Information and Computation 247 (2016) 106–129

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

Efficient approximation of optimal control for continuous-time Markov games ☆

John Fearnley [a], Markus N. Rabe [b], Sven Schewe [a], Lijun Zhang [c,*]

[a] Department of Computer Science, University of Liverpool, Liverpool, United Kingdom
[b] Electronical Engineering and Computer Science, University of California, Berkeley, USA
[c] State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, China

ARTICLE INFO

Article history:
Received 27 November 2013
Received in revised form 1 December 2014
Available online 30 December 2015

Keywords:
Continuous time Markov decision processes and games
Optimal control
Discretisation

ABSTRACT

We study the time-bounded reachability problem for continuous-time Markov decision processes (CTMDPs) and games (CTMGs). Existing techniques for this problem use discretisation techniques to partition time into discrete intervals of size ε, and optimal control is approximated for each interval separately. Current techniques provide an accuracy of $O(\varepsilon^2)$ on each interval, which leads to an infeasibly large number of intervals. We propose a sequence of approximations that achieve accuracies of $O(\varepsilon^3)$, $O(\varepsilon^4)$, and $O(\varepsilon^5)$, that allow us to drastically reduce the number of intervals that are considered. For CTMDPs, the performance of the resulting algorithms is comparable to the heuristic approach given by Buchholz and Schulz, while also being theoretically justified. All of our results generalise to CTMGs, where our results yield the first practically implementable algorithms for this problem. We also provide memoryless strategies for both players that achieve similar error bounds.

1. Introduction

Markov decision processes (MDPs) and continuous-time Markov decision processes (CTMDPs) are powerful stochastic models which have applications in many areas including automated planning, operations research, and decision support systems [3–6]. Over the past 15 years, probabilistic models are being used extensively in the formal analysis of complex systems, including networked, distributed, and most recently, biological systems. Probabilistic model checking for discrete-time MDPs and continuous-time Markov chains (CTMCs) has been successfully applied to these rich academic and industrial applications [7–10]. However, for continuous-time Markov decision processes (CTMDPs), which mix the nondeterminism of MDPs with the continuous-time setting of CTMCs [4], and continuous-time Markov games (CTMGs), which combine both helpful and hostile nondeterminism, practical approaches are less well developed.

This article studies the *time-bounded reachability* problem for CTMDPs and CTMGs, which is of paramount importance for model checking [11]. The time-bounded reachability problem is to determine, for a given set of goal locations G and time bound T, the best way to resolve the nondeterminism in order to maximise (or minimise) the probability of reaching G before the deadline T.

☆ An extended abstract appeared in FSTTCS 2011 [2].

* Corresponding author.
E-mail addresses: john.fearnley@liverpool.ac.uk (J. Fearnley), rabe@berkeley.edu (M.N. Rabe), sven.schewe@liverpool.ac.uk (S. Schewe), zhanglj@ios.ac.cn (L. Zhang).

http://dx.doi.org/10.1016/j.ic.2015.12.002
0890-5401/

Table 1
The number of intervals needed by our algorithms for precisions 10^{-7}, 10^{-9}, and 10^{-11}.

Technique	$\pi = 10^{-7}$	$\pi = 10^{-9}$	$\pi = 10^{-11}$
Current techniques	1,000,000,000	100,000,000,000	10,000,000,000,000
Double ε-nets	81,650	816,497	8,164,966
Triple ε-nets	3219	14,939	69,337
Quadruple ε-nets	605	1911	6043

For practical concerns it is often sufficient to closely approximate the time-bounded reachability. For CTMCs the approximation problem can be solved efficiently by *uniformisation* or by standard numerical approaches like Runge–Kutta, but both methods are not applicable in the presence of nondeterministic choices. Fourth order Runge–Kutta requires that the target function can be continuously differentiated four times, but at the points in time where the nondeterministic choice changes, the target function of CTMDPs and CTMGs can be differentiated only once.

The entity that resolves the nondeterminism in a CTMDP or a CTMG is called a *scheduler* (or *strategy*). The different classes of schedulers are contrasted by Neuhäußer et. al. [12], and they show that *late schedulers* are the most powerful class. Also it is possible to transfer results for late schedulers to early schedulers using a model transformation [13]. Several algorithms have been given to approximate the time-bounded reachability probabilities of CTMDPs for late schedulers [1, 14–16].

State-of-the-art techniques for solving this problem are based on different forms of *discretisation* [1]. This technique splits the time bound T into small intervals of length ε. Optimal control is approximated for each interval separately, and these approximations are combined to produce the final result. Current techniques can approximate optimal control on an interval of length ε with an accuracy of $O(\varepsilon^2)$. However, to achieve a precision of π with these techniques, one must choose $\varepsilon \approx \pi/T$, which leads to $O(T^2/\pi)$ many intervals. Since the desired precision is often high (it is common to require that $\pi \leq 10^{-6}$), this leads to an infeasibly large number of intervals that must be considered by the algorithms.

Our contribution In this article we present a method of obtaining larger interval sizes that satisfies both theoretical and practical concerns. Our approach is to provide more precise approximations for each ε length interval. While current techniques provide an accuracy of $O(\varepsilon^2)$, we propose a sequence of approximations, called double ε-nets, triple ε-nets, and quadruple ε-nets, with accuracies $O(\varepsilon^3)$, $O(\varepsilon^4)$, and $O(\varepsilon^5)$, respectively. Since these approximations are much more precise on each interval, they allow us to consider far fewer intervals while still maintaining high precision. For example, Table 1 gives the number of intervals considered by our algorithms, in the worst case, for a normed CTMDP with time bound $T = 10$.

Of course, in order to become more precise, we must spend additional computational effort. However, the cost of using double ε-nets instead of using current techniques requires only an extra factor of $\log|\Sigma|$, where Σ is the set of actions. Thus, in almost all cases, the large reduction in the number of intervals far outweighs the extra cost of using double ε-nets. Our worst case running times for triple and quadruple ε-nets are not so attractive: triple ε-nets require an extra $|L| \cdot |\Sigma|^2$ factor over double ε-nets, where L is the set of locations, and quadruple ε-nets require yet another $|L| \cdot |\Sigma|^2$ factor over triple ε-nets. However, these worst case running times only occur when the choice of optimal action changes frequently, and we speculate that the cost of using these algorithms in practice is much lower than our theoretical worst case bounds. Our experimental results with triple ε-nets support this claim.

Organisation of the article We first discuss related work in Section 2. Then, we recall the model Markov games and notations needed in Section 3. We present our main result in Section 4. We conclude the article in Section 5 by providing experimental supports of our results.

2. Related work

CTMDPs have been extensively studied in the control community. The analysis there has been focused on optimising expected reward [17,6,5,3,18]. Various techniques, including discretisation as well as value and strategy iteration, have been exploited for the analysis.

Baier *et al.* [19] have first studied the model checking problem for CTMDPs, in which they provide an algorithm that computes time-bounded reachability probabilities in globally uniform CTMDPs. Their approach refers only to the class of *time-abstract schedulers*, which are strictly less powerful than the schedulers we consider in this work. Although such schedulers do have access to the sequence of states that have been visited, they do not have access to the time. Time-abstract schedulers have been used for analysing various academic case studies, see [20,21]. The existence of optimal time-abstract schedulers for arbitrary CTMDPs and their game extensions is studied, independently, in [22,23] and [24].

It has already been pointed out in [19] that the time-independent class of schedulers is strictly less powerful than the time-dependent class. Later, in [12,25], notions of early and late time-dependent schedulers are introduced for locally uniform CTMDPs. Early schedulers make their decision upon entering a state, whereas late schedulers may wait until the sojourn time expires and then choose the next action. Their result shows that late time-dependent schedulers are the most powerful class of schedulers. The notion of late schedulers is generalised to arbitrary CTMDPs in [13].

The standard discretisation technique has then been exploited in [15,16] for computing the maximal probabilistic reachability under early and late time-dependent schedulers. The number of steps in the discretisation based approach is high: it is reciprocal in the required precision π, and quadratic in λT. Here λ denotes the uniformisation rate of the model, and T denotes the time bound. Slight improvement of the bound is reported in [14,26]. The discretisation approach has been extended to more powerful models. In [27,28], a compositional framework has been developed for models with continuous-time, probabilistic and nondeterministic choices. The model is coined as *Markov Automata* [27,28], which can be considered as an extension of CTMDPs and a related model *interactive Markov chains* [29]. In [30,31], the discretisation technique is extended to analyse Markov automata.

A recent article of Buchholz and Schulz [1] has addressed this problem for practical applications, by allowing the interval sizes to vary. In addition to computing an approximation of the maximal time-bounded reachability probability, which provides a lower bound on the optimum, they also compute an upper bound. When the upper and lower bounds do not diverge too much, the interval can be extended indefinitely. In applications, where the optimal choice of action changes infrequently, this idea allows their algorithm to consider far fewer intervals while still maintaining high precision. However, from a theoretical perspective, their algorithm is not particularly satisfying. Their method for extending interval lengths depends on a heuristic, and in the worst case their algorithm may consider $O(T^2/\pi)$ intervals, which is not better than other discretisation based techniques.

Further, an added advantage of our techniques is that they can be applied to continuous-time Markov games as well as to CTMDPs, whereas Buchholz and Schulz restrict their analysis to CTMDPs. Moreover, previous work on CTMGs has mostly been restricted to simplified settings, such as the time-abstract setting. Therefore, to the best of our knowledge, we present the first practically implementable approximation algorithms for the time-dependent time-bounded reachability problem in CTMGs. Each of our approximations also provides memoryless strategies, i.e., strategies only depending on the states, for both players that achieve similar error bounds.

For a thorough comparative evaluation of the different approximation methods for CTMDPs, we refer to the very recent study by Butkova et al. [32].

Finally, we discuss how our approach is related to numerical methods. In the numerical evaluations of CTMCs, numerical methods like collocation techniques (like the Runge–Kutta method) play an important role. In the CTMDP setting, these methods cannot realise the precision they can realise in CTMCs, because the functor describing the dynamics in the Bellman equation is not smooth enough: it is not even differentiable. We discuss the impact in Appendix A. A simple illustrating example is given in Appendix A. Our approach is an application of Picard's iteration [33], which uses the traditional Newton approximation for the Bellman equations [4] as a starting point.

Using Picard's iteration overcomes the numerical problems attached to approaches used for Markov chains, such as Runge–Kutta methods. As opposed to these techniques, we do not require the high degree of smoothness, which is not present in the Bellman equation due to min and max operators.

We derive guarantees for the precision of our techniques for all ε-nets and discuss the generation of near optimal schedulers as witnesses or counter examples. We have chosen to use memoryful schedulers for this. This may be counter intuitive at first glance, as the memoryless schedulers obtained by 'staying' on the same level of the ε-net would be simpler, while offering the same order of precision. However, we failed to establish similar constant factors for such memoryless schedulers.

3. Preliminaries

In this section we recall the model Markov games and CTMDPs. We recall the notion of schedulers, strategies and the characterisation of optimal time-bounded reachability in the game setting. To simplify the technical development, we then argue that it is sufficient to study the normed Markov games, i.e., games with the same uniformisation rate.

3.1. Markov games

For a finite set L, a distribution $\nu : L \to [0,1]$ over L is a function satisfying $\sum_{l \in L} \nu(l) = 1$. Below we denote $\mathit{Dist}(L)$ the set of distributions over L.

Definition 1. A continuous-time Markov game (or simply Markov game) is a tuple $(L, L_r, L_s, \Sigma, \mathbf{R}, \nu)$, consisting of a finite set L of locations, which is partitioned into locations L_r (controlled by a *reachability* player) and L_s (controlled by a *safety* player), a finite set Σ of actions, a rate matrix $\mathbf{R} : (L \times \Sigma \times L) \to \mathbb{Q}_{\geqslant 0}$, and an initial distribution $\nu \in \mathit{Dist}(L)$.

We require that the following side-conditions hold: For all locations $l \in L$, there must be an action $a \in \Sigma$ such that $\mathbf{R}(l,a,L) := \sum_{l' \in L} \mathbf{R}(l,a,l') > 0$, which we call *enabled*. We denote the set of enabled actions in l by $\Sigma(l)$. We define the *size* $|\mathcal{M}|$ of a Markov game as the number of non-zero rates in the rate matrix $\mathbf{R}$.

A Markov game is called *uniform* with uniformisation rate λ, if $\mathbf{R}(l,a,L) = \lambda$ holds for all locations l and enabled actions $a \in \Sigma(l)$. We further call a Markov game *normed*, if its uniformisation rate is 1. The semantics of Markov games is given by pure stochastic processes obtained after resolving the nondeterministic choices using strategies; the details are given in the following subsection.

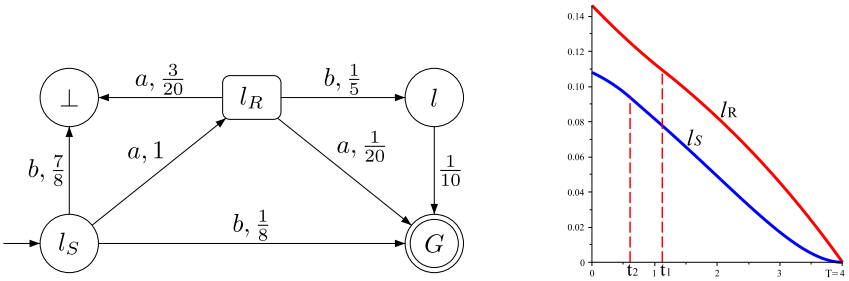

Fig. 1. Left: a normed Markov game. Right: the function f within $[0, 4]$ for l_R and l_S.

As a running example, we will use the normed Markov game shown in the left half of Fig. 1. Locations belonging to the safety player are drawn as circles, and locations belonging to the reachability player are drawn as rectangles. The self-loops of the normed Markov game are not drawn, but rates assigned to the self-loops can be derived from the other rates: for example, we have $\mathbf{R}(l_R, a, l_R) = 0.8$. The locations G and $\bot$ have only a single enabled action leading to itself, which is omitted in the drawing, and there is only a single enabled action for l. It therefore does not matter which player owns l, G, and $\bot$.

3.2. Schedulers and strategies

We consider Markov games in a time interval $[0, T]$ with $T \in \mathbb{R}_{\geq 0}$. The nondeterminism in the system needs to be resolved by a pair of strategies for the two players which together form a *scheduler* for the whole system. We use $Paths_r$ and $Paths_s$ to denote the sets of finite paths $l_0 \xrightarrow{a_0, t_0} l_1 \ldots \xrightarrow{a_{n-1}, t_{n-1}} l_n$ ending with location $l_n \in L_r$ and $l_n \in L_s$, respectively. Formally, a strategy is a function in $Paths_{r/s} \times [0, T] \to \Sigma$. We use $\mathcal{S}_r$ and $\mathcal{S}_s$ to denote the strategies of reachability player and the strategies of safety player, respectively. We use Π_r and Π_s to denote the set of all strategies for the reachability and safety players, respectively, and we use Π to denote $\Pi_r \cup \Pi_s$. (For technical reasons one has to restrict the schedulers to those which are measurable. This restriction, however, is of no practical relevance. In particular, simple piecewise constant timed-positional strategies $L \times [0, T] \to \Sigma$ suffice for optimal scheduling [13,15,4].)

If we fix a pair $\mathcal{S}_r, \mathcal{S}_s$ of strategies for the reachability player and the safety player, respectively, we obtain a scheduler $\mathcal{S}_{r+s}$ that resolves all nondeterministic choices in the Markov game. This results in a deterministic stochastic process $\mathcal{M}_{\mathcal{S}_{r+s}}$, which can be seen as a time inhomogeneous Markov chain. For $t \leq T$, we use $Pr_{\mathcal{S}_{r+s}}(t)$ to denote the transient distribution at time t over S under the scheduler $\mathcal{S}_{r+s}$.

Given a Markov game $\mathcal{M}$, a goal region $G \subseteq L$, and a time bound $T \in \mathbb{R}_{\geq 0}$, we are interested in the *optimal* probability of being in a goal state at time T (and the corresponding pair of optimal strategies). This is given by:

$$\sup_{\mathcal{S}_r \in \Pi_r} \inf_{\mathcal{S}_s \in \Pi_s} \sum_{l \in G} Pr_{\mathcal{S}_{r+s}}(l, T),$$

where $Pr_{\mathcal{S}_{r+s}}(l, T) := Pr_{\mathcal{S}_{r+s}}(T)(l)$. It is commonly referred to as the *maximum* time-bounded reachability probability problem in the case of CTMDPs with a reachability player only. For $t \leq T$, we define $f : L \times \mathbb{R}_{\geq 0} \to [0, 1]$, to be the optimal probability to be in the goal region at the time bound T, assuming that we start in location l and that t time units have passed already. By definition, it holds then that $f(l, T) = 1$ if $l \in G$ and $f(l, T) = 0$ if $l \notin G$. Optimising the vector of values $f(\cdot, 0)$ then yields the optimal value and its optimal strategy.

Let us return to the example shown in Fig. 1. The right half of the figure shows the optimal reachability probabilities, as given by f, for the locations l_R and l_S when the time bound $T = 4$. The points $t_1 \approx 1.123$ and $t_2 \approx 0.609$ represent the times at which the optimal strategies change their decisions. Before t_1 it is optimal for the reachability player to use action b at l_R, but afterwards the optimal choice is action a. Similarly, the safety player uses action b before t_2, and switches to a afterwards.

3.3. Characterisation of f

We define a matrix $\mathbf{Q}$ such that $\mathbf{Q}(l, a, l') = \mathbf{R}(l, a, l')$ if $l' \neq l$ and $\mathbf{Q}(l, a, l) = -\sum_{l' \neq l} \mathbf{R}(l, a, l')$. The optimal function f can be characterised as a set of differential equations [4], see also [17,6]. For each $l \in L$ we define $f(l, T) = 1$ if $l \in G$, and 0 if $l \notin G$. Otherwise, for $t < T$, we define:

$$-\dot{f}(l, t) = \operatorname*{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot f(l', t), \tag{1}$$

where $\mathrm{opt} \in \{\max, \min\}$ is max for reachability player locations and min for safety player locations. We will use the opt-notation throughout this article.

Using the matrix $\mathbf{R}$, Equation (1) can be rewritten to:

$$-\dot{f}(l,t) = \operatorname*{opt}_{a\in\Sigma(l)} \sum_{l'\in L} \mathbf{R}(l,a,l') \cdot \big(f(l',t) - f(l,t)\big). \tag{2}$$

For uniform Markov games, we simply have $\mathbf{Q}(l,a,l) = \mathbf{R}(l,a,l) - \lambda$, with $\lambda = 1$ for normed Markov games. This also provides an intuition for the fact that uniformisation does not alter the reachability probability: the rate $\mathbf{R}(l,a,l)$ does not appear in (1).

3.4. Uniformisation

We will present our results for normed Markov games only. However, we argue that our algorithms for normed Markov games can be applied to solve Markov games that are not normed.

We first show how our algorithms can be used to solve uniform Markov games, and then argue that this is sufficient to solve general Markov games. In order to solve uniform Markov games with arbitrary uniformisation rate λ, we will define a corresponding normed Markov game in which time has been compressed by a factor of λ. More precisely, for each Markov game $\mathcal{M} = (L, L_r, L_s, \Sigma, \mathbf{R}, \nu)$ with uniform transition rate $\lambda > 0$, we define $\mathcal{M}^{\|\cdot\|} = (L, L_r, L_s, \Sigma, \mathbf{P}, \nu)$ where $\mathbf{P} = \frac{1}{\lambda}\mathbf{R}$, which is the Markov game that differs from $\mathcal{M}$ only in the rate matrix. The following lemma allows us to translate solutions of $\mathcal{M}^{\|\cdot\|}$ to $\mathcal{M}$.

Lemma 2. *For every uniform Markov game $\mathcal{M}$, an approximation of some precision π of the optimal time-bounded reachability probabilities and strategies in $\mathcal{M}^{\|\cdot\|}$ for the time bound T is also an approximation of precision π of the optimal time-bounded reachability probabilities and strategies in $\mathcal{M}$ for the time bound $\frac{T}{\lambda}$.*

Proof. To prove this claim, we define the bijection $b : \Pi[\mathcal{M}^{\|\cdot\|}] \to \Pi[\mathcal{M}]$ between schedulers of $\mathcal{M}^{\|\cdot\|}$ and $\mathcal{M}$ that maps each scheduler $\mathcal{S} \in \Pi[\mathcal{M}^{\|\cdot\|}]$ to a scheduler $\mathcal{S}' \in \Pi[\mathcal{M}]$ with $\mathcal{S}'(l,t) = \mathcal{S}(l,\lambda t)$ for all $t \in [0,T]$. In other words, we map each scheduler of $\mathcal{M}^{\|\cdot\|}$ to a scheduler of $\mathcal{M}$ in which time has been stretched by a factor of λ. It is easy to see that the time-bounded reachability probability for time bound T in $\mathcal{M}$ under $\mathcal{S}' = b(\mathcal{S})$ is equivalent to the time-bounded reachability probability for time bound λT for $\mathcal{M}^{\|\cdot\|}$ under $\mathcal{S}$. This bijection therefore proves that the optimal time-bounded reachability probabilities are the same in both games, and it also provides a procedure for translating approximately optimal strategies of the game $\mathcal{M}^{\|\cdot\|}$ to the game $\mathcal{M}$. Since the optimal reachability probabilities are the same in both games, an approximation of the optimal reachability probability in $\mathcal{M}^{\|\cdot\|}$ with precision π must also be an approximation of the optimal reachability probability in $\mathcal{M}^{\|\cdot\|}$ with precision π. □

In order to solve general Markov games we can first uniformise them, and then apply Lemma 2. If $\mathcal{M} = (L, L_r, L_s, \Sigma, \mathbf{R}, \nu)$ is a continuous-time Markov game, then we define the uniformisation of $\mathcal{M}$ as $\text{unif}(\mathcal{M}) = (L, L_r, L_s, \Sigma, \mathbf{R}', \nu)$, where $\mathbf{R}'$ is defined as follows. If $\lambda = \max_{l\in L} \max_{a\in\Sigma(l)} \mathbf{R}(l,a,L\setminus\{l\})$, then we define, for every pair of locations $l, l' \in L$, and every action $a \in \Sigma(l)$:

$$\mathbf{R}'(l,a,l') = \begin{cases} \mathbf{R}(l,a,l') & \text{if } l \neq l', \\ \lambda - \mathbf{R}(l,a,L) & \text{if } l = l'. \end{cases}$$

Previous work has noted that, for the class of late schedulers, the optimal time-bounded reachability probabilities and schedulers in $\mathcal{M}$ are identical to the optimal time-bounded reachability probabilities and schedulers in $\text{unif}(\mathcal{M})$ [13]. To see why, note that Equation (2) does not refer to the entry $\mathbf{R}'(l,a,l)$, and therefore the modifications made to the rate matrix by uniformisation can have no effect on the choice of optimal action.

Lemma 3. *(See [13].) For every continuous-time Markov game $\mathcal{M}$, the optimal time-bounded reachability probabilities and schedulers of $\mathcal{M}$ are identical to the optimal time-bounded reachability probabilities and schedulers of* $\text{unif}(\mathcal{M})$.

Therefore, we have shown the following lemma, which states that our algorithms for normed Markov games can also be applied to arbitrary Markov games.

Lemma 4. *We can adapt an $O(f(\mathcal{M}))$ time algorithm for normed Markov games to solve an arbitrary Markov game in time $O(f(\mathcal{M}) \cdot \lambda)$, where λ is the uniformisation rate of $\mathcal{M}$.*

We are particularly interested in Markov games with a single player, which are continuous-time Markov decision processes (CTMDPs). In CTMDPs all positions belong to the reachability player ($L = L_r$), or to the safety player ($L = L_s$), depending on whether we analyse the *maximum* or *minimum* reachability probability problem.

4. Approximating optimal control for normed Markov games

In this section we describe ε-nets, which are a technique for approximating optimal values and strategies in a normed continuous-time Markov game. Thus, throughout the whole section, we fix a normed Markov game $\mathcal{M} = (L, L_r, L_s, \Sigma, \mathbf{R}, \nu)$.

Our approach to approximating optimal control within the Markov game is to break time into intervals of length ε, and to approximate optimal control separately in each of the $\lceil \frac{T}{\varepsilon} \rceil$ distinct intervals. Optimal time-bounded reachability probabilities are then computed iteratively for each interval, starting with the final interval and working backwards in time. The error made by the approximation in each interval is called the *step error*. In Section 4.1 we show that if the step error in each interval is bounded, then the *global error* made by our approximations is also bounded.

Our results begin with a simple approximation that finds the optimal action at the start of each interval, and assumes that this action is optimal for the duration of the interval. We refer to this as the *single* ε-net technique, and we will discuss this approximation in Section 4.2. While it only gives a simple linear function as an approximation, this technique gives error bounds of $O(\varepsilon^2)$, which is comparable to existing techniques.

However, single ε-nets are only a starting point for our results. Our main observation is that, if we have a piecewise polynomial approximation of degree c that achieves an error bound of $O(\varepsilon^k)$, then we can compute a piecewise polynomial approximation of degree $c+1$ that achieves an error bound of $O(\varepsilon^{k+1})$. Thus, starting with single ε-nets, we can construct double ε-nets, triple ε-nets, and quadruple ε-nets, with each of these approximations becoming increasingly more precise. The construction of these approximations will be discussed in Sections 4.3 and 4.4.

In addition to providing an approximation of the time-bounded reachability probabilities, our techniques also provide memoryless strategies for both players. For each level of ε-net, we will define two approximations: the function p_1 is the approximation for the time-bounded reachability probability given by single ε-nets, and the function g_1 gives the reachability probability obtained by following the memoryless strategy that is derived from p_1. This notation generalises to deeper levels of ε-nets: the functions p_2 and g_2 are produced by double ε-nets, and so on.

We will use $\mathcal{E}(k, \varepsilon)$ to denote the difference between p_k and f. In other words, $\mathcal{E}(k, \varepsilon)$ gives the difference between the approximation p_k and the true optimal reachability probabilities. We will use $\mathcal{E}_s(k, \varepsilon)$ to denote the difference between g_k and f. We defer formal definition of these measures to subsequent sections. Our objective in the following subsections is to show that the step errors $\mathcal{E}(k, \varepsilon)$ and $\mathcal{E}_s(k, \varepsilon)$ are in $O(\varepsilon^{k+1})$, with small constants.

4.1. Step error and global error

In subsequent sections we will prove bounds on the *ε-step* error made by our approximations. This is the error that is made in a single interval of length ε. However, in order for our approximations to be valid, they must provide a bound on the *global* error, which is the error made by our approximations over every ε interval. In this section, we prove that, if the ε-step error of an approximation is bounded, then the global error of the approximation is bounded by the sum of these errors.

We define $f : [0, T] \to [0, 1]^{|L|}$ as the vector valued function $f(t) \mapsto \bigotimes_{l \in L} f(l, t)$ that maps each point of time to a vector of reachability probabilities, with one entry for each location. Given two such vectors $f(t)$ and $p(t)$, we define the maximum norm $\|f(t) - p(t)\| = \max\{|f(l, t) - p(l, t)| \mid l \in L\}$, which gives the largest difference between $f(l, t)$ and $p(l, t)$.

We also introduce notation that will allow us to define the values at the start of an ε interval. For each interval $[t - \varepsilon, t]$, we define $f_x^t : [t - \varepsilon, t] \to [0, 1]^{|L|}$ to be the function obtained from the differential equations (1) when the values at the time t are given by the vector $x \in [0, 1]^{|L|}$. More formally, if $\tau = t$ then we define $f_x^t(\tau) = x$, and if $t - \varepsilon \leq \tau < t$ and $l \in L$ then we define:

$$-\dot{f}_x^t(l, \tau) = \operatorname*{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') f_x^t(l', \tau). \tag{3}$$

The following lemma states that if the ε-step error is bounded for every interval, then the global error is bounded by the sum of these errors.

Lemma 5. *Let*

1. *f be a function obtained from the differential equations (1),*
2. *p be an approximation of f that satisfies $\|f(t) - p(t)\| \leq \mu$ for some time point $t \in [0, T]$, and*
3. *$\|f_{p(t)}^t(t - \varepsilon) - p(t - \varepsilon)\| \leq \nu$ for some $\varepsilon \geq 0$.*

Then we have $\|f(t - \varepsilon) - p(t - \varepsilon)\| \leq \mu + \nu$.

Proof. Let $\tau \in [0, \varepsilon]$. We prove first that the maximum norm cannot diverge (to the left) by showing $\|f(t - \tau) - f_{p(t)}^t(t - \tau)\| \leq \mu$. In the proof, we assume $\|f(t - \tau) - f_{p(t)}^t(t - \tau)\| > 0$, remarking that $\|f(t - \tau) - f_{p(t)}^t(t - \tau)\| = 0$ implies $f = f_{p(t)}^t$.

Let l^* be a maximising location, such that

1. $f^t_{p(t)}(l^*, t-\tau) - f(l^*, t-\tau) = \|f(t-\tau) - f^t_{p(t)}(t-\tau)\|$ holds and l^* is owned by the safety player,
2. $f^t_{p(t)}(l^*, t-\tau) - f(l^*, t-\tau) = \|f(t-\tau) - f^t_{p(t)}(t-\tau)\|$ holds and l^* is owned by the reachability player,
3. $f(l^*, t-\tau) - f^t_{p(t)}(l^*, t-\tau) = \|f(t-\tau) - f^t_{p(t)}(t-\tau)\|$ holds and l^* is owned by the safety player,
4. $f(l^*, t-\tau) - f^t_{p(t)}(l^*, t-\tau) = \|f(t-\tau) - f^t_{p(t)}(t-\tau)\|$ holds and l^* is owned by the reachability player, or

We complete the proof only for the first case, remarking that the proofs for all cases are quite similar. By definitions, we have:

$$-\dot{f}(l^*, t-\tau) = \min_{a \in \Sigma(l^*)} \sum_{l \in L} \mathbf{R}(l^*, a, l)(f(l, t-\tau) - f(l^*, t-\tau)).$$

Let a^* be a minimising action, then we have:

$$-\dot{f}(l^*, t-\tau) = \left(\sum_{l \in L} \mathbf{R}(l^*, a^*, l) f(l, t-\tau)\right) - f(l^*, t-\tau)$$

Further, we can derive:

$$\begin{aligned}
\dot{f}(l^*, t-\tau) - \dot{f}^t_{p(t)}(l^*, t-\tau) &= f(l^*, t-\tau) - \left(\sum_{l \in L} \mathbf{R}(l^*, a^*, l) f(l, t-\tau)\right) \\
&\quad - f^t_{p(t)}(l^*, t-\tau) + \min_{a \in \Sigma(l^*)} \left(\sum_{l \in L} \mathbf{R}(l^*, a, l) f^t_{p(t)}(l, t-\tau)\right) \\
&\leq f(l^*, t-\tau) - \left(\sum_{l \in L} \mathbf{R}(l^*, a^*, l) f(l, t-\tau)\right) \\
&\quad - f^t_{p(t)}(l^*, t-\tau) + \left(\sum_{l \in L} \mathbf{R}(l^*, a^*, l) f^t_{p(t)}(l, t-\tau)\right).
\end{aligned}$$

Taking into account that $\sum_{l \in L} \mathbf{R}(l^*, a^*, l)$ describes an affine combination and that $f^t_{p(t)}(l, t-\tau) - f(l, t-\tau) \leq f^t_{p(t)}(l^*, t-\tau) - f(l^*, t-\tau)$ holds (as we are in case (1)), this implies $\dot{f}(l^*, t-\tau) - \dot{f}^t_{p(t)}(l^*, t-\tau) \leq 0$, and thus

$$\dot{f}(l^*, t-\tau) \leq \dot{f}^t_{p(t)}(l^*, t-\tau).$$

Consequently, f and $f^t_{p(t)}$ do not diverge to the left at location l^* at time $t-\tau$. With similar estimations for cases (2), (3), and (4), we obtain that f and $f^t_{p(t)}$ do not diverge at any location l with $|f(l^*, t-\tau) - f^t_{p(t)}(l^*, t-\tau)| = \|f(t-\tau) - f^t_{p(t)}(t-\tau)\|$, and with the smoothness of these functions we obtain that f and $f^t_{p(t)}$ do not diverge to the left in the maximum norm.

The result can then be obtained by a simple triangulation. By assumption we have $\|f^t_{p(t)}(t-\varepsilon) - p(t-\varepsilon)\| \leq \nu$ and $\|f(t) - p(t)\| \leq \mu$. For the latter, we have shown that it implies $\|f(t-\varepsilon) - f^t_{p(t)}(t-\varepsilon)\| \leq \mu$. The triangle inequality implies that $\|f(t-\varepsilon) - p(t-\varepsilon)\| \leq \mu + \nu$. □

4.2. Single ε-nets

4.2.1. The approximation function

In single ε-nets, we compute the gradient of the function f at the end of each interval, and we assume that this gradient remains constant throughout the interval. This yields a *linear* approximation function p_1, which achieves a local error of ε^2.

We now define the function p_1. For initialisation, we define $p_1(l, T) = 1$ if $l \in G$ and $p_1(l, T) = 0$ otherwise. Then, if p_1 is defined for the interval $[t, T]$, we will use the following procedure to extend it to the interval $[t-\varepsilon, T]$. We first determine the optimising enabled actions for each location for $f^t_{p_1(t)}$ at time t. That is, we choose, for all $l \in L$, an action:

$$a^t_l \in \operatorname*{arg\,opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot p_1(l', t). \tag{4}$$

We then fix $c_l^t = \sum_{l' \in L} \mathbf{Q}(l, a_l^t, l') \cdot p_1(l', t)$ as the descent of $p_1(l, \cdot)$ in the interval $[t - \varepsilon, t]$. Therefore, for every $\tau \in [0, \varepsilon]$ and every $l \in L$ we have:

$$-\dot{p}_1(l, t - \tau) = c_l^t \quad \text{and} \quad p_1(l, t - \tau) = p_1(l, t) + \tau \cdot c_l^t. \tag{5}$$

Let us return to our running example. We will apply the approximation p_1 to the example shown in Fig. 1. We will set $\varepsilon = 0.1$, and focus on the interval $[1.1, 1.2]$ with initial values $p_1(G, 1.2) = 1$, $p_1(l, 1.2) = 0.244$, $p_1(l_R, 1.2) = 0.107$, $p_1(l_S, 1.2) = 0.075$, $p_1(\bot, 1.2) = 0$. These are close to the true values at time 1.2. Note that the point t_1, which is the time at which the reachability player switches the action played at l_R, is contained in the interval $[1.1, 1.2]$. Applying Equation (4) with these values allows us to show that the maximising action at l_R is a, and the minimising action at l_S is also a. As a result, we obtain the approximation $p_1(l_R, t - \tau) = 0.0286\tau + 0.107$ and $p_1(l_S, t - \tau) = 0.032\tau + 0.075$.

We now prove error bounds for p_1. Recall that $\mathcal{E}(1, \tau)$ denotes the difference between f and p_1 after τ time units. We can now formally define this as $\mathcal{E}(1, \varepsilon) := \| f^t_{p_1(t)}(t - \varepsilon) - p_1(t - \varepsilon) \|$. We now give a sequence of three lemmas, with the goal of proving an upper bound on $\mathcal{E}(1, \tau)$. We begin by showing bounds on the range of values that p_1 may take.

Lemma 6. *If $\varepsilon \leq 1$, then we have $p_1(l, t) \in [0, 1]$ for all $t \in [0, T]$.*

Proof. We will prove this by induction over the intervals $[t - \varepsilon, t]$. The base case is trivial since we have by definition that either $p_1(l, T) = 0$ or $p_1(l, T) = 1$. Now suppose that $p_1(l, t) \in [0, 1]$ for some ε-interval $[t - \varepsilon, t]$. We will prove that $p_1(l, t - \tau) \in [0, 1]$ for all $\tau \in [0, \varepsilon]$.

Since $\tau \leq \varepsilon \leq 1$, from Equation (5) we have:

$$\begin{aligned} p_1(l, t - \tau) &= p_1(l, t) + \tau \cdot \sum_{l' \in L} \mathbf{R}(l, a_l^t, l') \cdot (p_1(l', t) - p_1(l, t)) \\ &\leq p_1(l, t) + \sum_{l' \in L} \mathbf{R}(l, a_l^t, l') \cdot (p_1(l', t) - p_1(l, t)) \\ &= \left(1 - \sum_{l' \in L} \mathbf{R}(l, a_l^t, l')\right) \cdot p_1(l, t) + \sum_{l' \in L} \mathbf{R}(l, a_l^t, l') \cdot p_1(l', t). \end{aligned}$$

Since we are considering normed Markov games, we have that $\sum_{l' \neq L} \mathbf{R}(l, a_l^t, l') \leq 1$, and therefore $p_1(l, t - \tau)$ is a weighted average over the values $p_1(l', t)$ where $l' \in L$. From the inductive hypothesis, we have that $p_1(l', t) \in [0, 1]$ for every $l' \in L$, and therefore a weighted average over these values must also lie in $[0, 1]$. □

Next, we show bounds on the range of values that $-\dot{f}^t_{p_1(t)}$ may take.

Lemma 7. *If $\varepsilon \leq 1$ then we have $-\dot{f}^t_{p_1(t)}(l, t - \tau) \in [-1, 1]$ for every $\tau \in [0, \varepsilon]$.*

Proof. Lemma 6 implies that $f^t_{p_1(t)}(l, t) = p_1(l, t) \in [0, 1]$ for all $l \in L$. The following argument is similar of that used in 5: When some value $f^t_{p_1(t)}(l, t - \tau) = 0$ and other values $f^t_{p_1(t)}(l, t - \tau) \geq 0$, then $\dot{f}^t_{p_1(t)}(l, t - \tau) \geq 0$. Similarly, when some value $f^t_{p_1(t)}(l, t - \tau) = 1$ and other values $f^t_{p_1(t)}(l, t - \tau) \leq 1$, then $\dot{f}^t_{p_1(t)}(l, t - \tau) \leq 0$. Thus, the analytical function $f^t_{p_1(t)}(l, t - \tau)$ cannot break out of the $[0, 1]$ interval for all $\tau \in [0, \varepsilon]$.

We first prove that $-\dot{f}^t_{p_1(t)}(l, t - \tau) \leq 1$. We will prove this for the reachability player, the proof for the safety player is analogous. By definition we have:

$$-\dot{f}^t_x(l, t - \tau) = \max_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l, a, l')(f^t_{p_1(t)}(l', t - \tau) - f^t_{p_1(t)}(l, t - \tau)).$$

Since we have shown that $f^t_{p_1(t)}(l', t - \tau) \in [0, 1]$ for all l, and we have $\sum_{l' \in L} \mathbf{R}(l, a, l') = 1$ for every action a in a normed Markov game, we obtain:

$$\begin{aligned} &\max_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l, a, l')(f^t_{p_1(t)}(l', t - \tau) - f^t_{p_1(t)}(l, t - \tau)) \\ &\leq \max_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l, a, l')(1 - 0) = 1. \end{aligned}$$

To prove that $-\dot{f}^t_{p_1(t)}(l, t-\tau) \geq -1$ we use a similar argument:

$$\max_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l, a, l')(f^t_{p_1(t)}(l', t-\tau) - f^t_{p_1(t)}(l, t-\tau))$$
$$\geq \max_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l, a, l')(0 - 1) = -1.$$

Therefore we have $-\dot{f}^t_{p_1(t)}(l, t-\tau) \in [-1, 1]$. □

Finally, we can use Lemma 7 to provide an upper bound on $\mathcal{E}(1, \varepsilon)$.

Lemma 8. *If $\varepsilon \leq 1$, then $\mathcal{E}(1, \varepsilon) := \| f^t_{p_1(t)}(t-\varepsilon) - p_1(t-\varepsilon) \| \leq \varepsilon^2$.*

Proof. Lemma 7 implies that $-\dot{f}^t_{p_1(t)}(l, t-\tau) \in [-1, 1]$ for every $\tau \in [0, \varepsilon]$. Since the rate of change of $f^t_{p_1(t)}$ is in the range $[-1, 1]$, we know that $f^t_{p_1(t)}$ can change by at most τ in the interval $[t-\tau, t]$. We also know that $f^t_{p_1(t)}(l, t) = p_1(l, t)$, and therefore we must have the following property:

$$\| f^t_{p_1(t)}(t-\tau) - p_1(t) \| \leq \tau. \tag{6}$$

The key step in this proof is to show that $\| \dot{f}^t_{p_1(t)}(t-\tau) - \dot{p}_1(t-\tau) \| \leq 2 \cdot \tau$ for all $\tau \in [0, \varepsilon]$. Note that by definition we have $\dot{p}_1(l, t-\tau) = \dot{p}_1(l, t)$ for all $\tau \in [0, \varepsilon]$, and so it suffices to prove that $\| \dot{f}^t_{p_1(t)}(t-\tau) - \dot{p}_1(t) \| \leq 2 \cdot \tau$.

Suppose that l is a location for the reachability player, let a^t_l be the optimal action at time t, and let $a^{t-\tau}_l$ be the optimal action at $t-\tau$. We have the following:

$$\begin{aligned}
-\dot{p}_1(l, t) - 2 \cdot \tau &= \sum_{l' \in L} \mathbf{R}(l, a^t_l, l')(p_1(l', t) - p_1(l, t)) - 2 \cdot \tau \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^t_l, l')(f^t_{p_1(t)}(l', t-\tau) - f^t_{p_1(t)}(l, t-\tau)) \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^{t-\tau}_l, l')(f^t_{p_1(t)}(l', t-\tau) - f^t_{p_1(t)}(l, t-\tau)) \\
&= -\dot{f}^t_{p_1(t)}(l, t-\tau).
\end{aligned}$$

The first equality is the definition of $-\dot{p}_1(l, t)$. The first inequality follows from Equation (6) and the fact that $\mathbf{R}(l, a, l') = 1$. The second inequality follows from the fact that $a^{t-\tau}_l$ is an optimal action at time $t-\tau$, and the final equality is the definition of $-\dot{f}^t_{p_1(t)}(l, t-\tau)$. Using the same techniques in a different order gives:

$$\begin{aligned}
-\dot{f}^t_{p_1(t)}(l, t-\tau) &= \sum_{l' \in L} \mathbf{R}(l, a^{t-\tau}_l, l')(f^t_{p_1(t)}(l', t-\tau) - f^t_{p_1(t)}(l, t-\tau)) \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^{t-\tau}_l, l')(p_1(l', t) - p_1(l, t)) + 2 \cdot \tau \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^t_l, l')(p_1(l', t) - p_1(l, t)) + 2 \cdot \tau \\
&= -\dot{p}_1(l, t) + 2 \cdot \tau.
\end{aligned}$$

To prove the claim for a location l belonging to the safety player, we use the same arguments, but in reverse order. That is, we have:

$$\begin{aligned}
-\dot{p}_1(l, t) - 2 \cdot \tau &= \sum_{l' \in L} \mathbf{R}(l, a^t_l, l')(p_1(l', t) - p_1(l, t)) - 2 \cdot \tau \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^{t-\tau}_l, l')(p_1(l', t) - p_1(l, t)) - 2 \cdot \tau \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^{t-\tau}_l, l')(f^t_{p_1(t)}(l', t-\tau) - f^t_{p_1(t)}(l, t-\tau)) \\
&= -\dot{f}^t_{p_1(t)}(l, t-\tau).
\end{aligned}$$

We also have:

$$\begin{aligned}
-\dot{f}^t_{p_1(t)}(l, t-\tau) &= \sum_{l' \in L} \mathbf{R}(l, a^{t-\tau}_l, l')(f^t_{p_1(t)}(l', t-\tau) - f^t_{p_1(t)}(l, t-\tau)) \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^t_l, l')(f^t_{p_1(t)}(l', t-\tau) - f^t_{p_1(t)}(l, t-\tau)) \\
&\leq \sum_{l' \in L} \mathbf{R}(l, a^t_l, l')(p_1(l', t) - p_1(l, t)) + 2 \cdot \tau \\
&= -\dot{p}_1(l, t) + 2 \cdot \tau.
\end{aligned}$$

Therefore, we have shown that $\|\dot{f}^t_{p_1(t)}(t-\tau) - \dot{p}_1(t-\tau)\| \leq 2 \cdot \tau$ for all $\tau \in [0, \varepsilon]$.

We can complete the proof by observing that $\int_0^\tau 2 \cdot \tau d\tau = \tau^2$. This allows us to conclude that $\mathcal{E}(1, \varepsilon) := \|f^t_{p_1(t)}(t-\varepsilon) - p_1(t-\varepsilon)\| \leq \varepsilon^2$. □

4.2.2. Strategies

The approximation p_1 can also be used to construct strategies for the two players with similar error bounds. We will describe the construction for the reachability player. The construction for the safety player can be derived analogously.

The strategy for the reachability player is to play the action chosen by p_1 during the entire interval $[t-\varepsilon, t]$. We will define a system of differential equations $g_1(l, \tau)$ that describe the outcome when the reachability fixes this strategy, and when the safety player plays an optimal counter strategy. For each location l, we define $g_1(l, t) = f^t_{p_1(t)}(l, t)$, and we define $g_1(l, \tau)$, for each $\tau \in [t-\varepsilon, t]$, as:

$$-\dot{g_1}(l, \tau) = \sum_{l' \in L} \mathbf{Q}(l, a^t_l, l') \cdot g_1(l', \tau) \qquad \text{if } l \in L_r, \tag{7}$$

$$-\dot{g_1}(l, \tau) = \min_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g_1(l', \tau) \qquad \text{if } l \in L_s. \tag{8}$$

We will prove bounds for $\mathcal{E}_s(1, \varepsilon)$, which is the difference between g_1 and $f^t_{p_1(t)}$ on an interval of length ε. We begin by proving the following auxiliary lemma, which shows that the difference between p_1 and g_1 is bounded by ε^2.

Lemma 9. *We have* $\|g_1(t-\varepsilon) - p_1(t-\varepsilon)\| \leq \varepsilon^2$.

Proof. Suppose that we apply single ε-nets to approximate the solution of the system of differential equations g_1 over the interval $[t-\varepsilon, t]$ to obtain an approximation p^g_1. To do this, we select for each location $l \in L_s$ an action a that satisfies:

$$a \in \operatorname*{arg\,opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g_1(l', t).$$

Since $g_1(l, t) = f^t_{p_1(t)}(l, t) = p_1(l, t)$ for every location l, we have that $a = a^t_l$, where a^t_l is the action chosen by p_1 at l. In other words, the approximations p_1 and p^g_1 choose the same actions for every location in L_s. Therefore, for all locations $l \in L$, we have $c^t_l = \sum_{l' \in L} \mathbf{Q}(l, a^t_l, l') \cdot p^g_1(l', t) = \sum_{l' \in L} \mathbf{Q}(l, a^t_l, l') \cdot p_1(l', t)$, which implies that for every time $\tau \in [0, \varepsilon]$ we have:

$$p^g_1(l, t-\tau) = p^g_1(l, t) + \tau \cdot c^t_l = p_1(l, t) + \tau \cdot c^t_l = p_1(l, t-\tau).$$

That is, the approximations p_1 and p^g_1 are identical.

Note that the system of differential equations g_1 describes a continuous-time Markov game in which some actions for the reachability player have been removed. Since g_1 describes a CTMG, we can apply Lemma 8 to obtain $\|g_1(t-\tau) - p^g_1(t-\tau)\| \leq \varepsilon^2$. Since $p_1(t-\varepsilon) = p^g_1(t-\varepsilon)$, we can conclude that $\|g_1(t-\tau) - p_1(t-\tau)\| \leq \varepsilon^2$. □

We can now combine Lemma 9 with Lemma 8 to obtain the following bound on $\mathcal{E}_s(1, \varepsilon)$.

Lemma 10. *We have* $\mathcal{E}_s(1, \varepsilon) := \|g_1(t-\varepsilon) - f^t_{p_1(t)}(t-\varepsilon)\| \leq 2 \cdot \varepsilon^2$.

4.2.3. The approximation algorithm

Lemma 8 gives the ε-step error for p_1, and we can apply Lemma 5 to show that the global error is bounded by $\varepsilon^2 \cdot \frac{T}{\varepsilon} = \varepsilon T$. If π is the required precision, then we can choose $\varepsilon = \frac{\pi}{T}$ to produce an algorithm that terminates after $\frac{T}{\varepsilon} \approx \frac{T^2}{\pi}$ many steps. Hence, we obtain the following known result.

Theorem 11. *For a normed Markov game $\mathcal{M}$ of size $|\mathcal{M}|$, we can compute a π-optimal strategy and determine the quality of $\mathcal{M}$ up to precision π in time $O(|\mathcal{M}| \cdot T \cdot \frac{T}{\pi})$.*

Proof. As we have argued, in order to guarantee a precision of π, it suffices to choose $\varepsilon = \frac{\pi}{T}$, which gives $\frac{T^2}{\pi}$ many intervals $[t-\varepsilon, t]$ for which p_1 must be computed. It is clear that, for each interval, the approximation p_1 can be computed in $O(\mathcal{M})$ time, and therefore, the total running time will be $O(|\mathcal{M}| \cdot T \cdot \frac{T}{\pi})$. □

4.3. Double ε-nets

4.3.1. The approximation function

In this section we show that only a small amount of additional computation effort needs to be expended in order to dramatically improve over the precision obtained by single ε-nets. This will allow us to use much larger values of ε while still retaining our desired precision.

In single ε-nets, we computed the gradient of f at the start of each interval and assumed that the gradient remained constant for the duration of that interval. This gave us the approximation p_1. The key idea behind double ε-nets is that we can use the approximation p_1 to approximate the gradient of f throughout the interval.

We define the approximation p_2 as follows: we have $p_2(l, T) = 1$ if $l \in G$ and 0 otherwise, and if $p_2(l, \tau)$ is defined for every $l \in L$ and every $\tau \in [t, T]$, then we define $p_2(l, \tau)$ for every $\tau \in [t-\varepsilon, t]$ as:

$$-\dot{p}_2(l, \tau) = \operatorname*{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l, a, l') \cdot \big(p_1(l', \tau) - p_1(l, \tau)\big) \quad \forall l \in L. \tag{9}$$

By comparing Equations (9) and (2), we can see that double ε-nets uses p_1 as an approximation for f during the interval $[t-\varepsilon, t]$. Furthermore, in contrast to p_1, note that the approximation p_2 can change it's choice of optimal action during the interval. The ability to change the choice of action during an interval is the key property that allows us to prove stronger error bounds than previous work.

Lemma 12. *If $\varepsilon \leq 1$ then $\mathcal{E}(2, \varepsilon) := \|p_2(\tau) - f^t_{p_2(t)}(\tau)\| \leq \frac{2}{3}\varepsilon^3$.*

Proof. We begin by considering the system of differential equations that define p_2, as given in Equation (9):

$$-\dot{p}_2(l, \tau) = \operatorname*{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l, a, l') \cdot \big(p_1(l', \tau) - p_1(l, \tau)\big) \quad \forall l \in L.$$

The error bounds given by Lemma 8 imply that $\|p_1(t-\tau) - f^t_{p_2(t)}(t-\tau)\| \leq \tau^2$ for every $\tau \in [0, \varepsilon]$. Therefore, for every pair of locations $l, l' \in L$ and every $\tau \in [t-\varepsilon, t]$ we have:

$$|(p_1(l', t-\tau) - p_1(l, t-\tau)) - (f^t_{p_2(t)}(l', t-\tau) - f^t_{p_2(t)}(l, t-\tau))| \leq 2 \cdot \tau^2.$$

Since we are dealing with normed Markov games, we have $\sum_{l' \in L} \mathbf{R}(l, a, l') = 1$ for every location $l \in L$ and every action $a \in A(l)$. Therefore, we also have for every action a:

$$\left| \sum_{l' \in L} \mathbf{R}(l, a, l') \cdot \big(p_1(l', t-\tau) - p_1(l, t-\tau)\big) - \sum_{l' \in L} \mathbf{R}(l, a, l') \cdot (f^t_{p_2(t)}(l', t-\tau) - f^t_{p_2(t)}(l, t-\tau) \right| \leq 2 \cdot \tau^2.$$

This implies that $\|\dot{p}_2(t-\tau) - \dot{f}^t_{p_2(t)}(t-\tau)\| \leq 2 \cdot \tau^2$.

We can obtain the claimed result by integrating over this difference:

$$|p_2(l, t-\tau) - f^t_{p_2(t)}(l, t-\tau)| \leq \int_0^\tau |\dot{p}_2(l, t-\tau) - \dot{f}^t_{p_2(t)}(l, t-\tau)| \leq \frac{2}{3}\tau^3.$$

Therefore, the total amount of error incurred by p_2 in the interval $[t-\varepsilon, t]$ is at most $\frac{2}{3}\varepsilon^3$. □

Let us apply the approximation p_2 to the example shown in Fig. 1. We will again use the interval $[1.1, 1.2]$, and we will use initial values that were used when we applied single ε-nets to the example in Section 4.2. We will focus on the location l_R. From the previous section, we know that $p_1(l_R, t-\tau) = 0.0286\tau + 0.107$, and for the actions a and b we have:

- $\sum_{l' \in L} \mathbf{R}(l_R, a, l') p_1(l', t-\tau) = \frac{1}{20} + \frac{4}{5} p_1(l_R, t-\tau)$.
- $\sum_{l' \in L} \mathbf{R}(l_R, b, l') p_1(l', t-\tau) = \frac{1}{5} p_1(l, t-\tau) + \frac{4}{5} p_1(l_R, t-\tau)$.

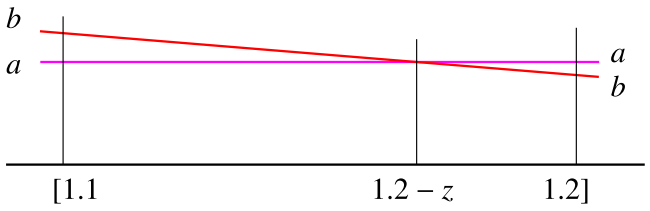

Fig. 2. This figure shows how $-\dot{p}_2$ is computed on the interval $[1.1, 1.2]$ for the location l_R. The function is given by the upper envelope of the two functions: it agrees with the quality of a on the interval $[1.2 - z, 1.2]$ and with the quality of b on the interval $[1.1, 1.2 - z]$.

These functions are shown in Fig. 2. To obtain the approximation p_2, we must take the maximum of these two functions. Since p_1 is a linear function, we know that these two functions have exactly one crossing point, and it can be determined that this point occurs when $p_1(l, t - \tau) = 0.25$, which happens at $\tau = z := \frac{5}{63}$. Since $z \leq 0.1 = \varepsilon$, we know that the lines intersect within the interval $[1.1, 1.2]$. Consequently, we get the following piecewise quadratic function for p_2:

- When $0 \leq \tau \leq z$, we use the action a and obtain $-\dot{p}_2(l_R, t - \tau) = -0.00572\tau + 0.0286$, which implies that $p_2(l_R, t - \tau) = -0.00286\tau^2 + 0.0286\tau + 0.107$.
- When $z < \tau \leq 0.1$ we use action b and obtain $-\dot{p}_2(l_R, t - \tau) = 0.0094\tau + 0.0274$, which implies that $p_2(l_R, t - \tau) = 0.0047\tau^2 + 0.0274\tau + 0.107047619$.

4.3.2. Strategies

As with single ε-nets, we can provide a strategy that obtains similar error bounds. Once again, we will consider only the reachability player, because the proof can easily be generalised for the safety player. In much the same way as we did for g_1, we will define a system of differential equations $g_2(l, \tau)$ that describe the outcome when the reachability player plays according to p_2, and the safety player plays an optimal counter strategy. For each location l, we define $g_2(l, t) = f^t_{p_2(t)}(l, t)$. If a^τ_l denotes the action that maximises Equation (9) at the time point $\tau \in [t - \varepsilon, t]$, then we define $g_2(l, \tau)$, as:

$$-\dot{g_2}(l, \tau) = \sum_{l' \in L} \mathbf{Q}(l, a^\tau_l, l') \cdot g_2(l', \tau) \qquad \text{if } l \in L_r, \tag{10}$$

$$-\dot{g_2}(l, \tau) = \min_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g_2(l', \tau) \qquad \text{if } l \in L_s. \tag{11}$$

The following lemma proves that difference between g_2 and $f^t_{p_2(t)}$ has similar bounds to those shown in Lemma 12. The rest of this subsection is dedicated to proving this lemma.

Lemma 13. *If $\varepsilon \leq 1$ then we have $\mathcal{E}_s(2, \varepsilon) := \|g_2(t - \varepsilon) - f^t_{p_2(t)}(t - \varepsilon)\| \leq 2 \cdot \varepsilon^3$.*

To begin, we prove an auxiliary lemma, that will be used throughout the rest of the proof.

Lemma 14. *Let f and g be two functions such that $\|f(t - \tau) - g(t - \tau)\| \leq c \cdot \tau^k$. If a^f is an action that maximises (resp. minimises)*

$$\operatorname*{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot f(l', t - \tau), \tag{12}$$

and a^g is an action that maximises (resp. minimises)

$$\operatorname*{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g(l', t - \tau), \tag{13}$$

then we have:

$$\left| \sum_{l' \in L} \mathbf{R}(l, a^g, l') \cdot g(l', t - \tau) - \sum_{l' \in L} \mathbf{R}(l, a^f, l') \cdot f(l', t - \tau) \right| \leq 3 \cdot c \cdot \tau^k.$$

Proof. We will provide a proof for the case where the equations are maximising, the proof for the minimisation case is similar. We begin by noting that the property $\|f(t - \tau) - g(t - \tau)\| \leq c \cdot \tau^k$, and the fact that we consider only normed Markov games imply that, for every action a we have:

$$\left| \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot f(l', t - \tau) - \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g(l', t - \tau) \right| \leq 2 \cdot c \cdot \tau^k. \tag{14}$$

We use this to claim that the following inequality holds:

$$\left| \sum_{l' \in L} \mathbf{Q}(l, a^g, l') \cdot g(l', t-\tau) - \sum_{l' \in L} \mathbf{Q}(l, a^f, l') \cdot f(l', t-\tau) \right| \leq 2 \cdot c \cdot \tau^k. \tag{15}$$

To see why, suppose that

$$\sum_{l' \in L} \mathbf{Q}(l, a^g, l') \cdot g(l', t-\tau) > \sum_{l' \in L} \mathbf{Q}(l, a^f, l') \cdot f(l', t-\tau) + 2 \cdot c \cdot \tau^k.$$

Then we could invoke Equation (14) to argue that $\sum_{l' \in L} \mathbf{Q}(l, a^g, l') \cdot f(l', t-\tau) > \sum_{l' \in L} \mathbf{Q}(l, a^f, l') \cdot f(l', t-\tau)$, which contradicts the fact that a^f achieves the maximum in Equation (12). Similarly, if $\sum_{l' \in L} \mathbf{Q}(l, a^f, l') \cdot f(l', t-\tau) > \sum_{l' \in L} \mathbf{Q}(l, a^g, l') \cdot g(l', t-\tau) + 2 \cdot c \cdot \tau^k$, then we can invoke Equation (14) to argue that a^g does not achieve the maximum in Equation (13). Therefore, Equation (15) must hold.

Now, to finish the proof, we apply the fact that $\|f(t-\tau) - g(t-\tau)\| \leq c \cdot \tau^k$ to Equation (15) to obtain:

$$\left| \sum_{l' \in L} \mathbf{R}(l, a^g, l') g(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^f, l') f(l', t-\tau) \right| \leq 3 \cdot c \cdot \tau^k.$$

This completes the proof. □

To prove Lemma 13 we will consider the following class of strategies: play the action chosen by p_2 for the first k transitions, and then play the action chosen by p_1 for the remainder of the interval. We will denote the reachability probability obtained by this strategy as g_2^k, and we will denote the error of this strategy as $\mathcal{E}_s^k(2, \varepsilon) := \|g_2^k(t-\varepsilon) - f_{p_2(t)}^t(t-\varepsilon)\|$. Clearly, as k approaches infinity, we have that g_2^k approaches g_2, and $\mathcal{E}_s^k(2, \varepsilon)$ approaches $\mathcal{E}_s(2, \varepsilon)$. Therefore, in order to prove Lemma 13, we will show that $\mathcal{E}_s^k(2, \varepsilon) \leq 2 \cdot \varepsilon^3$ for all k.

We will prove error bounds on g_2^k by induction. The following lemma considers the base case, where $k = 1$. In other words, it considers the strategy that plays the action chosen by p_2 for the first transition, and then plays the action chosen by p_1 for the rest of the interval.

Lemma 15. *If $\varepsilon \leq 1$, then we have $\mathcal{E}_s^1(2, \varepsilon) \leq 2 \cdot \varepsilon^3$.*

Proof. Suppose that the first discrete transition occurs at time $t - \tau$, where $\tau \in [0, \varepsilon]$. Let l be a location belonging to the reachability player, and let $a_l^{t-\tau}$ be the action that maximises p_2 at time $t - \tau$. By definition, we know that the probability of moving to a location l' is given by $\mathbf{R}(l, a_l^{t-\tau}, l')$, and we know that the time-bounded reachability probabilities for each state l' are given by $g_1(l', t-\tau)$. Therefore, the outcome of choosing $a_l^{t-\tau}$ at time $t - \tau$ is $\sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') g_1(l', t-\tau)$. If a^* is an action that would be chosen by $f_{p_2(t)}^t$ at time $t - \tau$, then we have the following bounds:

$$\begin{aligned}
&\left| \sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') g_1(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f_{p_2(t)}^t(l', t-\tau) \right| \\
&\leq \left| \sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') p_1(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f_{p_2(t)}^t(l', t-\tau) \right| + \tau^2 \\
&\leq 4 \cdot \tau^2.
\end{aligned}$$

The first inequality follows from Lemma 9, and the second inequality follows from Lemma 14.

Now suppose that l is a location belonging to the safety player. Since the reachability player will follow p_1 during the interval $[t - \tau, t]$, we know that the safety player will choose an action a^g that minimises:

$$\min_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g_1(l', t-\tau).$$

If a^* is the action chosen by f at time $t - \tau$, then Lemma 10 and Lemma 14 imply:

$$\left| \sum_{l' \in L} \mathbf{R}(l, a^g, l') g_1(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f_{p_2(t)}^t(l', t-\tau) \right| \leq 6 \cdot \tau^2.$$

So far we have proved that the total amount of error made by g_2^1 when the first transition occurs at time $t - \tau$ is at most $6 \cdot \tau^2$. To obtain error bounds for g_2^1 over the entire interval $[t - \varepsilon, t]$, we consider the probability that the first transition actually occurs at time $t - \tau$:

$$\mathcal{E}_s^1(2,\varepsilon) \leq \int_0^\varepsilon e^{\tau-\varepsilon} 6\tau^2 d\tau \leq \int_0^\varepsilon 6\tau^2 d\tau = 2\cdot\varepsilon^3.$$

This completes the proof. □

We now prove the inductive step, by considering g_2^k. This is the strategy that follows the action chosen by p_2 for the first k transitions, and then follows p_1 for the rest of the interval.

Lemma 16. *If $\mathcal{E}_s^k(2,\varepsilon) \leq 2\cdot\varepsilon^3$ for some k, then $\mathcal{E}_s^{k+1}(2,\varepsilon) \leq 2\cdot\varepsilon^3$.*

Proof. The structure of this proof is similar to the proof of Lemma 15, however, we must account for the fact that g_2^{k+1} follows g_2^k after the first transition rather than g_1.

Suppose that we play the strategy for g_2^{k+1}, and that the first discrete transition occurs at time $t-\tau$, where $\tau \in [0,\varepsilon]$. Let l be a location belonging to the reachability player, and let $a_l^{t-\tau}$ be the action that maximises p_2 at time $t-\tau$. If a^* is an action that would be chosen by $f_{p_2(t)}^t$ at time $t-\tau$, then we have the following bounds:

$$\begin{aligned}
&\left|\sum_{l'\in L}\mathbf{R}(l,a_l^{t-\tau},l')g_2^k(l',t-\tau) - \sum_{l'\in L}\mathbf{R}(l,a^*,l')f_{p_2(t)}^t(l',t-\tau)\right| \\
&\leq \left|\sum_{l'\in L}\mathbf{R}(l,a_l^{t-\tau},l')p_1(l',t-\tau) - \sum_{l'\in L}\mathbf{R}(l,a^*,l')f_{p_2(t)}^t(l',t-\tau)\right| + \tau^2 + 2\cdot\tau^3 \\
&\leq 4\cdot\tau^2 + 2\cdot\tau^3 \leq 6\cdot\tau^2.
\end{aligned}$$

The first inequality follows from the inductive hypothesis, which gives bounds on how far g_2^k is from $f_{p_2(t)}^t$, and from Lemma 8, which gives bounds on how far $f_{p_2(t)}^t$ is from p_1. The second inequality follows from Lemma 8 and Lemma 14, and the final inequality follows from the fact that $\tau \leq 1$.

Now suppose that the location l belongs to the safety player. Let a^g be an action that minimises:

$$\min_{a\in\Sigma(l)}\sum_{l'\in L}\mathbf{Q}(l,a,l')\cdot g_2^k(l',t-\tau).$$

If a^* is the action chosen by f at time $t-\tau$, then Lemma 10 and Lemma 14 imply:

$$\left|\sum_{l'\in L}\mathbf{R}(l,a^g,l')g_2^k(l',t-\tau) - \sum_{l'\in L}\mathbf{R}(l,a^*,l')f_{p_2(t)}^t(l',t-\tau)\right| \leq 6\cdot\tau^3 \leq 6\cdot\tau^2.$$

The first inequality follows from the inductive hypothesis and Lemma 14, and the second inequality follows from the fact that $\tau \leq 1$.

To obtain error bounds for g_2^{k+1} over the entire interval $[t-\varepsilon,t]$, we consider the probability that the first transition actually occurs at time $t-\tau$:

$$\mathcal{E}_s^{k+1}(2,\varepsilon) \leq \int_0^\varepsilon e^{\tau-\varepsilon} 6\cdot\tau^2\, d\tau \leq \int_0^\varepsilon 6\tau^2\, d\tau = 2\cdot\varepsilon^3.$$

This completes the proof. □

Having shown Lemmas 15 and 16, Lemma 13 follows with the observation that $\mathcal{E}_s(2,\varepsilon) = \lim_{k\to\infty}\mathcal{E}_s^k(2,\varepsilon)$.

4.3.3. The approximation algorithm

Computing the approximation p_2 for an interval $[t-\varepsilon,t]$ is not expensive. The fact that p_1 is linear implies that each action can be used for at most one subinterval of $[t-\varepsilon,t]$. Therefore, there are less than $|\Sigma|$ points at which the strategy changes, which implies that p_2 is a piecewise quadratic function with at most $|\Sigma|$ pieces. We now present an algorithm that uses sorting to compute these pieces.

Lemma 17. *Computing p_2 for an interval $[t-\varepsilon,t]$ takes $O(|\mathcal{M}| + |L|\cdot|\Sigma|\cdot\log|\Sigma|)$ time.*

Proof. We give an algorithm for the reachability player. The algorithm for the safety player is symmetric. For every location $l \in L$, and time point $\tau \in [0,\varepsilon]$, we define the *quality* of an action a as:

Algorithm 1 BestActions.

```
Sort the actions into a list ⟨a_1, a_2, ..., a_m⟩ such that a_i ≼_l^0 a_{i+1} for all i.
O := ⟨(a_1, 0)⟩.
for i := 2 to m do
  (a, τ) := the last element in O.
  if a ≺_l^ε a_i then
    while true do
      x := the point at which q_l^x(a) = q_l^x(a_i).
      if x ≥ τ then
        Add (a_i, x) to the end of O.
        break
      else
        Remove (a, τ) from O.
        (a, τ) := the last element in O.
      end if
    end while
  end if
end for
return O.
```

$$q_l^\tau(a) := \sum_{l' \in L} \mathbf{Q}(l, a, l') p_1^t(l', t - \tau).$$

We also define an operator that compares the quality of two actions. For two actions a_1 and a_2, we have $a_1 \preceq_l^\tau a_2$ if and only if $q_l^\tau(a_1) \leq q_l^\tau(a_2)$, and we have $a_1 \prec_l^\tau a_2$ if and only if $q_l^\tau(a_1) < q_l^\tau(a_2)$.

Algorithm 1 shows the key component of our algorithm for computing the approximation p_2 during the interval $[t-\varepsilon, t]$. The algorithms outputs a list O containing pairs (a, τ), where a is an action and τ is a point in time, which represents the optimal actions during the interval $[t-\varepsilon, t]$: if the algorithm outputs the list $O = \langle (a_1, \tau_1), (a_2, \tau_2), \ldots, (a_n, \tau_n) \rangle$, then a_1 maximises Equation (9) for the interval $[t-\tau_2, t-\tau_1]$, a_2 maximises Equation (9) for the interval $[t-\tau_3, t-\tau_2]$, and so on.

The algorithm computes O as follows. It begins by sorting the actions according to their quality at time t. Since a_1 maximises the quality at time t, we know that a_1 is chosen by Equation (9) at time t. Therefore, the algorithm initialises O by assuming that a_1 maximises Equation (9) for the entire interval $[t-\varepsilon, t]$. The algorithm then proceeds by iterating through the actions $\langle a_2, a_3, \ldots a_m \rangle$.

We will prove the following invariant on the outer loop of the algorithm: if the first i actions have been processed, then the list O gives the solution to:

$$-\dot{p}_2(l, \tau, i) = \max_{a \in \langle a_1, a_2, \ldots, a_i \rangle} \sum_{l' \in L} \mathbf{R}(l, a, l') \cdot \big(p_1(l', \tau) - p_1(l, \tau)\big). \tag{16}$$

In other words, the list O would be a solution to Equation (9) if the actions $\langle a_{i+1}, a_{i+2}, \ldots a_m \rangle$ did not exist. Clearly, when $i = m$ the list O will actually be a solution to Equation (9).

We will prove this invariant by induction. The base case is trivially true, because when $i = 1$ the maximum in Equation (16) considers only a_1, and therefore a_1 is optimal throughout the interval $[t-\varepsilon, t]$. We now prove the inductive step. Assume that O is a solution to Equation (16) for $i-1$. We must show that Algorithm 1 correctly computes O for i. Let us consider the operations that Algorithm 1 performs on the action a_i. It compares a_i with the pair (a, τ), which is the final pair in O, and one of three actions is performed:

- If $a_i \prec_l^\varepsilon a$, then the algorithm ignores a_i. This is because we have $a_i \prec_l^0 a_1$, which means that a_i is worse than a_1 at time t, and we have $a_i \prec_l^\varepsilon a$, which implies that a_i is worse than a at time $t-\varepsilon$. Since $q_l^\tau(a_i)$ is a linear function, we can conclude that a_i never maximises Equation (9) during the interval $[t-\varepsilon, t]$.
- If x, which is the point at which the functions $q_l^x(a)$ and $q_l^x(a_i)$ intersect, is greater than τ, then we add (a_i, x) to O. This is because the fact that $q_l^x(a_i)$ and $q_l^x(a)$ are linear functions implies that a_i cannot be optimal for every time $\tau' < \tau$.
- Finally, if x is smaller than τ, then we remove (a, τ) from O and continue by comparing a_i to the new final pair in O. From the inductive hypothesis, we have that a is not optimal for every time point $\tau' \leq \tau$, and the fact that $x < \tau$ and the fact that $q_l^x(a_i)$ and $q_l^x(a)$ are linear functions implies that a_i is better than a for every time point $\tau' > \tau$. Therefore, a can never be optimal.

These three observations are sufficient to prove that Algorithm 1 correctly computes O, and O can obviously be used to compute the approximation p_2.

We claim that Algorithm 1 runs in time $O(|\Sigma| \log |\Sigma|)$. Since sorting can be done in $O(|\Sigma| \log |\Sigma|)$ time, the first step of this algorithm also takes $O(|\Sigma| \log |\Sigma|)$. We claim that the remaining steps of the algorithm take $O(|\Sigma|)$ time. To see this, note that after computing a crossing point x, the algorithm either adds an action to the list O, or removes an action from O. Moreover each action a can enter the list O at most once, and leave the list O at most once. Therefore at most $2 \cdot |\Sigma|$ crossing points are computed in total.

We can now complete the proof of this lemma. In order to compute the approximation p_2, we simply run Algorithm 1 for each location $l \in L$, which takes $O(|L| \cdot |\Sigma| \log |\Sigma|)$ time. Finally, we must account for the time taken to compute the approximation p_1, which takes $O(|\mathcal{M}|)$ time, as argued in Theorem 11. Therefore, we can compute p_2 in time $O(|\mathcal{M}| + |L| \cdot |\Sigma| \log |\Sigma|)$. □

Since the ε-step error for double ε-nets is bounded by ε^3, we can apply Lemma 5 to conclude that the global error is bounded by $\varepsilon^3 \cdot \frac{T}{\varepsilon} = \varepsilon^2 T$. Therefore, if we want to compute f with a precision of π, we should choose $\varepsilon \approx \sqrt{\frac{\pi}{T}}$, which gives $\frac{T}{\varepsilon} \approx \frac{T^{1.5}}{\sqrt{\pi}}$ distinct intervals.

Theorem 18. *For a normed Markov game $\mathcal{M}$ we can approximate the time-bounded reachability, construct π optimal memoryless strategies for both players, and determine the quality of these strategies with precision π in time $O(|\mathcal{M}| \cdot T \cdot \sqrt{\frac{T}{\pi}} + |L| \cdot T \cdot \sqrt{\frac{T}{\pi}} \cdot |\Sigma| \log |\Sigma|)$.*

Proof. Lemma 12 gives the step error for double ε-nets to be $\frac{2}{3}\varepsilon^3$. Since there are $\frac{T}{\varepsilon}$ intervals, Lemma 5 implies that the global error of double ε-nets is $\frac{2}{3}\varepsilon^3 \cdot \frac{T}{\varepsilon} = \frac{2}{3}\varepsilon^2 \cdot T$. In order to achieve a precision of π, we must select an ε that satisfies $\frac{2}{3}\varepsilon^2 \cdot T = \pi$. Therefore, we choose $\varepsilon = \sqrt{\frac{3\pi}{2T}}$, which gives $T \cdot \sqrt{\frac{2T}{3\pi}}$ intervals.

The cost of computing each interval is given by Lemma 17 as $O(|\mathcal{M}| + |L| \cdot |\Sigma| \cdot \log |\Sigma|)$, and there are $T \cdot \sqrt{\frac{T}{3\pi}}$ intervals overall, which gives the claimed complexity of $O(|\mathcal{M}| \cdot T \cdot \sqrt{\frac{T}{\pi}} + |L| \cdot T \cdot \sqrt{\frac{T}{\pi}} \cdot |\Sigma| \log |\Sigma|)$. □

4.4. Triple ε-nets and beyond

4.4.1. The approximation function

The techniques used to construct the approximation p_2 from the approximation p_1 can be generalised. This is because the only property of p_1 that is used in the proof of Lemma 12 is the fact that it is a piecewise polynomial function that approximates f. Therefore, we can inductively define a sequence of approximations p_k as follows:

$$-\dot{p}_k(l,\tau) = \operatorname*{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{R}(l,a,l') \cdot \big(p_{k-1}(l',\tau) - p_{k-1}(l,\tau)\big). \tag{17}$$

We can repeat the arguments from the previous sections to obtain our error bounds. The following lemma is a generalisation of Lemma 12.

Lemma 19. *For every $k > 1$, if we have $\mathcal{E}(k,\varepsilon) \leq c \cdot \varepsilon^{k+1}$, then we have $\mathcal{E}(k+1,\varepsilon) \leq \frac{2}{k+2} \cdot c \cdot \varepsilon^{k+2}$.*

Proof. The inductive hypothesis implies that $\|p_k(t-\tau) - f^t_{p_{k+1}(t)}(t-\tau)\| \leq c \cdot \tau^{k+1}$ for every $\tau \in [0,\varepsilon]$. Therefore, for every pair of locations $l, l' \in L$ and every $\tau \in [t-\varepsilon, t]$ we have:

$$|(p_k(l',t-\tau) - p_k(l,t-\tau)) - (f^t_{p_{k+1}(t)}(l',t-\tau) - f^t_{p_{k+1}(t)}(l,t-\tau))| \leq 2 \cdot c \cdot \tau^{k+1}.$$

Since we are dealing with normed Markov games, we have $\sum_{l' \in L} \mathbf{R}(l,a,l') = 1$ for every location $l \in L$ and every action $a \in A(l)$. Therefore, we also have for every action a:

$$\left| \sum_{l' \in L} \mathbf{R}(l,a,l') \cdot \big(p_k(l',t-\tau) - p_k(l,t-\tau)\big) - \sum_{l' \in L} \mathbf{R}(l,a,l') \cdot (f^t_{p_{k+1}(t)}(l',t-\tau) - f^t_{p_{k+1}(t)}(l,t-\tau)) \right| \leq 2 \cdot c \cdot \tau^{k+1}.$$

This implies that $\|\dot{p}_k(t-\tau) - \dot{f}^t_{p_{k+1}(t)}(t-\tau)\| \leq 2 \cdot c\tau^{k+1}$.

We can obtain the claimed result by integrating over this difference:

$$\mathcal{E}(k+1,\tau) = \int_0^\tau \|\dot{p}_k(t-\tau) - \dot{f}^t_{p_{k+1}(t)}(t-\tau)\| \leq \frac{2}{k+2} \cdot c \cdot \tau^{k+2}.$$

Therefore, the total amount of error incurred by p_{k+1} in $[t-\varepsilon, t]$ is at most $\frac{2}{k+2} \cdot c \cdot \varepsilon^{k+2}$. □

4.4.2. Strategies

As before, we can construct strategies for both of the players. We will give the prove only for the reachability player, because the proof for the safety player is entirely symmetric. We begin by defining the approximation g_k, which gives the time-bounded reachability probability when the reachability player follows the actions chosen by p_k. If a_l^τ is the action that maximises Equation (17) at the location l for the time point $\tau \in [t-\varepsilon, t]$ then we define $g_k(l,\tau)$ as:

$$-\dot{g_k}(l,\tau) = \sum_{l' \in L} \mathbf{Q}(l, a_l^\tau, l') \cdot g_k(l',\tau) \qquad \text{if } l \in L_r, \tag{18}$$

$$-\dot{g_k}(l,\tau) = \min_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g_k(l',\tau) \qquad \text{if } l \in L_s. \tag{19}$$

Our approach to proving error bounds for g_k follows the approach that we used in the proof of Lemma 13. We will consider the following class of strategies: play the action chosen by p_k for the first i transitions, and then play the action chosen by p_1 for the remainder of the interval. We will denote the reachability probability obtained by this strategy as g_k^i, and we will denote the error of this strategy as $\mathcal{E}_s^i(k,\varepsilon) := \|g_k^i(t-\varepsilon) - f^t_{p_2(t)}(t-\varepsilon)\|$. Clearly, as i approaches infinity, we have that g_k^i approaches g_k, and $\mathcal{E}_s^i(k,\varepsilon)$ approaches $\mathcal{E}_s(k,\varepsilon)$. Therefore, if a bound can be established on $\mathcal{E}_s^i(k,\varepsilon)$ for all i, then that bound also holds for $\mathcal{E}_s(k,\varepsilon)$.

We have by assumption that $\mathcal{E}(k,\varepsilon) \leq c \cdot \varepsilon^{k+1}$ and $\mathcal{E}_s(k,\varepsilon) \leq d \cdot \varepsilon^{k+1}$, and our goal is to prove that $\mathcal{E}_s(k+1,\varepsilon) \leq \frac{8c+3d}{k+2} \cdot \varepsilon^{k+2}$. We will prove error bounds on g_{k+1}^i by induction. The following lemma considers the base case, where $i = 1$. In other words, it considers the strategy that plays the action chosen by p_{k+1} for the first transition, and then plays the action chosen by p_k for the rest of the interval.

Lemma 20. *If $\varepsilon \leq 1$, $\mathcal{E}(k,\varepsilon) \leq c \cdot \varepsilon^{k+1}$, and $\mathcal{E}_s(k,\varepsilon) \leq d \cdot \varepsilon^{k+1}$, then we have $\mathcal{E}_s^1(k+1,\varepsilon) \leq \frac{4c+3d}{k+2} \cdot \varepsilon^{k+2}$.*

Proof. Suppose that the first discrete transition occurs at time $t-\tau$, where $\tau \in [0,\varepsilon]$. Let l be a location belonging to the reachability player, and let $a_l^{t-\tau}$ be the action that maximises p_{k+1} at time $t-\tau$. By definition, we know that the probability of moving to a location l' is given by $\mathbf{R}(l, a_l^{t-\tau}, l')$, and we know that the time-bounded reachability probabilities for each state l' are given by $g_k(l', t-\tau)$. Therefore, the outcome of choosing $a_l^{t-\tau}$ at time $t-\tau$ is $\sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') g_k(l', t-\tau)$. If a^* is an action that would be chosen by $f^t_{p_{k+1}(t)}$ at time $t-\tau$, then we have the following bounds:

$$\begin{aligned}
&\left| \sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') g_k(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f^t_{p_{k+1}(t)}(l', t-\tau) \right| \\
\leq &\left| \sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') p_k(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f^t_{p_{k+1}(t)}(l', t-\tau) \right| \\
&+ c \cdot \tau^{k+1} + d \cdot \tau^{k+1} \\
\leq &\, 4 \cdot c \cdot \tau^{k+1} + d \cdot \tau^{k+1}.
\end{aligned}$$

The first inequality follows from the bounds given for $\mathcal{E}(k,\varepsilon)$ and $\mathcal{E}_s(k,\varepsilon)$. The second inequality follows from the bounds given for $\mathcal{E}(k,\varepsilon)$ and Lemma 14.

Now suppose that l is a location belonging to the safety player. Since the reachability player will follow p_k during the interval $[t-\tau, t]$, we know that the safety player will choose an action a^g that minimises:

$$\min_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g_k(l', t-\tau).$$

If a^* is the action chosen by f at time $t-\tau$, then the following inequality is a consequence of Lemma 14:

$$\left| \sum_{l' \in L} \mathbf{R}(l, a^g, l') g_k(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f^t_{p_{k+1}(t)}(l', t-\tau) \right| \leq 3 \cdot d \cdot \tau^{k+1}.$$

So far we have proved that the total amount of error made by g_k^1 when the first transition occurs at time $t-\tau$ is at most $(4c+3d) \cdot \tau^{k+1}$. To obtain error bounds for g_{k+1}^1 over the entire interval $[t-\varepsilon, t]$, we consider the probability that the first transition actually occurs at time $t-\tau$:

$$\mathcal{E}_s^1(k+1,\varepsilon) \leq \int_0^\varepsilon e^{\tau-\varepsilon}(4c+3d) \cdot \tau^{k+1}\, d\tau \leq \int_0^\varepsilon (4c+3d) \cdot \tau^{k+1} d\tau = \frac{4c+3d}{k+2} \varepsilon^{k+2}.$$

This completes the proof. □

Lemma 21. *If* $\mathcal{E}_s^i(k+1,\varepsilon) \leq \frac{8c+3d}{k+2} \cdot \varepsilon^{k+2}$ *for some* k *and* $\mathcal{E}(k,\varepsilon) \leq c \cdot \varepsilon^{k+1}$, *then* $\mathcal{E}_s^{i+1}(k+1,\varepsilon) \leq \frac{8c+3d}{k+2} \cdot \varepsilon^{k+2}$.

Proof. The structure of this proof is similar to the proof of Lemma 20, however, we must account for the fact that g_{k+1}^{i+1} follows g_{k+1}^i after the first transition rather than g_k.

Suppose that we play the strategy for g_{k+1}^{i+1}, and that the first discrete transition occurs at time $t-\tau$, where $\tau \in [0,\varepsilon]$. Let l be a location belonging to the reachability player, and let $a_l^{t-\tau}$ be the action that maximises p_k at time $t-\tau$. If a^* is an action that would be chosen by $f_{p_2(t)}^t$ at time $t-\tau$, then we have the following bounds:

$$\begin{aligned}
&\left|\sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') g_{k+1}^i(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f_{p_{k+1}(t)}^t(l', t-\tau)\right| \\
&\leq \left|\sum_{l' \in L} \mathbf{R}(l, a_l^{t-\tau}, l') p_k(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f_{p_{k+1}(t)}^t(l', t-\tau)\right| \\
&\quad + c \cdot \tau^{k+1} + (4c+3d) \cdot \tau^{k+2} \\
&\leq 4c \cdot \tau^{k+1} + \frac{8c+3d}{k+1} \cdot \tau^{k+2} \\
&\leq (8c+3d) \cdot \tau^{k+1}.
\end{aligned}$$

The first inequality follows from the inductive hypothesis, which gives bounds on how far g_{k+1}^i is from $f_{p_{k+1}(t)}^t$, and from the assumption about $\mathcal{E}(k,\varepsilon)$. The second inequality follows from our assumption on $\mathcal{E}(k,\varepsilon)$ and Lemma 14, and the final inequality follows from the fact that $\tau \leq 1$ and $k > 2$.

Now suppose that the location l belongs to the safety player. Let a^g be an action that minimises:

$$\min_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l, a, l') \cdot g_{k+1}^i(l', t-\tau).$$

If a^* is the action chosen by f at time $t-\tau$, then our assumption about $\mathcal{E}_s^i(k+1,\varepsilon)$ and Lemma 14 imply:

$$\begin{aligned}
&\left|\sum_{l' \in L} \mathbf{R}(l, a^g, l') g_{k+1}^{i+1}(l', t-\tau) - \sum_{l' \in L} \mathbf{R}(l, a^*, l') f_{p_{k+1}(t)}^t(l', t-\tau)\right| \\
&\leq \frac{24c+9d}{k+2} \cdot \tau^{k+2} \\
&\leq (8c+3d) \cdot \tau^{k+1}.
\end{aligned}$$

The first inequality follows from the inductive hypothesis and Lemma 14, and the second inequality follows from the fact that $\tau \leq 1$ and $k > 2$.

To obtain error bounds for g_2^{k+1} over the entire interval $[t-\varepsilon, t]$, we consider the probability that the first transition actually occurs at time $t-\tau$:

$$\begin{aligned}
\mathcal{E}_s^{i+1}(k+1,\varepsilon) &\leq \int_0^\varepsilon e^{\tau-\varepsilon}(8c+3d) \cdot \tau^{k+1}\, d\tau \\
&\leq \int_0^\varepsilon (8c+3d) \cdot \tau^{k+1}\, d\tau = \frac{8c+3d}{k+2} \cdot \varepsilon^{k+2}.
\end{aligned}$$

This completes the proof. □

Our two lemmas together imply that $\mathcal{E}_s^i(k+1,\varepsilon) \leq \frac{8c+3d}{k+2} \cdot \varepsilon^{k+2}$ for all i, and hence we can conclude that $\mathcal{E}_s(k+1,\varepsilon) \leq \frac{8c+3d}{k+2} \cdot \varepsilon^{k+2}$. This gives us the following Lemma.

Lemma 22. *For every* $k > 2$, *if we have that* $\mathcal{E}_s(k,\varepsilon) \leq d \cdot \varepsilon^{k+1}$, *then we have that* $\mathcal{E}_s(k+1,\varepsilon) \leq \frac{8c+3d}{k+2} \cdot \varepsilon^{k+2}$.

4.4.3. The approximation algorithm

Computing the accuracies of Lemmas 19 and 22 explicitly for the first four levels of ε-nets gives:

k	1	2	3	4	...
$\mathcal{E}(k,\varepsilon)$	ε^2	$\frac{2}{3}\varepsilon^3$	$\frac{1}{3}\varepsilon^4$	$\frac{2}{15}\varepsilon^5$	...
$\mathcal{E}_s(k,\varepsilon)$	$2\varepsilon^2$	$2\varepsilon^3$	$\frac{17}{6}\varepsilon^4$	$\frac{67}{30}\varepsilon^5$	...

We can also compute, for a given precision π, the value of ε that should be used in order to achieve an accuracy of π with ε-nets of level k.

Lemma 23. *To obtain a precision π with an ε-net of level k, we choose $\varepsilon \approx \sqrt[k]{\frac{\pi}{T}}$, resulting in $\frac{T}{\varepsilon} \approx T\sqrt[k]{\frac{T}{\pi}}$ steps.*

Proof. Lemma 22 implies that the step error of using a k-net is $\mathcal{E}(k,\varepsilon) \leq c \cdot \varepsilon^{k+1}$ for some small constant $c < 1$. Since we have $\frac{T}{\varepsilon}$ many intervals, Lemma 5 implies that the global error is $T \cdot \varepsilon^k$. Therefore, to obtain a precision of π we must choose $\varepsilon = \sqrt[k]{\frac{\pi}{T}}$. □

Unfortunately, the cost of computing ε-nets of level k becomes increasingly prohibitive as k increases. To see why, we first give a property of the functions p_k. Recall that p_2 is a piecewise quadratic function. It is not too difficult to see how this generalises to the approximations p_k.

Lemma 24. *The approximation p_k is piecewise polynomial with degree less than or equal to k.*

Proof. We will prove this claim by induction. For the base case, we have by definition that p_1 is a linear function over the interval $[t-\varepsilon, t]$. For the inductive step, assume that we have proved that p_{k-1} is piecewise polynomial with degree at most $k-1$. From this, we have that $\sum_{l' \in L} \mathbf{Q}(l,a,l') \cdot p_{k-1}$ is a piecewise polynomial function with degree at most $k-1$ for every action a, and therefore $\mathsf{opt}_{a \in \Sigma(l)} \sum_{l' \in L} \mathbf{Q}(l,a,l') p_{k-1}(l',\cdot)$ is also a piecewise polynomial function with degree at most $k-1$. Since $\dot{p}_k$ is a piecewise polynomial function of degree at most $k-1$, we have that p_k is a piecewise polynomial of degree at most k. □

Although these functions are well-behaved in the sense that they are always piecewise polynomial, the number of pieces can grow exponentially in the worst case. The following lemma describes this bound.

Lemma 25. *If p_{k-1} has c pieces in the interval $[t-\varepsilon, t]$, then p_k has at most $\frac{1}{2} \cdot c \cdot k \cdot |L| \cdot |\Sigma|^2$ pieces in the interval $[t-\varepsilon, t]$.*

Proof. Let $[t-\tau_1, t-\tau_2]$ be the boundaries of a piece in p_{k-1}. Since there can be at most $|\Sigma(l)|$ actions at l, we have that optimum computed by Equation (17) must choose from at most $|\Sigma(l)|$ distinct polynomials of degree $k-1$. Since each pair of polynomials can intersect at most k times, we have that p_k can have at most $k \cdot \frac{1}{2}|\Sigma(l)|^2$ pieces for each location l in the interval $[t-\tau_1, t-\tau_2]$. Since p_{k-1} has c pieces in the interval $[t-\varepsilon, t]$, and $|L|$ locations, we have that p_k can have at most $\frac{1}{2} \cdot c \cdot k \cdot |L| \cdot |\Sigma|^2$ during this interval. □

The upper bound given above is quite coarse, and we would be surprised if it were found to be tight. Moreover, we do not believe that the number of pieces will grow anywhere close to this bound in practice. This is because it is rare, in our experience, for optimal strategies to change their decision many times within a small time interval.

However, there is a more significant issue that makes ε-nets become impractical as k increases. In order to compute the approximation p_k, we must be able to compute the roots of polynomials with degree $k-1$. Since we can only directly compute the roots of polynomials up to a degree of 4 and for higher degrees we have to approximate the roots, it is unclear whether approximations beyond p_4 or p_5 are useful.

Once again it is possible to provide a smart algorithm that uses sorting in order to find the switching points, which gives the following bounds on the cost of computing the functions p_3 and p_4.

Theorem 26. *For a normed Markov game $\mathcal{M}$ we can construct π optimal memoryless strategies for both players and determine the quality of these strategies with precision π in time $O(|L|^2 \cdot \sqrt[3]{\frac{T}{\pi}} \cdot T \cdot |\Sigma|^4 \log|\Sigma|)$ when using triple ε-nets, and in time $O(|L|^3 \cdot \sqrt[4]{\frac{T}{\pi}} \cdot T \cdot |\Sigma|^6 \log|\Sigma|)$ when using quadruple ε-nets.*

Proof. We know that double ε-nets can produce at most $|\Sigma|$ pieces per interval, and therefore Lemma 25 implies that triple ε-nets can produce at most $\frac{3}{2} \cdot |L| \cdot |\Sigma|^3$ pieces per interval, and there are $T \cdot \sqrt[3]{\frac{T}{\pi}}$ many intervals. To compute each piece, we must sort $O(|\Sigma|)$ crossing points, which takes time $O(|\Sigma| \log|\Sigma|)$. Therefore, the total amount of time required to compute p_3 is $O(T \cdot \sqrt[3]{\frac{T}{\pi}} \cdot |L| \cdot |\Sigma|^4 \cdot \log|\Sigma|)$.

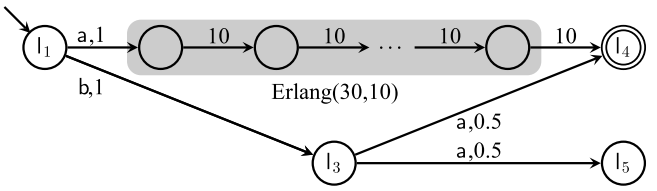

Fig. 3. A CTMDP offering the choice between a long chain of fast transition and a slower path that looses some probability mass in l_5.

Table 2
Running times of two experiments with our prototype implementation for different precisions. The running times of MRMC are shown for comparison.

Precision	Erlang model			Game model	
	MRMC [11]	Double-nets	Triple-nets	Double-nets	Triple-nets
10^{-4}	0.05 s	0.04 s	0.01 s	0.29 s	0.06 s
10^{-5}	0.20 s	0.10 s	0.02 s	0.93 s	0.13 s
10^{-6}	1.32 s	0.32 s	0.03 s	2.94 s	0.28 s
10^{-7}	8 s	0.98 s	0.06 s	9.35 s	0.60 s
10^{-8}	475 s	3.11 s	0.12 s	29.21 s	1.29 s
10^{-9}	–	9.91 s	0.27 s	94 s	2.78 s
10^{-10}	–	31.24 s	0.58 s	299 s	6.05 s

For quadruple ε-nets, Lemma 25 implies that there will be at most $6 \cdot |L|^2 \cdot |\Sigma|^5$ pieces per interval, and at most $T \cdot \sqrt[3]{\frac{T}{\pi}}$ many intervals. Therefore, we can repeat our argument for triple ε-nets to obtain an algorithm that runs in time $O(T \cdot \sqrt[4]{\frac{T}{\pi}} \cdot |L|^2 \cdot |\Sigma|^6 \cdot \log|\Sigma|)$ □

From these estimations, it is not clear if triple and quadruple ε-nets are mainly of theoretical interest, or if they will be useful in practice. While their dependency on ε is clearly reduced, the worst case complexity bounds measured in the sizes of Σ and L provided by Theorem 26 are high. They do, however, arise purely from the upper bound on the number of switching points given in Lemma 25. Thus, if the number of switching points that occur is small, these techniques become very attractive.

It is our belief that the number of switching points will be small in practice, and our experiments from the following section give evidence to support this assumption.

5. Experimental results

In order to test the practicability of our algorithms, we have implemented both double and triple ε-nets. We have evaluated these algorithms on some examples.

Firstly, we have tested our algorithms on the Erlang-example (see Fig. 3) presented in [11] and [16]. We have chosen to consider the same parameters used in those papers: we study the maximal probability to reach location l_4 from l_1 within 7 time units. Since this example is a CTMDP, we were able to compare our results with the Markov Reward Model Checker (MRMC) [11] implementation, which includes an implementation of the techniques proposed by Buchholz and Schulz. The results of our experiments are shown in Table 2. The MRMC implementation was unable to provide results for precisions beyond $1.86 \cdot 10^{-9}$. For the Erlang examples we found that, as the desired precision increases, our algorithms draw further ahead of the current techniques. The most interesting outcome of these experiments is the validation of triple ε-nets for practical use. While the worst case theoretical bounds arising from Lemma 25 indicated that the cost of computing the approximation for each interval may become prohibitive, these results show that the worst case does not seem to play a role in practice. In fact, we found that the number of switching points summed over all intervals and locations never exceeded 2 in this example.

Second, we test our algorithms on continuous-time Markov games. We use the model depicted in Fig. 4, consisting of two chains of locations $l_1, l_2, \ldots, l_{100}$ and $l'_1, l'_2, \ldots, l'_{100}$ that are controlled by the maximising player and the minimising player, respectively. This example is designed to produce a large number of switching points. In every location l_i of the maximising player, there is the choice between the short but slow route along the chain of maximising locations, and the slightly longer route which uses the minimising player's locations. If very little time remains, the maximising player prefers to take the slower actions, as fewer transitions are required to reach the goal using these actions. The maximiser also prefers these actions when a large amount of time remains. However, between these two extremes, there is a time interval in which it is advantageous for the maximising player to take the action with rate 3. A similar situation occurs for the minimising player, and this leads to a large number of points where the players change their strategy.

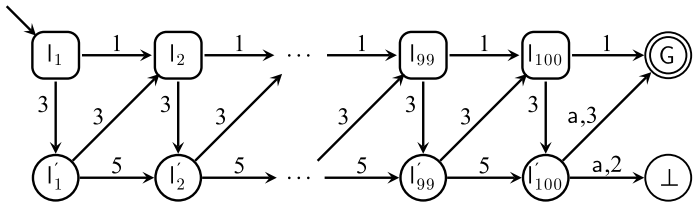

Fig. 4. A CTMG with many switching points.

Table 3
Running times for the working cluster example where the first part we have $t = 1$ and the second part we have $t = 100$. Time is given in seconds, and – denotes timeout of an hour.

Precision	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
Single-nets	73.91	704.69	–	–	–	–	–
Double-nets	0.22	0.84	1.93	5.72	18.74	56.13	167.28
Triple-nets	0.14	0.19	0.35	0.71	1.53	3.06	6.06
Double-nets	45.85	152.19	472.59	1740.15	–	–	–
Triple-nets	7.23	15.12	32.53	69.34	150.30	325.90	688.45

Our results (see Tables 2 and 3) on Markov games demonstrate that our algorithms are capable of solving games of non-trivial size in practice. For the workstation cluster (see Table 3), we once again find that triple ε-nets provide a substantial performance increase over double ε-nets, and that the worst case bounds given by Lemma 25 do not seem to occur. Double ε-nets found 297 points where the strategy changed during an interval, and triple ε-nets found 684 such points. Hence, the $|L||\Sigma|^2$ factor given in Lemma 25 does not seem to arise here.

Finally, we also consider the case study of a fault-tolerant workstation cluster [34]. The workstation cluster has two sub-clusters, which are connected via a backbone. Each cluster has a central switch, which connects to the backbone, and also N workstations in a star-topology. Components in the system may fail with given rates. There is a single repair unit for the cluster, which can repair failing components. Moreover, it is only capable of repairing one failed component at a time, with the repair rate provided, which depends on the component. In case multiple components are down, the repair unit is allocated to repair one broken unit. The model is thus a CTMDP, where the nondeterminism stems from the allocation of the repair unit. Details can be found in [35].

We say that the system provides premium service whenever at least N workstations in the entire system are operational, and moreover, these workstations should be connected. We consider the probability to reach non-premium service within time t. We consider the model for $N = 16$ workstations. The system has 10,130 states. Table 3 provides the running time (in seconds) for precision values ranging from 10^{-4} to 10^{-10} for time bounds $t = 1$ and $t = 100$, respectively. Again, the calculation based on triple nets is considerably faster, especially for higher precision. When increasing the time bound t, the running time grows proportionally to t. When increasing the precision by a factor of 10, the runtime increases by a factor of around 2.2. This is to be expected, as the third root of 10 is approximately 2.15.

In this example, the discretisation technique used in MRMC [15] is slower, but comparable, with our single nets configuration. The uniformisation technique in [11] is faster than our technique on this example. A reason for this is that there are only three states in the model where the decision of how to allocate the repair unit makes a difference; these are the positions where the backbone and one (or both) of the switches are down. Thus, whereas our approach iterates through all actions for finding the optimal, the uniformisation based heuristics in [11] ignore this and treats this model almost like a CTMC. This inspires an interesting future work of dynamically adjustable the interval length, especially for models with only few switching points.

6. Conclusion

In this article we have proposed efficient approximation algorithms for solving the time-bounded reachability problem for CTMDPs. Existing approaches based on discretisation or uniformisation provide an accuracy of $O(\varepsilon^2)$ for each ε length interval. The bottleneck of these approaches is the high number of discretised steps needed, which is reciprocal of the required precision. We have proposed a sequence of approximations achieving $O(\varepsilon^3)$, $O(\varepsilon^4)$ and $O(\varepsilon^5)$ accuracies, allowing us to reduce the number of steps considerably, which is also confirmed by our experimental results.

Furthermore, we would like to extend our approach to other properties such as finite-horizon expected rewards [6], or reward bounded reachability problem [36].

Acknowledgments

This work was partly supported by the Engineering and Physical Science Research Council (EPSRC) through the grants EP/H046623/1 ‘Synthesis and Verification in Markov Game Structures’ and EP/M027287/1 ‘Energy Efficient Control’, the

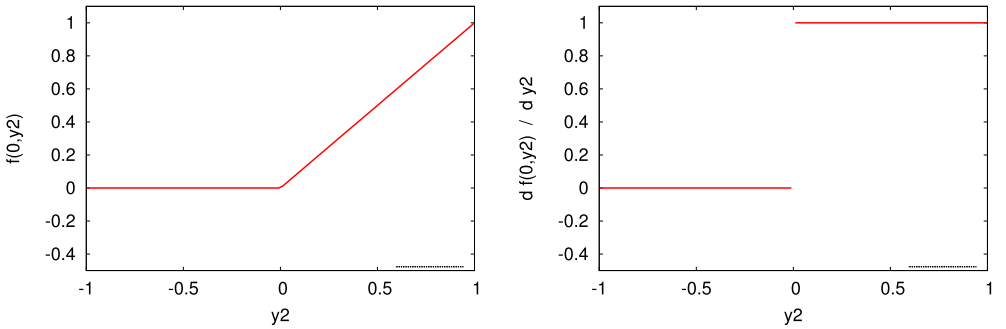

Fig. 5. The left graph shows the variation of the first projection of the functor f (that is, of $\max\{0, y2\}$) in the second argument (that is, of $y2$). The right graph shows the respective partial derivation in direction $y2$ on for the values on this line. In the origin (0,0) itself, f is clearly not differentiable.

Transregional Collaborative Research Center "Automatic Verification and Analysis of Complex Systems" (SFB/TR 14 AVACS) of the DFG, the National Natural Science Foundation of China (Grant No. 61532019, 61472473), the CDZ project CAP (GZ 1023), and the CAS/SAFEA International Partnership Program for Creative Research Teams.

Appendix A. Collocation methods for CTMDPs

In the numerical evaluations of CTMCs, numerical methods like collocation techniques play an important role. We briefly discuss the limits of these methods when applied to CTMDPs, and in particular we will focus on the Runge–Kutta method. On sufficiently smooth functions, the Runge–Kutta methods obtain very high precision. For example, the RK4 method obtains a step error of $O(\varepsilon^5)$ for each interval of length ε. However, these results critically depend on the degree of smoothness of the functor describing the dynamics. To obtain this precision, the functor needs to be four times continuously differentiable [37, p. 157]. Unfortunately, the Bellman equations describing CTMDPs do not have this property. In fact, the functor defined by the Bellman equations is not even once continuously differentiable due to the inf and/or sup operators they contain.

In this appendix we demonstrate on a simple example that the reduced precision is not merely a problem in the proof, but that the precision deteriorates once an inf or sup operator is introduced. We then show that the effect observed in the simple example can also be observed in the Bellman equations on the example CTMDP from Fig. 1.

A.1. A simplified example

Maximisation (or minimisation) in the functor that describes the dynamics of the system results in functors with limited smoothness, which breaks the proof of the precision of Runge–Kutta method (incl. Collocation techniques). In order to demonstrate that this is not only a technicality in the proof of the quality of Runge–Kutta methods, we show on a simple example how the step precision deteriorates.

Using the notation of http://en.wikipedia.org/wiki/Runge-Kutta_methods (but dropping the dependency in t, that is $y' = f(y)$), consider a function $y = (y1, y2)$ with dynamics — the functor f — defined by $y1' = \max\{0, y2\}$ and $y2' = 1$. Note that the functor f is not partially differentiable at $(0, 0)$ in the second argument, see Fig. 5. The solution of the ODE in the time interval $[0, 2]$ is given in Fig. 6.

Let us study the effect this has on the Runge–Kutta method on an interval of size h, using the start value $y_n = (0, -\frac{1}{2}h)$. Applying RK4, we get

- $k_1 = f\big((0, -\frac{1}{2}h)\big) = (0, 1)$,
- $k_2 = f\big((0, 0)\big) = (0, 1)$,
- $k_3 = f\big((0, 0)\big) = (0, 1)$,
- $k_4 = f\big((0, \frac{1}{2}h)\big) = (\frac{1}{2}h, 1)$, and
- $y_{n+1} = y_n + \frac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4) = (h^2/12, h/2)$.

The analytical evaluation, however, provides $(\mathbf{h^2/8}, \mathbf{h/2})$ which differs from the provided result by $\frac{\mathbf{1}}{\mathbf{24}}\mathbf{h^2}$ in the first projection. Note that the expected difference in the first projection is in the order of h^2 if we place the point where max is in balance (the 'swapping point' that is related to the point where optimal strategies change) uniformly at random at some point in the interval.

Still, one could object that we had to vary both the left and the right border of the interval. But note that, if we take the initial value $y(0) = (0, -1) = y_0$, seek $y(2)$, and cut the interval into $2n + 1$ pieces of equal length $h = \frac{2}{2n+1}$, then this is the middle interval. (This family contains interval lengths of arbitrary small size.)

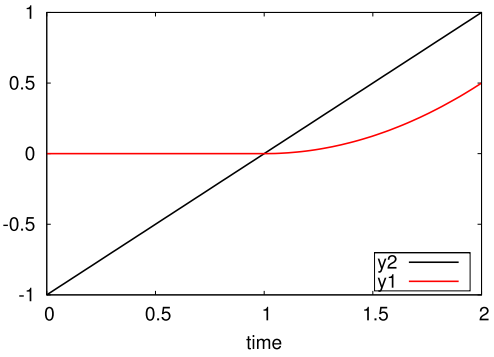

Fig. 6. $y1$ and $y2$ from the solution of the ODE of the simplified example in the time interval $[0, 2]$.

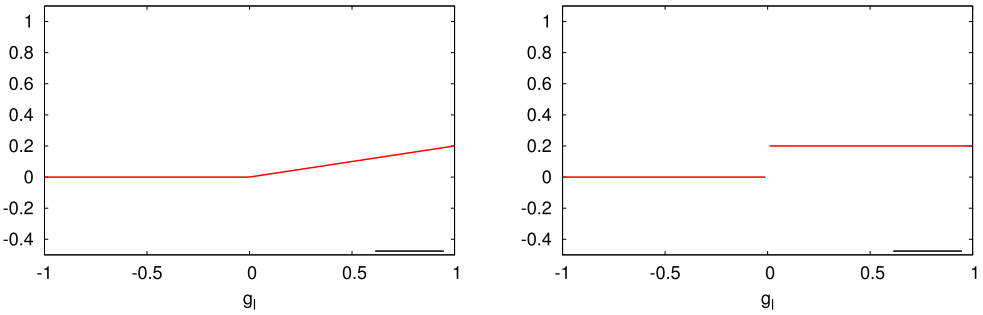

Fig. 7. The left graph shows the variation of the first projection of the functor F in the argument g_l at the origin. The right graph shows the respective partial derivation in direction g_l on for the values on this line. In the origin $\vec{0}$ itself, F is clearly not differentiable.

A.2. Connection to the Bellman equations

The first step when applying this to the Bellman equations is to convince ourselves that their functor $F = \bigotimes_{l \in L} F_l$ with $F_l = opt \sum \dots$ is indeed not differentiable. We use g for the arguments of F in order to distinguish it from the solution f, where $f(t)$ is the time-bounded reachability probability at time t.

For this, we simply re-use the example from Fig. 1. The particular functor F is not differentiable in the origin: varying F_{l_R} in the direction g_l provides the function shown in Fig. 7, showing that F_{l_R} is not differentiable in the origin.

(Due to the direction of the evaluation, this is the 'rightmost' point where the optimal strategy changes.)

Again, differentiating $F_{l_R}(f(t_1))$ in the direction g_l provides a non-differentiable function. (In fact, a function similar to the function shown in Fig. 7, but with adjusted x-axis.)

An analytical argument with e functions is more involved than with the toy example from the previous subsection. However, when the mesh length (or: interval size) goes towards 0, then the ascent of the e functions is almost constant throughout the mesh/interval. In the limit, the effect is the same and the error in the order of h^2.

References

[1] P. Buchholz, I. Schulz, Numerical analysis of continuous time Markov decision processes over finite horizons, Comput. Oper. Res. 38 (3) (2011) 651–659.
[2] J. Fearnley, M. Rabe, S. Schewe, L. Zhang, Efficient approximation of optimal control for continuous-time Markov games, in: Proc. of FSTTCS, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2011, pp. 399–410.
[3] M.L. Puterman, Markov Decision Processes: Discrete Stochastic Dynamic Programming, Wiley-Interscience, 1994.
[4] R. Bellman, Dynamic Programming, Princeton University Press, 1957.
[5] B.L. Miller, Finite state continuous time Markov decision processes with an infinite planning horizon, J. Math. Anal. Appl. 22 (3) (1968) 552–569.
[6] B.L. Miller, Finite state continuous time Markov decision processes with a finite planning horizon, SIAM J. Control 6 (2) (1968) 266–280.
[7] H. Garavel, R. Mateescu, F. Lang, W. Serwe, CADP 2006: a toolbox for the construction and analysis of distributed processes, in: Proc. of CAV, 2007, pp. 158–163.
[8] N. Coste, H. Hermanns, E. Lantreibecq, W. Serwe, Towards performance prediction of compositional models in industrial gals designs, in: Proc. of CAV, 2009, pp. 204–218.
[9] T.A. Henzinger, M. Mateescu, V. Wolf, Sliding window abstraction for infinite Markov chains, in: Proc. of CAV, 2009, pp. 337–352.
[10] M. Bozzano, A. Cimatti, M. Roveri, J.-P. Katoen, V.Y. Nguyen, T. Noll, Verification and performance evaluation of AADL models, in: ESEC/SIGSOFT FSE, 2009, pp. 285–286.
[11] P. Buchholz, E.M. Hahn, H. Hermanns, L. Zhang, Model checking algorithms for CTMDPs, in: Proc. of CAV, 2011, pp. 225–242.
[12] M.R. Neuhäußer, M. Stoelinga, J.-P. Katoen, Delayed nondeterminism in continuous-time Markov decision processes, in: Proc. of FOSSACS, 2009, pp. 364–379.
[13] M. Rabe, S. Schewe, Finite optimal control for time-bounded reachability in continuous-time Markov games and CTMDPs, Acta Inform. (2011) 291–315.

[14] T. Chen, T. Han, J.-P. Katoen, A. Mereacre, Reachability probabilities in Markovian timed automata, in: CDC-ECE, IEEE, 2011, pp. 7075–7080.
[15] M.R. Neuhäußer, L. Zhang, Time-bounded reachability probabilities in continuous-time Markov decision processes, in: Proc. of QEST, 2010, pp. 209–218.
[16] L. Zhang, M.R. Neuhäußer, Model checking interactive Markov chains, in: Proc. of TACAS, 2010, pp. 53–68.
[17] A. Martin-Löfs, Optimal control of a continuous-time Markov chain with periodic transition probabilities, Oper. Res. 15 (5) (1967) 872–881.
[18] X.P. Guo, O. Hernández-Lerma, T. Prieto-Rumeau, A survey of recent results on continuous-time Markov decision processes, Top 14 (2) (2006) 177–261.
[19] C. Baier, H. Hermanns, J.-P. Katoen, B. Haverkort, Efficient computation of time-bounded reachability probabilities in uniform continuous-time Markov decision processes, Theor. Comput. Sci. 345 (1) (2005) 2–26.
[20] H. Hermanns, S. Johr, Uniformity by construction in the analysis of nondeterministic stochastic systems, in: Proc. of DSN, IEEE Computer Society, 2007, pp. 718–728.
[21] E. Böde, M. Herbstritt, H. Hermanns, S. Johr, T. Peikenkamp, R. Pulungan, J. Rakow, R. Wimmer, B. Becker, Compositional dependability evaluation for statemate, IEEE Trans. Softw. Eng. 35 (2) (2009) 274–292.
[22] T. Brázdil, V. Forejt, J. Krcál, J. Kretínský, A. Kucera, Continuous-time stochastic games with time-bounded reachability, in: Proc. of FSTTCS, 2009, pp. 61–72.
[23] T. Brázdil, V. Forejt, J. Krcál, J. Kretínský, A. Kucera, Continuous-time stochastic games with time-bounded reachability, Inf. Comput. 224 (2013) 46–70.
[24] M.N. Rabe, S. Schewe, Optimal time-abstract schedulers for CTMDPs and continuous-time Markov games, Theor. Comput. Sci. 467 (2013) 53–67.
[25] N. Wolovick, S. Johr, A characterization of meaningful schedulers for continuous-time Markov decision processes, in: Proceedings of FORMATS'06, 2006, pp. 352–367.
[26] A. Mereacre, Verification of continuous-space stochastic systems, Ph.D. thesis, RWTH Aachen University, 2011.
[27] C. Eisentraut, H. Hermanns, L. Zhang, On probabilistic automata in continuous time, in: Proc. of LICS, IEEE Computer Society, 2010, pp. 342–351.
[28] Y. Deng, M. Hennessy, On the semantics of Markov automata, Inf. Comput. 222 (2013) 139–168.
[29] H. Hermanns, Interactive Markov Chains: The Quest for Quantified Quality, Lecture Notes in Computer Science, vol. 2428, Springer, 2002.
[30] H. Hatefi, H. Hermanns, Model checking algorithms for Markov automata, ECEASST 53.
[31] D. Guck, H. Hatefi, H. Hermanns, J.-P. Katoen, M. Timmer, Modelling, reduction and analysis of Markov automata, in: Proc. of QEST, in: Lecture Notes in Computer Science, vol. 8054, Springer, 2013, pp. 55–71.
[32] Y. Butkova, H. Hatefi, H. Hermanns, J. Krčál, Optimal continuous time Markov decisions, in: Automated Technology for Verification and Analysis, Springer International Publishing, 2015, pp. 166–182.
[33] G. Birkhoff, G.-C. Rota, Ordinary Differential Equations, 3rd ed., John Wiley and Sons, New York, NY, 1978.
[34] B.R. Haverkort, H. Hermanns, J.-P. Katoen, On the use of model checking techniques for dependability evaluation, in: SRDS, 2000, pp. 228–237.
[35] J.-P. Katoen, I.S. Zapreev, E.M. Hahn, H. Hermanns, D.N. Jansen, The ins and outs of the probabilistic model checker MRMC, in: QEST, 2009, pp. 167–176.
[36] H. Fu, Maximal cost-bounded reachability probability on continuous-time Markov decision processes, CoRR, arXiv:1310.2514.
[37] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations I: Nonstiff Problems, 2nd revised ed., Springer-Verlag, New York, 1993.