RWTH Aachen

Department of Computer Science
Technical Report

Computing Game Metrics on Markov Decision Processes

Hongfei Fu

ISSN 0935–3232 · Aachener Informatik-Berichte · AIB-2012-08

RWTH Aachen · Department of Computer Science · May 2012

Computing Game Metrics on Markov Decision Processes

Hongfei Fu

Lehrstuhl für Informatik II
RWTH Aachen, Germany
Email: hongfeifu@informatik.rwth-aachen.de

Abstract. In this paper we study the complexity of computing the game bisimulation metric defined by de Alfaro *et al.* on Markov Decision Processes. It is proved by de Alfaro *et al.* that the undiscounted version of the metric is characterized by a quantitative game μ-calculus defined by de Alfaro and Majumdar, which can express reachability and ω-regular specifications. And by Chatterjee *et al.* that the discounted version of the metric is characterized by the discounted quantitative game μ-calculus. In the discounted case, we show that the metric can be computed exactly by extending the method for Labelled Markov Chains by Chen *et al.* And in the undiscounted case, we prove that the problem whether the metric between two states is under a given threshold can be decided in $\mathsf{NP}\cap\mathsf{coNP}$, which improves the previous PSPACE upperbound by Chatterjee *et al.*

1 Introduction

In recent years, probabilistic behavioral equivalences have been extensively studied. Many equivalence notions for probabilistic systems such as probabilistic (bi)simulation [LS91,SL95,JL91] have been established. And many efficient algorithms have been proposed for these notions [BEMC00,CS02]. Generally, probabilistic (bi)simulation is a class of formal notions judging whether two probabilistic systems are equivalent. In practical situations, they are often used to compare if the implemented system is semantically equivalent to the specification. One can also tackle the state explosion problem in model checking by reducing a large probabilistic system into its (possibly much smaller) quotient system w.r.t probabilistic (bi)simulation. This is because the quotient system is equivalent to the original one in the sense that they satisfy the same set of logical formulae [ASB95,SL95].

However, the definition of probabilistic bisimulation relies on exact probability values and a slight variation of probability values will differentiate two originally equivalent systems. In this sense, probabilistic bisimulation is too restrictive and not robust, and a notion of *approximate bisimulation* or *bisimilarity metric* is needed. In the context of approximate bisimulation, the difference of two states is measured by a value in $[0, 1]$, rather than by a boolean argument stating they are either "equal" or "different". This yields a smooth, quantitative notion of probabilistic bisimulation. The smaller the value, the more alike the behaviours of the two states are. In particular, the value is zero if they are probabilistic bisimilar. In practical situations, the value for the difference would suggest if one component can substitute another: if the value between them is small enough, then one may choose the cheaper component.

The notion of approximate bisimulation is first considered by Giacalone *et al* [GJS90]. They defined a notion of "ϵ-bisimilarity" to measure the distance between two probabilistic processes encoded in the PCCS calculus, which extends Milner's CCS [Mil89] with probabilities. Then various notions of approximate bisimulation are defined on discrete-time probabilistic systems such as Labelled Markov Chains (LMC) [vBW01b,vBW01a,vBSW08], Markov Decision Processes (MDP) [FPP04,dAMRS07,DLT08,TDZ11,DJGP02] and Concurrent Games [dAMRS07], and continuous-time probabilistic systems such as Labelled Markov Processes (LMP) [Pan09], Continuous-Time Markov Chains (CTMC) [GJP04] and Stochastic Hybrid Systems [JGP06].

Here we focus on the bisimilarity metric on concurrent games by de Alfaro *et al.* [dAMRS07], called *game bisimulation metric*. It is proved that this metric is characterized by a quantitative game μ-calculus [dAM04], where various properties such as the maximum reachability (to reach some set) and the maximum recurrent reachability (to reach some set infinitely often) can be expressed. This means that this metric serves as the exact bound for the differences of these properties across states. Furthermore, Chatterjee *et al.* [CdAMR10] proved that this metric is a tight bound for the difference in long-run average and discounted average behaviors. In this paper, we will also study a discounted version of this game bisimulation metric [dAHM03,CdAMR10]. In the discounted version, future difference is discounted by a factor and does not contribute fully to the metric, which is in contrast to the case of the original undiscounted metric. Analogous to the undiscounted version, the discounted metric is characterized by a discounted quantitative μ-calculus [dAHM03,CdAMR10].

If one restricts the game bisimulation metric to MDPs (a turn-based degenerate class of concurrent games) and LMCs (MDPs without nondeterminism), one can obtain a metric on MDPs and LMCs, respectively. In this paper we consider the game bisimulation metric [dAMRS07] on MDPs. We briefly compare this metric to another two metrics on Markov Decision Processes which are related with strong probabilistic bisimulation, namely the metrics by Ferns *et al.* [FPP04] and Desharnais *et al.* [DLT08,TDZ11]. The three metrics are different. Both the game bisimulation metric and the metric by Ferns *et al.* are defined as a least fixpoint on the lattice of pseudometrics, however the latter focuses on the difference in accumulated rewards. The one by Desharnais is defined directly as a binary relation and focuses on one-step difference. It is shown that the metric by Desharnais *et al.* is PTIME-decidable [TDZ11]. However, this metric does not serve as a bound for properties such as the reachability probability to some state labelled with a. This can be observed in [DLT08, Example 7], where we may label a on states s_n and t_n. In this example, $d(s,t) \leq 0.1$. However the difference between s, t in the probability to reach $\{s_n, t_n\}$ is $1 - 0.95^n$, which approaches 1 when n goes to infinity. On the other hand, the game bisimulation metric serves as a bound for this property, since this property can be encoded in the quantitative game μ-calculus [dAM04]. It is also worth noting that the game bisimulation metric on LMCs coincides with the metric by van Breugel *et al.* [vBW01b,vBW01a,vBSW08].

In this paper, we study the complexity of computing the discounted and undiscounted game bisimulation metric [dAMRS07,dAHM03,CdAMR10] on Markov Decision Processes. It is shown by Chatterjee *et al.* [CdAMR10] that the undiscounted metric can be decided in PSPACE. In other words, one can decide in PSPACE whether the undiscounted metric between two states in an MDP is under a given threshold. And very recently, Chen *et al.* [CvBW12] proved that the undiscounted metric is PTIME-decidable on LMCs. Here we prove that the undiscounted metric (on MDPs) can be decided in $\mathsf{NP} \cap \mathsf{coNP}$, which is one-step closer to obtain the PTIME-decidability of the problem. We prove this result by establishing a notion of "self-closed" sets, which in some sense characterizes this metric. We remark that the method devised by Chen *et al.* [CvBW12] cannot be (at least directly) extended to Markov Decision Processes. This is because their method heavily relies on the fact that the metric is the unique fixpoint of a bisimulation-minimal LMC, which generally is not the case on MDPs. For the discounted case, we show that the discounted metric can be computed exactly in polynomial time by simply extending the method by Chen *et al.* [CvBW12] for LMCs.

The organization of this paper is as follows: Section 2 introduces Markov Decision Processes. Section 3 introduces the discounted and undiscounted game bisimulation metric on MDPs. In Section 4 we discuss approximations of the game metrics, where we derive that the discounted metric on MDPs can be computed exactly in polynomial time. Then in Section 5 we prove that the undiscounted metric on MDPs can be decided in $\mathsf{NP} \cap \mathsf{coNP}$. Section 6 concludes the paper.

2 Markov Decision Processes

We define Markov Decision Processes (MDP) in the context of game structures, following the definitions in [dAMRS07].

Definition 1. *Let S be a finite set. A function $\mu : S \to [0,1] \cap \mathbb{Q}$ is a* probability distribution *over S if $\sum_{s \in S} \mu(s) = 1$. We denote the set of probability distributions over S by* $\mathrm{Dist}(S)$*.*

Definition 2. *A* Markov Decision Process *is a tuple $(S, \mathcal{V}, [\cdot], Moves, \Gamma, \delta)$ which consists of the following components:*

- *A finite set $S = \{s, t \ldots\}$ of* states*;*
- *A finite set $\mathcal{V}$ of* observational variables*;*
- *A variable interpretation $[\cdot] : \mathcal{V} \times S \mapsto [0,1] \cap \mathbb{Q}$, which associates with each variable $\mathrm{v} \in \mathcal{V}$ a valuation $[\mathrm{v}]$;*
- *A finite set $Moves = \{a, b \ldots\}$ of* moves*;*
- *A move assignments $\Gamma : S \mapsto 2^{Moves} \backslash \emptyset$, which associates with each state $s \in S$ the nonempty set $\Gamma(s) \subseteq Moves$ of moves available at state s.*
- *A probabilistic transition function $\delta : S \times Moves \mapsto \mathrm{Dist}(S)$, which gives the probability $\delta(s,a)(t)$ of a transition from s to t through the move $a \in \Gamma(s)$.*

Intuitively, Γ is the set of moves available at each state which can be controlled by a (sole) player that tries to maximize or minimize certain property.

Below we define *mixed moves* [dAMRS07] on a Markov Decision Process. Intuitively, a mixed move is a probabilistic combination of single moves. This notion corresponds to randomized strategies on Markov Decision Processes, which coincides with combined transitions defined in [SL95].

Definition 3. *Let $(S, \mathcal{V}, [\cdot], Moves, \Gamma, \delta)$ be an MDP. A* mixed move *at state $s \in S$ is a probability distribution over $\Gamma(s)$. We denote by $\mathcal{D}(s) = \mathrm{Dist}(\Gamma(s))$ the set of mixed moves at state s. We extend the probability transition function δ to mixed moves as follows: for $s \in S$ and $x \in \mathcal{D}(s)$, we define $\delta(s,x)$ by $\delta(s,x)(t) := \sum_{a \in \Gamma(s)} x(a) \cdot \delta(s,a)(t)$, for $t \in S$.*

3 Game Metrics on Markov Decision Processes

In this section we define discounted and undiscounted game bisimulation metrics on MDPs [dAMRS07,CdAMR10]. Both of them are defined as a least fixpoint on the complete lattice of pseudometrics w.r.t an MDP. For technical reasons we also extend these definitions to premetrics, a wider class of pseudometrics.

Below we fix an MDP $(S, \mathcal{V}, [\cdot], Moves, \Gamma, \delta)$. The following definition illustrates the concepts of premetrics and pseudometrics.

Definition 4. *A function $d : S \times S \to [0,1]$ is a* premetric *iff $d(s,s) = 0$ for all $s \in S$. A premetric d is further a* pseudometric *iff for all $r,s,t \in S$, $d(s,t) = d(t,s)$ (symmetry) and $d(r,t) \le d(r,s) + d(s,t)$ (triangle inequality). We denote the set of premetrics (*resp. *pseudometrics) by $\mathcal{M}_{\mathrm{r}}$ (*resp. *$\mathcal{M}_{\mathrm{p}}$).*

Given $d_1, d_2 \in \mathcal{M}_\kappa$ (where $\kappa \in \{\mathrm{r}, \mathrm{p}\}$), we define the partial order $d_1 \le d_2$ in the pointwise fashion, i.e., $d_1 \le d_2$ iff $d_1(s,t) \le d_2(s,t)$ for all $s,t \in S$. It is not hard to prove the following lemma [Pan09,vBSW08,dAMRS07].

Lemma 1. *For $\kappa \in \{\mathrm{r}, \mathrm{p}\}$, the structure $(\mathcal{M}_\kappa, \le)$ is a complete lattice.*

We concern the least fixpoint of $(\mathcal{M}_{\mathrm{p}}, \le)$ w.r.t a monotone function H^α, where $\alpha \in [0,1]$ is a discount factor. The function H^α is defined as follows.

Definition 5. *Given $\mu, \nu \in \mathrm{Dist}(S)$, we define $\mu \otimes \nu$ as the following set*

$$\{\lambda : S \times S \mapsto [0,1] \mid (\forall u \in S. \sum_{v \in S} \lambda(u,v) = \nu(u)) \wedge (\forall v \in S. \sum_{u \in S} \lambda(u,v) = \mu(v))\}$$

Further we lift $d \in \mathcal{M}_{\mathrm{r}}$ to $\mu, \nu \in \mathrm{Dist}(S)$ as follows,

$$d(\mu,\nu) := \inf_{\lambda \in \mu \otimes \nu} \left(\sum_{u,v \in S} d(u,v) \cdot \lambda(u,v) \right)$$

Then the function $H^\alpha_\kappa : \mathcal{M}_\kappa \mapsto \mathcal{M}_\kappa$ is defined as follows: given $d \in \mathcal{M}_\kappa$, $H^\alpha_\kappa(d)(s,t) = \max\{p(s,t), \alpha \cdot H_1(d)(s,t), \alpha \cdot H_2(d)(s,t)\}$ for $s,t \in S$, for which:

- $p(s,t) = \max_{\mathrm{v} \in \mathcal{V}} |[\mathrm{v}](s) - [\mathrm{v}](t)|$;
- $H_1(d)(s,t) = \sup_{a \in \Gamma(s)} \inf_{y \in \mathcal{D}(t)} d(\delta(s,a), \delta(t,y))$;
- $H_2(d)(s,t) = H_1(d)(t,s)$.

We denote by d^α_κ the least fixpoint of H^α_κ, i.e., $d^\alpha_\kappa = \bigsqcap\{d \in \mathcal{M}_\kappa \mid H^\alpha_\kappa(d) \le d\}$.

One can verify that H^α_κ is indeed a monotone function on $(\mathcal{M}_\kappa, \le)$ [CdAMR10,dAMRS07]. The pseudometric d^1_{p} corresponds to the undiscounted game bisimulation metric [dAMRS07], and the pseudometric d^α_{p} with $\alpha \in [0,1)$ corresponds to the discounted metric with discount factor α [CdAMR10]. Note that the definitions of H^α_κ and d^α_κ take a different form from the original ones [dAMRS07,CdAMR10] which cover concurrent games. However by [CdAMR10, Lemma 1 and Lemma 2], these two definitions are equivalent on Markov Decision Processes.

Note that the set $\mu \otimes \nu$ is a bounded polyhedron on the vector space $S \times S \mapsto \mathbb{R}$. Thus $d(\mu,\nu)$ equals the optimal value of the linear programming (LP) problem with feasible region $\mu \otimes \nu$ and objective function $\min \sum_{u,v \in S} d(u,v) \cdot \lambda(u,v)$. We denote by $\mathrm{OP}[d](\mu,\nu)$ the set of optimum solutions that reach the optimal value $d(\mu,\nu)$.

Remark 1. It is proved in [CdAMR10] that for $d \in \mathcal{M}_\mathrm{r}$, $s, t \in S$ and $a \in \Gamma(s)$, the value $h[d](s,a,t) := \inf_{y \in \mathcal{D}(t)} d(\delta(s,a), \delta(t,y))$ equals the optimal value of the LP problem $\mathrm{LP}[d](s,a,t)$ whose variables are $\{\lambda_{u,v}\}_{u,v \in S}$ and $\{y_b\}_{b \in \Gamma(t)}$, whose objective function is $\min \sum_{u,v \in S} d(u,v) \cdot \lambda_{u,v}$, and whose feasible region is the bounded polyhedron $P(s,a,t)$ specified by:

$$\sum_{v \in S} \lambda_{u,v} = \sum_{b \in \Gamma(t)} \delta(t,b)(u) \cdot y_b \qquad \sum_{u \in S} \lambda_{u,v} = \delta(s,a)(v)$$
$$\sum_{b \in \Gamma(t)} y_b = 1 \qquad \lambda_{u,v} \geq 0 \qquad y_b \geq 0$$

Thus $h[d](s,a,t)$ (and $H^\alpha_\kappa(d)$) can be computed in polynomial time [CdAMR10, Theorem 4.3]. Further there is $y^* \in \mathcal{D}(t)$ such that $h[d](s,a,t) = d(\delta(s,a), \delta(t,y^*))$, where y^* can be $\{y^*_b\}_{b \in \Gamma(t)}$ of some optimum solution $\{\lambda^*_{u,v}\}_{u,v \in S}$, $\{y^*_b\}_{b \in \Gamma(t)}$.

Below we give an example. Consider the MDP with states $\{s_1, s_2, t_1, t_2\}$, moves $\{a, b\}$ and variables $\{\mathrm{u}_1, \mathrm{u}_2, \mathrm{v}\}$. The variable evaluation is specified by: $\mathrm{u}_i(t_i) = 1$ and $\mathrm{u}_i(s) = 0$ for $s \neq t_i$; $\mathrm{v}(s_1) = \mathrm{v}(s_2) = 1$ and $\mathrm{v}(s) = 0$ for $s \in S \backslash \{s_1, s_2\}$. The probability transition function is depicted in Fig. 1. One can verify that the set

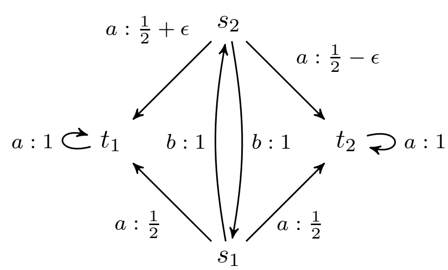

Fig. 1.

of undiscounted fixpoints on this MDP is

$$\{d \in \mathcal{M}_\mathrm{p} \mid d(s_1, s_2) \in [\epsilon, 1], d(s,t) = 1 \text{ if } s \neq t \text{ and } \{s,t\} \neq \{s_1, s_2\}\}.$$

Note that even in this simple example which is bisimulation minimal, there exists no unique fixpoint. This is in contrast to the case on Labelled Markov Chains [CvBW12].

4 Approximations of the Game Metrics

In this section we describe the approximation of discounted and undiscounted game metric by Picard's Iteration. Then we prove that the discounted metric can be computed exactly in polynomial time by simply extending the method by Chen *et al.* [CvBW12].

We fix an MDP $(S, \mathcal{V}, [\cdot], Moves, \Gamma, \delta)$. The size of the MDP, denoted by M, is the space needed to store the MDP, where all numerical values appearing in $\mathcal{V}$ and δ are represented in binary. We denote by $\|\alpha\|$ the space needed to represent the rational number α in binary. Below we define approximations of d^α_κ.

Definition 6. *The family $\{d^{\kappa,\alpha}_i\}_{i \in \mathbb{N}_0}$ of approximants of d^α_κ is inductively defined as follows: $d^{\kappa,\alpha}_0 := \mathbf{0}$ (i.e., $d^{\kappa,\alpha}_0(s,t) = 0$ for all $s,t \in S$); $d^{\kappa,\alpha}_{i+1} = H^\alpha_\kappa(d^{\kappa,\alpha}_i)$. We* *denote by* $d_\omega^{\kappa,\alpha}$ *the limit of* $\{d_i^{\kappa,\alpha}\}_{i\in\mathbb{N}_0}$*, i.e.,* $d_\omega^{\kappa,\alpha}(s,t) = \lim_{i\to\infty} d_i^{\kappa,\alpha}(s,t)$ *for all* $s,t \in S$*. (By monotonicity of* H_κ^α*, one can prove inductively that* $d_i^{\kappa,\alpha} \le d_{i+1}^{\kappa,\alpha}$ *for* $i \in \mathbb{N}_0$*. Thus* $d_\omega^{\kappa,\alpha}$ *exists.)*

For $\alpha < 1$, it is not hard to prove that $d_\omega^{\kappa,\alpha} = d_\kappa^\alpha$ since H_κ^α is a contraction mapping. Below we define $\|d_1 - d_2\| = \max_{s,t\in S} |d_1(s,t) - d_2(s,t)|$.

Lemma 2. $d_\omega^{\kappa,\alpha} = d_\kappa^\alpha$ *for* $\alpha \in [0,1)$*.*

Proof. We prove inductively that $\|d_i^{\kappa,\alpha} - d_\kappa^\alpha\| \le \alpha^i$. The situation when $i = 0$ is clear. Suppose $\|d_i^{\kappa,\alpha} - d_\kappa^\alpha\| \le \alpha^i$, we prove that $\|d_{i+1}^{\kappa,\alpha} - d_\kappa^\alpha\| \le \alpha^{i+1}$. This can be observed as follows:

$$\begin{aligned}
&|d_{i+1}^{\kappa,\alpha}(s,t) - d_\kappa^\alpha(s,t)| \\
&= |H_\kappa^\alpha(d_i^{\kappa,\alpha})(s,t) - H_\kappa^\alpha(d_\kappa^\alpha)(s,t)| \\
&\le \alpha \cdot \max\{\max_{a\in\Gamma(s)} |h[d_i^{\kappa,\alpha}](s,a,t) - h[d_\kappa^\alpha](s,a,t)|, \\
&\quad \max_{b\in\Gamma(t)} |h[d_i^{\kappa,\alpha}](t,b,s) - h[d_\kappa^\alpha](t,b,s)|\}
\end{aligned}$$

where $h[d](s,a,t)$ is defined in Remark 1, which is the optimal value of the objective function $\sum_{u,v} d(u,v)\cdot\lambda_{u,v}$ over the polyhedron $P(s,a,t)$. Note that

$$|\sum_{u,v} d_i^{\kappa,\alpha}(u,v)\cdot\lambda_{u,v} - \sum_{u,v} d_\kappa^\alpha(u,v)\cdot\lambda_{u,v}| \le \|d_i^{\kappa,\alpha} - d_\kappa^\alpha\| \le \alpha^i$$

for all $\{\lambda_{u,v}\}_{u,v\in S}, \{y_b\}_{b\in\Gamma(s)} \in P(s,a,t)$. Thus $|h[d_i^{\kappa,\alpha}](s,a,t) - h[d_\kappa^\alpha](s,a,t)| \le \alpha^i$ and $|h[d_i^{\kappa,\alpha}](t,b,s) - h[d_\kappa^\alpha](t,b,s)| \le \alpha^i$ for all $a \in \Gamma(s)$ and $b \in \Gamma(t)$. So we obtain $\|d_{i+1}^{\kappa,\alpha} - d_\kappa^\alpha\| \le \alpha^{i+1}$. □

The situation when $\alpha = 1$ is not that direct, as H_κ^1 is not necessarily a contraction mapping. This situation is handled in the following lemma.

Lemma 3. [dAMRS07] $d_\omega^{\kappa,1} = d_\kappa^1$.

Proof. We prove that $d_\omega^{\kappa,1} \le d_\kappa^1$ and $d_\kappa^1 \le d_\omega^{\kappa,1}$. Below we abbreviate $d_i^{\kappa,1}$ as d_i^κ, $d_\omega^{\kappa,1}$ as d_ω^κ and d_κ^1 as d_κ. The case "$d_\omega^\kappa \le d_\kappa$" follows immediately from the fact that (i) $d_0^\kappa \le d_\kappa$ and (ii) H_κ^1 is monotone, and an inductive argument on the construction of $\{d_i^\kappa\}_{i\in\mathbb{N}_0}$. To prove "$d_\kappa \le d_\omega^\kappa$", we prove that d_ω^κ is a pre-fixpoint of H_κ^1 (i.e., $H_\kappa^1(d_\omega^\kappa) \le d_\omega^\kappa$).

Consider arbitrary $(s,t) \in S\times S$ and $a \in \Gamma(s)$. By the definition of $\{d_i^\kappa\}_{i\in\mathbb{N}_0}$, for all $i \in \mathbb{N}_0$, there exists $y \in \mathcal{D}(t)$ such that $d_i^\kappa(\delta(s,a),\delta(t,y)) \le d_{i+1}^\kappa(s,t) \le d_\omega^\kappa(s,t)$. Define $\mathcal{D}_i := \{y \in \mathcal{D}(t) \mid d_i^\kappa(\delta(s,a),\delta(t,y)) \le d_\omega^\kappa(s,t)\}$. Then by the analysis above $\mathcal{D}_i$ is nonempty. Further by $d_i^\kappa \le d_{i+1}^\kappa$, $d_i^\kappa(\delta(s,a),\delta(t,y)) \le d_{i+1}^\kappa(\delta(s,a),\delta(t,y))$ for all $y \in \mathcal{D}(t)$. Thus $\{\mathcal{D}_i\}_{i\in\mathbb{N}_0}$ is a decreasing sequence. Further we prove that each $\mathcal{D}_i$ is compact.

Suppose $\{y_j\}_{j\in\mathbb{N}}$ is a sequence in $\mathcal{D}_i$ that converges to some $\overline{y}$. Fix some $\lambda_j \in \mathrm{OP}[d_i^\kappa](\delta(s,a),\delta(t,y_j))$. By the boundedness of $\{\lambda_j\}_{j\in\mathbb{N}}$, there is a subsequence $\{\lambda_{j_k}\}_{k\in\mathbb{N}}$ that converges to some vector $\overline{\lambda}$. Since $\lim_{k\to\infty} y_{j_k} = \overline{y}$ and $\lim_{k\to\infty}\lambda_{j_k} = \overline{\lambda}$, we have $\overline{\lambda} \in \delta(s,a)\otimes\delta(t,\overline{y})$. Further by

$$\begin{aligned}
d_i^\kappa(\delta(s,a),\delta(t,\overline{y})) &\le \textstyle\sum_{u,v\in S} d_i^\kappa(u,v)\cdot\overline{\lambda}(u,v) \\
&= \lim_{k\to\infty} \textstyle\sum_{u,v\in S} d_i^\kappa(u,v)\cdot\lambda_{j_k}(u,v) \\
&= \lim_{k\to\infty} d_i^\kappa(\delta(s,a),\delta(t,y_{j_k})) \\
&\le d_\omega^\kappa(s,t)
\end{aligned}$$

We have $\overline{y} \in \mathcal{D}_i$. Thus $\{\mathcal{D}_i\}$ is a decreasing sequence of nonempty compact sets, which implies that $\bigcap_{i\in\mathbb{N}_0} \mathcal{D}_i$ is non-empty. In other words, there is $y^* \in \mathcal{D}(t)$ such that for all $i \in \mathbb{N}_0$, $d_i^\kappa(\delta(s,a),\delta(t,y^*)) \leq d_\omega^\kappa(s,t)$.

The last step is to prove that $\lim_{i\to\infty} d_i^\kappa(\delta(s,a),\delta(t,y^*)) = d_\omega^\kappa(\delta(s,a),\delta(t,y^*))$, from which we can deduce that $d_\omega^\kappa(\delta(s,a),\delta(t,y^*)) \leq d_\omega^\kappa(s,t)$. To prove this, we observe that for all $d \in \mathcal{M}_\kappa$,

$$d(\delta(s,a),\delta(t,y^*)) = \min\{\sum_{u,v\in S} \lambda(u,v)\cdot d(u,v) \mid \lambda \in V(\delta(s,a)\otimes\delta(t,y^*))\}$$

where $V(\delta(s,a)\otimes\delta(t,y^*))$ is the set of vertices of the polyhedron $\delta(s,a)\otimes\delta(t,y^*)$. Since the number of vertices of a polyhedron is finite [Sch86], $d(\delta(s,a),\delta(t,y^*))$ (viewed as a function on vector d) is continuous on d. Then the result follows.

Then we obtain $\inf_{y\in\mathcal{D}(t)} d_\omega^\kappa(\delta(s,a),\delta(t,y)) \leq d_\omega^\kappa(s,t)$. Similarly, we can prove that $\inf_{x\in\mathcal{D}(s)} d_\omega^\kappa(\delta(t,b),\delta(s,x)) \leq d_\omega^\kappa(s,t)$ for all $s,t \in S$ and $b \in \Gamma(t)$. Also note that $p(s,t) = d_1^\kappa(s,t) \leq d_\omega^\kappa(s,t)$. Thus $H_\kappa^1(d_\omega^\kappa) \leq d_\omega^\kappa$. □

Here we derive a corollary from Lemma 3 which states that $d_\mathrm{r}^1 = d_\mathrm{p}^1$. This allows us to reason about d_p^1 on the lattice of premetrics.

Corollary 1. $d_\mathrm{r}^1 = d_\mathrm{p}^1 = \bigsqcap\{d \in \mathcal{M}_\mathrm{r} \mid H_r^1(d) \leq d\}$.

Proof. This follows directly from Lemma 3 and the fact that $d_0^{\mathrm{r},1} = d_0^{\mathrm{p},1}$ (which implies $d_\omega^{\mathrm{r},1} = d_\omega^{\mathrm{p},1}$). □

Below we follow Chen *et al.* [CvBW12] to prove that for a fixed $\alpha \in [0,1)$, d_κ^α can be computed exactly in polynomial time. The method is divided into three steps. The first step is to prove that d_κ^α is a rational vector of polynomial size.

Lemma 4. *For $\alpha \in [0,1] \cap \mathbb{Q}$, d_κ^α is a rational vector of size polynomial in M and $\|\alpha\|$.*

Proof. We prove that d_κ^α (deemed as a vector) is a vertex of a LP problem of size polynomial in M and $\|\alpha\|$, which implies the desired result [Sch86, Theorem 10.2]. We construct the LP problem as follows:

1. For each $s,t \in S$ and $a \in \Gamma(s)$, we can choose a vertex $\{\lambda(s,a,t)_{u,v}\}_{u,v\in S}$, $\{y(s,a,t)_b\}_{b\in\Gamma(t)}$ of $P(s,a,t)$ that is an optimum solution of LP$[d_\kappa^\alpha](s,a,t)$ (cf. Remark 1) [Sch86, Section 8]. By [Sch86, Theorem 10.2], these vertices are of size polynomial in M.
2. The LP problem to be constructed is as follows: the variables are $\{d_{s,t}\}_{s,t\in S}$, the objective function is $\min \sum_{s,t\in S} d_{s,t}$, and the feasible region is specified by:
   (a) $d_{s,s} = 0$, $d_{r,t} \leq d_{r,s} + d_{s,t}$ and $0 \leq d_{s,t} \leq 1$ for $r,s,t \in S$;
   (b) $d_{s,t} \geq p(s,t)$ for $s,t \in S$;
   (c) $d_{s,t} \geq \alpha \cdot \sum_{u,v\in S} \lambda(s,a,t)_{u,v} \cdot d_{u,v}$ for $(s,t) \in S\times S$ and $a \in \Gamma(s)$;
   (d) $d_{s,t} \geq \alpha \cdot \sum_{u,v\in S} \lambda(t,b,s)_{u,v} \cdot d_{u,v}$ for $(s,t) \in S\times S$ and $b \in \Gamma(t)$.

We prove that $\{d_\kappa^\alpha(s,t)\}_{s,t\in S}$ is the unique optimum solution of this LP problem. First note that d_κ^α is a feasible solution of this LP problem since d_κ^α is a fixpoint of H_κ^α. Then suppose that $\{d(s,t)\}_{s,t\in S}$ is a feasible solution of this LP.

Since $\{\{\lambda(s,a,t)_{u,v}\}_{u,v\in S}, \{y(s,a,t)_b\}_{b\in\Gamma(t)}\}$ is a vertex of $P(s,a,t)$, we obtain $\sum_{u,v\in S}\lambda(s,a,t)_{u,v}\cdot d(u,v) \geq h[d](s,a,t)$. This implies that $d(s,t) \geq \alpha\cdot h[d](s,a,t)$ and $d(s,t) \geq \alpha \cdot h[d](t,b,s)$ for all $a \in \Gamma(s)$ and $b \in \Gamma(t)$. So by Definition 5, $H^{\alpha}_{\kappa}(d) \leq d$. Thus $d^{\alpha}_{\kappa} \leq d$ and consequently $\sum_{s,t\in S} d^{\alpha}_{\kappa}(s,t) \leq \sum_{s,t\in S} d(s,t)$. So d^{α}_{κ} is the unique optimum solution of this LP problem, which is also a vertex of this LP. Thus d^{α}_{κ} is a rational vector of size polynomial in M and $\|\alpha\|$. □

The second step is to prove that d^{α}_{κ} can be approximated to polynomially many bits of precision in polynomial time.

Lemma 5. *For a fixed $\alpha \in [0,1) \cap \mathbb{Q}$ and any $\epsilon > 0$, we can compute, in polynomial time in M and $\|\epsilon\|$, a $d \in \mathcal{M}_{\mathrm{p}}$ such that $\|d - d^{\alpha}_{\kappa}\| \leq \epsilon$.*

Proof. From the proof of Lemma 2, we see that $\|d^{\kappa,\alpha}_{\lceil\log\epsilon/\log\alpha\rceil} - d^{\alpha}_{\kappa}\| \leq \epsilon$. We show that $d^{\kappa,\alpha}_{\lceil\log\epsilon/\log\alpha\rceil}$ can be computed in polynomial time in M and $\|\epsilon\|$. Since $d^{\kappa,\alpha}_{\lceil\log\epsilon/\log\alpha\rceil}$ can be computed by iterating H^{α}_{κ} $\lceil\log\epsilon/\log\alpha\rceil$ times, we only need to prove that each step in the iteration can be done in polynomial time. We prove this by showing that the number of bits needed to represent $d^{\kappa,\alpha}_{i}$ in each iteration is polynomial. More specifically, we prove by induction that the least common multiplier of the denominators of entries in $d^{\kappa,\alpha}_{k}$, denoted by $T(k)$, is bounded by $m^k \cdot 2^{\mathcal{O}(kM^8)}$ where m is the denominator of α. The base step $k = 0$ is clear since $T(0) = 1$. We prove the inductive step by clarifying the recursive relation between $T(k+1)$ and $T(k)$. Let $(s,t) \in S \times S$. By $d^{\kappa,\alpha}_{k+1}(s,t) = H^{\alpha}_{\kappa}(d^{\kappa,\alpha}_{k})(s,t)$, Definition 5 and Remark 1, there are three cases:

1. $d^{\kappa,\alpha}_{k+1}(s,t) = p(s,t)$. Then the denominator of $d^{\kappa,\alpha}_{k+1}(s,t)$ is bounded by $2^{\mathcal{O}(M)}$.
2. $d^{\kappa,\alpha}_{k+1}(s,t) = \alpha \cdot h[d^{\kappa,\alpha}_{k}](s,a,t)$ for some $a \in \Gamma(s)$. Let $\{\lambda_{u,v}\}_{u,v\in S}, \{y_b\}_{b\in\Gamma(t)}$ be a vertex of $P(s,a,t)$ that reaches the optimal value $h[d^{\kappa,\alpha}_{k}](s,a,t)$. By [Sch86, Theorem 10.2], $\{\lambda_{u,v}\}_{u,v\in S}, \{y_b\}_{b\in\Gamma(t)}$ is of size $\mathcal{O}(M^4)$. Thus the denominator of $d^{\kappa,\alpha}_{k+1}(s,t)$ is a factor of $T(k)\cdot m\cdot 2^{\mathcal{O}(M^6)}$ since $h[d^{\kappa,\alpha}_{k}](s,a,t) = \sum_{u,v\in S} d^{\kappa,\alpha}_{k}(u,v)\cdot\lambda_{u,v}$. Then we can deduce that $T(k+1) \leq T(k)\cdot m\cdot 2^{\mathcal{O}(M^8)}$.
3. $d^{\kappa,\alpha}_{k+1}(s,t) = \alpha \cdot h[d^{\kappa,\alpha}_{k}](t,b,s)$ for some $b \in \Gamma(t)$. This case is symmetrical to the previous one.

Thus $T(k+1) \leq T(k)\cdot m \cdot 2^{\mathcal{O}(M^8)}$ and consequently $T(k) \leq m^k \cdot 2^{\mathcal{O}(kM^8)}$. It follows that the number of bits to represent $d^{\kappa,\alpha}_{k}$ is $\mathcal{O}(kM^{10} + kM^2\log m)$. □

Finally, the last step is to combine the previous two steps.

Theorem 1. *For a fixed $\alpha \in [0,1)\cap\mathbb{Q}$, d^{α}_{κ} can be computed exactly in polynomial time in M.*

Proof. By Lemma 5, we can find in polynomial time in M and $\|\epsilon\|$ a vector that is ϵ-close to d^{α}_{κ}. And by Lemma 4, d^{α}_{κ} is a rational vector of size polynomial in M. So we can use the continued fraction algorithm [Sch86, Section 6] to compute d^{α}_{κ} in polynomial time, as is illustrated in [CvBW12] and [EY10, Page 2540]. □

5 Complexity for the Undiscounted Metric

In this section we prove that the undiscounted metric d^1_{p} can be decided in NP ∩ coNP. More formally, we prove that the problem MDPMetric:

– **Input:** a MDP $(S, \mathcal{V}, [\cdot], Moves, \Gamma, \delta)$, $s_{\mathrm{in}}, t_{\mathrm{in}} \in S$ and a number $\epsilon \in \mathbb{Q}_{\geq 0}$
– **Output:** whether $d^1_{\mathrm{p}}(s_{\mathrm{in}}, t_{\mathrm{in}}) \leq \epsilon$ or not

lies in NP and coNP, where numerical values in $\mathcal{V}$, δ and ϵ are represented in binary. Recall that $d^1_{\mathrm{p}} = d^1_{\mathrm{r}}$ (Corollary 1), so we can work on the lattice $(\mathcal{M}_{\mathrm{r}}, \leq)$ instead of $(\mathcal{M}_{\mathrm{p}}, \leq)$. For convenience we shall abbreviate d^1_{r} as d_{r}.

Our proof method is divided into three steps: First we establish a characterization of the least fixpoint d_{r} called "self-closed" sets; Then we show that whether a given $d \in \mathcal{M}_{\mathrm{r}}$ equals d_{r} is polynomial-time decidable; Finally, we complete the proof by showing how we can guess a premetric $d \in \mathcal{M}_{\mathrm{r}}$ which is also a fixpoint of H^1_{r}.

Below we fix an MDP $(S, \mathcal{V}, [\cdot], Moves, \Gamma, \delta)$. First we introduce the characterization of d_{r}, called "self-closed" sets, as follows:

Definition 7. *Let $d \in \mathcal{M}_{\mathrm{r}}$ satisfying $d = H^1_{\mathrm{r}}(d)$. A subset $X \subseteq S \times S$ is* self-closed *w.r.t d iff for all $(s,t) \in X$, the following conditions hold:*

1. *$d(s,t) > p(s,t)$ (i.e., $d(s,t) \neq p(s,t)$ from Definition 5);*
2. *for all $a \in \Gamma(s)$ such that $d(s,t) = \inf_{y \in \mathcal{D}(t)} d(\delta(s,a), \delta(t,y))$, there is $y^* \in \mathcal{D}(t)$ and $\lambda^* \in \mathrm{OP}[d](\delta(s,a), \delta(t,y^*))$ such that $d(s,t) = d(\delta(s,a), \delta(t,y^*))$ and $\lfloor \lambda^* \rfloor \subseteq X$.*
3. *for all $b \in \Gamma(t)$ such that $d(s,t) = \inf_{x \in \mathcal{D}(s)} d(\delta(t,b), \delta(s,x))$, there is $x^* \in \mathcal{D}(s)$ and $\lambda^* \in \mathrm{OP}[d](\delta(t,b), \delta(s,x^*))$ such that $d(s,t) = d(\delta(t,b), \delta(s,x^*))$ and $\lfloor \lambda^* \rfloor \subseteq X$.*

where $\lfloor \lambda \rfloor := \{(u,v) \in S \times S \mid \lambda(u,v) > 0\}$ for a given λ.

Intuitively, a self-closed set X w.r.t d is a set such that for all $(s,t) \in X$, the value $d(s,t)$ can be reached by some λ with $\lfloor \lambda \rfloor \subseteq X$. This allows us to reduce all $\{d(u,v)\}_{(u,v) \in X}$ simultaneously by a small amount so that d still is a pre-fixpoint of H^1_{r}. Thus if d has a nonempty self-closed set, then d is not the least fixpoint d_{r}. Below we show that nonempty self-closed sets in some sense characterize d_{r}.

Theorem 2. *Let $d \in \mathcal{M}_{\mathrm{r}}$ satisfying $d = H^1_{\mathrm{r}}(d)$. If $d \neq d_{\mathrm{r}}$, then there exists a nonempty self-closed set X w.r.t d.*

Proof. Suppose $d \neq d_{\mathrm{r}}$, we construct a nonempty self-closed set X as described below. Define $\theta(s,t) = d(s,t) - d_{\mathrm{r}}(s,t)$. Then $\theta(s,t) \geq 0$ for all $s,t \in S$, and there is (s,t) such that $\theta(s,t) > 0$. Define X to be the following set:

$$X := \{(s,t) \in S \times S \mid \theta(s,t) = \max\{\theta(u,v) \mid (u,v) \in S \times S\}\}$$

We prove that X is a nonempty self-closed set. The non-emptiness of X is clear. We further prove that all $(s,t) \in X$ satisfy the conditions specified in Definition 7. Note that $\theta(s,t) > 0$ for all $(s,t) \in X$. Consider an arbitrary $(s,t) \in X$:

1. It is clear that $d(s,t) > p(s,t)$, otherwise $d(s,t) = d_{\mathrm{r}}(s,t) = p(s,t)$ by Definition 5 and $\theta(s,t) = 0$. So (s,t) satisfies the first condition in Definition 7.
2. Let $a \in \Gamma(s)$ be a move such that $d(s,t) = \inf_{y \in \mathcal{D}(t)} d(\delta(s,a), \delta(t,y))$. Since d_{r} is a fixpoint of H^1_{r}, $d_{\mathrm{r}}(s,t) \geq \inf_{y \in \mathcal{D}(t)} d_{\mathrm{r}}(\delta(s,a), \delta(t,y))$. Choose a $y^* \in \mathcal{D}(t)$

that reaches the value $\inf_{y\in\mathcal{D}(t)} d_\mathrm{r}(\delta(s,a),\delta(t,y))$. By the definition of X, $\theta(u,v) \le \theta(s,t)$ for all $(u,v)\in S\times S$. Thus for all $\lambda \in \delta(s,a)\otimes\delta(t,y^*)$,

$$\begin{aligned}\textstyle\sum_{u,v\in S} d_\mathrm{r}(u,v)\cdot\lambda(u,v) &\ge \textstyle\sum_{u,v\in S}(d(u,v)-\theta(s,t))\cdot\lambda(u,v)\\ &= (\textstyle\sum_{u,v\in S} d(u,v)\cdot\lambda(u,v)) - \theta(s,t)\end{aligned}$$

The last equality is obtained by $\sum_{u,v\in S}\lambda(u,v)=1$. By taking the infimum at the both sides, we obtain $d(\delta(s,a),\delta(t,y^*)) \le d_\mathrm{r}(\delta(s,a),\delta(t,y^*)) + \theta(s,t)$. Thus, we have:

$$d(s,t) \le d(\delta(s,a),\delta(t,y^*)) \le d_\mathrm{r}(\delta(s,a),\delta(t,y^*)) + \theta(s,t) \le d_\mathrm{r}(s,t)+\theta(s,t)$$

where the last one equals $d(s,t)$. This means that $d(s,t) = d(\delta(s,a),\delta(t,y^*))$ and $d_\mathrm{r}(s,t) = d_\mathrm{r}(\delta(s,a),\delta(t,y^*))$. Let $\lambda^*\in\mathrm{OP}[d_\mathrm{r}](\delta(s,a),\delta(t,y^*))$ be an optimum solution. We prove that $\lambda^*\in\mathrm{OP}[d](\delta(s,a),\delta(t,y^*))$ and $\lfloor\lambda^*\rfloor\subseteq X$. This can be observed as follows:

$$\begin{aligned}&\textstyle\sum_{u,v\in S} d(u,v)\cdot\lambda^*(u,v)\\ &\ge d(s,t) && (\text{by } d(s,t)=d(\delta(s,a),\delta(t,y^*)))\\ &= d_\mathrm{r}(s,t)+\theta(s,t)\\ &= \textstyle\sum_{u,v\in S} d_\mathrm{r}(u,v)\cdot\lambda^*(u,v)+\theta(s,t) && (\text{by } d_\mathrm{r}(s,t)=d_\mathrm{r}(\delta(s,a),\delta(t,y^*)))\\ &= \textstyle\sum_{u,v\in S}(d_\mathrm{r}(u,v)+\theta(s,t))\cdot\lambda^*(u,v)\\ &\ge \textstyle\sum_{u,v\in S} d(u,v)\cdot\lambda^*(u,v)\end{aligned}$$

Then it must be the case that $\lambda^*\in\mathrm{OP}[d](\delta(s,a),\delta(t,y^*))$ and $\theta(u,v)=\theta(s,t)$ whenever $\lambda^*(u,v)>0$. The latter implies $\lfloor\lambda^*\rfloor\subseteq X$. So (s,t) satisfies the second condition in Definition 7.

3. It can be argued symmetrically to the second condition that the third condition is also satisfied.

Hence in conclusion, X is a nonempty self-closed set. □

Theorem 3. *Let $d\in\mathcal{M}_\mathrm{r}$ such that $d=H^1_\mathrm{r}(d)$. If there exists a nonempty self-closed set $X\subseteq S\times S$ w.r.t d, then $d\ne d_\mathrm{r}$.*

Proof. Suppose X is a nonempty self-closed set w.r.t d. We construct a premetric $d'\lneq d$ such that $H^1_\mathrm{r}(d')\le d'$. For each $s,t\in S$, $a\in\Gamma(s)$ and $b\in\Gamma(t)$, define

- $\theta[s,a,t] := d(s,t) - \inf_{y\in\mathcal{D}(t)} d(\delta(s,a),\delta(t,y))$.
- $\theta[s,t,b] := d(s,t) - \inf_{x\in\mathcal{D}(s)} d(\delta(t,b),\delta(s,x))$.

Note that $\theta[s,a,t]$ and $\theta[s,t,b]$ are always non-negative since d is a fixpoint of H^1_r. Further we define

- $\theta_1 := \min\{\theta[s,a,t] \mid (s,t)\in X, a\in\Gamma(s) \text{ and } \theta[s,a,t]>0\}$;
- $\theta_2 := \min\{\theta[s,t,b] \mid (s,t)\in X, b\in\Gamma(t) \text{ and } \theta[s,t,b]>0\}$.
- $\theta_3 := \min\{d(s,t)-p(s,t) \mid (s,t)\in X\}$

where $\min\emptyset := 0$. Finally we define $\theta := \min\{\theta' \mid \theta'\in\{\theta_1,\theta_2,\theta_3\} \text{ and } \theta'>0\}$. Note that $\theta>0$ since $\theta_3>0$. Then we construct $d'\in\mathcal{M}_\mathrm{r}$ by:

$$d'(s,t) := \begin{cases} d(s,t)-\frac{1}{2}\theta & \text{if } (s,t)\in X\\ d(s,t) & \text{if } (s,t)\notin X\end{cases}$$

It is clear that $d' \lneq d$ since X is non-empty and $\theta > 0$. We prove that $H_{\mathrm{r}}^1(d') \le d'$.

Let $(s,t) \in S \times S$. Suppose first that $(s,t) \notin X$. Then by $d' \le d$ we have $H_{\mathrm{r}}^1(d') \le H_{\mathrm{r}}^1(d)$. Thus $H_{\mathrm{r}}^1(d')(s,t) \le d'(s,t)$ by $d'(s,t) = d(s,t)$ and $d(s,t) = H_{\mathrm{r}}^1(d)(s,t)$. Suppose now that $(s,t) \in X$. Consider an arbitrary move $a \in \Gamma(s)$. We clarify two cases below:

(i) $\theta[s,a,t] > 0$. Then $\theta_1 > 0$ and $\theta \le \theta_1 \le \theta[s,a,t]$. So we have

$$d'(s,t) > d(s,t) - \theta[s,a,t] = \inf_{y \in \mathcal{D}(t)} d(\delta(s,a), \delta(t,y)) \ge \inf_{y \in \mathcal{D}(t)} d'(\delta(s,a), \delta(t,y))$$

(ii) $\theta[s,a,t] = 0$. Since X is self-closed, there is $y^* \in \mathcal{D}(t)$ such that $d(s,t) = d(\delta(s,a), \delta(t,y^*))$ and $\lambda^* \in \mathrm{OP}[d](\delta(s,a), \delta(t,y^*))$ such that $\lfloor \lambda^* \rfloor \subseteq X$. By $\lfloor \lambda^* \rfloor \subseteq X$ we obtain

$$\sum_{u,v \in S} d'(u,v) \cdot \lambda^*(u,v) = \sum_{u,v \in S} d(u,v) \cdot \lambda^*(u,v) - \frac{1}{2}\theta = d(\delta(s,a), \delta(t,y^*)) - \frac{1}{2}\theta$$

Then:

$$d'(\delta(s,a), \delta(t,y^*)) \le \sum_{u,v \in S} d'(u,v) \cdot \lambda^*(u,v) = d(\delta(s,a), \delta(t,y^*)) - \frac{1}{2}\theta = d'(s,t)$$

Thus we have $\inf_{y \in \mathcal{D}(t)} d'(\delta(s,a), \delta(t,y)) \le d'(\delta(s,a), \delta(t,y^*)) \le d'(s,t)$. Symmetrically, we can prove that $\inf_{x \in \mathcal{D}(s)} d'(\delta(t,b), \delta(s,x)) \le d'(s,t)$ for all $b \in \Gamma(t)$. Also by the definition of θ, we have $d'(s,t) > p(s,t)$. So we also obtain $H_{\mathrm{r}}^1(d')(s,t) \le d'(s,t)$. Thus $H_{\mathrm{r}}^1(d') \le d'$ and hence $d_{\mathrm{r}} \le d' \lneq d$ by Corollary 1.

□

Note that in the proof d' may not be a pseudometric, especially the triangle inequality may not hold. This is the reason why we need Corollary 1.

Thus for each fixpoint d, $d \neq d_{\mathrm{r}}$ iff there exists a nonempty self-closed set w.r.t d. This characterization means that to check whether $d \neq d_{\mathrm{r}}$, we can equivalently check whether there exists a nonempty self-closed set. The intuition to check the latter is that for self-closed sets X, Y, $X \cup Y$ is still a self-closed set. Thus there exists a largest self-closed set Z. This gives rise to a refinement algorithm that computes Z.

Theorem 4. *Denote by* $FP := \{d \in \mathcal{M}_{\mathrm{r}} \mid d = H_{\mathrm{r}}^1(d) \text{ and } \forall s,t \in S. d(s,t) \in \mathbb{Q}\}$ *the set of rational fixpoints of* H_{r}^1. *The problem whether a given* $d \in FP$ *equals* d_{r} *is decidable in polynomial time.*

Proof. By Theorem 2 and Theorem 3, we can solve the problem by checking whether there exists a nonempty self-closed set w.r.t the given $d \in FP$. Note that for self-closed sets X, Y w.r.t d, $X \cup Y$ is still a self-closed set w.r.t d. So there exists a largest self-closed set w.r.t d, which is denoted by Z. Then $d \neq d_{\mathrm{r}}$ iff Z is nonempty. Below we develop a refinement algorithm to compute Z.

First we define a refining function $ref : \mathcal{E} \mapsto \mathcal{E}$, where the set $\mathcal{E}$ is given by:

$$\mathcal{E} := \{X \subseteq S \times S \mid d(s,t) > p(s,t) \text{ for all } (s,t) \in X\}$$

Note that $\mathcal{E}$ is nonempty since $\emptyset \in \mathcal{E}$. Given $X \in \mathcal{E}$, we define $\theta_X := \min\{d(s,t) - p(s,t) \mid (s,t) \in X\}$ (where $\min \emptyset := 0$), and the premetric $d_X \in \mathcal{M}_{\mathrm{r}}$ as follows:

$$d_X(s,t) = \begin{cases} d(s,t) - \theta_X & \text{if } (s,t) \in X \\ d(s,t) & \text{if } (s,t) \notin X \end{cases}$$

Then $ref(X) \in \mathcal{E}$ is defined as the set of all (s,t) which satisfies the following conditions:

1. $(s,t) \in X$;
2. for all $a \in \Gamma(s)$, if $d(s,t) = \inf_{y\in\mathcal{D}(t)} d(\delta(s,a),\delta(t,y))$ then
$$d_X(s,t) \geq \inf_{y\in\mathcal{D}(t)} d_X(\delta(s,a),\delta(t,y)).$$
3. for all $b \in \Gamma(t)$, if $d(s,t) = \inf_{x\in\mathcal{D}(s)} d(\delta(t,b),\delta(s,x))$ then
$$d_X(s,t) \geq \inf_{x\in\mathcal{D}(s)} d_X(\delta(t,b),\delta(s,x)).$$

Now we construct a sequence $\{Z_i\}_{i\in\mathbb{N}_0}$ by: $Z_0 = \{(s,t) \in S\times S \mid d(s,t) > p(s,t)\}$; $Z_{i+1} = ref(Z_i)$. By $Z_{i+1} \subseteq Z_i$, there exists $n \leq |Z_0| \leq |S|^2$ such that $Z_{n+1} = ref(Z_n) = Z_n$. We show that $Z_n = Z$.

"$Z \subseteq Z_n$": We prove by induction that $Z \subseteq Z_i$ for all $i \in \mathbb{N}_0$. The base step $Z \subseteq Z_0$ is clear from the definition. For inductive step, suppose that $Z \subseteq Z_i$. We show that $Z \subseteq ref(Z_i)$. Consider $(s,t) \in Z$. Suppose $a \in \Gamma(s)$ is a move such that $d(s,t) = \inf_{y\in\mathcal{D}(t)} d(\delta(s,a),\delta(t,y))$. Since Z is self-closed, there is $y^* \in \mathcal{D}(t)$ and $\lambda^* \in \mathrm{OP}[d](\delta(s,a),\delta(t,y^*))$ such that $d(s,t) = d(\delta(s,a),\delta(t,y^*))$ and $\lfloor\lambda^*\rfloor \subseteq Z$. Since $Z \subseteq Z_i$, we have $d_{Z_i}(s,t) = d(s,t) - \theta_{Z_i}$ and $d_{Z_i}(u,v) = d(u,v) - \theta_{Z_i}$ for all $(u,v) \in \lfloor\lambda^*\rfloor$. Thus we obtain

$$\sum_{u,v\in S} d_{Z_i}(u,v)\cdot\lambda^*(u,v) = \sum_{u,v\in S} (d(u,v) - \theta_{Z_i})\cdot\lambda^*(u,v) = d(s,t) - \theta_{Z_i} = d_{Z_i}(s,t)$$

Hence $d_{Z_i}(s,t) \geq \inf_{y\in\mathcal{D}(t)} d_{Z_i}(\delta(s,a),\delta(t,y))$. By a similar reasoning we can prove the opposite direction for all $b \in \Gamma(t)$. So $(s,t) \in Z_{i+1}$. Thus $Z \subseteq Z_{i+1}$.

"$Z_n \subseteq Z$": We prove that Z_n is a self-closed set w.r.t d, i.e., Z_n satisfies the conditions specified in Definition 7. W.l.o.g we can assume that $Z_n \neq \emptyset$. The first condition is directly satisfied since $Z_n \subseteq Z_0$. As for the second condition, consider $(s,t) \in Z_n$ and $a \in \Gamma(s)$ such that $d(s,t) = \inf_{y\in\mathcal{D}(t)} d(\delta(s,a),\delta(t,y))$. By $Z_n = ref(Z_n)$, $d_{Z_n}(s,t) \geq \inf_{y\in\mathcal{D}(t)} d_{Z_n}(\delta(s,a),\delta(t,y))$. Choose $y^* \in \mathcal{D}(t)$ such that $d_{Z_n}(\delta(s,a),\delta(t,y^*)) = \inf_{y\in\mathcal{D}(t)} d_{Z_n}(\delta(s,a),\delta(t,y))$ and an arbitrary $\lambda^* \in \mathrm{OP}[d_{Z_n}](\delta(s,a),\delta(t,y^*))$. Then since $d_{Z_n}(s,t) = d(s,t) - \theta_{Z_n}$, we have

$$\begin{aligned}
d(s,t) &\geq d_{Z_n}(\delta(s,a),\delta(t,y^*)) + \theta_{Z_n}\\
&= \textstyle\sum_{u,v\in S}(d_{Z_n}(u,v) + \theta_{Z_n})\cdot\lambda^*(u,v)\\
&\geq \textstyle\sum_{u,v\in S} d(u,v)\cdot\lambda^*(u,v)\\
&\geq d(\delta(s,a),\delta(t,y^*))\\
&\geq d(s,t) \quad (\text{by } d(s,t) = \inf_{y\in\mathcal{D}(t)} d(\delta(s,a),\delta(t,y)))
\end{aligned}$$

Thus $d(s,t) = d(\delta(s,a),\delta(t,y^*))$, $\lambda^* \in \mathrm{OP}[d](\delta(s,a),\delta(t,y^*))$ and $d_{Z_n}(u,v) = d(u,v) - \theta_{Z_n}$ for all $(u,v) \in \lfloor\lambda^*\rfloor$. The latter implies $\lfloor\lambda^*\rfloor \subseteq Z_n$ since $\theta_{Z_n} > 0$. The justification for the third condition can be carried out in the symmetric way as for the second one.

Thus to compute Z, we need only to apply ref to Z_0 at most $|Z_0|$ times. The computation of ref can be carried out in polynomial time by applying [CdAMR10, Theorem 4.3]. Hence Z is polynomial-time computable. Then we obtain that whether a given $d \in FP$ equals d_{r} is decidable in polynomial time. □

By Theorem 4, we can decide if a given element in FP is d_{r} in polynomial time. Below we present the last step to complete the proof for the membership in $\mathsf{NP} \cap \mathsf{coNP}$, where we illustrate how we can guess an element from FP.

Theorem 5. *The problem* MDPMetric *can be decided in* $\mathsf{NP} \cap \mathsf{coNP}$.

Proof. We only prove the membership of coNP, since the membership of NP is similar. Let $(S, \mathcal{V}, [\cdot], Moves, \Gamma, \delta)$, $(s_{\text{in}}, t_{\text{in}})$ and ϵ be the input. Our strategy to obtain an NP algorithm to decide whether $d_{\text{r}}(s_{\text{in}}, t_{\text{in}}) > \epsilon$ is as follows:

1. We guess a $d \in FP$ by guessing vertices of some polyhedra.
2. We check whether $d = d_{\text{r}}$ by Theorem 4. If $d = d_{\text{r}}$ then we compare $d(s,t)$ and ϵ and return the result, otherwise we abort.

Below we show that how we can guess a $d \in FP$ using polynomial bits. The guessing procedure is illustrated step by step as follows.

1. For each $s, t \in S$ and $a \in \Gamma(s)$, we guess a vertex of $P(s,a,t)$ (cf. Remark 1) as follows:
   - We guess $|S|^2 + |\Gamma(t)|$ constraints (denoted by $Con(s,a,t)$) specified in each polyhedron $P(s,a,t)$.
   - We check (by Gaussian Elimination) whether for all s, a, t, the linear equality system obtained by modifying all comparison operators (e.g., $\geq, \leq$) in $Con(s,a,t)$ to equality (i.e. $=$) has a unique solution. If not, we abort the guessing.
   - We compute the unique solutions. We denote these unique solutions by $\{\lambda(s,a,t)_{u,v}\}_{u,v \in S}$, $\{y(s,a,t)_b\}_{b \in \Gamma(t)}$ for s, a, t.
   - We check if $\{\{\lambda(s,a,t)_{u,v}\}_{u,v \in S}, \{y(s,a,t)_b\}_{b \in \Gamma(t)}\} \in P(s,a,t)$ for all s, a, t. If not, then we abort the guessing.
2. We find in polynomial time [Sch86] a premetric $d \in \mathcal{M}_{\text{r}}$ where $\{d(s,t)\}_{s,t \in S}$ is an (arbitrary) optimum solution of the LP problem with variables $\{d_{s,t}\}_{s,t \in S}$, objective function $\min \sum_{s,t \in S} d_{s,t}$, and the feasible region specified by:
   (a) $d_{s,s} = 0$ and $0 \leq d_{s,t} \leq 1$ for $s, t \in S$
   (b) $d_{s,t} \geq p(s,t)$ for $s, t \in S$
   (c) $d_{s,t} \geq \sum_{u,v \in S} \lambda(s,a,t)_{u,v} \cdot d_{u,v}$ for $(s,t) \in S \times S$ and $a \in \Gamma(s)$.
   (d) $d_{s,t} \geq \sum_{u,v \in S} \lambda(t,b,s)_{u,v} \cdot d_{u,v}$ for $(s,t) \in S \times S$ and $b \in \Gamma(t)$.

   If the linear programming system above has no feasible solution, we abort the guessing. Otherwise we proceed to the next step.
3. We check whether $d \in FP$, which can be done in polynomial time. We abort if the checking is unsuccessful. Otherwise we have guessed a $d \in \text{FP}$.

Then we check whether $d = d_{\text{r}}$ and (if yes) return the comparison result of $d(s,t) > \epsilon$. It is clear that if we can guess a $d \in \text{FP}$ and check successfully that $d = d_{\text{r}}$, then the returned result is correct. Below we prove that d_{r} can be guessed through the guessing procedure.

Consider Step 1. For all $s, t \in S$ and $a \in \Gamma(s)$, we can guess $Con(s,a,t)$ whose unique solution $\{\{\lambda(s,a,t)_{u,v}\}_{u,v \in S}, \{y(s,a,t)_b\}_{b \in \Gamma(t)}\}$ is a vertex of $P(s,a,t)$ that reaches the optimal value $h[d_{\text{r}}](s,a,t)$ of $\text{LP}[d_{\text{r}}](s,a,t)$ (cf. Remark 1) [Sch86, Section 8]. Then in Step 2, we can follow the proof of Lemma 4 to prove that $\{d_{\text{r}}(s,t)\}_{s,t \in S}$ is the unique optimum solution of the LP problem specified in Step 2. Thus d_{r} can be computed polynomially in Step 2. □

6 Conclusion and Related Work

We have shown that for Markov Decision Processes, the discounted game bisimulation metric [dAHM03,CdAMR10] can be computed exactly in polynomial time, and the undiscounted game bisimulation metric [dAMRS07] can be decided in $\mathsf{NP} \cap \mathsf{coNP}$. Our results extend the one for the discounted metric on Labelled Markov Chains [CvBW12], and improves the PSPACE upperbound for the undiscounted metric on Markov Decision Processes [CdAMR10]. It is proved by Chen *et al.* [CvBW12] that the undiscounted metric on Labelled Markov Chains can be decided in polynomial time, however their result cannot be directly applied to Markov Decision Processes. The exact complexity for the undiscounted metric could be of theoretical interest. It is also worth noting that deciding the undiscounted metric on concurrent games is at least as hard as the square-root sum problem [CdAMR10], which is in PSPACE but whose inclusion in NP is a long-standing open problem [EY10].

References

[ASB95] Adnan Aziz, Vigyan Singhal, and Felice Balarin. It usually works: The temporal logic of stochastic systems. In *CAV*, pages 155–165, 1995.

[BEMC00] Christel Baier, Bettina Engelen, and Mila E. Majster-Cederbaum. Deciding bisimilarity and similarity for probabilistic processes. *J. Comput. Syst. Sci.*, 60(1):187–231, 2000.

[CdAMR10] Krishnendu Chatterjee, Luca de Alfaro, Rupak Majumdar, and Vishwanath Raman. Algorithms for game metrics (full version). *Logical Methods in Computer Science*, 6(3), 2010.

[CS02] Stefano Cattani and Roberto Segala. Decision algorithms for probabilistic bisimulation. In *CONCUR*, pages 371–385, 2002.

[CvBW12] Di Chen, Franck van Breugel, and James Worrell. On the complexity of computing probabilistic bisimilarity. In *FoSSaCS*, pages 437–451, 2012.

[dAHM03] Luca de Alfaro, Thomas A. Henzinger, and Rupak Majumdar. Discounting the future in systems theory. In *ICALP*, pages 1022–1037, 2003.

[dAM04] Luca de Alfaro and Rupak Majumdar. Quantitative solution of omega-regular games. *J. Comput. Syst. Sci.*, 68(2):374–397, 2004.

[dAMRS07] Luca de Alfaro, Rupak Majumdar, Vishwanath Raman, and Mariëlle Stoelinga. Game relations and metrics. In *LICS*, pages 99–108, 2007.

[DJGP02] Josee Desharnais, Radha Jagadeesan, Vineet Gupta, and Prakash Panangaden. The metric analogue of weak bisimulation for probabilistic processes. In *LICS*, pages 413–422, 2002.

[DLT08] Josée Desharnais, François Laviolette, and Mathieu Tracol. Approximate analysis of probabilistic processes: Logic, simulation and games. In *QEST*, pages 264–273, 2008.

[EY10] Kousha Etessami and Mihalis Yannakakis. On the complexity of Nash equilibria and other fixed points. *SIAM J. Comput.*, 39(6):2531–2597, 2010.

[FPP04] Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite Markov decision processes. In *AAAI*, pages 950–951, 2004.

[GJP04] Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Approximate reasoning for real-time probabilistic processes. In *QEST*, pages 304–313, 2004.

[GJS90] Alessandro Giacalone, Chi-Chang Jou, and Scott A. Smolka. Algebraic reasoning for probabilistic concurrent systems. In *Proc. IFIP TC2 Working Conference on Programming Concepts and Methods*, pages 443–458. North-Holland, 1990.

[JGP06] Agung A. Julius, Antoine Girard, and George J. Pappas. Approximate bisimulation for a class of stochastic hybrid systems. In *American Control Conference*, pages 4724–4729, Portland, Etats-Unis, June 2006. IEEE.

[JL91] Bengt Jonsson and Kim Guldstrand Larsen. Specification and refinement of probabilistic processes. In *LICS*, pages 266–277, 1991.

[LS91] Kim Guldstrand Larsen and Arne Skou. Bisimulation through probabilistic testing. *Inf. Comput.*, 94(1):1–28, 1991.

[Mil89] R. Milner. *Communication and concurrency.* Prentice-Hall, Inc., 1989.

[Pan09] Prakash Panangaden. *Labelled Markov Processes.* Imperial College Press, 2009.

[Sch86] Alexander Schrijver. *Theory of Linear and Integer Programming.* John Wiley & Sons, Inc., 1986.

[SL95] Roberto Segala and Nancy A. Lynch. Probabilistic simulations for probabilistic processes. *Nord. J. Comput.*, 2(2):250–273, 1995.

[TDZ11] Mathieu Tracol, Josée Desharnais, and Abir Zhioua. Computing distances between probabilistic automata. In *QAPL*, pages 148–162, 2011.

[vBSW08] Franck van Breugel, Babita Sharma, and James Worrell. Approximating a behavioural pseudometric without discount for probabilistic systems. *Logical Methods in Computer Science*, 4(2), 2008.

[vBW01a] Franck van Breugel and James Worrell. An algorithm for quantitative verification of probabilistic transition systems. In *CONCUR*, pages 336–350, 2001.

[vBW01b] Franck van Breugel and James Worrell. Towards quantitative verification of probabilistic transition systems. In *ICALP*, pages 421–432, 2001.

Aachener Informatik-Berichte

This list contains all technical reports published during the past three years. A complete list of reports dating back to 1987 is available from:

http://aib.informatik.rwth-aachen.de/

To obtain copies please consult the above URL or send your request to:

Informatik-Bibliothek, RWTH Aachen, Ahornstr. 55, 52056 Aachen,
Email: `biblio@informatik.rwth-aachen.de`

2009-01 * Fachgruppe Informatik: Jahresbericht 2009
2009-02 Taolue Chen, Tingting Han, Joost-Pieter Katoen, Alexandru Mereacre: Quantitative Model Checking of Continuous-Time Markov Chains Against Timed Automata Specifications
2009-03 Alexander Nyßen: Model-Based Construction of Embedded Real-Time Software - A Methodology for Small Devices
2009-05 George B. Mertzios, Ignasi Sau, Shmuel Zaks: A New Intersection Model and Improved Algorithms for Tolerance Graphs
2009-06 George B. Mertzios, Ignasi Sau, Shmuel Zaks: The Recognition of Tolerance and Bounded Tolerance Graphs is NP-complete
2009-07 Joachim Kneis, Alexander Langer, Peter Rossmanith: Derandomizing Non-uniform Color-Coding I
2009-08 Joachim Kneis, Alexander Langer: Satellites and Mirrors for Solving Independent Set on Sparse Graphs
2009-09 Michael Nett: Implementation of an Automated Proof for an Algorithm Solving the Maximum Independent Set Problem
2009-10 Felix Reidl, Fernando Sánchez Villaamil: Automatic Verification of the Correctness of the Upper Bound of a Maximum Independent Set Algorithm
2009-11 Kyriaki Ioannidou, George B. Mertzios, Stavros D. Nikolopoulos: The Longest Path Problem is Polynomial on Interval Graphs
2009-12 Martin Neuhäußer, Lijun Zhang: Time-Bounded Reachability in Continuous-Time Markov Decision Processes
2009-13 Martin Zimmermann: Time-optimal Winning Strategies for Poset Games
2009-14 Ralf Huuck, Gerwin Klein, Bastian Schlich (eds.): Doctoral Symposium on Systems Software Verification (DS SSV'09)
2009-15 Joost-Pieter Katoen, Daniel Klink, Martin Neuhäußer: Compositional Abstraction for Stochastic Systems
2009-16 George B. Mertzios, Derek G. Corneil: Vertex Splitting and the Recognition of Trapezoid Graphs
2009-17 Carsten Kern: Learning Communicating and Nondeterministic Automata
2009-18 Paul Hänsch, Michaela Slaats, Wolfgang Thomas: Parametrized Regular Infinite Games and Higher-Order Pushdown Strategies
2010-01 * Fachgruppe Informatik: Jahresbericht 2010
2010-02 Daniel Neider, Christof Löding: Learning Visibly One-Counter Automata in Polynomial Time

2010-03 Holger Krahn: MontiCore: Agile Entwicklung von domänenspezifischen Sprachen im Software-Engineering

2010-04 René Wörzberger: Management dynamischer Geschäftsprozesse auf Basis statischer Prozessmanagementsysteme

2010-05 Daniel Retkowitz: Softwareunterstützung für adaptive eHome-Systeme

2010-06 Taolue Chen, Tingting Han, Joost-Pieter Katoen, Alexandru Mereacre: Computing maximum reachability probabilities in Markovian timed automata

2010-07 George B. Mertzios: A New Intersection Model for Multitolerance Graphs, Hierarchy, and Efficient Algorithms

2010-08 Carsten Otto, Marc Brockschmidt, Christian von Essen, Jürgen Giesl: Automated Termination Analysis of Java Bytecode by Term Rewriting

2010-09 George B. Mertzios, Shmuel Zaks: The Structure of the Intersection of Tolerance and Cocomparability Graphs

2010-10 Peter Schneider-Kamp, Jürgen Giesl, Thomas Ströder, Alexander Serebrenik, René Thiemann: Automated Termination Analysis for Logic Programs with Cut

2010-11 Martin Zimmermann: Parametric LTL Games

2010-12 Thomas Ströder, Peter Schneider-Kamp, Jürgen Giesl: Dependency Triples for Improving Termination Analysis of Logic Programs with Cut

2010-13 Ashraf Armoush: Design Patterns for Safety-Critical Embedded Systems

2010-14 Michael Codish, Carsten Fuhs, Jürgen Giesl, Peter Schneider-Kamp: Lazy Abstraction for Size-Change Termination

2010-15 Marc Brockschmidt, Carsten Otto, Christian von Essen, Jürgen Giesl: Termination Graphs for Java Bytecode

2010-16 Christian Berger: Automating Acceptance Tests for Sensor- and Actuator-based Systems on the Example of Autonomous Vehicles

2010-17 Hans Grönniger: Systemmodell-basierte Definition objektbasierter Modellierungssprachen mit semantischen Variationspunkten

2010-18 Ibrahim Armaç: Personalisierte eHomes: Mobilität, Privatsphäre und Sicherheit

2010-19 Felix Reidl: Experimental Evaluation of an Independent Set Algorithm

2010-20 Wladimir Fridman, Christof Löding, Martin Zimmermann: Degrees of Lookahead in Context-free Infinite Games

2011-01 * Fachgruppe Informatik: Jahresbericht 2011

2011-02 Marc Brockschmidt, Carsten Otto, Jürgen Giesl: Modular Termination Proofs of Recursive Java Bytecode Programs by Term Rewriting

2011-03 Lars Noschinski, Fabian Emmes, Jürgen Giesl: A Dependency Pair Framework for Innermost Complexity Analysis of Term Rewrite Systems

2011-04 Christina Jansen, Jonathan Heinen, Joost-Pieter Katoen, Thomas Noll: A Local Greibach Normal Form for Hyperedge Replacement Grammars

2011-07 Shahar Maoz, Jan Oliver Ringert, Bernhard Rumpe: An Operational Semantics for Activity Diagrams using SMV

2011-08 Thomas Ströder, Fabian Emmes, Peter Schneider-Kamp, Jürgen Giesl, Carsten Fuhs: A Linear Operational Semantics for Termination and Complexity Analysis of ISO Prolog

2011-09 Markus Beckers, Johannes Lotz, Viktor Mosenkis, Uwe Naumann (Editors): Fifth SIAM Workshop on Combinatorial Scientific Computing

2011-10 Markus Beckers, Viktor Mosenkis, Michael Maier, Uwe Naumann: Adjoint Subgradient Calculation for McCormick Relaxations

2011-11 Nils Jansen, Erika Ábrahám, Jens Katelaan, Ralf Wimmer, Joost-Pieter Katoen, Bernd Becker: Hierarchical Counterexamples for Discrete-Time Markov Chains

2011-12 Ingo Felscher, Wolfgang Thomas: On Compositional Failure Detection in Structured Transition Systems

2011-13 Michael Förster, Uwe Naumann, Jean Utke: Toward Adjoint OpenMP

2011-14 Daniel Neider, Roman Rabinovich, Martin Zimmermann: Solving Muller Games via Safety Games

2011-16 Niloofar Safiran, Uwe Naumann: Toward Adjoint OpenFOAM

2011-18 Kamal Barakat: Introducing Timers to pi-Calculus

2011-19 Marc Brockschmidt, Thomas Ströder, Carsten Otto, Jürgen Giesl: Automated Detection of Non-Termination and NullPointerExceptions for Java Bytecode

2011-24 Callum Corbett, Uwe Naumann, Alexander Mitsos: Demonstration of a Branch-and-Bound Algorithm for Global Optimization using McCormick Relaxations

2011-25 Callum Corbett, Michael Maier, Markus Beckers, Uwe Naumann, Amin Ghobeity, Alexander Mitsos: Compiler-Generated Subgradient Code for McCormick Relaxations

2011-26 Hongfei Fu: The Complexity of Deciding a Behavioural Pseudometric on Probabilistic Automata

2012-01 * Fachgruppe Informatik: Annual Report 2012

2012-02 Thomas Heer: Controlling Development Processes

2012-03 Arne Haber, Jan Oliver Ringert, Bernhard Rumpe: MontiArc - Architectural Modeling of Interactive Distributed and Cyber-Physical Systems

2012-04 Marcus Gelderie: Strategy Machines and their Complexity

2012-05 Thomas Ströder, Fabian Emmes, Jürgen Giesl, Peter Schneider-Kamp, and Carsten Fuhs: Automated Complexity Analysis for Prolog by Term Rewriting

2012-06 Marc Brockschmidt, Richard Musiol, Carsten Otto, Jürgen Giesl: Automated Termination Proofs for Java Programs with Cyclic Data

* These reports are only available as a printed version.

Please contact `biblio@informatik.rwth-aachen.de` to obtain copies.