View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Georgia Southern University: Digital Commons@Georgia Southern

Georgia Southern University

Digital Commons@Georgia Southern

Mathematical Sciences Faculty Publications

Mathematical Sciences, Department of

5-24-2019

Exploration Using Without-Replacement Sampling of Actions Is Sometimes Inferior

Stephen W. Carden

S. Dalton Walker

Follow this and additional works at: https://digitalcommons.georgiasouthern.edu/math-sci-facpubs

Part of the Mathematics Commons

This article is brought to you for free and open access by the Mathematical Sciences, Department of at Digital Commons@Georgia Southern. It has been accepted for inclusion in Mathematical Sciences Faculty Publications by an authorized administrator of Digital Commons@Georgia Southern. For more information, please contact digitalcommons@georgiasouthern.edu.

machine learning &
knowledge extraction

MDPI

Article

Exploration Using Without-Replacement Sampling of Actions Is Sometimes Inferior

Stephen W. Carden [1,*] **and S. Dalton Walker** [2]

[1] Department of Mathematical Sciences, Georgia Southern University, Statesboro, GA 30460, USA
[2] Air Force Material Command, Robins Air Force Base, GA 31098, USA; s.d.walker6249@gmail.com
* Correspondence: scarden@georgiasouthern.edu; Tel.: +1-912-584-0018

Received: 4 April 2019; Accepted: 13 May 2019; Published: 24 May 2019

Abstract: In many statistical and machine learning applications, without-replacement sampling is considered superior to with-replacement sampling. In some cases, this has been proven, and in others the heuristic is so intuitively attractive that it is taken for granted. In reinforcement learning, many count-based exploration strategies are justified by reliance on the aforementioned heuristic. This paper will detail the non-intuitive discovery that when measuring the goodness of an exploration strategy by the stochastic shortest path to a goal state, there is a class of processes for which an action selection strategy based on without-replacement sampling of actions can be worse than with-replacement sampling. Specifically, the expected time until a specified goal state is first reached can be provably larger under without-replacement sampling. Numerical experiments describe the frequency and severity of this inferiority.

Keywords: count-based exploration; without-replacement sampling; stochastic shortest path; reinforcement learning; Markov decision processes

1. Introduction

The idea that learning is more efficient or estimates will be more precise when new, previously unseen information is processed has a strong intuitive appeal, and is often verified by theory. Consider the following elementary examples likely to be encountered by an undergraduate student. In sampling theory, consider estimating the mean of a collection of N values with variance σ^2 by taking a sample of size $n, n < N$. The variance of the sample mean when sampling with-replacement is $\frac{\sigma^2}{n}$, whereas the variance when sampling without-replacement is $\left(\frac{N-n}{N-1}\right)\frac{\sigma^2}{n}$, which is strictly smaller for $n > 1$. For a probabilistic example, consider an urn with N marbles, exactly one of which is white. When searching for the white marble by drawing from the urn with-replacement, the number of draws is a geometric random variable and has expected value N. When searching by drawing without-replacement, the number of draws is a discrete uniform random variable and has expected value $\frac{N+1}{2}$.

In more complex situations, it is generally difficult to analyze the behavior of without-replacement sampling due to the dependence between chosen items, so the argument in favor of without-replacement sampling or its variants is often made with numerical evidence. Examples include choosing ensemble classifiers [1], constructing low-rank approximations of large matrices [2,3], kernel embeddings [4], computational learning theory [5], and as a general purpose tool in evaluating data analysis applications [6]. A well-studied case is stochastic incremental gradient descent, for which there is a growing body of theoretical analysis [7–10] in favor of random reshuffling, which makes multiple passes through the data set, sampling without-replacement at each epoch.

Consider specifically the subdomain of reinforcement learning, in which an agent must learn how to behave optimally in an unknown Markovian environment [11]. Formally, the problem is cast as a

Markov Decision Process [12], a sequential decision model in which a system can reside in one of a given set of states, S. At discrete points in time called epochs, the agent observes the state and chooses from among a set of actions, A. The system then transitions to the next state according to transition probabilities that depend on the current state and action, and a numeric reward is received. In the most general case, the reward is a random variable with a distribution that depends on the current state and action. The agent collects data by observing the state, choosing an action, and observing the resulting transition and reward. This data is used to construct a function from the state space to the action space, known as a policy $\pi : S \to A$, so that using action $\pi(s)$ when in state s will optimize a given measure of reward.

In order to guarantee that the learned policy converges to the optimal policy, it is required that the number of observations goes to infinity [13]. In the case of discrete states and actions, each state–action pair must be observed an infinite number of times [14]. In the case of continuous states and actions, the distribution of observed states and actions must have a density that is positive everywhere [15]. A practical corollary is that for an algorithm to learn well from a finite amount of data, this data should contain observations that represent the state-action space as completely as possible. Therefore, a key component of a reinforcement learning algorithm is how they explore, that is, how actions are chosen so that the data has enough variety for meaningful learning.

Exploration in reinforcement learning has been, and remains, an active research topic since the inception of the field. Some of the most common strategies include ϵ-greedy [16], in which the agent chooses the best known action with probability $1 - \epsilon$ and a random action with probability ϵ; softmax exploration [17], in which a distribution (usually the Boltzmann distribution) links the probability of selecting an action to its estimated value; “optimism in the face of uncertainty” [18], which assumes that actions with unknown or uncertain rewards are better than known actions, encouraging their use by a greedy agent; and statistical approaches such as maintaining a confidence interval for the value of each state-action pair, and choosing the action with the greatest upper confidence bound [19]. Another class of exploration strategies keeps a history of the actions used in each state. Upon visiting a state, recency-based exploration [20,21] incentivizes the use of actions that have not been recently used from that state, which is especially helpful for continued learning in changing environments. Similarly, count-based exploration [22–24] incentivizes actions that have been used less frequently.

Of particular interest to this paper is a count-based strategy used by Kearns and Singh [25] in their “Explicit Explore or Exploit” (E^3) algorithm. This algorithm is notable for achieving near-optimal performance in a time with a proven polynomial bound. The "Explore" part of the algorithm is a simple yet appealing strategy that the authors call balanced wandering. Under this strategy, a record is kept of how many times each action has been used for each state. When a state is visited, the action is chosen uniformly from among the actions that have been used in that state the least number of times. Note that this is equivalent to the random reshuffling used in without-replacement stochastic incremental gradient descent, but on a per-state basis. Upon subsequent visits to a particular state, actions will be sampled without-replacement until all actions have been tried, at which point sampling begins anew.

There are several reasons to believe that balanced wandering may provably be a uniform improvement over exploration using purely random, with-replacement action selection. First, balanced wandering is a relatively simple exploration strategy which has already been proven amenable to theoretical analysis [25]. Second, it can be viewed as an application of random reshuffling to reinforcement learning, which has been proven to be an improvement for stochastic gradient descent. Third, as argued in the opening paragraph, the intuition that without-replacement sampling yields more and better information than with-replacement sampling is undeniably strong.

The research presented in this paper is the result of an investigation initially intending to prove that balanced wandering is uniformly superior to with-replacement sampling of actions. However, we were surprised to discover that the hypothesis in general is false, and there exists a class of Markov decision processes for which balanced wandering is worse in terms of expected time until a goal state is reached.

The latter half of the project then became a search for understanding how this seeming paradox can occur, and finding a counter-proof to the hypothesis that balanced wandering is always superior.

The remainder of this paper is organized as follows. Section 2 sets up the mathematical preliminaries, describes the two exploration strategies under investigation, defines the metric by which they are compared, and states the hypothesis under investigation. Section 3 proves that for the smallest decision processes, balanced wandering is indeed a strict, uniform improvement over with-replacement action selection. Section 4 finds and proves conditions on transition probabilities that are sufficient for balanced wandering to be worse, and presents an intuitive explanation as to how the paradox may occur. Section 5 contains the result of a numerical experiment to investigate the frequency and magnitude of the paradox. Section 6 concludes with ideas for future research.

2. Preliminaries

Throughout this paper, N will denote the number of states and M the number of actions of a Markov Decision Process (hereafter MDP). Let $S = \{1, 2, \ldots, N\}$ be a finite set of states in which the system may reside. Let state 1 be an initial state the system begins in, and state N a goal state the agent is trying to reach. Let $A = \{1, 2, \ldots, M\}$ be a finite set of actions from which the agent may choose. Associated with each state-action pair $(s, a) \in S \times A$ is a probability distribution on the states, $\Pr(s, \cdot, a) : S \to [0, 1]$, which governs the transition to the next state. For example, $\Pr(i, j, k)$ denotes the probability of transitioning to state j when action k is used from state i. The transition probabilities are assumed to be stationary in time and have the Markov property: they are conditionally independent of past states and actions given the current state and action.

MDPs typically include a reward received at each epoch. This paper focuses on the exploration portion of the learning problem, so rewards can be ignored. If this bothers the reader, we suggest supposing that all states emit a reward of zero except for the goal state N, which emits a positive reward. This would be pertinent to a learning algorithm for a process with a single reward state; no meaningful learning can take place until the first epoch at which a state with a positive reward is encountered, so it is desirable to find a state with positive rewards as quickly as possible.

Consider the following two methods for selecting actions from a finite set:

- Random Action Selection (hereafter denoted RAS): Whenever a state is visited, the agent chooses the action randomly and uniformly from the set of all actions.
- Balanced Wandering (hereafter denoted BW): For each state, a history is kept of the number of times each action has been tried. Whenever a state is visited, the agent inspects the history of actions tried from that state, and creates a subset of the actions that have been tried the least number of times. The agent chooses the action randomly and uniformly from the actions in that subset.

There is an equivalent formulation of BW which will be helpful in a forthcoming analysis. Notice that under BW, for a given state and any positive integer n, each action must be tried n times before any action can be tried $n + 1$ times. Then, as M represents the number of actions, each state is initially assigned a random permutation of the M actions to be executed in the first M visits to that state. After visit M, a new random permutation of the M actions is assigned to the next M visits to that state, and so on.

Define τ_R and τ_B to be the hitting times for the goal state, that is, the first time at which the system reaches state N, under each exploration strategy. That is,

$$\tau_R = \min\{t : \text{state at time } t \text{ is } N,\ \text{RAS is used}\},$$
$$\tau_B = \min\{t : \text{state at time } t \text{ is } N,\ \text{BW is used}\}.$$

In Sections 3 and 4, we assume that state N is accessible from all states, so $P(\tau_R = \infty) = P(\tau_B = \infty) = 0$. Because the state space is finite, standard Markov chain theory implies $\mathbb{E}[\tau_R]$, $\mathbb{E}[\tau_B] < \infty$. In Section 5, we will consider the possibility of an absorbing class of states or *dead ends* [26] from which the goal state cannot be reached.

Hypothesis 1. *For an MDP with an arbitrary number of states N and arbitrary number of actions M, the expected hitting time for the goal state under BW is less than or equal to the expected hitting time for the goal state under RAS. That is,*

$$\mathbb{E}[\tau_B] \leq \mathbb{E}[\tau_R].$$

The following two sections will show that this hypothesis is true for $N = 2$ states, but false in general for $N \geq 3$.

By using expected hitting time to measure the effectiveness of an exploration strategy, we are essentially casting the problem as a special case of a stochastic shortest path (SSP) scenario [27]. An SSP is a type of MDP, often considered as a weighted directed graph, with the goal of selecting actions so that the journey from an initial state to a target state has the minimum possible expected sum of weights for traversed edges. Because we are seeking the strategy with expected minimum hitting time, this corresponds to an SSP where each edge is the transition time between states, so all edges have a weight of one.

3. A Proof for $N = 2$

Consider the simple MDP which consists of a finite state space $S = \{1, 2\}$ and a finite action space $A = \{1, 2, \ldots, M\}$. Let state 1 be the initial state of the system, and state 2 be the goal state. Then, τ_R is the random variable denoting the number of epochs until the first time the system transitions to state 2. Similarly, τ_B denotes the number of epochs until first reaching state 2 under BW.

Before stating and proving the theorem, we review some established definitions and results that will be needed in the proof.

Definition 1. *Let $x_1, \ldots, x_n$ be real numbers. For integer $k \leq n$, the* ***elementary symmetric polynomial of degree*** *k in $x_1, \ldots, x_n$, denoted $e_k(x_1, \ldots, x_n)$, is the sum of all products of k distinct elements from $x_1, \ldots, x_n$. That is,*

$$\begin{aligned}
e_0(x_1, \ldots, x_n) &:= 1, \\
e_1(x_1, \ldots, x_n) &:= x_1 + x_2 + \ldots + x_n, \\
&\vdots \\
e_k(x_1, \ldots, x_n) &:= \sum_{1 \leq i_1 < i_2 < \cdots < i_k \leq n} x_{i_1} x_{i_2} \cdots x_{i_k} = \sum_{\substack{I \subset \{1, \ldots, n\} \\ |I| = k}} \prod_{i \in I} x_i, \\
&\vdots \\
e_n(x_1, \ldots, x_n) &:= \prod_{i=1}^{n} x_i,
\end{aligned}$$

for $1 \leq k \leq n$. It is the sum of $\binom{n}{k}$ terms, where each term is the product of an unordered sample without-replacement of size k. The polynomials corresponding to $k = 1$ and $k = n$ are the sum of the terms and the product of the terms respectively, $e_1 = \sum x_i$ and $e_n = \prod x_i$.

Definition 2. *The* ***elementary symmetric mean of degree*** *k of $x_1, \ldots, x_n$, denoted $E_k(x_1, \ldots, x_n)$, is the mean of the terms in the elementary symmetric polynomial of the same degree.*

$$E_k(x_1, \ldots, x_n) := \frac{e_k(x_1, \ldots, x_n)}{\binom{n}{k}}.$$

Notice that $E_1(x_1, \ldots, x_n)$ is the arithmetic mean, and $E_n(x_1, \ldots, x_n)$ is the geometric mean raised to the power n.

Theorem 1 (Maclaurin Inequality). *For $x_1, \ldots, x_n > 0$,*

$$E_1(x_1, \ldots, x_n) \geq \sqrt{E_2(x_1, \ldots, x_n)} \geq \cdots \sqrt[k]{E_k(x_1, \ldots, x_n)} \cdots \geq \sqrt[n]{E_n(x_1, \ldots, x_n)}.$$

Furthermore, the inequalities are strict unless $x_1 = x_2 = \ldots = x_n$.

See Biler [28] for a proof. Notice that this is a refinement of the well-known inequality between the arithmetic mean and the geometric mean, which can be recovered by comparing the first and last terms in the sequence of inequalities. Theorem 2 can now be proved.

Theorem 2. *For a two-state MDP,*

$$\mathbb{E}[\tau_B] \leq \mathbb{E}[\tau_R].$$

This inequality is strict unless every action has an equal probability of transitioning to state 2.

Proof. The first step is recognizing that tail probabilities for the hitting times can be expressed as elementary symmetric means of the transition probabilities under each action. For compactness of notation, denote the probability of transitioning from state 1 to state 2 using action m by $p_m := \Pr(1, 2, m)$, and the probability of remaining in state 1 under action m by $q_m = 1 - p_m$. First, consider RAS. Because every action has probability $\frac{1}{M}$ of being chosen, the overall probability of transitioning from state 1 to state 2 is given by the Law of Total Probability as

$$\overline{p} := \sum_{i=1}^{M} p_i \frac{1}{M},$$

which is simply the average of the probabilities of transitioning from state 1 to state 2 under each action. Then τ_R, the number of epochs until the first successful transition under random action selection, is a geometric random variable with parameter $\overline{p}$, and has tail probabilities

$$\Pr(\tau_R > k) = (1 - \overline{p})^k = \overline{q}^k = E_1(q_1, \ldots, q_M)^k.$$

Now consider tail probabilities under BW. Recall that this action selection scheme is equivalent to assigning a permutation of the M actions to each state to be executed sequentially. For a value $k \leq M$, the event $\{\tau_B > k\}$ means that the first k actions that were tried did not result in a successful transition to state 2. Partitioning this event according to the first k actions in the permutation and applying the Law of Total Probability shows that this probability can also be expressed in terms of elementary symmetric means. Let C_k denote the set of all combinations of k integers out of the first M integers, representing the first k actions used. Each combination of k actions is equally likely, so the probability of each set in the partition is $1/\binom{M}{k}$. Formally,

$$\Pr(\tau_B > k) = \sum_{c \in C_k} \Pr(\tau_B > k | c) P(c) = \sum_{c \in C_k} \left(\prod_{i \in c} q_i \right) \frac{1}{\binom{M}{k}} = E_k(q_1, \ldots, q_M).$$

A direct application of the Maclaurin Inequality shows that, for $k \leq M$,

$$\Pr(\tau_R > k) = E_1(q_1, \ldots, q_M)^k \geq \left(E_k(q_1, \ldots, q_M)^{1/k} \right)^k = E_k(q_1, \ldots, q_M) = \Pr(\tau_B > k).$$

The next step is to use this inequality to obtain an inequality between conditional expectations for the hitting times.

$$
\begin{aligned}
\mathbb{E}[\tau_B \mid \tau_B \leq M] &= \sum_{k=1}^{M} k \Pr(\tau_B = k \mid \tau_B \leq M) = \sum_{k=1}^{M} k \frac{\Pr(\tau_B = k)}{\Pr(\tau_B \leq M)} \\
&= \frac{1}{\Pr(\tau_B \leq M)} \sum_{k=1}^{M} k \Pr(\tau_B = k) = \frac{1}{\Pr(\tau_B \leq M)} \sum_{k=1}^{M} \Pr(\tau_B > k) \\
&\leq \frac{1}{\Pr(\tau_R \leq M)} \sum_{k=1}^{M} \Pr(\tau_R > k) = \mathbb{E}[\tau_R \mid \tau_R \leq M].
\end{aligned}
$$

For the conditional expectation $\mathbb{E}[\tau_B \mid \tau_B > M]$, notice that after epoch M every action has been tried exactly once, and the state receives a new permutation of the M actions to execute in the following M time steps. Hence, if the system has not transitioned to state 2 by M time steps, then the process probabilistically restarts. For RAS, this is actually true at every time step, but under BW this only occurs at multiples of M. This yields the expressions

$$
\begin{aligned}
\mathbb{E}[\tau_B \mid \tau_B > M] &= M + \mathbb{E}[\tau_B], \\
\mathbb{E}[\tau_R \mid \tau_R > M] &= M + \mathbb{E}[\tau_R].
\end{aligned}
$$

Finally, consider the unconditional expectation under each action selection scheme.

$$
\begin{aligned}
\mathbb{E}[\tau_B] =& \mathbb{E}[\tau_B \mid \tau_B \leq M] \Pr(\tau_B \leq M) + \mathbb{E}[\tau_B \mid \tau_B > M] \Pr(\tau_B \leq M) \\
=& \mathbb{E}[\tau_B \mid \tau_B \leq M] \Pr(\tau_B \leq M) + (M + \mathbb{E}[\tau_B]) \Pr(\tau_B > M).
\end{aligned}
$$

Solve this expression for $\mathbb{E}[\tau_B]$.

$$
\mathbb{E}[\tau_B] = \mathbb{E}[\tau_B \mid \tau_B \leq M] - M + \frac{M}{\Pr(\tau_B \leq M)}.
$$

The same logic is used to show

$$
\mathbb{E}[\tau_R] = \mathbb{E}[\tau_R \mid \tau_R \leq M] - M + \frac{M}{\Pr(\tau_R \leq M)}.
$$

A term-by-term comparison shows that $\mathbb{E}[\tau_B] \leq \mathbb{E}[\tau_R]$, which completes the proof. □

4. A Counter-Proof for $N \geq 3$

The argument used in the preceding proof does not extend to $N \geq 3$ states. After several other proof strategies also failed, we began to wonder if the hypothesis is true in general. This section states and proves conditions that show analytically Hypothesis 1 is false in general. Because there are now an arbitrary number of states, we return to the general notation for transition probabilities, in which $\Pr(i, j, k)$ represents the probability of transitioning to state j when action k is used from state i. First, a quantity is defined which will be useful in the proof.

Definition 3. *Let $m \in A$. Define $\overline{p}_{-m}$ to be the average probability of transitioning from the initial state to the goal state if action m is excluded. That is,*

$$
\overline{p}_{-m} := \frac{\sum_{k=1, k \neq m}^{M} \Pr(1, N, k)}{M-1}.
$$

Theorem 3. *Consider an MDP with $N \geq 3$ states and an arbitrary number of actions M.*

1. *If there exists $c \in (0,1)$ such that*

$$\Pr(n, N, m) \geq c \textit{ for all } n \in S, m \in A \tag{1}$$

and

$$\frac{1}{M^2} \sum_{m=1}^{M} \Pr(1,1,m) \left(\Pr(1,N,m) - \overline{p}_{-m}\right) > \frac{(1-c)^3}{c}, \tag{2}$$

then $\mathbb{E}[\tau_B] > \mathbb{E}[\tau_R]$.

2. *Furthermore, there exist values of c in a neighborhood of one such that an MDP satisfying (1) and (2) can be constructed by choosing*

$$\begin{aligned} \Pr(1,1,1) &= \frac{1-c}{2}, & \Pr(1,1,m) &= 0 \textit{ for } m = 2, \dots M, \\ \Pr(1,N,1) &= \frac{1+c}{2}, & \Pr(n,N,m) &= c \textit{ for } n \neq 1 \textit{ or } m \neq 1. \end{aligned} \tag{3}$$

Before starting the formal proof, see that Equation (2) has an interpretation that helps with an intuitive understanding of how the paradox may happen. BW is inferior relative to RAS when the left side is large, which happens when the actions having the greatest probability to transition directly from the initial state to the goal state also are likely to remain in the initial state. Therefore, if the use of one of these actions results in the system remaining in the initial state, it would be desirable to use the same action again. Under RAS, it is possible to use the same action from the initial state multiple times in a row, but BW forces the use of other, inferior actions. When this effect is large enough, $\mathbb{E}[\tau_B] > \mathbb{E}[\tau_R]$.

Proof. A well-known fact from probability theory states that the expected value of a non-negative integer-valued random variable X can be found as a sum over tail probabilities, $\mathbb{E}[X] = \sum_{k=0}^{\infty} \Pr(X > k)$ ([29], p.3). Applying this to the hitting times under RAS and BW, we have

$$\mathbb{E}[\tau_B] - \mathbb{E}[\tau_R] = \sum_{k=0}^{\infty} \Pr(\tau_B > k) - \sum_{k=0}^{\infty} \Pr(\tau_R > k) = \sum_{k=2}^{\infty} \left(\Pr(\tau_B > k) - \Pr(\tau_R > k)\right).$$

Notice that the first two differences in the sum are zero and have been dropped. For $k = 0$, both tail probabilities are one. For $k = 1$, using the rule of complements, $\Pr(\tau_B > 1) - \Pr(\tau_R > 1) = \Pr(\tau_R = 1) - \Pr(\tau_B = 1) = 0$ because all actions are available, thus the probabilities of transitioning directly to the goal are equivalent. Then, moving the $k = 2$ term out of the sum,

$$\begin{aligned} &\mathbb{E}[\tau_B] > \mathbb{E}[\tau_R] \\ \iff \; &\left(\Pr(\tau_B > 2) - \Pr(\tau_R > 2)\right) > \sum_{k=3}^{\infty} \left(\Pr(\tau_R > k) - \Pr(\tau_B > k)\right). \end{aligned} \tag{4}$$

The strategy will be to choose transition probabilities so that the left side is positive and large, and the right side has a small absolute value. Consider first the left side. Elementary but tedious calculations, which are deferred to the Appendix A, show that

$$\Pr(\tau_B > 2) - \Pr(\tau_R > 2) = \frac{1}{M^2} \sum_{m=1}^{M} \Pr(1,1,m) \left(\Pr(1,N,m) - \overline{p}_{-m}\right). \tag{5}$$

Now consider the right side. Assumption (1) says that no matter what the current state and action are, there is always a probability of at least c of transitioning directly to the goal. Therefore, $\Pr(\tau_R > k)$ and $\Pr(\tau_B > k)$ are each strictly less than $(1-c)^k$, as is the absolute value of their difference. Using this with well-known formulas for sums and partial sums of geometric series, we obtain

$$
\begin{aligned}
&\sum_{k=3}^{\infty} \left(\Pr(\tau_R > k) - \Pr(\tau_B > k)\right) \leq \sum_{k=3}^{\infty} |\Pr(\tau_R > k) - \Pr(\tau_B > k)| \\
&< \sum_{k=3}^{\infty} (1-c)^k = \sum_{k=0}^{\infty} (1-c)^k - \sum_{k=0}^{2} (1-c)^k \\
=& \frac{1}{1-(1-c)} - \frac{1-(1-c)^3}{1-(1-c)} = \frac{(1-c)^3}{c}.
\end{aligned}
$$

Combining the expressions for the left and right sides, we see that Equation (2) is sufficient for $\mathbb{E}[\tau_B] > \mathbb{E}[\tau_R]$. It still remains to show existence; that there are probabilities and a value of c satisfying Equations (1) and (2). Now consider the MDP with probabilities defined in Equation (3). Clearly, these probabilities satisfy Equation (1). The left side of Equation (2) only has one non-zero term, so combining with $\overline{p}_{-1} = c$, it can be simplified as follows:

$$
\frac{1}{M^2} \sum_{m=1}^{M} \Pr(1,1,m) \left(\Pr(1,N,m) - \overline{p}_{-m}\right) = \frac{1}{M^2}(1/2 - c/2)(1/2 + c/2 - c) = \frac{1}{4M^2}(1-c)^2.
$$

It remains to show there is a value $c \in (0,1)$ such that

$$
\begin{aligned}
&\frac{1}{4M^2}(1-c)^2 > \frac{(1-c)^3}{c} \\
\iff &\frac{1}{4M^2}(c - 2c^2 + c^3) - 1 + 3c - 3c^2 + c^3 > 0.
\end{aligned}
$$

Call the left hand side $f(c)$. Note that $f(1) = 0$. Find first and second derivatives at $c = 1$:

$$
\begin{aligned}
f'(c) &= \frac{1}{4M^2}(1 - 4c + 3c^2) + 3 - 6c + 3c^2, \\
f'(1) &= 0, \\
f''(c) &= \frac{1}{4M^2}(-4 + 6c) - 6 + 6c, \\
f''(1) &= \frac{1}{2M^2} > 0.
\end{aligned}
$$

A continuous function with $f(1) = 0, f'(1) = 0$ and $f''(1) > 0$ is necessarily positive for some value $c < 1$. Figure 1 illustrates this by plotting $f(c)$ in a neighborhood of one when $M = 2$. This completes the proof. □

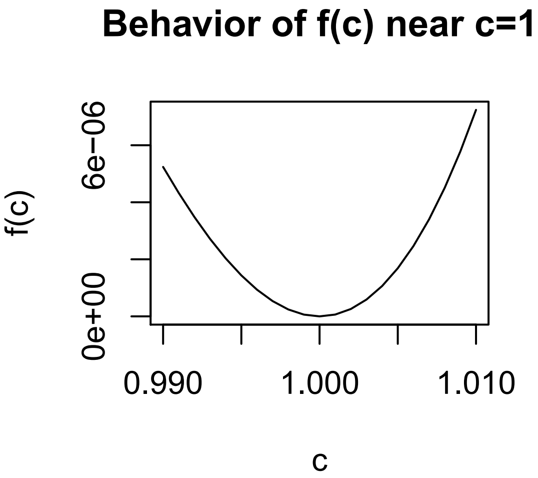


Figure 1. The behavior of $f(c)$ near $c = 1$ for $M = 2$. $\mathbb{E}[\tau_B] > \mathbb{E}[\tau_R]$ when $f(c) > 0$, for which a neighborhood exists for values of c just below 1.

5. Numerical Experiments

Now that Hypothesis 1 is known to be false in general, several questions are immediately raised.

1. How often is $\mathbb{E}[\tau_B] > \mathbb{E}[\tau_R]$? If an MDP is randomly generated, what is the probability BW will be worse than RAS for reaching the goal state?
2. How much worse can BW be than RAS? Does this depend on N and M?
3. For a randomly generated MDP, the difference $\mathbb{E}[\tau_B] - \mathbb{E}[\tau_R]$ can be regarded as a random variable. What are the characteristics of the density of the difference?
4. If we allow the existence of dead ends so that the goal state is not accessible from all states, can Hypothesis 1 still be violated?

This section explains how we investigate answers to these questions via numeric calculation of $\mathbb{E}[\tau_B]$ and $\mathbb{E}[\tau_R]$ for randomly generated MDPs.

First, we need a procedure for constructing a random MDP, which essentially reduces to repeatedly generating discrete distributions where the probabilities are themselves random. This is accomplished by generating independent uniform random variables, and standardizing so that they sum to one. Algorithm 1 gives pseudocode for this procedure, which the reader can find implemented in the language R [30] in the Supplementary Material as the function `makeProbArray()`.

Algorithm 1: Pseudocode for randomly generating the probabilities for an MDP with a specified number of states and actions.

```
Input  : $N$, the number of states; $M$, the number of actions.
Output : $\Pr(\cdot,\cdot,\cdot) : S \times S \times A \to [0,1]$, a randomly generated probability transition structure
         for an MDP.
for $i \leftarrow 1$ to $N$ do
    for $j \leftarrow 1$ to $M$ do
        Generate $N$ independent observations from a Uniform(0,1) distribution, $x_1, \ldots, x_N$.
        for $k \leftarrow 1$ to $N$ do
            $\Pr(i,k,j) = \frac{x_k}{\sum_{n=1}^{N} x_n}$
        end
    end
end
```

Once the transition probabilities are generated, finding $\mathbb{E}[\tau_R]$ is straightforward. Under RAS, the system is an ordinary Markov chain with transition probabilities

$$p_{ij} = \frac{\sum_{k=1}^{M} \Pr(i, j, k)}{M}.$$

Finding the expected hitting time for a specified state in a Markov chain is a standard technique (see, for example, Section 2.11 of Resnick [29]) and so is stated without derivation. Force the goal state N to be absorbing by setting $p_{NN} = 1$ and $p_{Ni} = 0$ for $i = 1, \ldots, N-1$. Let Q be the matrix containing transition probabilities between transient states, I the identity matrix of the same size as Q, and $\vec{1}$ a column vector of all ones with the same number of rows as Q. Then,

$$(I - Q)^{-1}\vec{1} \tag{6}$$

is a vector with entry k containing the expected time until absorption starting from state k. Therefore, $\mathbb{E}[\tau_R]$ is found as the first entry. In the Supplementary Material, this is implemented in the function `expectedHittingTimeRAS()`

Finding $\mathbb{E}[\tau_B]$ is less straightforward. The process under BW is not a Markov chain because transition probabilities depend on which actions are available from which state, which depends on actions used in the past. However, the Markov property can be recovered by including enough information about the history of used actions in the state space. We call the resulting process the *induced* MDP, in contrast with the *original* MDP, and will now detail its construction.

Definition 4. *Let N and M be the number of states and actions respectively in the original MDP. A* ***history matrix*** *H is an $N \times M$ binary matrix with the restriction that a row cannot consist entirely of ones. The set of all history matrices is denoted by $\mathbb{H}$.*

An entry of one in row i and column j of H indicates that action j has already been used in the current permutation of actions for state i, therefore action j is currently unavailable from state i. Likewise, an entry of zero in row i and column j means action j is available from state i. The restriction that a row cannot consist entirely of ones corresponds to BW assigning a new permutation of actions once the current pass has ended, meaning the row would consist entirely of zeros.

For example, for the following history matrix, if the system is in state 1, then action 2 must be used, at which point the first row reverts to all zeros. If the system is in state 2, action 1 must be used, and the second row resets. If the system is in state 3, actions 1 and 2 are eligible, so each action has a 50% chance of being selected. The entry in the third row for the chosen action would change to 1.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

The state space for the induced process is $S' = S \times \mathbb{H}$; that is, for the Markov property to hold, one must know both the state from the original process and information about the actions used from each state. The number of states in S' is much larger than the number of states in S. There are $2^M - 1$ possibilities for each row of the history matrix, and there are N rows, so there are $(2^M - 1)^N$ possible history matrices. Finally a state in S' must also include one of the N states from S, so there are $N(2^M - 1)^N$ states total in S'.

Now we define transitions between states in the induced process. Let $i, j \in S$, and $H_k, H_l \in \mathbb{H}$, so that $(i, H_k), (j, H_l) \in S'$. First recognize that it will not be possible to transition from (i, H_k) to (j, H_l) in one epoch unless the history matrix for the second state reflects the most recently used action but is otherwise identical to the history matrix from the first state. This motivates the following definition.

Definition 5. *Let $(i, H_k), (j, H_l) \in S'$, and $m \in A$. We say that H_l is* ***m-compatible*** *with (i, H_k) if H_k has an entry of zero in row i and column m, and changing that entry to a one results in H_l.*

Intuitively, m-compatibility means that action m is available to be selected from the current state, and that H_l is the resulting history matrix when action m is used (remembering the possibility that if this causes row i to consist entirely of ones, it resets to a row entirely of zeros).

Let $n_{(i,H_k)}$ be the number of entries in row i of H_k that are equal to zero. That is, $n_{(i,H_k)}$ is the number of actions available to choose from when in state i. Then, transition probabilities in the induced process are defined by

$$p_{(i,H_k)(j,H_l)} = \begin{cases} \frac{\Pr(i,j,m)}{n_{(i,H_k)}} & \text{if } H_l \text{ is } m\text{-compatible with } (i, H_k) \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Equation (7) can be understood in three parts. First, the requirement that induced states are compatible ensures that the history matrix is properly updated. Second, the denominator accounts for the number of actions available to choose from. Third, the numerator accounts for the transition probabilities from the original MDP. In the Supplementary Materials, the construction of the transition matrix in the induced MDP is implemented in the function `makeInducedChain()`. Once the transition matrix is obtained, the submatrix corresponding to transitions between transient states can be extracted, and finally $\mathbb{E}[\tau_B]$ can be obtained from the calculation in Equation (6). These final steps are implemented in the function `expectedHittingTimeBW()`.

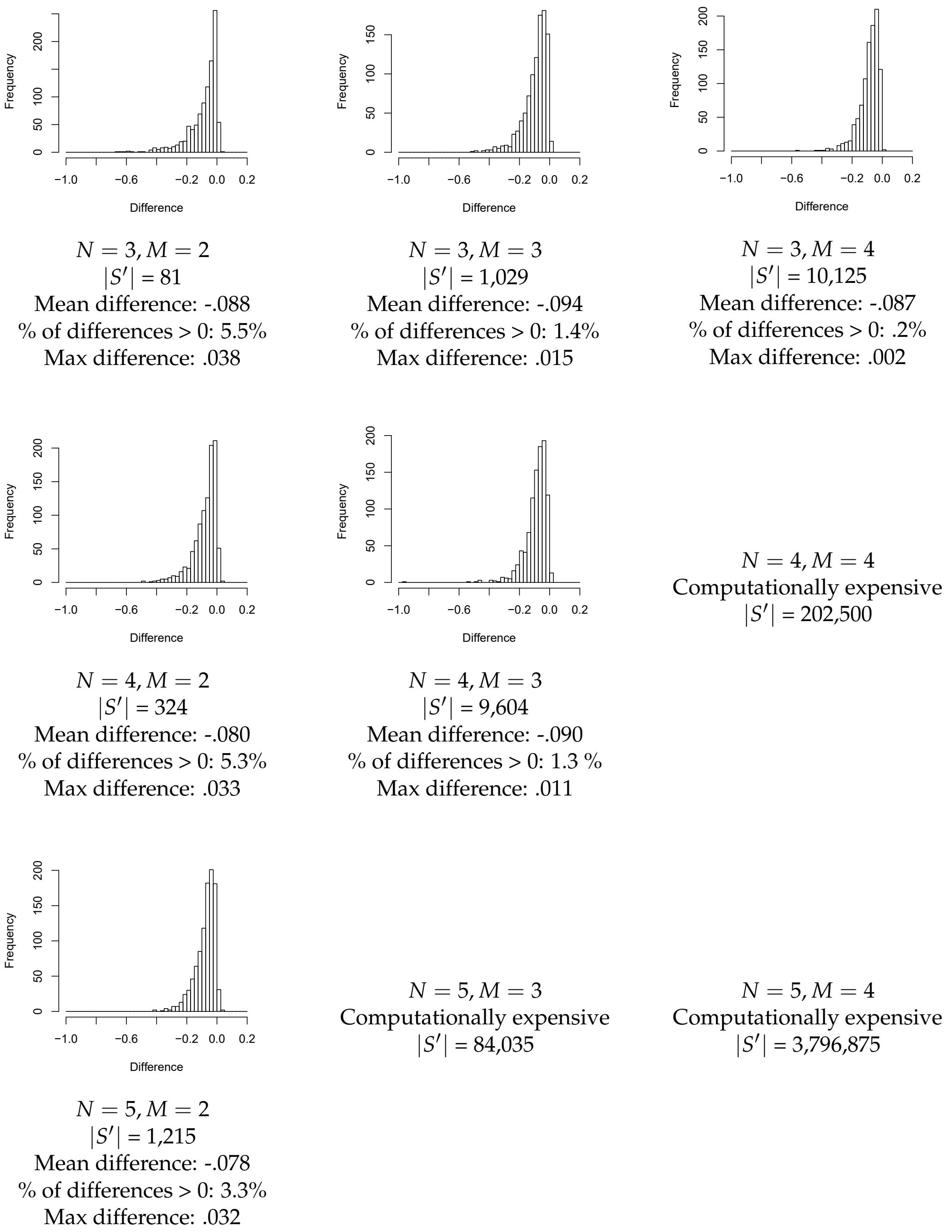
The results of the numerical investigation are presented in Figure 2. For each combination of values of N and M, Algorithm 1 was used to generate 1000 MDPs. Notice that because the size of the induced state space S' grows so rapidly with N and M, results are only given for $N + M \leq 7$. For $N + M > 7$, our hardware could not complete the experiment in a reasonable amount of time.

For each MDP, the difference $\mathbb{E}[\tau_B] - \mathbb{E}[\tau_R]$ was calculated. The figure displays a histogram of the differences, which all share commonalities. The peak is consistently slightly less than zero, indicating that typically RAS is slightly better than BW. The distribution has a left skew, so there are MDPs for which RAS is significantly better. Finally, very little of the distribution is to the right of zero, so while Hypothesis 1 can be false, it is rare and BW is not worse by a large degree.

In addition to the histograms, Figure 2 gives the mean difference, the percentage of differences that are greater than zero (signifying how often Hypothesis 1 is false), and the maximum observed distance, giving a sense of how strongly Hypothesis 1 can be violated. The mean difference is consistently close to $-.085$, and deviations are small relative to the sampling error, so there is little evidence that the mean difference changes substantially with N and M. On the other hand, the percentage of positive differences and the maximum difference appear to decline slightly as N, the number of states increases, and decline sharply as M, the number of actions increases. Therefore, it seems that the frequency and magnitude of violations of Hypothesis 1 decrease as the MDP becomes more complex.

Until now, we have assumed that no matter what state the system is in, the goal state is accessible. That is, there is always a sequence of states and actions with a positive probability of reaching the goal. In many real life applications, the system may enter a dead end [26], which is a state from which it is impossible to reach the goal. If such states are allowed, then the SSP implicitly becomes a multi-objective optimization problem [31], as the agent must simultaneously try to avoid the dead ends while still minimizing the expected cost of reaching the goal.

In this investigation, we handle dead ends by allowing the system to reset to the initial state, but with a penalty of additional time to do so. Formally, this is implemented by adding a *return state*, R. If the system enters a dead end, it will transition to the return state with probability one, and then transition to the initial state with probability one, at which point the search for the goal state begins anew.



$N = 3, M = 2$
$|S'|$ = 81
Mean difference: -.088
% of differences > 0: 5.5%
Max difference: .038

$N = 3, M = 3$
$|S'|$ = 1,029
Mean difference: -.094
% of differences > 0: 1.4%
Max difference: .015

$N = 3, M = 4$
$|S'|$ = 10,125
Mean difference: -.087
% of differences > 0: .2%
Max difference: .002

$N = 4, M = 2$
$|S'|$ = 324
Mean difference: -.080
% of differences > 0: 5.3%
Max difference: .033

$N = 4, M = 3$
$|S'|$ = 9,604
Mean difference: -.090
% of differences > 0: 1.3 %
Max difference: .011

$N = 4, M = 4$
Computationally expensive
$|S'|$ = 202,500

$N = 5, M = 2$
$|S'|$ = 1,215
Mean difference: -.078
% of differences > 0: 3.3%
Max difference: .032

$N = 5, M = 3$
Computationally expensive
$|S'|$ = 84,035

$N = 5, M = 4$
Computationally expensive
$|S'|$ = 3,796,875

Figure 2. Results of numerical experiments calculating $\mathbb{E}[\tau_B] - \mathbb{E}[\tau_R]$ for randomly generated Markov Decision Processes (MDPs). In total, 1000 MDPs were generated for each combination of N and M.

This modification can be observed in the following example. The matrix on the left represents the transition probabilities under an arbitrary action, with state 2 being absorbing. If state 2 is absorbing for all actions, then state 2 is a dead end. On the right, the matrix has been modified so that state 2 leads to the return state R.

$$\begin{array}{c|cccc|} & 1 & 2 & 3 & 4 \\ 1 & .62 & 0 & .16 & .22 \\ 2 & 0 & 1 & 0 & 0 \\ 3 & 0 & .02 & .29 & .69 \\ 4 & 0 & 0 & 0 & 1 \end{array} \rightarrow \begin{array}{c|ccccc|} & 1 & 2 & 3 & R & 4 \\ 1 & .62 & 0 & .16 & 0 & .22 \\ 2 & 0 & 0 & 0 & 1 & 0 \\ 3 & 0 & .02 & .29 & 0 & .69 \\ R & 1 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 1 & 0 \end{array}$$

As mentioned previously, the state space for the induced process under BW grows rapidly, and exact calculation of the expected hitting time rapidly becomes infeasible. This is even more so when dead ends are allowed, as the additional return state increases the size of the induced process. For this reason, we could not replicate the entire experiment shown in Figure 2 allowing dead ends, but we did discover that processes still exist for which BW is worse than RAS. Figure 3 gives an example of such a process, and code for reconstructing it is in the Supplementary Materials.

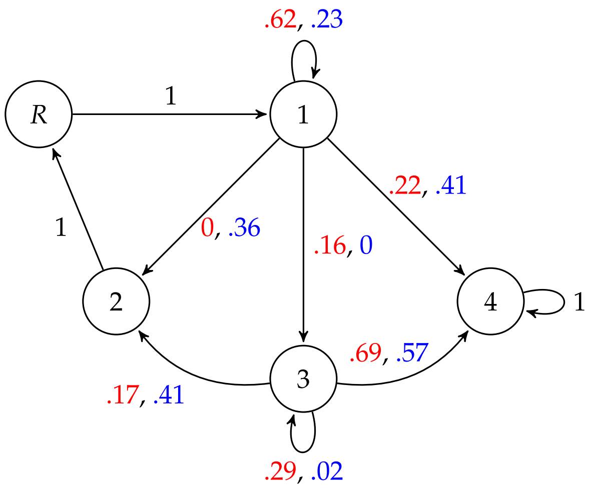


Figure 3. An example of an Markov Decision Process (MDP) with a dead end for which $\mathbb{E}[\tau_B] > \mathbb{E}[\tau_R]$. The first and second values on each edge, in red and blue, are the transition probabilities under actions 1 and 2 respectively. Edges with a single weight, in black, are irrespective of actions. Note that state 2 is a dead end, and returns to the initial state after passing through the return state. For this example, $\mathbb{E}[\tau_B] = 4.014$ and $\mathbb{E}[\tau_R] = 3.991$.

6. Conclusions

This paper has compared two strategies for exploring Markov decision processes with the goal of reaching a specified state in the shortest average number of epochs. Though intuition suggests that the strategy using without-replacement sampling of actions would be uniformly superior, we have shown that this is only true for processes with two states.

As with most non-intuitive discoveries, more questions are raised than answered, and there are many avenues for possible future research. Some of these questions are addressed here.

How can behavior under BW be efficiently investigated when the number of states and actions is large? The state space of the induced process grows rapidly, and it soon becomes impractical to use the standard method for calculating the expected hitting time which requires inversion of the transition matrix. We have also attempted to estimate the hitting time statistically by repeated simulation, but the sampling variance is large relative to the difference $\mathbb{E}[\tau_B] - \mathbb{E}[\tau_R]$, making it difficult to have confidence in any inferences. Further investigation will require the application or development of techniques beyond those used in this paper.

Though it is difficult to analyze large processes exactly, we can make a conjecture. From the numerical experiments, it seems that the difference between RAS and BW (in terms of the percentage of times BW is worse and the maximum degree by which RAS is better) decreases as the number of actions increases. An insight gained from Equation (2) gives reason to believe that this may be true. The term p_{-m}, the probability of transitioning when action m is unavailable, is key. However, as the number of actions increases, p_{-m} will necessarily differ little for each value of m. This corresponds to the notion that sampling with and without replacement become more similar as the number of elements that can be sampled increases. Therefore, we conjecture that the phenomena studied in this paper will be less relevant for larger processes.

Is the use of count-based exploration practically harmful in any state-of-the-art applications? This has not yet been investigated. For future research, the authors intend to implement each of RAS and BW as exploration strategies in benchmark problems and see if the behavior learned is significantly different.

Are there other measures beyond expected hitting time for which BW can be proven strictly better? By using the expected hitting time criteria for a single goal state, we are admittedly limiting ourselves to the study of stochastic shortest path MDPs. Perhaps a more complete metric would be the performance of an agent trained on the data collected by each exploration strategy. This type of analysis is deferred for future research.

In Theorem 3, Equation (1) explicitly requires the goal be accessible from all states, so the same proof strategy cannot be extended to processes with dead ends. Computations also become more difficult with dead ends. At this point, we know that Hypothesis 1 can still be violated when dead ends exist, but nothing else is known for certain. Dead ends provide another avenue for future research.

Theorem 3 is essentially an existence proof which finds sufficient conditions for Hypothesis 1 to be false; can it be extended to finding necessary conditions as well, providing a complete characterization? There is still much work to be done before a complete understanding of optimal exploration in Markovian environments is reached.

Supplementary Materials: The following are available online at http://www.mdpi.com/2504-4990/1/2/41/s1, Script S1: A script in the language R containing functions for implementing the numeric experiments.

Author Contributions: Conceptualization, S.W.C.; investigation, S.W.C. and S.D.W.; writing—original draft preparation, S.W.C.; writing—review and editing, S.W.C. and S.D.W.

Funding: This research received no external funding.

Acknowledgments: The authors are grateful to the anonymous reviewers for their suggestions regarding stochastic shortest paths and dead ends.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MDP	Markov Decision Process
SSP	Stochastic Shortest Path
RAS	Random Action Selection
BW	Balanced Wandering

Appendix A. Proof of Equation (5)

A hitting time greater than two means that the goal state was not reached in the first two transitions. Consider all possible combinations of states and actions that do not reach the goal in the first two epochs. There are three categories:

1. Paths that transition away from the initial state at the first epoch.
2. Paths that stay in the initial state, but use a different action on the second epoch.

3. Paths that stay in the initial state, but use the same action on the first two epochs.

Under RAS, any path is possible, with the probability of a path being the product of the probability of choosing the actions (which is always $1/M^2$) and the transition probabilities between states. Thus partitioning according to the categories described above,

$$\begin{aligned} P(\tau_R > 2) = & \frac{1}{M^2} \sum_{m_1=1}^{M} \sum_{m_2=1}^{M} \sum_{n_1=2}^{N-1} \sum_{n_2=1}^{N-1} P(1, n_1, m_1) P(n_1, n_2, m_2) \\ & + \frac{1}{M^2} \sum_{m_1=1}^{M} \sum_{\substack{m_2=1 \\ m_2 \neq m_1}}^{M} \sum_{n_2=1}^{N-1} P(1, 1, m_1) P(1, n_2, m_2) \\ & + \frac{1}{M^2} \sum_{m_1=1}^{M} \sum_{n_2=1}^{N-1} P(1, 1, m_1) P(1, n_2, m_1). \end{aligned}$$

Under BW, paths in the first category have the same $1/M^2$ weighting as under RAS. However, paths in the second category have a weight of $1/(M(M-1))$ because the action used at the first time step cannot be used again from the initial state, so the probability of choosing any specific action for the second time step is $1/(M-1)$. Furthermore, paths in the third category are not possible under BW, thus

$$\begin{aligned} P(\tau_B > 2) = & \frac{1}{M^2} \sum_{m_1=1}^{M} \sum_{m_2=1}^{M} \sum_{n_1=2}^{N-1} \sum_{n_2=1}^{N-1} P(1, n_1, m_1) P(n_1, n_2, m_2) \\ & + \frac{1}{M(M-1)} \sum_{m_1=1}^{M} \sum_{\substack{m_2=1 \\ m_2 \neq m_1}}^{M} \sum_{n_2=1}^{N-1} P(1, 1, m_1) P(1, n_2, m_2). \end{aligned}$$

When the difference is taken, the probabilities for paths in the first category will cancel. The index n_1 is no longer used, so the subscript on n_2 can be dropped without ambiguity.

$$\begin{aligned} P(\tau_B > 2) - P(\tau_R > 2) = & \frac{1}{M(M-1)} \sum_{m_1=1}^{M} \sum_{\substack{m_2=1 \\ m_2 \neq m_1}}^{M} \sum_{n=1}^{N-1} P(1, 1, m_1) P(1, n, m_2) \\ & - \frac{1}{M^2} \sum_{m_1=1}^{M} \sum_{\substack{m_2=1 \\ m_2 \neq m_1}}^{M} \sum_{n_2=1}^{N-1} P(1, 1, m_1) P(1, n, m_2) \\ & - \frac{1}{M^2} \sum_{m_1=1}^{M} \sum_{n=1}^{N-1} P(1, 1, m_1) P(1, n, m_1). \end{aligned}$$

Obtain a common denominator of $M^2(M-1)$ so that the quantities can be combined. This will introduce a $M-1$ in the numerator of the third quantity, which will disappear when it is absorbed into the sum over m_2.

$$\begin{aligned} & P(\tau_B > 2) - P(\tau_R > 2) \\ = & \frac{1}{M^2(M-1)} \sum_{m_1=1}^{M} \sum_{\substack{m_2=1 \\ m_2 \neq m_1}}^{M} \sum_{n=1}^{N-1} \left(P(1, 1, m_1) P(1, n, m_2) - P(1, 1, m_1) P(1, n, m_1) \right) \\ = & \frac{1}{M^2(M-1)} \sum_{m_1=1}^{M} \left(P(1, 1, m_1) \sum_{\substack{m_2=1 \\ m_2 \neq m_1}}^{M} \sum_{n=1}^{N-1} \left(P(1, n, m_2) - P(1, n, m_1) \right) \right). \end{aligned}$$

From the rule of complements, $P(1,N,m) = 1 - \sum_{N=1}^{N-1} P(1,n,m)$ for any action m. Apply this to the previous expression. Then

$$\begin{aligned}
P(\tau_B > 2) - P(\tau_R > 2) =& \frac{1}{M^2(M-1)} \sum_{m_1=1}^{M} \left(P(1,1,m_1) \sum_{\substack{m_2=1 \\ m_2 \neq m_1}}^{M} (P(1,N,m_1) - P(1,N,m_2)) \right) \\
=& \frac{1}{M^2} \sum_{m_1=1}^{M} \left(P(1,1,m_1) \left(P(1,N,m_1) - \frac{\sum_{m_2=1, m_2 \neq m_1}^{M} P(1,N,m_2)}{M-1} \right) \right) \\
=& \frac{1}{M^2} \sum_{m_1=1}^{M} P(1,1,m_1) \left(P(1,N,m_1) - \overline{p}_{-m_1} \right)
\end{aligned}$$

as desired.

References

1. Pathical, S.; Serpen, G. Comparison of subsampling techniques for random subspace ensembles. In Proceedings of the 2010 International Conference on Machine Learning and Cybernetics, Qingdao, China, 11–14 July 2010; Volume 1, pp. 380–385. [CrossRef]
2. Kumar, S.; Mohri, M.; Talwalkar, A. Sampling Methods for the Nyström Method. *J. Mach. Learn. Res.* **2012**, *13*, 981–1006.
3. Williams, C.K.I.; Seeger, M. Using the Nyström Method to Speed Up Kernel Machines. In *Advances in Neural Information Processing Systems 13*; Leen, T.K., Dietterich, T.G., Tresp, V., Eds.; MIT Press: Cambridge, MA, USA, 2001; pp. 682–688.
4. Schneider, M. Probability Inequalities for Kernel Embeddings in Sampling without Replacement. In Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, Cadiz, Spain, 9–11 May 2016; Volume 51, pp. 66–74.
5. Cannon, A.; Ettinger, J.M.; Hush, D.; Scovel, C. Machine Learning with Data Dependent Hypothesis Classes. *J. Mach. Learn. Res.* **2002**, *2*, 335–358.
6. Feng, X.; Kumar, A.; Recht, B.; Ré, C. Towards a Unified Architecture for in-RDBMS Analytics. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, SIGMOD '12*; ACM: New York, NY, USA, 2012; , pp. 325–336. [CrossRef]
7. Gürbüzbalaban, M.; Ozdaglar, A.; Parrilo, P. Why Random Reshuffling Beats Stochastic Gradient Descent. *arXiv* **2015**, arXiv:1510.08560, [arXiv:math.OC/1510.08560].
8. Meng, Q.; Chen, W.; Wang, Y.; Ma, Z.M.; Liu, T.Y. Convergence analysis of distributed stochastic gradient descent with shuffling. *Neurocomputing* **2019**. [CrossRef]
9. Ying, B.; Yuan, K.; Vlaski, S.; Sayed, A.H. Stochastic Learning Under Random Reshuffling with Constant Step-sizes. *IEEE Trans. Signal Process.* **2019**, *67*. [CrossRef]
10. Shamir, O. Without-Replacement Sampling for Stochastic Gradient Methods. In *Advances in Neural Information Processing Systems 29*; Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; pp. 46–54.
11. Sutton, R.S.; Barto, A.G. *Introduction to Reinforcement Learning*, 1st ed.; MIT Press: Cambridge, MA, USA, 1998.
12. Puterman, M. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*; Wiley Series in Probability and Statistics; Wiley-Interscience: Hoboken, NJ, USA, 2005.
13. Tsitsiklis, J.N.; Roy, B.V. An Analysis of Temporal-Difference Learning with Function Approximation. *IEEE Trans. Autom. Control* **1997**, *42*, 674–690. [CrossRef]
14. Watkins, C.J.C.H.; Dayan, P. Q-learning. *Mach. Learn.* **1992**, *8*, 279–292. [CrossRef]
15. Carden, S.W. Convergence of a Q-learning Variant for Continuous States and Actions. *J. Artif. Intell. Res.* **2014**, *49*, 705–731. [CrossRef]

16. Wunder, M.; Littman, M.; Babes, M. Classes of Multiagent Q-learning Dynamics with ϵ-greedy Exploration. In Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 1167–1174.
17. Tijsma, A.D.; Drugan, M.M.; Wiering, M.A. Comparing exploration strategies for Q-learning in random stochastic mazes. In Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 6–9 December 2016; pp. 1–8. [CrossRef]
18. Kaelbling, L. Learning in Embedded Systems. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 1990.
19. Auer, P. Using Confidence Bounds for Exploitation-Exploration Trade-offs. *J. Mach. Learn. Res.* **2002**, *3*, 397–422.
20. Sutton, R.S. Integrated Architectures for Learning, Planning, and Reacting Based on Approximating Dynamic Programming. In *Machine Learning Proceedings 1990*; Porter, B., Mooney, R., Eds.; Morgan Kaufmann: San Francisco, CA, USA, 1990; pp. 216–224. [CrossRef]
21. Thrun, S.B. *Efficient Exploration in Reinforcement Learning*; Technical Report CMU-CS-92-102; Carnegie-Mellon University: Pittsburgh, PA, USA, 1992.
22. Martin, J.; Sasikumar, S.N.; Everitt, T.; Hutter, M. Count-Based Exploration in Feature Space for Reinforcement Learning. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, Melbourne, Australia, 19–25 August 2017; pp. 2471–2478. [CrossRef]
23. Xu, Z.X.; Chen, X.L.; Cao, L.; Li, C.X. A study of count-based exploration and bonus for reinforcement learning. In Proceedings of the 2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China, 28–30 Arpil 2017, pp. 425–429. [CrossRef]
24. Tang, H.; Houthooft, R.; Foote, D.; Stooke, A.; Xi Chen, O.; Duan, Y.; Schulman, J.; DeTurck, F.; Abbeel, P. #Exploration: A Study of Count-Based Exploration for Deep Reinforcement Learning. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; pp. 2753–2762.
25. Kearns, M.; Singh, S. Near-Optimal Reinforcement Learning in Polynomial Time. *Mach. Learn.* **2002**, *49*, 209–232. [CrossRef]
26. Kolobov, A.; Mausam.; Weld, D.S. A Theory of Goal-oriented MDPs with Dead Ends. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence, UAI'12*; AUAI Press: Arlington, VA, USA, 2012; pp. 438–447.
27. Bertsekas, D.P.; Tsitsiklis, J.N. An Analysis of Stochastic Shortest Path Problems. *Math. Oper. Res.* **1991**, *16*, 580–595. [CrossRef]
28. Biler, P.; Witkowski, A. *Problems in Mathematical Analysis*; CRC Press: Boca Raton, FL, USA, 1990.
29. Resnick, S.I. *Adventures in Stochastic Processes*; Birkhäuser: Basel, Switzerland, 1992.
30. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2017. Available online: https://www.R-project.org/ (accessed on 3 April 2019).
31. Trevizan, F.W.; Teichteil-Königsberg, F.; Thiébaux, S. Efficient solutions for Stochastic Shortest Path Problems with Dead Ends. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*; AUAI Press: Corvallis, OR, USA, 2017.

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).