In Proc. IEEE CDC 2020, pp. 1744-1749, IEEE, 2020.

Lower Bounds for Policy Iteration on Multi-action MDPs

Kumar Ashutosh[†], Sarthak Consul[†], Bhishma Dedhia[†], Parthasarathi Khirwadkar[†], Sahil Shah[†],
and Shivaram Kalyanakrishnan[✉]

Abstract— **Policy Iteration (PI) is a classical family of algorithms to compute an optimal policy for any given Markov Decision Problem (MDP). The basic idea in PI is to begin with some initial policy and to repeatedly update the policy to one from an improving set, until an optimal policy is reached. Different variants of PI result from the (switching) rule used for improvement. An important theoretical question is how many iterations a specified PI variant will take to terminate as a function of the number of states n and the number of actions k in the input MDP. While there has been considerable progress towards upper-bounding this number, there are fewer results on lower bounds. In particular, existing lower bounds primarily focus on the special case of $k = 2$ actions. We devise lower bounds for $k \geq 3$. Our main result is that a particular variant of PI can take $\Omega(k^{n/2})$ iterations to terminate. We also generalise existing constructions on 2-action MDPs to scale lower bounds by a factor of k for some common deterministic variants of PI, and by $\log(k)$ for corresponding randomised variants.**

I. Introduction

Markov Decision Problems (MDPs) [1][2] are a popular abstraction of sequential decision making tasks in stochastic environments. An MDP is a tuple $\langle S, A, T, R, \gamma \rangle$, where S is a set of states and A is a set of actions. $T : S \times A \times S \to [0, 1]$ is a function such that $T(s, a, s')$ is the probability of reaching state $s' \in S$ from state $s \in S$ by taking action $a \in A$. The reward function $R : S \times A \to \mathbb{R}$, assigns a bounded reward $R(s, a)$ when the agent takes action $a \in A$ from state $s \in S$.

An MDP serves as an environment, which describes the consequences of an agent's actions. The agent itself has control only over its own behaviour, encapsulated as a *policy* $\pi : S \to A$ (by this definition, policies are Markovian, stationary, and deterministic—sufficient for our purposes). If $r_0, r_1, r_2, \ldots$ denotes the sequence of rewards obtained by an agent that follows policy π, starting at state $s \in S$, then its expected long-term reward

$$V^{\pi}(s) \stackrel{\text{def}}{=} \mathbb{E}_{\pi,s}[r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots] \tag{1}$$

is denoted the *value* of s under π; $V^{\pi} : S \to \mathbb{R}$ is the *value function* of π. In (1), $\gamma \in [0, 1]$ is a discount factor. In general, γ is set to be less than 1 so that the *infinite discounted reward* is well-defined. However, we may set $\gamma = 1$, thereby taking value to be the *total reward*, when trajectories in the input MDP are guaranteed to reach a terminal state. In this paper, we adopt the total reward formulation, but our results can all be extended to the infinite discounted setting.

Every MDP is guaranteed to have an *optimal* policy $\pi^{\star} : S \to A$ whose value at each state is at least as large as any other policy's [1]. Hence, given an MDP, a natural objective is to compute an optimal policy for it. There are many approaches to this planning problem, among them Value Iteration and Linear Programming [3]. In this paper, we consider a third popular approach: Policy Iteration (PI).

PI [4] is based on the Policy Improvement Theorem, which facilitates a relatively straightforward computation of a set of locally-improving policies $\mathbf{IP}(\pi)$ for any given policy π. $\mathbf{IP}(\pi)$ is represented implicitly through "improvable states" for π, as well as "improving actions" for such states. If π is optimal, $\mathbf{IP}(\pi)$ is guaranteed to be empty; if not, every policy $\pi' \in \mathbf{IP}(\pi)$ strictly dominates π in terms of state values. Indeed every such policy $\pi' \in \mathbf{IP}(\pi)$ is obtained by switching the actions taken by π in some improvable states to corresponding improving actions.

Given an arbitrary initial policy π_0, a PI algorithm generates a sequence of policies $\pi_0, \pi_1, \ldots, \pi_T$ wherein $\pi_{t+1} \in \mathbf{IP}(\pi_t)$ for $t = 0, 1, \ldots, T-1$, and π_T is an optimal policy. Even for the same MDP and starting policy π_0, different PI variants could select improving policies in different ways, thereby yielding different sequences. In this paper, our aim is to *lower-bound* the length of these sequences. We restrict our attention to finite MDPs, assuming that S shall comprise n non-terminal states and a constant number of terminal states. We take $A = \{0, 1, \ldots, k-1\}$; thus $|A| = k$. With this setup, observe that policies can be viewed as n-length k-ary strings.

Since PI increases some state value in each iteration, it cannot visit the same policy more than once. Hence, k^n, which is the total number of policies, serves as a trivial upper bound on the iterations taken by every PI variant. Howard's PI [4], a classical variant, has been shown to incur no more than $O(k^n/n)$ iterations [5]. Among upper bounds that are solely in terms of n and k, the tightest are $O(k^{0.7019n})$ iterations for deterministic PI variants [6], and $O\left((2 + \ln(k-1))^n\right)$ expected iterations for randomised variants [7]. Even tighter upper bounds (still exponential in n) have been shown for $k = 2$ [6]. Interestingly, the only *lower* bounds that have been shown for PI are either for the special case of $k = 2$ [8][9] or when k is related to n [10][11]. We contribute lower bounds for arbitrary $n \geq 2$, $k \geq 2$.

Every PI variant must choose which improvable states to switch. Notably, this is all that PI needs to do on 2-action MDPs, since selecting an improvable state fixes the improving action. The main technical difference that arises on k-action MDPs, $k \geq 3$, is that there can be multiple improving actions associated with an improvable state, and

[†] indicates equal contribution. [✉] indicates corresponding author. All authors are affiliated to the Indian Institute of Technology Bombay, Mumbai, India. E-mail: {kumar.ashutosh@, sarthakconsul@, bhishma@, parthasarathi.k@, sahilshah@cse., shivaram@cse.}iitb.ac.in.

PI must additionally choose among them. We consider both deterministic and randomised strategies for action selection. Our main contribution is a novel MDP construction that yields a trajectory of length $\Omega(k^{n/2})$ for a particular deterministic variant of PI. From a theoretical perspective, it is significant that the base of the exponent is an increasing (in fact polynomial) function of k. We also generalise existing constructions for 2-action MDPs, scaling lower bounds by a factor of k for some deterministic PI variants, and by $\log(k)$ for some randomised variants. We present our constructions in sections IV–VI, after first formalising PI in Section II and discussing existing lower bounds in Section III. We present conclusions and discuss future directions in Section VII.

II. Policy Iteration

In this section, we describe Policy Iteration (PI), borrowing notation from previous work [6][7]. Note that for any given policies π and π', the relation $\pi \succeq \pi'$ means that for all $s \in S$, $V^{\pi}(s) \geq V^{\pi'}(s)$. If $\pi \succeq \pi'$, and for some $s \in S$, $V^{\pi}(s) > V^{\pi'}(s)$, then we also have $\pi \succ \pi'$.

Policy evaluation. Each iteration of PI considers some policy π, and begins by computing its value function V^{π}. From the definition in (1), it is seen that V^{π} satisfies a set of linear equations (called Bellman's Equations): for $s \in S$,

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

The "action value function" of π, $Q^{\pi} : S \times A \to \mathbb{R}$, is defined as follows: for $s \in S, a \in A$, $Q^{\pi}(s,a)$ is the expected long-term reward the agent receives if it takes action a from state s for the first time-step, and then follows policy π. Thus,

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

Policy improvement. Define $\mathbf{IS}(\pi)$ to be the set of states s on which π is *not* greedy with respect to its own action-value function: that is,

$$\mathbf{IS}(\pi) \stackrel{\text{def}}{=} \left\{ s \in S : Q^{\pi}(s,\pi(s)) < \max_{a \in A} Q^{\pi}(s,a) \right\}.$$

For each state $s \in \mathbf{IS}(\pi)$, the set of improving actions $\mathbf{IA}(\pi,s)$ is defined as:

$$\mathbf{IA}(\pi,s) \stackrel{\text{def}}{=} \{a \in A : Q^{\pi}(s,a) > Q^{\pi}(s,\pi(s))\}.$$

If $\mathbf{IS}(\pi)$ is not empty, let π' be a policy that takes some action from $\mathbf{IA}(\pi,s)$ for one or more states $s \in \mathbf{IS}(\pi)$, and takes the same action as π in the remaining states. In other words, π' satisfies

$$\begin{aligned} &\exists s \in S : \pi'(s) \in \mathbf{IA}(\pi,s), \text{ and} \\ &\forall s \in S : (\pi'(s) = \pi(s)) \vee (\pi'(s) \in \mathbf{IA}(\pi,s)). \end{aligned} \tag{2}$$

Denote the set of all π' satisfying (2) as the set $\mathbf{IP}(\pi)$:

$$\mathbf{IP}(\pi) \stackrel{\text{def}}{=} \{\pi' \in \Pi : \pi' \text{ satisfies (2)}\}.$$

The Policy Improvement Theorem shows that every policy $\pi' \in \mathbf{IP}(\pi)$ improves upon (or *dominates*) π as follows.

Theorem 1 (Policy improvement): For every $\pi : S \to A$:
(1) if $\mathbf{IS}(\pi) \neq \emptyset$, then for all $\pi' \in \mathbf{IP}(\pi)$, $\pi' \succ \pi$;
(2) if $\mathbf{IS}(\pi) = \emptyset$, then for all $\pi' : S \to A$, $\pi \succeq \pi'$.

The proof of this well-known theorem is available from many sources [7][12].

Switching rules. For a given policy π, it is immediate that $\mathbf{IS}(\pi)$ and $\mathbf{IA}(\pi,\cdot)$—which implicitly represent $\mathbf{IP}(\pi)$—can be computed using $\text{poly}(n,k)$ arithmetic operations. The overall running-time of the algorithm may therefore be obtained by multiplying this per-iteration complexity with the total number of iterations taken to terminate. In turn, the number of iterations is determined by the rule used to pick $\pi' \in \mathbf{IP}(\pi)$ as the policy following π.

Recall that π' is obtained by modifying π: by selecting one or more states from $s \in \mathbf{IS}(\pi)$, and switching to some action from $\mathbf{IA}(\pi,s)$ for such states s. The most common variant of PI, called Howard's PI or Greedy PI [4], switches *every* state $s \in \mathbf{IS}(\pi)$. By contrast, under the Random PI variant [5], a non-empty subset of $\mathbf{IS}(\pi)$ is selected uniformly at random, and the states within this subset are switched. Under Simple PI [8], which is yet another variant, only a single improvable state is switched. Assuming a fixed indexing of states for the entire run of the algorithm, in each iteration the improvable state with the largest index is switched.

In 2-action MDPs, it suffices to specify which states to switch, since an improvable state will have exactly one improving action. On the other hand, if there are $k \geq 3$ actions, one might encounter improvable states with multiple improving actions, requiring yet another decision to be made.

- A common strategy for action-selection is to pick an action that maximises the Q-value: that is, setting $\pi'(s) \leftarrow \text{argmax}_{a \in A} Q^{\pi}(s,a)$ for a selected improvable state $s \in \mathbf{IS}(\pi)$. In this paper, we are unable to furnish meaningful lower bounds for this "max-Q" strategy. We make headway with two other natural approaches.
- Our first, "index-based" action-selection strategy assumes a fixed indexing of actions for the entire run of the algorithm, and always switches to the improving action with the smallest index. Since we have assumed $A = \{0,1,\ldots,k-1\}$, we set $\pi'(s) \leftarrow \min(\mathbf{IA}(\pi,s))$.
- Our second, "random" strategy sets $\pi'(s)$ to an action picked uniformly at random from $\mathbf{IA}(\pi,s)$.

We couple these action-selection strategies with several state-selection strategies and then lower-bound the number of iterations taken by the resulting PI variants. Before presenting our contributions, we review existing lower bounds for PI.

III. Existing Lower Bounds

For n-state, 2-action MDPs, Melekopoglou and Condon [8] show that Simple PI can take $\Omega(2^n)$ iterations to terminate. In Section VI, we generalise both their construction and their proof to $k \geq 2$, obtaining lower bounds of $\Omega(k \cdot 2^n)$ and $\Omega(\log(k) \cdot 2^n)$ when Simple PI is applied with index-based and random action selection, respectively.

The tightest lower bounds known for Howard's PI [9] and Random PI [6] on n-state, 2-action MDPs are only $\Omega(n)$.

Hansen and Zwick [9] construct a deterministic MDP on which, under the "average reward" criterion [13], Howard's PI can take as many as $2n - O(1)$ iterations. We show linear dependence on n using a simpler construction, and obtain linear and logarithmic scaling in k for index-based and random action selection, respectively (see Section V). Interestingly, our construction also implies a lower bound of $\Omega(kn)$ (or $\Omega(\log(k) \cdot n)$) iterations for index-based (respectively, random) action selection regardless of the state-selection strategy used.

Indeed a trajectory of exponential length ($\Omega(2^{n/7})$) has been shown for Howard's PI both under the total reward [10] and infinite discounted reward [11] settings. However, the MDPs used in these constructions do not have a constant number of actions per state—rather, this number is itself $\theta(n)$. Yet another exponential lower bound (of $\Omega(2^{n/2})$ iterations) has been shown for Howard's PI on a class of objects called Acyclic Unique Sink Orientations (AUSOs), which may be derived from n-state, 2-action MDPs [14]. The proof does not imply the same bound for MDPs [6].

The bounds mentioned above, and also the ones we provide, only depend on n and k. While there are upper bounds for PI in terms of parameters such as the discount factor, we are not aware of any such lower bounds.

IV. A Trajectory of Length $\Omega(k^{n/2})$

In this section, we propose a novel family of n-state, k-action MDPs on which a particular variant of PI can take $\Omega(k^{n/2})$ iterations to terminate. This lower bound becomes the tightest shown yet for the PI family. In subsequent sections, we generalise lower bounds for specific, commonly-used variants of PI to $k \geq 2$, but the resulting bounds are only linear or logarithmic in k.

A. Construction of Family $F(m,k)$

We construct a family of MDPs with $n = 2m$ non-terminal states, $m \geq 1$, a single terminal state, and k-actions, as shown in Fig. 1. The idea behind the construction is to implement a k-ary "counter" on a set of non-terminal states $s_1, s_2, \dots, s_m$, ensuring that all k^m sub-policies on these states are visited. To this end, we employ a "partner" state s'_i for each such state s_i, $i \in \{1, 2, \dots, m\}$. Recall that $A = \{0, 1, \dots, k-1\}$.

As shown in Fig. 1, all transitions in $F(m,k)$ are deterministic. Moreover, each state s_i in the counter and its partner s'_i have identical next states and rewards for each action. From state s_1 all actions $j \in A$ lead to the terminal state s_T. From state s_i, $i \in \{2, 3, \dots, m\}$, action 0 alone leads to s'_{i-1}, while actions $j \in A \setminus \{0\}$ all lead to s_{i-1}. For $i \in \{1, 2, \dots, m\}, j \in A$, the associated reward is $R(s_i, j) = jk^{m-i}$. Observe that there can be at most m transitions before termination; no discounting is used in the calculation of values.

B. Policies

We find it convenient to denote policies for $F(m,k)$ in the form $x \cdot y$, where $x, y \in A^m$. In this notation, the sequence $x = x_1 x_2 \dots x_m$ lists the actions taken from states $s_1, s_2, \dots, s_m$, respectively, and $y = y_1 y_2 \dots y_m$ does the same for states $s'_1, s'_2, \dots, s'_m$, respectively. For every $x \in A^m$ and $r \in \{0, 1, 2, \dots, m\}$, let $\mathtt{pre}(x : r)$ denote the prefix sequence $x_1 x_2 \dots x_r$. This (possibly empty) sequence may be viewed as a sub-policy on the counter states or the partner states.

Our proof relies on associating numbers with policies. For every sequence $x = x_1 x_2 \dots x_r$, where $r \geq 1$ and $x_u \in A$ for $u \in \{1, 2, \dots, r\}$, let $[x]$ denote the natural number represented in base k by x: that is, $[x] \overset{\text{def}}{=} \sum_{u=1}^{r} x_u k^{r-u}$. Let N denote the set of numbers $\{0, 1, \dots, k^m - 1\}$. It is immediately clear that A^m, which is the set of m-length k-ary sequences, is in 1-to-1 correspondence with N, each $x \in A^m$ associated with $[x] \in N$.

Of especial interest to us is policies of the form $x \cdot x$ for $x \in A^m$: we refer to such policies as *balanced* policies. Since every counter state s_i and its partner s'_i, $i \in \{1, 2, \dots, m\}$, have the same outgoing transitions and rewards in $F(m,k)$, it follows that $V^{x \cdot x}(s_i) = V^{x \cdot x}(s'_i)$. Moreover, since all transitions either terminate or move to states with lower indices, these values only depend on $\mathtt{pre}(x : i)$. Incorporating the corresponding rewards, we observe:

$$V^{x \cdot x}(s_i) = V^{x \cdot x}(s'_i) = \sum_{u=1}^{i} x_i k^{m-u} = k^{m-i}[\mathtt{pre}(x : i)], \qquad (3)$$

and in particular, $V^{x \cdot x}(s_m) = V^{x \cdot x}(s'_m) = [x]$. The format in (3) is convenient to establish a key property of $F(m,k)$.

Proposition 2 (Comparability of balanced policies): For $x, y \in A^m$, if $[y] > [x]$, then $y \cdot y \succ x \cdot x$.

Proof: "$[y] > [x]$" is equivalently stated as: "there exists $r \in \{0, 1, 2, \dots, m-1\}$ such that for $u \in \{0, 1, \dots, r\}$, $[\mathtt{pre}(y : u)] = [\mathtt{pre}(x : u)]$ and for $u \in \{r+1, r+2, \dots, m\}$, $[\mathtt{pre}(y : u)] > [\mathtt{pre}(x : u)]$. From (3), it follows that for $i \in \{1, 2, \dots, r\}$, $V^{y \cdot y}(s_i) = V^{y \cdot y}(s'_i) = V^{x \cdot x}(s_i) = V^{x \cdot x}(s'_i)$, and for $i \in \{r+1, r+2, \dots, m\}$, $V^{y \cdot y}(s_i) = V^{y \cdot y}(s'_i) > V^{x \cdot x}(s_i) = V^{x \cdot x}(s'_i)$, in turn implying that $y \cdot y \succ x \cdot x$. ■

The proposition is seen to induce a total order on policies of the form $x \cdot x$ via their value functions. The maximal element, $k^m \cdot k^m$, is also the sole optimal policy for $F(m,k)$. Our proof will construct a trajectory for PI that visits each balanced policy; notice that there are $k^m = k^{n/2}$ in total.

At this point, one might wonder why we need the partner states in $F(m,k)$ at all. Consider an MDP $F'(m,k)$ that results from removing partner states from $F(m,k)$ and redirecting their incoming transitions to corresponding counter states. On $F'(m,k)$, there would be a total order on the *entire* set of k^m polices, suggesting the possibility of an even tighter—in fact maximally tight—lower bound. However, crucially, it does not appear possible to get any *PI variant* to visit all k^m policies in $F'(m,k)$. Although PI guarantees a dominating policy after each step, it is not necessary that every policy π' that dominates π is reachable from π using PI. With partner states, indeed we are able to show a chain of length k^m for PI, but consequently m is only half the number of states.

C. A Long Trajectory for PI

We now present the main structural property of $F(m,k)$: that there is a sequence of policy improvements from every non-optimal balanced policy to its successor.

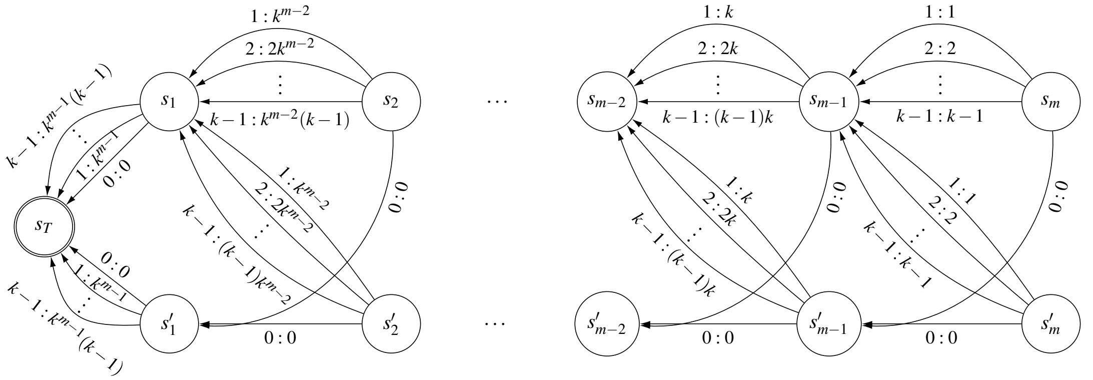

Fig. 1. The deterministic MDP $F(m,k)$ with $2m$ non-terminal states, a single terminal state s_T, and k actions. States $s_1, s_2, \ldots, s_m$ implement a k-ary "counter"; each has an associated partner state. Each edge, labeled "action: reward" represents the corresponding transitions. No discounting is used.

Lemma 3 (Segments of Long PI Trajectory): Consider $x, y \in A^m$ such that $[y] = [x] + 1$. There is a sequence of policies $\pi_1, \pi_2, \ldots, \pi_{t+1}$, $t \geq 2$, for $F(m,k)$ such that $\pi_1 = x \cdot x$; $\pi_{t+1} = y \cdot y$; and for $i \in \{1, 2, \ldots, t\}$, $\pi_{i+1} \in \mathbf{IP}(\pi_i)$.

Proof: We furnish a proof by showing a chain of policy improvements from $x \cdot x$ to $x \cdot y$, and another chain from $x \cdot y$ to $y \cdot y$. For the proof, we find it useful to denote as $I(z)$, for $z \in A^m \setminus \{(k-1)^m\}$, the largest index of z whose value is not $k-1$. Also, we write the concatenation of sequences z_1 and z_2 as $z_1 \ \# \ z_2$. With this notation, $[y] = [x] + 1$ implies

$$x = \mathtt{pre}(x : I(x) - 1) \ \# \ x_{I(x)} \ \# \ (k-1)^{m-I(x)}, \text{ and}$$
$$y = \mathtt{pre}(x : I(x) - 1) \ \# \ x_{I(x)} + 1 \ \# \ 0^{m-I(x)}.$$

We show that PI can lead from $x \cdot x$ to $x \cdot y$ by switching, in sequence, the states $s'_{I(x)}, s'_{I(x)+1}, \ldots, s'_m$; thereafter, switching $s_m, s_{m-1}, \ldots, s_{I(x)}$ in sequence leads from $x \cdot y$ to $y \cdot y$. Concretely, for $r \in \{1, 2, \ldots, m - I(x) + 1\}$, define

$$p_r \overset{\text{def}}{=} \mathtt{pre}(x : I(x) - 1) \ \# \ x_{I(x)} + 1 \ \# \ 0^{r-1} \ \# \ (k-1)^{m-I(x)-r+1};$$
$$q_r \overset{\text{def}}{=} \mathtt{pre}(x : I(x) - 1) \ \# \ x_{I(x)} \ \# \ (k-1)^{m-I(x)-r+1} \ \# \ 0^{r-1}.$$

We establish that the following sequences of policy improvements can be performed.

$$x \cdot x \to x \cdot p_1 \to x \cdot p_2 \to \cdots \to x \cdot p_{m-I(x)+1} = x \cdot y;$$
$$x \cdot y = q_1 \cdot y \to q_2 \cdot y \to \cdots \to q_{m-I(x)+1} \cdot y \to y \cdot y.$$

For the first chain, observe that $x \cdot x$ and $x \cdot p_1$ differ only in one action: on state $s'_{I(x)}$, $x \cdot x$ takes action $x_{I(x)}$ and $x \cdot p_1$ takes $x_{I(x)} + 1$. Using the structure of $F(m,k)$ and (3), we get

$$\begin{aligned} Q^{x \cdot x}(s'_{I(x)}, x_{I(x)} + 1) &= (x_{I(x)} + 1)k^{m-I(x)} + V^{x \cdot x}(s_{I(x)-1}) \\ &> x_{I(x)} k^{m-I(x)} + V^{x \cdot x}(s_{I(x)-1}) \\ &= V^{x \cdot x}(s'_{I(x)}), \end{aligned}$$

with the convention that $s_0 = s'_0 = s_T$. Now, for $r \in \{1, 2, \ldots, m - I(x)\}$, policies $x \cdot p_r$ and $x \cdot p_{r+1}$ take actions $k-1$ and 0 at state $s'_{I(x)+r}$, respectively, but on other states are alike. Substituting values calculated using the structure of $F(m,k)$, we get

$$\begin{aligned} Q^{x \cdot p_r}(s'_{I(x)+r}, 0) &= 0 + V^{x \cdot p_r}(s'_{I(x)+r-1}) \\ &= V^{x \cdot p_r}(s_{I(x)+r-1}) + k^{m-I(x)-r+1} \\ &> V^{x \cdot p_r}(s_{I(x)+r-1}) + (k-1) \cdot k^{m-I(x)-r} \\ &= V^{x \cdot p_r}(s_{I(x)+r}). \end{aligned}$$

Intuitively, action 0 is improving because the decrease in immediate reward on switching from action $k-1$ to 0 at state $s'_{I(x)+r}$ is offset by the gain from moving to state $s'_{I(x)+r-1}$ instead of $s_{I(x)+r-1}$. Recall that counter states follow x, while partner states follow p_{r+1}, with a higher-index action at $I(x)$.

For the second chain we first show that for $r \in \{1, 2, \ldots, m - I(x)\}$, $q_{r+1} \cdot y$ is improvable over $q_r \cdot y$. Note that the two policies take actions 0 and $k-1$ at state s_{m-r+1} respectively, but are alike at all other states. Substituting values based on $F(m,k)$, we get

$$\begin{aligned} Q^{q_r \cdot y}(s_{m-r+1}, 0) &= 0 + V^{q_r \cdot y}(s'_{m-r}) \\ &= k^r + V^{q_r \cdot y}(s_{m-r}) \\ &> (k-1) \cdot k^{r-1} + V^{q_r \cdot y}(s_{m-r}) \\ &= V^{q_r \cdot y}(s_{m-r+1}). \end{aligned}$$

Lastly, we need to show that the policy $y \cdot y$ improves over $q_{m-I(x)+1} \cdot y$. Since the policies differ only at $s_{I(x)}$, showing

$$\begin{aligned} &Q^{q_{m-I(x)+1} \cdot x}(s_{I(x)}, x_{I(x)} + 1) \\ &\quad = (x_{I(x)} + 1)k^{m-I(x)} + V^{q_{m-I(x)+1} \cdot x}(s_{I(x)-1}) \\ &\quad > x_{I(x)} k^{m-I(x)} + V^{q_{m-I(x)+1} \cdot x}(s_{I(x)-1}) \\ &\quad = V^{q_{m-I(x)+1} \cdot x}(s_{I(x)}) \end{aligned}$$

concludes the proof. ■

In short, we have demonstrated that a sequence of policy improvements, each switching only a single state, can take us from $x \cdot x$ to $y \cdot y$. We denote the variant of PI that facilitates such a trajectory *Peculiar PI*. For illustration, Appendix A shows the sequence of policies visited by Peculiar PI on $F(3,3)$.[1] While it might appear that going from $x \cdot x$ to $y \cdot y$ requires keeping an intermediate sequence of policies in memory, indeed Peculiar PI can be implemented concisely

[1] Appendices are provided in a version available at `https://arxiv.org/abs/2009.07842`.

as a memoryless variant, as shown in Appendix B. From Lemma 3, it is clear that if initialised with policy $0^m \cdot 0^m$, this variant will visit all k^m balanced policies.

Theorem 4 ($\Omega(k^{n/2})$ Lower Bound for Peculiar PI): On $F(m,k)$, if initialised with policy $0^m \cdot 0^m$, Peculiar PI takes $\Omega(k^m)$ iterations.

Although this lower bound—and those from the next two sections—are shown using the total reward setting, they continue to hold with discounting (see Appendix C).

V. Generic Lower Bounds

In this section, we give k-dependent lower bounds for *every* PI variant that uses index-based or random action selection; that is, the state-selection strategy can be arbitrary.

A. Construction

Fig. 2 shows our family of MDPs $G(n,k)$ with non-terminal states $s_1, s_2, \dots, s_n$. Rewards are only given on reaching terminal states, of which there are $n+1$.[2] From each state s_i, $i \in \{1,2,\dots,n\}$, action 0 deterministically terminates with a reward of -2^i. On the other hand, action $k-1$ moves deterministically from s_i to s_{i+1} for $i \in \{1,2,\dots,n-1\}$, and moves s_n into a terminal state with no reward.

As before, let us denote policies as n-length, k-ary strings. Observe that for $i \in \{1,2,\dots,n\}$, the policy $0^i(k-1)^{n-i}$ has exactly one improvable state: s_i. If the only actions were 0 and $k-1$, any PI variant initialised with 0^n would be forced to visit all n of these policies. To get PI to also take more actions from $\{1,2,\dots,k-2\}$, we implement *stochastic* transitions for each of these actions. In particular, action $j \in A \setminus \{0, k-1\}$ behaves like 0 with probability p_j, and like $k-1$ with probability $1-p_j$, where $p_j = \frac{1}{2} + \frac{k-j}{2k}$. The intuition behind this construction is that (1) so long as state s_{i+1}, $i \in \{1,2,\dots,n-1\}$ follows any action other than $k-1$, action 0 is the most rewarding at s_i; (2) once s_{i+1} switches to $k-1$, actions in $A \setminus \{0\}$ become profitable at s_i. Concretely, we obtain the following structure within the set of policies.

Lemma 5: For $i \in \{1,2,\dots,n\}$, $j \in \{0,1,\dots,k-2\}$ let $\pi_{ij} = 0^{i-1}j(k-1)^{n-i}$. Then

$$\mathbf{IS}(\pi_{ij}) = \{i\}, \text{ and}$$
$$\mathbf{IA}(\pi_{ij}, i) = \{j+1, j+2, \dots, k-1\}.$$

It is straightforward to construct the proof by writing out and comparing Q-values in $G(n,k)$, as shown in Appendix D.

From Lemma 5, it follows directly that if initialised with the policy 0^n, index-based action selection will go through $0^{n-1}1, 0^{n-1}2, \dots, 0^{n-1}(k-1)$; thereafter $0^{n-2}1(k-1), 0^{n-2}2(k-1), \dots, 0^{n-2}(k-1)^2$, and so on until the optimal policy $(k-1)^n$ is evaluated after $n(k-1)+1$ iterations.

In case random action selection is used, it remains that the policies $0^{n-1}(k-1), 0^{n-2}(k-1)^2, \dots, (k-1)^n$ will be visited, but the number of policies visited in between any successive pair of these will be random, since improving actions are picked uniformly at random. For $i \in \{1,2,\dots,n\}$, $j \in \{0,1,\dots,k-2\}$, let t_{ij} denote the expected number of iterations needed to go from $\pi_{ij} = 0^{i-1}j(k-1)^{n-i}$ to $0^{i-1}(k-1)^{n-i+1}$. Clearly t_{ij} is independent of i, and may be written as t_j. We have $t_{k-2} = 1$ and for $j \in \{0,1,\dots,k-3\}, t_j = 1 + \frac{1}{k-j-1}\sum_{j'=j+1}^{k-2} t_{j'}$. Solving this recurrence yields $t_0 = \theta(\log(k))$; in other words, there are $\theta(\log(k))$ expected iterations corresponding to each improvable state.

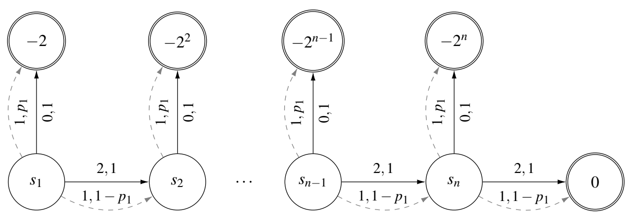

Fig. 2. The stochastic MDP $G(n,3)$, used to illustrate the structure of $G(n,k)$. Labels on arrows mark "action, probability"; terminal states show rewards. While actions 0 and $k-1$ are deterministic, all others are stochastic, with transition probabilities as specified in Section V-A.

Theorem 6 (Generic Lower Bounds): On $G(n,k)$, if initialised with policy 0^n, every PI variant doing index-based action selection takes $\Omega(kn)$ iterations, and every PI variant doing random action selection takes $\Omega(\log(k) \cdot n)$ iterations.

This result is significant for Howard's PI and Random PI, whose current lower bounds are $\theta(n)$ even for $k = 2$.

VI. Simple Policy Iteration

In Section IV, we showed a lower bound of $\Omega(k^{n/2})$ iterations for a new, carefully-designed variant of PI, while in Section V, we provided lower bounds that apply to all PI variants that use index-based or random action-selection. In this section, we investigate the behaviour of Simple PI on multi-action MDPs. Recall that this variant can visit each of the 2^n policies for an n-state, 2-action MDP [8]. Simple PI assumes an arbitrary, fixed indexing of states, and always switches the improvable state with the largest index. We consider index-based and random action selection for $k \geq 3$.

A. Construction

Fig. 3 shows our construction $H(n,k)$, which generalises the one proposed by Melekopoglou and Condon [8]. The MDP has n non-terminal states, $s_1, s_2, \dots, s_n$, and two terminal states. For $i \in \{1,2,\dots,n\}$, each state s_i has a "partner" state s'_i, from which two equiprobable outgoing transitions do not depend on action. In principle these states can be removed and the transition probabilities from $s_1, s_2, \dots, s_n$ modified accordingly. Whereas the original construction for $k = 2$ only gives a reward of -1 on reaching one of the terminal states, our generalisation also associates rewards with state-action pairs.

As in the original construction, one action, say 0, transitions deterministically, with no reward, from each state s_i to state s_{i-1} for $i \in \{2,3,\dots,n\}$, and from s_1 to a terminal state. We design the other actions $j \in \{1,2,\dots,k-1\}$ from each state to transition deterministically to the corresponding partner state; action j gets reward $\varepsilon/2^{k-1-j}$, where $\varepsilon = 2^{-n}$.

[2] If $\rho(s')$ is the reward given on reaching $s' \in S$ in addition to reward $R(s,a)$ given for taking action a from state s, we can use $R'(s,a) = R(s,a) + \sum_{s' \in S} T(s,a,s')\rho(s')$ as an equivalent reward function that complies with our definition in Section I. We use this idea here and in Section VI.

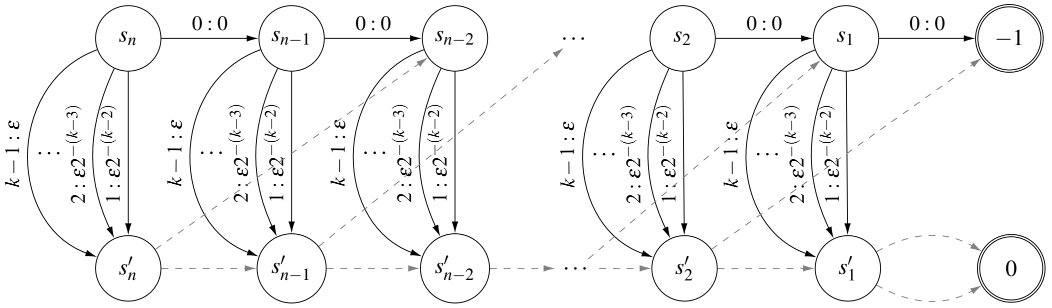

Fig. 3. The stochastic MDP $H(n,k)$. The original construction of Melekopoglou and Condon is obtained by setting $k=2$ and $\varepsilon=0$. Edges are labeled "action: reward". The introduction of $k-2$ new actions and related details are presented in Section VI-A.

B. Lower Bounds

While the original construction uses $\varepsilon=0$, we require the rewards on the actions to be different so that more policies can be visited by PI. Our generalised setup ensures that if s is an improvable state currently taking action $a \in \{0,1,\ldots,k-2\}$, then actions $a+1, a+2, \ldots, k-1$ are all improving actions. Moreover, taking $\varepsilon = 2^{-n}$ retains the structure of the trajectory taken by Simple PI on $H(n,2)$. Indeed for $t \in \{1,2,3,\ldots,2^n\}$, if $\pi_t \in \{0,1\}^n$ is the t-th policy visited on $H(n,2)$, then $\pi'_t \in \{0,k-1\}^n$, which has every occurrence of 1 replaced by $k-1$ in π_t, is the t-th policy from $\{0,k-1\}^n$ visited on $H(n,k)$.

The reason we get scaling of lower bound with k is that corresponding to every switch from action 0 to action 1 on $H(n,2)$, there is a progression through $k-1$ actions—$0,1,\ldots,k-1$—on $H(n,k)$, if using index-based action selection. With random action selection $\theta(\log(k))$ actions are visited in expectation, following the reasoning given in Section V. Since Simple PI makes $\Omega(2^n)$ switches from action 0 to action 1 on $H(n,2)$, we can generalise as below.

Theorem 7 (Simple PI Lower Bounds): On $H(n,k)$, if initialised with policy 0^n, Simple PI takes $\Omega(k \cdot 2^n)$ iterations with index-based action selection, and $\Omega(\log(k) \cdot 2^n)$ iterations in expectation with random action selection.

VII. Conclusion and Future Work

PI [4] is a widely-used family of algorithms for solving MDPs, which model sequential decision making tasks in stochastic domains. While there is a fair amount of work on the theoretical analysis of PI, the literature on lower bounds is relatively sparse. In particular, existing lower bounds on the running-time of PI on n-state, k-action MDPs either assume $k=2$ or take k to be dependent on n. We present the first non-trivial lower bounds for the general case of $k \geq 2$.

We consider a deterministic, index-based action-selection strategy, as well as a randomised one. When coupled with Simple PI [8]—earlier analysed for $k=2$—these strategies increase the corresponding lower bound by factors of k and $\log(k)$, respectively. We also show the same scaling in terms of k for the tightest lower bound known yet for Howard's PI on 2-action MDPs. Indeed the resulting lower bounds of $\Omega(kn)$ and $\Omega(\log(k) \cdot n)$ iterations apply to all PI variants that use index-based and random action-switching, respectively. Our constructions do not yield non-trivial lower bounds when used in conjunction with the popular "max-Q" action selection strategy, which needs further investigation.

From a lower-bounding perspective, the major open question is whether there is an n-state, k-action MDP on which some variant of PI can visit all of the k^n policies. While the answer is affirmative for $k=2$ [8], we are yet unaware what it is for $k \geq 3$. The tightest lower bound we show in this paper is $\Omega(k^{n/2})$ iterations, which is significant in having $\sqrt{k}$, rather than a constant, in the base of the exponent. Future work could explore improvements to our lower bound. Another possibility is to show an upper bound smaller than k^n that simultaneously holds for all PI variants in the case of $k \geq 3$.

Acknowledgements

Shivaram Kalyanakrishnan was partially supported by SERB grant ECR/2017/002479.

References

[1] Richard Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, USA, 1st edition, 1957.

[2] Martin L. Puterman. *Markov Decision Processes*. Wiley, 1994.

[3] Michael L. Littman, Thomas L. Dean, and Leslie Pack Kaelbling. On the complexity of solving Markov decision problems. In *Proc. UAI 1995*, pages 394–402. Morgan Kaufmann, 1995.

[4] R. A. Howard. *Dynamic Programming and Markov Processes*. MIT Press, Cambridge, MA, 1960.

[5] Yishay Mansour and Satinder Singh. On the complexity of policy iteration. In *Proc. UAI 1999*, pages 401–408. Morgan Kaufmann, 1999.

[6] Meet Taraviya and Shivaram Kalyanakrishnan. A tighter analysis of randomised policy iteration. In *Proc. UAI 2019*, page ID 174. AUAI Press, 2019.

[7] Shivaram Kalyanakrishnan, Neeldhara Misra, and Aditya Gopalan. Randomised procedures for initialising and switching actions in policy iteration. In *Proc. AAAI 2016*, pages 3145–3151. AAAI Press, 2016.

[8] Mary Melekopoglou and Anne Condon. On the complexity of the policy improvement algorithm for Markov decision processes. *INFORMS Journal on Computing*, 6:188–192, 1994.

[9] Thomas Dueholm Hansen and Uri Zwick. Lower bounds for Howard's algorithm for finding minimum mean-cost cycles. In *Algorithms and Computation*, pages 415–426. Springer, 2010.

[10] John Fearnley. Exponential lower bounds for policy iteration. In *Proc. ICALP 2010*, pages 551–562. Springer, 2010.

[11] Romain Hollanders, Balázs Gerencsér, and Jean-Charles Delvenne. The complexity of policy iteration is exponential for discounted Markov decision processes. In *Proc. CDC 2012*, pages 5997–6002. IEEE, 2012.

[12] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[13] Sridhar Mahadevan. Average reward reinforcement learning: Foundations, algorithms, and empirical results. *Machine Learning*, 22(1–3):159–195, 1996.

[14] Ingo Schurr and Tibor Szabó. Jumping doesn't help in abstract cubes. In *Integer Programming and Combinatorial Optimization*, pages 225–235. Springer, 2005.

Appendix

A. Trajectory of Peculiar PI on $F(3,3)$

Below we list the sequence of trajectories visited by Peculiar PI (the PI variant from Section IV) on $F(3,3)$, when initialised with policy $0^3 \cdot 0^3$. Each line begins with a "balanced" policy (of the form $x \cdot x$ for $x \in \{0,1,2\}^3$); to its right is the sequence of policies taken to reach the next balanced policy.

000·000	000·001				
001·001	001·002				
002·002	002·012	002·010	000·010		
010·010	010·011				
011·011	011·012				
012·012	012·022	012·020	010·020		
020·020	020·021				
021·021	021·022				
022·022	022·122	022·102	022·100	020·100	000·100
100·100	100·101				
101·101	101·102				
102·102	102·112	102·110	100·110		
110·110	110·111				
111·111	111·112				
112·112	112·122	112·120	110·120		
120·120	120·121				
121·121	121·122				
122·122	122·222	122·202	122·200	120·200	100·200
200·200	200·201				
201·201	201·202				
202·202	202·212	202·210	200·210		
210·210	210·211				
211·211	211·212				
212·212	212·222	212·220	210·220		
220·220	220·221				
221·221	221·222				
222·222					

For $F(m,k)$, the exact number of policies that are visited using our construction is $\frac{2k}{k-1}(k^m-1)-2m+1$, seen here to be 73 for $F(3,3)$.

B. Memoryless Encoding of Peculiar PI

The idea of our construction in Section IV is to proceed from one balanced policy to the next through a sequence of policy improvement steps. Below we provide a memoryless specification of Peculiar PI, the variant we have designed for this purpose. Given an arbitrary policy of the form $x \cdot y$, where $x, y \in A^m$, Peculiar PI identifies the state to switch, denoted $\bar{s} \in S$, as follows.

```
Define $d = [y] - [x]$ and if $d \geq 1$, define $b = \lfloor \log_k(d) \rfloor$.
If $d < 0$: //Cannot arise on $F(m,k)$, starting from $0^m \cdot 0^m$.
    Set $\bar{s}$ to be an arbitrary state.
Else if $d = 0$:
    $\bar{s} \leftarrow s'_{I(x)}$.
Else if $d = 1$:
    $\bar{s} \leftarrow s_m$.
Else if $y_m = k - 1$:
    $\bar{s} \leftarrow s'_{m-b+1}$.
Else:
    $\bar{s} \leftarrow s_{m-b}$.
```

First, note that $\bar{s}$ is *not* guaranteed to be an improvable state on every MDP. In fact, it might not be improvable even for $F(m,k)$ for some policies $x \cdot y$. However, if Peculiar PI is initialised with policy $0^m \cdot 0^m$ on $F(m,k)$, then the procedure outlined here will exactly simulate the trajectory of policies described in the proof of Lemma 3. This property suffices for the purpose of our lower bound. We allow Peculiar PI to be defined arbitrarily when $\bar{s}$ is not an improvable state, or when it does not have the desired choice of improving action (specified next).

If $\bar{s}$ is indeed improvable and $x \cdot y(\bar{s}) \neq k-1$, then Peculiar PI switches the action j for $\bar{s}$ to $j+1$ (if $j+1$ is an improving action). If $\bar{s}$ is improvable and $x \cdot y(\bar{s}) = k-1$, then Peculiar PI switches the action for $\bar{s}$ to 0 (if 0 is an improving action). In summary, if $\bar{s}$ is an improvable state and $(x.y(\bar{s})+1) \mod k$ is an improving action, Peculiar PI switches to this action.

Observe that the procedure outlined above can be implemented using poly(n,k) arithmetic operations and space.

C. Extending Lower Bounds to Discounted Reward Setting

All three of our MDP families—$F(m,k)$ (Section IV), $G(m,k)$ (Section V), and $H(m,k)$ (Section VI)—are defined under the total reward setting. To generalise our lower bounds to the discounted reward setting, we begin by observing that for each MDP family, the following properties are satisfied.

1) For all $\pi : S \to A$, $s \in S$, and $a, a' \in A$:
$$(a \neq a') \implies Q^\pi(s,a) \neq Q^\pi(s,a').$$
2) There is a finite number L such that starting from any state, taking any actions, the number of steps to termination is at most L.
3) Rewards are all bounded; assume they lie in $[-R_{\max}, R_{\max}]$ for finite $R_{max} > 0$.

Define
$$\Delta \overset{\text{def}}{=} \min_{\pi:S\to A, s\in S, a,a'\in A, a\neq a'} |Q^\pi(s,a) - Q^\pi(s,a')|.$$
Since the first property is satisfied, we have $\Delta > 0$. The second property implies that every Q-value may be written as a sum of L (expected) rewards:
$$Q = X_1 + X_2 + X_3 + \cdots + X_L.$$
Now, if we use a discount factor $\gamma \in [0,1]$, we have
$$Q_\gamma = X_1 + \gamma X_2 + \gamma^2 X_3 + \cdots + \gamma^{L-1} X_L.$$
Consequently, we have
$$\begin{aligned}|Q^\pi(s,a) - Q^\pi_\gamma(s,a)| &= |\sum_{i=2}^{L}(1-\gamma^{i-1})X_i| \\ &\leq |\sum_{i=2}^{L}(1-\gamma^{i-1})R_{\max}| \\ &\leq (L-1)(1-\gamma^{L-1})R_{\max}.\end{aligned}$$
For $\gamma > \gamma_0 = \left(\max\{1 - \frac{\Delta}{2(L-1)R_{\max}}, 0\}\right)^{\frac{1}{L-1}}$, we observe that $|Q^\pi(s,a) - Q^\pi_\gamma(s,a)| < \frac{\Delta}{2}$. Hence, for all $\gamma \in (\gamma_0, 1]$, the relative order of Q_γ values is identical for all policies, states, and actions.

The lower bounds we have provided are all for PI variants that are defined solely based on the relative order among Q-values for each state and action. Consequently these algorithms follow the same trajectories for all $\gamma \in (\gamma_0, 1]$.

D. Proof of Lemma 5

Recall that for $i \in \{1,2,\dots,n\}$, $j \in \{0,1,\dots,k-2\}$ we have $\pi_{ij} = 0^{i-1} j (k-1)^{n-i}$. For $u \in \{1,2,\dots,n\}$, we observe

$$V^{\pi_{ij}}(s_u) = \begin{cases} -2^u & \text{if } u < i, \\ -2^i \left(\frac{1}{2} + \frac{k-j}{2k}\right) & \text{if } u = i, \\ 0 & \text{if } u > i. \end{cases}$$

In order to prove that $\mathbf{IS}(\pi_{ij}) = \{i\}$ and $\mathbf{IA}(\pi_{ij}, i) = \{j+1, j+2, \dots, k-1\}$, first we show that $i \in \mathbf{IS}(\pi_{ij})$ and $\{j+1, j+2, \dots, k-1\} \subset \mathbf{IA}(\pi_{ij}, i)$. Observe that for $j' \in \{j+1, j+2, \dots, k-2\}$,

$$\begin{aligned} V^{\pi_{ij}}(s_i) &= -2^i \left(\frac{1}{2} + \frac{k-j}{2k}\right) < -2^i \left(\frac{1}{2} + \frac{k-j'}{2k}\right) \\ &= Q^{\pi_{ij}}(s_i, j'), \end{aligned}$$

and also, $V^{\pi_{ij}}(s_i) < 0 = Q^{\pi_{ij}}(s_i, k-1)$. Hence, $i \in \mathbf{IS}(\pi_{ij})$ and $\{j+1, j+2, \dots, k-1\} \subset \mathbf{IA}(\pi_{ij}, i)$.

Next, we show that $u \notin \mathbf{IS}(\pi_{ij})$ for $u \in \{1,2,\dots,i-1\} \cup \{i+1, i+2, \dots, n\}$ by considering separate cases.

$\underline{u \in \{1,2,\dots,i-2\}.}$ In this case, for $j' \in \{1,2,\dots,k-2\}$,

$$Q^{\pi_{ij}}(s_u, j') = -2^u \left(\frac{1}{2} + \frac{k-j'}{2k}\right) - 2^{u+1} \left(\frac{1}{2} - \frac{k-j'}{2k}\right),$$

and $Q^{\pi_{ij}}(s_u, k-1) = -2^{u+1}$. Thus, for $j' \in \{1,2,\dots,k-1\}$, $Q^{\pi_{ij}}(s_u, j') < -2^u = V^{\pi_{ij}}(s_u)$.

$\underline{u = i-1.}$ In this case, for $j' \in \{1,2,\dots,k-2\}$,

$$Q^{\pi_{ij}}(s_u, j') = -2^u \left(\frac{1}{2} + \frac{k-j'}{2k}\right) + V^{\pi_{ij}}(s_i) \left(\frac{1}{2} - \frac{k-j'}{2k}\right),$$

and $Q^{\pi_{ij}}(s_u, k-1) = V^{\pi_{ij}}(s_i)$. Substituting for $V^{\pi_{ij}}(s_i)$, we get, for $j' \in \{1,2,\dots,k-1\}$, $Q^{\pi_{ij}}(s_u, j') < V^{\pi_{ij}}(s_u)$.

$\underline{u \in \{i+1, i+2, \dots, n\}.}$ In this case for $j' \in \{0,1,\dots,k-2\}$,

$$Q^{\pi_{ij}}(s_u, j') = -2^u \left(\frac{1}{2} + \frac{k-j'}{2k}\right) < 0 = V^{\pi_{ij}}(s_u).$$