MDP Planning as Policy Inference

Anonymous authors
Paper under double-blind review

Abstract

We formulate episodic Markov decision process (MDP) planning as Bayesian inference over *policies*. The primary contribution is conceptual: the policy itself is treated as the latent variable, and expected return defines an unnormalized posterior density over policies. This preserves the standard expected-return objective, in contrast to trajectory-centric planning-as-inference formulations that introduce auxiliary optimality variables and to entropy-regularized policy optimization methods that solve a different objective.

In the exact formulation, the posterior over deterministic policies induces what we define here as an *optimal stochastic policy under preference uncertainty*, namely the stochastic policy induced by that posterior. For discrete MDPs with stochastic transitions, we study variational sequential Monte Carlo (VSMC) as one approximate inference method for this posterior, introducing policy consistency under state revisitation and coupled transition randomness across particles.

Experiments on grid worlds, Blackjack, Triangle Tireworld, and Academic Advising examine the consequences of inference over policies and compare its induced behavior with entropy-regularized policy optimization. The results support the view that MDP planning can be naturally cast as Bayesian inference over policies.

1 Introduction

We cast episodic Markov decision process (MDP) planning as Bayesian inference over *policies*. The central claim of the paper is that the natural object of inference in planning is the policy itself: not a trajectory augmented with auxiliary optimality variables, and not a single stochastic policy trained under entropy regularization, but a posterior distribution over policies whose density is determined by expected return. Treating the policy as a latent variable has two benefits: it enables the use of general-purpose inference algorithms for planning, and it makes uncertainty about optimal behavior explicit as posterior dispersion rather than an artifact of approximation or heuristic regularization.

Prior probabilistic formulations of planning and control—including control-as-inference, maximum-entropy reinforcement learning, and active inference—typically modify the classical planning objective to fit a standard latent–observation template, for example by introducing entropy-regularized or evidence-based surrogate criteria (Dayan & Hinton, 1997; Toussaint & Storkey, 2006; Ziebart, 2010; Levine, 2018). In these settings, stochasticity is often a modeling preference or an exploration device, and uncertainty over the solution of the original expected-return problem is not directly interpretable from the inferred policy.

This distinction in the object of inference matters. In trajectory- centric planning-as-inference methods, randomness is attached to latent trajectories and auxiliary optimality variables are introduced to recover a control objective. In entropy-regularized reinforcement learning, stochasticity is built directly into the optimized policy and reflects a modified objective. Here, by contrast, stochasticity arises from posterior uncertainty over deterministic policies, while the underlying objective remains the standard expected-return criterion.

In this work, we propose a Bayesian formulation that preserves the standard MDP objective. We define an unnormalized probability of optimality for each policy that is monotone in its expected return, yielding a posterior distribution whose modes coincide with return-maximizing policies, while posterior dispersion

represents uncertainty over optimal behavior. Acting is performed by sampling from the posterior predictive distribution induced by this policy posterior. This yields what we define in Section **??** as the model-induced optimal stochastic policy under preference uncertainty, through a Thompson-sampling interpretation rather than through entropy regularization.

Once planning is formulated this way, approximate Bayesian inference is needed to work with the resulting posterior. In this paper we use variational sequential Monte Carlo (VSMC) (Naesseth et al., 2018; Maddison et al., 2017) as one approximate inference tool for discrete MDPs with stochastic transitions. The algorithmic development is therefore in service of the policy-inference formulation, rather than the main contribution of the paper. Concretely, our adaptation enforces policy consistency across revisited states and couples transition randomness across particles within a sweep so that particle weights reflect policy differences rather than independent realizations of simulator noise. We further interpret the reward scale as controlling uncertainty over agent preferences: larger rewards induce posterior concentration and near-deterministic behavior, while smaller rewards yield a diffuse posterior and a stochastic control policy that reflects preference uncertainty through the inferred variational posterior.

What this paper is and is not. This paper is primarily about the *object of inference*. Our claim is that MDP planning can be cast naturally as Bayesian inference over *policies* while preserving the standard expected-return semantics. Accordingly, VSMC is used here as one approximate inference method for the resulting posterior; it is not the main conceptual contribution, nor do we present the method as a new entropy-regularized or "soft" planning objective. In contrast to trajectory-centric planning-as-inference methods, we do not introduce optimality variables or fictitious observations, and in contrast to entropy-regularized RL we do not optimize a single stochastic policy under an added entropy term.

Our contributions are:

- A formulation of episodic MDP planning as Bayesian inference over *policies*, in which the policy itself is the latent variable and expected return defines an unnormalized posterior density. In the exact formulation, this posterior induces what we define here as an **optimal stochastic policy under preference uncertainty**, namely, the stochastic policy obtained by marginalizing posterior uncertainty over deterministic policies, without introducing trajectory-level optimality variables or entropy regularization.
- An adaptation of VSMC for inference over deterministic policies in discrete MDPs with stochastic transitions, serving as an approximate inference algorithm for this policy posterior and including policy consistency under revisitation and coupled transition randomness across particles.
- An empirical evaluation of the *induced stochastic control policy* obtained by posterior predictive (Thompson-style) action sampling, and a comparison to discrete Soft Actor-Critic across diverse discrete benchmarks.

2 Background

2.1 Markov Decision Process

We consider episodic MDPs with state space S, action space A, stochastic transition kernel $p(s' \mid s, a)$, reward function $R(s, a, s')$, initial state s_1, and a set of absorbing goal states $G \subseteq S$. A (Markov) policy π maps states to actions; we write either $\pi(s) \in A$ for a deterministic policy or $p_\pi(a \mid s)$ for a stochastic policy. Episodes end upon first reaching a state in G. This absorbing-goal formulation is standard for episodic, goal-directed tasks, where interaction ends upon reaching a terminal state and policies are evaluated by expected return up to termination (Sutton & Barto, 2018; Puterman, 1994). For planning and inference, we use a finite rollout horizon H as an algorithm parameter and evaluate policies on truncated trajectories $\tau_\pi = (s_1, a_1, s_2, \ldots, a_H, s_{H+1})$ generated by iterating, for $t = 1, \ldots, H$,

$$s_{t+1} \leftarrow \textsc{Step}(s_t, a_t), \qquad r_t \leftarrow \textsc{Reward}(s_t, a_t, s_{t+1}), \tag{1}$$

where STEP samples $s_{t+1} \sim p(\cdot \mid s_t, a_t)$. If $s_t \in G$, the process remains in that absorbing state and accrues no further reward, so the truncated rollout continues only for notational convenience. If no goal state is reached by time H, the rollout is simply truncated. Throughout, inference algorithms access the MDP only through this simulator interface.

2.2 Variational Sequential Monte Carlo

Sequential Monte Carlo (SMC) approximates a posterior over latent variables in a sequential model using a population of weighted particles sampled from a proposal distribution. SMC also produces an unbiased estimate $\hat{Z}$ of the evidence (normalizing constant) Z of the observations:

$$\hat{Z} = \prod_{t=1}^{H} \frac{1}{N} \sum_{i=1}^{N} w_{t,i}, \tag{2}$$

where $w_{t,i}$ are the incremental weights of particle i at step t. Variational SMC (VSMC) (Naesseth et al., 2018; Maddison et al., 2017) treats the parametrized proposal q_λ as a variational family and maximizes a stochastic lower bound given by $\mathbb{E}[\log \hat{Z}]$, where the expectation is over the particle system induced by q_λ.

VSMC learns proposal parameters λ by maximizing $\mathbb{E}[\log \hat{Z}]$, where the expectation is over the particle system induced by q_λ. In standard latent-variable settings, the optimal proposal uses future information (e.g., $p(x_{t+1} \mid x_t, y_{t:H})$); VSMC can be viewed as learning an approximation to such conditionals via this objective.

3 Probabilistic Model

Much of the control-as-inference literature introduces auxiliary *optimality* variables or other fictitious observations in order to cast planning into a standard latent–observed graphical model. Here we avoid such augmentation and instead work directly with an unnormalized target distribution over the object of interest—the policy. Posterior inference only requires access to an unnormalized density $\tilde{p}(\pi)$ (up to a multiplicative constant), possibly through an unbiased stochastic estimator.

Since we are interested in inferring a policy, the *policy* π is the latent random variable. The objective of MDP planning is to identify policies that maximize expected return. To align the probabilistic model with this objective, we assign to each policy an unnormalized **probability of optimality** that is monotone in its expected return.

Specifically, we define the unnormalized log probability of a policy as the *expected return* obtained by the agent following the policy over a truncated rollout of length at most H, with expectation over trajectories τ_π distributed according to the dynamics induced by policy π:

$$\log \tilde{p}(\pi) = \mathbb{E}_{\tau_\pi} \sum_{t=1}^{H} R\left(s_t, a_t^{(\pi)}, s_{t+1})\right), \tag{3}$$

where $a_t^{(\pi)} = \pi(s_t)$ and $s_{t+1} = T\left(s_t, a_t^{(\pi)}\right)$ for all $t \in 1 \ldots H$. Because goal states are absorbing, this sum coincides with the episode return when the goal is reached before H, and otherwise uses the rollout truncation horizon as a computational cutoff. This induces a Boltzmann–Gibbs distribution over *policies* (Ziebart et al., 2008; Todorov, 2006; Levine, 2018).

Note that neither actions nor states are treated as Bayesian random variables for which a posterior is sought. Although the policy (if stochastic policies are considered) induces a stochastic action selection rule and the environment induces stochastic state transitions, these are generative rather than inferential sources of randomness: actions and state transitions are sampled forward from their respective distributions rather than conditioned for the purpose of posterior inference. The randomness they induce propagates into the estimation of the unnormalized log probability of the policy, so that $\log \tilde{p}(\pi)$ is available only through noisy

Monte Carlo evaluations—by computing the return of a single truncated rollout:

$$\log \widehat{p}(\pi) = \sum_{t=1}^{H} R\left(s_t, a_t^{(\pi)}, s_{t+1}\right). \tag{4}$$

Stochastic estimate (4) lays the basis for posterior inference of the policy distribution.

3.1 Posterior-Induced Stochastic Policy

The posterior over deterministic policies also induces a stochastic action-selection rule by marginalization. For any state s, define

$$p^\star(a \mid s) = \Pr_{\pi \sim p(\pi)}[\pi(s) = a]. \tag{5}$$

We refer to $p^\star(\cdot \mid s)$ as the **optimal stochastic policy under preference uncertainty**. This is a notion introduced in this paper: it denotes the stochastic policy obtained by marginalizing the posterior over deterministic policies under the model in Eq. (3). Its action probabilities are therefore the posterior probabilities that the corresponding actions are prescribed by posterior-supported deterministic policies at state s. Stochasticity is thus not introduced through entropy regularization; it arises from posterior uncertainty over which deterministic policy is preferred by the model.

4 Inference

We perform posterior inference over **deterministic policies** with a uniform prior.[1] This keeps uncertainty at the level of coherent behaviors. The stochastic control rule is the posterior-induced policy $p^\star(a \mid s)$ defined in Section **??**; in practice, we use the variational approximation $q(a \mid s)$ and sample actions from $q(\cdot \mid s)$ at execution time.

4.1 Algorithm

A natural baseline for posterior inference under the rollout estimator in (4) is structured variational inference (Hoffman et al., 2013), but single-trajectory objectives are often underdispersed and prone to mode collapse. The sequential structure of returns makes variational sequential Monte Carlo (VSMC) a natural alternative: its multi-particle objective aggregates diverse trajectory proposals and yields a tighter, more robust approximation.

In our setting, the unnormalized policy log density is available only through the unbiased Monte Carlo estimator (4); this can be treated as exogenous estimator noise on the target density (analogous to pseudo-marginal inference (Andrieu & Roberts, 2009)) and incorporated directly into VSMC optimization.

We assume a countable state space and a finite action space to enable revisit bookkeeping and categorical action proposals; these assumptions simplify the inference mechanics and do not affect the probabilistic model. For deterministic policy inference, a policy assigns a single action to each visited state, sampled on first visit:

$$\pi(s) \equiv a \mid s \sim \text{Categorical}(\boldsymbol{p}(s)). \tag{6}$$

In our case studies, $\boldsymbol{p}(s)$ is parameterized by a neural network over a factored representation of s.

SMC sweep Two adjustments to the vanilla SMC sweep are required:

1. **Deterministic policy consistency.** For each particle, the action for a state is sampled from the proposal only on the first visit to that state and is reused on all revisits (i.e., the particle memoizes $\pi(s)$). Equivalently, on revisits the proposal and prior assign probability 1 to the previously sampled action and 0 to all others.

[1] The formulation also admits inference over stochastic policies; we focus on deterministic policies to avoid introducing an additional layer of action-level randomness and to keep uncertainty at the policy level.

2. **Coupled transition randomness.** To ensure particle weights reflect policy differences rather than independent realizations of environment noise, transition randomness is shared across particles within a sweep. Specifically, if two particles visit the same state s and take the same action a on the same visit count k, they are forced to transition to the same successor state s'. This can be implemented by lazily sampling and caching $\hat{T}^k_{s,a} \sim T(\cdot \mid s, a)$ once per queried (s, a, k) and reusing it for all particles in the sweep, so that inference proceeds under a shared random realization $\hat{T}$ of the dynamics.[2]

Optimization objective The original VSMC formulation assumes reparameterizable proposals and typically omits score-function terms associated with the non-differentiable *resampling* operation due to their high variance. In finite-action MDPs, however, the categorical proposal over actions is not reparameterizable. Consequently, while the resampling-induced score terms may still be dropped, the score-function contribution from *sampling actions from the proposal* must be retained to obtain meaningful gradients. Using a temporally stratified, variance-reduced learning signal (Schulman et al., 2015), we optimize

$$\mathcal{L} = \log \hat{Z} + \sum_{t=1}^{H} \left(\overline{\log \hat{Z}_t} \cdot \sum_{i=1}^{N} \log q(a_{t,i} \mid s_{t,i}) \right), \tag{7}$$

where $\log \hat{Z}_t$ denotes the contribution of steps $t, \ldots, H$ to $\log \hat{Z}$, and the overline denotes a stop-gradient operation.

With these two modifications, each sweep proceeds as in standard SMC: particles advance under Step, update weights, and resample as needed. Full pseudocode appears in the appendix.

Theorem 1 shows that optimizing the surrogate objective corresponds to stochastic gradient ascent on a well-defined scalar objective, rather than a heuristic update rule.

Theorem 1 (Unbiased gradient estimator)**.** *Let $\hat{Z}(\hat{T}, \mathbf{a})$ denote the SMC normalizing constant estimator produced by one sweep of the procedure above, where $\hat{T}$ is the shared random realization of the transition dynamics induced by the coupled-transition rule, and $\mathbf{a} = \{a_{t,i}\}_{t=1,i=1}^{H,N}$ are the actions sampled from the proposal q_θ. Define*

$$\mathcal{J}(\theta) \;=\; \mathbb{E}_{\hat{T}\sim T,\, \mathbf{a}\sim q_\theta}\big[\log \hat{Z}(\hat{T}, \mathbf{a})\big]. \tag{8}$$

Then the gradient of the surrogate objective in Eq. (6) *is an unbiased estimator of $\nabla_\theta \mathcal{J}(\theta)$.*

Proof sketch. Conditioned on $\hat{T}$, the randomness in $\log \hat{Z}(\hat{T}, \mathbf{a})$ arises only from sampling actions from q_θ, and $\log \hat{Z}$ is a deterministic function of the sampled actions. The score-function identity therefore gives

$$\nabla_\theta \mathbb{E}_{\mathbf{a}}\big[\log \hat{Z}(\hat{T}, \mathbf{a})\big] = \mathbb{E}_{\mathbf{a}}\left[\log \hat{Z}(\hat{T}, \mathbf{a}) \sum_{t,i} \nabla_\theta \log q_\theta(a_{t,i} \mid s_{t,i})\right].$$

Introducing a stop-gradient baseline preserves unbiasedness, and taking expectation over $\hat{T}$ completes the result. The temporally stratified signal $\log \hat{Z}_t$ in Eq. (6) is a standard variance-reduction (Rao–Blackwellization) that leaves the expectation unchanged. □

5 Related Work

This work relates to (i) probabilistic formulations of planning and control, and (ii) entropy-regularized reinforcement learning. Our key distinction is that inference is carried out over *policies* themselves, rather than over trajectory-level auxiliary variables or within a directly optimized parametric stochastic policy.

[2]This is related to common random numbers (Mohamed et al., 2020), but used here for correctness rather than variance reduction.

5.1 Control and Planning as Inference

Casting control and planning as probabilistic inference has a long history, including planning-as-inference in graphical models (Attias, 2003) and subsequent formulations that encode optimality via auxiliary variables or likelihood terms that bias trajectories toward high return (e.g., Botvinick & Toussaint (2012); Toussaint & Storkey (2006)). Active inference likewise casts action selection as approximate Bayesian inference under a generative model with preferences over outcomes. More recently, control-as-inference derivations have shown that entropy-regularized RL objectives arise from variational inference constructions (e.g., Levine (2018)), typically by introducing fictitious observations or optimality variables.

We adopt this perspective but make a different modeling choice: the *policy itself* is the latent random variable, and its expected return defines an unnormalized log density. This yields a posterior over policies directly, without introducing additional observation channels or trajectory-level optimality variables.

5.2 Entropy-Regularized Reinforcement Learning

Entropy-regularized RL and stochastic policy optimization methods, including policy gradients (Sutton et al., 1999), soft Q-learning (Haarnoja et al., 2017), and Soft Actor-Critic (SAC) (Haarnoja et al., 2018; Christodoulou, 2019), optimize objectives of the form $\mathbb{E}[R(\pi)]+\alpha\mathcal{H}(\pi)$. Connections and equivalences among these approaches have been studied extensively (e.g., Schulman et al. (2017)), and from a control-as-inference viewpoint entropy can be interpreted as arising from a variational bound.

While related, our approach differs in two respects. First, stochasticity reflects posterior uncertainty over deterministic policies, rather than entropy within a single learned stochastic policy. Second, this inference formulation makes **variational sequential Monte Carlo (VSMC)** (Naesseth et al., 2018) a natural approximation method, in contrast to the single-trajectory variational objectives that underlie many entropy-regularized RL algorithms.

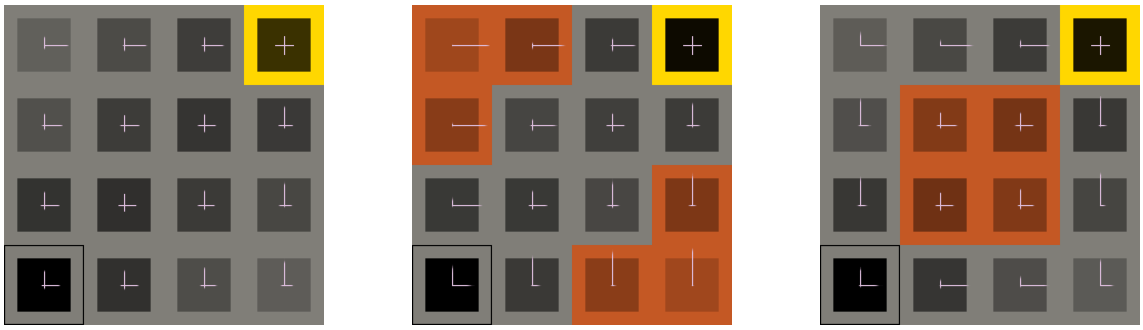
6 Experiments

We evaluate the proposed policy inference framework across a range of domains designed to expose different structural aspects of decision making under uncertainty, and compare it to entropy-regularized policy optimization. Throughout the experiments we contrast inference over distributions of deterministic policies (VSMC) with direct optimization of entropy-regularized stochastic policies (SAC). The experiments are designed to examine (i) qualitative structure of induced behavior in a diagnostic domain, and (ii) differences between deterministic-policy inference and entropy-regularized optimization across increasingly stochastic and complex planning problems. The purpose of this comparison is not to establish uniform superiority in expected reward, since the two methods optimize different objectives, but to examine how these objectives induce different solution structure, uncertainty representations, and action-selection behavior across domains. The experiments are intended to probe the consequences of this modeling choice—inference over policies—rather than to position VSMC as a new state-of-the-art optimizer for entropy-regularized control benchmarks.

We begin by exploring the proposed policy inference framework on a grid world domain. The ease of static visualization and apparent simplicity of grid worlds facilitates qualitative inspection of the induced stochastic policy. We then use the grid worlds and three standard discrete benchmarks from the literature—Blackjack, Triangle Tireworld, and Academic Advising—to compare policy VSMC to discrete Soft Actor-Critic (SAC)[3] (Haarnoja et al., 2018; Christodoulou, 2019), highlighting differences in the resulting policies and their suitability to particular MDP types.

Throughout the experiments, we run VSMC with 10 particles for 50,000 iterations (SMC sweeps), adjusting the initial learning rate between 10^{-5} and $3 \cdot 10^{-4}$ for each domain, with cosine decay to 0.1 of the initial

[3]SAC optimizes an entropy-regularized expected-return objective, whereas policy VSMC maximizes an SMC log-evidence bound $\mathbb{E}[\log \hat{Z}]$, where $\log \hat{Z}$ aggregates particle weights via a log-mean-exp across the sweep. The two objectives are not the same, so differences in performance are informative primarily as differences in induced behavior rather than as a pure leaderboard comparison.



(a) flat world (b) unimodal world (c) multimodal world

Figure 1: Grid worlds: policies and occupancies

rate. These settings are reported to support reproducibility rather than to suggest an "optimal" tuning: in our experiments VSMC is stable across a relatively broad range of learning rates and schedules, the iteration budget is chosen to be large enough to ensure convergence across all domains, and we intentionally keep the number of particles small. This is consistent both with the original VSMC results of Naesseth et al. (2018), which show strong performance with small particle counts, and with the broader particle-variational-inference analysis of Rainforth et al. (2018), which shows that increasing the number of particles can tighten the bound while degrading the signal-to-noise ratio of inference-network gradients. We adapt SAC from CleanRL (Huang et al., 2022) for discrete actions by using MLP critics with two hidden layers, and train for 1,000,000 time steps. In both VSMC and SAC, all networks use two hidden layers of width 64. Each algorithm–domain pair is evaluated over 25 independent training runs, and policies are evaluated using 10,000 trajectories per run. The entropy weight for SAC is kept at 1 except where varied explicitly.

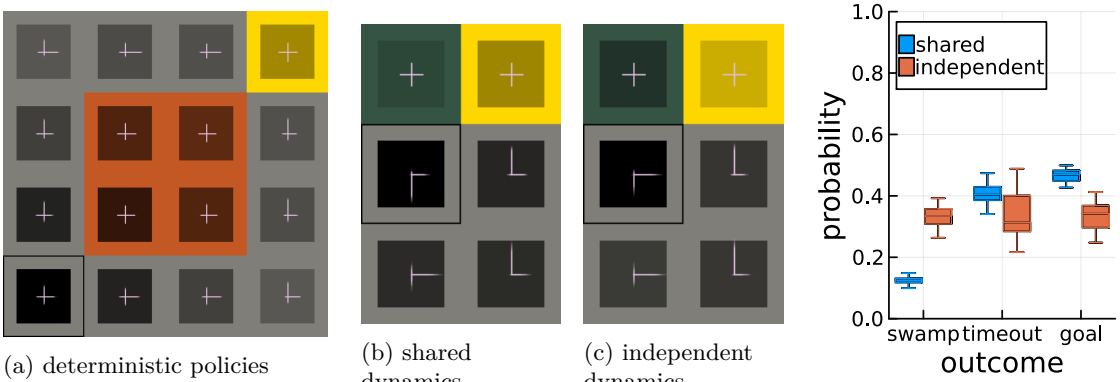
6.1 Grid Worlds

Grid worlds provide a controlled setting in which policy distributions can be visualized directly, allowing qualitative inspection of multimodality, uncertainty, and variability across runs. They therefore serve as a diagnostic domain for understanding the behavior of the inference procedure itself.

In the grid world domain we use in this study, the environment is a rectangular grid. Four actions—Right, Up, Down, and Left—advance the agent to the corresponding adjacent cell. An action that would take the agent out of the grid has no effect. The environment is slippery: upon any action the agent moves, with probability $p_{succ} = 0.8$, in the direction of the action, and otherwise in an adjacent direction. The reward collected by the agent upon each action is determined by the color of the cell to which the agent moves: grey (pavement) — 0, red (gravel) — -1, yellow (goal) — +5, green (swamp) — -5. Each step incurs a cost of -0.1. Yellow and green cells are absorbing: once the agent reaches a yellow or a green cell, no further reward is collected and no action moves the agent out of the cell.

Inferred policies are represented by *policy and occupancy maps* in the figures below (Figure 1–3). A black border denotes the initial cell. The darkness of the middle part of a cell represents the occupancy: the darker a cell, the more trajectories passed through the cell. The white cross with unevenly sized beams in the center of each cell represents the policy distribution in that cell, with the length of each beam denoting the probability of the corresponding direction. The maps are averaged over 25 runs: blurrier crosses mean more variation in the inferred policies across runs.

We begin by applying policy VSMC to three 4×4 grid worlds (Figure 1), using a rollout horizon of 20 steps. In the flat world (Figure 1a), trajectories cover the grid evenly. In the "unimodal" world (Figure 1b) the expensive to travel red regions in the complementary corners of the grid push trajectories closer to the diagonal connecting the starting and the goal cells. In the "multimodal" world (Figure 1c) the red region is in the center of the grid, pushing the trajectories to pass along the edges of the grid. Because a policy *distribution* is inferred (rather than just a single policy with the highest expected return), multiple actions in each cell have non-zero probabilities.



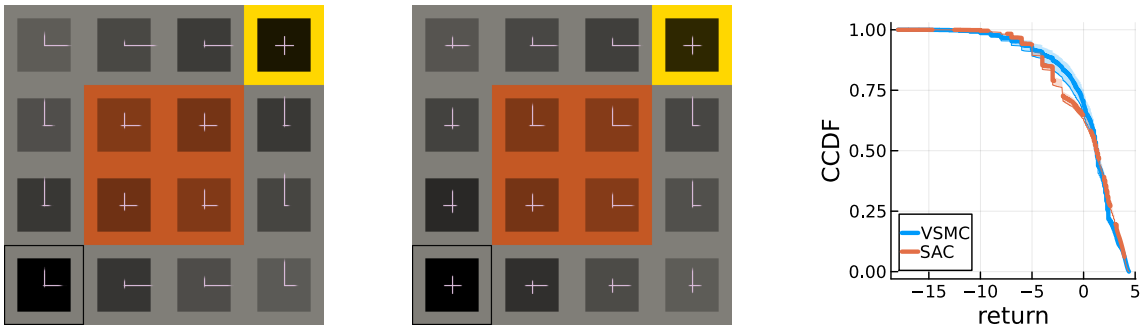
(a) deterministic policies not enforced

(b) shared dynamics

(c) independent dynamics

(d) shared vs. independent

Figure 2: Grid worlds: ablation studies



(a) VSMC

(b) SAC

(c) trajectory return distributions

Figure 3: Grid worlds: VSMC vs. SAC

For policy inference, the SMC sweep of VSMC was modified to a) enforce deterministic policies b) share the same environment dynamics among all particles. Figure 2a shows the effect of dropping the enforcement of deterministic policies: a higher-entropy policy distribution. Figures 2b–2d explore the effect of sharing environment dynamics among all particles on the inferred policies. A small slippery instance with two absorbing cells, a swamp with reward of -5 at (1, 3) and a goal with reward of 5 at (2, 3), and $p_{succ} = 0.5$ is used for the comparison. The starting position is at (1, 2) and the rollout horizon is 10 steps. To avoid slipping into the swamp, the agent should move *down*, to (1, 1), and this is what the policy with shared dynamics mostly suggests. With independent dynamics the agent frequently moves *right*, along the shortest path to the goal.

Finally, we compare VSMC to SAC. Figure 3 shows the policies inferred by VSMC (Figure 3a) and SAC (Figure 3b) and compares their trajectory return distributions (Figure 3c). VSMC and SAC trajectory return distributions are close but different, and the policies differ in particular along the grid edges — SAC, optimizing an entropy-regularized stochastic policy, uses actions directed toward the grid boundaries to increase the entropy. VSMC penalizes such actions strongly because a deterministic policy directing the agent into a grid boundary can escape the current cell only due to environment stochasticity.

6.2 Blackjack

Blackjack is a card game where the goal is to beat the dealer by obtaining cards that sum to closer to 21 (without going over 21) than the dealers cards. Blackjack provides a stochastic control problem with a compact state space and a known optimal policy, making it possible to compare inferred policies against a ground-truth solution while examining the effect of entropy regularization in a domain where randomness arises primarily from the environment rather than exploration. The game starts with the dealer having one

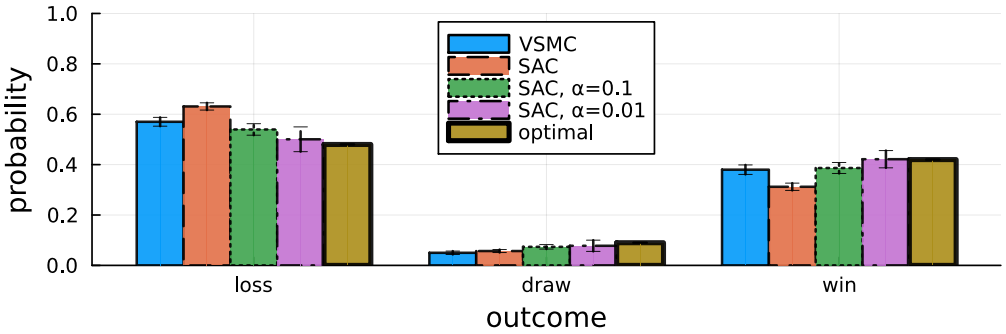


Figure 4: Blackjack: outcome probabilities

face up and one face down card, while the player has two face up cards. All cards are drawn from an infinite deck (i.e. with replacement). The player has the sum of cards held. The player can request additional cards (hit) until they decide to stop (stick) or exceed 21 (bust). After the player sticks, the dealer reveals their facedown card, and draws cards until their sum is 17 or greater. If the dealer goes bust, the player wins. If neither the player nor the dealer busts, the outcome (win, lose, draw) is decided by whose sum is closer to 21.

The variant described in Sutton & Barto (2018, Example 5.1), as implemented in Gymnasium (Towers et al., 2024), is used in this study. VSMC and SAC are compared to each other and to the optimal policy, as a baseline, in Figure 4. The optimal (return-maximizing) player's policy was computed by value iteration and the policy's statistics were estimated by Monte-Carlo evaluation. Neither VSMC nor SAC with the default temperature/entropy weight matches the optimal policy, which is expected because both methods are deliberately regularized. The main observation is not that one method dominates the other in mean return, but that the two objectives induce different operating points: VSMC produces a lower draw probability than SAC at $\alpha = 1$, while matching this behavior in SAC requires substantially weaker entropy regularization ($\alpha = 0.1$ or $\alpha = 0.01$).

These results highlight that, under comparable regularization strength, policy inference and entropy-regularized optimization induce different trade-offs between exploration and outcome variance, even in a domain with a known optimal policy.

6.3 Triangle Tireworld

In Triangle Tireworld (Little & Thiébaux, 2007), the agent travels through a triangular structure of locations connected by directed roads. The agent must arrive from the initial location to the goal. With a fixed probability, the agent gets a flat. Some locations carry a spare tire. If the agent loaded a spare tire prior to getting a flat, it changes the tire and continues the travel. Otherwise, the agent is stuck. The agent gets the reward of 10 for reaching the goal, -10 for getting stuck, and -0.1 for every step. Triangle Tireworld introduces irreversible stochastic events and explicit risk–reward trade-offs. Successful policies must plan for low-probability failures whose consequences cannot be undone. The domain and the instances are based on the version from the 2014 International planning competition Vallati et al. (2015).

With the original rewards, Triangle Tireworld induces a large return gap between "fast but risky" and "safe but slow" behaviors. Under our Bayesian formulation this makes the posterior highly peaked, yielding low mean return and high variance. Scaling rewards down by a factor of five reduces this separation, producing a less concentrated posterior; under this setting VSMC exhibits performance comparable to SAC. Table 1 reports results for instances 1, 4, 7, and 10, which are representative instances spanning the range of difficulty in this domain; the omitted instances follow the same qualitative trend.

Triangle Tireworld highlights a limitation of Bayesian policy inference: unlike classical MDP planning, which is invariant to affine reward scaling, the posterior depends on return magnitudes, so the method works best

Table 1: Triangle Tireworld: policy statistics

#	policy	expected return	success probability
1	VSMC	-0.80 ± 2.38	0.47 ± 0.12
	VSMC $(0.2 \cdot r)$	1.25 ± 0.13	0.58 ± 0.02
	SAC	1.27 ± 0.11	0.58 ± 0.01
4	VSMC	-4.39 ± 1.46	0.30 ± 0.08
	VSMC $(0.2 \cdot r)$	-1.82 ± 0.22	0.44 ± 0.02
	SAC	-1.90 ± 0.17	0.43 ± 0.01
7	VSMC	-2.91 ± 1.14	0.39 ± 0.02
	VSMC $(0.2 \cdot r)$	-1.33 ± 0.16	0.48 ± 0.03
	SAC	-1.35 ± 0.09	0.48 ± 0.01
10	VSMC	-5.79 ± 0.84	0.22 ± 0.06
	VSMC $(0.2 \cdot r)$	-4.89 ± 0.19	0.30 ± 0.02
	SAC	-5.27 ± 0.11	0.27 ± 0.01

Table 2: Academic Advising: policy statistics

#	policy	expected return	0.05 quantile	0.05 tail mean	0.95 quantile	0.95 tail mean
1	VSMC	-65.3 ± 1.3	-141.1 ± 3.6	-175.7 ± 5.0	-20.4 ± 0.5	-18.4 ± 0.4
	SAC	-48.6 ± 0.8	-84.0 ± 2.8	-97.1 ± 2.6	-24.8 ± 1.2	-21.9 ± 0.9
2	VSMC	-98.5 ± 1.8	-184.2 ± 5.2	-222.1 ± 7.1	-45.0 ± 1.2	-39.5 ± 1.0
	SAC	-106.7 ± 3.6	-184.6 ± 8.2	-216.1 ± 11.1	-52.8 ± 1.4	-45.7 ± 1.2
3	VSMC	-86.7 ± 1.1	-174.1 ± 3.0	-207.5 ± 3.4	-32.6 ± 0.2	-28.1 ± 0.5
	SAC	-84.4 ± 2.2	-147.7 ± 4.5	-174.6 ± 5.3	-39.0 ± 1.6	-34.2 ± 0.7

when reward scale meaningfully encodes the strength of preferences/regrets rather than merely ranking policies.

6.4 Academic Advising

Academic Advising models a student choosing which courses to take over a sequence of semesters in order to complete a curriculum. Academic Advising represents a large combinatorial planning problem with long horizons and delayed rewards. The branching action space and stochastic course outcomes create highly multimodal trajectory returns, providing a test of scalability and behavior under complex long-term dependencies. At each semester the action is to enroll in up to a maximum course load of currently eligible courses. Course outcomes are stochastic: each enrolled course is passed with a given probability (otherwise it remains incomplete and can be retaken later), and passed courses persist until curriculum completion, at which point the process enters an absorbing goal state. The reward is specified as step costs: taking a course incurs -1, retaking a previously attempted course incurs -2, and if the program is not yet complete the agent also incurs an additional -5 at every step, encouraging completion before rollout truncation. In the IPC 2014-derived instances we use, the curriculum contains 10–30 courses; easier (odd-numbered) instances restrict the course load to at most 1 course per step, while harder (even-numbered) instances allow up to 2 courses per step.

Without a non-trivial baseline policy or a domain-specific heuristic, SAC and VSMC reliably find policies with a non-negligible probability of completing the program for instances 1–3. For harder instances, either the variation between runs is very high, or the policy / policy distribution concentrates around a random walk that minimizes the per-step cost but does not lead to program completion. Table 2 summarizes policy statistics for instances 1–3. The comparison is most informative at the level of return distributions rather than mean return alone: VSMC and SAC often achieve comparable average performance, but VSMC induces heavier-tailed trajectory return distributions, as manifested by the 0.05 and 0.95 quantiles and their

conditional tail means. This is consistent with the broader claim of the paper that the main difference lies in how the two objectives represent and act under trajectory uncertainty, not in uniform reward dominance.

The Academic Advising results demonstrate that the differences between the two approaches persist in larger combinatorial settings, where long horizons and stochastic outcomes amplify differences in how trajectory uncertainty is represented.

7 Discussion

We view episodic MDP planning as posterior inference over policies, with expected return defining an unnormalized posterior density (Eq. (3)). The resulting posterior concentrates on return-maximizing policies, while its dispersion captures uncertainty about optimal behavior.

Sources of uncertainty. With a posterior over deterministic policies, three sources of uncertainty are disentangled:

(i) **aleatoric** transition randomness, sampled forward and appearing as noise in the Monte Carlo estimate of policy log-probability (Eq. (4));

(ii) **epistemic** uncertainty over optimal behavior, represented by posterior dispersion; and

(iii) **execution-time stochasticity**, obtained by marginalizing over deterministic policies. In exact form, posterior-predictive control is therefore a structured form of Thompson sampling: actions randomize only to the extent that multiple deterministic behaviors remain plausible.

Inference mechanics. Once planning is cast as inference with an intractable target density, the sequential structure of returns makes SMC a natural fit, and VSMC provides a principled variational objective. Policy inference, however, requires two adaptations: enforcing policy consistency under revisitation (by sampling each state's action only on first visit) and coupling transition randomness across particles within a sweep so that weights reflect policy differences rather than independent simulator noise.

Scalability considerations. The main computational scaling of the method remains that of particle inference: runtime grows with rollout horizon and number of particles. The additional bookkeeping introduced here—memoizing a sampled action for each visited state and caching coupled transition realizations within a sweep—adds memory cost proportional to the number of distinct state–action–visit-count queries encountered by the particles. In the discrete domains considered here this overhead is modest and is offset by the need to enforce deterministic-policy consistency under revisitation. However, in very large state spaces or in domains with little state revisitation, the bookkeeping may become a practical limitation. More generally, this bookkeeping should not be conflated with the effect of resampling itself: the original VSMC work (Naesseth et al., 2018) reports that, relative to the IWAE / variational importance sampling without resampling, VSMC can make more effective use of a small particle budget and in some settings reaches comparable accuracy with less computation. Thus, the main scalability issue is not a separate pathology of memoization, but the usual interaction between particle-based inference, rollout length, and state-space size.

Relation to control-as-inference and entropy-regularized RL. Compared to trajectory-centric control-as-inference formulations, our latent variable is the policy itself. Compared to entropy-regularized RL, stochasticity reflects posterior uncertainty over deterministic behaviors rather than an explicit entropy preference.

Empirical takeaways. Across domains, the comparison to SAC should be read primarily as a comparison of objectives and the behaviors they induce, rather than as a claim that policy VSMC is a uniformly stronger optimizer in expected reward. In grid worlds, the variationally approximated induced policy avoids boundary-directed actions that can increase entropy under SAC while reducing goal reachability. In Blackjack, matching VSMC-like behavior requires substantially weaker entropy regularization in SAC. In Triangle Tireworld, the sensitivity of VSMC to reward scale reveals a substantive property of the Bayesian formulation: return

magnitudes affect posterior concentration. In Academic Advising, the main contrast is distributional: both methods can achieve similar mean performance on some instances, but they differ in tail behavior and variability. Taken together, these results support the claim that policy inference and entropy-regularized optimization produce qualitatively different solutions even when their aggregate returns are similar.

Scope and extensions. We focus on discrete state spaces to make revisit bookkeeping and shared transition caching explicit; the policy-inference semantics do not depend on discreteness. In continuous domains, determinism can be enforced via the policy representation or a hashable state abstraction, and shared stochasticity can be implemented via common random numbers or keyed random streams. Finally, strict memoization can be relaxed via *stochastic memoization*, allowing occasional resampling on revisits to reduce the brittleness of committing to an early no-op action.

References

Christophe Andrieu and Gareth O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

Hagai Attias. Planning by probabilistic inference. In Christopher M. Bishop and Brendan J. Frey (eds.), *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, volume R4 of *Proceedings of Machine Learning Research*, pp. 9–16. PMLR, 03–06 Jan 2003.

Matthew Botvinick and Marc Toussaint. Planning as inference. *Trends in Cognitive Sciences*, 16(10):485–488, October 2012.

Petros Christodoulou. Soft actor-critic for discrete action settings, 2019.

Peter Dayan and Geoffrey E Hinton. Using expectation-maximization for reinforcement learning. *Neural Computation*, 1997.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1352–1361, Sydney, NSW, Australia, 06–11 Aug 2017. PMLR.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 10–15 Jul 2018.

Matthew D. Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic variational inference. *J. Mach. Learn. Res.*, 14(1):1303–1347, May 2013.

Shengyi Huang, Rousslan Fernand Julien Dossa, Chang Ye, Jeff Braga, Dipam Chakraborty, Kinal Mehta, and João G.M. Araújo. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022. URL `http://jmlr.org/papers/v23/21-1342.html`.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review, 2018.

Iain Little and Sylvie Thiébaux. Probabilistic planning vs replanning. In *ICAPS 2007 Workshop on The Probabilistic Planning Competition*, pp. 55–62, Providence, RI, 2007.

Chris J. Maddison, Dieterich Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Whye Teh. Filtering variational objectives. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6576–6586, Red Hook, NY, USA, 2017. Curran Associates Inc.

Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte Carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(1), January 2020.

Christian Naesseth, Scott Linderman, Rajesh Ranganath, and David Blei. Variational sequential Monte Carlo. In Amos Storkey and Fernando Perez-Cruz (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 968–977. PMLR, 09–11 Apr 2018.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 1994.

Tom Rainforth, Adam R. Kosiorek, Tuan Anh Le, Chris J. Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4277–4285, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL `https://proceedings.mlr.press/v80/rainforth18b.html`.

John Schulman, Nicolas Heess, Theophane Weber, and Pieter Abbeel. Gradient estimation using stochastic computation graphs. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, pp. 3528–3536, Cambridge, MA, USA, 2015. MIT Press.

John Schulman, Xi Chen, and Pieter Abbeel. Equivalence between policy gradients and soft Q-learning, 2017.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

Richard S. Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of the 13th International Conference on Neural Information Processing Systems*, NIPS'99, pp. 1057–1063, Cambridge, MA, USA, 1999. MIT Press.

Emanuel Todorov. Linearly-solvable Markov decision problems. In B. Schölkopf, J. Platt, and T. Hoffman (eds.), *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.

Marc Toussaint and Amos Storkey. Probabilistic inference for solving discrete and continuous state Markov decision processes. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 945–952, New York, NY, USA, 2006. Association for Computing Machinery.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

Mauro Vallati, Lukas Chrpa, Marek Grześ, Thomas Leo McCluskey, Mark Roberts, Scott Sanner, and Managing Editor. The 2014 international planning competition: Progress and trends. *AI Magazine*, 36: 90–98, Sep. 2015.

Brian D. Ziebart. Modeling purposeful adaptive behavior with the principle of maximum causal entropy. *PhD thesis, CMU*, 2010.

Brian D. Ziebart, Andrew L. Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pp. 1433–1438, 2008.

Algorithmic Details

Algorithm 1 VSMC sweep for deterministic-policy inference. The procedure returns the surrogate objective in Eq. (6). Resampling is shown in the canonical SMC form; the implementation resamples adaptively (ESS threshold of 0.5).

procedure PolicyVSMC(s_1, Step, Reward, H, N, q_θ)
  $M_T \leftarrow \varnothing$
  for $i \leftarrow 1$ to N **do**
    $s^{(i)} \leftarrow s_1$
    $M_A^{(i)} \leftarrow \varnothing$ ▷ action memoization: $s \mapsto a$
    $M_C^{(i)} \leftarrow \varnothing$ ▷ counts: $(s, a) \mapsto k$
  end for
  $\ell_{1:H} \leftarrow 0$ ▷ per-step log mean weight increments
  $g_{1:H} \leftarrow 0$ ▷ per-step proposal log-prob terms (first-visit only)
  for $t \leftarrow 1$ to H **do**
    for $i \leftarrow 1$ to N **do**
      $s \leftarrow s^{(i)}$
      if $s \in \operatorname{dom} M_A^{(i)}$ **then** ▷ revisit: reuse memoized action
        $a \leftarrow M_A^{(i)}(s)$
        $\log p_a \leftarrow 0,\ \log q_a \leftarrow 0$
      else ▷ first visit: sample and memoize
        $a \sim q_\theta(\cdot \mid s),\ \ M_A^{(i)}(s) \leftarrow a$
        $\log p_a \leftarrow -\log |A|,\ \ \log q_a \leftarrow \log q_\theta(a \mid s)$
        $g_t \leftarrow g_t + \log q_a$
      end if
      $k \leftarrow M_C^{(i)}(s, a) + 1$ ▷ default $M_C^{(i)}(s, a) = 0$ if absent
      $M_C^{(i)}(s, a) \leftarrow k$
      if $(s, a, k) \in \operatorname{dom} M_T$ **then** ▷ coupled transition randomness
        $s' \leftarrow M_T(s, a, k)$
      else
        $s' \leftarrow \text{Step}(s, a)$
        $M_T(s, a, k) \leftarrow s'$
      end if
      $w^{(i)} \leftarrow \text{Reward}(s, a, s') + \log p_a - \log q_a$
      $s^{(i)} \leftarrow s'$
    end for
    $\ell_t \leftarrow \log \sum_{i=1}^{N} \exp(w^{(i)}) - \log N$
    $(s^{(1:N)}, M_A^{(1:N)}, M_C^{(1:N)}) \leftarrow \text{Resample}\big((s^{(1:N)}, M_A^{(1:N)}, M_C^{(1:N)}),\ w^{(1:N)}\big)$
  end for
  $\log \hat{Z} \leftarrow \sum_{t=1}^{H} \ell_t$
  $\log \hat{Z}_{H+1} \leftarrow 0$
  for $t \leftarrow H$ down to 1 **do**
    $\log \hat{Z}_t \leftarrow \ell_t + \log \hat{Z}_{t+1}$ ▷ suffix sums for Eq. (6)
  end for
  return $\log \hat{Z} + \sum_{t=1}^{H} \left(\overline{\log \hat{Z}_t} \cdot g_t \right)$
end procedure