that every further transition $\mu^* \Rightarrow_C \mu'$ (where $\mu' \approx \mu^*$) and analogously $\gamma^* \Rightarrow \gamma'$ (where $\gamma' \approx \gamma^*$) do not change the equivalence classes. Assume we are given the automata in Fig. A.20. The states s, s' and t are nn-vanishing. Assume that $\mu = \Delta_s, \gamma = \Delta_t$. According to [2] we see that t and s' are continuous while s is not. By the construction from [2] we would get then $\mu^* = \mu$ (no state in μ can change its equivalence class with a strong transition) and $\gamma^* = \frac{1}{2}\Delta_A \oplus \frac{1}{2}\Delta_B$. But now it is trivially wrong that μ^* and*

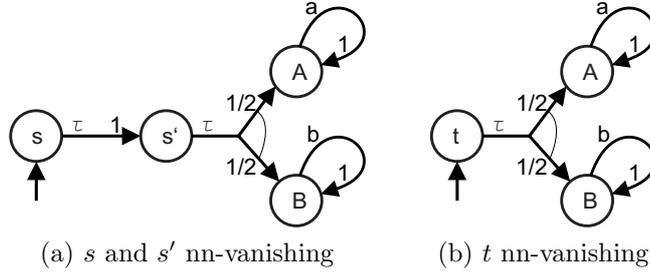


Figure A.20: Examples of vanishing states

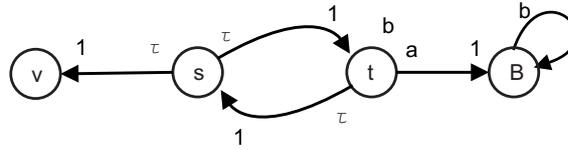


Figure B.21: Example 6 from [8] – s and t in one class

γ^* coincide on classes, as s is clearly nn-vanishing while A and B are not. Therefore the proof of [2, Lemma 16] is incorrect.

Still, [2, Lemma 16] remains correct, and its correctness can be proven in the nn-vanishing context: We can substitute every nn-vanishing state by its canonical vanishing representation consisting of nn-tangible states, and the resulting distributions are naïvely weakly bisimilar (with respect to the lifting of $\approx_{\text{naïve}}$ to distributions). This is, with the help of our notion of nn-vanishing states, now proven by Lemma 8 and Thm. 2. The fact that non-combined transitions can be used in this lemma is now proven by our Lemma 10.

Appendix B. Examples from [8]

The following two examples from [8] are called “pitfalls” there. We show why – in the context of the nn-vanishing state concept – these pitfalls are no pitfalls at all.

Appendix B.1. Strong challenger characterisation

In example 6 of [8] (cf. Fig. B.21) it is clear that the preserving-approach fails as long as v remains absorbing: States s and t belong to one class with respect to \approx_{Δ} . It can be easily verified that both states cannot be

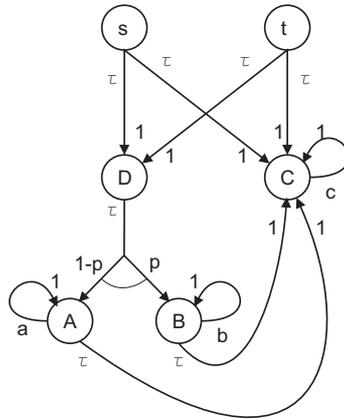


Figure B.22: An instance of Example 7 from [8] – nn-vanishing case

vanishing, as no vanishing representation can be found. As both states are tangible, they actually don't need a special treatment in our algorithm. Note that “preserving transitions” in our understanding are only necessary for nn-vanishing states and *not* for all vanishing states in order to solve the decision problem.

Appendix B.2. Brute force attack for “preserving” transitions

The problem of example 7 of [8], as exemplified by Fig. B.22, doesn't hit the bull's eye. The basic question is not “which transitions can be omitted?” (or alternatively “which transitions are preserving?”). The first question must rather be “are states s and t nn-vanishing or not?”. If they are nn-tangible, not any transition may be omitted. If they are nn-vanishing, Lemma 10 justifies that for both states the *same* transition must be omitted (as long as the successor distributions are *not* bisimilar), as Fig. B.22 shows (assume that the “triangle” (“pentagon”) distribution from [8] corresponds to state C (D) in our example. Obviously the automaton is MEC-contracted. Clearly states A , B and C are not weakly bisimilar. Assume that $p \in (0, 1)$. Then D is trivially nn-vanishing. Further it is clear that the τ transitions from s and t to C can be omitted, as they may be weakly mimicked. But then (and only then) we have vanishing representations of states s and t . So we conclude that the statement in [8] that “Then, clearly, none of the transitions is preserving” is in general wrong. It rather depends on the context whether s or t are nn-vanishing or not.