International Journal on Software Tools for Technology Transfer (2026) 28:55–70
https://doi.org/10.1007/s10009-026-00851-3

GENERAL

QEST+FORMATS 2024

Active discount factor elicitation via reward modification

Shadi Tasdighi Kalat[1] · **Sriram Sankaranarayanan**[1] · **Ashutosh Trivedi**[1]

Accepted: 27 February 2026 / Published online: 25 March 2026

Abstract
Agent behavior is shaped by latent decision parameters that govern how rewards are interpreted and traded off over time. A key example is the *discount factor*, which encodes time preference. Mis-specifying the discount factor can confound reward-centric behavioral models (e.g., inverse RL), motivating the need to infer time preference directly from behavior. This paper presents methods for *discount factor elicitation* in finite-state Markov Decision Processes via policy observations and controlled reward modifications. First, we introduce an algorithm that bounds the set of discount factors consistent with an agent's observed (near-)optimal policy, and show how observations across heterogeneous reward settings progressively tighten these bounds. Building on this result, we propose an *active elicitation framework* in which an ego agent strategically adjusts rewards (with fixed dynamics) to refine its estimate of another agent's discount factor. Through case studies, we demonstrate that active elicitation accelerates interval refinement relative to passive observation and enables *targeted exploration* in strategic multi-agent settings. Overall, our results establish reward modification as a principled mechanism for eliciting discount factors and improving behavioral modeling, prediction, and control.

Keywords Discount factor · Markov decision process · Active elicitation · Reward modification

1 Introduction

The recent success of *reinforcement learning* (RL) algorithms [33] in discovering creative solutions with superhuman performance in complex domains [26, 39] has contributed to the widespread adoption of the discounted-sum objective as a canonical optimization criterion. The discounted-sum formalism captures the notion of time preference, or rate of impatience [14], which governs how rational agents trade off immediate rewards against larger rewards realized in the future. However, characterizing this preference is challenging, as it is inherently subjective and not identifiable from a single *choice*. This paper develops algorithms for identifying exact and approximate ranges of an agent's discount factor from observed policies, addressing a fundamental identifiability challenge that arises when time preference is misspecified.

These challenges are well illustrated by simple behavioral settings in which agents must choose between immediate and delayed rewards. A canonical example is the *marshmallow experiment* [24], shown in Fig. 1(a), which studies children's ability to delay gratification by offering a choice between one marshmallow immediately or two marshmallows after a fixed delay. Under a discounted-sum model, consuming the immediate reward is optimal for discount factors in the range $[0, \frac{1}{2})$, while waiting is optimal for discount factors in $(\frac{1}{2}, 1)$.

At first glance, this setting appears to allow direct inference of time preference: agents who wait are labeled patient or future-oriented, while those who consume the immediate reward are labeled myopic (Fig. 1(b)). However, this interpretation masks a fundamental identifiability issue. From a single intertemporal choice, it is impossible to distinguish whether an agent's behavior reflects impatience (a low discount factor) or a different internal valuation of rewards. As illustrated in Fig. 1(c), a purely reward-centric approach such as inverse reinforcement learning can rationalize the same observed choice using widely different discount factors by biasing the recovered reward function, leading to incorrect behavioral models when the discount factor is misspecified.

✉ S. Tasdighi Kalat
Shadi.TasdighiKalat@colorado.edu

S. Sankaranarayanan
srirams@colorado.edu

A. Trivedi
ashutosh.trivedi@colorado.edu

[1] Department of Computer Science, University of Colorado Boulder, Boulder, USA

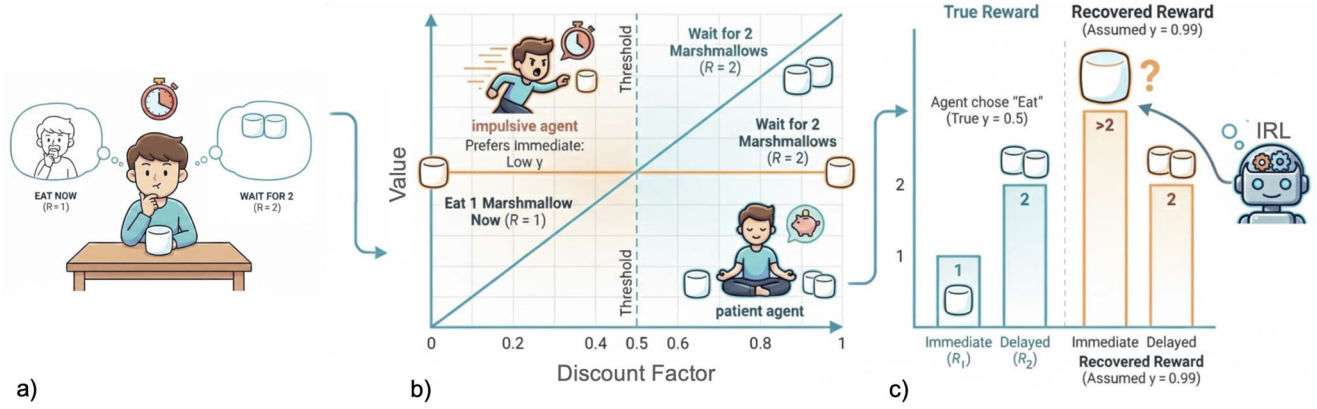

Fig. 1 The Marshmallow Experiment. (a) The experiment probes an agent's time preference by offering a choice between an immediate reward (one marshmallow) and a larger reward (two marshmallows) after a fixed delay. (b) Under a naive interpretation, agents who consume the immediate reward are labeled myopic, while those who wait are considered future-oriented. (c) However, a single intertemporal choice does not reveal whether an agent's behavior reflects impatience or a different internal valuation of rewards, making reward-centric inference unidentifiable. If the discount factor is misspecified, an inverse reinforcement learning (IRL) algorithm (e.g., maximum likelihood) can recover a distorted reward function that rationalizes the observed behavior, thereby confounding impatience with reward valuation

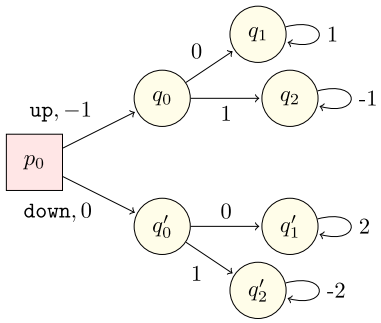

Fig. 2 A turn-based discounted game in which knowledge of the opponent's discount factor enables the selection of actions that yield higher long-term utility. Player Max may take a locally suboptimal action to disambiguate whether Player Min is myopic or foresighted, and exploit this information in subsequent rounds

This example highlights the need to elicit discount factors explicitly, rather than treating them as fixed or known parameters.

Discount factor elicitation. This work focuses on settings in which an agent's subjective discount factor remains constant over the duration of the interaction. In such setups, eliciting discount factor becomes a canonical inverse-RL [27] task[1] as it reveals a key hyperparameter for transfer across environments. Our goal is to uncover intrinsic characteristics of an agent that shape its decision-making dynamics, thereby enabling more accurate prediction of behavior in unfamiliar situations.

[1] Inverse reinforcement learning (IRL) typically assumes the agent's discount factor is known and infers latent reward functions from observed behavior in a fixed environment. In contrast, discount factor elicitation treats rewards as controllable and seeks to recover the agent's discount factor from behavior observed across multiple reward settings.

The importance of discount factor elicitation is particularly pronounced in multi-agent interactions, where knowledge of other agents' discount factors can inform strategic planning, improve policy anticipation, and help avoid overly optimistic or overly conservative responses. Consider the two-player turn-based discounted game shown in Fig. 2, where circular nodes are controlled by Player Min and the square node by Player Max. Suppose Player Min's discount factor is either 0 (myopic) or 1 (fully foresighted), while Player Max's discount factor is 1. If Player Max does not know Player Min's discount factor, their optimal action is `down`. However, Player Max can instead take the locally suboptimal action `up` to disambiguate whether Player Min is myopic or foresighted, and then exploit this information to obtain a higher payoff in subsequent rounds.

Passive discount factor inference. For finite MDPs, Blackwell [4] introduced the notion of *discount-optimal* policies, which are optimal for all sufficiently large discount factors. A policy is called *Blackwell-optimal* if there exists a threshold γ_* such that the policy is optimal for every $\gamma \in [\gamma_*, 1)$. Gurvich and Miltersen [17] derived an explicit upper bound on γ_*, showing that

$$\gamma_* \leq 1 - \frac{1}{(n!)^2 2^{2n+3} M^{2n^2}},$$

where n denotes the number of states in the MDP and M bounds the absolute values of transition probabilities and rewards. Puterman [29, Sect. 10.1.2] provides an accessible proof of the existence of Blackwell-optimal policies based on a rational characterization of the value as a function of the discount factor, together with the finiteness of stationary policies. These results imply that, for any finite MDP, there

exists a finite sequence of discount-factor thresholds

$$\mathcal{L} = \langle a_0 (= 0) \leq a_1 \leq a_2 \leq \cdots \leq a_N = \gamma_* \leq 1 \rangle$$

such that, for each interval $[a_i, a_{i+1}]$ with $0 \leq i < N$, the set of optimal policies remains invariant.

Building on this landscape, Smallwood [32] established a key structural refinement: at each threshold value a_i, the sets of optimal policies on either side of a_i differ in exactly one state. This observation enables the identification of discount-factor values at which an agent is indifferent between two optimal policies. Given an observed optimal policy, one can compute the range of discount factors for which that policy remains optimal. However, a direct application of this approach is computationally prohibitive in general. To address this challenge, we develop algorithms for computing exact and approximate discount-factor ranges for both optimal and sub-optimal policies, and present experimental results demonstrating the effectiveness of the proposed methods.

From passive inference to active discount factor elicitation. We propose an *active* elicitation approach in which the environment is strategically modified so that each interaction refines the estimate of an agent's discount factor. Our method keeps transition dynamics fixed and selectively adjusts reward values to construct environments that are maximally informative about the agent's time preferences. We formalize this process as a multi-stage framework in which an observer iteratively designs reward modifications, observes the agent's optimal policy response, and updates the feasible range of discount factors. Each stage reduces to a one-step optimization problem that selects reward updates minimizing the remaining uncertainty about the agent's discount factor. Together, these ideas yield a principled mechanism for inferring time preferences through controlled reward shaping. Beyond improving inference accuracy, active discount-factor elicitation enables both predictive and prescriptive capabilities in downstream decision-making tasks.

Applications of discount factor elicitation: behavior prediction. Knowledge of an agent's discount factor enables more accurate prediction of behavior in sequential decision-making contexts. For example, in personalized recommendation systems on online shopping platforms, the discount factor reflects how much a user values future utility relative to immediate gratification. A higher discount factor may indicate that a user anticipates long-term use of a product and is therefore more willing to pay a premium for durability or sustained benefits. By estimating users' discount factors, such platforms can better predict purchasing behavior and tailor recommendations, pricing strategies, and promotions accordingly.

Applications of discount factor elicitation: behavior shaping. In addition to supporting behavior prediction, discount factor elicitation plays a central role in behavior shaping in multi-agent and population-level settings. By designing policies and incentives that account for heterogeneity in agents' discount factors, it becomes possible to anticipate responses to interventions and align individual choices with long-term system objectives. This enables more effective and targeted interventions that respect agents' time preferences while promoting desired outcomes.

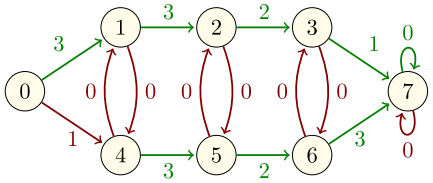

Fig. 3 Each state offers two actions, shown in red and green. The green action is optimal for myopic agents (discount factor < 0.3334), while more patient agents choose the red action at state 3. Introducing a penalty on edge $6-7$ shifts this threshold, redirecting agents with intermediate discount factors (e.g., < 0.5) toward the green action

For a concrete illustration, consider the MDP shown in Fig. 3, which represents alternative routes between states 0 and 7 in a stylized traffic network. At each state, an agent can choose between two actions, shown in red and green. Empirically, agents with low discount factors (myopic agents) select the green action at every state, while more patient agents (with discount factor strictly greater than 0.3334) diverge at state 3, choosing the red action there and the green action elsewhere.

Now suppose system designer wishes to predict and influence routing behavior for agents with intermediate discount factors, in particular those in the range $[0, 0.5)$. By introducing a penalty on edge $6-7$, the designer can shift the policy threshold so that these agents also prefer the green action at state 3. In this example, a unit penalty on edge $6-7$ suffices to move the policy transition point from 0.3334 to 0.5, thereby redirecting the behavior of agents with intermediate time preferences. This example illustrates how knowledge of an agent's discount factor enables principled reward modifications that both predict and shape behavior in sequential decision-making environments.

Contributions. This paper makes the following contributions:

1. **Discount factor inference.** We develop algorithms for computing exact and approximate ranges of discount factors consistent with an agent's observed policy in a finite MDP, extending classical results on discount-optimal policies to practical inference settings.
2. **Handling sub-optimal policies.** We extend the inference framework to accommodate sub-optimal and noisy policies, enabling robust discount factor estimation when optimality cannot be assumed.

3. **Active discount factor elicitation.** We introduce an active elicitation framework in which an observer strategically modifies reward functions while keeping transition dynamics fixed, ensuring that successive interactions are maximally informative about an agent's discount factor.
4. **Empirical evaluation.** We present case studies and experimental results demonstrating that active discount factor elicitation accelerates uncertainty reduction relative to passive observation, and that inferred discount factors can be leveraged for both behavior prediction and behavior shaping in sequential decision-making settings.

This paper is an extended version of the study published in [34], with the primary novel contribution being the development of an *active* elicitation framework for discount factors.

Organization. We begin with a discussion of related work in Sect. 2, followed by a statement of the discount factor elicitation problem in Sect. 3. Our approach builds on classical results due to Smallwood, which are reviewed in Sect. 3.1.1. The core algorithmic contributions are developed in Sect. 4, with an extension to accommodate sub-optimal policies presented in Sect. 3.1.2. In Sect. 6, we present case studies illustrating how an agent's discount factor can be inferred from observed policies in varying environments. Motivated by these examples, we then present results from implementing the *elicitation framework*, in which an observer strategically shapes reward functions to obtain precise estimates of another agent's discount factor over a finite number of interactions. Finally, we apply this framework to address the exploration–exploitation tradeoff in multi-agent interactions.

2 Related work

Discounted optimization. Puterman [29] and Filar & Vrieze [13] provide comprehensive treatments of optimization over Markov Decision Processes (MDPs) and stochastic games. Discounted, total, and average reward are among the most studied optimization objectives. Among these, the discounted-sum objective is the best understood, theoretically elegant, and admits effective optimization [29] and reinforcement learning [33] algorithms. The notion of Blackwell optimality [4] allows average reward optimization to be reduced to discounted-sum optimization.

Discount factor. Starting from the work of Fisher [14], the discount factor has received several interpretations, including rate of impatience [14], delayed gratification [15, 25], geometrically distributed horizons [22], system failure [29], timing of consumption and resource expenditure [19], and stochastic shortest path formulations under proper-policy assumptions [28]. Despite its central role in optimization and games, there is no consensus among practitioners on how to select an appropriate discount factor, and the problem has received comparatively limited attention [2, 15, 18, 35].

Smallwood's work [32] is perhaps closest to our setting; however, his objective is to characterize discount factors under which two optimal policies are equally profitable. Giwa et al. [16] propose an inverse reinforcement learning approach that jointly optimizes rewards and the discount factor from observed behavior. The key differences between our work and that of Giwa et al. are that (a) we assume the reward function and the full policy are observable for a given MDP and infer the discount factor, whereas Giwa et al. do not assume rewards are known; and (b) we characterize the entire set of discount factors for which a given policy is optimal, whereas Giwa et al. rely on gradient-based likelihood optimization, which yields a local optimum.

Mismatching discounts factors. In strategic games, the presence of agents with different planning horizons—and hence different discount factors—is well documented [1, 7, 20]. Lehrer et al. [20] show that in complete-information zero-sum games, equilibrium payoffs remain zero-sum even under heterogeneous discounting. They further demonstrate that differences in discounting can induce partial cooperation through inter-temporal trades, despite the underlying competitive structure [1].

Discount factor elicitation. Mechanism design studies how to structure interactions so as to induce desirable outcomes [41], including the elicitation of agents' preferences or private information [5, 12]. A variety of mechanisms—such as prediction markets [8], peer prediction [23], truth serum methods [36, 37], and ludic games [6]—have been proposed to elicit beliefs or preferences.

In this context, our work focuses on eliciting the discount factor, which captures an agent's preference for immediate versus future rewards. In economics, this problem has been studied using survey-based methods such as multiple price lists, where agents choose between a smaller immediate payment and a larger delayed one [3, 9, 38]. At a high level, these approaches correspond to varying the temporal placement of rewards and observing resulting choices, which aligns closely with the elicitation framework studied in this paper.

3 Problem definition

We study the problem of inferring an agent's unknown discount factor from an observed policy in a finite MDPs, which we define next.

Definition 1
A Markov Decision Process (MDP) $\mathcal{M}$ is a tuple $(S, A, (P_a)_{a\in A}, (R_a)_{a\in A})$, where:

- $S = \{1, \ldots, N\}$ is a finite set of states;

- A is a finite set of actions;
- for each action $a \in A$, P_a is a $|S| \times |S|$ transition probability matrix, where $P_a(i,j)$ denotes the probability of transitioning from state i to state j when action a is taken; and
- for each action $a \in A$, $R_a(i,j) \in \mathbb{R}$ denotes the reward obtained by taking action a in state i and transitioning to state j.

Assumption 1
We assume that for any state $i \in S$ and any two actions $a_1, a_2 \in A$ such that $a_1 \neq a_2$, the probabilities $P_{a_1}(i,j) \neq P_{a_2}(i,j)$ for some $j \in S$. This assumption avoids cases in which two actions induce identical transition distributions.

The agent seeks to maximize the discounted sum of rewards. It is well known [4, 29] that for discounted objectives, deterministic and memoryless policies suffice for optimality. Accordingly, we restrict attention to such policies. Formally, a deterministic, memoryless *policy* $\pi : S \to A$ for $\mathcal{M}$ is a mapping from the set of states S to the set of actions A. We write Π for the set of all possible policies of the observed agent. We represent every policy as a $|S|$-vector that maps each state to an action. Given a policy π, we define the transition matrix P_π as $P_\pi(i,j) = P_{\pi(i)}(i,j)$. In other words, $P_\pi(i,j)$ is the probability of moving from i to j upon the action $\pi(i)$. Similarly, let $R_\pi(i,j)$ denote the reward $R_{\pi(i)}(i,j)$.

The *value* associated with policy $\pi \in \Pi$, denoted $\mathbf{v}_\pi$, maps every $s \in S$ to the discounted sum of rewards gained by following policy π starting from s. The value function is characterized by the following equations:

$$\mathbf{v}_\pi(i) = \sum_{j=1}^{N} P_\pi(i,j)\,(R_\pi(i,j) + \gamma \mathbf{v}_\pi(j)) \tag{1}$$

where $\gamma \in [0,1)$ is the *discount factor*.

3.1 Discount factor inference

The discount factor controls the importance of immediate versus future rewards in decision making. Small values of γ emphasize near-term gains, whereas larger values place greater weight on future rewards. Using the definitions above, we now formulate the discount factor elicitation problem and discuss approaches for solving it.

Problem 1 (Discount Factor Inference)
Assume that the agent plays according to a given policy $\pi \in \Pi$, which is an optimal policy corresponding to some unknown discount factor γ. We seek to identify a closed interval (or union of intervals) I of the form $[\gamma_l, \gamma_u]$ for $0 \leq \gamma_l \leq \gamma_u < 1$ or a half-open interval of the form $I : [\gamma_l, 1)$ such that for all $\gamma \in I$, the given policy π is optimal for the value of γ provided, i.e., for each $\gamma \in I$, and for any policy $\xi \in \Pi$ and for all states i, we have

$$\mathbf{v}_\pi(i) = \sum_{j=1}^{N} P_\pi(i,j)\,(R_\pi(i,j) + \gamma \mathbf{v}_\pi(j)) \geq \mathbf{v}_\xi(i). \tag{2}$$

This problem can be reduced to a decision problem involving univariate rational functions through a standard linear-programming (LP) based approach. The value function for a given discount factor γ is obtained as a vector $\mathbf{v}$, wherein $\mathbf{v}(i)$ is the value associated with state i, via the following LP [29]:

$$\begin{aligned} \min_{\mathbf{v}} \quad & \sum_{i=1}^{N} \mathbf{v}(i) \\ \text{s.t.} \quad & \mathbf{v}(i) - \gamma \sum_{j=1}^{N} P_a(i,j)\,\mathbf{v}(j) \;\geq\; \mathbf{q}_a(i), \\ & \forall\, a \in A,\ \forall\, i \in \{1,\ldots,N\}, \end{aligned}$$

where $\mathbf{q}_a(i) = \sum_{j=1}^{n} R_a(i,j) P_a(i,j)$ for $a \in A$ is the vector of expected immediate rewards. Given the value function, the policy π can be extracted from the optimal solution of the LP. In particular, whenever $\pi(i) = a$, the constraint corresponding to state i and action a is saturated. Note that if there are multiple optimal policies π, π' for a given discount factor, then the constraints corresponding to the actions chosen by each of the policies are saturated by the optimal solution to the LP above. To state such constraints, we start by the following lemma which proves the invertability of matrix $(I - \gamma P_\pi)^{-1}$.

Lemma 1
For $\gamma \in [0,1)$ *we have* $\det(I - \gamma P_\pi) > 0$.

Proof
Since P_π is a stochastic matrix, its eigenvalues λ are such that $|\lambda| \leq 1$. The real roots of $\det(I - \gamma P_\pi)$ are in fact $\frac{1}{\lambda}$ where λ is a non-zero real eigenvalue of P_π. Since $|\lambda| \leq 1$, we conclude that $\det(I - \gamma P_\pi)$ has no real roots in $[0,1)$ and is thus sign invariant. Thus, $\det(I - \gamma P_\pi) > 0$ since it is positive for $\gamma = 0$. □

Next, we state the conditions for optimality of a policy using the following lemma.

Lemma 2
A policy π *is optimal for a given discount factor* $\gamma \in [0,1)$ *if and only if the following constraints hold for each action* $a \in A$*:*

$$(I - \gamma P_a)(I - \gamma P_\pi)^{-1} \mathbf{q}_\pi \geq \mathbf{q}_a\,,$$

wherein $\mathbf{q}_\pi(i) = \mathbf{q}_{\pi(i)}(i)$.

Proof
Let π be the optimal policy and $\mathbf{v}^*$ be the corresponding value function. It holds that $\mathbf{v}^*$ must be a feasible solution to the LP above that satisfies:

$$\mathbf{v}^*(i) - \gamma \sum_{j=1}^{n} P_\pi(i,j)\mathbf{v}^*(j) = \mathbf{q}_\pi(i) \;\; \forall i \in \{1,\ldots,N\}\,.$$

In other words, $\mathbf{v}^* - \gamma P_\pi \mathbf{v}^* = \mathbf{q}_\pi$. Since $I - \gamma P_\pi$ is an invertible matrix for $\gamma \in [0,1)$ and P_π being a stochastic matrix, we have $\mathbf{v}^* = (I - \gamma P_\pi)^{-1}\mathbf{q}_\pi$. Also, $\mathbf{v}^*$ must be primal feasible as well, yielding $(I - \gamma P_a)\mathbf{v}^* \geq \mathbf{q}_a$ for every $a \in A$. Therefore, we may eliminate $\mathbf{v}^*$ to obtain

$$(I - \gamma P_a)(I - \gamma P_\pi)^{-1}\mathbf{q}_\pi \geq \mathbf{q}_a\,, \forall a \in A.$$

This completes the proof. □

As a result, a policy π is optimal for some discount factor $\gamma \in [0,1)$ if and only if the following assertion over the reals is valid.

$$\exists \gamma \in [0,1) \bigwedge_{a \in A} (I - \gamma P_a)(I - \gamma P_\pi)^{-1}\mathbf{q}_\pi \geq \mathbf{q}_a\,. \quad (3)$$

Note that each $(I - \gamma P_\pi)^{-1}$ is a $N \times N$ matrix whose entries are rational functions (ratio of two polynomials in γ), wherein the degrees of the numerator and denominator are at most N. The denominators are all given by $\det(I - \gamma P_\pi)$, which is positive over $\gamma \in [0,1)$.

Therefore, the problem of checking whether a given policy π is optimal for some discount factor γ can be reduced to checking whether for some $\gamma \in [0,1)$, a given list of $N|A|$ polynomials involving γ of degree at most $N+1$ is all non-negative. The roots of a given polynomial delineate intervals where the sign of the polynomial does not change. Therefore, to find intervals where the polynomials are all non-negative, we first identify their roots. In practice, this can be solved using real-root isolation for the polynomials using Sturm sequences or the Descartes rule of signs to count the number of real roots in an interval and bisection to refine these intervals [30, 40]. Having identified all such sign invariant intervals, the procedure checks if the intervals corresponding to the polynomials have a non-empty intersection. In what follows, we will provide a more elegant approach using a result from Smallwood [32].

3.1.1 Optimal policy regions

Our approach to Problem 1 builds upon the result of Smallwood [32], which shows that as we vary the discount factor γ over the interval $[0,1)$, we effectively partition the interval $[0,1)$ into finitely many intervals $I_1,\ldots,I_K$ such that for each interval I_l, there is an associated policy π_l which is optimal for any discount factor $\gamma \in I_l$. Furthermore, two neighboring intervals I_l, I_{l+1} that share an end-point have associated policies π_l, π_{l+1} that differ only at a single state.

Theorem 1 (Smallwood [32])
For a given MDP $\mathcal{M}$, there exists finitely many points $0 = a_0 \leq a_1 \leq a_2 \cdots \leq a_N < 1$ and policies $\pi_0, \pi_1, \ldots, \pi_N$ such that

(a) for $i < N$, π_i is an optimal policy for any discount factor $\gamma \in [a_i, a_{i+1}]$,
(b) π_N is an optimal policy for $[a_N, 1)$,
(c) "neighboring" policies π_i, π_{i+1} differ only at a single state, and
(d) the optimal values associated with π_i, π_{i+1} are the same for all states at discount factor a_{i+1}.

Following Smallwood, we call an interval of the form $[a_i, a_{i+1}]$ (or $[a_N, 1)$) an *optimal policy region*. For any two neighboring optimal policies π_{i-1}, π_i for $i \geq 1$, the point a_i that is common to their respective optimal policy regions is called an *indifference point*. Note that a policy π can be optimal for two separate intervals. Therefore, there could be multiple indifference points between two policies. At an indifference point, the optimal action for one state of the MDP changes, resulting in a change from a policy π_i to π_{i+1}. Note that at the indifference point, the value functions associated with π_i and π_{i+1} are the same.

Based on Smallwood's Theorem, we propose the following simple strategy for identifying the optimal policy intervals corresponding to a given policy π:

1. Iterate through all policies π' that differ from π at just a single state. There are $N(|A|-1)$ such policies.
2. For each such policy π' find if an indifference point exists between π, π'.
3. If π, π' have an indifference point γ for which they are optimal, then we have discovered one of the end points of an interval for which π is optimal.

The key observation [32] is that we can check if two policies π, π' have an indifference point in common using polynomial root solving. Let π, π' be two policies that differ in just one state. The value function $\mathbf{v}$ for π satisfies the equation: $\mathbf{v} - \gamma P_\pi \mathbf{v} = \mathbf{q}_\pi$. Likewise, π' must have the same value function that also satisfies: $\mathbf{v} - \gamma P_{\pi'}\mathbf{v} = \mathbf{q}_{\pi'}$. Subtracting, we obtain:

$$\mathbf{q}_\pi - \mathbf{q}_{\pi'} + \gamma(P_\pi - P_{\pi'})\mathbf{v} = 0.$$

Since, π and π' differ from each other in one state, $\mathbf{q}_\pi$, and $\mathbf{q}_{\pi'}$ differ in just one entry corresponding to the state where the policies diverge. Similarly, P_π, and $P_{\pi'}$ also differ just in a single row where the two policies diverge. Therefore, we obtain a single equation pertaining to just the row where the two policies differ:

$$\Delta q + \gamma \Delta P (I - \gamma P_\pi)^{-1}\mathbf{q}_\pi = 0\,. \quad (4)$$

Here Δq is a scalar representing the difference between immediate expected rewards obtained by the two policies, and $\Delta P = (P_\pi - P_{\pi'})$ is a row vector. Assumption 1 ensures that $\Delta P \neq 0$. Otherwise, note that π, π' cannot share an indifference point. By Cramer's rule, the mapping

$$\gamma \mapsto (I - \gamma P)^{-1}$$

is a rational function whose common denominator is $\det(I - \gamma P)$ [21]. Therefore, solving Eq. (4) for γ reduces to finding the roots of the resulting rational function within the interval $[0, 1)$.

The numerators and common denominator for the entries of $(I - \gamma P_\pi)^{-1}$ can be computed efficiently by computing the characteristic polynomial of P_π and noting that P_π satisfies its own characterisitic polynomial using the Cayley-Hamilton theorem. The details are explained in Smallwood's paper [32] wherein the method of [11] is employed. Computing the polynomial in Eq. (4) will require $O(N^3)$ additions and multiplications involving entries of $\Delta q, \Delta P, P_\pi$ and $\mathbf{q}_\pi$.

Example 1
Consider an MDP with state set $S = \{0, 1, 2, 3\}$ and action set $A = \{0, 1\}$. The transition probabilities and rewards are given below.

Action 0:

$$P_0 = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{8} & \frac{1}{8} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}, \qquad q_0 = \begin{bmatrix} 8 & 20 & 2 & 3 \end{bmatrix}$$

Action 1:

$$P_1 = \begin{bmatrix} \frac{1}{16} & \frac{3}{8} & \frac{3}{16} & \frac{3}{8} \\ \frac{1}{16} & \frac{7}{16} & \frac{1}{16} & \frac{7}{16} \\ \frac{1}{8} & \frac{3}{8} & \frac{1}{8} & \frac{3}{8} \\ \frac{1}{8} & \frac{3}{8} & \frac{1}{8} & \frac{3}{8} \end{bmatrix}, \qquad q_1 = \begin{bmatrix} 3 & 15 & 4 & 40 \end{bmatrix}$$

As the discount factor γ varies over $[0, 1)$, the optimal policy changes at specific threshold values of γ, partitioning the interval $[0, 1)$ into subintervals over which the optimal policy remains invariant. For this MDP, the optimal policy is $[0, 0, 1, 1]$ for $\gamma \in [0, 0.38)$, $[0, 1, 1, 1]$ for $\gamma \in [0.38, 0.91)$, and $[1, 1, 1, 1]$ for $\gamma \in [0.91, 1)$.

To identify these intervals, it suffices to compute the boundary points at which two neighboring optimal policies yield identical value functions. Applying Eq. (4) to the policies $\pi_1 = [0, 1, 1, 1]$ and $\pi_2 = [0, 0, 1, 1]$, we obtain:

$$d_{\pi_1} = [0, 1, 1, 1],$$

$$P_{\pi_1} = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{8} & \frac{1}{8} \\ \frac{1}{16} & \frac{7}{16} & \frac{1}{16} & \frac{7}{16} \\ \frac{1}{8} & \frac{3}{8} & \frac{1}{8} & \frac{3}{8} \\ \frac{1}{8} & \frac{3}{8} & \frac{1}{8} & \frac{3}{8} \end{bmatrix},$$

$$q_{\pi_1} = \begin{bmatrix} 8 & 15 & 4 & 40 \end{bmatrix},$$

$$d_{\pi_2} = [0, 0, 1, 1],$$

$$\Delta P = \begin{bmatrix} -\frac{7}{16} & \frac{7}{16} & -\frac{7}{16} & \frac{7}{16} \end{bmatrix}, \qquad \Delta q = -5.$$

Solving Eq. (4) yields a root $\gamma_0 = 0.38$ in $[0, 1)$, which corresponds to the indifference point between π_1 and π_2. Figure 4(a) illustrates the resulting optimal policy regions and the corresponding optimal value as a function of γ.

So far, we formulated the problem of finding the range of discount factors for which a given policy is optimal. Next, we extend these results to find the discount factor range for which a given policy $\hat{\pi}$ is ϵ-optimal.

3.1.2 Near-optimal policy regions

Assume that $\hat{\pi}$ is a policy that is ϵ-far away from being optimal. We say that for a discount factor γ, $\hat{\pi}$ is ϵ optimal if the value function $\mathbf{v}_{\hat{\pi}}$ and the optimal value function $\mathbf{v}^*$ satisfy the inequality for some given $\epsilon \in [0, 1)$: $\mathbf{v}_{\hat{\pi}} \geq (1 - \epsilon)\mathbf{v}^*$.

Problem 2
Assume the agent plays according to known policy $\hat{\pi}$, find all discount factors such that $\hat{\pi}$ is ϵ optimal.

Clearly, $\hat{\pi}$ is ϵ optimal for all regions where it is optimal. However, (a) it need not be optimal anywhere and (b) it can be ϵ-optimal for discount factors that are "far away" from regions where $\hat{\pi}$ is optimal. From results in Sect. 3.1.1, we can compute all regions of discount factors as well as their respective optimum policies for a given MDP. Suppose π is an optimal policy over some region $[\gamma_1, \gamma_2]$. Let $\hat{P} = P_{\hat{\pi}}, \hat{\mathbf{q}} = \mathbf{q}_{\hat{\pi}}, P = P_\pi$ and $\mathbf{q} = \mathbf{q}_\pi$. Our goal is to compute a subset of the interval $[\gamma_1, \gamma_2]$ where $\hat{\pi}$ is ϵ-optimal. We seek to find γ such that

$$(I - \gamma\hat{P})^{-1}\hat{\mathbf{q}} \geq (1 - \epsilon)(I - \gamma P)^{-1}\mathbf{q}$$

or equivalently by defining $\Delta q = \mathbf{q} - \hat{\mathbf{q}}$, and $\Delta P = P - \hat{P}$, Equation (5) below can be solved for γ:

$$\begin{aligned} \Delta q \; + \; & (1 - \epsilon)(I - \gamma P)^{-1}\mathbf{q} \\ & - \; \gamma \Delta P (I - \gamma\hat{P})^{-1}\hat{\mathbf{q}} \; \geq \; 0. \end{aligned} \tag{5}$$

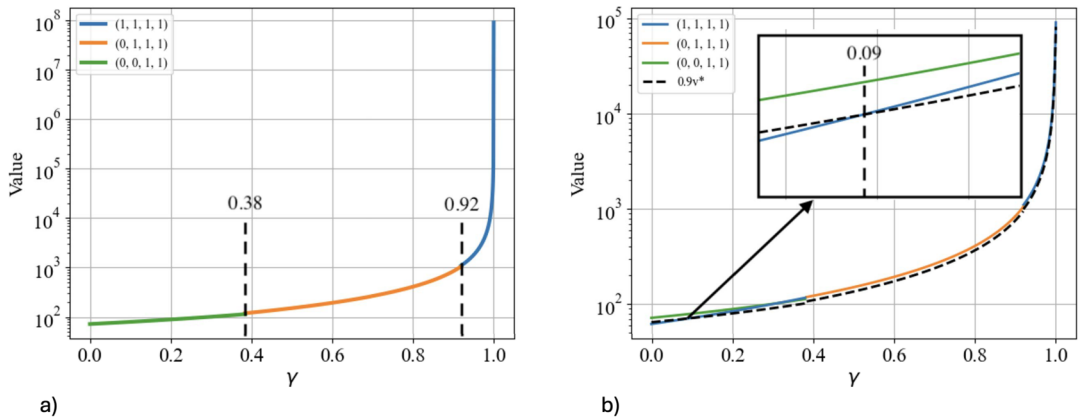

Fig. 4 (a) Optimum policy regions for the MDP in Example 1 (b) Range of discount factor for which policy $[1, 1, 1, 1]$ is near optimal for $\epsilon = 0.1$ is $[0.09, 1)$

However, we must note that $\hat{\pi}$ and π are not necessarily neighboring policies. Therefore, ΔP and Δq may have multiple non-zero entries, and Eq. (5) involves multiple polynomials of degree $2N$ that need to be non-negative for the value of γ. This can be solved through univariate polynomial quantifier elimination.

Example 2
Consider the MDP in Example 1 and its optimal policy regions. Suppose we wish to determine the range of discount factors $[\gamma_l^{0.1}, \gamma_u^{0.1}]$ for which the value of the policy $[1, 1, 1, 1]$ remains within $0.9\,v^*$ of the optimal value. Figure 4(b) illustrates this range.

4 Passive inference

Having formalized the problem of discount factor inference, we now turn to algorithmic solutions. We first study the *passive* setting, in which an observer infers an agent's discount factor solely from observed policies in fixed environments. This analysis establishes the structural and computational foundations underlying discount inference.

Algorithm 1 computes a union of intervals of the discount factor γ for which a given policy $\hat{\pi}$ is optimal for an MDP $\mathcal{M} = (S, A, (P_a)_{a\in A}, (R_a)_{a\in A})$ with $|S| = N$ states and $|A| = M$ actions. Let $H_1(\hat{\pi})$ denote the set of all policies that differ from $\hat{\pi}$ in exactly one state. Line 2 iterates over all policies $\pi_n \in H_1(\hat{\pi})$; for each such policy, the algorithm computes candidate indifference points by solving the corresponding polynomial equation, as described in Sect. 3.1.1.

For clarity of exposition, we assume that the roots of these polynomials can be computed exactly. In practice, the algorithm readily extends to the more realistic setting in which each root is isolated within an interval $\gamma \pm \delta$ for some tolerance $\delta > 0$; Sect. 4.1 details this refinement.

Each candidate indifference point γ_n is then tested using the isOptimal subroutine to determine whether $\hat{\pi}$ is optimal at γ_n. Valid indifference points are collected into a set Γ, which is subsequently sorted in ascending order. For each pair of consecutive points $\gamma_i, \gamma_{i+1} \in \Gamma$ (line 12), the algorithm checks whether $\hat{\pi}$ is optimal at the midpoint of the interval $[\gamma_i, \gamma_{i+1}]$. If so, this interval is added to the output.

Finally, the algorithm considers the interval $[\gamma_{\max}, 1)$, where $\gamma_{\max}$ is the largest element of Γ, to determine whether $\hat{\pi}$ is optimal for all sufficiently large discount factors, i.e., whether $\hat{\pi}$ is *Blackwell-optimal* [4].

The subroutine isOptimal, which checks whether a policy $\hat{\pi}$ is optimal for a given discount factor γ, proceeds as follows:

1. Compute the value function

$$\mathbf{v} = (I - \gamma P_{\hat{\pi}})^{-1} \mathbf{q}_{\hat{\pi}}.$$

2. For each action $a \in A$, check whether

$$(I - \gamma P_a)\,\mathbf{v} \;\geq\; \mathbf{q}_a.$$

The computational complexity of the isOptimal procedure is $O(N^3 + N^2 M)$ for an MDP with N states and M actions. Algorithm 1 invokes isOptimal at most $O(N^2 M)$ times in the worst case, and calls computeIndifferencePoint at most $O(NM)$ times. Consequently, the overall time complexity of Algorithm 1 is

$$O\Big(NM \cdot \mathsf{Poly}(N) \;+\; N^5 M \;+\; N^3 M^2\Big),$$

where $\mathsf{Poly}(d)$ denotes the cost of isolating the roots of a univariate polynomial of degree d.

Theorem 2
For a given MDP $\mathcal{M}$ and policy $\hat{\pi}$, Algorithm 1 returns the set $\mathcal{I}$ of all discount-factor intervals for which $\hat{\pi}$ is an optimal policy.

Algorithm 1: Compute discount factor ranges for which $\hat{\pi}$ is optimal

Data: MDP $\mathcal{M}$ and policy $\hat{\pi}$.
Result: Union of intervals $\mathcal{I} = \bigcup_j [\gamma_l^{(j)}, \gamma_u^{(j)}]$ for which $\hat{\pi}$ is optimal.

1 $\Gamma \leftarrow \{0\}$
2 **for** $\pi_n \in H_1(\hat{\pi})$ **do**
3 $\quad R_n \leftarrow$ computeIndifferencePoint$(\hat{\pi}, \pi_n)$
$\quad$ /* $R_n \subseteq [0,1)$ */
4 $\quad$ **for** $\gamma_n \in R_n$ **do**
5 $\quad\quad$ **if** isOptimal$(\hat{\pi}, \gamma_n)$ **then append** γ_n to Γ
6 $\quad$ **end**
7 **end**
8 **sort** Γ in ascending order
9 $\mathcal{I} \leftarrow \emptyset$
10 **for** $i = 1, \ldots, |\Gamma| - 1$ **do**
11 $\quad \gamma_i \leftarrow \Gamma[i]$,
12 $\quad \gamma_{i+1} \leftarrow \Gamma[i+1]$
13 $\quad \gamma_m \leftarrow (\gamma_i + \gamma_{i+1})/2$
14 $\quad$ **if** isOptimal$(\hat{\pi}, \gamma_m)$ **then** $\mathcal{I} \leftarrow \mathcal{I} \cup \{[\gamma_i, \gamma_{i+1}]\}$
15 **end**
16 $\gamma_{\max} \leftarrow \Gamma[|\Gamma|]$ /* Largest element of Γ */
17 $\gamma_b \leftarrow (\gamma_{\max} + 1)/2$ /* Test for Blackwell optimality */
18 **if** isOptimal$(\hat{\pi}, \gamma_b)$ **then** $\mathcal{I} \leftarrow \mathcal{I} \cup \{[\gamma_{\max}, 1)\}$
19 **return** $\mathcal{I}$

Proof
Let $[\ell, u]$, with $0 \le \ell \le u < 1$, be any maximal interval over which $\hat{\pi}$ is optimal for all discount factors. By the results of Sect. 3.1.1, the endpoints ℓ and u are indifference points between $\hat{\pi}$ and some policy in $H_1(\hat{\pi})$. Therefore, both ℓ and u appear in the list of indifference points Γ.

It is possible that there exist additional indifference points $\ell < \ell_1 < \ell_2 < \cdots < \ell_k < u$ within the interval $[\ell, u]$ identified by the algorithm. This can occur when other policies in $H_1(\hat{\pi})$ intersect the value of $\hat{\pi}$ at a single discount factor. As a result, the loop in Line 10 iterates over the sequence $\ell, \ell_1, \ldots, \ell_k, u$. For each consecutive pair of points, the algorithm checks the midpoint and correctly determines that $\hat{\pi}$ is optimal there. Consequently, the intervals $[\ell, \ell_1], \ldots, [\ell_k, u]$ are all included in $\mathcal{I}$. An analogous argument applies when $\hat{\pi}$ is optimal over an interval of the form $[0, u]$ or $[\ell, 1)$. □

4.1 Approach using root isolation

Thus far, Algorithm 1 assumes that the roots of the relevant polynomials can be computed exactly. This requires working with algebraic numbers, and exact root computation is intractable in general for polynomials of degree greater than five. We therefore modify Algorithm 1 to operate using polynomial root isolation.

Given a polynomial $p(\gamma)$ of degree N and a tolerance parameter δ, a root isolation procedure returns a collection of approximate roots $\{\gamma_1, \ldots, \gamma_k\}$ such that each interval $[\gamma_i - \delta, \gamma_i + \delta]$ contains exactly one real root of $p(\gamma)$. We assume that δ is chosen small enough so that any two distinct roots of $p(\gamma)$ are separated by at least 2δ.

Such a procedure can be implemented using algorithms in the spirit of Collins and Akritas [10], which rely on the Descartes rule of signs to determine whether an interval contains zero, one, or multiple real roots. The method recursively bisects intervals containing roots until the interval width falls below a desired threshold. These techniques have seen significant refinement in recent years. In particular, the ANewDsc algorithm of Sagralof and Mehlhorn combines Descartes-based bisection with Newton iteration, achieving a running time of $\tilde{O}\big(N(N^2 + N\tau + \log(1/\delta))\big)$, where τ bounds the bit-length of the polynomial coefficients and $\tilde{O}$ suppresses polylogarithmic factors [31].

Suppose that computeIndifferencePoints, invoked by Algorithm 1, returns intervals of the form $(\gamma - \delta, \gamma + \delta)$, each guaranteed to contain a single indifference point. We modify the algorithm as follows. In Line 5, we test the optimality of $\hat{\pi}$ at both endpoints $\gamma - \delta$ and $\gamma + \delta$, and insert into Γ those endpoints for which $\hat{\pi}$ is optimal. Assume that the exact version of Algorithm 1 identifies a set of indifference points Γ. Let δ be such that

$$2\delta < \min_{\substack{\gamma_1, \gamma_2 \in \Gamma \\ \gamma_1 \neq \gamma_2}} |\gamma_2 - \gamma_1|,$$

i.e., no two indifference points are closer than 2δ.

Theorem 3
Let $[\ell, u]$ be a maximal interval with $\ell < u$ over which $\hat{\pi}$ is optimal. Then the output $\mathcal{I}$ produced by Algorithm 1, modified to use interval-based root isolation with width δ, contains the interval $[\ell + \delta, u - \delta]$.

Proof
Since $[\ell, u]$ is maximal, the endpoints ℓ and u are indifference points. By the separation assumption, we have $u - \ell > 2\delta$. Consequently, the values $\ell + \delta$ and $u - \delta$ are inserted into the set Γ by the modified algorithm.

Now suppose there exist additional indifference points $\ell < \ell_1 < \ell_2 < \cdots < \ell_k < u$. By the same separation argument, $\hat{\pi}$ is found to be optimal at each $\ell_j \pm \delta$ for $j = 1, \ldots, k$. As a result, the intervals

$$[\ell + \delta,\ \ell_1 - \delta],\ [\ell_1 - \delta,\ \ell_1 + \delta],\ \ldots,\ [\ell_k + \delta,\ u - \delta]$$

are all included in $\mathcal{I}$. Their union contains the interval $[\ell + \delta, u - \delta]$, completing the proof. □

5 Active elicitation

Building on the results of Sect. 4, we now move from *passive inference* to *active elicitation*. In the passive setting, we showed how an agent's discount factor can be inferred by observing its optimal policy across multiple finite-state MDPs, with successive environments gradually refining the feasible discount-factor range. In practice, however, many such observations are uninformative and leave the inferred interval unchanged.

We therefore study an active variant of the problem. Rather than relying on environments that vary autonomously, we fix the transition dynamics and strategically design reward functions so that each interaction is guaranteed to reduce uncertainty about the agent's discount factor. This controlled reward shaping eliminates uninformative environments and ensures that every round provably narrows the admissible range. Together, this section contrasts passive observation with active intervention, highlighting how targeted reward design enables more efficient and reliable discount-factor identification.

Definition 2 (Reward Tuning)
Let $\mathcal{M} = (S, A, (P_a)_{a\in A}, (R_a)_{a\in A})$ be an MDP as defined in Definition 1. The *reward-tuned MDP* obtained by updating the reward at a state–action pair (t, b) to a value $r' \in \mathbb{R}$ is denoted by $\mathcal{M}[R(t, b) := r']$ and is the MDP $\mathcal{M}' = (S, A, (P_a)_{a\in A}, (R'_a)_{a\in A})$, where

$$R'_a(s, s') = \begin{cases} r' & \text{if } a = b \text{ and } s = t, \\ R_a(s, s') & \text{otherwise.} \end{cases}$$

Intuitively, $\mathcal{M}[R(t, b) := r']$ denotes the MDP obtained from $\mathcal{M}$ by modifying the reward associated with the state–action pair (t, b) to r', while preserving all transition probabilities and other reward values. Reward tuning is a core primitive in our *elicitation game* framework, defined next.

Definition 3 (Elicitation Games)
An *elicitation game framework* is a J-stage interaction between agents $\mathcal{P}_1$ and $\mathcal{P}_2$ initialized with an MDP $\mathcal{M}^0$.

Agent $\mathcal{P}_2$ has a fixed discount factor, known to themselves and unknown to agent $\mathcal{P}_1$, and acts optimally with respect to this discount factor at every stage. At stage 0, agent $\mathcal{P}_2$ adopts an optimal policy π^0 for $\mathcal{M}^0$. For each subsequent stage $j \in \{1, \ldots, J\}$, the interaction proceeds as follows:

1. Agent $\mathcal{P}_1$ selects a state–action pair (s_j, a_j) and a reward value r_j. A new MDP $\mathcal{M}^j$ is constructed from $\mathcal{M}^{j-1}$ via reward tuning as $\mathcal{M}^j = \mathcal{M}^{j-1}[R(s_j, a_j) := r_j]$.
2. Agent $\mathcal{P}_2$ responds by adopting an optimal policy π^j for $\mathcal{M}^j$.

Agent $\mathcal{P}_2$ seeks to maximize their discounted sum of rewards at each stage, while agent $\mathcal{P}_1$ chooses the sequence of reward-tuning actions $\{(s_j, a_j, r_j)\}_{j=1}^J$ to maximize the information gained about $\mathcal{P}_2$'s discount factor from the observed policies $\{\pi^j\}_{j=0}^J$.

Solving the elicitation game reduces to repeatedly solving a one-step optimization problem for agent $\mathcal{P}_1$. At each stage, given the current MDP $\mathcal{M}$ and the observed policy π, agent $\mathcal{P}_1$ selects a reward-tuning action that determines the next MDP so as to maximally refine the estimate of $\mathcal{P}_2$'s discount factor. The goal is to induce a policy whose optimality structure yields the greatest possible information about the underlying discount factor. This one-step decision problem is formalized below.

Problem 3 (Active Discount Factor Elicitation)
Given an MDP $\mathcal{M}$ and an observed optimal policy π, design a tuned MDP $\mathcal{M}'$ that minimizes the maximum length of the discount-factor intervals consistent with the policy adopted by agent $\mathcal{P}_2$ at the next stage.

Solution outline. Assume that policy π is observed on $\mathcal{M}$, with the widest optimality region $(\gamma_l, \gamma_u]$, bounded by two distinct neighboring optimal policies π_l and π_u, as illustrated in Fig. 5. If we construct a tuned MDP $\mathcal{M}'$ such that the new indifference points satisfy $\gamma'_l = \gamma_l + L/3$ and $\gamma'_u = \gamma_u - L/3$, where $L = \gamma_u - \gamma_l$, then the next observed policy must be one of π_l, π, or π_u. Consequently, regardless of the observation, at least two-thirds of the previous interval is eliminated in the next round.

Relocating indifference points. Now, we discuss how to tune the rewards in MDP $\mathcal{M}$ such that the indifference points in the resulting MDP $\mathcal{M}'$ are at a desired location. We first recall some of the results from Sect. 3.1.1. For simplicity, let us consider an optimality interval $[\gamma_l, \gamma_u]$ for policy π, with the neighboring policies π_l and π_u that their values intersect with $\mathbf{v}_\pi$ at the boundary points. The boundary points γ_l and γ_u are zeros of rational functions

$$\begin{aligned} F_l(\gamma_l) &= \Delta q_l + \gamma_l \Delta P_l (I - \gamma_l P_\pi)^{-1} \mathbf{q}_\pi = 0, \\ F_u(\gamma_u) &= \Delta q_u + \gamma_u \Delta P_u (I - \gamma_u P_\pi)^{-1} \mathbf{q}_\pi = 0. \end{aligned} \tag{6}$$

where Δq_l and ΔP_l are the difference between expected immediate rewards, and probability transition matrix for π and π_l. Similarly Δq_u and ΔP_u are defined for π and π_u.

We wish to change the reward values assigned to specific state, action pairs to ensure that the function $F_l(\gamma)$ has a root at a desired point $\gamma'_l > \gamma_l$. Likewise, we wish to modify the rewards to have $F_u(\gamma)$ have a root at $\gamma'_u < \gamma_u$ (also $\gamma'_u > \gamma'_l$). This ensures that the interval containing the actual discount factor $[\gamma_l, \gamma_u]$ is narrowed down to one of $[\gamma_l, \gamma'_l]$, $[\gamma'_l, \gamma'_u]$ or $[\gamma'_u, \gamma_u]$.

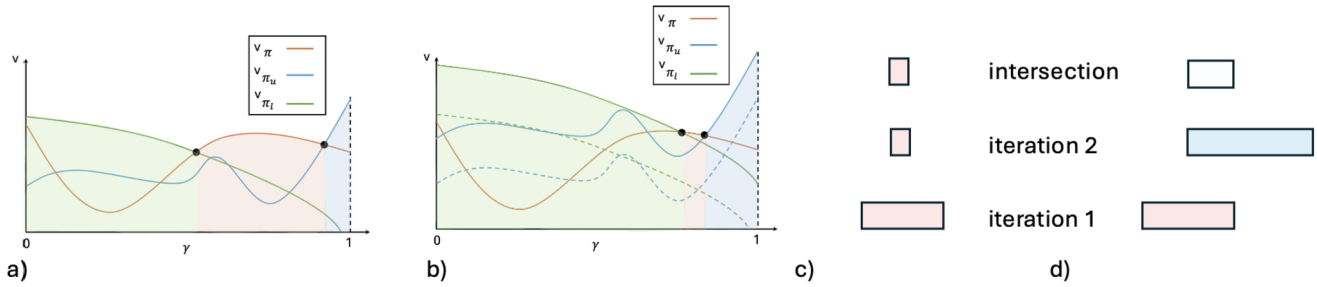

Fig. 5 A schematic representation of reward tuning process. a) The highlighted region in red indicates the optimality region for policy π on $\mathcal{M}$. b)The reward tuning process constructs $\mathcal{M}'$ by changing two reward values in $\mathcal{M}$. In $\mathcal{M}'$ the previous highlighted region in red is divided into 3 sub-intervals that each belong to the optimality region for a different policy (green for π_l, and blue for π_u). c) Assuming the observed policy on $\mathcal{M}'$ is π, the range of discount factor is reduced at every interval. This is true regardless of the policy observed. d) Even if we observe π_l or π_u in a round, or the obtained range of discount factor for a round is increased, the intersection of discount factor range is always decreasing

Another key point regarding the choice of state, action pairs is that the neighboring optimal policies only differ in one state. Therefore, changing the reward value for this state will change the location of zeros of F_l and F_u while keeping the value of policy π the same.

5.1 Reward tuning

Given a desired γ', the value of Δq_l for which γ' is a root of F can be computed by solving the following equations

$$\Delta q_l^{j+1} = -\gamma_l' + \Delta P_l (I - \gamma_l' P_\pi)^{-1} \mathbf{q}_{\hat{\pi}}, \tag{7}$$

and

$$\Delta q_u^{j+1} = -\gamma_u' \Delta P_u (I - \gamma_u' P_\pi)^{-1} \mathbf{q}_{\hat{\pi}}. \tag{8}$$

Now, let $\mathcal{L}_{\max}^j = \max |[\gamma_l, \gamma_u]|$ denote the maximum length of discount factor intervals computed at stage j. To minimize the length of $\mathcal{L}_{j+1}^{\max}$, we want to remove maximum possible length from $\mathcal{L}_j^{\max}$. Assuming we cannot control the policy that the opponent chooses next, we aim to divide $\mathcal{L}_{j+1}^{\max}$ into 3 sub-intervals where the optimal policy for each one is different. As a result, the next observation will rule out 2 of such sub-intervals. This is achieved if the neighboring optimal policies at stage $j+1$ intersect with π_{j+1} at $\gamma_l' = \gamma_l + \frac{\mathcal{L}_j^{\max}}{3}$, and $\gamma_u' = \gamma_u - \frac{\mathcal{L}_j^{\max}}{3}$.

Assume π differs from π_l at state l, and from π_u at state u. The MDP formed by reward modification at each stage j is represented as:

$$\mathcal{M}^j = \mathcal{M}^{j-1}[R(s_l, a_l) := r_l^j][R(s_u, a_u) := r_u^j] \tag{9}$$

where $r_l = \frac{\Delta q_l}{P_{a_l}(l, j)}$ and $r_u = \frac{\Delta q_u}{P_{a_u}(u, j)}$. This approach is explained schematically in Fig. 5.

Number of neighboring optimal policies. In Sect. 5, we studied the case where we have at least two distinct neighboring optimal policies π_u and π_l for $\hat{\pi}$, and their respective intersection points γ_u and γ_l. However, there could be a case where $\pi_u = \pi_l$. In this case, we can move one boundary point to $\gamma_l' = \gamma_l + \frac{\mathcal{L}_j^{\max}}{2}$, and $\gamma_u' = \gamma_u - \frac{\mathcal{L}_j^{\max}}{2}$ in Equations (7) and (8).

6 Experiments

This section presents experimental results for both the passive inference and active elicitation methods (Sects. 4 and 5). We show that, in dynamic environments, an agent's discount factor can be inferred from repeated observations of its policy.

Section 6.1 examines passive inference in settings where the environment evolves and the agent adapts its policy accordingly. In such cases, successive observations may refine the inferred discount-factor range, but this refinement depends on how the environment changes and is not guaranteed.

To address this limitation, Sect. 6.2 introduces active elicitation, in which the observer strategically tunes rewards to induce informative policy responses. Finally, Sect. 6.3 presents an application to targeted exploration, framing discount factor elicitation as an information–theoretic exploration–exploitation tradeoff.

6.1 Iterative refinement of discount factor estimates

Thus far, we have shown that a single policy can be optimal for a range of discount factors, making it impossible to identify a unique discount factor from a single observation. However, when the underlying MDP changes over time and

the agent adapts its policy accordingly, repeated observations can narrow the set of feasible discount factors.

We illustrate this setting through a simple case study in which an agent interacts with an environment whose rewards and transition probabilities evolve over time. We assume that, after each environmental change, the agent learns an optimal policy using an RL procedure that converges faster than the rate at which the environment changes, and that the resulting policy is observable.

Formally, we consider a repeated decision-making process in which, at each stage, the agent faces an MDP with fixed state and action spaces but varying transition probabilities and rewards. The agent acts optimally with respect to a fixed but unknown discount factor throughout the experiment. At each stage, we use Algorithm 1 to compute the interval of discount factors consistent with the observed policy, and intersect these intervals across stages to refine the estimate. The resulting range is then used to predict the agent's policy at the next stage.

Figure 6 shows an illustration of this over four different MDPs with varying number of states. For each instance, we move through 20 different stages that consist of varying transition probabilities and rewards. The discount factor is unknown but is assumed to be fixed for all stages. Algorithm 1 was used with a numerical root finding procedure based on Newton-Raphson method to find ranges of the discount factor. In each case, we identified a single range of discount factors for which the given policy was optimal. In all simulations, the agent's actual discount factor was fixed to 0.9. The estimated range for every simulation is reported in Fig. 6. We note that our approach rapidly converges on the underlying discount factor by exploiting the information on the discount factor ranges given from varying environments.

6.2 Accelerating discount factor convergence

The passive approach to discount factor inference in varying environments relies on the assumption that, when an agent's discount factor is fixed, intersecting the discount-factor ranges obtained across iterations will eventually yield a more accurate estimate. However, when rewards are assigned randomly, changes in the environment can alter the structure of optimal policy regions. As a result, the interval returned by Algorithm 1 at iteration $j+1$ is not guaranteed to be contained within the interval obtained at iteration j.

In contrast, the reward tuning mechanism introduced in Sect. 5.1 enables systematic control over how optimal policy regions evolve, significantly reducing the number of iterations required to localize the agent's discount factor within a desired precision. Figure 7 illustrates this effect in a two-player elicitation game, where reward tuning is applied at each iteration until the length of the remaining discount-factor interval falls below 0.1.

If the observed policy is optimal over the entire range $0 < \gamma < 1$, then it has no neighboring optimal policies. In this case, we select a reference value $\gamma = 0.5$, choose an arbitrary policy π that differs from $\hat{\pi}$, and compute the reward adjustment Δq such that the values of π and $\hat{\pi}$ intersect at $\gamma = 0.5$. The subsequent policy observation $\hat{\pi}_{n+1}$ then eliminates either the lower or the upper half of the interval, ensuring progress in narrowing the feasible discount-factor range.

To evaluate the effectiveness of the proposed reward tuning method, we conduct experiments analogous to those in Fig. 6 within a finite-state MDP. Unlike the passive setting, where the environment and rewards evolve autonomously, here we leverage the elicitation framework introduced in Sect. 5 to explicitly design the reward function at each round in order to elicit the agent's discount factor.

In these experiments, state transition probabilities remain fixed across rounds. The elicitation game begins by assigning each state–action pair a reward drawn uniformly from the range $[-5, 5]$. After observing the agent's policy in the initial round, we apply Algorithm 1 to compute the discount-factor interval(s) consistent with the observed behavior. In subsequent rounds, we apply the reward tuning described in Sect. 5, modifying the rewards of two selected state–action pairs so that the policy induced in the next MDP refines the inferred interval. This process is repeated until the maximum length of the remaining discount-factor interval falls below a threshold of 0.1.

Figure 7 reports the results of these experiments for MDPs with varying numbers of states and actions. Compared to passive policy observations in dynamic environments (Fig. 6), active reward tuning achieves substantially faster refinement of the discount-factor range, demonstrating the effectiveness of the proposed elicitation mechanism.

6.3 Targeted exploration in strategic games

We frame active discount factor identification as an information acquisition problem in a strategic environment. One agent seeks to predict an opponent's decision-making policy while retaining the option to incur a cost in order to acquire information about the opponent's internal decision parameters. In many settings, such information has direct economic value. A canonical example is personalized recommendation, where a user's behavior can be modeled as an MDP with two actions at each state. The user's policy determines the probability of selecting an item, and a recommender system derives utility when its prediction is correct.

This perspective naturally gives rise to a sequential exploration-exploitation tradeoff. When the observer's uncertainty about the opponent's discount factor is sufficiently small, the observer benefits from exploiting its current estimate to maximize immediate payoff. When uncertainty is

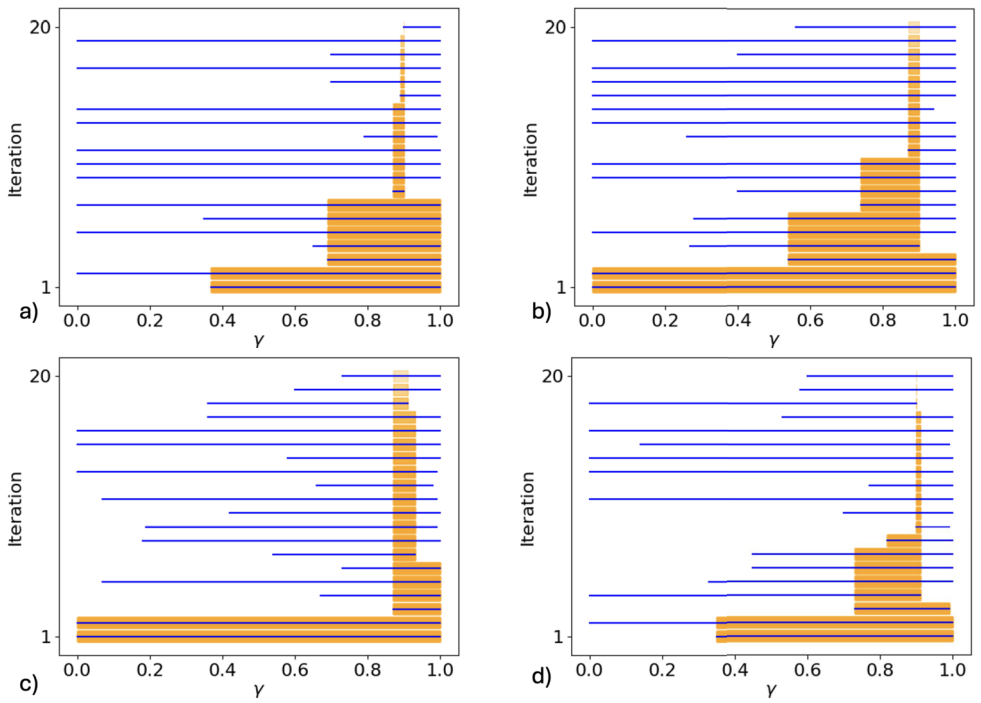

Fig. 6 Simulation results of implementing Algorithm 1 in a MDP with (a) 10 states and 3 actions results in discount factor range $(0.90, 0.90)$, (b) 20 states and 5 actions results $(0.87, 0.90)$, (c) 50 states and 7 actions results $(0.87, 0.91)$, and (d) 100 states and 10 actions result $(0.90, 0.90)$. The discount factor used by the agent in all experiments was 0.9

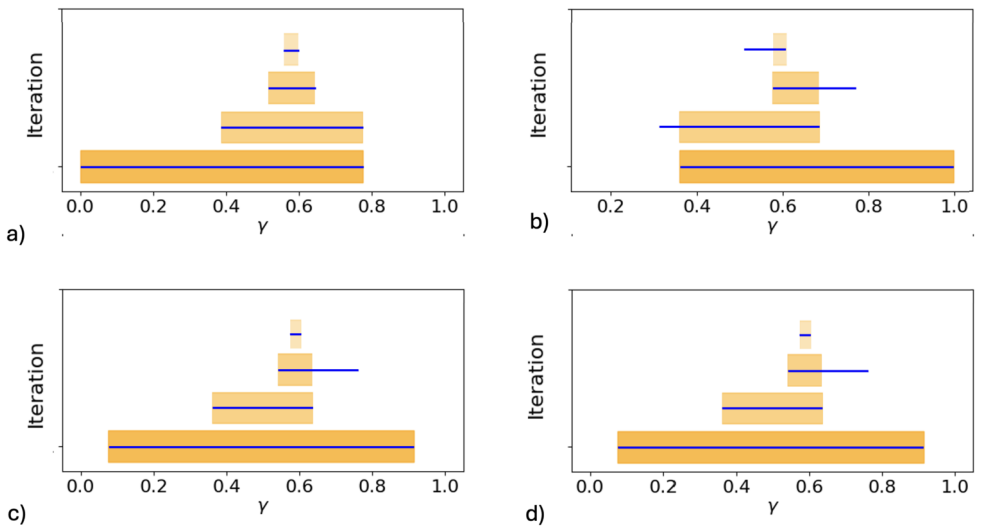

Fig. 7 Simulation results of implementing the reward tuning methodology in an elicitation game with (a) 10 states and 3 actions, (b) 20 states and 5 actions, (c) 50 states and 7 actions, and (d) 100 states and 10 actions. The discount factor used by $\mathcal{P}_2$ in all experiments was 0.6, and the iterations were performed until the length of possible discount factor range was less than 0.1

large, the observer may instead choose to explore by perturbing the reward function and observing the opponent's response. Although such perturbations may reduce short-term performance, they generate information that improves future predictions.

We formalize this tradeoff through a repeated two-player game. The interaction is modeled as a finite MDP with $|S| = 100$ states and $|A| = 30$ actions, with fixed and known transition probabilities. At each round, non-negative rewards in $[0, 1]$ are assigned to state–action pairs, and the game is repeated for $J = 30$ rounds. In each round, $\mathcal{P}_1$ predicts the policy adopted by $\mathcal{P}_2$; the round is won if the prediction is correct and lost otherwise.

If $\mathcal{P}_1$ wins a round, it receives a payoff equal to the value of $\mathcal{P}_2$'s policy under the reward function used in that round; otherwise, it receives no reward. While $\mathcal{P}_1$ has full knowledge of the transition dynamics and reward assignments, it does not know $\mathcal{P}_2$'s discount factor. Instead, $\mathcal{P}_1$ maintains an uncertainty interval over the set of plausible discount factors and updates this interval over time based on observed behavior.

In particular, $\mathcal{P}_1$ may actively modify the reward before a round begins. With probability equal to the length of the current discount-factor uncertainty interval, $\mathcal{P}_1$ perturbs the rewards assigned to selected state–action pairs. This modification incurs a cost proportional to the magnitude of the

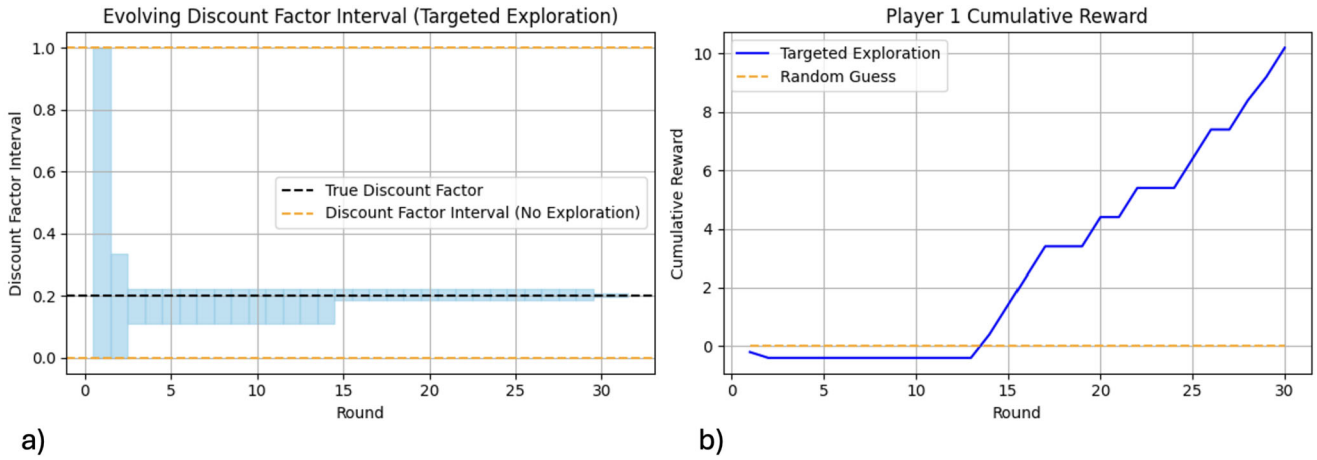

Fig. 8 Plots of a) the uncertainty interval ($\mathcal{P}_1$'s knowledge of $\mathcal{P}_2$'s discount factor) at every round of the game, and, b) cumulative reward of $\mathcal{P}_1$ at every round of the game. In "No Exploration" scenario, $\mathcal{P}_1$ predicts $\mathcal{P}_2$'s policy by observing the game MDP, and selecting an arbitrary policy from the set of optimal policies belong to each discount factor region. In "Targeted Exploration" scenario, $\mathcal{P}_1$ can pay to adjust the rewards for 2 state-action pairs in the game MDP. In each round, the probability of $\mathcal{P}_1$ choosing to "explore" is equal to the length of uncertainty interval on $\mathcal{P}_2$'s discount factor. Therefore, as the game progresses, the probability of exploration decreases.

perturbation, which is deducted from $\mathcal{P}_1$'s cumulative reward regardless of the round's outcome. The purpose of this intervention is not immediate payoff, but information. By shaping rewards, $\mathcal{P}_1$ influences which policies are optimal for different discount factors, thereby inducing observations that shrink the uncertainty interval.

This construction makes the informational role of reward design explicit. Each round becomes a decision between exploiting the current estimate of the opponent's policy and investing in information that improves future predictions. The setting thus instantiates discount-factor elicitation as a sequential decision problem in which short-term reward may be sacrificed to enable long-term policy identification.

Figure 8(a) illustrates the knowledge of $\mathcal{P}_1$ under pure exploitation, without actively perturbing rewards to gain information about $\mathcal{P}_2$'s discount factor. Figure 8(b) plots the cumulative reward obtained by $\mathcal{P}_1$ when employing the methodology described in Sect. 5 to actively modify rewards and narrow the range of possible discount factors used by $\mathcal{P}_2$. In this setting, $\mathcal{P}_1$ chooses to explore—by perturbing rewards and predicting $\mathcal{P}_2$'s policy—with probability equal to the length of the current discount-factor uncertainty interval at each round.

Comparing the accumulated reward after $J = 30$ rounds reveals a significant advantage of our *targeted exploration* strategy. While the performance difference is modest in the initial rounds, it becomes increasingly pronounced as the game progresses and $\mathcal{P}_1$'s estimate of $\mathcal{P}_2$'s discount factor is refined, leading to more accurate policy predictions.

This performance gap grows with the size of the state and action spaces. In small MDPs, a random guess has a nontrivial probability of being correct; for example, with two actions, a uniformly random choice succeeds with probability 0.5. As the action space expands, this probability decreases rapidly, making random guessing increasingly ineffective and amplifying the benefit of targeted exploration. Finally, unlike traditional exploration-exploitation strategies, the proposed approach ensures that even when $\mathcal{P}_1$ loses a round, the resulting information gain about $\mathcal{P}_2$'s discount factor can be exploited in subsequent rounds to improve future predictions.

7 Conclusion

This paper studies the problem of inferring an agent's discount factor from observed behavior, represented as a policy on a finite MDP. We leverage the notion of optimal policy regions to characterize intervals of discount factors over which optimal policies remain invariant. The boundaries of these regions correspond to indifference points, where two neighboring optimal policies attain equal value. Exploiting this structure, we show that the range of discount factors consistent with an observed policy can be identified by locating the points at which its value intersects with that of neighboring optimal policies.

We develop numerical algorithms to compute these boundary points under varying assumptions on policy optimality, analyze their computational complexity, and demonstrate their effectiveness through a series of case studies. Building on these results, we extend the framework to active discount factor elicitation via reward tuning. To this end, we introduce an elicitation game formulation, modeling the interaction as a finite-stage repeated game in which an ego

agent strategically modifies rewards to refine its estimate of an opponent's discount factor. Experimental results show that this active approach substantially reduces the number of iterations required to localize the discount factor compared to passive observation with randomly varying rewards. We further illustrate how discount factor elicitation enables principled solutions to exploration–exploitation tradeoffs and improves policy prediction in dynamic environments.

Several directions remain open for future work. One natural extension is to infer discount factors from partial policies, where actions are observed only for a subset of states. Another important direction is to relax the assumption of perfect rationality and study discount factor inference for agents whose behavior is noisy or systematically suboptimal. Our framework also has potential applications in inverse reinforcement learning and transfer learning, where knowledge of an agent's discount factor can provide valuable information about its preferences in previously unseen environments.

Finally, while this work assumes a constant discount factor, there is growing evidence that discounting may be stateful or history-dependent. For example, agents may alter their effective planning horizon following sequences of successes or failures, and institutions may adjust discounting in response to historical transactions. Inspired by the notion of reward machines, an interesting extension of our framework is the development of discount machines—automata that output discount factors as a function of interaction history. Such models could be learned by combining the techniques introduced here with active or passive automata learning methods.

Acknowledgements This research was supported in part by the National Science Foundation through CAREER Award CCF-2146563. Ashutosh Trivedi is a Royal Society Wolfson Visiting Fellow and gratefully acknowledges the support of the Wolfson Foundation and the Royal Society.

References

1. Agranov, M., Kim, J., Yariv, L.: Coordination with differential time preferences: Experimental evidence. Tech. Rep., National Bureau of Economic Research (2023)
2. Amit, R., Meir, R., Ciosek, K.: Discount factor as a regularizer in reinforcement learning. In: International Conference on Machine Learning, pp. 269–278. PMLR (2020)
3. Andersen, S., Harrison, G.W., Lau, M.I., Rutström, E.E.: Eliciting risk and time preferences. Econometrica **76**(3), 583–618 (2008)
4. Blackwell, D.: Discrete dynamic programming. Ann. Math. Stat., 719–726 (1962)
5. Boutilier, C.: A pomdp formulation of preference elicitation problems. In: AAAI/IAAI, Edmonton, AB, pp. 239–246 (2002)
6. Cao, Y., McGill, W.L.: Linkit: a ludic elicitation game for eliciting risk perceptions. Risk Anal. **33**(6), 1066–1082 (2013)
7. Chen, B., Takahashi, S.: A folk theorem for repeated games with unequal discounting. Games Econ. Behav. **76**(2), 571–581 (2012)
8. Chen, Y., Kash, I.A., Ruberry, M., Shnayder, V.: Eliciting predictions and recommendations for decision making. ACM Trans. Econ. Comput. **2**(2), 1–27 (2014)
9. Coller, M., Williams, M.B.: Eliciting individual discount rates. Exp. Econ. **2**, 107–127 (1999)
10. Collins, G.E., Akritas, A.G.: Polynomial real root isolation using descarte's rule of signs. In: Proceedings of the Third ACM Symposium on Symbolic and Algebraic Computation, SYMSAC'76, pp. 272–275. Association for Computing Machinery, New York (1976). https://doi.org/10.1145/800205.806346
11. Faddeev, D.K., Faddeeva, V.N., Williams, R.C.: Computational Methods of Linear Algebra (1963)
12. Faltings, B., Radanovic, G.: Game Theory for Data Science: Eliciting Truthful Information. Springer, Berlin (2022)
13. Filar, J., Vrieze, K.: Competitive Markov Decision Processes. Springer, Berlin (2012)
14. Fisher, I.: The theory of interest. New York **43**, 1–19 (1930)
15. François-Lavet, V., Fonteneau, R., Ernst, D.: How to discount deep reinforcement learning: Towards new dynamic strategies (2015). arXiv:1512.02011
16. Giwa, B.H., Lee, C.G.: A marginal log-likelihood approach for the estimation of discount factors of multiple experts in inverse reinforcement learning. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7786–7791 (2021). https://doi.org/10.1109/IROS51168.2021.9636479
17. Gurvich, V., Miltersen, P.B.: On the computational complexity of solving stochastic mean-payoff games. CoRR (2008). arXiv:0812.0486
18. Hu, H., Yang, Y., Zhao, Q., Zhang, C.: On the role of discount factor in offline reinforcement learning. In: Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., Sabato, S. (eds.) Proceedings of the 39th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 162, pp. 9072–9098 (2022). PMLR (17–23 Jul 2022)
19. Jackson, M.O., Yariv, L.: Collective dynamic choice: the necessity of time inconsistency. Am. Econ. J. Microecon. **7**(4), 150–178 (2015)
20. Lehrer, E., Pauzner, A.: Repeated games with differential time preferences. Econometrica **67**(2), 393–412 (1999)
21. Lehrer, E., Solan, E., Solan, O.N.: The value functions of Markov decision processes. Oper. Res. Lett. **44**(5), 587–591 (2016)
22. Littman, M.L., Topcu, U., Fu, J., Isbell, C., Wen, M., MacGlashan, J.: Environment-independent task specifications via gltl (2017). arXiv:1704.04341
23. Miller, N., Resnick, P., Zeckhauser, R.: Eliciting informative feedback: the peer-prediction method. Manag. Sci. **51**(9), 1359–1373 (2005)
24. Mischel, W.: to willpower. The psychology of action: Linking cognition and motivation to behavior, p. 197 (1996)
25. Mischel, W., Ebbesen, E.B., Raskoff Zeiss, A.: Cognitive and attentional mechanisms in delay of gratification. J. Pers. Soc. Psychol. **21**(2), 204 (1972)
26. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing Atari with deep reinforcement learning (2013). arXiv:1312.5602

27. Ng, A.Y., Russell, S., et al.: Algorithms for inverse reinforcement learning. In: Icml, vol. 1, p. 2 (2000)
28. Patek, S.D., Bertsekas, D.P.: Stochastic shortest path games. SIAM J. Control Optim. **37**(3), 804–824 (1999)
29. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley, New York (2014)
30. Roy, M.F., Basu, S., Pollack, R.: Algorithms in Real Algebraic Geometry. Algorithms and Computation in Mathematics, vol. 10 (2006)
31. Sagraloff, M., Mehlhorn, K.: Computing real roots of real polynomials. J. Symb. Comput. **73**, 46–86 (2016). https://www.sciencedirect.com/science/article/pii/S0747717115000292
32. Smallwood, R.D.: Optimum policy regions for Markov processes with discounting. Oper. Res. **14**(4), 658–669 (1966)
33. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (2018)
34. Tasdighi Kalat, S., Sankaranarayanan, S., Trivedi, A.: What is your discount factor? In: International Conference on Quantitative Evaluation of Systems and Formal Modeling and Analysis of Timed Systems, pp. 322–336. Springer, Berlin (2024)
35. Tessler, C.: Deep reinforcement learning works - now what? (2020). https://tesslerc.github.io/posts/drl_works_now_what/
36. Tramèr, F., Shokri, R., San Joaquin, A., Le, H., Jagielski, M., Hong, S., Carlini, N.: Truth serum: poisoning machine learning models to reveal their secrets. In: Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, pp. 2779–2792 (2022)
37. Trautmann, S.T., van de Kuilen, G.: Belief elicitation: a horse race among truth serums. Econ. J. **125**(589), 2116–2135 (2015)
38. Ubfal, D.: How general are time preferences? Eliciting good-specific discount rates. J. Dev. Econ. **118**, 150–170 (2016)
39. Vinyals, O., Babuschkin, I., Czarnecki, W.M., Mathieu, M., Dudzik, A., Chung, J., Choi, D.H., Powell, R., Ewalds, T., Georgiev, P., et al.: Grandmaster level in starcraft ii using multi-agent reinforcement learning. Nature **575**(7782), 350–354 (2019)
40. Wilf, H.S.: Mathematics for the Physical Sciences. Courier Corporation (2013)
41. Zohar, A., Rosenschein, J.S.: Mechanisms for information elicitation. Artif. Intell. **172**(16–17), 1917–1939 (2008)