Preprint of article to appear in Mathematics of Operations Research, 2025.

https://pubsonline.informs.org/journal/moor

MATHEMATICS OF OPERATIONS RESEARCH

Vol. 00, No. 0, Xxxxx 0000, pp. 000–000

Submitted to *Mathematics of Operations Research*

ISSN 0364-765X, EISSN 1526-5471

Upper Bounds for All and Max-gain Policy Iteration Algorithms on Deterministic MDPs

Ritesh Goenka
Mathematical Institute, University of Oxford, Oxford OX2 6GG, United Kingdom, goenka@maths.ox.ac.uk

Eashan Gupta†
Department of Computer Science, University of Illinois at Urbana-Champaign, Champaign, Illinois 61820, USA, eashang2@illinois.edu

Sushil Khyalia†
Machine Learning Department, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, USA, skhyalia@andrew.cmu.edu

Shivaram Kalyanakrishnan
Department of Computer Science and Engineering, Indian Institute of Technology Bombay, Mumbai 400076, India, shivaram@cse.iitb.ac.in

† denotes equal contributions.

Abstract. Policy Iteration (PI) is a widely used family of algorithms to compute optimal policies for Markov Decision Problems (MDPs). Howard's PI is one of the most commonly used algorithms from this family. Despite its popularity, theoretical analysis of the running time complexity of Howard's PI has remained elusive. For n-state, 2-action MDPs, the best known lower and upper bounds are $\Omega(n)$ and $O(2^n/n)$ iterations, respectively. Based on computational evidence for a combinatorial relaxation of this problem, Hansen and Zwick (2012) conjectured that the upper bound can be improved to $O(\phi^n)$, where $\phi = (1+\sqrt{5})/2$ is the golden ratio. We prove this conjecture for Deterministic MDPs (DMDPs), albeit up to a poly(n) factor.

More generally, we derive a non-trivial upper bound for DMDPs that applies to the entire family of PI algorithms. We also derive an improved bound that applies to all "max-gain" switching variants. These bounds hold both under discounted and average reward settings. Combined with a result of Melekopoglou and Condon (1994), our results imply that stochasticity makes 2-action MDPs harder to solve for PI. Our analysis is based on certain graph-theoretic results, which may be of independent interest.

Key words: Markov decision problem, Deterministic MDP, Policy improvement, Policy iteration, Computational complexity

MSC2000 subject classification: Primary: 90C40, 68Q25; Secondary: 05C35, 05C38

OR/MS subject classification: Primary: dynamic programming/optimal control/Markov/finite state, analysis of algorithms/computational complexity; Secondary: programming/linear, networks/graphs/theory

1. Introduction.

A Markov Decision Problem (MDP) (Puterman [49]) is an abstraction of a decision-making task in which the effect of any given *action* from any given *state* is a stochastic transition to a next state, coupled with a numeric reward. A *policy* (taken in this article to be stationary and deterministic) for an MDP specifies the action to take from each state. The utility of a policy is usually taken as some form of the expected *long-term* reward it yields. Two common definitions of long-term reward are as a discounted sum of the individual rewards over an infinite horizon, and as the limiting average reward. We focus on the discounted reward setting though our results (theorems 1 and 2) also hold for the average reward setting, as we shall discuss later in Section 4.3. For an MDP with a finite number of states and actions, the set of policies is also finite, and this set contains an *optimal* policy, which maximises the expected long-term reward starting from each state in the MDP (Puterman [49, Theorem 6.2.10]; see also Bellman [11, Chapter XI]).

For a given MDP—specified by its sets of states and actions, transition probabilities, rewards, and discount factor—the desired solution is an optimal policy for the MDP.

Policy Iteration (PI) (Howard [34]) is a widely used family of algorithms to solve MDPs. A PI algorithm is initialised with some arbitrary policy, and iterates through a sequence of policies that is guaranteed to terminate in an optimal policy. In each iteration, a set of "improving" actions is identified for each state. Any policy obtained by switching one or more of the current policy's actions to improving actions is guaranteed to dominate the current policy, and hence can be selected as the subsequent iterate. A policy with no improving actions is guaranteed to be optimal. Algorithms from the PI family are distinguished by their choice of improving actions for switching to at each step. An alternative perspective of PI algorithms emerges by considering a Linear Program (LP) P_M induced by the input MDP M (Puterman [49, see Section 6.9.1]). The vertices of the feasible polytope of P_M are in bijective correspondence with policies for M. PI algorithms restricted to changing the action only at a single state essentially perform a Simplex update on the feasible polytope. On the other hand, the generic PI update, which could involve changing actions at multiple states, amounts to a block-pivoting step. The classical simplex method of Dantzig with most-negative-reduced-cost pivoting rule [14] and Howard's PI [34] are arguably the most commonly used variants of PI.

More generally, linear programming, policy iteration, and value iteration are the three major approaches to computing optimal policies for MDPs (Puterman [49]). It is well known that solving MDPs is P-complete (Papadimitriou and Tsitsiklis [46]). The natural next question is whether there exists a *strongly polynomial* algorithm for solving MDPs: this means that if any arithmetic or relational operation can be performed in constant time, regardless of the size of the operands, then the total number of such operations required is at most a polynomial in the number of states and actions, with no dependence on other input parameters. Ye [59] showed that Dantzig's simplex algorithm with most-negative-reduced-cost pivoting rule and Howard's PI are strongly polynomial for MDPs with a fixed discount factor. On the other hand, Feinberg and Huang [23] showed that value iteration is not strongly polynomial for MDPs with a fixed discount factor. Moreover, Fearnley [22] and Hollanders *et al.* [32] showed that Howard's PI is exponential (in particular, not strongly polynomial) for MDPs with a general discount factor. Let n denote the number of states and k denote the maximum number of actions per state in the input MDP. Notably, in the constructions of Fearnley [22] and Hollanders *et al.* [32], k grows to infinity as n goes to infinity. Since many natural tasks can be modelled as MDPs with at most a constant number of actions at each state, the following question emerges.

QUESTION 1. Does Howard's PI converge in poly(n) steps for n-state MDPs with a maximum of k actions at each state, where k is a fixed constant?

Consider an MDP with $n \geq 2$ states and $k \geq 2$ actions, with the convention that each of the k actions is available from each state. This convention is justified because if there are less than k available actions at a state, then the existing actions at that state can be duplicated to ensure exactly k available actions. The total number of policies, k^n, is a trivial upper bound on the number of iterations taken by any PI algorithm. Mansour and Singh [41] showed that this trivial bound can be improved to $O(k^n/n)$. In the simplest case $k = 2$, one can identify the set of actions at each state with $\{0, 1\}$ so that each policy corresponds to an n-bit binary vector. Hansen [28] showed that when these binary vectors associated with the policies visited by Howard's PI are arranged as rows of a binary matrix, the resulting matrix satisfies the so-called *order regularity* property (Hollanders *et al.* [33, Definition 1]). Moreover, they made the following conjecture (they stated it for acyclic unique sink orientations but we restrict to 2-action MDPs in the statement below).

CONJECTURE 1 **(Conjecture 3.3.2, Hansen [28])**. *For $n \in \mathbb{N}$, let $f(n)$ denote the maximum number of rows in any order regular matrix with n columns. Then*

(1) $f(n) = \mathrm{Fib}(n+2)$, where $\mathrm{Fib}(m)$ denotes the m-th Fibonacci number, and

(2) consequently, the number of steps taken by Howard's PI to find the optimal policy for 2-action MDPs is $O(\phi^n)$, where $\phi = (1+\sqrt{5})/2$ is the golden ratio.

Hollanders *et al.* [33] disproved the first part of Conjecture 1 by showing that $f(7) = 33 < 34 = \mathrm{Fib}(9)$. However, it is still possible that $f(n) \leq \mathrm{Fib}(n+2)$ so that the second part of Conjecture 1 is true. In this paper, we consider Conjecture 1 in the context of Deterministic MDPs (DMDPs)—MDPs in which the transitions are all deterministic. In other words, for every state s and action a in a DMDP, there is a unique state s' which is reached whenever a is taken from s. A special case of our main result (Theorem 1) settles Conjecture 1 for DMDPs, albeit up to a poly(n) factor.

THEOREM 1. *The number of iterations taken by any PI algorithm on any n-state, k-action DMDP is at most*

$$5n^3k^2 \cdot \alpha(k)^n, \text{ where } \alpha(k) = \frac{k-1+\sqrt{(k-1)^2+4}}{2}. \tag{1}$$

In particular, Howard's PI takes $O(n^3 \cdot \phi^n)$ steps to find an optimal policy for 2-actions DMDPs.

With the challenge of stochasticity removed, DMDPs would appear to be an easier class of problems to solve than MDPs—and several results affirm this intuition. For example, solving

DMDPs is in NC (Papadimitriou and Tsitsiklis [46]). Moreover, it has been established that DMDPs can be solved in strongly polynomial time. Madani *et al.* [40] propose a specialised algorithm for DMDPs that enjoys a strongly polynomial upper bound, while Post and Ye [48] establish that the classical simplex method of Dantzig also runs in strongly polynomial time on DMDPs.

By upper-bounding the complexity of specific algorithms on DMDPs, the preceding results indirectly upper-bound the running time of the *best* algorithm from some corresponding class of algorithms. In Theorem 1, we adopt a complementary perspective as we derive running-time upper bounds that apply to the entire family of PI algorithms, and hence to the *worst* among them. The significance of Theorem 1 is most apparent for the special case of $k = 2$ actions. Melekopoglou and Condon [44] constructed an n-state, 2-action MDP on which a specific variant of PI visits all the 2^n policies. On the other hand, our result establishes that no PI algorithm can exceed $\text{poly}(n) \cdot \phi^n$ iterations on any n-state, 2-action DMDP. Therefore, we conclude that stochasticity makes 2-action MDPs harder to solve for PI. In other words, the worst PI algorithm takes strictly longer to solve 2-action MDPs than 2-action DMDPs. This observation is also consistent with our current understanding of the best (PI) algorithms for MDPs: strongly polynomial (PI) algorithms are known for DMDPs but not for MDPs.

The "LP digraph" (Avis and Moriyama [10]) of an LP has vertices corresponding to the vertices of the feasible polytope, and directed edges from vertices to neighbours that improve the objective function. A direct consequence of Theorem 1 is that the upper bound (1) also holds for the length of the longest directed path in the LP digraph induced by the LP P_M arising from any n-state k-action DMDP M. This is interesting since the length of the longest path in the LP digraph is an intrinsic characteristic of the LP, and not dependent on any specific algorithm.

When a state has multiple improving actions, one common rule to select the action to switch to is based on the actions' "gains". The gain of an action a is the difference in utility arising from replacing the current policy with a for only the very first time step. Switching to an action with the maximum gain ("max-gain") plays a role in the proof of the strongly polynomial upper bound for the simplex method with Dantzig's pivoting rule on DMDPs (Post and Ye [48]). Max-gain action selection has also been observed to be efficient in practice with other PI variants (Taraviya and Kalyanakrishnan [57]). We obtain a smaller upper bound than (1) for sufficiently large k when we restrict the PI algorithm to perform max-gain action selection (while the algorithm is still free to select on which *states* to switch actions).

THEOREM 2. *The number of iterations taken by any max-gain PI algorithm on any n-state, k-action DMDP is at most*

$$(n+1)n^2k(k+1)!^{(n-1)/(k+1)} = O\left(n^3 \cdot \left(\frac{k}{e}\left(1+O\left(\frac{\log k}{k}\right)\right)\right)^n\right). \tag{2}$$

For the specific case of Howard's PI on DMDPs, theorems 1 and 2 give a non-trivial improvement over Mansour and Singh's [41] bound. However, these bounds are still exponentially far from the much stronger conjectured upper bound of nk steps by Hansen and Zwick [31].

We briefly sketch our proofs for theorems 1 and 2. Every DMDP M induces a directed multigraph G_M in which the vertices are the states, and the edges are the actions in M. In turn, each policy induces a subgraph of G_M, with the restriction of having a single outgoing edge from each vertex. For each state (equivalently vertex) s, the long-term reward accrued by a policy is fully determined by a directed path starting at s, and a directed cycle that the path reaches. Since each PI iterate strictly dominates the preceding one, it follows that the digraph induced by any newly visited policy contains a path-cycle that does not appear in any of the previous policy digraphs. Following this, we bound the number of path-cycles in two suitable modifications of G_M by proving new bounds on the number of cycles in such digraphs (theorems 3 and 4). Correspondingly, we obtain upper bounds on the running time of PI algorithms: (1) for arbitrary PI variants, and (2) for variants that perform "max gain" action selection.

THEOREM 3. *The maximum number of cycles in any multi-digraph on n vertices with outdegree k such that each multi-edge has multiplicity at most $k-1$ is $\Theta(\alpha(k)^n)$, where $\alpha(k)$ is as in* (1)*.*

THEOREM 4. *The number of cycles in any simple digraph on n vertices with outdegree at most k is bounded above by $(k+1)!^{n/(k+1)}$.*

We restate and prove the above theorems in Section 3 as theorems 7 and 6, respectively. The problem of bounding the number of cycles in graphs has been considered by several authors previously (see Section 3 for a brief literature review). However, to the best of our knowledge, this is the first work to consider this problem on digraphs with degree constraints. Our main contribution is Theorem 3, a special case of which gives an asymptotically tight "Fibonacci" upper bound on the number of cycles in a 2-regular digraph on n vertices. The proof of this theorem required some new ideas. In particular, we use a hybrid proof approach, namely, enumeration if the graph has a specific structure, and induction otherwise. Dvořák *et al.* [20] proved a more general version of Theorem 4 for the case of undirected graphs, and the same proof idea also works for digraphs.

The article is organised as follows. We provide requisite definitions and background along with some related work in Section 2, before proving our graph-theoretic results in Sections 3 and running-time complexity results for PI in Section 4. Finally, we end with some concluding remarks and directions for future work in Section 5.

2. Policy Iteration.

We begin by defining MDPs, and thereafter describe the PI family of algorithms for solving them. Next, we present known upper bounds on the number of steps taken by PI to find an optimal policy.

2.1. Markov decision problems.

DEFINITION 1. An MDP is a 5-tuple (S, A, T, R, γ), where S is a set of states; A is a set of actions; $T : S \times A \times S \to [0, 1]$ is the transition function with $T(s, a, s')$ being the probability of reaching state $s' \in S$ by taking action $a \in A$ from state $s \in S$ (hence $\sum_{s'} T(s, a, s') = 1$); $R : S \times A \to \mathbb{R}$ is the reward function with $R(s, a)$ being the expected reward obtained on taking action $a \in A$ from state $s \in S$; and $\gamma \in [0, 1)$ is the discount factor.

Given an MDP $M = (S, A, T, R, \gamma)$, a *policy* (assumed deterministic and Markovian) $\pi : S \to A$ specifies an action $a \in A$ for each state $s \in S$. We denote the set of all policies for a given MDP M by Π_M (or just Π if the underlying MDP M is evident from the context). In this work, we assume that S and A are finite, with $|S| = n \geq 2$ and $|A| = k \geq 2$. Consequently, Π is also finite, and contains k^n policies.

DEFINITION 2. For policy $\pi \in \Pi$, the *value function* $V^\pi : S \to \mathbb{R}$ gives the expected infinite discounted reward obtained by starting from each state $s \in S$ and following the policy π. Let $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$ be the state-action-reward trajectory generated by M over time (the subscript indicates the time step). Then for $s \in S$,

$$V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)\right],$$

where $s_0 = s$, and for $t \geq 0$, $a_t = \pi(s_t), s_{t+1} \sim T(s_t, a_t)$. The *action value function* $Q^\pi : S \times A \to \mathbb{R}$ applied to $s \in S, a \in A$ is the expected infinite discounted reward obtained by starting from s taking a, and thereafter following π. Finally, the *gain function* $\rho^\pi : S \times A \to \mathbb{R}$ provides the difference between Q^π and V^π.

All three functions, V^π, Q^π, and ρ^π, can be computed efficiently (in poly(n,k) arithmetic operations) by solving the following equations for $s \in S, a \in A$.

$$V^\pi(s) = R(s,\pi(s)) + \gamma \sum_{s' \in S} T(s,\pi(s),s')V^\pi(s').$$
$$Q^\pi(s,a) = R(s,a) + \gamma \sum_{s' \in S} T(s,a,s')V^\pi(s').$$
$$\rho^\pi(s,a) = Q^\pi(s,a) - V^\pi(s).$$

The first set of equations above, used to compute V^π for some fixed policy $\pi \in \Pi$, are called the Bellman equations for π.

We now define relations $\preceq$ and $\prec$ to compare policies in Π.

DEFINITION 3. For $\pi_1, \pi_2 \in \Pi$, $\pi_1 \preceq \pi_2$ if $V^{\pi_1}(s) \leq V^{\pi_2}(s)$ for all $s \in S$. Moreover, $\pi_1 \prec \pi_2$ if $\pi_1 \preceq \pi_2$ and additionally $V^{\pi_1}(s) < V^{\pi_2}(s)$ for some state $s \in S$.

DEFINITION 4. An policy $\pi^\star \in \Pi$ is called an *optimal* policy if $\pi \preceq \pi^\star$ for all $\pi \in \Pi$.

2.2. Policy improvement.

DEFINITION 5. For policy $\pi \in \Pi$, the *improvable set* I^π is defined as the set of state-action pairs $(s,a) \in S \times A$ such that $\rho^\pi(s,a) > 0$. A set $I \subseteq I^\pi$ is said to be a *valid* improvement set for π if for each $s \in S$, there exists at most one action $a \in A$ such that $(s,a) \in I$, and moreover, $|I| \geq 1$.

DEFINITION 6. Suppose that for policy $\pi \in \Pi$, the set I^π is non-empty. Fix an arbitrary, valid improvement set $I \subseteq I^\pi$. Consider policy $\pi' \in \Pi$, given by

$$\pi'(s) = \begin{cases} a, & \text{if } (s,a) \in I, \\ \pi(s), & \text{otherwise.} \end{cases} \tag{3}$$

Then π' is called a *locally-improving* policy of π. The operation of obtaining π' from π, by switching to corresponding actions in the improvement set I, is called *policy improvement*.

Notice that if $|I^\pi| > 1$, there are multiple possible choices of valid improvement sets $I \subseteq I^\pi$. The well-known policy improvement theorem, stated below, provides a guarantee that applies to every such choice of I.

THEOREM 5. *Fix $\pi \in \Pi$. (1) If $I^\pi = \emptyset$, then π is an optimal policy. (2) If $I^\pi \neq \emptyset$, let $\pi' \in \Pi$ be obtained from policy improvement to π using any valid improvement set $I \subseteq I^\pi$. Then $\pi \prec \pi'$, and moreover, for $s \in S$ such that $\pi'(s) \neq \pi(s)$, we have $V^{\pi'}(s) > V^\pi(s)$.*

We omit the proof of the theorem, which is verifiable from standard references (Puterman [49, see Section 6.4.2], Szepesvári [55, see Appendix A.2]). The theorem establishes the existence of an optimal policy for every MDP. Indeed "solving" an MDP amounts to computing an optimal policy for it. Although there are many possible solution techniques, a natural approach is evident from the theorem itself: to iterate through policies. Algorithms from the PI family proceed through a sequence of policies $\pi_0 \to \pi_1 \to \cdots \to \pi_\ell$, wherein $\pi_0 \in \Pi$ is an arbitrary initial policy; π_ℓ for some $\ell \geq 1$ is an optimal policy; and for $0 \leq i < \ell$, π_{i+1} is obtained from policy improvement on π_i.

Given any MDP M, we can now define its Policy Improvement Directed Acyclic Graph (PI-DAG) as follows: it is a directed graph with Π as its set of vertices, and there is an edge from π to π' if π' can be obtained from π by a single step of policy improvement. Note that the resulting digraph is acyclic since policy improvement always yields a strictly dominating policy. Further, note that directed paths starting at some vertex and ending at a sink vertex in the PI-DAG are in bijective correspondence with the possible trajectories of PI algorithms on the MDP M. Therefore, Theorem 1 can be reinterpreted as an upper bound on the length of directed paths in the PI-DAG of an n-state k-action DMDP. And since the corresponding LP digraph is a subgraph of the PI-DAG, this bound also applies to directed paths in the LP digraph, as noted previously in the introduction. See Figure 1 for an example MDP, its PI-DAG, and the corresponding LP digraph.

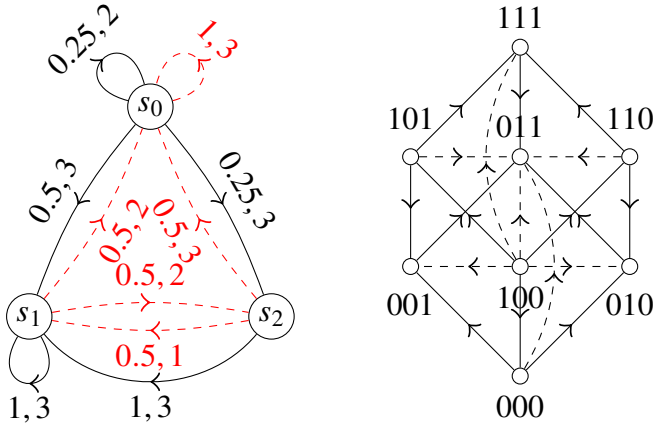

FIGURE 1. An example of a 3-state 2-action MDP with $\gamma = 0.9$ (left) and its PI-DAG (right). In the left figure, the dashed (red) edges and solid (black) edges correspond to actions 0 and 1, respectively. Each transition in the MDP is marked with its (transition probability, reward) pair. In the right figure, the solid and dashed edges correspond to policy improvement steps that switch action on a single state and multiple states, respectively. The digraph on the right induced only by the solid edges is the corresponding LP digraph, which is clearly a subgraph of the PI-DAG.

2.3. Switching rules. Variants from the PI family are distinguished by their "switching rule"—in other words their choice of valid improvement set $I \subseteq I^\pi$ to improve the current policy π. In principle, this choice can depend on the entire sequence of policies visited yet, along with any accompanying information gathered from each iteration. However, most common variants of PI are "memoryless": that is, they select the improvement set $I \in I^\pi$ solely based on I^π, and sometimes with additional knowledge of $\rho^\pi(s,a)$ for $(s,a) \in I^\pi$. For our purposes, it is convenient to view switching rules as a sequence of two steps: the first to select which *states* will be given new actions, and the second to select improving actions for each of these states. Concretely, let $S^+(\pi)$ denote the set of all states $s \in S$ for which there exists some action $a \in A$ such that $(s,a) \in I^\pi$. For each state $s \in S^+(\pi)$, let $A^+(\pi,s)$ be the set of actions $a \in A$ such that $(s,a) \in I^\pi$. Any switching rule must select a non-empty subset $S_{\text{switch}} \subseteq S^+(\pi)$, and for each $s \in S_{\text{switch}}$, select an action $a_{\text{switch}}(s) \in A^+(\pi,s)$. Note that in general, S_{switch} and $a_{\text{switch}}(\cdot)$ can both be random. Also, note that $a_{\text{switch}}(s)$ is trivially determined for each $s \in S_{\text{switch}}$ when the MDP has only $k = 2$ actions.

In our upcoming analysis of the number of iterations taken by PI on DMDPs, we place no restriction on how S_{switch} is selected from $S^+(\pi)$. However, we consider two distinct settings for action selection. (1) With *arbitrary* action selection, there is no restriction on how $a_{\text{switch}}(s)$ is selected from $A^+(\pi,s)$ for $s \in S_{\text{switch}}$. (2) Under *max-gain* action selection, we have $a_{\text{switch}}(s) \in \arg\max_{a \in A^+(\pi,s)} \rho^\pi(s,a)$ for $s \in S_{\text{switch}}$, with arbitrary tie-breaking. The max-gain action selection rule is used widely in practice. To the best of our knowledge, existing upper bounds on the complexity of PI on MDPs (presented shortly) all assume some constraint on the state selection step in the switching rule. Since we place no such restriction, our upper bounds when action selection is arbitrary apply to *every* PI algorithm, including those whose switching choices depend on memory and additional information.

2.4. Known results on complexity. We briefly review results on the running time of PI, restricting ourselves to bounds that depend only on the number of states $n \geq 2$ and actions $k \geq 2$ in the input MDP. Arguably the most common variant from the PI family is Howard's PI [34], under which $S_{\text{switch}} = S^+(\pi)$. Mansour and Singh [41] show an upper bound of $O(k^n/n)$ iterations when Howard's PI is coupled with arbitrary action selection; Taraviya and Kalyanakrishnan [57] obtain a tighter bound of $\left(O(\sqrt{k \log k})\right)^n$ iterations when action selection is random. Mansour and Singh [41] also propose a randomised PI variant in which S_{switch} is chosen uniformly at random from among the non-empty subsets of $S^+(\pi)$. They give an upper bound of $O\left(\left((1+\frac{2}{\log k})\frac{k}{2}\right)^n\right)$

iterations (with high probability) when action selection is arbitrary. Their bound of $O(2^{0.78n})$ expected iterations for the special case of $k = 2$ was subsequently improved by Hansen *et al.* [30] to $O(n^5)\left(\frac{3}{2}\right)^n$.

PI variants that switch only a single state in each iteration (that is, which enforce $|S_{\text{switch}}| = 1$) may be interpreted as variants of the Simplex algorithm, being run on an LP P_M induced by the input MDP M. Vertices in the feasible polytope of P_M are in bijective correspondence with the set of policies Π; single switch policy improvements amount to shifting to a neighbouring vertex that increases the objective function, which at the vertex corresponding to policy $\pi \in \Pi$ is $\sum_{s\in S} V^\pi(s)$. Suppose the set of states S is indexed; without loss of generality take $S = \{1, 2, \ldots, n\}$. Kalyanakrishnan et al. [36] show an upper bound of $(2+\ln(k-1))^n$ expected iterations for a variant of PI in which $S_{\text{switch}} = \{\max_{s\in S^+(\pi)} s\}$, and action selection is random. Interestingly, Melekopoglou and Condon [44] show that the same rule results in a policy improvement sequence of length 2^n for an n-state, 2-action MDP. Among deterministic variants of PI, the best known upper bound is $\text{poly}(n, k) \cdot k^{0.7207n}$ iterations (Gupta and Kalyanakrishnan [27]), for a variant that is based on "batch-switching" PI (Kalyanakrishnan *et al.* [35]).

The upper bounds listed above for MDPs also apply to DMDPs. However, a much stronger result has been shown when PI is applied to DMDPs. Post and Ye [48] demonstrate that the max-gain variant of the Simplex indeed terminates after a polynomial number of iterations on n-state, k-action DMDP. In this variant, the improvement set is $\{(\bar{s}, \bar{a})\}$, where $\bar{s}, \bar{a} \in \arg\max_{(s,a)\in S\times A} \rho^\pi(s, a)$, with ties broken arbitrarily. As we shall see in Section 4, every DMDP M induces a directed multigraph G_M, with each policy inducing a subgraph that is guaranteed to contain a directed cycle. Post and Ye establish that the max-gain Simplex algorithm registers a significant jump in the objective function when proceeding from π to π' if some cycle induced by π is *not* induced by π'. Moreover, such a break of a cycle must occur within a polynomial number of iterations, resulting in an overall upper bound of $O(n^5k^2\log^2 n)$ iterations for the max-gain Simplex algorithm. This upper bound has subsequently been improved by a factor of n (Hansen *et al.* [29]) and also generalised (Scherrer [52]). Even if the max-gain simplex algorithm is strongly polynomial for DMDPs, there do exist PI variants that are exponentially lower-bounded. Ashutosh *et al.* [9] construct an n-state, k-action whose PI-DAG has a path of length $\Omega(k^{n/2})$.

Unlike preceding analyses to obtain upper bounds for DMDPs (Post and Ye [48], Hansen *et al.* [29], Scherrer [52]), ours does not principally rely on bounding the change in continuous quantities such as the objective function of policies. Rather, our arguments are based on bounding discrete quantities: the number of directed cycles induced by policies in certain subgraphs of G_M.

3. Number of Cycles in Digraphs. In this section, we present results on the maximum number of cycles in directed multigraphs with certain constraints on degree and edge multiplicity.

The problem of bounding the number of cycles in graphs has a long history. Bounds have been established in terms of several basic graph parameters, including the number of edges, vertices, degree sequence, minimum/maximum/average degree, and cyclomatic number (Ahrens [1], Arman and Tsaturian [8], Dvořák *et al.* [20], Entringer and Slater [21], Guichard [26], Volkmann [58]). Furthermore, restrictions of this problem to specific classes of graphs are also well studied: planar graphs (Aldred and Thomassen [4], Alt *et al.* [6], Buchin *et al.* [13], Dvořák *et al.* [20]), graphs with forbidden subgraphs (Morrison *et al.* [45]), Hamiltonian graphs (Rautenbach and Stella [50], Shi [53]), triangle-free graphs (Arman *et al.* [7], Durocher *et al.* [19]), random graphs (Takács [56]), bipartite graphs (Alt *et al.* [6]), k-connected graphs (Knor [38]), grid graphs (Alt *et al.* [6]), outerplanar and series–parallel graphs (Mier and Noy [15]), complement of a tree (Reid [51], Zhou [60]), 3-connected cubic (Hamiltonian) graphs (AlBdaiwi [2], Aldred and Thomassen [3]), and 3-colorable triangulated graphs (Alt *et al.* [6]).

Analogous literature for digraphs is relatively sparse. Yoshua Perl [47] proved bounds for directed multigraphs with a fixed number of edges. Some restrictions to specific classes of digraphs have also been studied: digraphs with large girth (Allender [5]) and digraphs with restricted cycle lengths (Gerbner *et al.* [24]). A few other studies have considered the very closely related question of bounding the number of paths between two given vertices in the digraph: (acyclic) simple digraphs with a given number of edges (Delivorias and Richter [16], Perl [47]) and simple acyclic digraphs with a given number of vertices and edges (Golumbic and Perl [25]).

We consider directed multigraphs with restrictions on degree and edge multiplicity. To the best of our knowledge, this is the first work to consider such families of digraphs in the context of bounding the number of cycles. We prove two results, Theorem 6 and Theorem 7. The former is a straightforward adaptation of a result by Dvořák *et al.* [20, Theorem 5]. The latter is our main contribution and its proof required some new ideas. In particular, many of the results mentioned above either use an enumerative approach, an inductive approach, or a structural approach. For our problem, it seems hard to use a purely enumerative or structural approach. And the natural inductive approach quickly runs into the issue that the smaller graphs obtained during the proof do not belong to the same family of graphs. We are able to circumvent this issue by separating our analysis into two cases: use an enumerative approach if the graph has a certain structure (which interestingly coincides with that of an asymptotically extremal example), and use an inductive

approach otherwise. We remark that both the problems we consider are directly motivated by their application to PI on DMDPs, although they might find alternate applications.

In Section 4 (in particular, lemmas 5 and 6), we illustrate how the bounds on the number of cycles can be used to establish bounds on the number of paths and path-cycles (see Section 4 for definition) in the considered digraphs. Before we prove the main results of this section, we provide some basic definitions and set up some notation.

3.1. Definitions. We use the term *digraph* to refer to directed graphs possibly containing multi-edges and self-loops. A digraph is said to be *simple* if it does not contain self-loops or multi-edges. For any digraph G, we shall denote its set of vertices and edges by $V(G)$ and $E(G)$, respectively. By an edge $(u, v) \in E(G)$, we mean the multi-edge from u to v in G unless otherwise specified, and denote its multiplicity by $\text{mult}(u, v)$. For $V_0 \subseteq V(G)$, we denote the induced subgraph of G on V_0 by $G[V_0]$. Finally, we shall denote the digraph obtained by deleting an edge (u, v) from G by $G \setminus (u, v)$ and the digraph obtained by contracting (defined below) an edge (u, v) in G by $G/(u, v)$.

DEFINITION 7. For integers $n \geq 0, k \geq 2$, we define $\mathcal{G}^{n,k}_{\text{simple}}$ as the set of all digraphs with n vertices, outdegree at most k, with the additional restriction that the digraph does not contain any multi-edge.

DEFINITION 8. For integers $n \geq 0, k \geq 2$, we define $\mathcal{G}^{n,k}_{\text{multi}}$ as the set of all digraphs with n vertices, outdegree exactly k, with the restriction that the multiplicity of edges connecting distinct vertices is at most $k - 1$.

Note that digraphs in $\mathcal{G}^{n,k}_{\text{simple}}$ and $\mathcal{G}^{n,k}_{\text{multi}}$ might contain self-loops. Also note that $\mathcal{G}^{0,k}_{\text{simple}} = \mathcal{G}^{0,k}_{\text{multi}}$ and both of these sets contain a single element, the empty digraph $(\emptyset, \emptyset)$.

DEFINITION 9. Given a digraph G, its *skeleton* $\text{Skel}(G)$ is defined as the digraph obtained by replacing each edge in G with the corresponding edge of multiplicity 1.

We remark that a digraph can be specified by its skeleton and the multiplicities of its edges.

DEFINITION 10. Let G be a digraph, and (u, v) be an edge with distinct end points u and v. The digraph $G/(u, v)$ obtained by contracting the edge (u, v) in G is defined as

$$\begin{aligned} G/(u,v) = ((V(G) \setminus \{u,v\}) \cup \{w\}, \\ (E(G) \setminus \{(x,y) \in E(G) : x, y \in \{u,v\}\}) \\ \cup \{(w,y) : (u,y) \in E(G) \text{ or } (v,y) \in E(G) \text{ with } y \notin \{u,v\}\} \\ \cup \{(x,w) : (x,v) \in E(G) \text{ or } (x,u) \in E(G) \text{ with } x \notin \{u,v\}\} \\ \cup \{(w,w) : (v,u) \in E(G) \text{ or } (u,u) \in E(G) \text{ or } (v,v) \in E(G)\}). \end{aligned}$$

The multi-edge contraction operation (or edge contraction as defined above) on $(u, v) \in E(G)$ replaces u and v with a single vertex w such that all edges incident to u or v, other than the said multi-edge, are now incident to w.

DEFINITION 11. Let G be a simple digraph. An edge $(u, v) \in E(G)$ is said to be *in*-contractible if there does not exist any vertex $x \in V(G)$ distinct from u and v such that $(x, u), (x, v) \in E(G)$. Similarly, (u, v) is said to be *out*-contractible if there does not exist any vertex $x \in V(G)$ distinct from u and v such that $(u, x), (v, x) \in E(G)$. Finally, (u, v) is said to be *contractible* if it is both in-contractible and out-contractible.

We refer the reader to Figure 2 for an example illustrating the above notions of contractibility and the edge contraction operation.

It follows directly from the definition that if an edge (u, v) in a simple digraph G is contractible, it can be contracted without forming multi-edges in the resulting digraph. Note that the digraph obtained may contain self-loops even if the above property is satisfied.

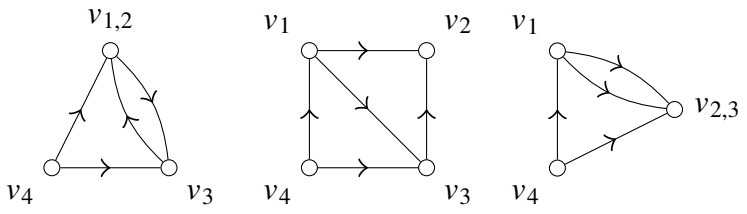

FIGURE 2. An example of a simple digraph G (centre). The edges (v_1, v_2) and (v_4, v_3) are contractible, (v_4, v_1) is not out-contractible, (v_3, v_2) is not in-contractible, and (v_1, v_3) is neither in-contractible nor out-contractible. Contraction of the edge (v_1, v_2) leads to the graph $G/(v_1, v_2)$ (left) with no multi-edges since (v_1, v_2) is contractible, while contraction of the edge (v_3, v_2) leads to the graph $G/(v_3, v_2)$ (right) containing a multi-edge since (v_1, v_2) is not in-contractible.

We use the term *cycle* to refer to a directed cycle unless otherwise specified. For a digraph G, we shall denote the number of cycles in G by $C(G)$. Further, for a vertex $v \in V(G)$, we denote the number of cycles in G passing through v by $C(G, v)$. Similarly, for a multi-edge $e \in E(G)$, we denote the number of cycles in G passing through e by $C(G, e)$. Finally, for $E \subseteq E(G)$, we denote the number of cycles in G passing through at least one edge in E by $C(G, E)$. We remark that our definitions incorporate the multiplicity of edges while computing the number of cycles, i.e., cycles passing through distinct edges that are part of the same multi-edge are considered distinct.

Now, we define $M_k(n) = \max_{G \in \mathcal{G}^{n,k}_{\text{simple}}} C(G)$: that is, $M_k(n)$ denotes the maximum number of cycles in any digraph in $\mathcal{G}^{n,k}_{\text{simple}}$. Similarly, we define $F_k(n) = \max_{G \in \mathcal{G}^{n,k}_{\text{multi}}} C(G)$: that is, $F_k(n)$ denotes the maximum number of cycles in any digraph in $\mathcal{G}^{n,k}_{\text{multi}}$.

DEFINITION 12. Given digraphs G and H, we say that G is H-free if it has no subgraph isomorphic to H.

3.2. Bounds on the number of cycles.

We now prove upper bounds on $M_k(n)$ and $F_k(n)$.

3.2.1. Bounds for simple digraphs.

Dvořák *et al.* [20] proved a more general version of the theorem below for the case of undirected graphs. Their proof idea works for digraphs, as well. We include the full proof for the sake of completeness.

THEOREM 6. *For integers $n \geq 0$, $k \geq 2$, $M_k(n) \leq (k+1)!^{n/(k+1)}$.*

Proof. The result clearly holds for $n = 0, 1$, and 2 since $M_k(0) = 0, M_k(1) = 1$, and $M_k(2) = 3 \leq (k+1)!^{2/(k+1)}$ for all $k \geq 2$. Further, it is easy to check that $M_2(3) = 5 \leq 6 = (2+1)!^{3/(2+1)}$ and $M_k(3) = 8 \leq (k+1)!^{3/(k+1)}$ for all $k \geq 3$. We shall henceforth assume that $n \geq 4$. Let $G \in \mathcal{G}^{n,k}_{\text{simple}}$ and $V(G) = \{v_1, v_2, \ldots, v_n\}$. For $1 \leq i \leq n$, let $\ell_i \in \{0, 1\}$ denote the number of self-loops on v_i in G and $d_i = \text{outdegree}(v_i)$. Further, let G_0 be the simple digraph obtained by removing all self-loops from G. Let A_0 be the adjacency matrix of G_0 and $A' = A_0 + I$. Note that there exists an injection from the set of cycles in G_0 to the symmetric group $\text{Sym}(n)$. This injection maps any given cycle in G_0 to the permutation $\sigma \in \text{Sym}(n)$ whose cycle decomposition contains the given cycle while fixing all other vertices. Therefore, $C(G_0)$ is less than or equal to the permanent of A', which is defined by

$$\text{perm}(A') = \sum_{\sigma \in \text{Sym}(n)} \prod_{i=1}^{n} a'_{i,\sigma(i)}.$$

Note that the sum of the entries in the i-th row of A' is equal to $d_i + 1 - \ell_i$. Hence, using Brègman's theorem (Brègman [12]), we obtain

$$\text{perm}(A') \leq \prod_{i=1}^{n} (d_i + 1 - \ell_i)!^{1/(d_i+1-\ell_i)},$$

which further yields

$$C(G_0) \leq \text{perm}(A') \leq \prod_{i=1}^{n} (k + 1 - \ell_i)!^{1/(k+1-\ell_i)} \tag{4}$$

since $d_i \leq k$ for each $1 \leq i \leq n$ and the function $f : \mathbb{N} \to \mathbb{R}$ defined by $f(m) = m!^{1/m}$ for $m \in \mathbb{N}$ is monotonically increasing. Now, using $C(G) = C(G') + \sum_{i=1}^{n} \ell_i$ in (4), we obtain

$$C(G) \leq \prod_{i=1}^{n} (k + 1 - \ell_i)!^{1/(k+1-\ell_i)} + \sum_{i=1}^{n} \ell_i. \tag{5}$$

We shall now show that $C(G) \leq (k+1)!^{n/(k+1)}$. If $\ell_i = 0$ for all $1 \leq i \leq n$, then (5) yields $C(G) \leq (k+1)!^{n/(k+1)}$. Therefore, we may assume $\ell_1 = 1$ without loss of generality. We show that changing the value of ℓ_1 to 0 while keeping $l_2, l_3, \ldots, l_n$ fixed increases the value of the RHS of (5). For $k \geq 2$, we have $(k+1)!^{1/(k+1)} - k!^{1/k} \geq 1/e$, and

$$\prod_{i=2}^{n} (k + 1 - \ell_i)!^{1/(k+1-\ell_i)} \geq 2^{(n-1)/2} \geq 2^{3/2}.$$

Combining these two inequalities, we obtain

$$k!^{1/k} \prod_{i=2}^{n} (k + 1 - \ell_i)!^{1/(k+1-\ell_i)} + \frac{2^{3/2}}{e} \leq (k+1)!^{1/(k+1)} \prod_{i=2}^{n} (k + 1 - \ell_i)!^{1/(k+1-\ell_i)},$$

which further yields

ATTENTION: The following displayed equation, in its current form, exceeds the column width that will be used in the published edition of your article. Please break or rewrite this equation to fit, including the equation number, within a column width of 470 pt / 165.81 mm / 6.53 in (the width of this red box).

$$k!^{1/k} \prod_{i=2}^{n} (k + 1 - \ell_i)!^{1/(k+1-\ell_i)} + 1 + \sum_{i=2}^{n} \ell_i \leq (k+1)!^{1/(k+1)} \prod_{i=2}^{n} (k + 1 - \ell_i)!^{1/(k+1-\ell_i)} + 0 + \sum_{i=2}^{n} \ell_i.$$

Repeating the same argument for each ℓ_i that is equal to 1, we conclude that changing the value of ℓ_i to 0 for all $1 \leq i \leq n$ increases the value of the RHS of (5). Hence, $C(G) \leq (k+1)!^{n/(k+1)}$.

Since the above bound holds for each $G \in \mathcal{G}^{n,k}_{\text{simple}}$, we obtain the desired result. □

We now provide a particular example of a digraph in $\mathcal{G}^{n,k}_{\text{simple}}$. For $k \geq 2$, $\ell \geq 1$, we define the simple digraph $G^{\ell,k}_{\text{example}}$ which consists of ℓ units of single vertices and k-cliques arranged in an alternating cyclic fashion. Formally, $G^{\ell,k}_{\text{example}} = (V, E)$, where $V = \{v_{i,j} : 0 \leq i < \ell, 0 \leq j \leq k\}$ and $E = \{(v_{i,0}, v_{i,j}) : 0 \leq i < \ell, 1 \leq j \leq k\} \cup \{(v_{i,j_1}, v_{i,j_2}) : 0 \leq i < \ell, 1 \leq j_1, j_2 \leq k\} \cup \{(v_{i_1,j}, v_{i_2,0}) : i_2 - i_1 \equiv 1 \bmod n, 0 \leq i_1 \leq \ell, 0 \leq i_2 \leq \ell, 1 \leq j \leq k\}$. See Figure 3 for an example digraph $G^{3,3}_{\text{example}}$. It can be shown using direct enumeration that

$$C(G^{\ell,k}_{\text{example}}) = \ell\left(2^{k+1} - \binom{k}{2} - 2k - 2\right) + \left(\sum_{r=0}^{k} \frac{k!}{r!} - 1\right)^{\ell}. \tag{6}$$

We remark that the upper bound on $M_k(n)$ in Theorem 6 is asymptotically sharp in the sense that

$$\lim_{k\to\infty} \left(\frac{C(G^{\ell,k}_{\text{example}})}{(k+1)!^{\ell}}\right)^{1/(\ell(k+1))} = 1, \tag{7}$$

which implies that $M_k(n)^{1/n}$ is equal to $(k+1)!^{1/(k+1)}$ upto a multiplicative factor that gets arbitrarily close to 1 as k goes to infinity.

3.2.2. Bounds for directed multigraphs. We begin by providing two particular examples of digraphs in $\mathcal{G}^{n,k}_{\text{multi}}$, with the same skeleton. For $n \geq 3$, we define the simple digraph $G_n = (V, E)$ with $V = \{v_i : 0 \leq i < n\}$ and $E = \{(v_i, v_j) : j - i \equiv 1 \bmod n \text{ or } j - i \equiv 2 \bmod n\}$. Let $G_{n,k}$ be the digraph with skeleton $\text{Skel}(G_{n,k}) = G_n$ in which edges (v_i, v_j) with $j - i \equiv 1 \bmod n$ have multiplicity $k-1$ while the remaining edges have multiplicity 1. Similarly, let $G'_{n,k}$ be the digraph with skeleton

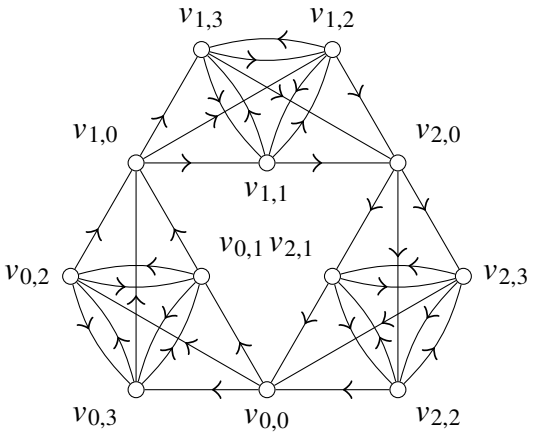

FIGURE 3. The digraph $G^{3,3}_{\text{example}}$.

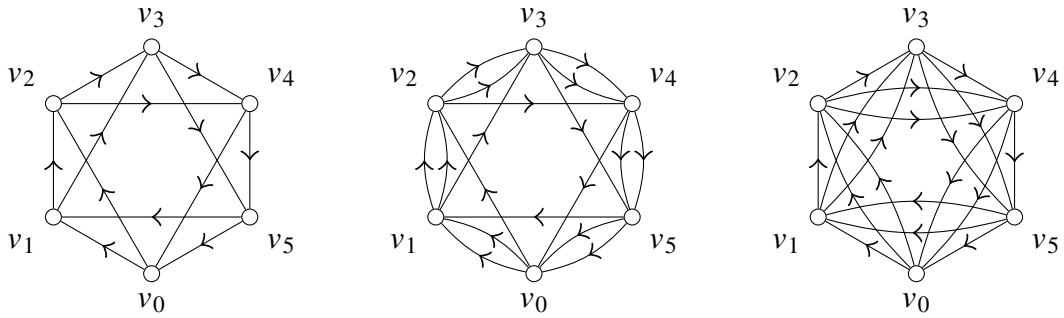

FIGURE 4. The digraphs G_6, $G_{6,3}$ and $G'_{6,3}$ (from left to right).

$\text{Skel}(G'_{n,k}) = G_n$ in which edges (v_i, v_j) with $j - i \equiv 1 \bmod n$ having multiplicity equal to 1 while the remaining edges have multiplicity $k-1$. See Figure 4 for example digraphs G_6, $G_{6,3}$, and $G'_{6,3}$. From the above definitions, it is easy to see that $G_{n,2}$, $G'_{n,2}$, and G_n are isomorphic, and $G_{n,k}, G'_{n,k} \in \mathcal{G}^{n,k}_{\text{multi}}$. Further, it can be shown using direct enumeration that

$$C(G_{n,k}) = \begin{cases} S_{n-2} + S_n + 1, & \text{if } n \text{ is odd,} \\ S_{n-2} + S_n, & \text{otherwise,} \end{cases} \tag{8}$$

where S_n is defined by the recurrence relation $S_n = (k-1)S_{n-1} + S_{n-2}$ with boundary condition $S_0 = 1, S_1 = k-1$. For any vertex $v \in V(G_{n,k})$, $C(G_{n,k})$ can be written as a sum of the number of cycles passing through v and those not passing through v. In this case, $C(G_{n,k}) - C(G_{n,k}, v) = S_{n-2}$ and the 1 extra cycle contributing to $C(G_{n,k}, v)$ for odd n corresponds to the Hamiltonian cycle comprised of all edges of multiplicity 1 in $G_{n,k}$. Similarly, one can also show that

$$C(G'_{n,k}) = \begin{cases} (k-1)T_{n-2} + T_n + (k-1)^n, & \text{if } n \text{ is odd,} \\ (k-1)T_{n-2} + T_n, & \text{otherwise,} \end{cases} \tag{9}$$

where T_n is defined by the recurrence relation $T_n = T_{n-1} + (k-1)T_{n-2}$ with boundary condition $T_0 = 1, T_1 = 1$.

In Lemma 1, we show that the number of cycles in $G_{n,k}$ is greater than or equal to the number of cycles in $G'_{n,k}$.

LEMMA 1. *For natural numbers $n \geq 3$, $k \geq 2$, $C(G_{n,k}) \geq C(G'_{n,k})$.*

Proof. We will show using induction that $(k-1)T_{n-2} + T_n + (k-1)^n \leq S_{n-2} + S_n + 1$ for all $n \geq 3$, thereby implying $C(G_{n,k}) \geq C(G'_{n,k})$ for each $n \geq 3$. For simplicity, let L_n and R_n denote the LHS and RHS of the above inequality, respectively. It is easy to check that $L_3 = R_3$ and

$L_4 + 2(k-2)^2 = R_4$. Now, by induction hypothesis, we have $L_{n-1} \leq R_{n-1}$ and $L_n \leq R_n$, which implies $(k-1)L_n + L_{n-1} \leq (k-1)R_n + R_{n-1}$, which when expanded and rearranged yields

$$(k-1)((k-1)T_{n-2} + T_{n-3}) + ((k-1)T_n + T_{n-1}) + (k-1)^{n+1} + (k-1)^{n-1} \leq S_{n-1} + S_{n+1} + k. \quad (10)$$

Since $\{T_n\}$ is a monotonically increasing sequence, $(k-1)T_m + T_{m-1} \geq T_m + (k-1)T_{m-1} = T_{m+1}$ for all $m \in \mathbb{N}$. Using this inequality and $(k-1)^{n-1} \geq k-1$ in (10), we obtain

$$(k-1)T_{n-1} + T_{n+1} + (k-1)^{n+1} \leq S_{n-1} + S_{n+1} + 1.$$

□

We now compute $C(G_{n,k})$. Solving the recurrence relation for S_n with appropriate boundary conditions, we obtain

$$S_n = \frac{\left(\frac{k-1+\sqrt{(k-1)^2+4}}{2}\right)^{n+1} - \left(\frac{k-1-\sqrt{(k-1)^2+4}}{2}\right)^{n+1}}{\sqrt{(k-1)^2+4}},$$

which when used in (8) yields $C(G_{n,k}) = \lceil \alpha(k)^n \rceil$, where $\lceil . \rceil$ is the ceiling function and

$$\alpha(k) = \frac{k-1+\sqrt{(k-1)^2+4}}{2}. \quad (11)$$

REMARK 1. The digraph G_n is an example of a Cayley graph (Meier [43, see Section 1.5]) since $G_n = (G, \{(v, v+s) : v \in G, s \in S\})$, where $G = \mathbb{Z}/N\mathbb{Z}$ and $S = \{1, 2\}$ is a generating set for G. Cayley graphs are known to be extremal examples for various problems in graph theory.

For $i \in \{1, 2\}$, let H_i be the digraph (V, E_i), where $V = \{v_1, v_2, v_3\}$, $E_1 = \{(v_2, v_1), (v_3, v_1), (v_2, v_3),$
$(v_3, v_2)\}$, and $E_2 = \{(v_1, v_2), (v_1, v_3), (v_2, v_3), (v_3, v_2)\}$. See Figure 5 for drawings of H_1 and H_2.

In Lemma 2, we prove a bound on the number of edges that are not in-contractible in a simple two-regular H_1-free and H_2-free digraph. Furthermore, we characterise the graphs for which this

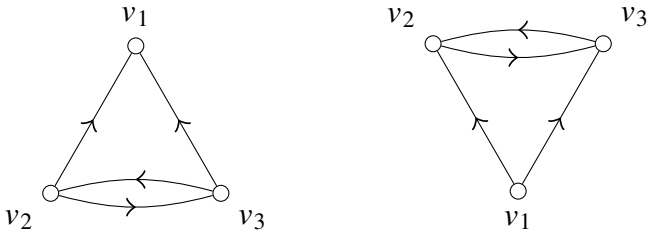

FIGURE 5. The digraphs H_1 and H_2 (from left to right).

bound is achieved. In Lemma 3, we provide an upper bound on the number of cycles for these characterised graphs. These results are used in the proof of Theorem 7 in the following way: if the graph has few edges that are not in-contractible, then we contract edges to obtain a recursive bound on the number of cycles; if the graph has many edges that are not in-contractible, then we use Lemma 3 to bound the number of cycles.

LEMMA 2. *Let G be a simple two-regular H_1-free and H_2-free digraph on $n \geq 4$ vertices. Then, G can have at most n edges that are not in-contractible. Further, G has exactly n edges that are not in-contractible if and only if each connected component of G is isomorphic to G_m for some $m \geq 4$.*

Proof. For any $v \in V(G)$, we have distinct vertices $x, y \in V(G)$ such that $(x, v), (y, v) \in E(G)$ since G is simple. If the edge (x, v) is not in-contractible, then $(y, x) \in E(G)$. Similarly, if the edge (y, v) is not in-contractible, then $(x, y) \in E(G)$. Suppose both (x, v) and (y, v) are not in-contractible. Then, the subgraph $(\{x, y, v\}, \{(x, v), (y, v), (x, y), (y, x)\})$ of G is isomorphic to H_1, a contradiction. The sets consisting of incoming edges to a vertex in G constitute a uniform n-partition of $E(G)$. Since at most one edge in each such set is not in-contractible, the digraph G can have at most n edges that are not in-contractible.

Now, suppose that G has exactly n edges that are not in-contractible. Then, for any vertex v in G, exactly one of the two incoming edges to v must be in-contractible. Let us pick a vertex $v_1 \in G$ with $(v_2, v_1), (v_3, v_1) \in E(G)$. Without loss of generality, we may assume that the edge (v_2, v_1) is not in-contractible: that is, $(v_3, v_2) \in E(G)$. The other incoming edge to v_2 cannot originate from (i) v_1 since otherwise G would contain a subgraph isomorphic to H_2, (ii) v_2 since otherwise G would contain a self-loop, and (iii) v_3 since otherwise G would contain a multi-edge. Therefore, it originates from a vertex $v_4 \notin \{v_1, v_2, v_3\}$. Now, among the incoming edges (v_3, v_2) and (v_4, v_2) to v_2, the edge (v_4, v_2) must be in-contractible since otherwise $(v_3, v_4) \in E(G)$, which is a contradiction to the fact that outdegree$(v_3) = 2$. Therefore, the edge (v_3, v_2) is not in-contractible and hence $(v_4, v_3) \in E(G)$. The other incoming edge to v_3 cannot originate from (i) v_2 since otherwise G would contain a subgraph isomorphic to H_1, (ii) v_3 since otherwise G would contain a self-loop, and (iii) v_4 since otherwise G would contain a multi-edge. Therefore, the other incoming edge to v_3 could either originate from v_1 or a vertex $v_5 \notin \{v_1, v_2, v_3, v_4\}$.

In the case where the other incoming edge to v_3 originates from v_1, the edge (v_1, v_3) must be in-contractible since otherwise $(v_4, v_1) \in E(G)$, which is a contradiction to the fact that outdegree$(v_4) = 2$. Therefore, the edge (v_4, v_3) is not in-contractible and hence $(v_1, v_4) \in E(G)$.

Finally, $(v_2, v_4) \in E(G)$ since one incoming edge to v_4 must not be in-contractible, yielding a connected component G_4 of G.

In the case where the other incoming edge to v_3 originates from a vertex $v_5 \notin \{v_1, v_2, v_3, v_4\}$, we have $(v_5, v_4) \in E(G)$. Now, the other incoming edge to v_4 could either originate from v_1, v_2 or a vertex $v_6 \notin \{v_1, v_2, v_3, v_4, v_5\}$ since the outdegree of v_3, v_4, and v_5 is already satisfied. It cannot originate from v_2 since both incoming edges to v_4 would otherwise be in-contractible. If it originates from v_1, we get a connected component G_5 of G. If it originates from a vertex $v_6 \notin \{v_1, v_2, v_3, v_4, v_5\}$, we continue in a similar way until we get a connected component G_m of G for some $m > 5$.

Conversely, suppose each connected component of G is isomorphic to G_m for some $m \geq 4$. Then, within a connected component G_m, it is easy to check that the m edges (v_i, v_j) with $j - i \equiv 1 \bmod m$ are not in-contractible while the m edges with $j - i \equiv 2 \bmod m$ are in-contractible. Summing over the connected components of G, we get exactly n edges that are not in-contractible in G. □

REMARK 2. Under the same hypothesis as Lemma 2, one can prove the following analogous symmetric result (although we require only one of these results): G can have at most n edges that are not out-contractible. Further, G has exactly n edges that are not out-contractible if and only if each connected component of G is isomorphic to G_m for some $m \geq 4$.

LEMMA 3. *Let G be a digraph with $n \geq 4$ vertices, each of whose connected components is $G_{m,k}$ or $G'_{m,k}$ for some $m \geq 4$. Then, $C(G) \leq (k-1)F_k(n-1) + F_k(n-2)$.*

Proof. For any natural number $m \geq 3$, we have $C(G_{m,k}) \leq F_k(m)$ since $G_{m,k} \in \mathcal{G}^{m,k}_{\text{multi}}$. Using (8) in this inequality, we obtain $S_{m-2} + S_m + 1 \leq (k-1)F_k(m-1) + F_k(m-2)$. For natural numbers $p, q \geq 3$, we have

$$\begin{aligned} C(G_{p,k}) + C(G_{q,k}) &\leq (S_{p-2} + S_p + 1) + (S_{q-2} + S_q + 1) \\ &= (S_{p-2} + (S_{q-2} + 1)) + (S_p + (S_q + 1)) \\ &\leq (S_{p+q-3} + S_{p+q-4}) + (S_{p+q-1} + S_{p+q-2}) \\ &\leq S_{p+q-2} + S_{p+q} \\ &\leq C(G_{p+q,k}). \end{aligned}$$

Now, we have $C(G) = \sum_{i=1}^{l} C(G^i)$, where G^i is the i-th connected component of G and l is the number of connected components in G. Let $m_i = |V(G^i)|$, so that $\sum_{i=1}^{l} m_i = n$. Using the subadditivity of the function $C(G_{.,k})$ and the fact that $C(G'_{m,k}) \leq C(G_{m,k})$ for all $m \geq 3$, we get

$$\begin{aligned} C(G) &= \sum_{i=1}^{l} C(G^i) \\ &\leq \sum_{i=1}^{l} C(G_{m_i,k}) \\ &\leq C(G_{\sum_{i=1}^{l} m_i,k}) \\ &= C(G_{n,k}) \\ &\leq S_{n-2} + S_n + 1 \\ &\leq (k-1)F_k(n-1) + F_k(n-2). \end{aligned}$$

□

We now prove the main result of this section, an asymptotically tight formula for $F_k(n)$. The proof proceeds by reducing the digraph into smaller digraphs belonging to the same family to obtain several non-homogeneous linear recursive bounds of order up to 3 based on a case analysis, and finally combining all these bounds to obtain the final result.

THEOREM 7. *For $k \geq 2$,*

$$F_k(n) = \Theta(\alpha(k)^n)$$

as $n \to \infty$, where $\alpha(k)$ is as defined in (11)*. In particular, $F_2(n) = \Theta(\mathrm{Fib}(n))$, where* $\mathrm{Fib}(m)$ *denotes the m-th Fibonacci number.*

Proof. We begin by making a few elementary observations about the function F_k. Clearly, $F_k(0) = 0, F_k(1) = k$, and $F_k(2) = \max(\{2k, (k-1)^2 + 2\})$, corresponding to the empty digraph, the digraph with a single vertex having a self-loop of multiplicity k, and the digraph with 2 vertices, each having a self-loop of multiplicity k or each having a self-loop of multiplicity 1 along with an edge of multiplicity $k-1$ to the other vertex, respectively. For $n \in \mathbb{N}$, let $G^* \in \mathcal{G}_{\text{multi}}^{n-1,k}$ be such that $C(G^*) = F_k(n-1)$ (such a digraph exists since $\mathcal{G}_{\text{multi}}^{n-1,k}$ is finite) and let G be the digraph obtained by adding a single vertex with a self-loop of multiplicity k to G^*. Then, $F_k(n) \geq C(G) = C(G^*) + k =$

$F_k(n-1)+k$. In particular, $F_k(n) \geq k$ for all $n \in \mathbb{N}$. Since we have already computed the values of $F_k(n)$ for $n \in \{0,1,2\}$, we shall henceforth assume that $n \geq 3$.

Let $G \in \mathcal{G}^{n,k}_{\text{multi}}$. In the next three paragraphs of this proof, we argue that one may assume without loss of generality certain restrictions on the structure of G without reducing the number of cycles or $C(G) \leq (k-1)F_k(n-1)+F_k(n-2)$.

Suppose that G contains a vertex v with a self-loop of multiplicity k. Then $C(G) = C(G[V(G) \setminus \{v\}]) + k \leq F_k(n-1)+k \leq (k-1)F_k(n-1)+F_k(n-2)$. Note that the digraph $G[V(G) \setminus \{v\}]$ might not necessarily be in $\mathcal{G}^{n-1,k}_{\text{multi}}$. However, one can add a self-loop of sufficient multiplicity to each vertex of $G[V(G) \setminus \{v\}]$ to obtain a digraph $G' \in \mathcal{G}^{n-1,k}_{\text{multi}}$ so that $C(G) \leq C(G') \leq F_k(n-1)$.

Henceforth, we shall assume that each vertex in G has at least one outgoing edge, which is not a self-loop. Note that for any vertex $v \in V(G)$, $C(G)$ is equal to the sum of the number of cycles passing through the vertex v and those not passing through v. Let $e_1, e_2, \ldots, e_k$ be the outgoing simple edges from v (some of these edges might have the same end points since G is a multigraph). Then, the number of cycles passing through v is equal to the sum of the number of cycles passing through each of these edges. Now, there exists a permutation $\sigma \in \text{Sym}(k)$ such that $C(G, e_{\sigma(1)}) \geq C(G, e_{\sigma(2)}) \geq \cdots \geq C(G, e_{\sigma(k)})$ and the end points of $e_{\sigma(1)}$ are not the same as the end points of $e_{\sigma(k)}$ (it is possible to satisfy the latter condition since G does not contain edges of multiplicity k connecting distinct vertices). In such a case, we can construct a digraph G^* from G by deleting $e_{\sigma(2)}, \ldots, e_{\sigma(k-1)}$ and adding $(k-2)$ copies of $e_{\sigma(1)}$. The resulting digraph G^* has the property $C(G^*) \geq C(G)$ and that the vertex v in G^* has two outgoing edges, one with multiplicity 1 and the other with $k-1$. Applying this operation successively to every vertex in the digraph, we obtain a digraph G' with $C(G) \leq C(G')$, which also has the property that every vertex in G' has two outgoing edges, one with multiplicity 1 and the other with $k-1$. Further, for any vertex $v \in V(G')$, the number of cycles in G' passing through the outgoing edge of multiplicity $k-1$ from v is greater than $(k-1)$ times the number of cycles passing through the outgoing edge of multiplicity 1 from v. Therefore, we may assume without loss of generality (renaming G' to G) that each vertex in $\text{Skel}(G)$ has outdegree 2 (one of these outgoing edges has multiplicity $k-1$ and the other has multiplicity 1 in G).

If $\text{Skel}(G)$ contains a vertex v with a self-loop but no other incoming edge, we have $C(G) \leq C(G[V(G) \setminus \{v\}]) + k - 1 \leq F_k(n-1)+k-1 \leq (k-1)F_k(n-1)+F_k(n-2)$. If $\text{Skel}(G)$ contains an indegree 2 vertex v with a self-loop and the other incoming edge (u,v) participating in a 2-cycle, we have $C(G) \leq C(G[V(G) \setminus \{v\}]) + (k-1)^2 + 1 \leq F_k(n-1) + (k-1)k \leq F_k(n-1) +$

$(k-1)F_k(n-2) \leq (k-1)F_k(n-1) + F_k(n-2)$. Finally, we consider the case where each vertex with a self-loop in $\text{Skel}(G)$ has an incoming edge other than the self-loop not participating in a 2-cycle. For any such vertex v and an incoming edge (u,v) not participating in a 2-cycle, the digraph G' formed by deleting the self-loop on v from G and adding the edge (v,u) with the same multiplicity as the self-loop, has at least as many cycles as the original digraph G. We repeatedly apply this operation to the digraph G until no self-loops remain to obtain a digraph G' satisfying $C(G) \leq C(G')$. Therefore, we may assume without loss of generality (renaming G' to G) that G contains no self-loops.

We now consider the following mutually exclusive exhaustive set of cases:

Case 1. There is a vertex $v \in \text{Skel}(G)$ with $\text{indegree}(v) = 0$. In this case, we have $C(G) = C(G[V(G) \setminus \{v\}]) \leq F_k(n-1) \leq (k-1)F_k(n-1) + F_k(n-2)$ since none of the cycles in G pass through v.

Case 2. There is a vertex $v \in \text{Skel}(G)$ with $\text{indegree}(v) = 1$. Let (u,v) be the unique incoming edge incident to v in G. Let (u,w) be the other outgoing edge from u in G. The number of cycles passing through (u,w) in G is bounded above by the number of cycles in the digraph obtained by deleting the vertex v and the incoming edges to w other than the edge (u,w), and contracting the edge (u,w), which is less than or equal to $\text{mult}(u,w)F_k(n-2)$. Similarly, the number of cycles not passing through (u,w) in G is equal to the number of cycles in the digraph obtained by deleting the edge (u,w) from G and contracting the edge (u,v), which is bounded above by $\text{mult}(u,v)F_k(n-1)$. Therefore, we obtain $F_k(n) \leq \max(\{(k-1)F_k(n-1) + F_k(n-2), (k-1)F_k(n-2) + F_k(n-1)\}) = (k-1)F_k(n-1) + F_k(n-2)$.

Case 3. All vertices in $\text{Skel}(G)$ have indegree equal to 2: that is, $\text{Skel}(G)$ is 2-regular.

Case 3.1. Suppose that $k \geq 3$. We consider the case when G contains a vertex v, both of whose incoming edges (a,v) and (b,v) have multiplicity 1. We shall assume without loss of generality that $C(G,(b,v)) \geq C(G,(a,v))$. Let (b,w) be the other outgoing edge from b in G. Then, $C(G,(b,w)) \geq (k-1)C(G,(b,v))$. Further, we have

$$C(G,v) = C(G,(a,v)) + C(G,(b,v)) \leq 2C(G,(b,v)),$$

and

$$C(G) \geq C(G,b) = C(G,(b,v)) + C(G,(b,w)) \geq kC(G,(b,v)).$$

Combining the above inequalities, we obtain $C(G,v) \leq 2C(G)/k$, which further implies $C(G) \leq kC(G[V(G) \setminus \{v\}])/(k-2) \leq kF_k(n-1)/(k-2)$. For $k \geq 4$, we have $k/(k-2) \leq k-1$, which implies $C(G) \leq (k-1)F_k(n-1)$. Now, we shall focus on the case $k = 3$. Let (v,y) and (v,z) be the outgoing edges from v in G with multiplicities 1 and 2, respectively. For $v_1 \in \{a,b\}$ and $v_2 \in \{y,z\}$, we define x_{v_1,v_2} to be the fraction of cycles in G passing through the path $v_1 \to v \to v_2$; that is,

$$x_{v_1,v_2} = C(G,\{(v_1,v),(v,v_2)\})/C(G).$$

Let $\boldsymbol{x} = [x_{a,y}\ x_{a,z}\ x_{b,y}\ x_{b,z}]^T$. We consider the following mutually exclusive exhaustive set of cases.

Case 3.1.1. We first consider the case when $(a,z),(b,z) \notin E(G)$. Let G' be the digraph obtained by adding the edges (a,z) and (b,z) with multiplicity 1 to the digraph $G[V(G) \setminus \{v\}]$. Then, we have

$$\begin{aligned}
C(G') &= C(G[V(G) \setminus \{v\}) + C(G',(a,z)) + C(G',(b,z)) \\
&= C(G) - C(G,v) + C(G,\{(a,v),(v,z)\})/2 + C(G,\{(b,v),(v,z)\})/2 \\
&\geq C(G) - C(G,v) + C(G,v)/3 \\
&\geq C(G) - 4C(G)/9 \\
&= 5C(G)/9.
\end{aligned}$$

Now, since $C(G') \leq F_3(n-1)$, we obtain $C(G) \leq 9F_3(n-1)/5$.

Case 3.1.2. We now consider the case when $(a,z) \in E(G)$ with multiplicity 2. Let G' be the digraph obtained by adding the edges (b,z) and (a,y) with multiplicity 1 to the digraph $G[V(G) \setminus \{v\}]$. Then, we have

ATTENTION: The following displayed equation, in its current form, exceeds the column width that will be used in the published edition of your article. Please break or rewrite this equation to fit, including the equation number, within a column width of 470 pt / 165.81 mm / 6.53 in (the width of this red box).

$$\begin{aligned} C(G') &= C(G[V(G)\setminus\{v\}) + C(G', \{(b,z),(a,y)\}) + C(G'\setminus(a,y),(b,z)) + C(G'\setminus(b,z),(a,y)) \\ &\geq C(G) - C(G,v) + x_{b,z}C(G)/2 + x_{a,y}C(G) \\ &= C(G)(1 - x_{a,z} - x_{b,y} - x_{b,z}/2). \end{aligned} \tag{12}$$

Further, we have

$$\begin{aligned} (x_{a,y} + x_{a,z})C(G) = C(G,(a,v)) &\leq C(G,(b,v)) = (x_{b,y} + x_{b,z})C(G), \\ 2(x_{a,y} + x_{b,y})C(G) = 2C(G,(v,y)) &\leq C(G,(v,z)) = (x_{a,z} + x_{b,z})C(G), \\ 3(x_{b,y} + x_{b,z}) = 3C(G,(b,v)) &\leq C(G), \\ (x_{a,z} + x_{b,z})C(G) + 2(x_{a,y} + x_{a,z})C(G) &\leq C(G,(v,z)) + C(G,(a,z)) = C(G,z) \leq C(G). \end{aligned}$$

Minimizing the function $\varphi(\boldsymbol{x}) = -x_{a,z} - x_{b,y} - x_{b,z}/2$ subject to the above constraints along with the constraint $\boldsymbol{x} \in [0,1]^4$, we obtain $\varphi(\boldsymbol{x}) \geq -9/16$, which when used in (12) yields $C(G') \geq 7C(G)/16$. Therefore, $C(G) \leq 16F_3(n-1)/7$.

Case 3.1.3. We now consider the case when $(b,z) \in E(G)$ with multiplicity 2. Let G' be the digraph obtained by adding the edges (a,z) and (b,y) with multiplicity 1 to the digraph $G[V(G)\setminus\{v\}]$. Then, we have

ATTENTION: The following displayed equation, in its current form, exceeds the column width that will be used in the published edition of your article. Please break or rewrite this equation to fit, including the equation number, within a column width of 470 pt / 165.81 mm / 6.53 in (the width of this red box).

$$\begin{aligned} C(G') &= C(G[V(G)\setminus\{v\}) + C(G', \{(a,z),(b,y)\}) + C(G'\setminus(b,y),(a,z)) + C(G'\setminus(a,z),(b,y)) \\ &\geq C(G) - C(G,v) + x_{a,z}C(G)/2 + x_{b,y}C(G) \\ &= C(G)(1 - x_{a,y} - x_{a,z}/2 - x_{b,z}). \end{aligned} \tag{13}$$

Similar to the previous case, we have

$$\begin{aligned}
(x_{a,y}+x_{a,z})C(G) = C(G,(a,v)) &\le C(G,(b,v)) = (x_{b,y}+x_{b,z})C(G),\\
2(x_{a,y}+x_{b,y})C(G) = 2C(G,(v,y)) &\le C(G,(v,z)) = (x_{a,z}+x_{b,z})C(G),\\
3(x_{b,y}+x_{b,z}) = 3C(G,(b,v)) &\le C(G),\\
(x_{a,z}+x_{b,z})C(G) + 2(x_{b,y}+x_{b,z})C(G) &\le C(G,(v,z)) + C(G,(b,z)) = C(G,z) \le C(G).
\end{aligned}$$

Minimizing the function $\varphi(\boldsymbol{x}) = -x_{a,y} - x_{a,z}/2 - x_{b,z}$ subject to the above constraints along with the constraint $\boldsymbol{x} \in [0,1]^4$, we obtain $\varphi(\boldsymbol{x}) \ge -11/20$, which when used in (13) yields $C(G') \ge 9C(G)/20$. Therefore, $C(G) \le 20F_3(n-1)/9$.

We shall henceforth assume that each vertex of G has one incoming edge with multiplicity $k-1$ and the other with multiplicity 1.

Case 3.2. We consider the case in which $\mathrm{Skel}(G)$ contains a subgraph isomorphic to H_1 and/or H_2. As shown in Figure 6, we consider the following mutually disjoint exhaustive set of cases:

Case 3.2.1. G contains distinct vertices v, a, and b such that $(v,a),(v,b),(a,b),(b,a),(a,v),(b,v) \in E(G)$. In this case, we have $C(G) = C(G[V(G) - \{a,b,v\}]) + (k-1)^3 + 3(k-1) + 1 \le F_k(n-3) + (k-1)^3 + 3(k-1) + 1$.

Case 3.2.2. G contains distinct vertices u, v, a, and b such that $(u,a),(u,b),(a,b),(b,a),(a,v),(b,v) \in E(G)$. In this case, we have $C(G) \le ((k-1)^3 + 2(k-1)+1)C(G') + (k-1) \le ((k-1)^3 + 2(k-1)+1)F_k(n-3) + (k-1)$, where G' is the digraph obtained by merging the vertices u, v, a and b in G.

Case 3.2.3. G contains distinct vertices u, a, and b such that $(u,a),(u,b),(a,b),(b,a) \in E(G)$ but there does not exist any vertex v such that $(a,v),(b,v) \in E(G)$. In this case, we get $C(G) \le ((k-1)^2+1)C(G') + (k-1) \le ((k-1)^2+1)F_k(n-2) + (k-1)$, where G' is the digraph

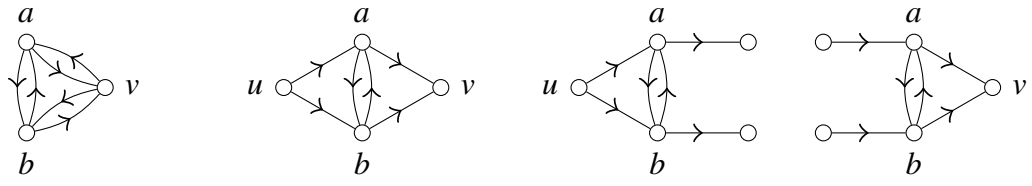

FIGURE 6. The cases 3.2.1, 3.2.2, 3.2.3, and 3.2.4 (from left to right).

obtained by merging the vertices a, b and u in G.

Case 3.2.4. G contains distinct vertices v, a, and b such that $(a,v),(b,v),(a,b),(b,a) \in E(G)$ but there does not exist any vertex u such that $(u,a),(u,b) \in E(G)$. In this case, we get $C(G) \leq ((k-1)^2+1)C(G') + (k-1) \leq ((k-1)^2+1)F_k(n-2) + (k-1)$, where G' is the digraph obtained by merging the vertices a, b and v in G.

Case 3.3. We now consider the case when $\mathrm{Skel}(G)$ is H_1-free and H_2-free. For any vertex $v \in V(\mathrm{Skel}(G))$, let us define $E_v = \{(x,y) : (x,v) \in E(\mathrm{Skel}(G)), (x,y) \in E(\mathrm{Skel}(G)), v \neq y\}$. Note that $|E_v| = 2$ and $E_v \cap E_u = \emptyset$ for any two distinct vertices $u, v \in V(\mathrm{Skel}(G))$. Therefore, the sets E_v form a uniform n-partition of $E(\mathrm{Skel}(G))$.

Case 3.3.1. If $\mathrm{Skel}(G)$ contains exactly n edges that are not in-contractible, then by Lemma 2, each connected component of $\mathrm{Skel}(G)$ must be isomorphic to G_m for some $m \geq 4$. Now, it is easy to check that $G_{m,k}$ and $G'_{m,k}$ are the only digraphs whose skeleton is isomorphic to G_m with each vertex in the digraph incident to two incoming (outgoing) edges, one with multiplicity 1 and the other with multiplicity $k-1$. Therefore, each connected component of G is isomorphic to one of $G_{m,k}$ and $G'_{m,k}$ for some $m \geq 4$. Finally, from Lemma 3, we have $C(G) \leq (k-1)F_k(n-1) + F_k(n-2)$.

Case 3.3.2. If $\mathrm{Skel}(G)$ contains strictly less than n edges that are not in-contractible, then we can find a vertex $x \in V(\mathrm{Skel}(G))$ such that both edges in E_x are in-contractible. Let (v,x) and (w,x) be the incoming edges to x in G. Let us assume that (v,y) and (w,z) are the other outgoing edges from v and w, respectively. Without loss of generality, we may assume that (v,x) has multiplicity 1 in G. The number of cycles not passing through (v,x) in G is less than or equal to $(k-1)C(G')$, where G' is the digraph obtained from G by deleting the edge (v,x) followed by contracting the edge (v,y). The number of cycles passing through (v,x) in G is equal to the number of cycles passing through v in the digraph G'' formed by deleting the edges (v,y) and (w,x) from G, which is further less than or equal to $C(G''')$, where G''' is the digraph obtained by contracting the edges (w,z) and (v,x) in G'. Thus, we obtain $C(G) \leq (k-1)C(G') + C(G''') \leq (k-1)F_k(n-1) + F_k(n-2)$.

Combining all the recurrent upper bounds obtained through the proof, we obtain

ATTENTION: The following displayed equation, in its current form, exceeds the column width that will be used in the published edition of your article. Please break or rewrite this equation to fit, including the equation number, within a column width of 470 pt / 165.81 mm / 6.53 in (the width of this red box).

$$F_k(n) \leq \max(\{(k-1)F_k(n-1)+F_k(n-2), ((k-1)^2+1)F_k(n-2)+(k-1), \\ ((k-1)^3+2(k-1)+1)F_k(n-3)+(k-1), F_k(n-3)+(k-1)^3+3(k-1)+1\})$$

for $k \geq 2, k \neq 3$. Now, we will show that $F_k(n) \leq 5\alpha(k)^n$, where $\alpha(k)$ is as defined in (11). Using the expressions for $F_k(0), F_k(1)$ and $F_k(2)$ from the beginning of this proof, it is easy to check that $F_k(n) \leq 5\alpha(k)^n$ for $n = 0, 1, 2$. Now, assuming that the result holds for all natural numbers up to $n-1$, we have

$$(k-1)F_k(n-1)+F_k(n-2) \leq 5(k-1)\alpha(k)^{n-1}+5\alpha(k)^{n-2} = 5\alpha(k)^n.$$

Further, it is easy to check that if the inequality

$$((k-1)^2+1)5\alpha(k)^{n-2}+(k-1) \leq 5\alpha(k)^n$$

holds for $n = 3$, then it holds for all $n \geq 3$ since $\alpha(k) \geq 1$. And it is also easy to verify that the result indeed holds for $n = 3$. Therefore, we have

$$((k-1)^2+1)F_k(n-2)+(k-1) \leq ((k-1)^2+1)5\alpha(k)^{n-2}+(k-1) \leq 5\alpha(k)^n$$

for all $n \geq 3$. Similar results can be shown for $F_k(n) \leq ((k-1)^3+2(k-1)+1)F_k(n-3)+(k-1)$ and $F_k(n) \leq F_k(n-3)+(k-1)^3+3(k-1)+1$. Hence, we obtain the result $F_k(n) \leq 5\alpha(k)^n$ for all non-negative integers n and k with $k \geq 2, k \neq 3$. Recall that for $k = 3$, we have the extra inequality $F_3(n) \leq 16F_3(n-1)/7$. But since $16\alpha(3)^{n-1}/7 \leq \alpha(3)^n$, the result holds for $k = 3$ as well. Note that the constant 5 in this result can be improved; however, we do not concern ourselves with finding the best possible constant. Finally, since $F_k(n) \geq C(G_{n,k}) \geq \alpha(k)^n$, we obtain $F_k(n) = \Theta(\alpha(k)^n)$. The constant $\alpha(2) = (1+\sqrt{5})/2$ is the golden ratio and hence $F_2(n) = \Theta(\text{Fib}(n))$, where $\text{Fib}(n)$ is the n-th Fibonacci number. □

REMARK 3. It is possible to show that $F_k(n) \leq (k-2+6^{1/3})^n$ using the star bound (Soules [54]), a generalization of Brègman's theorem. However, this bound is exponentially weaker than the one we have proved in Theorem 7, which is tight up to a known constant.

4. Bounds for Policy Iteration on Deterministic MDPs. We begin with the formal definition of a DMDP and then discuss the structure of the policy space for DMDPs.

DEFINITION 13. A DMDP is an MDP (S, A, T, R, γ) with a deterministic transition function: that is, the codomain of T is $\{0, 1\}$.

A DMDP M can be viewed as a directed multigraph G_M with S as the set of vertices, which contains an edge corresponding to each state-action pair. Further, a policy $\pi \in \Pi_M$ can be viewed as a digraph G_π in which each vertex has outdegree one. Such digraphs are known as functional graphs since they correspond to functions defined on the set of vertices. A functional graph is a union of its connected components, each containing a single cycle and paths leading into the cycle. See Figure 7 for an example. Post and Ye's [48] analysis suggests that the value function of any policy is primarily dictated by the cycles present in the corresponding digraph. We now describe an alternative, slightly different way to view policies for DMDPs.

DEFINITION 14. A digraph isomorphic to $G = (\{v_1, \ldots, v_n\}, \{(v_i, v_{i+1}) : 1 \le i < n\} \cup \{(v_n, v_m)\})$ for some $m, n \in \mathbb{N}$ with $m \le n$ is called a path-cycle.

Let M be a DMDP with its set of states $S = \{s_1, \ldots, s_n\}$. A policy $\pi \in \Pi_M$ can be viewed as an n-tuple $(P^\pi_{s_1}, \ldots, P^\pi_{s_n})$ of path-cycles, where P^π_s is the path-cycle obtained by following π starting from state s. We shall call $(P^\pi_{s_1}, \ldots, P^\pi_{s_n})$ the (path-cycle) representation of policy π. Note that $V^\pi(s)$ is completely determined by the corresponding path-cycle P^π_s for any $s \in S$.

LEMMA 4. *Let M be a DMDP and $\pi_1 \prec \cdots \prec \pi_\ell$ be an increasing sequence of policies in Π_M. Then, for each $1 < i \le \ell$, the representation of π_i contains a path-cycle which is not part of representations of π_j for any $1 \le j < i$. Therefore, we can associate a possibly non-unique sequence of distinct path-cycles to any increasing sequence of policies.*

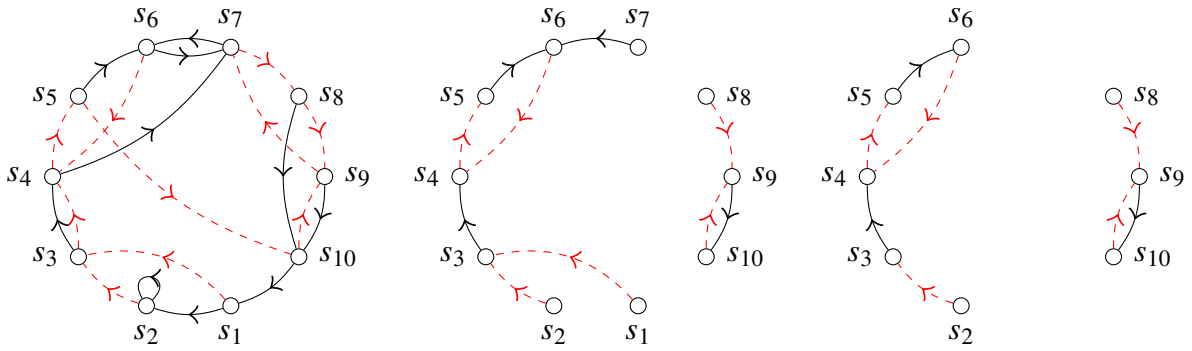

FIGURE 7. An example of a directed multigraph corresponding to a 10-state 2-action DMDP, the policy digraph G_π for $\pi =$ 0010101010, and the path-cycles $P^\pi_{s_2}$ and $P^\pi_{s_8}$ (from left to right), where the dashed (red) edges and solid (black) edges correspond to actions 0 and 1, respectively.

Proof. Let $1 < i \le \ell$. Since $\pi_{i-1} \prec \pi_i$, we have $V^{\pi_{i-1}}(s) < V^{\pi_i}(s)$ for some state $s \in S$. Now since $\pi_j \preceq \pi_{i-1}$ for $1 \le j < i$, $V^{\pi_j}(s) < V^{\pi_i}(s)$ for each $1 \le j < i$. If $P_s^{\pi_i}$ were a part of the path-cycle representation of π_j for some $1 \le j < i$, then $V^{\pi_j}(s) = V^{\pi_i}(s)$, a contradiction. □

Before we prove results about the number of steps PI takes to converge, we establish some graph theoretic notation and results, which shall be useful later.

For any digraph G, we denote the number of path-cycles and paths in G by $C'(G)$ and $C''(G)$, respectively. Clearly, $C(G) \le C'(G)$ since every cycle is a path-cycle.

For integers $n \ge 0$, $k \ge 2$, we define $\mathcal{G}^{n,k}$ as the set of all digraphs with n vertices and outdegree k (we allow digraphs to contain loops and multi-edges, as mentioned in Section 3). Note that if M is an n-state k-action DMDP, then $G_M \in \mathcal{G}^{n,k}$.

DEFINITION 15. For $G \in \mathcal{G}^{n,k}$, we define $N_1(G)$ to be the number of path-cycles in the digraph G' obtained from G by replacing each multi-edge of multiplicity k with a corresponding edge of multiplicity 1: that is, $N_1(G) = C'(G')$. Similarly, we define $N_2(G)$ to be the number of path-cycles in the skeleton of G: that is, $N_2(G) = C'(\mathrm{Skel}(G))$.

We prove below some bounds on $N_1(G)$ and $N_2(G)$ using the bounds on the number of cycles established in Section 3.

LEMMA 5. *Let $G \in \mathcal{G}^{n,k}_{\mathrm{simple}}$. Then $C'(G) \le n^2 k (k+1)!^{(n-1)/(k+1)}$.*

Proof. A path-cycle in G can be viewed as a pair consisting of a path and an edge from the terminal vertex of the path to a vertex in the path. Therefore, $C'(G) \le kC''(G)$. Let $s, t \in V(G)$ and the number of paths from s to t in G be denoted by $C''_{s,t}(G)$. Then $C''_{s,t}(G) \le C(G')$, where G' is the digraph obtained by deleting all incoming edges to s and outgoing edges from t in G followed by merging vertices s and t. It is easy to check that $G' \in \mathcal{G}^{n-1,k}_{\mathrm{simple}}$. Therefore, we have $C(G') \le (k+1)!^{(n-1)/(k+1)}$ from Theorem 6. Hence, $C''_{s,t}(G) \le (k+1)!^{(n-1)/(k+1)}$. Summing over all pairs of vertices $s, t \in V(G)$, we obtain $C''(G) \le n^2 (k+1)!^{(n-1)/(k+1)}$, which combined with $C'(G) \le kC''(G)$ yields the desired result. □

LEMMA 6. *Let $G \in \mathcal{G}^{n,k}_{\mathrm{multi}}$. Then $C'(G) \le 5n^2 k \alpha(k)^{n-1}$.*

Proof. As in the proof of Lemma 5 above, we have $C'(G) \le kC''(G)$ and $C''_{s,t}(G) \le C(G')$ for all $s, t \in V(G)$. Note that some vertices in G' could have outdegree less than k. We add a sufficient number of self-loops to each such vertex so that the outdegree of each vertex is equal to k in the resulting digraph G'', which satisfies $C(G') \le C(G'')$. Since $G'' \in \mathcal{G}^{n,k}_{\mathrm{multi}}$, we have $C(G'') \le 5\alpha(k)^{n-1}$ from Theorem 7. Therefore, $C''_{s,t}(G) \le 5\alpha(k)^{n-1}$. Summing over all pairs of

vertices $s, t \in V(G)$, we obtain $C''(G) \leq 5n^2\alpha(k)^{n-1}$, which combined with $C'(G) \leq kC''(G)$ yields the desired result. □

PROPOSITION 1. *Let $G \in \mathcal{G}^{n,k}$. Then we have*

(1) $N_1(G) \leq 5n^2k\alpha(k)^{n-1}$, *and*

(2) $N_2(G) \leq n^2k(k+1)!^{(n-1)/(k+1)}$.

Proof. Suppose that $G \in \mathcal{G}^{n,k}$. Let G' be the digraph obtained from G by replacing each multi-edge of multiplicity k with a corresponding edge of multiplicity 1. Note that some vertices in the digraph G' could have outdegree less than k. We add a sufficient number of self-loops to each such vertex so that the outdegree of each vertex is equal to k in the resulting digraph G'', which satisfies $C'(G') \leq C'(G'')$. Now, from Lemma 6, we obtain $C'(G'') \leq 5n^2k\alpha(k)^{n-1}$ since $G'' \in \mathcal{G}^{n,k}_{multi}$. Therefore, $N_1(G) = C'(G') \leq C'(G'') \leq 5n^2k\alpha(k)^{n-1}$. This finishes the proof of (1).

Now suppose that $G \in \mathcal{G}^{n,k}$. Note that $\mathrm{Skel}(G) \in \mathcal{G}^{n,k}_{simple}$. Therefore, by Lemma 5, we obtain $N_2(G) = C'(\mathrm{Skel}(G)) \leq n^2k(k+1)!^{(n-1)/(k+1)}$. This finishes the proof of (2). □

4.1. Policy iteration with arbitrary action selection. We begin by defining an equivalence relation on the edges of a DMDP digraph and an induced equivalence relation on the path-cycles therein.

DEFINITION 16. Let $M = (S, A, T, R, \gamma)$ be a DMDP. We say that a state $s \in S$ is *non-branching* if there exists $s' \in S$ such that $T(s, a, s') = 1$ for all $a \in A$. For any $a_1, a_2 \in A$, we say that the edges corresponding to (s, a_1) and (s, a_2) in G_M are equivalent if s is a non-branching state. Given two path-cycles P_1 and P_2 in G_M, we say that $P_1 \sim P_2$ if P_1 and P_2 differ only in equivalent edges.

Given a DMDP M, $\sim$ is an equivalence relation on the set of path-cycles in G_M.

Our approach for proving a bound on the number of policies visited by a PI algorithm with arbitrary action selection is as follows:

1. associate a sequence of distinct path-cycles to the sequence of policies visited (by Lemma 4),
2. show that this sequence does not contain many equivalent path-cycles (see Lemma 7), and
3. use Proposition 1 to bound the number of path-cycles in the digraph obtained by identifying edges under the equivalence relation defined in Definition 16.

In the following lemma, we show that a sequence of distinct path-cycles associated with an increasing sequence of policies obtained via policy improvement with arbitrary action selection does not contain many equivalent path-cycles.

LEMMA 7. *Let $M = (S, A, T, R, \gamma)$ be an n-state, k-action DMDP and $\pi_1 \prec \cdots \prec \pi_\ell$ be an increasing sequence of policies obtained via policy improvement with arbitrary action selection. Further, let $P = P_{s_1}^{\pi_1}, \ldots, P_{s_\ell}^{\pi_\ell}$ be an associated sequence of path-cycles. Then, the size of each equivalence class of P under the equivalence relation $\sim$ is at most kn.*

Proof. Let $S' \subseteq S$ be the set of non-branching states. Let $Q = P_s^{\pi_{i_1}}, \ldots, P_s^{\pi_{i_m}}$ be a subsequence of P consisting of equivalent path-cycles. And let $1 \leq j < m$. Then $P_s^{\pi_{i_j}}$ and $P_s^{\pi_{i_{j+1}}}$ differ in edges coming out of a non-branching state, say $\pi_{i_j}(s') = a$ and $\pi_{i_{j+1}}(s') = a'$ for some $s' \in S'$ and distinct actions $a, a' \in A$. We remark that $P_s^{\pi_{i_j}}$ and $P_s^{\pi_{i_{j+1}}}$ may additionally differ in other equivalent edges. Let P_s^{π} be a path-cycle that appears after $P_s^{\pi_{i_j}}$ in Q. We claim that $\pi(s') \neq a$.

To see this, note that there exists some $i' \in [i_j, i_{j+1})$ such that the action at state s' is switched from a to a'' ($\neq a$) while going from $\pi_{i'}$ to $\pi_{i'+1}$. Therefore, $V^{\pi_{i'}}(s') < Q^{\pi_{i'}}(s', a'')$. But since $s' \in S'$, this implies $R(s', a'') > R(s', a)$. Since s' is a non-branching state, it follows from a straightforward inductive argument that the sequence $\{R(s', \pi_j(s'))\}_{j=i'+1}^{\ell}$ is non-decreasing. Therefore, action a is not taken at the state s' in any policy that appears after $\pi_{i'}$ in the sequence $\pi_1, \ldots, \pi_\ell$. In particular, this implies $\pi(s') \neq a$.

Thus, each time we transition along Q from one path-cycle to the next, one edge is eliminated from G_M in the sense that it cannot be a part of subsequent path-cycles in Q. Hence, the number of path-cycles in Q is bounded above by the number of edges in G_M, which equals kn. □

In Theorem 8, we show that the length of any increasing sequence of policies obtained via policy improvement with arbitrary action selection (as in Lemma 7) for DMDP M is bounded above by $N_1(G_M)$ up to a multiplicative factor which is polynomial in n and k. Equivalently, one may view the following theorem as providing an upper bound on the length of the longest directed path in the PI-DAG of the DMDP M.

THEOREM 8. *Let M be an n-state, k-action DMDP and let $\pi_1 \prec \pi_2 \prec \cdots \prec \pi_\ell$ be an increasing sequence of policies obtained via policy improvement with arbitrary action selection on M. Then $\ell \leq knN_1(G_M)$.*

Proof. Let $\pi_1 \prec \pi_2 \prec \cdots \prec \pi_\ell$ be an increasing sequence of policies obtained via policy improvement with arbitrary action selection on M. And let $P = P_{s_1}^{\pi_1}, \ldots, P_{s_\ell}^{\pi_\ell}$ be an associated sequence of

path-cycles. Then, the number of path-cycles in P equals the sum of the sizes of the equivalence classes of P under $\sim$. Using Lemma 7, we obtain

$$\begin{aligned}\ell &\leq kn \times \text{the number of equivalence classes of } P \text{ under } \sim \\ &\leq knC'(G'_M),\end{aligned}$$

where G'_M is the digraph obtained from G_M by replacing each multi-edge of multiplicity k with a corresponding edge of multiplicity 1. Using $N_1(G_M) = C'(G'_M)$ in the above inequality yields the desired result. □

Proof of Theorem 1. Let M be an n-state, k-action DMDP and $\pi_1 \prec \pi_2 \prec \cdots \prec \pi_\ell$ be the sequence of policies in Π_M encountered during a run of a PI algorithm with arbitrary action selection on M. Then, using Theorem 8, we obtain $\ell \leq knN_1(G_M)$. Further, we have $N_1(G_M) \leq 5n^2k\alpha(k)^{n-1}$ from Proposition 1 (1), which implies $\ell \leq 5n^3k^2\alpha(k)^{n-1}$. Finally, since $\alpha(k) \geq k/2$, we obtain $\ell = O(n^3k\alpha(k)^n)$. □

4.2. Policy iteration with max-gain action selection. We begin by defining an equivalence relation on the edges of a DMDP digraph and an induced equivalence relation on the path-cycles therein.

DEFINITION 17. Let $M = (S, A, T, R, \gamma)$ be a DMDP. For $s \in S$ and $a_1, a_2 \in A$, we say that the edges corresponding to (s, a_1) and (s, a_2) in G_M are equivalent if there exists $s' \in S$ such that $T(s, a_1, s') = T(s, a_2, s') = 1$. Given two path-cycles P_1 and P_2 in G_M, we say that $P_1 \approx P_2$ if P_1 and P_2 differ only in equivalent edges.

Given a DMDP M, $\approx$ is an equivalence relation on the set of path-cycles in G_M. We define a stronger notion of equivalence between edges below, which shall be useful in the proof of Lemma 8.

DEFINITION 18. For $s \in S$ and $a_1, a_2 \in A$, we say that the edges corresponding to (s, a_1) and (s, a_2) are quasi-equal, denoted $(s, a_1) \equiv (s, a_2)$, if (s, a_1) and (s, a_2) are equivalent in the sense of Definition 17 and $R(s, a_1) = R(s, a_2)$.

Let $E_{\max}$ be the set of state-action pairs with the highest reward among the state-action pairs in their respective equivalence classes. Note that if $(s, a) \in E_{\max}$, then the intersection of $E_{\max}$ with the equivalence class of (s, a) is equal to the quasi-equality class of (s, a).

We use the same proof strategy as for PI with arbitrary action selection. In the following lemma, we show that a sequence of distinct path-cycles associated with an increasing sequence of policies

obtained via policy improvement with max-gain action selection does not contain many equivalent path-cycles.

LEMMA 8. *Let $M = (S, A, T, R, \gamma)$ be an n-state, k-action DMDP and $\pi_1 \prec \pi_2 \prec \cdots \prec \pi_\ell$ be an increasing sequence of policies obtained via policy improvement with max-gain action selection on M. Further, let $P = P_{s_1}^{\pi_1}, \ldots, P_{s_\ell}^{\pi_\ell}$ be an associated sequence of path-cycles. Then, the size of each equivalence class of P under the equivalence relation $\approx$ is at most $n+1$.*

Proof. Let $Q = P_s^{\pi_{i_1}}, \ldots, P_s^{\pi_{i_m}}$ be a subsequence of P consisting of equivalent path-cycles. Let $1 \leq j < m$. Then $P_s^{\pi_{i_j}}$ and $P_s^{\pi_{i_{j+1}}}$ differ in equivalent but not quasi-equal edges (due to strict improvement in the value function) coming out of a state, say (s', a_1) and (s', a_2), respectively, for some $s' \in S$ and distinct actions $a_1, a_2 \in A$. We remark that $P_s^{\pi_{i_j}}$ and $P_s^{\pi_{i_{j+1}}}$ may additionally differ in other equivalent edges. Let P_s^π be a path-cycle that appears after $P_s^{\pi_{i_j}}$ in Q. We claim that $(s', \pi(s')) \equiv (s', a_2)$.

To see this, note that there exists some $i' \in [i_j, i_{j+1})$ such that the action at state s' is switched to a_2 while going from $\pi_{i'}$ to $\pi_{i'+1}$. Since we are considering policy improvement with max-gain action selection, any edge that is switched to must be contained in $E_{\max}$. Therefore, $(s', a_2) \in E_{\max}$. Similarly, for any subsequent path-cycle P_s^π in Q, we have $(s', \pi(s')) \in E_{\max}$. Further, the edges (s', a_2) and $(s', \pi(s'))$ must be equivalent since all path-cycles in Q are equivalent. Therefore, we conclude that $(s', \pi(s')) \equiv (s', a_2)$.

Thus, each time we transition along Q from one path-cycle to the next, the action at one state becomes fixed in the sense that the edge coming out of that state cannot change its quasi-equality class in any subsequent transitions along the subsequence Q. Hence, there can be at most n transitions, implying $m \leq n+1$. □

In Theorem 9, we show that the length of any increasing sequence of policies obtained via policy improvement with max-gain action selection (as in Lemma 8) for DMDP M is bounded above by $N_2(G_M)$ up to a multiplicative factor which is polynomial in n and k.

THEOREM 9. *Let M be an n-state, k-action DMDP and let $\pi_1 \prec \pi_2 \prec \cdots \prec \pi_\ell$ be an increasing sequence of policies obtained via policy improvement with max-gain action selection on M. Then $\ell \leq (n+1)N_2(G_M)$.*

Proof. Let $\pi_1 \prec \pi_2 \prec \cdots \prec \pi_\ell$ be an increasing sequence of policies obtained via policy improvement with max-gain action selection. And let $P = P_{s_1}^{\pi_1}, \ldots, P_{s_\ell}^{\pi_\ell}$ be an associated sequence of

path-cycles. Then, the number of path-cycles in P equals the sum of the sizes of the equivalence classes of P under $\approx$. Using Lemma 8, we obtain

$$\begin{aligned}\ell &\leq (n+1) \times \text{the number of equivalence classes of } P \text{ under } \approx \\ &\leq (n+1)C'(\text{Skel}(G_M)).\end{aligned}$$

Now using $N_2(G_M) = C'(\text{Skel}(G_M))$ in the above inequality yields the desired result. □

Proof of Theorem 2. Let M be an n-state, k-action DMDP and let $\pi_1 \prec \pi_2 \prec \cdots \prec \pi_\ell$ be the sequence of policies in Π_M encountered during a run of a PI algorithm with max-gain action selection on M. Then, using Theorem 9, we obtain $\ell \leq (n+1)N_2(G_M)$. Further, we have $N_2(G_M) \leq n^2k(k+1)!^{(n-1)/(k+1)}$ from Proposition 1 (2), which implies $\ell \leq (n+1)n^2k(k+1)!^{(n-1)/(k+1)}$. Now, since $n+1 \leq 2n$ and $(k+1)!^{1/(k+1)} \geq (k+1)/e \geq k/e$, we obtain $\ell = O(n^3(k+1)!^{n/(k+1)})$.

We will now show that $(k+1)!^{1/(k+1)} = \left(1 + O\left(\frac{\log k}{k}\right)\right)\frac{k}{e}$ to finish the proof. From Stirling's approximation, we have $k! \leq ek^{k+1/2}e^{-k}$, which further yields $(k+1)! \leq e(k+1)k^{k+1/2}e^{-k}$. Taking the $(k+1)$-th root on both sides, we obtain $(k+1)!^{1/(k+1)} \leq \frac{k}{e}(e^2(k+1)k^{-1/2})^{1/(k+1)}$. Further, we have $(e^2(k+1)k^{-1/2})^{1/(k+1)} \leq (k+1)^{1/(k+1)}$ for $k \geq e^4$. Since $(k+1)^{1/(k+1)} = 1 + O(\frac{\log k}{k})$, we conclude that $(k+1)!^{1/(k+1)} = \left(1 + O\left(\frac{\log k}{k}\right)\right)\frac{k}{e}$. □

Theorem 1 and Theorem 2 provide upper bounds on the number of steps required by all and max-gain policy iteration algorithms to converge for DMDPs. As mentioned previously, Howard's PI (with max-gain action selection) is one of the most commonly used variants of PI. We make a few observations about this PI variant below.

REMARK 4. Combining the bounds in Theorem 1 and Theorem 2, the tightest upper bound that we have proved for Howard's PI on n-state k-action DMDPs is of the form $O(\text{poly}(n,k)\beta(k)^n)$, where $\beta(k)$ is given by

$$\beta(k) = \begin{cases} (1+\sqrt{5})/2, & \text{if } k = 2, \\ (k+1)!^{1/(k+1)}, & \text{for } k \geq 3. \end{cases}$$

REMARK 5. Post and Ye [48] derived several properties for the run of max-gain Simplex PI on DMDPs and used them to prove polynomial bounds on the complexity of max-gain Simplex PI. We observe that lemmas 5, 6, and 7 in their work also hold for the run of Howard's PI on DMDPs. Therefore, at most polynomial many steps elapse between the creation of new cycles during a run

of Howard's PI on a DMDP. This result, coupled with the fact that a cycle can be formed at most once during the entire run, allows one to bound above the number of steps taken by Howard's PI by the number of cycles in the skeleton of the DMDP digraph G_M up to a polynomial factor. This method yields similar upper bounds for Howard's PI on DMDPs as Theorem 2, albeit with a larger polynomial factor. However, this method cannot be used to establish bounds on the complexity of a general PI algorithm.

4.3. Policy iteration for average reward DMDPs. We refer the reader to Puterman [49, Chapter 8] for the general theory of average reward MDPs, and Hansen and Zwick [31, Section 2] for Howard's PI on average reward DMDPs. Below, we provide a brief overview of PI on average reward DMDPs before explaining why our main results also hold in this setting.

Let $M = (S, A, T, R)$ be a DMDP (note that there is no discount factor in the definition since we are in the average rewards setting). Let $\pi \in \Pi$. We begin by defining the expected infinite horizon average reward under the policy π, which is also called the *gain* of the policy π (denoted by g^π; not to be confused with the gain function ρ^π defined below). For $s \in S$, we define

$$g^\pi(s) := \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1} R(s_t, a_t)\right],$$

where $s_0 = s$, and for $t \geq 0$, $a_t = \pi(s_t)$, $s_{t+1} \sim T(s_t, a_t)$. We now define the *bias* of the policy π (denoted by b^π). Recall that the digraph G_π induced by the policy π is a union of connected components, each containing a unique cycle and possibly non-disjoint paths leading into the cycle. Let H be a connected component of G_π and let C be the unique cycle in H. We assume that the states in S are indexed by the set $\{1, 2, \ldots, n\}$. Let s_* be the state with the smallest index in C. As a convention, we define

$$b^\pi(s_*) := 0.$$

For any state s in H, suppose that we begin at s and follow the policy π to reach s_* for the first time after ℓ steps. Then we obtain a corresponding path $s = s_0, s_1, \ldots, s_\ell = s_*$, and define

$$b^\pi(s) := \sum_{i=0}^{\ell-1} (R(s_i, \pi(s_i)) - g^\pi(s)).$$

We define the *value function* of the policy π by $V^\pi : S \to \mathbb{R}\times\mathbb{R}$, where

$$V^\pi(s) = (g^\pi(s), b^\pi(s)),$$

for $s \in S$. Further, we define the *action value function* of the policy π by $Q^\pi : S \times A \to \mathbb{R} \times \mathbb{R}$, where

$$Q^\pi(s,a) = (g^\pi(s'), R(s,a) - g^\pi(s') + b^\pi(s')),$$

for $s \in S$ and $a \in A$. Here, s' is the unique state in S such that $T(s,a,s') = 1$. Finally, we define the *gain function* of the policy π by $\rho^\pi : S \times A \to \mathbb{R} \times \mathbb{R}$, where

$$\rho^\pi(s,a) = Q^\pi(s,a) - V^\pi(s).$$

Here, the minus symbol denotes coordinate-wise subtraction. Gains are primary, and biases are secondary to the value of a state under a given policy. Therefore, we overload notation to use $<$ for the lexicographic ordering on $\mathbb{R} \times \mathbb{R}$. For $(x_1, y_1), (x_2, y_2) \in \mathbb{R} \times \mathbb{R}$, we define $(x_1, y_1) < (x_2, y_2)$ if $x_1 < x_2$ or ($x_1 = x_2$ and $y_1 < y_2$), and $(x_1, y_1) = (x_2, y_2)$ if $x_1 = x_2$ and $y_1 = y_2$. This overloading is useful since many of the definitions and proofs for discounted DMDPs hold as they are for average reward DMDPs, as we shall note below.

First, we define policy comparison and optimal policies using definitions 3 and 4, respectively. Next, we define the *improvable set* I^π as the set of state-action pairs $(s,a) \in S \times A$ such that $(0,0) < \rho^\pi(s,a)$. Finally, we define valid improvement sets and policy improvement using definitions 5 and 6, respectively. Then it is easy to show that the policy improvement theorem (Theorem 5) holds as it is, and PI algorithms proceed exactly in the same way as for discounted DMDPs (see Derman [17, Chapter 6]; see also Howard [34] and Puterman [49]). We reuse the definitions for arbitrary action selection and max-gain action selection from Section 2.3, with the understanding that $<$ is used to compare $\rho^\pi(s,a)$ values when computing $\arg\max_{a \in A^+(\pi,s)} \rho^\pi(s,a)$.

To show that theorems 1 and 2 hold for PI on average reward DMDPs, we need to verify that lemmas 4, 7, and 8 hold in this setting. A quick inspection of the proofs yields that exactly the same proofs work for these three lemmas. The key is that a path-cycle P_s^π determines both the gain and the bias, and hence the value of state s under policy π. This implies that a new path-cycle is formed during each step of PI since there is a strict increment in the value of some state.

5. Summary and Outlook. We have presented a new perspective on the policy space of DMDPs, which yields running-time upper bounds that apply to the entire family of PI algorithms. We obtain an even tighter upper bound for the family of PI algorithms that switch only to max-gain actions. Central to our analysis is the set of cycles in directed multigraphs induced by DMDPs (Madani [40], Post and Ye [48]). Our core results are upper bounds on the number of cycles in

certain families of directed multigraphs, which are defined based on the properties of corresponding PI algorithms. For 2-action MDPs, our results imply that PI can complete in strictly fewer iterations when transitions are constrained to be deterministic.

Our results give rise to several questions for further investigation.

• For the case of $k = 2$ actions, Melekopoglou and Condon [44] furnish an MDP and a PI variant that can visit *all* the 2^n policies for the MDP. It remains unknown if such a Hamiltonian path through the set of policies exists for any MDP with $k \geq 3$ actions. Interestingly, the tightest lower bound known for MDPs with $k \geq 3$ actions—of $\Omega(k^{n/2})$ iterations (Ashutosh *et al.* [9])—arises from a DMDP. There appears to be room to obtain a tighter lower bound for MDPs with $k \geq 3$ actions. Proving a "Hamiltonian" lower bound for $k \geq 3$ would imply that stochasticity makes MDPs harder to solve for PI, irrespective of the number of states and actions. A concrete first step in this pursuit could be to attempt constructing a 3-state, 3-action MDP in which all 27 policies can be visited by PI (it is not too hard to construct a 2-state, 3-action MDP with this property).

• Our upper bounds for PI on DMDPs depend directly on our upper bounds on the number of path-cycles in induced multigraphs. Our current analysis only utilises the fact that a path-cycle, once replaced, cannot appear again. Tighter upper bounds might be provable by exploiting additional constraints. For example, it might be possible to argue that path-cycles P_1 and P_2 mutually exclude each other in the trajectory taken by PI since they necessarily induce incomparable value functions. Similarly, if it can be shown that there is no legal sequence of switches to go from any policy containing path-cycle P_1 to any policy containing path-cycle P_2, it would imply that only one of P_1 and P_2 can be visited by PI.

A particularly interesting case is Howard's PI on DMDPs, for which Hansen and Zwick [31] conjectured an upper bound of nk steps. Notably, an $O(n^n)$ bound holds for Howard's PI, since any edge (action) that is switched to must have the highest reward among all edges with the same initial and next states. A natural question is whether one can establish an improved exponential upper bound independent of k, analogous to the $O(\text{poly}(n) \cdot 2^{n/2})$ bound for energy games and mean payoff games on graphs with n vertices (Dorfman *et al.* [18]).

• It is easy to check that the dual of the LP P_M induced by a DMDP M contains at most two variables per constraint (Littman *et al.* [39]; see Section 4.1). LPs with this property have been well-studied; for instance, it is known that they can be solved in strongly polynomial time (Megiddo [42]). Hence, our results for DMDPs can possibly be generalised to a larger class of LPs. In particular, one could attempt to furnish non-trivial upper bounds on path lengths in the primal

LP digraph when the dual is restricted to having at most two variables per constraint. Kitahara and Mizuno [37] prove a similar generalization of Ye's [59] results on the strong polynomiality of the simplex method, for MDPs with a fixed discount factor, to a broader class of LPs (including, for example, LPs with a totally unimodular constraint matrix and integer constraint vector).

Acknowledgments

We thank Pratyush Agarwal and Mulinti Shaik Wajid for carefully reading the manuscript and providing helpful suggestions. We also thank the anonymous reviewers and editors for their valuable comments, which helped improve the quality of this paper.

References

[1] Ahrens W (1897) Ueber das gleichungssystem einer Kirchhoff'schen galvanischen stromverzweigung. *Mathematische Annalen* 49(2):311–324.

[2] AlBdaiwi BF (2018) On the number of cycles in a graph. *Mathematica Slovaca* 68(1):1–10, URL http://dx.doi.org/doi:10.1515/ms-2017-0074.

[3] Aldred REL, Thomassen C (1997) On the number of cycles in 3-connected cubic graphs. *Journal of Combinatorial Theory, Series B* 71(1):79–84, ISSN 0095-8956, URL http://dx.doi.org/https://doi.org/10.1006/jctb.1997.1771.

[4] Aldred REL, Thomassen C (2008) On the maximum number of cycles in a planar graph. *Journal of Graph Theory* 57(3):255–264, URL http://dx.doi.org/https://doi.org/10.1002/jgt.20290.

[5] Allender EW (1985) On the number of cycles possible in digraphs with large girth. *Discrete Applied Mathematics* 10(3):211–225, ISSN 0166-218X, URL http://dx.doi.org/https://doi.org/10.1016/0166-218X(85)90044-7.

[6] Alt H, Fuchs U, Kriegel K (1999) On the number of simple cycles in planar graphs. *Combinatorics, Probability and Computing* 8(5):397–405, URL http://dx.doi.org/10.1017/S0963548399003995.

[7] Arman A, Gunderson DS, Tsaturian S (2016) Triangle-free graphs with the maximum number of cycles. *Discrete Mathematics* 339(2):699–711, ISSN 0012-365X, URL http://dx.doi.org/https://doi.org/10.1016/j.disc.2015.10.008.

[8] Arman A, Tsaturian S (2019) The maximum number of cycles in a graph with fixed number of edges. *The Electronic Journal of Combinatorics* 26(4), URL http://dx.doi.org/10.37236/8747.

[9] Ashutosh K, Consul S, Dedhia B, Khirwadkar P, Shah S, Kalyanakrishnan S (2020) Lower bounds for policy iteration on multi-action MDPs. *2020 59th IEEE Conference on Decision and Control (CDC)*, 1744–1749, URL http://dx.doi.org/10.1109/CDC42340.2020.9303956.

[10] Avis D, Moriyama S (2009) On combinatorial properties of linear program digraphs. *Polyhedral computation*, volume 48 of *CRM Proc. Lecture Notes*, 1–13 (Amer. Math. Soc., Providence, RI), ISBN 978-0-8218-4633-9, URL http://dx.doi.org/10.1090/crmp/048/01.

[11] Bellman R (1957) *Dynamic Programming* (Princeton, NJ, USA: Princeton University Press), 1 edition.

[12] Brègman LM (1973) Certain properties of nonnegative matrices and their permanents. *Dokl. Akad. Nauk SSSR* 211(1):27–30, ISSN 0002-3264.

[13] Buchin K, Knauer C, Kriegel K, Schulz A, Seidel R (2007) On the number of cycles in planar graphs. Lin G, ed., *Computing and Combinatorics*, 97–107 (Berlin, Heidelberg: Springer), ISBN 978-3-540-73545-8.

[14] Dantzig GB (1963) *Linear programming and extensions* (Princeton University Press, Princeton, NJ).

[15] de Mier A, Noy M (2012) On the maximum number of cycles in outerplanar and series–parallel graphs. *Graphs and Combinatorics* 28(2):265–275, ISSN 1435-5914, URL http://dx.doi.org/10.1007/s00373-011-1039-9.

[16] Delivorias PN, Richter RJ (1994) Maximum path digraphs. *Discrete Applied Mathematics* 50(3):221–237, ISSN 0166-218X, URL http://dx.doi.org/https://doi.org/10.1016/0166-218X(92)00032-H.

[17] Derman C (1970) *Finite state Markovian decision processes*, volume Vol. 67 of *Mathematics in Science and Engineering* (Academic Press, New York-London).

[18] Dorfman D, Kaplan H, Zwick U (2019) A faster deterministic exponential time algorithm for energy games and mean payoff games. *46th International Colloquium on Automata, Languages, and Programming*, volume 132 of *LIPIcs. Leibniz Int. Proc. Inform.*, Art. No. 114, 14 (Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern), ISBN 978-3-95977-109-2.

[19] Durocher S, Gunderson DS, Li PC, Skala M (2015) Cycle-maximal triangle-free graphs. *Discrete Mathematics* 338(2):274–290, ISSN 0012-365X, URL http://dx.doi.org/https://doi.org/10.1016/j.disc.2014.10.002.

[20] Dvořák Z, Morrison N, Noel JA, Norin S, Postle L (2021) Bounding the number of cycles in a graph in terms of its degree sequence. *European Journal of Combinatorics* 91:103206, ISSN 0195-6698, URL http://dx.doi.org/https://doi.org/10.1016/j.ejc.2020.103206.

[21] Entringer RC, Slater PJ (1981) On the maximum number of cycles in a graph. *Ars Combinatoria* 11:289–294.

[22] Fearnley J (2010) Exponential lower bounds for policy iteration. Abramsky S, Gavoille C, Kirchner C, Meyer auf der Heide F, Spirakis PG, eds., *Automata, Languages and Programming*, 551–562 (Berlin, Heidelberg: Springer Berlin Heidelberg), ISBN 978-3-642-14162-1.

[23] Feinberg EA, Huang J (2014) The value iteration algorithm is not strongly polynomial for discounted dynamic programming. *Oper. Res. Lett.* 42(2):130–131, ISSN 0167-6377,1872-7468, URL http://dx.doi.org/10.1016/j.orl.2013.12.011.

[24] Gerbner D, Keszegh B, Palmer C, Patkós B (2018) On the number of cycles in a graph with restricted cycle lengths. *SIAM J. Discret. Math.* 32(1):266–279, ISSN 0895-4801, URL http://dx.doi.org/10.1137/16M109898X.

[25] Golumbic MC, Perl Y (1979) Generalized Fibonacci maximum path graphs. *Discrete Mathematics* 28(3):237–245, ISSN 0012-365X, URL http://dx.doi.org/https://doi.org/10.1016/0012-365X(79)90131-6.

[26] Guichard DR (1996) The maximum number of cycles in graphs. *Congressus Numerantium* 121:211–215.

[27] Gupta A, Kalyanakrishnan S (2017) Improved strong worst-case upper bounds for MDP planning. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI '17*, 1788–1794 (AAAI Press), URL http://dx.doi.org/10.24963/ijcai.2017/248.

[28] Hansen TD (2012) *Worst-case Analysis of Strategy Iteration and the Simplex Method*. Ph.D. thesis, Aarhus University.

[29] Hansen TD, Kaplan H, Zwick U (2014) Dantzig's pivoting rule for shortest paths, deterministic MDPs, and minimum cost to time ratio cycles. *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, 847–860, SODA '14 (Society for Industrial and Applied Mathematics), ISBN 9781611973389.

[30] Hansen TD, Paterson M, Zwick U (2014) Improved upper bounds for random-edge and random-jump on abstract cubes. *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, 874–881, SODA '14 (Society for Industrial and Applied Mathematics), URL http://dx.doi.org/10.1137/1.9781611973402.65.

[31] Hansen TD, Zwick U (2010) Lower bounds for howard's algorithm for finding minimum mean-cost cycles. Cheong O, Chwa KY, Park K, eds., *Algorithms and Computation*, 415–426 (Berlin, Heidelberg: Springer Berlin Heidelberg), ISBN 978-3-642-17517-6.

[32] Hollanders R, Delvenne JC, Jungers RM (2012) The complexity of policy iteration is exponential for discounted markov decision processes. *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 5997–6002, URL http://dx.doi.org/10.1109/CDC.2012.6426485.

[33] Hollanders R, Gerencsér B, Delvenne JC, Jungers RM (2016) Improved bound on the worst case complexity of policy iteration. *Oper. Res. Lett.* 44(2):267–272, ISSN 0167-6377, URL http://dx.doi.org/10.1016/j.orl.2016.01.010.

[34] Howard RA (1960) *Dynamic Programming and Markov Processes* (Cambridge, MA: MIT Press).

[35] Kalyanakrishnan S, Mall U, Goyal R (2016) Batch-switching policy iteration. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 3147–3153, IJCAI '16 (AAAI Press), ISBN 9781577357704.

[36] Kalyanakrishnan S, Misra N, Gopalan A (2016) Randomised procedures for initialising and switching actions in policy iteration. *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 3145–3151, AAAI '16 (AAAI Press).

[37] Kitahara T, Mizuno S (2013) A bound for the number of different basic solutions generated by the simplex method. *Math. Program.* 137(1-2):579–586, ISSN 0025-5610,1436-4646, URL http://dx.doi.org/10.1007/s10107-011-0482-y.

[38] Knor M (1994) On the number of cycles in k-connected graphs. *Acta Mathematica Universitatis Comenianae* 63(2):315–321.

[39] Littman ML, Dean TL, Kaelbling LP (1995) On the complexity of solving markov decision problems. *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)* 394–402.

[40] Madani O, Thorup M, Zwick U (2010) Discounted deterministic Markov decision processes and discounted all-pairs shortest paths. *ACM Trans. Algorithms* 6(2):1–25, ISSN 1549-6325, URL http://dx.doi.org/10.1145/1721837.1721849.

[41] Mansour Y, Singh S (1999) On the complexity of policy iteration. *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, 401–408, UAI'99 (Morgan Kaufmann Publishers Inc.), ISBN 1558606149.

[42] Megiddo N (1983) Towards a genuinely polynomial algorithm for linear programming. *SIAM Journal on Computing* 12(2):347–353.

[43] Meier J (2008) *Groups, Graphs and Trees: An Introduction to the Geometry of Infinite Groups*. London Mathematical Society Student Texts (Cambridge University Press), URL http://dx.doi.org/10.1017/CBO9781139167505.

[44] Melekopoglou M, Condon A (1994) On the complexity of the policy improvement algorithm for Markov decision processes. *ORSA Journal on Computing* 6(2):188–192, URL http://dx.doi.org/10.1287/ijoc.6.2.188.

[45] Morrison N, Roberts A, Scott A (2021) Maximising the number of cycles in graphs with forbidden subgraphs. *Journal of Combinatorial Theory, Series B* 147:201–237, ISSN 0095-8956, URL http://dx.doi.org/https://doi.org/10.1016/j.jctb.2020.03.006.

[46] Papadimitriou CH, Tsitsiklis JN (1987) The complexity of Markov decision processes. *Mathematics of Operations Research* 12(3):441–450, URL http://dx.doi.org/10.1287/moor.12.3.441.

[47] Perl Y (1987) Digraphs with maximum number of paths and cycles. *Networks* 17(3):295–305, ISSN 0028-3045, URL http://dx.doi.org/10.1002/net.3230170305.

[48] Post I, Ye Y (2015) The simplex method is strongly polynomial for deterministic Markov decision processes. *Math. Oper. Res.* 40(4):859–868, ISSN 0364-765X, URL http://dx.doi.org/10.1287/moor.2014.0699.

[49] Puterman ML (1994) *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics (John Wiley & Sons, Inc., New York), ISBN 0-471-61977-9, a Wiley-Interscience Publication.

[50] Rautenbach D, Stella I (2005) On the maximum number of cycles in a hamiltonian graph. *Discrete Mathematics* 304(1):101–107, ISSN 0012-365X, URL http://dx.doi.org/https://doi.org/10.1016/j.disc.2005.09.007.

[51] Reid KB (1976) Cycles in the complement of a tree. *Discrete Mathematics* 15(2):163–174, ISSN 0012-365X, URL http://dx.doi.org/https://doi.org/10.1016/0012-365X(76)90082-0.

[52] Scherrer B (2016) Improved and generalized upper bounds on the complexity of policy iteration. *Math. Oper. Res.* 41(3):758–774, ISSN 0364-765X,1526-5471, URL http://dx.doi.org/10.1287/moor.2015.0753.

[53] Shi Y (1994) The number of cycles in a hamilton graph. *Discrete Mathematics* 133(1):249–257, ISSN 0012-365X, URL http://dx.doi.org/https://doi.org/10.1016/0012-365X(94)90031-0.

[54] Soules GW (2005) Permanental bounds for nonnegative matrices via decomposition. *Linear Algebra and its Applications* 394:73–89, ISSN 0024-3795, URL http://dx.doi.org/https://doi.org/10.1016/j.laa.2004.06.022.

[55] Szepesvári C (2010) *Algorithms for Reinforcement Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning (Morgan & Claypool Publishers).

[56] Takács L (1988) On the limit distribution of the number of cycles in a random graph. *Journal of Applied Probability* 25:359–376, ISSN 00219002, URL http://www.jstor.org/stable/3214169.

[57] Taraviya M, Kalyanakrishnan S (2020) A tighter analysis of randomised policy iteration. *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, volume 115 of *Proceedings of Machine Learning Research*, 519–529, URL http://proceedings.mlr.press/v115/taraviya20a.html.

[58] Volkmann L (1996) Estimations for the number of cycles in a graph. *Periodica Mathematica Hungarica* 33(2):153–161, ISSN 1588-2829, URL http://dx.doi.org/10.1007/BF02093512.

[59] Ye Y (2011) The simplex and policy-iteration methods are strongly polynomial for the Markov decision problem with a fixed discount rate. *Math. Oper. Res.* 36(4):593–603, ISSN 0364-765X,1526-5471, URL http://dx.doi.org/10.1287/moor.1110.0516.

[60] Zhou B (1988) The maximum number of cycles in the complement of a tree. *Discrete Mathematics* 69(1):85–94, ISSN 0012-365X, URL http://dx.doi.org/https://doi.org/10.1016/0012-365X(88)90180-X.