Probability in the Engineering and Informational Sciences (2022), **36**:4 999–1026
doi:10.1017/S0269964821000206

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

Temporal concatenation for Markov decision processes

Ruiyang Song[1] and Kuang Xu[2]

[1] Department of Electrical Engineering, Stanford University, Stanford, CA, USA. E-mail: ruiyangs@stanford.edu
[2] Graduate School of Business, Stanford University, Stanford, CA, USA. E-mail: kuangxu@stanford.edu.

Keywords: Markov decision process, Stochastic dynamic programming

Abstract

We propose and analyze a temporal concatenation heuristic for solving large-scale finite-horizon Markov decision processes (MDP), which divides the MDP into smaller sub-problems along the time horizon and generates an overall solution by simply concatenating the optimal solutions from these sub-problems. As a "black box" architecture, temporal concatenation works with a wide range of existing MDP algorithms. Our main results characterize the regret of temporal concatenation compared to the optimal solution. We provide upper bounds for general MDP instances, as well as a family of MDP instances in which the upper bounds are shown to be tight. Together, our results demonstrate temporal concatenation's potential of substantial speed-up at the expense of some performance degradation.

1. Introduction

We are interested in devising computationally efficient architectures for solving finite-horizon Markov decision processes (MDP), a popular framework for modeling multi-stage decision-making problems [1,22] with a wide range of applications from scheduling in data and call centers [12] to energy management with intermittent renewable resources [13]. In an MDP, at each stage, an agent makes a decision based on the state of the system, which leads to an instantaneous reward and the state is updated accordingly; the agent aims to find an optimal policy that maximizes the total expected rewards over the time horizon. While finding efficient algorithms for solving MDPs has long been an active area of research (see [17,20] for a survey), we will, however, take a different approach. Instead of creating new algorithms from scratch, we ask how to design *architectures* that leverage existing MDP algorithms as "black boxes" in creative ways, in order to harness additional performance gains.

As a first step in this direction, we propose the temporal concatenation heuristic, which takes a divide-and-conquer approach along the time axis: for an MDP with horizon $\{0, \ldots, T-1\}$, we divide the original problem instance ($\mathcal{I}_0$) over the horizon into two sub-instances: $\{0, \ldots, T/2-1\}$ ($\mathcal{I}_1$) and $\{T/2, \ldots, T-1\}$ ($\mathcal{I}_2$), respectively. Temporal concatenation then evokes an MDP algorithm, one that takes as input an MDP instance and outputs an optimal policy, to find the optimal policies π_1^* and π_2^* for the two sub-instances $\mathcal{I}_1$ and $\mathcal{I}_2$, separately. Finally, temporal concatenation outputs a policy, π_{TC}, for the original MDP by simply concatenating π_1^* and π_2^*: run π_1^* during the first half of the horizon, and π_2^* the second.[1]

In a nutshell, temporal concatenation is intended as a simple "black box" architecture to substantially speed up existing MDP algorithms, at the expense of potentially minor performance degradation. First, acceleration comes from the fact that the optimal policies for the sub-instances can be derived entirely in parallel. In particular, a classical MDP problem can be solved by conventional methods such as the value

[1] More generally, a similar temporal concatenation procedure can be performed over K sub-instances, with $K \geq 2$. Our theoretical analysis will focus on the case of $K = 2$ because it captures the majority of structural insights.

iteration, which has a time complexity growing linearly with the horizon, T. By applying the temporal concatenation architecture in this set-up, the computing time can, in principle, be reduced by half. This speed-up from parallelism can be significant if the original MDP algorithm's run-time is sufficiently long. In addition, we point out that a parallelism architecture is well suited for modern machine learning systems where the instance of a large-scale problem may be stored in separate servers to start with [8,18,19] . Moreover, temporal concatenation can speed up computation even more significantly if the MDP algorithm in question admits a run-time that scales super-linearly in the horizon. Typically, these algorithms suffer a worse dependence on T in exchange for a more favorable scaling in the size of the state and action spaces; see for instance, the complexity of a linear-programming-based algorithm in [26] that scales as $O(T^4)$, and that of the stochastic primal-dual method proposed by Chen & Wang [7], which scales as $O(T^6)$.

While the computational benefit from using temporal concatenation is evident, the quality of its solution is not: by solving two sub-instances independently, it could be overly short-sighted and lead to strictly sub-optimal MDP policies. Therefore, our theoretical results will focus on addressing the following question:

How good is the policy generated by temporal concatenation, π_{TC}, compared to the optimal policy to the original problem, π^*?

Preview of main results. On the positive side, we provide sufficient conditions under which the performance gap between π_{TC} and π^* is *small*. Specifically, we establish upper bounds to show that the performance gap is bounded by a function that depends linearly on an MDP's *diameter* (a measure that reflects the ease with which the agent can traverse the state space) but *independent* from the horizon, T. Conversely, we provide lower bounds by showing that, for *any* finite diameter, there exist MDP instances for which the upper bounds are tight for all large T.

Organization. The remainder of this paper is organized as follows. In Section 2, we formally introduce the problem formulation and performance metrics. In Section 3, we summarize the main results and contrast our approach to the extant literature. Section 4 provides several examples of MDP instances, including one that is motivated by the application of dynamic energy management with on-site storage. We also provide simulation results that substantiate the theoretical results and illustrate the run-time reduction obtained by running the temporal concatenation heuristic on a multi-core PC. Section 5 concludes the paper.

Notation. We will denote by $[n]$ the set of integers $\{0, 1, \ldots, n-1\}$, $n \in \mathbb{N}$. We will use $\delta_{\mathrm{TV}}(\mu, \nu)$ to denote the total variation distance between two distributions μ and ν: $\delta_{\mathrm{TV}}(\mu, \nu) = \frac{1}{2}\sum_s |\mu(s) - \nu(s)| = \sum_{s:\mu(s)\geq\nu(s)}(\mu(s) - \nu(s))$. For a sequence $\{a_i\}_{i\in\mathbb{N}}$, and $s, t \in \mathbb{N}$, $s \leq t$, we use $s \to t$ to denote the set $\{s, s+1, \ldots, t\}$, and use $a_{s\to t}$ to denote the sub-sequence $\{a_s, a_{s+1}, \ldots, a_{t-1}, a_t\}$. Similarly, for some $\mathcal{S} \subseteq \mathbb{N}$, $a_{\mathcal{S}}$ denotes the set $\{a_i : i \in \mathcal{S}\}$. For $x \in \mathbb{R}$, we will denote by $(x)^+$ and $(x)^-$ the positive and negative portion of x, respectively: $(x)^+ = \max\{x, 0\}$ and $(x)^- = \max\{-x, 0\}$. For $c, d \in \mathbb{R}$ with $c \leq d$, define $x_{[c,d]}$ to be the projection of x onto the interval $[c, d]$, i.e., $x_{[c,d]} \triangleq \mathbb{I}(x < c)c + \mathbb{I}(c \leq x \leq d)x + \mathbb{I}(x > d)d$, where $\mathbb{I}(\cdot)$ is the indicator function.

2. Problem formulation and performance metric

2.1. System set-up

We consider a discrete-time Markov decision process with a finite time horizon $[T]$, state space $\mathcal{S}$, and action set $\mathcal{A}$. The decision maker chooses at each step $t \in [T]$ an action, $a_t \in \mathcal{A}$. We will assume that $\mathcal{A}$ and $\mathcal{S}$ stay fixed, and hence omit them from our notation when appropriate. The state of the system at time t is denoted by S_t. The initial state S_0 is drawn from some probability distribution μ_0, and the state evolution depends on the present state as well as the action chosen:

$$S_{t+1} = p_t(a_t, S_t, Y_t^S), \quad t \in [T]. \tag{1}$$

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

The Y_t^S's are i.i.d. uniform random variables over a finite set $\mathcal{Y}^S$, capturing the randomness in the state transition. The collection $\{p_t\}_{t\in[T]}$ is the set of (deterministic) transition functions. The decision maker receives a *reward* at each time slot t, $R_t(a_t, S_t, Y_t^R)$, which depends on the present state, action, and some i.i.d. idiosyncratic random variables taking values in a finite set, $Y_t^R \in \mathcal{Y}^R$, with a fixed distribution. We refer to $\{R_t\}_{t\in[T]}$ as the set of reward functions. We assume that the rewards are nonnegative and bounded from above by a constant, $\bar{r} \in \mathbb{R}_+$.[2] The decision maker's behavior is described by a *policy* $\pi(\cdot)$, such that $a_t = \pi(t, S_t, Y^P)$, $t \in [T]$. In other words, the policy chooses an action based on the current state, and some idiosyncratic randomization Y^P, which, without loss of generality, can be thought of as a uniform random variable over $[0, 1]$.

An MDP as described above is specified by the triple, $(R_{[T]}, p_{[T]}, T)$, which we will refer to as a *problem instance*. We will refer to the original horizon-T MDP problem instance as the *original instance*, denoted by $\mathcal{I}_0 \triangleq (R_{[T]}, p_{[T]}, T)$. For an instance $\mathcal{I}_0$, policy π, and initial distribution μ_0, the *total expected reward* is defined by

$$V(\mathcal{I}_0, \pi, \mu_0) = \mathbb{E}^{\pi}_{S_0\sim\mu_0}\left[\sum_{t=0}^{T-1} R_t\left(a_t, S_t, Y_t^R\right)\right], \tag{2}$$

where the expectation is taken with respect to all the randomness in the system, and the actions are chosen according to π. A policy π is *optimal* if it attains the maximum total expected reward for all initial distributions, μ_0.

2.2. Temporal concatenation

We now define the main object of study, the temporal concatenation heuristic. An *MDP algorithm*, denoted by ALG$(\cdot)$, takes as input a problem instance and outputs the optimal policy, π^*, for that instance. As such, the notion of an MDP algorithm captures the "functionality" of an algorithm that is used to compute an optimal policy, but abstracts away the inner working of the algorithm, effectively treating it as a "black box." By this definition, we have that

$$\pi^* = \text{ALG}(\mathcal{I}_0). \tag{3}$$

Definition 1 (Temporal concatenation)**.** *For an original instance $\mathcal{I}_0$, denote by $\mathcal{I}_1$ and $\mathcal{I}_2$ the sub-instances generated by partitioning $\mathcal{I}_0$ in half along the time horizon:*

$$\mathcal{I}_1 \triangleq \left(R_{0\to T/2-1}, p_{0\to T/2-1}, T/2\right), \quad \textit{and} \quad \mathcal{I}_2 \triangleq \left(R_{T/2\to T-1}, p_{T/2\to T-1}, T/2\right), \tag{4}$$

and by π_1^* and π_2^* their corresponding optimal policies:

$$\pi_1^* \triangleq ALG(\mathcal{I}_1), \quad \textit{and} \quad \pi_2^* \triangleq ALG(\mathcal{I}_2). \tag{5}$$

The temporal concatenation heuristic generates a policy, π_{TC}, by temporally concatenating optimal solutions for $\mathcal{I}_1$ and $\mathcal{I}_2$, π_1^* and π_2^*, i.e.,

$$\pi_{TC}(t, S_t, Y^P) = \begin{cases} \pi_1^*(t, S_t, Y_1^P), & 0 \le t \le \dfrac{T}{2} - 1, \\ \pi_2^*(t - T/2, S_t, Y_2^P), & \dfrac{T}{2} \le t \le T - 1, \end{cases} \tag{6}$$

where Y_1^P and Y_2^P are two independent uniform random variables generated from Y^P as follows: express $Y^P \in [0, 1]$ as an infinite binary sequence, and set Y_1^P and Y_2^P to be the sub-sequence corresponding to all odd and even elements in the binary sequence, respectively.

[2] Note that the results in this paper would be unchanged if the reward function R_t were shifted by a constant. In particular, the general case in which $R_t(\cdot, \cdot, \cdot) \in [r_{\min}, r_{\max}]$ for $-\infty < r_{\min} \le r_{\max} < \infty$, is equivalent to having $R_t \in [0, \bar{r}]$ where $\bar{r} = r_{\max} - r_{\min}$. Throughout this paper, we let $R_t(\cdot, \cdot, \cdot) \in [0, \bar{r}]$ for simplicity of notation unless otherwise specified.

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

2.3. Performance metric

The following definition of regret is our main metric, which measures how the expected reward of the policy π_{TC} deviates from the optimal policy π^*:

Definition 2 (Regret of temporal concatenation)**.** *For an original instance $\mathcal{I}_0$ and initial distribution μ_0, the regret of temporal concatenation, or regret for short, is defined by:*

$$\Delta(\mathcal{I}_0, \mu_0) \triangleq V(\mathcal{I}_0, \pi^*, \mu_0) - V(\mathcal{I}_0, \pi_{TC}, \mu_0), \tag{7}$$

where π^* is an optimal policy for $\mathcal{I}_0$, and π_{TC} is defined in Definition 1.

Note that the above definition differs from the conventional notion of regret of the online learning literature and reinforcement learning. For instance, regret in [4] is due to not having complete information of the MDP in hindsight, whereas in our case, it is due to the intrinsic sub-optimality from dividing an original MDP instance into smaller sub-problems and solving each separately.

3. Main results

We present our main results in this section. The first result, Theorem 1, provides an upper bound on the regret of temporal concatenation in an MDP, which does not depend on the length of the horizon, T. Instead, the regret is shown to be related to a notion of *diameter* of the MDP, which we define below.

The diameter captures how easy it is for the decision maker to reach different state distributions. Let $\mathcal{P}$ be the collection of all distributions over $\mathcal{S}$. We will denote by μ_t^π the state distribution at time t under policy π. Starting at time $t_0 \geq 0$, for two distributions $\mu, \nu \in \mathcal{P}$, we say that ν is *ϵ-reachable* from μ in t steps for some $\epsilon \in [0, 1]$, if there exists a policy π such that under π and with the distribution of S_t at time t_0 being μ, we have that

$$\delta_{\mathrm{TV}}(\mu_{t_0+t}^\pi, \nu) \leq \epsilon. \tag{8}$$

Denote by $\mathcal{P}_\epsilon^{t_0}(\mu, t)$ the set of all distributions that are ϵ-reachable from μ in t steps starting from time $t_0 \in [T - t]$. We have the following definition of diameter.

Definition 3 (ϵ-Diameter)**.** *For an MDP instance $\mathcal{I}_0$ with horizon $[T]$ and transition functions $\{p_t\}_{t\in[T]}$, we define the ϵ-diameter as the least number of steps with which, starting from any time step, all possible distributions in $\mathcal{P}$ are ϵ-reachable from one another:*

$$\tau_\epsilon(\mathcal{I}_0) \triangleq \inf\{t \geq 0 : \nu' \in \mathcal{P}_\epsilon^{t_0}(\nu, t) \text{ for all } \nu, \nu' \in \mathcal{P} \text{ and all } t_0 \in [T - t]\}. \tag{9}$$

Note that since τ_ϵ captures the hardness of traversing the state space by applying feasible actions, it will depend on the sizes of the state and action spaces in general. We have the following theorem.

Theorem 1 (Upper bound on regret of temporal concatenation)**.** *Fix an original instance $\mathcal{I}_0$ with horizon $[T]$, and an initial distribution μ_0. If there exists $\epsilon \geq 0$ such that $\tau_\epsilon(\mathcal{I}_0) \leq T/2$, then the regret of temporal concatenation satisfies:*

$$\Delta(\mathcal{I}_0, \mu_0) \leq \frac{\bar{r}\tau_\epsilon(\mathcal{I}_0)}{1 - \epsilon}, \tag{10}$$

where $\bar{r}$ is the maximum reward in a given time slot. In particular, if $\tau_0(\mathcal{I}_0) \leq T/2$, then the above inequality implies that

$$\Delta(\mathcal{I}_0, \mu_0) \leq \bar{r}\tau_0(\mathcal{I}_0). \tag{11}$$

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

A direct implication of the above theorem is that, for problems that admit a moderate ϵ-diameter for some $\epsilon \in [0, 1)$, temporal concatenation produces a near-optimal policy *regardless* of the length of the horizon, T, thus making the heuristic especially appealing for problems with a relatively large horizon.

It is also worth noting that while the original temporal concatenation algorithm requires an optimal policy to be used in each of the two sub-instances, it is easy to substitute these optimal policies with sub-optimal ones with a bounded regret and obtain similar regret bounds to those in Theorem 1. In particular, suppose we use in each sub-instance a policy whose total expected reward over the sub-instance is at most δ less than that of an optimal policy starting from any initial state, then it follows the resulting regret bounds would be those in Theorem 1 with an additional 2δ additive factor.

The next result provides a lower bound that demonstrates that a small diameter is also *necessary* for temporal concatenation to perform well, in a worst-case sense. We look at MDP instances with a bounded 0-diameter, $\tau_0(\mathcal{I}_0)$. In Theorem 2, we show that, for any $d_0 \in \mathbb{N}$, there exists an instance with a 0-diameter equal to d_0 such that the performance regret is essentially $\bar{r}d_0$ for any horizon $T > 2d_0 + 2$. This result implies that the upper bound in (11) of Theorem 1 is tight in a worst-case instance.

Theorem 2 (Lower bound on regret of temporal concatenation)**.** *Fix $\bar{r} \in \mathbb{R}_+$, $\sigma \in (0, \bar{r}/2)$, and integer $d_0 \geq 5$. Then there exists an MDP instance $\mathcal{I}_0$ with maximum per-slot reward $\bar{r}$, finite 0-diameter $\tau_0(\mathcal{I}_0) = d_0$, and an initial distribution μ_0, such that for any $T > 2d_0 + 2$, the regret satisfies*

$$\Delta(\mathcal{I}_0, \mu_0) = (\tau_0(\mathcal{I}_0) - 2)\bar{r} - \sigma. \tag{12}$$

The proofs of Theorems 1 and 2 will be presented in Appendix A.

Theorem 1 shows that for MDP instances that admit a bounded ϵ-diameter, the regret of temporal concatenation is bounded from above by a value independent of the time horizon T. This is encouraging, since it would suggest that the quality of approximation afforded by our heuristic does not degrade over longer time horizons. On the other hand, Theorem 2 shows that the regret could be very large if the diameter is large, though it would appear from our proof that the "bad" examples we know so far tend to be fairly pathological. We accompany these findings by examining in Section 4 a number of specific MDP models. Our theoretical and simulation results there suggest that temporal concatenation at least performs reasonably well for several such "average" instances.

3.1. Related work

Our method is related to the literature on MDP decomposition methods, which aim to overcome the so-called curse of dimensionality by breaking down the original MDP with a large state space into sub-problems with smaller state spaces. Hierarchical MDP algorithms utilize hierarchical structures to decompose the state space and action space and transform the original problem into a collection of sequential sub-tasks [9,10,20,21] . The method in [24] decomposes the problem into parallel sub-tasks that can be computed simultaneously, where each sub-task is an MDP with the same state and action space, but with different reward functions. Steimle *et al*. [23] adopts a decomposition via mixture models. Finally, Ie *et al*. [14] leverages decomposition in the Q-function by exploiting combinatorial structures of the recommender systems.

While our approach also works by dividing the original MDP instance into smaller sub-problems, there is a number of crucial features that differentiate our approach. Firstly, our method focuses on decomposing the problem along the time axis rather than over the state space or action space, which is a more common approach in the literature. Secondly, the focus of temporal concatenation is to serve as a simple "black box" architecture, rather than a custom-made MDP algorithm. As such, each sub-problem can be solved by any MDP algorithm of the user's choice, and the procedure is very simple to implement and does not involve complex procedures to transform the structure of the original problem. Finally, as alluded to in the introduction, temporal concatenation lends itself easily to parallel processing, and thus

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

achieving speed-up that is not possible under a decomposition algorithm in which sub-problems still need to be solved in a sequential manner (cf. [24]).

Related to our approach in spirit is [16], which proposes a heuristic for finite-horizon MDPs by sequentially solving a series of smaller MDPs with increasing horizons, and the numerical results show that the heuristic provides good performance even if the process is terminated prematurely. However, no rigorous guarantees in terms of regret of this heuristic relative to the optimal policy were established.

Related notions of the diameter of an MDP have been used in the literature to capture the ease with which the system can transit between any pair of states in the state space. For instance, a diameter D^* is defined in [15] Definition 1 as

$$D^* \triangleq \max_{s,s' \in \mathcal{S},\, s \neq s'} \min_{\pi} \mathbb{E}^{\pi} \left[\min_{N \geq 1, S_N = s'} N \,\middle|\, S_0 = s \right]. \tag{13}$$

The diameter D^* has been used for analyzing the total regret of reinforcement learning algorithms (see [15,25] for example). In [15], the authors introduced a learning algorithm for MDP with total regret $O(D^*|\mathcal{S}|\sqrt{|\mathcal{A}|T})$. In [25], an improved upper bound for the total regret of MDP is introduced, which depends on the variance of the bias function defined in [25] Definition 2 but does not depend on D^*. In comparison, our definition of ϵ-diameter is different, and in some sense stronger. First, while D^* is the expected number of steps necessary to transit between any pair of states, ϵ-diameter corresponds to the number of steps required to traverse between any pair of *distributions* over the state space. Second, the definition of D^* implies the existence of a policy under which the target state is reached with no more than D^* time steps, while for τ_ϵ, we require a policy such that the target distribution is achieved after exactly τ_ϵ time steps (a total variation distance no greater than ϵ is allowed). Notably, the lower bound in Theorem 2 shows that our notion of diameter cannot be weakened when applied to the analysis of temporal concatenation, thus suggesting that our formulation reveals structural properties of the MDP distinct from those in the extant literature. We will further discuss the connection between the ϵ-diameter and the D^* diameter in Appendix B.

4. Examples and illustrative applications

In this section, we discuss several examples to illustrate the properties of the ϵ-diameter and corroborate the theoretical results in Section 3. In Section 4.1, we introduce the deterministic graph traversal (DGT) problems, a family of MDP instances with finite 0-diameter and noiseless transitions. In Section 4.2, we introduce the ξ-stochastic graph traversal (ξ-SGT) problems, which is a generalization of the DGT with stochastic transitions. In Section 4.3, we present a model of dynamic energy management with storage, which is an illustrative example of the ξ-SGT family. We also present simulation results for the deterministic graph traversal models in Section 4.4 to explore the average-case scaling behavior of the regret within this family. In Section 4.5, we provide additional simulation examples of this model to illustrate the run-time reduction from using the temporal concatenation heuristic. In Section 4.6, we present simulation results of a more popular family of MDP instances known as the Generalized Average-Reward Non-stationary Environment Test-bench (GARNET) model.

4.1. Deterministic graph traversal problems

In this subsection, we introduce the *deterministic graph traversal (DGT)* problems, a family of MDP instances with finite 0-diameter. Let $\mathcal{G}_{\text{csl}}$ be the set of all strongly connected graphs that include at least one self-loop. A DGT instance denoted by $\mathcal{I}_G$ has a time-homogeneous deterministic transition function, i.e. $p_t = p$ for all t, which can be described by a strongly connected directed graph $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}_{\text{csl}}$ where at least one vertex in G has a self-loop. Here, $\mathcal{V}$, $\mathcal{E}$ are the collections of vertices and edges of G, respectively. In other words, for any $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}_{\text{csl}}$, there exists a vertex $v_i \in \mathcal{V}$ such that the self-loop edge exists, i.e., $e_{ii} \in \mathcal{E}$. In a DGT instance, once the current state S_t and action a_t are given,

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

the next state S_{t+1} is determined. We formally define a DGT instance with state space $\mathcal{S}$, action space $\mathcal{A}$, and transition function p as follows.

Definition 4 (Deterministic graph traversal instance)**.** *Let $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}_{csl}$. A DGT instance depicted by G, $\mathcal{I}_G$, is an MDP instance whose state space and transition function satisfy:*

(1) Each state $i \in \mathcal{S}$ corresponds to a vertex $v_i \in \mathcal{V}$.
(2) The state transition is deterministic and can be described by the edges in $\mathcal{E}$.

In particular, for $i, j \in \mathcal{S}$, an edge $e_{ij} \in \mathcal{E}$ implies the existence of an action $a_{ij} \in \mathcal{A}$ such that starting from state i, the system will deterministically go to state j once the agent takes action a_{ij}, i.e., $j = p(a_{ij}, i, Y_t^S)$ with probability 1 for all t.

Note that we are not imposing additional restrictions on the reward functions in the definition of DGT instances. As an example, the MDP instance we construct for proving Theorem 2 is a special case that belongs to this family (see Appendix A.2).

Now we study the ϵ-diameter of DGT instances. We first briefly recall the definition of the classical diameter of a directed graph $G = (\mathcal{V}, \mathcal{E})$, denoted by $d_c(G)$. For any two vertices of the graph, $i, j \in \mathcal{V}$, let $d_G(i, j)$ be the distance between them on graph G, which is defined as the length of the shortest path from i to j. Here, a path is a sequence of distinct vertices such that each two consecutive vertices are connected by an edge in $\mathcal{E}$. The classical diameter is the maximum taken over all pairwise distances, i.e., $d_c(G) = \max_{i,j \in \mathcal{V}} d_G(i, j)$. For a strongly connected graph G, each pairwise distance is finite, in which case $d_c(G)$ is also finite. Further, it is not difficult to verify that for a strongly connected graph, the classical diameter is at most $|\mathcal{V}| - 1$. Literature has shown that $d_c(G)$ can be computed within at most $O(|\mathcal{V}|^3)$ time using classical algorithms such as the breadth first search (see [3] for example).

Recall that $d_c(G) < \infty$ because G is strongly connected. In the following lemma, we prove that DGT instances indeed have a finite 0-diameter, which is closely related to the classical diameter of the corresponding graph, $d_c(G)$.

Lemma 1 (From ϵ-diameter to classical graph diameter)**.** *For a DGT instance, $\mathcal{I}_G$, based on a graph $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}_{csl}$, we have*

$$d_c(G) \leq \tau_0(\mathcal{I}_G) \leq 2d_c(G). \tag{14}$$

With Lemma 1, we have shown that the 0-diameter of $\mathcal{I}_G$ is finite and bounded between $d_c(G)$ and $2d_c(G)$. It follows that for any $\epsilon > 0$, the ϵ-diameter of $\mathcal{I}_G$ satisfies

$$\tau_\epsilon(\mathcal{I}_G) \leq \tau_0(\mathcal{I}_G) \leq 2d_c(G). \tag{15}$$

The proof of this lemma is given in Appendix C.1.

4.2. ξ-stochastic graph traversal problems

In Section 4.1, we introduced a family of MDP instances with finite 0-diameter τ_0 where the transition is deterministic. However, this model can not capture the stochasticity in many real-world applications. To this end, we study in this subsection a generalization of the DGT family where transitions can be impacted by stochastic shocks. As a result, we will see concrete examples of instances where the ϵ-diameter is finite for $\epsilon > 0$, even though the 0-diameter may be infinite.

Specifically, we consider the *ξ-stochastic graph traversal (ξ-SGT)* problems with state space $\mathcal{S}$, action space $\mathcal{A}$, and time-homogeneous transition function p, defined as follows.

Definition 5 (ξ-Stochastic graph traversal instance)**.** *Fix $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}_{csl}$, and $\xi \in (0, 1)$. For $i \in \mathcal{V}$, define the neighbor set of i as $\mathcal{N}_i = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$. A ξ-SGT instance based on (G, ξ), $\mathcal{I}_G^\xi$, is an*

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

MDP instance whose state space and transition function satisfy:

(1) Each state $i \in \mathcal{S}$ corresponds to a vertex $v_i \in \mathcal{V}$.

(2) The state transition is stochastic and can be described by the edges in $\mathcal{E}$. In particular, for $i, j \in \mathcal{S}$ such that the edge $e_{ij} \in \mathcal{E}$ exists, there exists an action $a_{ij} \in \mathcal{A}$ under which starting from state i, the system goes to state j with a probability at least $1-\xi$, i.e.,

$$p(a_{ij}, i, Y_t^S) = \begin{cases} j & \text{with a probability at least } 1-\xi \\ Z_{t,i,j} & \text{otherwise} \end{cases}, \tag{16}$$

where $Z_{t,i,j}$ is a random variable that takes values in $\mathcal{N}_i \backslash \{j\}$ if $|\mathcal{N}_i| \geq 2$, and $Z_{t,i,j} = j$ if $|\mathcal{N}_i| = 1$.

From Definitions 4 and 5, we see that the ξ-SGT and DGT problems are closely related. For both of these families of MDP instances, the state space corresponds to the vertices of a directed graph, and the state transition function can be described by the edges in the same graph. The ξ-SGT instance can be regarded as a noisy version of the DGT instance, where the transition along the edges is perturbed by a random noise. The parameter ξ can be interpreted as the noise level. In a DGT instance based on G, the system can deterministically traverse the state space along the edges of G when appropriate actions are taken. In an ξ-SGT based on (G, ξ), however, when a proper action is chosen, the system will traverse along the "intended" edge with a probability at least $1-\xi$ but may be diverged to one of the other neighbors otherwise.

The following result connects the ϵ-diameter of the ξ-SGT instance, $\tau_\epsilon(\mathcal{I}_G^\xi)$, the 0-diameter of the DGT instance, $\tau_0(\mathcal{I}_G)$, and the classical diameter of the underlying graph, $d_c(G)$; the proof is given in Appendix C.2.

Lemma 2 (Diameter of ξ-SGT instances)**.** *Fix $G \in \mathcal{G}_{csl}$, and $\xi \in (0,1)$. Let $\mathcal{I}_G$ be the DGT instance characterized by G, and $\mathcal{I}_G^\xi$ the ξ-SGT instance described by (G, ξ). For $\epsilon \geq 1-(1-\xi)^{\tau_0(\mathcal{I}_G)}$, we have*

$$\tau_\epsilon(\mathcal{I}_G^\xi) \leq \tau_0(\mathcal{I}_G). \tag{17}$$

Combining the above with Lemma 1, we have

$$\tau_\epsilon(\mathcal{I}_G^\xi) \leq 2d_c(G) \tag{18}$$

for $\epsilon \geq 1-(1-\xi)^{2d_c(G)}$.

Lemma 2 implies the following communicating property of a ξ-SGT instance: when we are allowed a total variation distance ϵ and the noise level ξ is sufficiently small such that $\xi < 1-(1-\epsilon)^{1/\tau_0(\mathcal{I}_G)}$, we can traverse the state space in a ξ-SGT instance using no more than $\tau_0(\mathcal{I}_G)$ steps. Further, with Lemma 1, the 0-diameter of the DGT instance, $\mathcal{I}_G$, is bounded from above by two times the classical diameter of the corresponding graph G. This in turn implies that the ϵ-diameter is bounded from above by $2d_c(G)$.

While determining the closed-form 0-diameter of a general DGT instance remains a direction for future work, we have provided in this paper an example of DGT instance whose τ_0 can be derived explicitly. In particular, the MDP instance depicted by Figure 5 in Appendix A.2, which is designed for proving the lower bound in Theorem 2, is a DGT instance of $k+2$ states with $\tau_0(\mathcal{I}_G) = k+2$. For ξ-SGT instances corresponding to the graph in Figure 5, we can apply Lemma 2 and obtain $\tau_\epsilon(\mathcal{I}_G^\xi) \leq k+2$ for $\epsilon \geq 1-(1-\xi)^{k+2}$. If the 0-diameter of the DGT instance cannot be derived in closed form, we can apply the upper bound in Eq. (18) instead. This upper bound depends only on the classical diameter, which can be calculated for any directed graph by existing algorithms.

While Lemma 2 is a general result that holds for any ξ-SGT instance, we introduce another stronger characterization on the ϵ-diameter of ξ-SGT instances when the graph $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}_{\text{csl}}$ is undirected,

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

and each node of G has a noiseless self-loop, i.e., $(i,i) \in \mathcal{E}$, and $p(a_{ii}, i, Y_t^S) = i$ with probability one, for all $i \in \mathcal{S}$. We have the following lemma, which is proved in Appendix C.3.

Lemma 3. *Fix an undirected connected graph $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}_{csl}$ where each node has a noiseless self-loop, i.e., $(i,i) \in \mathcal{E}$ for all $i \in \mathcal{S}$, and $\xi \in (0, \frac{1}{2})$. Let $\mathcal{I}_G$ be the DGT instance characterized by G, and $\mathcal{I}_G^\xi$ be the ξ-SGT instance described by (G, ξ). Then for any $\epsilon \in (0,1)$, the ϵ-diameter of $\mathcal{I}_G^\xi$ satisfies that*

$$\tau_\epsilon(\mathcal{I}_G^\xi) \le \frac{d_c(G)}{1-2\xi} + \frac{f(d_c(G), \xi)}{\epsilon}, \tag{19}$$

with

$$f(d_c(G), \xi) = \frac{4\xi(1-\xi)}{(1-2\xi)^2}\left(2 + \sqrt{4 + \frac{(1-2\xi)d_c(G)}{\xi(1-\xi)}}\right), \tag{20}$$

where $d_c(G)$ is the classical diameter of graph G. Note that when $\xi \to 0$, Eq. (19) reduces to $\tau_\epsilon(\mathcal{I}_G) \le d_c(G)$.

Lemma 3 suggests that for a ξ-SGT instance with undirected G and noiseless self-loops for all nodes, there is an upper bound on the ϵ-diameter, which grows linearly in $\frac{1}{\epsilon}$. Moreover, we observe that the upper bound coincides with the classical diameter, $d_c(G)$, when the noise level ξ goes to 0, which corresponds to the result for DGT instances.

Note that the upper bound in Lemma 3 depends only on the classical diameter of the underlying graph, $d_c(G)$, and parameters ϵ, ξ, but not on $\tau_0(\mathcal{I}_G)$. Furthermore, in contrast to Lemma 2, in Lemma 3, the parameter ϵ can take any value in $(0,1)$. On the flip side, where the noise parameter ξ can take on any value in $(0,1)$ in Lemma 2, the result in Lemma 3 is restricted to the case where $\xi \in (0, 1/2)$. This restriction is due largely to the limitation of our analysis. Specifically, in the proof of Lemma 3, we employ a random walk-based argument. In each step, we move one step closer to a target state with probability $1-\xi$, and one step away from the target with probability ξ. Within this framework, we show that the process is able to reach a target node with high probability within a sufficiently large number of steps if the noise level $\xi < 1/2$, which leads to an upper bound on τ_ϵ. When the noise level ξ approaches or exceeds $1/2$, however, basic results from the theory of random walks show that the expected number of steps for a random walk to reach the target becomes infinite (see [11] for example). We are hopeful that improved analysis in a future work can help remedy this restriction and address the case where $\xi \ge 1/2$.

4.3. Dynamic energy management with storage

In Section 4.2, we introduced the ξ-SGT family of MDP instances, which is a stochastic variant of the DGT instance introduced in Section 4.1. In this subsection, we provide an illustrative application that can be modeled by the ξ-SGT family. We consider the following model of dynamic energy management with storage.

Consider an operator and a battery with B charging levels $\mathcal{S} = \{0, \ldots, B-1\}$, and a power parameter, $C \in \{1, \ldots, B-1\}$, representing the maximum units of charging and discharging within one time step. The state S_t corresponds to the battery level at time t. The transition function is given by

$$S_{t+1} = (S_t + \min\{a_t, Y_t^S\})_{[0,B-1]}, \tag{21}$$

where $(x)_{[a,b]}$ represents the projection of x onto the interval $[a,b]$. Here, Y_t^S is a nonnegative random variable representing the *on-site renewable generation* (e.g., wind or solar) at time t. We will assume that Y_t^S satisfies $\mathbb{P}(Y_t^S < C) = \beta$, and $\mathbb{P}(Y_t^S \ge C) = 1-\beta$. The value β is a noise level, such that $1-\beta$ corresponds to the probability that there is enough renewable generation for the decision maker to achieve maximum per-step charge, C.

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

The variable $a_t \in \{-C, \dots, C\}$ represents the *control* at time t: the decision maker may choose to sell the stored energy by setting $-C \le a_t < 0$, charge the battery by setting $0 < a_t \le C$, or hold the current battery level by setting $a_t = 0$. Any unused energy is stored in the battery, up to its capacity, $B - 1$. Note that when $a_t > 0$, the actual amount of energy charged to the battery is

$$a_t^C = \min\{B - 1 - S_t, (a_t)^+, Y_t^S\}. \tag{22}$$

When $a_t < 0$, the actual amount of energy sold is

$$a_t^S = \min\{(a_t)^-, S_t\}. \tag{23}$$

In other words, the charging process may be impacted by the random renewable generation, while the selling and holding actions are assumed to be noiseless in this model.

The goal of an operator is to maximize the expected total reward $V = \sum_{t=0}^{T-1} R_t(a_t, S_t, Y_t^R)$ for some reward function R_t, which takes values in $[r_{\min}, r_{\max}]$ for some $r_{\min}, r_{\max} \in \mathbb{R}$ with $r_{\max} - r_{\min} = \bar{r} > 0$. One example of the reward function can be the operator's net revenue, defined as the difference between the revenue generated from selling energy and the charging costs. In particular, the reward function can be expanded as

$$R_t(a_t, S_t, Y_t^R) = -a_t^C P_t^C + a_t^S P_t^S, \tag{24}$$

where P_t^C, P_t^S are the charging costs and the selling prices at time t, respectively. The prices P_t^C, P_t^S are bounded nonnegative random variables with mean p_t^C, p_t^S, respectively. At each time step, the agent plans the control a_t ahead when only the mean prices are available, but not the actual prices.

It is easy to verify that the dynamic energy management system depicted above is a ξ-SGT problem based on (G, ξ), where $\xi = \beta$ and G is a connected undirected graph with B vertices and a noiseless self-loop around each vertex. Each node in G has no more than $2C + 1$ edges. In particular, for any pair of states in the state space, $s, s' \in \{0, \dots, B - 1\}$, such that $0 < s' - s \le C$, we can choose an action $a = s' - s$ such that by taking action a, the state transitions from s to s' in one step with a probability at least $1 - \beta$. If $-C \le s' - s \le 0$, we can choose $a = s' - s$ such that by taking action a, the state goes from s to s' with probability one. Note also that with B battery levels and parameter C, we have

$$d_c(G) \le \frac{B}{C} + 1. \tag{25}$$

With these observations in mind, by applying Theorem 1 and Lemma 3, we have the following result that characterizes the regret of temporal concatenation in this family of problems. The proof is provided in Appendix C.4.

Theorem 3. *For a dynamic energy management system with B battery levels, power parameter $C \le B - 1$, and noise level β, the regret of temporal concatenation for any initial distribution μ_0 and horizon T is bounded from above as follows:*

$$\Delta(\mu_0, T) \le \bar{r}(\sqrt{\omega + \alpha} + \sqrt{\omega})^2. \tag{26}$$

where we use shorthands $\alpha := (B/C + 1)/(1 - 2\beta)$, $\omega := f(B/C + 1, \beta)$, with $f(\cdot, \cdot)$ defined in Eq. (20).

Figure 1 contains a numerical example for the right-hand side of (26), illustrating its relationship with the ratio B/C given fixed noise level β. To put the figure in context, let us look at the following illustrative example. Consider an energy-storage system with a total capacity of 36 MWh and hourly power rating of 9 MW [5]. Assume further that each time slot in the MDP amounts to approximately 10 min (e.g., the California ISO has a 5 or 15-min dispatch window for the real-time utility energy market [6]). This translates to $B = 36$, $C = 9/6 = 1.5$, and $B/C = 24$. With $\beta = 0.1$ and a normalized $\bar{r} = 1$, Theorem 3 would suggest that the total regret is bounded from above by 90, uniformly over all time horizons. Since

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

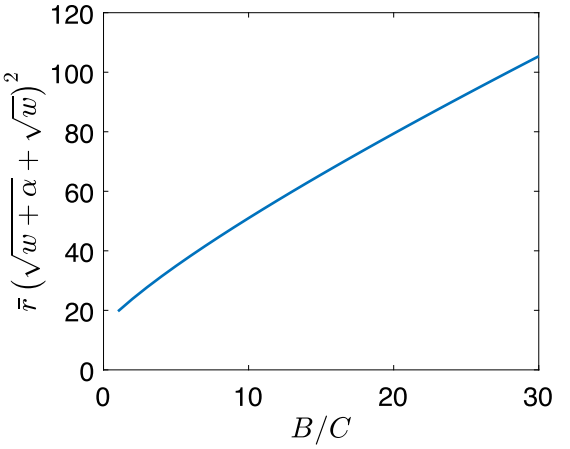

Figure 1. *An illustration of the upper bound in Theorem 3, with* $\beta = 0.1$, $\bar{r} = 1$, *and* B/C *ranging from 1 to 30.*

there are 144 time slots in a 24-h period, this suggests that the average regret of one step incurred by temporal concatenation is at most 62.5% over a one-day horizon, or 8.9% for a one-week horizon.

4.4. Simulation results for DGT instances

In this section, we provide numerical examples to illustrate the trajectory of the regret of temporal concatenation, which will allow us to investigate the degree to which the theoretical results in Section 3 hold in *"average" instances* with different diameters. We also explore the performance of a generalized temporal concatenation, which temporally concatenates the policies of K sub-instances, for $K \geq 2$.

We will consider the DGT instance based on a graph G with finite 0-diameter and deterministic state transition, as defined in Definition 4. Suppose the reward functions, $\{R_t\}_{t\in[T]}$, are also deterministic and depend only on the current state. In this case, each vertex in the graph G corresponds to a state of the MDP instance and is associated with a reward. In graph G, an edge from vertex i to vertex j means that the system can transition from state i to state j within one step when an appropriate action is taken.

We present a group of simulations where the transition function can be represented by a strongly connected directed graph with at least one self-loop.

4.4.1. Simulation set-up

We consider DGT instances depicted by directed graphs. This generative model allows us to control the diameter of the instance by varying the density of the randomly added edges. We randomly construct $N_D = 3 \times 10^4$ DGT instances with $|\mathcal{S}| = 200$ states. For each realization, the MDP has a deterministic transition function, which can be represented by $G_D^{(i)} \in \mathcal{G}_{\text{csl}}$, a strongly connected directed graph including at least one self-loop, for $i \in [N_D]$. Let $p_D^{(i)} = W_D^{(i)}/200$, where $W_D^{(i)} \overset{\text{i.i.d.}}{\sim} \text{Unif}(0, 1]$. We generate directed graphs $G_D^{(i)}$ by randomly adding edges to a ring, which is similar to the small-world model in [27]. Specifically, the graphs $G_D^{(i)}$ are independently generated using the following steps:

(i) construct a ring that connects all the states with edges $(1, 2), (2, 3), \ldots, (|\mathcal{S}| - 1, |\mathcal{S}|), (|\mathcal{S}|, 1)$;
(ii) add a self-loop around the vertex 1;
(iii) with probability $p_D^{(i)}$ add an edge (j, k) if there is currently no edge from j to k, for $j, k \in \mathcal{S}$. (If $j = k$ this will be a self-loop around j.)

In the ith realization, the reward associated with each node $R^{(i)}(j)$ is drawn uniformly at random from the set $\{1, 2, \ldots, 200\}$, for $j \in \mathcal{S}$, which implies that the maximal reward $\bar{r}^{(i)} \leq 200$. In Figure 2, we provide an example of $G_D^{(i)}$ with 20 nodes.

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

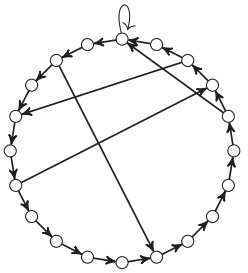

Figure 2. *An example of a strongly connected directed graph with at least one self-loop, $G_D^{(i)}$, with $n = 20$ nodes.*

Once an MDP instance is constructed, we compute the regret of temporal concatenation with a uniform initial state distribution $\mu_0 = (1/|\mathcal{S}|, \ldots, 1/|\mathcal{S}|)$ for different horizons T. Definition 1 can be easily generalized to temporal concatenation with K sub-instances for $K \geq 2$, which will be elaborated in the subsequent paragraph. For temporal concatenation with K sub-instances with $K = 2, 3, 4, 5$, we let T vary from K to 800. For each case, we run $N_D = 3 \times 10^4$ simulations and compute the classical diameter d_c of the graph in each realization. Let $\mathcal{N}_d = \{i : d_c(G_D^{(i)}) = d, i \in [N_D]\}$ be the collection of all graphs generated in the simulation with classical diameter d. For realizations with the same diameter d, we compute the (normalized) empirical average regret of temporal concatenation for different T, which can be expanded as

$$\widehat{\Delta}(d,T) \triangleq \frac{1}{|\mathcal{N}_d|} \sum_{i \in \mathcal{N}_d} \frac{1}{\bar{r}^{(i)}} \left(\mathbb{E}^{\pi^*} \left[\sum_{t=0}^{T-1} R^{(i)}(S_t^{(i)}) \right] - \mathbb{E}^{\pi_{\mathrm{TC}}} \left[\sum_{t=0}^{T-1} R^{(i)}(S_t^{(i)}) \right] \right). \tag{27}$$

Here, $S_t^{(i)}$ is the state at time t in the ith realization, $G_D^{(i)}$, while $R^{(i)}(j),\ j \in \mathcal{S}$ are regarded as parameters in (27). Note that in $\widehat{\Delta}(d,T)$, we normalize the regret of the ith instance by its maximal reward $\bar{r}^{(i)}$. For each diameter d, we find the (normalized) empirical maximum average regret with respect to T, i.e.,

$$\widehat{\Delta}_{\max}(d) = \max_T \widehat{\Delta}(d,T), \tag{28}$$

where the maximum is taken over all T included in the simulation.

Now we define temporal concatenation with K sub-instances ($K \geq 2$) in an analogous way as in Definition 1. For an original instance $\mathcal{I}_0$, denote by $\{\mathcal{I}_k\}_{k \in [K]}$ the sub-instances generated by partitioning $\mathcal{I}_0$ into K sub-instances of (approximately) equal length along the time horizon. Let $\pi_k^* = \mathrm{ALG}(\mathcal{I}_k)$, $k \in [K]$. The temporal concatenation heuristic with K sub-instances generates a policy, π_{TC}, by temporally concatenating optimal solutions for $\mathcal{I}_k$, which is analogous to (6).

4.4.2. Results

Our first finding shows that the empirical maximum average regret $\widehat{\Delta}_{\max}(d)$ increases linearly with respect to the diameter d. As illustrated in Figure 3(a), the empirical maximum average regret $\widehat{\Delta}_{\max}(d)$ exhibits an increasing trend as the diameter d increases from 20 to 50 for temporal concatenation with $K = 2, 3, 4, 5$ sub-instances. By Lemma 1, for a DGT instance based on G, $\mathcal{I}_G$, the 0-diameter is bounded between d and $2d$. Hence, the numerical result is consistent with Theorem 1, which bounds the performance regret from above by the 0-diameter τ_0 if the maximal reward is normalized to be 1. Note that the slopes in Figure 3(a) are much smaller than 1, which, as expected, is due to the worst-case nature of the upper bound. From the same figure, we also see that increasing the number of sub-instances in temporal concatenation will increase the average regret. In particular, when the horizon $[T]$ is fixed, as the number of sub-instances, K, increases, the length of each sub-instance decreases. Shorter

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

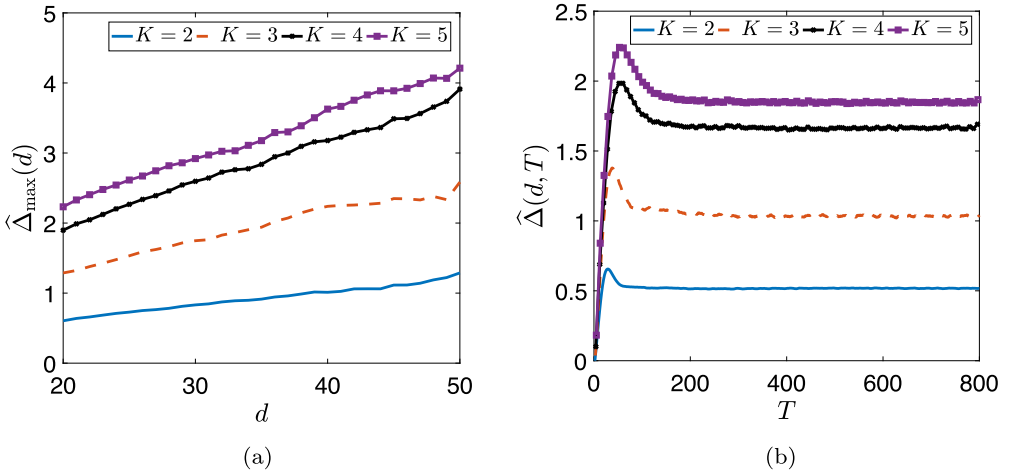

Figure 3. *The normalized regret of DGT instances based on directed graphs for temporal concatenation with $K = 2, 3, 4, 5$ sub-instances. (The original temporal concatenation corresponds to $K = 2$.) (a) Empirical maximum average regret $\widehat{\Delta}_{max}(d)$ as a function of the diameter d; (b) Empirical average regret $\widehat{\Delta}(d, T)$ as a function of the horizon T, for a fixed diameter $d = 23$. (The plots are smoothed by a 5-step moving-average filter.)*

sub-instances will more likely lead to overly short-sighted policies, which impede the performance of temporal concatenation.

The second finding suggests that for a fixed diameter d, as T grows, the empirical average regret $\widehat{\Delta}(d, T)$ first increases, then decreases after reaching a peak, and finally stabilizes when T is sufficiently large. This trend is illustrated in Figure 3(b). Intuitively, when T starts growing from zero, temporal concatenation starts to incur performance regret. Since the temporal concatenation policy is sub-optimal, the regret becomes larger with more time steps. When T is sufficiently large, however, the regret no longer increases. An intuitive explanation is that the temporal concatenation policy and the optimal policy become similar when the length of a sub-instance is sufficiently large, which causes the regret to start decreasing in this region. It remains an interesting open problem for finding the minimum horizon T beyond which the average regret starts to decrease.

4.5. Run-time reduction by temporal concatenation

In this subsection, we conduct additional numerical simulations to assess the benefit of run-time reduction from using temporal concatenation. We apply the classic value iteration algorithm to solve DGT instances and compare the run-time of the following two cases:

1. sequentially computing the optimal policy of the original instance;
2. using temporal concatenation, where we employ the built-in Matlab command spmd to solve the two sub-instances in parallel on a multi-core processor.

We run all experiments on a standard multi-core desktop computer; the specifications of the environment are described in detail in Appendix D. We construct DGT models with the same setting as described in Section 4.4, except that here we vary the number of states and time horizon length. We consider the cases that the number of states $|\mathcal{S}| = 1{,}000$, $2{,}000$, $3{,}000$, and the time horizon $T = 50$, 500, $12{,}000$, respectively. For each pair of $|\mathcal{S}|$ and T, we randomly generate 30 instances and present their respective run-time in Table 1(a)–(c). We will denote by T_{seq} the run-time of solving the original problem sequentially. T_{tc} denotes the time taken by the temporal concatenation method when run with

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

Table 1. *Computation time comparison between sequential value iteration and temporal concatenation.*

T	$\overline{T_{\text{seq}}}$ ($\sigma_{T_{\text{seq}}}$)	$\overline{T_{\text{tc}}}$ ($\sigma_{T_{\text{tc}}}$)	$\overline{\eta}$ (σ_{η})
	(a) $\lvert\mathcal{S}\rvert$ = 1,000		
50	0.292 (0.007)	0.280 (0.020)	0.958 (0.060)
500	2.881 (0.058)	2.065 (0.117)	0.717 (0.042)
12,000	68.494 (1.542)	48.482 (2.182)	0.708 (0.035)
	(b) $\lvert\mathcal{S}\rvert$ = 2,000		
50	1.463 (0.033)	0.989 (0.034)	0.676 (0.022)
500	14.747(0.149)	8.883 (0.248)	0.602 (0.018)
12,000	349.295 (5.910)	212.220 (10.334)	0.608 (0.020)
	(c) $\lvert\mathcal{S}\rvert$ = 3,000		
50	3.514 (0.026)	2.135 (0.061)	0.608 (0.017)
500	35.381 (0.143)	19.633 (0.327)	0.555 (0.009)
12,000	837.234 (15.070)	472.133 (14.640)	0.564 (0.015)

Matlab's native parallel computation framework. The ratio between the two values $\eta = T_{\text{tc}}/T_{\text{seq}}$ is therefore a metric of interest, showing the multiplicative speed-up obtained by temporal concatenation. For a value x, we will use $\overline{x}$ to denote its empirical mean, and σ_x its standard deviation.

We can see from the results that the temporal concatenation heuristic reduces the run-time thanks to parallelism. We also note that the time reduction is not exactly 50% as one might expect, and this is largely due to the fact that there is a non-trivial overhead for Matlab to initiate a parallel computation instance. That being said, the reduction is still fairly significant across the board and gets closer to 50% when the run-time of the original problem is sufficiently long (e.g., for the cases of $|\mathcal{S}| = 3{,}000$, $T = 500, 12{,}000$).

Interestingly, we notice that the value of $|\mathcal{S}|$ seems more significant in determining the ratio η than the horizon T. Suppose we fix $|\mathcal{S}|$. In this case, increasing T from 50 to 500 reduces η notably, while further increasing its value from 500 to 12,000 seems not very influential. On the other hand, if we fix T and increase $|\mathcal{S}|$ from 1,000 to 2,000 and then to 3,000, the value of η decreases remarkably. We conjecture that this is due to the fact that the run-time of each sub-instance scales quadratically in $|\mathcal{S}|$ but only linearly in T. Therefore, as the two sub-instances become larger, its computation time would tend to dwarf Matlab's computational overhead in setting up the parallel computation instance, leading to a more favorable ratio of speed-up η.

Admittedly, what we have here is a relatively simple proof-of-concept with off-the-shelf software, and we expect that a more optimized implementation of the parallel computation, possibly on distinct physical machines, would further decrease the run-time.

4.6. Simulation results for the GARNET MDP model

In this subsection, we provide numerical simulations for a more widely studied family of MDP instances introduced in [2], known as the GARNET model. In this generative model of MDP, a branching factor B determines the transition kernels. When sampling an MDP instance, under each action, every state is randomly assigned B possible next states, while the probability of transitioning to each of them is also randomly drawn. Here, we construct GARNET MDP models with $|\mathcal{S}| = 200$ states, $|\mathcal{A}| = 3$ actions, and let B range from 3 to 15. For each choice of B, we construct $N_G = 5{,}000$ examples independently and record their average regrets. To make sure that the MDP is aperiodic, we introduce a self-loop around each state for every action, i.e., each state has one edge to itself and $B - 1$ edges to other states under each action. The reward functions R_t are deterministic, which are determined by the current state and the chosen action. For each pair of state and action, the corresponding reward is drawn uniformly

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

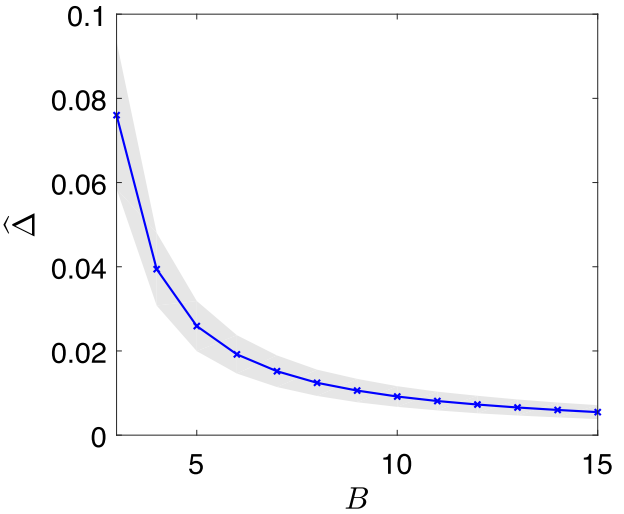

Figure 4. *The normalized regret of temporal concatenation with two sub-instances for the GARNET MDP model with* $|\mathcal{S}| = 200$, $T = 200$, $|\mathcal{A}| = 3$, $B = 3, 4, \ldots, 15$.

at random from 1 to 200. In this simulation, we focus on the regret of temporal concatenation with two sub-instances.

In Figure 4, we present the simulation results. Here, $\widehat{\Delta}$ is the empirical average regret normalized by the maximal reward, defined as follows:

$$\widehat{\Delta} = \frac{1}{N_G} \sum_{i=1}^{N_G} \frac{1}{\bar{r}^{(i)}} \left(\mathbb{E}^{\pi^*} \left[\sum_{t=0}^{T-1} R_t^{(i)}(a_t^{(i)}, S_t^{(i)}) \right] - \mathbb{E}^{\pi_{\mathrm{TC}}} \left[\sum_{t=0}^{T-1} R_t^{(i)}(a_t^{(i)}, S_t^{(i)}) \right] \right), \tag{29}$$

where $R_t^{(i)}$, $a_t^{(i)}$, $S_t^{(i)}$, $\bar{r}^{(i)}$ are, respectively, the reward functions, actions, states, and maximal reward in the i^{th} realization of the GARNET model, $i \in [N_G]$.

The shaded area in the figure depicts the area within one empirical standard deviation of the mean. Overall, the performance is favorable. We see that the regret in the GARGET model is substantially smaller than that of DGT (Figure 3), and it decreases even more as B becomes large. We suspect that the random and uniform nature with which GARGET generates transition kernels contributes to the resulting MDP having a relatively small ϵ-diameter, even when B is moderate, and the diameter becomes even smaller as the number of neighboring states, B, increases.

5. Conclusion

In this paper, we propose and analyze a heuristic architecture, temporal concatenation, for speeding up existing MDP algorithms when solving a finite-horizon Markov decision process. Temporal concatenation decomposes the problem over the time horizon into smaller sub-problems and subsequently concatenating their optimal policies.

Using a notion of ϵ-diameter, we provide upper bounds that show, when the underlying MDP instance admits a bounded ϵ-diameter the regret of temporal concatenation is bounded and *independent* of the length of the horizon. Conversely, we provide lower bounds by showing that, for any finite diameter, there exist MDP instances for which the regret upper bound is tight for all sufficiently large horizons.

At the high level, we aim to explore an alternative approach for solving large-scale MDPs: instead of creating new algorithms from scratch, we may be able to leverage existing MDP algorithms in creative ways to harness additional performance gains. The present paper takes a first step towards this direction by decomposing the problem along the time axis. There is a number of interesting directions for future

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

work. While we have demonstrated that the ϵ-diameter of an MDP has a substantial impact on the performance of temporal concatenation, it remains a challenge in general to compute such diameter for a given MDP. Understanding how to obtain sharp bounds for, or numerically compute, the ϵ-diameter can be an interesting direction of future research. The theory of our present paper has mostly focused on dividing the MDP into two equal-sized sub-instances, while in general, one may consider $K \geq 2$ sub-instances or study the case where the sizes of the sub-instances may even vary. It would be interesting to understand how best to choose the number and the sizes of the sub-instances, especially when the original MDP is time-inhomogeneous.

In addition, it will be interesting to explore connections between temporal concatenation and other approximation heuristics aimed at reducing the complexity of solving an MDP. For instance, if we further assume that the MDP is time-homogeneous (transitions and rewards do not vary with time) and that the time horizon is very large, then a natural alternative would be to directly deploy a stationary optimal policy associated with the average-reward version of the MDP, which can be calculated efficiently using a horizon-independent linear program. The regret of such an approach would likely depend on whether the steady-state optimal policy could quickly reach its stationary distribution from an arbitrary initial condition, and it would be interesting to understand, for instance, whether the latter is related to the notion of diameter studied in this paper.

Finally, in a broader sense, the present work only explores decomposing an MDP along the time axis, and it would be interesting to explore the efficiency of other forms of decomposition architectures, such as those that operate through states.

Acknowledgments. We thank the anonymous referees for their comments and feedback.

References

[1] Bellman, R. (1954). The theory of dynamic programming. *Bulletin of the American Mathematical Society* 60(6): 503–515.
[2] Bhatnagar, S., Sutton, R.S., Ghavamzadeh, M., & Lee, M. (2009). Natural actor–critic algorithms. *Automatica* 45(11): 2471–2482.
[3] Bondy, J. A. & Murty, U. S. R. (1976). *Graph theory with applications*, vol. 290. London: MacMillan.
[4] Burnetas, A.N. & Katehakis, M.N. (1997). Optimal adaptive policies for Markov decision processes. *Mathematics of Operations Research* 22(1): 222–255.
[5] BYD Company. (2017). BYD battery energy storage projects. https://sg.byd.com/wp-content/uploads/2017/10/Energy-Storage-System.pdf. Accessed: 27 May 2020.
[6] California ISO. CAISO market processes and products. http://www.caiso.com/market/Pages/MarketProcesses.aspx. Accessed: 27 May 2020.
[7] Chen, Y. & Wang, M. (2016). Stochastic primal-dual methods and sample complexity of reinforcement learning. Preprint arXiv:1612.02516.
[8] Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., & Zhang, Z.. (2015). Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. Preprint arXiv:1512.01274.
[9] Daoui, C., Abbad, M., & Tkiouat, M. (2010). Exact decomposition approaches for Markov decision processes: A survey. In *Advances in Operations Research 2010*. London, UK: Hindawi.
[10] Dietterich, T.G. (2000). Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of Artificial Intelligence Research* 13: 227–303.
[11] Durrett, R. (2019). *Probability: theory and examples*, vol. 49. Cambridge University Press.
[12] Harrison, J.M. & Zeevi, A. (2004). Dynamic scheduling of a multiclass queue in the Halfin-Whitt heavy traffic regime. *Operations Research* 52(2): 243–257.
[13] Harsha, P. & Dahleh, M. (2015). Optimal management and sizing of energy storage under dynamic pricing for the efficient integration of renewable energy. *IEEE Transactions on Power Systems* 30(3): 1164–1181.
[14] Ie, E., Jain, V., Wang, J., Narvekar, S., Agarwal, R., Wu, R., Cheng, H.-T., Chandra, T. & Boutilier, C. (2019). Slateq: A tractable decomposition for reinforcement learning with recommendation sets. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)*. California, USA: International Joint Conferences on Artificial Intelligence Organization, pp. 2592–2599.
[15] Jaksch, T., Ortner, R., & Auer, P. (2010). Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research 11* Apr: 1563–1600.
[16] Kolobov, A., Dai, P., Mausam, M., & Weld, D.S. (2012). Reverse iterative deepening for finite-horizon MDPs with large branching factors. In *Twenty-Second International Conference on Automated Planning and Scheduling*. Palo Alto, CA, USA: AAAI press, pp. 146–154.

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

[17] Littman, M.L., Dean, T.L. & Kaelbling, L.P. (1995). On the complexity of solving Markov decision problems. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*. Burlington, MA: Morgan Kaufmann Publishers Inc., pp. 394–402.
[18] Low, Y., Bickson, D., Gonzalez, J., Guestrin, C., Kyrola, A., & Hellerstein, J.M. (2012). Distributed GraphLab: A framework for machine learning and data mining in the cloud. *Proceedings of the VLDB Endowment* 5(8): 716–727.
[19] Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., Freeman, J., Tsai, D., Amde, M., Owen, S., Xin, D. , Xin, R., Franklin, M., Zadeh, R., Zaharia, M., & Talwalkar, A. (2016). MLlib: Machine learning in Apache Spark. *The Journal of Machine Learning Research* 17(1): 1235–1241.
[20] Mundhenk, M., Goldsmith, J., Lusena, C., & Allender, E. (2000). Complexity of finite-horizon Markov decision process problems. *Journal of the ACM (JACM)* 47(4): 681–720.
[21] Parr, R. & Russell, S.J. (1998). Reinforcement learning with hierarchies of machines. In Advances in neural information processing systems. Cambridge, MA: MIT press, pp. 1043–1049.
[22] Puterman, M.L. (2014). *Markov decision processes: Discrete stochastic dynamic programming*. Hoboken, NJ, USA: John Wiley & Sons.
[23] Steimle, L.N., Ahluwalia, V.S., Kamdar, C. & Denton, B.T. (2021). Decomposition methods for solving Markov decision processes with multiple models of the parameters. *IISE Transactions*, 1–58. doi:10.1080/24725854.2020.1869351.
[24] Sucar, L.E. (2007). Parallel Markov decision processes. In *Advances in Probabilistic Graphical Models*. Berlin, Heidelberg: Springer, pp. 295–309.
[25] Talebi, M. & Maillard, O.A. (2018). Variance-aware regret bounds for undiscounted reinforcement learning in MDPs. *Journal of Machine Learning Research* 83: 1–36.
[26] Tseng, P. (1990). Solving H-horizon, stationary Markov decision problems in time proportional to $\log(H)$. *Operations Research Letters* 9(5): 287–297.
[27] Watts, D.J. & Strogatz, S.H. (1998). Collective dynamics of "small-world" networks. *Nature* 393(6684): 440.

Appendix A. Proofs of main results

This section is devoted to the proofs of the main results. Before delving into the details, we first provide a high-level overview of the key ideas.

Upper bounds (Section A.1). For Theorem 1, observe that the temporal concatenation heuristic by construction achieves optimal total expected reward during the first sub-instance, $\mathcal{I}_1$. The problem arises, however, if acting greedily during $\mathcal{I}_1$ would result in the system being in a disadvantageous state at the beginning of the second sub-instance, $\mathcal{I}_2$, thus leading to a large regret. Our analysis for the regret upper bound in Theorem 1 will therefore focus on the dynamics of temporal concatenation during $\mathcal{I}_2$. To this end, we will employ a coupling argument, by bounding temporal concatenation's regret from above using that of a carefully constructed, and likely strictly sub-optimal, 'fictitious' policy, $\tilde{\pi}$, during $\mathcal{I}_2$. The policy $\tilde{\pi}$ consists of multiple phases of length approximately τ_ϵ. In the kth phase, it aims to reduce the (total variation) distance with the overall optimal policy π^* over the course of τ_ϵ steps. Using an argument based on recursion, we show that in the kth phase this policy incurs a regret that is up to $\epsilon^{k-1}\bar{r}\tau_\epsilon$. This will in turn allow us to show that the regret of $\tilde{\pi}$ incurred during the second phase is small.

Lower bounds (Sections A.2). For the lower bound in Theorem 2, we build on the insights gathered from the proof of Theorem 1 to generate worst-case MDP instances. The main idea is to construct instances in such a way that during the first sub-instance, the temporal concatenation heuristic is guaranteed to be lured by some *small* short-term rewards and end up in a 'bad' subset of the state space, from which it will suffer large losses in the second half of the time horizon compared to the optimal policy.

Proof of Theorem 1

Recall that μ_t^π is the distribution of the state S_t induced by a policy π. For simplicity of notation, we will write μ_t in place of μ_t^π when there is no ambiguity and use the shorthand:

$$\mu_t^{\pi^*} \triangleq \mu_t^*, \quad \text{and} \quad \mu_t^{\pi_{\text{TC}}} \triangleq \mu_t^{\text{TC}}. \tag{A.1}$$

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

We also define the cumulative rewards from time t_1 to t_2 as

$$\tilde{V}(t_1, t_2) \triangleq \sum_{t=t_1}^{t_2} R_t(a_t, S_t, Y_t^R), \quad t_1 < t_2. \tag{A.2}$$

The next lemma is the key technical result. Recall from (2) that, for a given instance, the total expected reward of a policy depends on the initial distribution. In Lemma 4, we provide an upper bound on the performance difference of the optimal policy when the system starts from two different initial distributions.

Lemma 4. *Fix an instance $\mathcal{I}$ with horizon $[T]$. Fix distributions $\mu_0, \nu_0 \in \mathcal{P}$. Let π^* be the optimal policy for the instance $\mathcal{I}$. If there exists $\epsilon > 0$ such that $\tau_\epsilon(\mathcal{I}) \le T$, the difference in total expected reward under π^* between the cases where the initial distribution is μ_0 versus ν_0 is bounded from above as follows:*

$$|V(\mathcal{I}, \pi^*, \mu_0) - V(\mathcal{I}, \pi^*, \nu_0)| \le \frac{\bar{r}\tau_\epsilon(\mathcal{I})}{1-\epsilon}. \tag{A.3}$$

We first prove Lemma 4, using a coupling argument. For state $s \in \mathcal{S}$, starting time $t \in [T]$, and policy π, we define the value function as follows:

$$V_t^\pi(s) = \mathbb{E}^\pi[\tilde{V}(t, T-1) \mid S_t = s]. \tag{A.4}$$

For the instance $\mathcal{I}$ and policy π^*, the total expected reward for initial distribution μ is

$$V(\mathcal{I}, \pi^*, \mu) = \sum_{s\in\mathcal{S}} \mathbb{E}^{\pi^*}[\tilde{V}(0, T-1) \mid S_0 = s]\mu(s) = \sum_{s\in\mathcal{S}} V_0^{\pi^*}(s)\mu(s). \tag{A.5}$$

Fixing the policy π^*, the difference in total expected rewards under π^* but starting with two initial distributions, μ_0 and ν_0, can be expanded as:

$$|V(\mathcal{I}, \pi^*, \mu_0) - V(\mathcal{I}, \pi^*, \nu_0)| = \left|\sum_{s\in\mathcal{S}} V_0^{\pi^*}(s)(\mu_0(s) - \nu_0(s))\right|. \tag{A.6}$$

Without loss of generality, suppose that

$$V(\mathcal{I}, \pi^*, \mu_0) \ge V(\mathcal{I}, \pi^*, \nu_0). \tag{A.7}$$

Now we provide an upper bound on the difference in total expected reward by introducing a "fictitious" policy $\tilde{\pi}$. Suppose $\tau_\epsilon(\mathcal{I}) \le T$ for some $\epsilon > 0$. Recall that by the definition of ϵ-diameter, there exists a policy $\tilde{\pi}_{\text{ap}}$ such that starting from ν_0, the state distribution at time $\tau_\epsilon(\mathcal{I})$ under $\tilde{\pi}_{\text{ap}}$, which is denoted by $\tilde{\nu}_{\tau_\epsilon(\mathcal{I})}$, satisfies

$$\delta_{\text{TV}}(\mu^*_{\tau_\epsilon(\mathcal{I})}, \tilde{\nu}_{\tau_\epsilon(\mathcal{I})}) \le \epsilon, \tag{A.8}$$

where $\mu^*_{\tau_\epsilon(\mathcal{I})}$ is the state distribution at time $\tau_\epsilon(\mathcal{I})$ starting from μ_0 under policy π^*. The policy $\tilde{\pi}$ is defined as follows: for time $t \in 0 \to \tau_\epsilon(\mathcal{I}) - 1$, let $\tilde{\pi} = \tilde{\pi}_{\text{ap}}$; for time $t \in \tau_\epsilon(\mathcal{I}) \to T-1$, let $\tilde{\pi} = \pi^*$. Note that $\tilde{\pi}$ is sub-optimal compared to π^*, and we have

$$V(\mathcal{I}, \pi^*, \nu_0) \ge V(\mathcal{I}, \tilde{\pi}, \nu_0). \tag{A.9}$$

Recall that the reward function R_t takes values in $[0, \bar{r}]$. We have that

$$|V(\mathcal{I}, \pi^*, \mu_0) - V(\mathcal{I}, \pi^*, \nu_0)| = V(\mathcal{I}, \pi^*, \mu_0) - V(\mathcal{I}, \pi^*, \nu_0) \tag{A.10}$$

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

$$\leq V(\mathcal{I}, \pi^*, \mu_0) - V(\mathcal{I}, \tilde{\pi}, \nu_0) \tag{A.11}$$

$$\leq \bar{r}\tau_\epsilon(\mathcal{I}) + \sum_{s\in\mathcal{S}} V^{\pi^*}_{\tau_\epsilon(\mathcal{I})}(s)(\mu^*_{\tau_\epsilon(\mathcal{I})}(s) - \tilde{\nu}_{\tau_\epsilon(\mathcal{I})}(s)), \tag{A.12}$$

where (A.10) follows from (A.7), (A.11) from (A.9), and (A.12) from $R_t \leq \bar{r}$.

For $s \in \mathcal{S}$, let $\omega(s) = \min\{\mu^*_{\tau_\epsilon(\mathcal{I})}(s), \tilde{\nu}_{\tau_\epsilon(\mathcal{I})}(s)\}$. Define $\epsilon_0 = \sum_{s\in\mathcal{S}}(\tilde{\nu}_{\tau_\epsilon(\mathcal{I})}(s) - \omega(s))$. Note that $\sum_{s\in\mathcal{S}}(\mu^*_{\tau_\epsilon(\mathcal{I})}(s) - \omega(s)) = \sum_{s\in\mathcal{S}}(\tilde{\nu}_{\tau_\epsilon(\mathcal{I})}(s) - \omega(s)) = \epsilon_0$.

Let $\mu^-_{\tau_\epsilon(\mathcal{I})}(s) = (\mu^*_{\tau_\epsilon(\mathcal{I})}(s) - \omega(s))/\epsilon_0$, and $\nu^-_{\tau_\epsilon(\mathcal{I})}(s) = (\tilde{\nu}_{\tau_\epsilon(\mathcal{I})}(s) - \omega(s))/\epsilon_0$. By the definition of total variation, we have $\epsilon_0 \leq \epsilon$. Note that $\mu^-_{\tau_\epsilon(\mathcal{I})}, \nu^-_{\tau_\epsilon(\mathcal{I})} \geq 0$, and $\sum_{s\in\mathcal{S}} \mu^-_{\tau_\epsilon(\mathcal{I})}(s) = 1$, $\sum_{s\in\mathcal{S}} \nu^-_{\tau_\epsilon(\mathcal{I})}(s) = 1$. Hence, $\mu^-_{\tau_\epsilon(\mathcal{I})}$, $\nu^-_{\tau_\epsilon(\mathcal{I})}$ are probability distributions.

Then

$$\begin{aligned}
|V(\mathcal{I}, \pi^*, \mu_0) - V(\mathcal{I}, \pi^*, \nu_0)| &= \left|\sum_{s\in\mathcal{S}} V^{\pi^*}_0(s)(\mu_0(s) - \nu_0(s))\right| \\
&\leq \bar{r}\tau_\epsilon(\mathcal{I}) + \sum_{s\in\mathcal{S}} V^{\pi^*}_{\tau_\epsilon(\mathcal{I})}(s)(\mu^*_{\tau_\epsilon(\mathcal{I})}(s) - \tilde{\nu}_{\tau_\epsilon(\mathcal{I})}(s)) \\
&= \bar{r}\tau_\epsilon(\mathcal{I}) + \epsilon_0 \sum_{s\in\mathcal{S}} V^{\pi^*}_{\tau_\epsilon(\mathcal{I})}(s)(\mu^-_{\tau_\epsilon(\mathcal{I})}(s) - \nu^-_{\tau_\epsilon(\mathcal{I})}(s)) \\
&\leq \bar{r}\tau_\epsilon(\mathcal{I}) + \epsilon_0 \left|\sum_{s\in\mathcal{S}} V^{\pi^*}_{\tau_\epsilon(\mathcal{I})}(s)(\mu^-_{\tau_\epsilon(\mathcal{I})}(s) - \nu^-_{\tau_\epsilon(\mathcal{I})}(s))\right|
\end{aligned} \tag{A.13}$$

$$\leq \bar{r}\tau_\epsilon(\mathcal{I}) + \epsilon \left|\sum_{s\in\mathcal{S}} V^{\pi^*}_{\tau_\epsilon(\mathcal{I})}(s)(\mu^-_{\tau_\epsilon(\mathcal{I})}(s) - \nu^-_{\tau_\epsilon(\mathcal{I})}(s))\right|. \tag{A.14}$$

Here (A.13) follows from the definition of $\mu^-_{\tau_\epsilon(\mathcal{I})}$ and $\nu^-_{\tau_\epsilon(\mathcal{I})}$, and (A.14) from $\epsilon_0 \leq \epsilon$. Let $N = \lfloor T/\tau_\epsilon(\mathcal{I})\rfloor$. For $k = 0, 1, \ldots, N-1$, starting from time $t = k\tau_\epsilon(\mathcal{I})$, we can use the same argument to derive the following inequality:

$$\begin{aligned}
&\left|\sum_{s\in\mathcal{S}} V^{\pi^*}_{k\tau_\epsilon(\mathcal{I})}(s)(\mu^-_{k\tau_\epsilon(\mathcal{I})}(s) - \nu^-_{k\tau_\epsilon(\mathcal{I})}(s))\right| \\
&\quad\leq \bar{r}\tau_\epsilon(\mathcal{I}) + \epsilon \left|\sum_{s\in\mathcal{S}} V^{\pi^*}_{(k+1)\tau_\epsilon(\mathcal{I})}(s)(\mu^-_{(k+1)\tau_\epsilon(\mathcal{I})}(s) - \nu^-_{(k+1)\tau_\epsilon(\mathcal{I})}(s))\right|,
\end{aligned} \tag{A.15}$$

where $\mu^-_{k\tau_\epsilon(\mathcal{I})}(s)$ and $\nu^-_{k\tau_\epsilon(\mathcal{I})}(s)$ are defined in the same way as $\mu^-_{\tau_\epsilon(\mathcal{I})}(s)$ and $\nu^-_{\tau_\epsilon(\mathcal{I})}(s)$. Note also that

$$V^{\pi^*}_{N\tau_\epsilon(\mathcal{I})}(s) \leq \bar{r}\tau_\epsilon(\mathcal{I}). \tag{A.16}$$

With (A.15) and (A.16), we have

$$|V(\mathcal{I}, \pi^*, \mu_0) - V(\mathcal{I}, \pi^*, \nu_0)| \leq \bar{r}\tau_\epsilon(\mathcal{I})(1 + \epsilon + \cdots + \epsilon^N t) \leq \frac{\bar{r}\tau_\epsilon(\mathcal{I})}{1-\epsilon}. \tag{A.17}$$

This completes the proof of Lemma 4.

Lemma 4 suggests that starting from two different initial distributions, μ_0 and ν_0, the optimal policy π^* is guaranteed to have similar performances when the ϵ-diameter is small. We now prove Theorem 1

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

using Lemma 4. First, for any initial distribution μ_0, we can expand the regret of temporal concatenation as the sum of the regrets incurred during the first and second sub-instance, separately:

$$\begin{aligned}\Delta(\mathcal{I}_0, \mu_0) &= V(\mathcal{I}_0, \pi^*, \mu_0) - V(\mathcal{I}_0, \pi_{\mathrm{TC}}, \mu_0) \\ &= (V(\mathcal{I}_1, \pi^*, \mu_0) - V(\mathcal{I}_1, \pi_1^*, \mu_0)) \\ &\quad + (V(\mathcal{I}_2, \pi^*, \mu_{T/2}^*) - V(\mathcal{I}_2, \pi_2^*, \mu_{T/2}^{\mathrm{TC}})),\end{aligned} \tag{A.18}$$

where, with a slight abuse of notation, we use $V(\mathcal{I}_1, \pi^*, \mu_0)$ to denote the total expected reward from applying the policy π^* during the first sub-instance. Note that during the first $T/2$ steps, the original optimal policy, π^*, does not necessarily maximize the reward for this sub-instance, because it aims at maximizing the overall reward of $\mathcal{I}_0$. Hence, for this sub-instance only, the temporal concatenation method is performing better than, or equally to, the original optimal policy, i.e., the first term in (A.18) satisfies:

$$V(\mathcal{I}_1, \pi^*, \mu_0) - V(\mathcal{I}_1, \pi_1^*, \mu_0) \leq 0. \tag{A.19}$$

We now bound the second term in (A.18). Suppose for some $\epsilon > 0$, we have $\tau_\epsilon(\mathcal{I}_0) \leq T/2$. Note that both π^* and π_2^* achieve the optimal performance for the second sub-instance $\mathcal{I}_2$, we have

$$V(\mathcal{I}_2, \pi_2^*, \mu_{T/2}^{\mathrm{TC}}) = V(\mathcal{I}_2, \pi^*, \mu_{T/2}^{\mathrm{TC}}). \tag{A.20}$$

Hence, using Lemma 4, we have that

$$\begin{aligned}V(\mathcal{I}_2, \pi^*, \mu_{T/2}^*) - V(\mathcal{I}_2, \pi_2^*, \mu_{T/2}^{\mathrm{TC}}) &= V(\mathcal{I}_2, \pi^*, \mu_{T/2}^*) - V(\mathcal{I}_2, \pi^*, \mu_{T/2}^{\mathrm{TC}}) \\ &\leq \frac{\bar{r}\tau_\epsilon(\mathcal{I}_2)}{1-\epsilon} \qquad \text{(A.21)} \\ &\leq \frac{\bar{r}\tau_\epsilon(\mathcal{I}_0)}{1-\epsilon}, \qquad \text{(A.22)}\end{aligned}$$

where (A.21) is derived by applying Lemma 4 to the second sub-instance $\mathcal{I}_2$ for initial distributions $\mu_{T/2}^*$ and $\mu_{T/2}^{\mathrm{TC}}$, and (A.22) follows from the fact that $\mathcal{I}_2$ is a sub-instance of $\mathcal{I}_0$, which leads to $\tau_\epsilon(\mathcal{I}_2) \leq \tau_\epsilon(\mathcal{I}_0)$.

To complete the proof, we substitute the regret upper bounds for the first (A.19) and second (A.22) sub-instances into (A.18), and obtain

$$V(\mathcal{I}_0, \pi^*, \mu_0) - V(\mathcal{I}_0, \pi_{\mathrm{TC}}, \mu_0) \leq \frac{\bar{r}\tau_\epsilon(\mathcal{I}_0)}{1-\epsilon}. \tag{A.23}$$

This completes the proof of Theorem 1. □

Proof of Theorem 2

We now prove Theorem 2 by constructing a family of MDP instances and showing that temporal concatenation suffers the regret given in the theorem on problems from this family. The key intuition is that the instances can be constructed in such a way that the temporal concatenation heuristic will be led astray by some small short-term rewards in the first half of the horizon and end up in a bad subset of the state space, from which it will suffer large losses in the second half of the time horizon compared to the optimal policy.

Fix $d_0 \geq 5$, and let $k = d_0 - 2$. Consider the MDP instance depicted in Figure 5. The state space has $|\mathcal{S}| = k + 2$ elements and we have $d_0 = k + 2$. The transition function is deterministic. In states $d_1, \ldots, d_k$, and e, the agent can choose between two actions, such that the system either stays in the same state or goes to the next state to the right. In state f , the system will always go to state d_1 in the next step. For state $s \in \{d_1, \ldots, d_k\}$, the reward function $R_t(a, s, y) = 0$, for all t, $a \in \mathcal{A}$, and $y \in \mathcal{Y}^R$. For

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

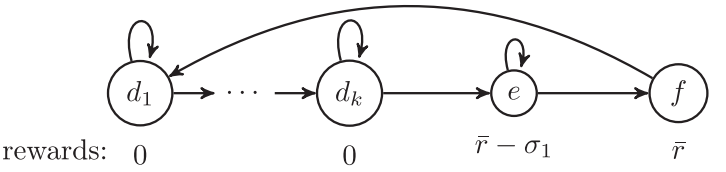

Figure 5. *An MDP instance with bounded diameter and large performance regret.*

state $s = e$ and f, the reward $R_t(a, s, y)$ is always equal to $\bar{r} - \sigma_1$, and $\bar{r}$, respectively, where $\sigma_1 \in (0, \sigma)$ is a constant to be specified subsequently.

We first verify that the setting has a finite diameter $\tau_0(\mathcal{I}_0) = d_0 = k + 2$. Suppose the system starts from an initial distribution $\nu \in \mathcal{P}$ and we try to reach another distribution $\nu' \in \mathcal{P}$. Consider the following policy:

Stage 1: If the initial state is in $\{d_1, \ldots, d_k, e\}$, stay for one step; if the initial state is f, go to d_1 in the first step. Hence, Stage 1 takes 1 step.

Stage 2: Starting from one of the states in $\{d_1, \ldots, d_k, e\}$, the agent reaches the state distribution ν' after another $k + 1$ steps. Note that starting from any state in $\{d_1, \ldots, d_k, e\}$, the system can reach any state $s \in \mathcal{S}$ using $k + 1$ steps by first staying at the current state for an appropriate number of steps and then moving forward to reach the target state. We refer to this stay-and-move process as a $(k + 1)$-path to state s. The agent can thus employ the following randomized policy: starting at state $s_0 \in \{d_1, \ldots, d_k, e\}$, with probability $\nu'(s)$ the agent chooses to take the $(k + 1)$-path to state s, for $s \in \mathcal{S}$. Stage 2 takes $k + 1$ steps.

Using the policy described above, we can reach any distribution ν' at time $t = k + 2$ starting from any initial distribution ν. Hence, we have shown that the diameter satisfies

$$\tau_0(\mathcal{I}_0) \leq k + 2. \tag{A.24}$$

We now establish a lower bound for $\tau_0(\mathcal{I}_0)$. Suppose the initial distribution is concentrated on state f, i.e., $\nu(f) = 1$. At time $t = 0, 1$, the state will be deterministically f and d_1, respectively. Hence, in order to reach a distribution ν' with $\nu'(d_1) = \nu'(f) = 0.5$, it takes at least another $k + 1$ steps. Then,

$$\tau_0(\mathcal{I}_0) \geq k + 2. \tag{A.25}$$

In light of (A.24) and (A.25), we conclude that $\tau_0(\mathcal{I}_0) = k + 2 = d_0$.

We now consider the regret. Suppose the initial state is deterministically d_1. Recall that $T > 2d_0 + 2 = 2k + 6$. For the optimal policy, the agent will first go to state e, stay at e until time $T - 2$, and finally go to f at time $T - 1$. Hence, the total reward of the optimal policy is

$$V(\mathcal{I}_0, \pi^*, \mu_0) = (T - k)(\bar{r} - \sigma_1) + \sigma_1. \tag{A.26}$$

Under temporal concatenation, since $T/2 > k + 3$, for time 1 to $T/2$, the agent will go to state e, stay at e until time $T/2 - 2$, and go to state f at time $T/2 - 1$. For time $T/2$ to $T - 1$, the agent will have to go to e again after passing through $d_1, \ldots, d_k$, stay at e until time $T - 2$, and then go to f at time $T - 1$. Recall that states d_1 through d_k provide zero reward. The total reward for the temporal concatenation policy is

$$V(\mathcal{I}_0, \pi_{\mathrm{TC}}, \mu_0) = 2\left(\left(\frac{T}{2} - k\right)(\bar{r} - \sigma_1) + \sigma_1\right). \tag{A.27}$$

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

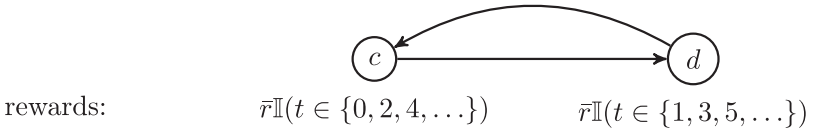

Figure 6. *An MDP instance with time-inhomogeneous reward function.*

Therefore, we have that the regret of temporal concatenation is given by

$$\begin{aligned}\Delta(\mathcal{I}_0, \mu_0) &= V(\mathcal{I}_0, \pi^*, \mu_0) - V(\mathcal{I}_0, \pi_{\mathrm{TC}}, \mu_0)\\ &= k\bar{r} - (k+1)\sigma_1\\ &= (\tau_0(\mathcal{I}_0) - 2)\bar{r} - (k+1)\sigma_1.\end{aligned}$$

By choosing $\sigma_1 = \sigma/(k+1)$, we have

$$\Delta(\mathcal{I}_0, \mu_0) = (\tau_0(\mathcal{I}_0) - 2)\bar{r} - \sigma. \tag{A.28}$$

This completes the proof of Theorem 2. □

Appendix B. Connection to the D^* diameter

In this section, we further discuss the connection between the ϵ-diameter, τ_ϵ, and the diameter D^* introduced in [15]. As shown in [15], for a time-homogeneous MDP, the value function of the optimal policy, π^*, satisfies that

$$\max_{s,s' \in \mathcal{S}} (V_t^{\pi^*}(s) - V_t^{\pi^*}(s')) \le \bar{r} D^*. \tag{B.1}$$

Hence,

$$\sup_{\mu,\mu' \in \mathcal{P}} (V(\mathcal{I}, \pi^*, \mu) - V(\mathcal{I}, \pi^*, \mu')) \le \bar{r} D^*. \tag{B.2}$$

Therefore, when both the reward function and the transition function are time-homogeneous, a small diameter D^* guarantees that the total expected rewards of different initial distributions are close. However, for the more general scenarios where either the reward function or the transition function is not time-homogeneous, a small diameter D^* is not sufficient for this to hold. In the remainder of this section, we present examples with either a time-inhomogeneous reward function, or a time-inhomogeneous transition function. For each instance, we show that there exist two initial distributions such that the difference of total expected reward is large although the diameter D^* is small. Hence, Lemma 4 is more general than (B.2) and the ϵ-diameter more precisely characterizes the communicating property of an MDP than the diameter D^*.

Time-inhomogeneous reward function

In this subsection, we consider the case where the transition function is time-homogeneous but the reward function is not. We construct an example with a small D^* and show that there exist two initial distributions such that the difference in total expected reward grows in T.

Consider the MDP instance in Figure 6, where there are two states c and d. The transition is deterministic: starting at state c, the agent can only go to d; starting at d, the agent can only go to c. The reward function, R_t, varies in time. In particular, at state c, the reward is $\bar{r}$ when t is an even number, and 0 when t is odd; at state d, the reward is $\bar{r}$ when t is an odd number, and 0 when t is even.

It is easy to verify that the D^* diameter is small for this instance. In particular, we can go from one state to another with exactly one step. Hence, $D^* = 1$. However, we show that the initial state distribution can significantly impact the total expected reward despite the small D^*. Suppose an agent starts at state

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

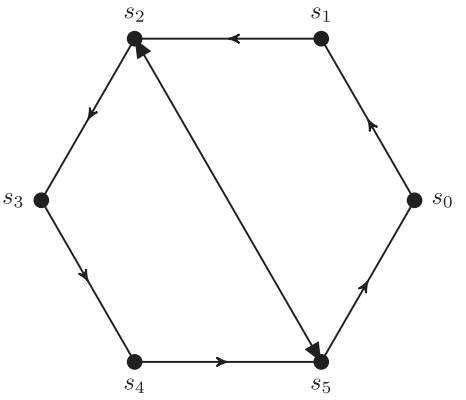

Figure 7. *An MDP instance with time-inhomogeneous transition function.*

c when $t = 0$, then the state at time $t = 0, 1, 2, \ldots$ is deterministically $c, d, c, \ldots$, generating a reward $\bar{r}$ at each time. However, if the agent starts at state d when $t = 0$, the reward at each step is always 0. Therefore,

$$V(\mathcal{I}, \pi^*, \delta_c) - V(\mathcal{I}, \pi^*, \delta_d) = \bar{r}T, \tag{B.3}$$

where δ_c and δ_d are the point measures at states c and d, respectively. Note that $\bar{r}T$ is the largest possible difference in total expected reward, which is attained in this example although D^* is small.

Time-inhomogeneous transition function

Now we consider the case that the reward function is time-homogeneous but the transition function is not. Consider the instance in Figure 7 with states $\mathcal{S} = \{s_0, s_1, s_2, s_3, s_4, s_5\}$. The reward function is deterministic and depends only on the state. The reward is $\bar{r}$ for state s_2, and 0 for the other five states. The state transition is also deterministic but varies in time, which is elaborated as follows:

1. Starting from state s_0, s_1, s_3, s_4, the system will deterministically transition to s_1, s_2, s_4, s_5, respectively.
2. Starting from state s_2, when t is an even number, the agent can choose to go to s_3 or s_5; when t is an odd number, the agent can only go to state s_3.
3. Starting from state s_5, the agent can choose to go to s_0 or s_2 if t is odd; the system can only go to s_0 if t is even.

Intuitively, there is a one-way 'bridge' between states s_2 and s_5 that shifts direction at each time step. It is easy to see that this instance has a small D^* diameter with $D^* \leq 5$.

Suppose the agent starts from state s_2 at $t = 0$. Then the optimal policy is to go to s_5, then keep returning between s_2 and s_5. Note that this is feasible because whenever the agent arrives in states s_2 and s_5, the bridge is always in the proper direction such that the agent can go through it. The total expected reward can be expanded as

$$V(\mathcal{I}, \pi^*, \delta_{s_2}) = \frac{\bar{r}T}{2}. \tag{B.4}$$

However, if the system starts from state s_4, whenever the agent arrives in states s_2 and s_5, the bridge is always in the opposite direction such that the agent can never use it. Then the optimal policy is to go to $s_5, s_0, s_1, \ldots$, with

$$V(\mathcal{I}, \pi^*, \delta_{s_4}) = \frac{\bar{r}T}{6}. \tag{B.5}$$

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

Hence, the difference in total expected reward with initial distributions δ_{s_2} and δ_{s_5} is

$$V(\mathcal{I}, \pi^*, \delta_{s_2}) - V(\mathcal{I}, \pi^*, \delta_{s_5}) = \frac{\bar{r}T}{3}, \tag{B.6}$$

which grows linearly in T despite the small D^* diameter.

Appendix C. Proofs of additional theoretical results

Proof of Lemma 1

Recall that $G \in \mathcal{G}_{\text{csl}}$ is strongly connected with classical graph diameter $d_c(G) < \infty$. For vertices $v, v' \in \mathcal{V}$, define $d_G(v, v')$ as the distance from vertex v to vertex v'. The classical diameter satisfies $d_c(G) = \max_{v,v' \in \mathcal{V}} d_G(v, v')$.

(1) First, we show that $\tau_0(\mathcal{I}_G) \geq d_c(G)$. By the definition of the classical diameter, there exist vertices $v_i, v_j \in \mathcal{V}$ with distance $d_G(v_i, v_j) = d_c(G)$. Consider the point measure on state i, ν with $\nu(i) = 1$, and the point measure on state j, ν' with $\nu'(j) = 1$. Starting from distribution ν, it takes at least $d_c(G)$ steps to achieve distribution ν', which implies that $\tau_0(\mathcal{I}_G) \geq d_c(G)$.

(2) Now, we prove that $\tau_0(\mathcal{I}_G) \leq 2d_c(G)$ analogously to the proof of Theorem 2. Recall that $G \in \mathcal{G}_{\text{csl}}$ has at least one self-loop. Without loss of generality, let $e_{11} \in \mathcal{E}$. For any pair of distributions ν, ν', we show that starting from the initial distribution ν, the system can reach ν' after $2d_c(G)$ steps. In particular, we consider the following policy:

Stage 1: If the initial state is 1, stay at state 1 for $d_c(G)$ steps until time $t = d_c(G)$; otherwise, if the initial state is i, $i \neq 1$, first go to state 1 using $d_G(i, 1)$ steps, and stay at state 1 for $(d_c(G) - d_G(i, 1))$ steps until time $t = d_c(G)$. This stage requires $d_c(G)$ steps.

Stage 2: Starting from state 1, we reach the distribution ν' in another $d_c(G)$ steps. Analogous to Stage 2 in the proof of Theorem 2, for any state $i \in \mathcal{S}$, we can take a $d_c(G)$-path to reach state i starting from state 1: first stay at state 1 for $(d_c(G) - d_G(1, i))$ steps, then go to state i using another $d_G(1, i)$ steps. In order to reach the distribution ν' in exactly $d_c(G)$ steps starting from state 1, with probability $\nu'(j)$ we take a $d_c(G)$-path to state j, for $j \in \mathcal{S}$. This stage requires $d_c(G)$ steps.

Hence, starting from an arbitrary distribution ν, the policy introduced above achieves any distribution ν' in $2d_c(G)$ steps, which implies that the 0-diameter is at most $2d_c(G)$, i.e. $\tau_0(\mathcal{I}_G) \leq 2d_c(G)$.

Combining (1) and (2) completes the proof of this lemma. □

Proof of Lemma 2

Fix $\epsilon \in [1 - (1 - \xi)^{\tau_0(\mathcal{I}_G)}, 1)$. It suffices to show that for any pair of distributions over the state space $\nu, \nu' \in \mathcal{P}$, ν' is ϵ-reachable from ν in $\tau_0(\mathcal{I}_G)$ steps.

Recall that when $\xi = 0$, the ξ-SGT instance becomes a DGT instance with 0-diameter $\tau_0(\mathcal{I}_G)$. Hence, for any pair of distributions ν, ν', there exists a policy $\pi_{\nu,\nu'}$ such that the state distribution transitions from ν to ν' precisely after $\tau_0(\mathcal{I}_G)$ steps.

We can formulate the following policy, $\pi^{\xi}_{\nu,\nu'}$, for the ξ-SGT instance: At each time step t, the agent makes the decision using the same policy as in the corresponding DGT instance, pretending $\xi = 0$, i.e.,

$$\pi^{\xi}_{\nu,\nu'}(t, S_t, Y^P) = \pi_{\mu_t,\nu'}(t, S_t, Y^P), \tag{C.1}$$

where μ_t is the state distribution at time t. In other words, at time t, the policy $\pi^{\xi}_{\nu,\nu'}$ makes the same decision as would $\pi_{\mu_t,\nu'}$.

When applying the policy $\pi^{\xi}_{\nu,\nu'}$ in the ξ-SGT instance, $\mathcal{I}^{\xi}_G$, during the first $\tau_0(\mathcal{I}_G)$ steps, let E be the event that the random perturbation never occurs, and $\bar{E}$ the event that the perturbation occurs in at least

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

one step. Recall that whether the perturbation occurs in each step is independent, the probability of the event E is at least $(1-\xi)^{\tau_0(\mathcal{I}_G)}$:

$$\mathbb{P}(E) \geq (1-\xi)^{\tau_0(I_G)}. \tag{C.2}$$

Therefore, with probability at least $(1-\xi)^{\tau_0(\mathcal{I}_G)}$, the state distribution becomes exactly ν' after $\tau_0(\mathcal{I}_G)$ steps under $\pi^{\xi}_{\nu,\nu'}$. Otherwise, the stochastic perturbation takes place in at least one step before the state distribution reaches ν'. Starting from ν at time $t=0$, the distribution at time $t=\tau_0(\mathcal{I}_G)$ under $\pi^{\xi}_{\nu,\nu'}$, denoted by $\tilde{\mu}_{\tau_0(\mathcal{I}_G)}$, can be expanded as

$$\begin{aligned}\tilde{\mu}_{\tau_0(\mathcal{I}_G)}(s) &= \mathbb{P}_{S_0\sim\nu}(S_{\tau_0(\mathcal{I}_G)}=s \mid E)\mathbb{P}(E) + \mathbb{P}_{S_0\sim\nu}(S_{\tau_0(\mathcal{I}_G)}=s \mid \bar{E})\mathbb{P}(\bar{E}) \\ &= \nu'(s)\mathbb{P}(E) + \mathbb{P}_{S_0\sim\nu}(S_{\tau_0(\mathcal{I}_G)}=s \mid \bar{E})\mathbb{P}(\bar{E}),\end{aligned} \tag{C.3}$$

for $s\in\mathcal{S}$, where Eq. (C.3) follows from the fact that the distribution of $S_{\tau_0(\mathcal{I}_G)}$ conditioned on the event E, i.e. conditioned on the event that no perturbation occurs, is exactly ν'. The total variation between $\tilde{\mu}_{\tau_0(\mathcal{I}_G)}$ and the target distribution ν' satisfies

$$\begin{aligned}\delta_{\mathrm{TV}}(\tilde{\mu}_{\tau_0(\mathcal{I}_G)},\nu') &= \frac{1}{2}\sum_{s\in\mathcal{S}}|\tilde{\mu}_{\tau_0(\mathcal{I}_G)}(s)-\nu'(s)| \\ &= \frac{1}{2}\sum_{s\in\mathcal{S}}|-\nu'(s)(1-\mathbb{P}(E)) + \mathbb{P}_{S_0\sim\nu}(S_{\tau_0(\mathcal{I}_G)}=s \mid \bar{E})\mathbb{P}(\bar{E})| \\ &= (1-\mathbb{P}(E))\cdot\frac{1}{2}\sum_{s\in\mathcal{S}}|\mathbb{P}_{S_0\sim\nu}(S_{\tau_0(\mathcal{I}_G)}=s \mid \bar{E})-\nu'(s)|\end{aligned} \tag{C.4}$$

$$\begin{aligned}&= (1-\mathbb{P}(E))\cdot\delta_{\mathrm{TV}}(\mathbb{P}_{S_0\sim\nu}(S_{\tau_0(\mathcal{I}_G)}=\cdot\,|\bar{E}),\nu') \\ &\leq 1-(1-\xi)^{\tau_0(\mathcal{I}_G)},\end{aligned} \tag{C.5}$$

where Eq. (C.4) follows from $\mathbb{P}(\bar{E})=1-\mathbb{P}(E)$, Eq. (C.5) from $\mathbb{P}(E)\geq(1-\xi)^{\tau_0(\mathcal{I}_0)}$ and $\delta_{\mathrm{TV}}(\cdot,\cdot)\leq 1$. Hence, for $\epsilon\geq 1-(1-\xi)^{\tau_0(\mathcal{I}_G)}$, we have shown that the policy $\pi^{\xi}_{\nu,\nu'}$ achieves the target distribution ν' in $\tau_0(\mathcal{I}_G)$ steps within a total variation distance ϵ, i.e. $\tau_\epsilon(\mathcal{I}_G^{\xi})\leq\tau_0(\mathcal{I}_G)$. This completes the proof. □

Proof of Lemma 3

In order to prove Lemma 3, we first introduce the following lemma.

Lemma 5. *Fix an undirected connected graph $G=(\mathcal{V},\mathcal{E})\in\mathcal{G}_{csl}$, and $\xi\in(0,\frac{1}{2})$. Let $\mathcal{I}_G$ be the DGT instance characterized by G, and $\mathcal{I}_G^{\xi}$ be the ξ-SGT instance described by (G,ξ). For any pair of vertices $v,v'\in\mathcal{V}$, denote by $d_G(v,v')$ the classical distance between v and v' on graph G. For a policy π, let $T^{\pi}_{v,v'}=\min\{t\geq 1:\ S_t=v' | S_0=v\}$ be the hitting time of v' starting from v under π. Then there exists a policy $\pi_{v,v'}$ such that for $t>(d_G(v,v')-1)/(1-2\xi)$,*

$$\mathbb{P}(T^{\pi_{v,v'}}_{v,v'}\leq t)\geq 1-\frac{16\xi(1-\xi)t}{((1-2\xi)t-(d_c(v,v')-1))^2}. \tag{C.6}$$

We first prove Lemma 5. Recall that G is undirected and connected. For $v,v'\in\mathcal{S}$, consider the following policy $\pi_{v,v'}$. Find the shortest path from v to v' on G, $p=(v_0,v_1,\ldots,v_{d-1},v_d)$, where $d=d_G(v,v')$, $v_0=v$, and $v_d=v'$. For each time step t, if the system state is on the path p, i.e., $S_t=v_i$ for some $i\in[d]$, the agent takes the action, $a_{v_i,v_{i+1}}$, to attempt to go to the next state, v_{i+1}, on the path p, and records the next state S_{t+1} in the actual path p'; if the system state $S_t\notin p$, then the agent attempts to go back to the path p by tracing back along the path p' with action a_{S_{t+1},S_t}, and records the next state S_{t+1} in p'.

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

Let W_i, $i = 1, 2, \ldots$ be binary random variables with $\mathbb{P}(W_i = 1) = 1 - \mathbb{P}(W_i = -1) = 1 - \xi$. Going from v to v' under policy $\pi_{v,v'}$ corresponds to the random walk $\sum_{i=1}^{t} W_t$. For each step t, the agent successfully achieves the intended next state with probability $1 - \xi$, in which case the agent is one step closer to v' under the policy $\pi_{v,v'}$, corresponding to $W_t = 1$; the agent goes elsewhere because of noise with probability ξ, in which case the system is one step farther from the target under $\pi_{v,v'}$, corresponding to one step back in the random walk, i.e., $W_t = -1$. Then for $t > \frac{d-1}{1-2\xi}$, we have

$$\mathbb{P}(T_{v,v'}^{\pi_{v,v'}} \geq t) = \mathbb{P}\left(\sum_{i=1}^{t} W_i \leq d - 1\right) \tag{C.7}$$

$$\leq \frac{16\xi(1-\xi)t}{((1-2\xi)t - (d-1))^2}, \tag{C.8}$$

where (C.8) follows from the Chebyshev inequality, which completes the proof of the lemma.

Now we prove Lemma 3 using Lemma 5. Given $\mu, \mu' \in \mathcal{P}$, let $\mu' = \mu F$, where the matrix $F \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ satisfies that $F_{i,j} \in (0, 1)$ for $i, j \in \mathcal{S}$, and $F\mathbf{1} = \mathbf{1}$. Let the initial state be drawn from μ, i.e., $S_0 \sim \mu$. We consider the following policy in order to achieve state distribution μ' starting from μ. If the initial state is $S_0 = i \in \mathcal{S}$, sample a random variable $J \in \mathcal{S}$ with $P(J = \cdot|S_0 = i) = F(i, \cdot)$. The agent then employs the policy $\pi_{i,J}$ described in the proof of Lemma 5 to attempt to go to state J, and stays at the target state J after reaching it. Note that this is feasible because each state has a noiseless self-loop. Hence, we have

$$\begin{aligned}\mathbb{P}(S_t = j|S_0 = i) &= \mathbb{P}(S_t = j, J = j|S_0 = i) + \mathbb{P}(S_t = j, J \neq j|S_0 = i)\\ &= \mathbb{P}(J = j|S_0 = i)\mathbb{P}(S_t = j|J = j, S_0 = i) + \mathbb{P}(S_t = j, J \neq j|S_0 = i)\\ &= F_{i,j}\mathbb{P}(T_{i,j}^{\pi_{i,j}} \leq t) + \mathbb{P}(S_t = j, J \neq j|S_0 = i) \qquad \text{(C.9)}\end{aligned}$$

$$\geq F_{i,j}\mathbb{P}(T_{i,j}^{\pi_{i,j}} \leq t), \tag{C.10}$$

where (C.9) follows from the definition of the matrix F and the hitting time $T_{i,j}^{\pi_{i,j}}$, and (C.10) from the fact that $\mathbb{P}(S_t = j, J \neq j|S_0 = i) \geq 0$. Recall that $d_c(G) = \max_{v,v'\in\mathcal{V}} d_G(v, v')$ is the classical diameter of graph G. Then for $t > (d_c(G) - 1)/(1 - 2\xi)$ and $j \in \mathcal{S}$, the state distribution at time t satisfies

$$\begin{aligned}\mu_t(j) &= \mathbb{P}_{S_0\sim\mu}(S_t = j)\\ &= \sum_{i\in\mathcal{S}} \mathbb{P}(S_t = j|S_0 = i)\mathbb{P}(S_0 = i)\\ &\geq \sum_{i\in\mathcal{S}} F_{i,j}\mathbb{P}(T_{i,j}^{\pi_{i,j}} \leq t)\mu(i) \qquad \text{(C.11)}\end{aligned}$$

$$\begin{aligned}&= \mathbb{P}(T_{i,j}^{\pi_{i,j}} \leq t)\sum_{i\in\mathcal{S}} F_{i,j}\mu(i)\\ &= \mathbb{P}(T_{i,j}^{\pi_{i,j}} \leq t)\mu'(j) \qquad \text{(C.12)}\end{aligned}$$

$$\geq \left(1 - \frac{16\xi(1-\xi)t}{((1-2\xi)t - (d_c(i,j) - 1))^2}\right)\mu'(j) \tag{C.13}$$

$$\geq \left(1 - \frac{16\xi(1-\xi)t}{((1-2\xi)t - (d_c(G) - 1))^2}\right)\mu'(j). \tag{C.14}$$

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

Here, (C.11) follows from (C.10), (C.12) from $\mu F = \mu'$, (C.13) from Lemma 5, and (C.14) from the fact that $t > (d_c(G) - 1)/(1 - 2\xi)$ and $d_c(G) \geq d_G(i, j)$. Thus, the total variation distance between μ_t and the target distribution μ' satisfies

$$\delta_{\mathrm{TV}}(\mu_t, \mu') = \sum_{j \in \mathcal{S}: \mu'(j) \geq \mu_t(j)} (\mu'(j) - \mu_t(j)) \tag{C.15}$$

$$\leq \sum_{j \in \mathcal{S}: \mu'(j) \geq \mu_t(j)} \frac{16\xi(1-\xi)t}{((1-2\xi)t - (d_c(G)-1))^2} \mu'(j) \tag{C.16}$$

$$\leq \frac{16\xi(1-\xi)t}{((1-2\xi)t - (d_c(G)-1))^2}, \tag{C.17}$$

where (C.15) follows from the definition of total variation, (C.16) from (C.14), and (C.17) from the fact that $\sum_{j \in \mathcal{S}: \mu'(j) \geq \mu_t(j)} \mu'(j) \leq 1$. With (C.17), we have $\delta_{\mathrm{TV}}(\mu_t, \mu') \leq \epsilon$ when

$$t \geq \frac{d_c(G)}{1-2\xi} + \frac{4\xi(1-\xi)}{\epsilon(1-2\xi)^2} \left(2 + \sqrt{4 + \frac{(1-2\xi)d_c(G)\epsilon}{\xi(1-\xi)}}\right). \tag{C.18}$$

Therefore,

$$\tau_\epsilon(\mathcal{I}_G^\xi) \leq \frac{d_c(G)}{1-2\xi} + \frac{4\xi(1-\xi)}{\epsilon(1-2\xi)^2} \left(2 + \sqrt{4 + \frac{(1-2\xi)d_c(G)\epsilon}{\xi(1-\xi)}}\right) \tag{C.19}$$

$$\leq \frac{d_c(G)}{1-2\xi} + \frac{4\xi(1-\xi)}{\epsilon(1-2\xi)^2} \left(2 + \sqrt{4 + \frac{(1-2\xi)d_c(G)}{\xi(1-\xi)}}\right), \tag{C.20}$$

with (C.20) following from $\epsilon \leq 1$,which completes the proof of Lemma 3. □

Proof of Theorem 3

Denote by $\mathcal{I}$ an instance of dynamic energy management with storage. With (25) and Lemma 3, we have $\tau_\epsilon(\mathcal{I}) \leq (B/C + 1)/(1 - 2\beta) + (f(B/C + 1, \beta))/\epsilon = \alpha + \omega/\epsilon$. By Theorem 1, we have

$$\Delta(\mu_0, T) \leq \frac{\bar{r}}{1-\epsilon} \left(\alpha + \frac{\omega}{\epsilon}\right). \tag{C.21}$$

Now we treat B, C, and β as fixed parameters determined by the system and the environment, and regard ϵ as a parameter we can tune. It is easy to see that

$$\frac{d}{d\epsilon} \left(\frac{\bar{r}}{1-\epsilon} \left(\alpha + \frac{\omega}{\epsilon}\right)\right) = \bar{r} \left(\frac{\alpha\epsilon^2 + 2\omega\epsilon - \omega}{\epsilon^2(1-\epsilon)^2}\right). \tag{C.22}$$

Hence, the right-hand side of (C.21) is minimized at $\epsilon = (\sqrt{\omega^2 + \omega\alpha} - \omega)/\alpha$, which implies that

$$\Delta(\mu_0, T) \leq \bar{r}(\sqrt{\omega + \alpha} + \sqrt{\omega})^2. \tag{C.23}$$

□

https://doi.org/10.1017/S0269964821000206 Published online by Cambridge University Press

Appendix D. System specifications

For the simulations in Section 4.5, we used a desktop computer with an Intel Core i7-8700k CPU and 32GB of memory. The operating system is Windows 10 and the Matlab version is R2016b.

Cite this article: Song R and Xu K (2022). Temporal concatenation for Markov decision processes. *Probability in the Engineering and Informational Sciences* **36**, 999–1026. https://doi.org/10.1017/S0269964821000206