Optimal composition of multiple value functions allows efficient, safe and stable dopamine-mediated learning

A PREPRINT

Pranav Mahajan
Nuffield Department of Clinical Neurosciences,
University of Oxford
pranav.mahajan@ndcn.ox.ac.uk

Ben Seymour
Nuffield Department of Clinical Neurosciences,
Institute of Biomedical Engineering,
University of Oxford
ben.seymour@ndcn.ox.ac.uk

October 14, 2025

ABSTRACT

The seminal reward prediction error theory of dopamine function faces several key challenges. Most notable is the difficulty learning multiple rewards simultaneously, inefficient on-policy learning, and accounting for heterogeneous striatal responses in the tail of the striatum. We propose a normative framework, based on linear reinforcement learning, that redefines dopamine's computational objective. We propose that dopamine optimises not just cumulative rewards, but a reward value function augmented by a penalty for deviating from a default behavioural policy, which effectively confers value on controllability. Our simulations show that this single modification enables optimal value composition, fast and robust adaptation to changing priorities, safer exploration in the context of threats, and stable learning amid uncertainty. Critically, this unifies disparate striatal observations, parsimoniously reconciling threat and action prediction error signals within the striatal tail. Our framework refines the core principle governing striatal dopamine, bridging theory with neural data and offering testable predictions.

1 Introduction

Almost three decades ago, Schultz et al. [1] proposed that midbrain dopamine (DA) neuron phasic activity encodes temporal difference errors (TDEs). This fundamental idea, leveraging the temporal difference (TD) learning algorithm [2], suggested the brain could assign credit in terms of expected future reward using temporally successive predictions. Numerous experiments have since substantiated this relationship within the TD reinforcement learning (TDRL) framework [3, 4, 5, 6, 7]. A core motivation was that engineered systems employ similar algorithms to optimise actions in complex environments, mirroring challenges faced by animals [1]. However, three key challenges currently impede TDRL research from fully realising this core motivation.

The first challenge arises because typical experiments and computational models, utilising single-attribute rewards (e.g., juice) and monolithic value functions, inadequately capture the multi-objective nature of challenges animals face in real life. That is, they constantly need to simultaneously satisfy distinct, often conflicting objectives under changing priorities, a kind of non-stationary rewards. This necessitates either multiple value functions [8, 9] or alternative efficient representations like the successor representation [10, 11, 12], as standard RL approaches struggle with instant revaluation under non-stationary rewards [8, 9, 13]. This need for fast adaptation to multiple objectives (often encoded by different rewarding attributes), under shifting priorities, is ubiquitous in homeostasis [14, 15, 16] and extends to human cognitive tasks [17]. Neural evidence further suggests dopaminergic circuits projecting to different targets may encode multiple value functions corresponding to various reward modalities (e.g., food, juice, water [18, 19, 20, 21, 22]), valence (e.g., threats versus rewards [23]), substance type [24]), or even abstract features and contexts [25, 26, 27], but this isn't captured by the standard TDRL framework.

The second challenge concerns the TD-learning rule used by Schultz et al. [1], which learns values under the current behavioural policy (on-policy algorithms in RL), as opposed to values under an optimal policy (off-policy algorithms).

A key implication is that the TD-learning algorithm learns values which are estimates of the so-called future returns, a long-run measure of cumulative rewards, assuming the agent continues to act using its current behavioural policy. While a core motivation was to learn optimal actions [1], on-policy algorithms characteristically learn suboptimal values under an often-exploring, suboptimal policy. This compromise, well-noted in traditional single-objective RL [28] and usually dealt with by tuning exploratory noise, becomes particularly problematic in multi-objective RL, where the behavioural policy can change drastically with shifts in needs or non-stationarity [8].

The third challenge stems from widespread dopamine heterogeneity, which calls into question the physiological basis of a single, broadcasted scalar reward prediction error (RPE). While a final scalar variable is computationally necessary to guide choice, the notion of a monolithic RPE signal is increasingly at odds with observations of diverse dopamine responses, both between and within striatal subregions. Furthermore, extending the classical model [1] to multiple objectives [9] only partially explains this diversity (e.g. between different DA targets), leaving more perplexing forms of heterogeneity unresolved.

Nowhere is this challenge more apparent than in the tail of the striatum (TS), whose normative role is hotly debated. Evidence suggests TS-projecting neurons encode threat prediction errors (TPEs) to guide avoidance, even without explicit aversive reinforcement [29, 30, 31, 32], with direct and indirect pathway heterogeneity [32]. Others posit that TS-projecting neurons encode action prediction errors (APEs) [33], which have been linked to "value-free" habits [34] and are difficult to integrate into a standard value-based TDRL framework, as they are often modelled as a separate ad-hoc component [34, 35, 36]. Reconciling these conflicting views is a central problem; recent unifying accounts, for instance, often highlight the TS's role in stimulus-associated salience predictions but tend to disregard APEs [37]. Thus, a parsimonious account that integrates the normative roles of both TPEs and APEs into the TDRL framework remains elusive.

While distinct, the first two challenges - the need for multi-objective learning and the pitfalls of on-policy methods - converge on a single, fundamental question of optimal composition of multiple values. The brain clearly possesses mechanisms for learning multiple values, but how can they be reliably combined to produce a coherent and optimal policy? Ideally, this "recipe" for combining values should satisfy two key properties: optimality and compositionality. Optimality, the bedrock of modern reinforcement learning [38], simply means choosing the best possible action [39]. Compositionality, conversely, is the formal principle of constructing solutions to complex problems from a set of modular components, for instance by allowing a multi-attribute task to be represented by a basis set of value functions tuned to individual reward dimensions. Crucially, this structure allows policies for novel priority landscapes to be constructed by flexibly recombining these components, obviating the need to learn each new configuration de novo.

As noted previously [7], the choice of learning algorithm is not a mere technicality but a foundational choice dictating the system's computational objectives. Recent proposals often overlook this, foregoing either optimality [9] or composability [8]. On-policy methods (e.g., SARSA), for instance, suffer from learning interference, especially under shifting contextual priorities. When the current context prioritises one reward, the agent's policy becomes biased; because learning is tied to this policy, the valuation of alternative rewards is not learned in isolation but is corrupted by being evaluated through the lens of the current trajectory. In contrast, off-policy methods (e.g., Q-learning) fail because of the non-linearity of the Bellman optimality (max) operator, which is not additive. This creates an unrealistic assumption: the agent is treated as perfectly rational during valuation (identifying the single best future action) yet as boundedly rational during action selection (making stochastic choices). This inconsistency between an ideal valuation and a bounded action-selection corrupts the composition of different reward values, revealing the need for an internally consistent framework for decision making.

To resolve this, we adopt a normative framework from control engineering [40, 41, 39, 42], broadly termed linear RL [43]. We propose that the dopamine system's objective is not merely to optimise cumulative reward, but to optimise returns augmented by a penalty for deviating from a default policy. This single modification provides a principled solution to the optimal composition problem [44]. By enforcing a consistent assumption of bounded rationality throughout both valuation and action selection, it resolves the paradoxical logic of standard algorithms, allowing multiple values to be robustly combined.

Remarkably, the same principle that ensures optimal composition also provides a unified normative account of dopamine heterogeneity. The framework's parallel architecture for outcome-specific prediction errors supports efficient learning across multiple rewards and rapid adaptation. Furthermore, composing values from different initialisations can explain within-target heterogeneity and threat prediction errors (TPEs) and how the flexible expression of such innate values can drive safe learning. Most strikingly, the penalty term itself manifests computationally as an action prediction error (APE), revealing its function in promoting stable learning by conferring a value on controllability. We substantiate these claims in our Results, which use a didactic example and simulations to demonstrate each of these respective normative advantages.

2 Results

2.1 Theory sketch

The model utilises the linearly-solvable Markov Decision Process (MDP) framework [40, 41, 39, 42], also known as or entropy-regularised or KL-regularised RL. Distinct from the original control theoretic formulations [41, 39], we focus on temporal difference learning rules for solving this and extend it to learning values for multiple objectives or reward types, leveraging inherent compositionality theory afforded by the linear MDP framework [44].

At Marr's computational level [45], our agent optimises returns augmented by an additional KL divergence term ($D_{\mathrm{KL}}(\pi(\cdot|s_t)\|\pi^d(\cdot|s_t))$) that penalises proportional to how much the current behavioural policy ($\pi(\cdot|s_t)$) deviates from some default policy ($\pi^d(\cdot|s_t)$) at each timestep:

$$G_t = \sum_{k=0}^{\infty} \gamma^k (r_{t+k+1} - \tau D_{\mathrm{KL}}(\pi(\cdot|s_t)\|\pi^d(\cdot|s_t)))$$

where G_t is the return from timestep t onwards and r_{t+k+1}, is the reward at the k timesteps after t, following standard notation from Sutton and Barto [28]. The KL divergence term acts as a relative-entropy regularisation. If π^d is uniform, this encourages random exploration (Max Entropy RL); if π^d slowly tracks the learned policy π, it promotes choice perseveration or soft-habit formation by penalising deviations from recently taken actions [34, 43, 46, 47].

At the algorithmic level, we implement this using soft Q-learning [48], an off-policy temporal difference (TD) method, adapted here for the general relative-entropy objective. Q-values are updated via:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\delta_t,$$

where α is the learning rate, and δ_t is the reward prediction error at timestep t, defined as:

$$\delta_t = r_{t+1} + \gamma V_Q(s_{t+1}) - Q(s_t, a_t),$$

where the soft state-value is given by $V_Q(s) = \tau \log \mathbb{E}_{a\sim\pi^d} \exp(Q_\pi(s,a)/\tau)$(see Methods, Eq. 9). $V_Q(s)$ replaces the $\max_a Q(s,a)$ operation in the Bellman optimality equation, which is often (biologically) intractable in large or continuous action spaces, with a more computationally feasible expectation, with other properties favouring optimal composition of multiple values. Notably, for this one-step algorithm, the KL-divergence from the objective does not appear in δ_t because the action a_t has already been taken [49, 48, 50]; however, it re-emerges in our derivations of multi-step extensions (Section 2.5). Actions are selected using a Boltzmann policy derived from the Q-values, which is optimal under this regularised objective (Methods, Eq. 8).

This soft Q-learning rule is applied independently to learn multiple Q-functions for different objectives (r_i), which are then composed using a weighted softmax or weighted summation operation f, with weight vector $\mathbf{w}$ (Theorem 1, Methods and Supplementary Methods 2). Note, taking an action at timestep t results in rewards $r_{i,t+1}$, for the i-th reward dimension, but we will omit the time subscript from here for convenience. The reward weighting $\mathbf{w}$ is determined by either internal homeostatic priorities (e.g. hunger, thirst, salt deprivation [14, 9] or injury belief state [51]) or inferred task-based rules [25, 26] or survival priorities (e.g. threat inference).

$$r_c(s,a) = f(r_1, r_2, \ldots, r_n; \mathbf{w})$$

Importantly, the compositionality of optimal control laws in linear MDPs [44] allows us to decompose the multi-objective RL problem, into individual value components $Q_i(s,a)$ optimising independent objectives r_i, whilst ensuring that combining these individual values, using the same composition f, ensures the resultant policy optimises the intended reward composition r_c on the multi-dimensional reward manifold.

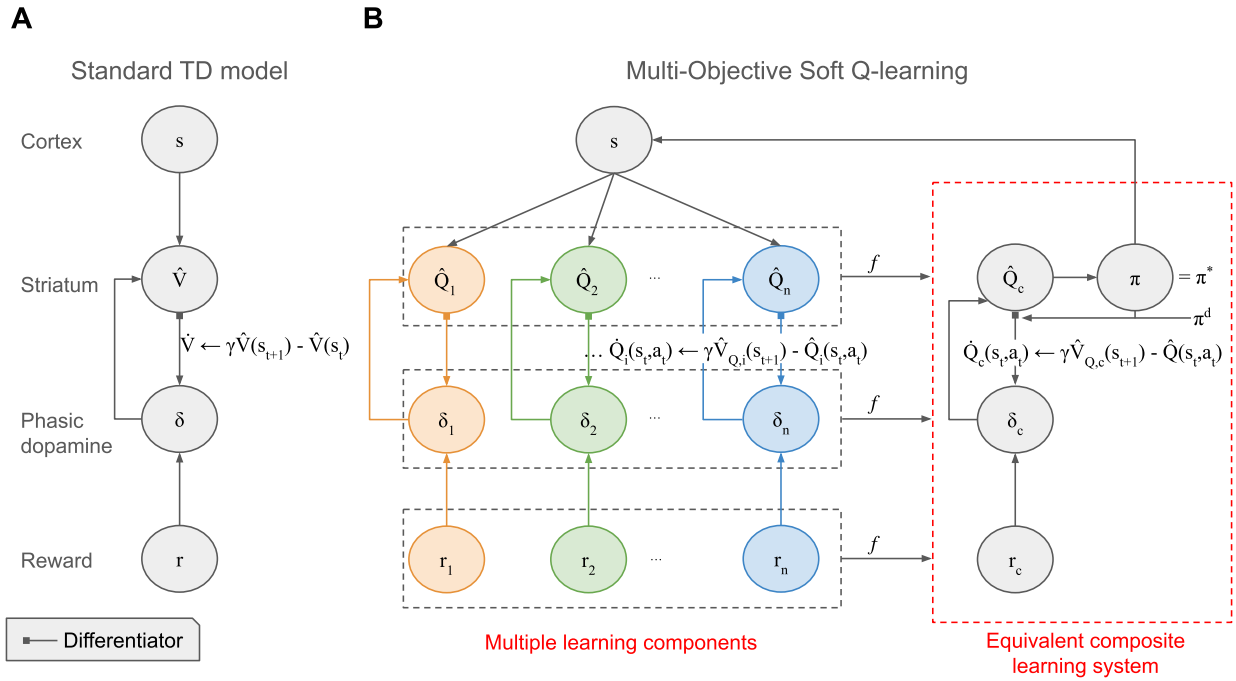

Figure 1: Proposed neural implementation for optimally composable multi-objective RL. (A) Conventional TD learning model [1]: Dopamine neurons compute a scalar reward prediction error (RPE) that drives learning of a single state value function in the striatum. Extensions feature parallel systems for different outcomes, generating vectors of state values and RPEs [9]. (B) Our model: Parallel soft Q-learning modules learn distinct state-action values (Q-values) for multiple or shared outcomes within the linearly solvable MDP framework [41]. These Q-values are optimally composed (f) to guide behaviour that optimises a weighted combination of rewards. This architecture supports parallelisation across different dopaminergic targets and allows for within-target heterogeneity through differing value initialisations (even without immediate outcomes). A key desideratum is access to the default policy π^d by the TD error computing differentiator [1] for computing the soft state values V_Q.

At Marr's implementation level, our model posits parallel soft Q-learning mechanisms to account for heterogeneity in dopaminergic responses, both between and within distinct neural targets (Fig. 1). Learning separate, composable Q-values for different reward attributes (e.g., appetitive, aversive) allows for efficient acquisition of multi-attribute rewards. It also enables the representation of dedicated threat or punishment values essential for safe behaviour, preventing them from being overridden by unrelated positive rewards [52, 53, 54]. However, within a single dopaminergic target (e.g. the tail of the striatum), we propose that heterogeneity can also arise from multiple value functions sharing a common outcome signal but differing in their initialisations, potentially reflecting priors for different contexts, associated with different priorities. Such value initialisations can drive behaviour and explain dopaminergic responses even without explicit outcomes [55, 56, 31], a concept addressed in Section 2.4.

We first illustrate the principles of optimal value composition with a didactic simulation, building on Todorov [44], while situating and comparing with contemporary biological multi-objective RL models, then proceed to more concrete experiments.

2.2 Optimal composition of multiple value functions for reliable optimisation

We present a didactic analysis and simulation demonstrating our claim that recently proposed methods for learning and combining multiple values either forego optimality [9] or composability [8]. A recipe for optimal (or near-optimal) composition allows a system to combine several independent components to reliably optimise the resultant composite reward function. Put simply, it is a strategy to get several selfish value functions to behave well together.

The general approach in prior works in multi-objective RL often linearly decomposes a composite reward r_c into attributes $(r_1, r_2, \dots)$. Separate value functions $(Q_1, Q_2, \dots)$ are then learned, each optimising returns for a single

attribute. These individual value functions are subsequently linearly combined (using the same weights as the reward decomposition) to form a composite value function Q_{comp} that guides a unified policy [57, 8, 9]. However, depending on the specific value learning algorithm, this can lead to a critical trade-off: either the individual values Q_i are optimal for their respective rewards r_i but their composition Q_{comp} is not optimal for r_c (i.e., $Q_{comp} \neq Q_c^*$), or the composition is well-defined but the individual values Q_i themselves are sub-optimal (i.e., $Q_i \neq Q_i^*$).

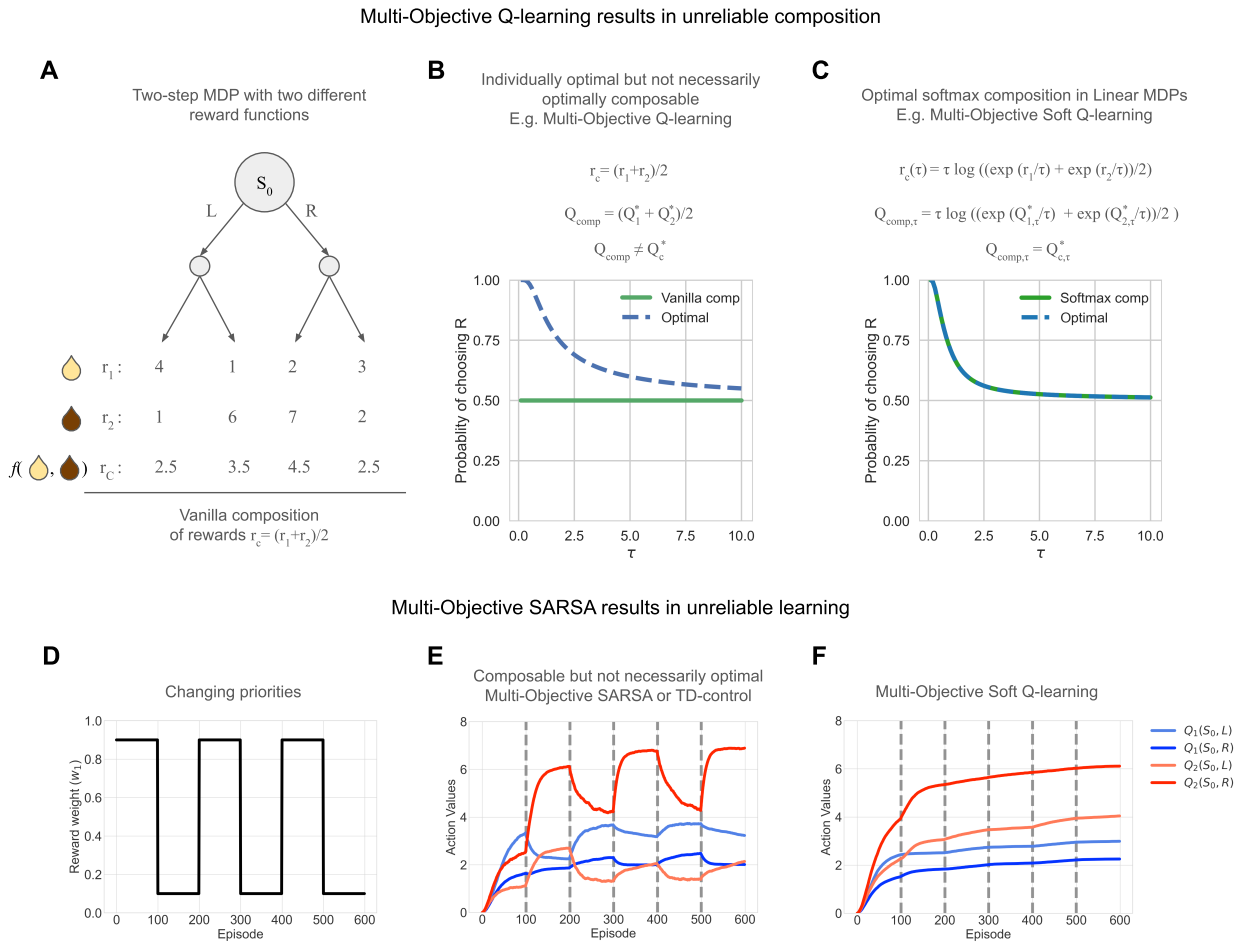

Figure 2: Demonstration of the reliable and optimal composition of values in linear MDP. (A) Two-step MDP with two (diverging) reward functions, say Juice 1 (r_1) and Juice 2 (r_2). (B) Action selection probabilities under sub-optimal additive composition of Q-values in MDPs (Q_{comp}) deviate from optimal behaviour mandated by Q_c^* optimising r_c. (C) Action selection probabilities under optimal softmax composition in Linear MDPs (Q_{comp}) match optimal behaviour mandated by Q_c^* optimising r_c. Weights set to $w_1 = w_2 = 0.5$ (equal priority for both rewards) and action probabilities plotted for a range of τ. (D) Changing reward weight w_1, denoting the change in priorities and $w_2 = 1 - w_1$. (E) Action values of multi-objective (MO) SARSA show unstable and unreliable learning over episodes. (F) Action values of MO Soft Q-learning show reliable and stable learning over episodes. Note that the focus is on value learning over episodes is stable or has interference; action values from (E) and (F) cannot be compared as the objective functions are different.

To illustrate the first issue - individually optimal values that compose sub-optimally, we consider the off-policy, multi-objective Q-learning, devised by Russell and Zimdars [57] and utilised by Dulberg et al. [8]. This method learns optimal Q_i for each r_i using standard Q-learning (using equation 10) and then additively combines them (referred to as Vanilla comp). In a two-step MDP with two reward functions r_1, r_2 (Fig. 2A), if we equally weight rewards ($w_1 = w_2 = 0.5$) to get r_c, the (undiscounted) optimal Q-values for r_c in the starting state S_0 are $Q_c^*(S_0, a = L) = 3.5$

and $Q_c^*(S_0, a = R) = 4.5$. However, the additively composed Q-values are $Q_{comp}(S_0, a = L) = Q_{comp}(S_0, a = R) = 5$. Consequently, an agent using Q_{comp} with a softmax policy chooses actions L and R with equal probability, irrespective of the temperature τ, deviating from the optimal behaviour for r_c (Fig. 2B). This sub-optimality arises from the non-linearity of the $\max$ operator in the Bellman optimality equation (see Methods, Section 4.3).

To make this concrete, one can view this as multi-attribute decision making problem with two juices of similar utility per millilitre consumption. The left path from S_0 leads to maximum consumption of Juice 1 (r_1) whereas the right path leads to maximum Juice 2 (r_2). The abstract result above — where the flawed multi-objective Q-learning algorithm computes both left and right actions at S_0 as equally valuable even when the path leading to $r_c = 4.5$ is objectively better — demonstrates how the inconsistent rationality assumption in standard Q-learning leads to suboptimal choices. Here r_c is an internally constructed composite reward based on priorities w_1 and w_2 and external rewards r_1 and r_2.

In contrast, our approach using soft Q-learning within the linear MDP framework allows for a reliable composition. Here, rewards $r_{c,\tau}$, a composite reward function composed from individual rewards r_1 and r_2, and Q-values $Q_{comp,\tau}$ are functions of τ, which controls the influence of a default policy π^d (here, uniform, though its specific choice does not alter this result). As shown in Fig. 2C, the resulting composed policy reliably optimises the target reward composition, achieving $Q_{comp,\tau} = Q_{c,\tau}^*$ (Theorem 1, Methods Section 4.3). This demonstrates optimal value composition [44]. For completeness, we also simulated a viable alternative - the additive reward composition within this linear MDP (supplementary Fig. S1 and text), which also outperforms MO Q-learning additive composition in MDPs, along with theoretical guarantees [58].

On the other hand, works that employ on-policy algorithms, i.e. TD(0) [1], its multi-objective extension [9] and extensions to control (SARSA), are known to not reliably learn optimal policies, especially from sub-optimal trajectories (see supplementary Fig. S2).

However, the limitations of on-policy learning become particularly acute in multi-objective scenarios with dynamically shifting priorities. This is because policies optimal to one value component are bound to be sub-optimal for other value components, but values for all components are learnt under a common behavioural policy. Here, we highlight a critical form of interference in multi-objective (MO) on-policy algorithms such as MO SARSA (or vanilla Reward Bases [9]): as priorities shift (Fig. 2D), the ensuing changes in the global behaviour policy directly impact the valuation of individual components (supplementary Fig. S1 illustrates different policies under different priorities). Because on-policy value updates (e.g., in MO SARSA) depend on the current policy π (i.e., $V_{i,\pi}(s')$), actions taken to optimise one reward modality can lead to unintended revaluation and even unlearning of values for other modalities (Fig. 2E). In our juice example, this models a task where the rules frequently switch, making only one juice the prioritised or most rewarding option at a time (Fig. 2D). This could also be implemented via changes in homeostatic needs, although these typically occur on a slower timescale. The unstable value estimates demonstrate how an on-policy agent struggles to adapt: its learned value for 'Juice 1' becomes corrupted while it pursues 'Juice 2', preventing a flexible switch when the rules change. In contrast, our off-policy multi-objective (MO) soft Q-learning framework effectively mitigates this interference, ensuring stable and accurate learning of all value components (Fig. 2F). These divergent learning dynamics — instability in on-policy versus stability in off-policy approaches under shifting priorities — offer clear, experimentally testable predictions.

In summary, this section demonstrates that MO soft Q-learning achieves a reliable and optimal composition of multiple values. This contrasts with MO SARSA, which suffers from unstable value learning under shifting priorities, and standard MO Q-learning, which can yield sub-optimal compositions. It is important to note that several of the diverging predictions arise in MDPs with two or more steps sharing some common states, whereas several experiments [21] and subsequent modelling [9] are limited to one-step tasks where all learning rules reduce to the Rescorla-Wagner rule and do not yield diverging predictions. These diverging predictions can be used to tease apart strategies used in multi-attribute decision making.

2.3 Efficient learning and off-policy fast adaptation

Efficient adaptation to non-stationary rewards is a hallmark of intelligence. While full model-based RL offers a solution, it is computationally costly. Part model-free solutions rely on either (i) composing independent value functions for different (pre-defined) reward types, as discussed previously [8, 9], or (ii) learning efficient representations such as the successor representation (SR) [10], which allow rapid revaluation of policies when reward functions change.

Crucially, in the case of on-policy model-free algorithms, Millidge et al. [9] show that a combination of TD(0) learning rule for each of the reward types is akin to a compressed SR with rewards tuned only to relevant dimensions. An equivalent relationship in off-policy algorithms is lacking. The default representation (DR), an off-policy counterpart to the SR derived from linear MDPs [43, 41, 39], overcomes the on-policy limitations of the SR and offers such a path.

We first establish a theoretical link between our multi-objective (MO) Soft Q-learning and the DR, and then proceed to empirically demonstrate its superior performance over on-policy alternatives.

Theoretical result: Relationship to the default representation (DR). MO Soft Q-learning learns values equivalent to those learned by a compressed DR tuned to only relevant (predefined) reward dimensions (Methods, Section 4.5). It further provides two benefits: First, MO Soft Q-learning scales linearly with state space size (assuming a fixed, smaller number of reward dimensions), whereas the full DR scales quadratically. Second, unlike the SR, the DR lacks an efficient TD learning algorithm and typically requires matrix inversions for its computation, which are biologically implausible. MO Soft Q-learning provides a TD-based mechanism to learn DR-like values for the relevant pre-defined reward dimensions.

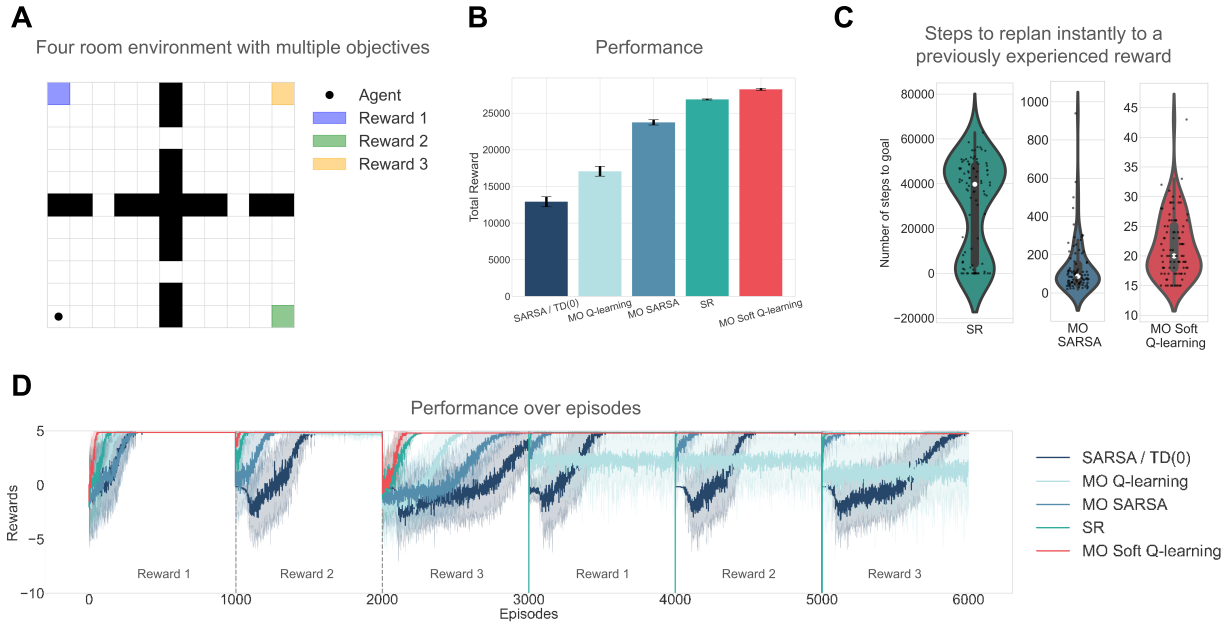

Figure 3: Demonstration of efficient learning and fast adaption to changing priorities in a four-room environment. (A) Four-room environment with changing priorities between three rewards every 1000 episodes, and the agent starts in the same starting position (bottom-left corner, akin to a shelter). Rewards for goals: +5 points, step cost: -0.01. The episode terminates only on reaching one of the rewarding goals. Meta-parameter $\tau = 0.5$ to allow all algorithms to converge to an efficient path to the goal. (B) Multi-objective (MO) Soft Q-learning algorithm performs the best amongst comparisons in terms of total rewards accrued, highlighting its fast adaptation capabilities. (C) Steps to the goal on episodes 3000, 4000 and 5000 are plotted for different algorithms to test replanning to a previously experienced reward. SR performs the worst at replanning, requiring substantial policy re-evaluation, while MO Soft Q-learning performs the best amongst comparisons. (D) Performance over episodes shows different rates of adaptation for different algorithms upon a change of priorities. Do note, SR accrues many losses on episodes 3000, 4000 and 5000, but the plot is truncated -10 average rewards for visualisation.

Simulation result: MO Soft Q-learning enables superior adaptation to shifting priorities. We empirically tested these advantages in a four-room grid world where an agent pursued one of three goals, with priorities shifting every 1000 episodes (Fig. 3A; see Methods for more details). This task, though standard, is more complex than some prior grid worlds [8, 9], by introducing walls.

MO Soft Q-learning outperformed SR and other MO TD algorithms in total rewards accrued (Fig. 3B), demonstrating superior adaptation. The policy-dependence of SR was particularly detrimental during priority shifts requiring substantial re-planning [59]. For instance, when needing to switch back to a previously learned goal after extensive training on another, the SR agent took orders of magnitude more steps than MO Soft Q-learning or even MO SARSA (Fig. 3C, episodes 3000, 4000, 5000). MO Soft Q-learning, by maintaining relatively stable, independent values for each reward, could immediately leverage the appropriate value function. The adaptation rates over episodes further illustrate these differences (Fig. 3D). These findings not only highlight SR's limitations in dynamic environments but also

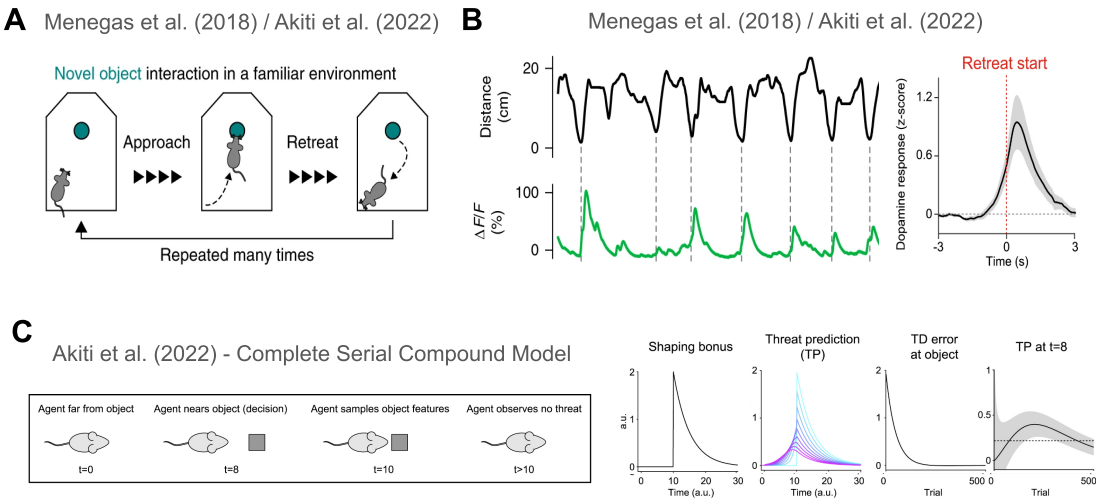

Figure 4: A summary of experimental findings of the role of TS in threat prediction. (A) Illustration of the novel object task and the approach-retreat behaviour. (B) Approach-retreat behaviour observed in the distance to the novel object and the TS responses observed on retreat start. (C) Complete Serial Compound model of TS responses modelled using a value initialisation, which is equivalent to a potential-based reward shaping. All figures are retrieved and minimally adapted from Menegas et al. [30] with permission from Springer Nature and from Akiti et al. [31] under CC-BY-NC-ND 4.0 license with permission from Elsevier.

offer differentiating testable predictions against SR-based models of dopamine [12] and other MO TD approaches like Reward Bases [9], previously unidentified in Millidge et al. [9].

Lastly, we observe that off-policy algorithms continue to propagate optimal Q-values for all components, while collecting data under different policies (priorities), unlike on-policy algorithms (supplementary Fig. S4). However, MO Q-learning requires commensurate (heuristic-based) temperature annealing to get the most benefits, as also seen in Dulberg et al. [8], whereas MO soft Q-learning manages this trade-off without such explicit annealing. Such a multi-objective RL solution for efficient learning can be mapped to different DA targets, responsible for different reward bases (for example, see Millidge et al. [9]). Next, we show how the same recipe can be extended to model and explain certain heterogeneities within DA targets.

2.4 Safe learning and explaining novelty responses in TS

Beyond efficient adaptation to changing rewards, relying on an aversive value initialisation that can be flexibly expressed can be useful in generating safe behaviours, such as avoiding potential threats. Recent experiments [23, 30, 31, 32] propose that the tail of the striatum (TS) encodes initial threat predictions for novel stimuli, contributing to avoidance, and are updated by dopamine-mediated threat prediction errors (TPEs). We revisit an idiosyncrasy in the observed data, propose a model qualitatively explaining some of the findings and discuss implications for pathway-dependent heterogeneity in the TS.

Simulation result: Threat belief-gated value composition reproduces approach-retreat dynamics and TS dopamine signals. Studies in mice reveal that interactions with a novel object involve approach-retreat bouts, importantly, accompanied by TS dopamine activity during retreat but not during approach [30, 31] (Figs. 4A, B). These studies model TS dopamine with aversive prediction errors and model this phenomenon using a complete serial compound (CSC) model [31], which incorporates value initialisation, which is equivalent to potential-based reward shaping bonus [60] (Fig. 4C). The CSC-based model, albeit tremendously helpful in describing the TS responses, relies on arbitrary thresholds for engage-avoid decisions. Further, it cannot produce approach-retreat bout behaviour in space, and therefore, it is unclear if reward shaping-based modelling of TS responses extends directly to 2D state-action spaces (e.g. Ng et al. [61]). Further, a shaping bonus is a non-distorting bonus [55, 61], which would mandate that the net effect of any cycle of states (such as an approach-retreat bout) is zero, a condition not adequately tested in CSC, which is unidirectional in space and time with no cycles.

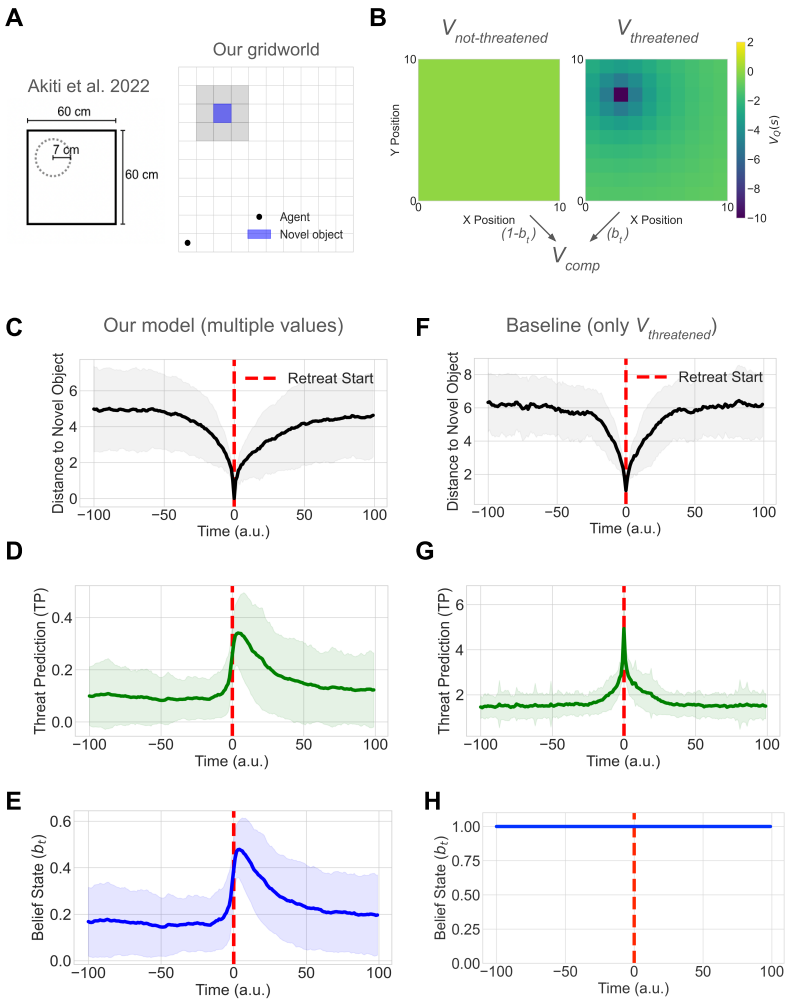

Figure 5: Threat belief-gated value composition reproduces approach-retreat dynamics and TS dopamine signals. (A) Grid-world environment analagous to Akiti et al. [31]. (B) Two value initialisations used to produce the observed results, gated by the threat belief state b_t. (C) Distance to novel object shows approach-retreat bout behaviour (D) Threat Prediction (TP) is analogous to the TS dopamine activity during retreat but not during approach. The relative magnitude of the composite Threat Prediction response depends on the nature of composition - soft maximum or additive, here we use soft maximum for consistency in the paper (E) The threat belief state b_t over time that gates the contribution of the two values. (F) When using only the $V_{threatened}$, the approach-retreat bout behaviour is also observed in distance to novel object (albeit we observe fewer bouts) (G) Threat Prediction (TP) is observed during approach and retreat phases unlike the TS dopamine activity and (H) Using only $V_{threatened}$ is akin to setting the $b_t = 1$.

To address these limitations, we simulate a grid world environment without explicit rewards or punishments (Fig. 5A). The agent receives threatening observations o_t from a Bernoulli process: if the agent is in the vicinity of the novel object (demarcated by the grey area), $p(o_t = 1|\text{threatened}) = 0.9$, otherwise outside the grey area, $p(o_t = 1|\text{not-threatened}) = 0.1$. Using these observations, the agent infers a threat belief state b_t via a Bayesian Beta posterior. Two value functions with different initialisations drive behaviour: $V_{not-threatened}$ (neutral initialisation) and $V_{threatened}$ (aversive initialisation at the novel object) (Fig. 5B). Both values share the common outcome signal for learning, here zero since there is no external outcome. These values are dynamically combined into a composed value V_{comp} using softmax composition, weighted by the belief state: $V_{threatened}$ and $(1 - b_t)$ weights $V_{not-threatened}$. However, all of the results would also hold true for additive composition, and with TD-learning or SARSA with two different initialisations, we simply use soft maximum for consistency throughout the paper.

We find that the proposed model produces approach-retreat bout behaviour and reproduces TS dopaminergic responses during retreat, where the threat prediction (TP = $-V_{comp}$) aligns with experimental observations [30] (Fig. 4C&D). Importantly, using only $V_{threatened}$ (analogous to PBRS in 2D space) fails to reproduce these responses in a grid world setting, as it generates high TP during both approach and retreat (Fig. 4F&G). This discrepancy arises because the CSC modelling approach conflates spatial and temporal information, which our grid-world model decouples. Our results make a testable prediction that gating aversive value based on a threat belief state (Fig. 4E), akin to a context-dependent switch, is a normative mechanism for adaptive safe behaviour. Similar asymmetric TP responses can also be generated by non-Bayesian, switch-like dynamics modulating the two values (supplementary Fig. S5). We did not include explicit curiosity-based exploration rewards for simplicity of modelling TS responses, but their inclusion in the behavioural model could increase agent tendencies to approach the novel object. Such a dynamic composition of multiple value initialisation also provides one way of unifying the distorting novelty bonuses and non-distorting shaping bonuses [55], i.e. under fixed weights, it acts like a shaping bonus and under varying weights it produces novelty-like bonuses.

The belief-gated, dual-value architecture further offers a potential neural implementation for TS function, where $V_{threatened}$ and $V_{not-threatened}$ could potentially map onto direct (D1) and indirect (D2) TS pathways, respectively (supplementary Fig. S6). This generates several testable predictions: First, in a task involving threat-reward conflicts, this predicts opposing effects on avoidance behaviour, with D1 promoting avoidance and D2 suppressing it. Second, ablating D1 TS neurons in the same task is predicted to reduce avoidance, while D2 TS ablation should increase it. Third, suppose the inferred threat belief b_t covaries with observable threat cues like object size or whether it is moving. In that case, the composite TS dopamine responses should scale accordingly with those features of the potentially threatening novel object. Fourth, it predicts distinct learning dynamics: with repeated exposure to the novel object, the D1-associated $V_{threatened}$ should diminish, reducing avoidance, while the D2-associated $V_{not-threatened}$ might remain stable. Fifth, in a one-step task, a punishing outcome with a magnitude falling between the initial $V_{threatened}$ and $V_{not-threatened}$ values should elicit negative threat prediction errors (TPEs) in D1-projecting SNL neurons and positive TPEs in D2-projecting SNL neurons [30]; such asymmetric TPEs could potentially encode outcome uncertainty [62]. Recent work by Tsutsui-Kimura et al. [32] in a threat-reward conflict task provides evidence supporting the first three predictions: opposing roles, ablation effects, and modulation by threat salience. Their findings on learning dynamics were mixed (non-significant D1 decrease, significant D2 increase), and the prediction of asymmetric TPEs remains to be directly tested, to our knowledge. Our model thus offers a novel explanation for within-target dopaminergic heterogeneity based on differentially initialised, composable values sharing a common outcome; distinct from feature-specific heterogeneity accounts [63].

2.5 Stable learning and reconciling conflicting views on TS function

While multiple value initialisations can account for threat prediction in the tail of striatum (TS), recent findings also implicate TS dopamine in encoding action prediction errors (APEs) [33] that support perseverative behaviours or habits [34, 36, 46]. This raises the question of how these seemingly distinct TPE and APE signals can be reconciled within a unified normative framework and what computational purpose such APE-like signals serve.

The linear MDP framework offers a parsimonious explanation. The default policy π^d, so far assumed uniform, can be slowly updated to track the agent's learned policy π, thereby encouraging perseveration [43]. We find it is worthwhile to mine the analogy from RPEs for $Q(s, a)$ and $V(s)$ to APEs - whether there exist analogous state-dependent aggregate APEs? We propose that the KL divergence term, $D_{\mathrm{KL}}(\pi(\cdot|s_t)\|\pi^d(\cdot|s_t))$, within the regularised objective itself normatively accounts for an aggregate, state-dependent APE. Although this KL term is implicit in one-step soft Q-learning TD errors (as action a_t is already chosen), it becomes explicit in multi-step formulations, which motivates our theoretical results.

Theoretical result: Multi-step soft Q-learning reveals an explicit APE-like term. We derive novel, multi-step extensions: N-step soft Q-learning and soft Q(λ) with eligibility traces (Methods Section 4.4; derivations in Supplementary Methods 3, 4). Note, these are fully off-policy as well, using Tree Backup operations, which simplify under Boltzmann policy. For these, the TD error for $Q(s_t, a_t)$ can be expressed in relation to state-value changes as: $\delta_t = [r_{t+1} + \gamma V_Q(s_{t+1}) - V_Q(s_t)] - \tau \mathrm{KL}_t$, where $\mathrm{KL}_t = D_{\mathrm{KL}}(\pi(\cdot|s_t)\|\pi^d(\cdot|s_t))$ is the action policy divergence term. We interpret this KL_t term as a normative, scalar APE signal. Thus, the Q-value update effectively becomes $\delta_t = \mathrm{RPE}_t - \tau \mathrm{APE}_t$. Given that TS-projecting neurons in SNL encode magnitude of aversive prediction errors $-\delta_t$ [30, 31], and do not respond to positive rewards [23, 33], their activity in tasks without explicit aversive stimuli could be primarily driven by this APE-like KL term (or APE_t).

Simulation result: KL-divergence dynamics in multi-step Soft Q-learning mimic APEs and support unified TS function. To test this, we simulated a multi-step two-choice task (loosely) inspired by Greenstreet et al. [33], where rewards depended probabilistically on an observable context (Fig. 6A, B). Here, the agent starts in the state S_0 and takes a series of primitive actions a_L or a_R to get closer to either of the terminal states S_L or S_R. This setup makes

the two-choice bandit tasks more granular by dividing them into simpler actions. Consistent with their finding that TS dopamine decreased over trials in an APE-like manner (Fig. 6C), the KL_t term in our soft Q(λ) model's TD errors also decreased across episodes (Fig. 6D). This decrease reflects the default policy π^d gradually aligning with the learned contextual policies π, effectively encoding soft habits (Fig. 6E). This result offers a unifying perspective: TS dopamine could reflect a composite signal where TPEs dominate in threat-relevant contexts, while APEs (the KL_t term) dominate when behaviour stabilises around a default/habitual policy.

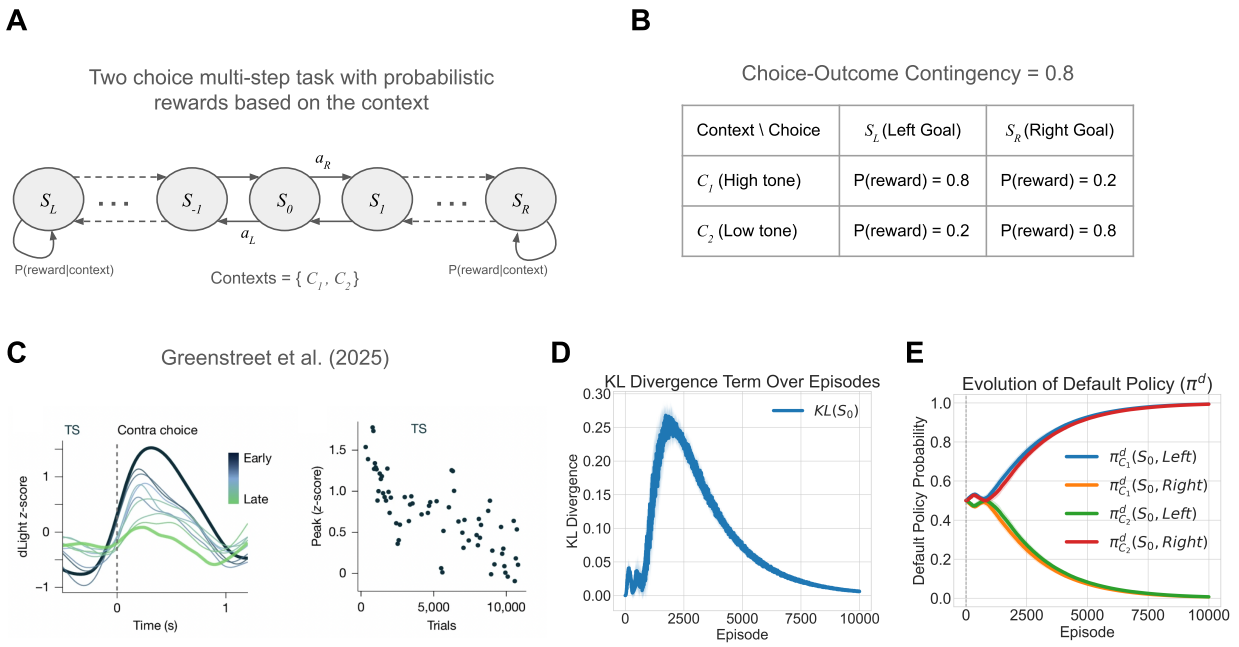

Figure 6: KL-divergence dynamics in multi-step Soft Q-learning mimic APEs and support unified TS function. (A) Two-choice multi-step task with probabilistic rewards based on the context (two possible contexts, treated as fully observable states). (B) Choice-Outcome Contingencies are dependent on the context, leading to different correct choices based on the context. (C) Evidence for APEs, figures adapted from Greenstreet et al. [33] under CC-BY 4.0 license. (Figures were cropped and combined) (D) KL divergence term in soft Q(λ) TD-errors qualitatively recapitulates the APE-like TS responses. Showing the KL divergence term for the middle starting state (S_0), using $\lambda = 0.5$. (E) The evolution of the default policy shows acquisitions of soft habits.

Lastly, we note that KL term implements the computational logic of the APEs implemented by the TS by capturing the divergence between the entire behavioural policy from the default policy, across all actions. Observations regarding choice bias contralateral to the TS hemisphere are captured in the updating of the default policy for individual discrete actions [43], which is similar to value-free updates [33, 34].

Simulation result: Perseverative bias confers a value on stability against uncontrollability. What is the normative benefit of such APE-driven perseveration or stickiness in choices? We hypothesised that the bias towards π^d (soft habits) promotes learning stability during periods of environmental uncontrollability. We tested this by introducing sporadic episodes where choice-outcome contingencies were degraded (Fig. 7A). During these uncontrollable periods, agents using soft Q-learning or soft Q(λ) exhibited perseveration of prior choices and limited unlearning of the behavioural policy, due to the influence of the slowly updated default policy (Fig. 7B, C). This resulted in better overall reward accrual compared to standard Q-learning (which lacks this perseverative bias) across a range of τ values (Fig. 7D).

Thus, the APE-like KL term in our framework not only arises normatively, in a way that would scale to any action space (unlike other models which explicitly include an APE term [34, 36, 35]), but also confers a functional advantage of stable learning by promoting conservative policy updates. Unlike [33], it is not included as an additional ad-hoc controller, but normatively shapes the policy through the objective function, reconciling with the TDRL view of dopamine. Distinct from Miller et al. [34], these "soft habits" are not entirely value-free but rather accounted for in the value itself, through the augmented RL objective and confer a value on controllability.

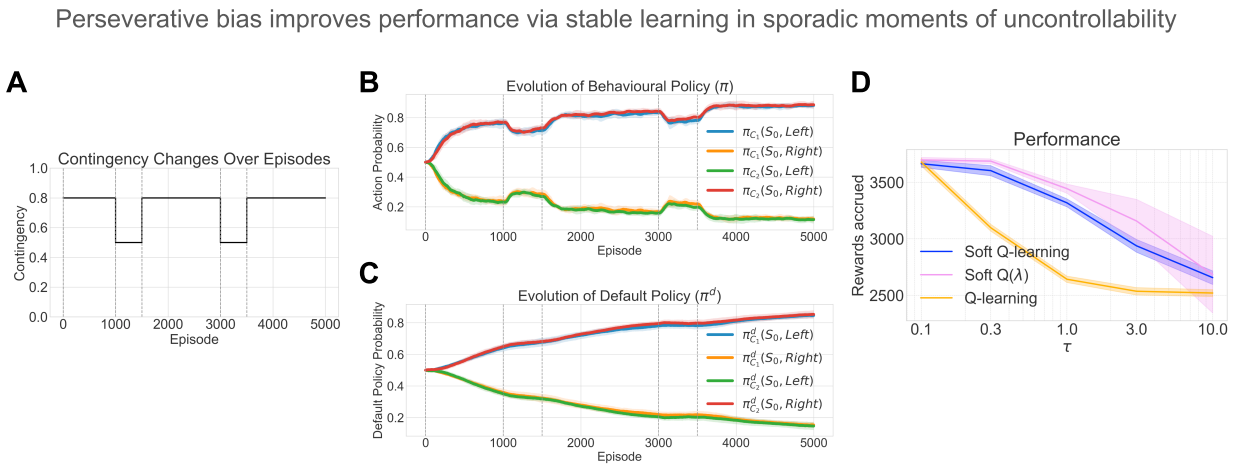

Figure 7: Perseverative bias confers a value on stability against uncontrollability. (A) Spurious contingency degradation to test the role of perseverative bias (B) Evolution of behavioural policy shows stickiness in actions (C) Evolution of the default policy. Both B & C are plotted for the soft Q-learning model with $\tau = 0.3$. (D) Soft Q-learning and Soft Q(λ) outperform Q-learning for a range of τ.

3 Discussion

In this work, we address the fundamental challenge of reliably optimising multiple objectives in reinforcement learning and propose a model that outperforms existing multi-objective methods. It invites a reconsideration of the dopaminergic system's computational goals: shifting from the classical view of cumulative discounted reward maximisation [1] towards optimising returns augmented by a KL-penalty in policies deviating from a default policy. This reframing not only yields distinct functional advantages but also generates novel testable predictions in efficient, safe and stable learning. In doing so, we resolve a major outstanding puzzle about how to reconcile conflicting views of the role of TS-projecting dopamine in threat prediction errors (TPEs) and action prediction errors APEs).

This approach was designed to directly address several limitations of the standard model, specifically the challenge of flexibly pursuing multiple, often conflicting, rewards without the learning process becoming unstable or inefficient—a known issue for classic on-policy methods. Our model's core innovation is to augment the classic reward prediction error with a regularisation term that imposes a 'cost' for deviating from a default policy. This single conceptual shift yields several powerful benefits; for instance, it confers a 'value on controllability,' which promotes stable learning. The framework further allows innate priors (e.g., for threat avoidance) to be flexibly expressed for safe behaviour. Crucially, this framework provides a clear normative function for the APE-like signals in the TS, uniting them with TPEs and RPEs under a single, coherent objective and thereby distinguishing our model from other multi-objective RL approaches.

At an algorithmic level, this work addresses how to ensure (often individually selfish) value functions cooperate effectively to drive reliable behaviour, despite constantly competing for control. This contrasts with "delegation" approaches [64, 65, 66], where only one value function controls, thus avoiding this problem. Regarding our optimal composition results, previous multi-objective TD(0) or SARSA approaches that scale on-policy prediction errors with weights (e.g. state-dependent RPE modulation [9, 67] or feature-specific weight updates [26, 63]) may slightly ameliorate unreliable learning or policy interference, but do not resolve it (supplementary Fig. S3). This excludes trivial conditions where only one value component is active, preventing interference. A better alternative might be to use importance sampling, which also ensures off-policy learning. However, it incurs higher update variance and cannot explain action prediction errors. Hence, we primarily utilise the Tree-Backup approach in our derivations for novel multi-step extensions of soft Q-learning (Supplementary Methods 3 and 4).

Off-policy learning is a prominent theme in this work, as it demonstrably improves performance over on-policy multi-objective RL algorithms and the successor representation, while also relating directly to its off-policy counterpart [43]. The brain may implement off-policy learning for several reasons: First, it prevents interference and unlearning between multiple values amidst changing priorities. Second, it facilitates learning amidst motor noise and competition from distributed control systems, like the motor cortex and cerebellum [35, 68]. Third, on-policy algorithms, such as

the successor representation, exhibit strong policy dependence where goal information contaminates the state map, hindering flexible transfer [59, 69], a problem solved by off-policy alternatives [43]. Fourth, the ability of episodic memories to utilise cached values [70, 71] or stale behavioural data for performance improvements points towards an underlying off-policy mechanism, akin to its necessity in deep Q-learning's episodic replay buffers [72]. However, finding strong neural and behavioural evidence for interference and unintended unlearning between two or more value systems (for different rewards) under changing priorities in a two-step task similar to Fig. 2D-F, would falsify our hypothesis of phasic dopamine performing off-policy multi-objective RL and find evidence for on-policy multi-objective RL (e.g. [9]). Rapid change in priorities could be potentially implemented as a task rule that needs to be inferred.

Our framework synthesises several threads of the evolving dopamine story, which has progressively expanded from a simple scalar reward signal [1] to a multifaceted control signal. The model naturally accommodates aversive and threat prediction errors [73, 23, 30, 31], aligning with multi-threaded and outcome-specific prediction error views [27, 9], and highlights their role in safe learning. This approach complements the feature-specific vector RPE model [63] while remaining compatible with it. Further, we propose an alternative account of within-target dopaminergic heterogeneity based on differentially initialised, composable values sharing a common outcome, distinct from that of Lee et al. [63]. Further, we find that such value compositions with different initialisations unify (reward-distorting) novelty bonuses and (non-distorting) shaping bonuses in TDRL [55], previously used to explain early observations of novelty responses in phasic dopamine [74, 75]. Our results (Fig. 3C) also highlight the behavioural inefficiencies of the SR model in replanning after overtraining [59]. Indeed, SR models of dopamine often need to treat reward as a feature to explain RPEs as a form of state prediction error [12], as Lee et al. [63] shows they otherwise fail to consistently respond to rewards. This highlights the benefit of outcome-specific models such as ours [and that of 9], which efficiently achieve values comparable to an SR/DR model [10, 43] when tuned to a subset of reward types. The introduction of belief-state weighted value composition bears resemblance to other contextual learning models [26, 25]; though not a focus of this work, this mechanism could be extended in the future to account for beliefs about reward timing [27] or state uncertainty, potentially explaining ramping effects [76, 77, 78]. Our derivation of eligibility traces allows the model to flexibly incorporate the presence or absence of a gradual shift in dopamine responses, with $\lambda = 0$ and higher learning rate or higher λ and lower learning rate, respectively [79, 80]. Finally, the soft Bellman value function links our work to a kind of risk-sensitive control (KL-control) due to its use of exponential utilities [81, 82, 42], as it inherently considers a spectrum of action values, not just the mean, with the temperature parameter τ modulating this sensitivity. Both our approach and distributional RL go beyond a single average value, but they do so differently, and future work is needed to compare with full distributional RL in outcomes [83] and time [84, 85, 86].

This work is the first, to our knowledge, to mathematically unify the conflicting views about the tail of the striatum's (TS) phasic dopamine. However, previous work has attempted to unify the general role of dopamine in learning and action inference [36]. Our proposal goes beyond the (non-mathematical but very helpful) unifying hypothesis that TS shifts attention to orient or avoid [37] by acknowledging the action prediction errors and their normative implications for stable learning. Biologically, one possibility is that threat and movement-related activity are encoded by separate dopaminergic subpopulations. In support of this, it has recently been reported that threat and acceleration-related dopamine responses are encoded in separate genetic subpopulations that both project to the TS, expressing Slc17a6 (also known as Vglut2) and Anxa1, respectively [87, 88, 89, 90, 33].

Our work introduces the concept of an inferred threat belief state (potentially cortical), demonstrating how it can titrate the balance between multiple values to guide flexible avoidance. This aligns with a performance effect, rather than a learning effect, in models of striatal direct (D1) and indirect (D2) pathway balance [91]. Our hypothesis, mapping different value initialisations to TS D1 and D2 pathways, yields testable predictions [32]. Furthermore, the flexible expression of innate values can reconcile associative and non-associative fear conditioning accounts [92], promoting more adaptable safe behaviour than previous models based on outcome uncertainty [53]. Crucially, we show that the same computations for efficient multi-reward acquisition can, with minor modifications, form a modular instrumental system for avoidance, unlike Pavlovian misbehaviour [53]. Lastly, our approach extends beyond existing models [31] by demonstrating approach-retreat bouts with associated TPEs, complementing risk-sensitive model-based RL efforts in modelling cautious behaviours [93].

In terms of limitations; first, though broadly applicable to decision-making under changing priorities, when applied to homeostatic priorities, it cannot explain physiological state-dependent modulation of prediction errors [94]. This is a limitation common to all multi-objective RL approaches (e.g. those by Dulberg et al. [8], Millidge et al. [9] and ours), which learn equally from all reward types, at all times. Second, despite its advantages, soft maximum composition optimises an objective different from a simple weighted sum of utilities. While this is not necessarily an issue for modelling homeostasis, where value representation is debated [15, 56, 8], and may even account for inhibitory effects of irrelevant drives [15, 95], we find it can fit poorly to human behaviour when participants explicitly maximise weighted sums of utilities or show generalisation in task structure rather than in values [17] (see supplementary Fig. S7). Third, while there are compelling reasons for the brain to implement off-policy learning [35, 68, 33], some early work [96, 97]

suggested phasic dopamine implements on-policy algorithms like SARSA. However, those experiments involved overtrained monkeys where no learning occurred, making observed TD error signals potentially epiphenomenal. Our framework could partly explain these differences in action values as the KL divergence from the overtrained default policy (APEs). Furthermore, the observed use of cached values in computing TDEs [70, 71] better aligns with off-policy rather than on-policy TD learning.

In offering a novel normative framework for multi-objective reinforcement learning, this paper re-conceptualises the computational role of striatal dopamine. Our findings demonstrate how the brain might achieve efficient, safe, and stable learning, simultaneously reconciling disparate experimental observations and generating testable predictions for future neuroscientific inquiry.

4 Methods

4.1 Reinforcement learning in MDPs

Let the environment be a Markov Decision Process, where at time $t = 0, 1, 2, ...$, the agent is in state $s_t \in \mathcal{S}$ and takes action $a_t \in \mathcal{A}$ and receives the next state $s_{t+1} \in \mathcal{S}$ and the reward $r_{t+1} = r(s_t, a_t) \in \mathcal{R}$ giving rise to trajectories $s_0, a_0, r_1, s_1, a_1, r_2, ...$ The dynamics of MDP are given by the conditional probability $p(s', r|s, a) \doteq \Pr(s_t = s', r_t = r|s_{t-1} = s, a_{t-1} = a)$.

The discounted return at time t is given by $G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$, where $\gamma \in [0, 1]$. Policy $\pi(a|s)$ is a mapping from states to the probabilities of choosing each possible action. The value function of a state s under the policy π is is the expected return when starting in s and following π thereafter, which is formalized as $V_\pi \doteq \mathbb{E}_\pi[G_t|s_t = s], \forall s \in \mathcal{S}$. Similarly, the value of taking action a in state s and following policy π thereafter is given by the Q-value or the action-value function, $Q_\pi(s, a) \doteq \mathbb{E}_\pi[G_t|s_t = s, a_t = a]$.

The Bellman equation of a value function v_π is a fundamental property in reinforcement learning expressing the recursive relationship between a value of state and the value of its possible successor states.

$$V_\pi(s) \doteq \mathbb{E}_{a\sim\pi(\cdot|s)}\mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r + \gamma V_\pi(s')], \forall s \in \mathcal{S} \tag{1}$$

Since value functions define a partial ordering over policies, there exists at least one optimal policy π^* that is better than all policies, where a policy $\pi \geq \pi'$ if and only if $V_\pi(s) \geq V_{\pi'}(s), \forall s \in \mathcal{S}$. The optimal state-value function is $V^*(s) \doteq \max_\pi V_\pi(s), \forall s \in \mathcal{S}$. Similarly, the optimal action-value function is $Q^*(s, a) \doteq \max_\pi Q_\pi(s, a) = \mathbb{E}[r_{t+1} + V^*(s_{t+1})|s_t = s, a_t = a]$. Once we have the optimal action-values, one can simply perform actions greedily to get the optimal policy $\pi^* = [\mathcal{G}Q^*](s) = \arg\max_a Q^*(s, a)$.

The recursive Bellman equations can also be written for the value function under the optimal policy, referred to as the Bellman optimality equations:

$$V^*(s) = \max_a \mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r + \gamma V^*(s')] \tag{2}$$

4.2 Entropy-regularised reinforcement learning in Linear MDPs

Entropy-regularised RL [41, 39, 98] augments the reward function with a term that penalises deviating from some default policy π^d, essentially making "soft" assumptions about the future policy (in the form of a stochastic action distribution). When π^d is an uniform policy, this reduces to max entropy reinforcement learning [49, 48]. The expected reward on taking action a_t in state s_t is given by $\mathbb{E}_{a_t\sim\pi}[r(s_t, a_t) - \tau D_{\mathrm{KL}}(\pi(\cdot|s_t)\|\pi^d(\cdot|s_t))]$, which can be further compactly written as $\mathbb{E}_{a_t\sim\pi}[r_{t+1} - \tau \mathrm{KL}(s_t)]$. Here, τ is the scalar temperature parameter, and $\mathrm{KL}(s_t)$ is the Kullback-Leibler divergence between the current policy π and a default policy π^d in state s_t. Thus, the entropy-augmented return is $G_t = \sum_{k=0}^{\infty} \gamma^k (r_{t+k+1} - \tau \mathrm{KL}(s_{t+k}))$.

The value function definitions under a policy π at any timestep t based on the entropy-augmented returns are as follows,

$$V_\pi(s) \doteq \mathbb{E}_\pi[G_t|s_t = s] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k (r_{t+k+1} - \tau \mathrm{KL}(s_{t+k}))\middle| s_t = s\right] \tag{3}$$

$$Q_\pi(s,a) \doteq \mathbb{E}_\pi[G_t|s_t=s,a_t=a] = \mathbb{E}_\pi\left[r_{t+1} + \sum_{k=1}^{\infty}\gamma^k(r_{t+k+1} - \tau\mathrm{KL}(s_{t+k}))\middle| s_t=s,a_t=a\right] \tag{4}$$

Note that this Q-function does not include the first KL penalty term ($\mathrm{KL}(s_t)$), as it does not depend on action action a_t which has already been chosen [49, 48, 50]. This gives the following relationship which holds for all policies π.

$$V_\pi(s) = \mathbb{E}_{a\sim\pi}[Q_\pi(s,a)] - \tau\mathrm{KL}(s) \tag{5}$$

The Bellman equation and the Bellman optimality equation are as follows:

$$V_\pi(s) \doteq \mathbb{E}_{a\sim\pi(\cdot|s)}\mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r - \tau\mathrm{KL}(s) + \gamma V_\pi(s')] \tag{6}$$

$$V^*(s) = \max_a \mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r - \tau\mathrm{KL}(s) + \gamma V^*(s')] \tag{7}$$

Note, unlike the greedy (deterministic) policy $[\mathcal{G}Q](s) = \arg\max_a Q(s,a)$ in standard RL, the greedy (stochastic) policy in entropy-regularised RL is the Boltzmann policy ($\pi^{\mathcal{B}}_Q$).

$$\pi^{\mathcal{B}}_Q(\cdot|s) = [\mathcal{G}Q](s) = \frac{\pi^d(a|s)\exp(Q(s,a)/\tau)}{\sum_{\mathcal{A}}\exp(Q_\pi(s,a')/\tau)\pi^d(a'|s)} \tag{8}$$

Prior work [41, 39, 48, 98] shows that this Boltzmann policy holds the two properties: (1) it is the optimal policy ($\pi^* = \pi^{\mathcal{B}}_{Q^*}$) i.e. it uniquely solves the Bellman optimality equations and (2) under the Boltzmann policy, the Bellman equation is equivalent to the "soft" Bellman equation, thus the value function $V_{\pi^{\mathcal{B}}_Q}(s) = V_Q(s)$, essentially performing a soft maximum operation over Q-values. These known results can be verified easily and for completeness purposes, we provide an intuitive explanation in Supplementary Methods 1.

$$\begin{aligned} V_Q(s) &= \tau\log\mathbb{E}_{a\sim\pi^d}\exp(Q_\pi(s,a)/\tau) \\ &= \tau\log\sum_{\mathcal{A}}\exp(Q_\pi(s,a)/\tau)\pi^d(a|s) \end{aligned} \tag{9}$$

Note, this log-sum-exp performs a soft maximum because, $\max\{x_1,...,x_n\} \leq \mathrm{softmax}(x_1,...,x_n) \leq \max\{x_1,...,x_n\} + \log(n)$.

4.3 Multi-objective reinforcement learning and optimal composition in Linear MDPs

Having discussed reinforcement learning (RL) in MDPs and Linear MDPs with single-attribute rewards, we now focus on multi-objective RL, which concerns multiple rewarding attributes $\mathbf{r} = [r_1, r_2, \ldots, r_n]$. The objective is to maximise a cumulative discounted return of a reward function composed of these attributes:

$$r_c(s,a) = f(r_1, r_2, \ldots, r_n; \mathbf{w}),$$

where $\mathbf{w}$ is a set of non-negative parameters that weight each attribute, satisfying $\sum w_i = 1$. We address the problem of *optimal compositions*: determining how the reward function should be composed of multiple attributes to motivate meaningful behaviour and how to compose value functions to ensure the resulting policy acts optimally with respect to the composed reward function.

As shown in Results section 2.2, a simple linear composition of Q-values, such as $w_1Q^*_1(s,a) + w_2Q^*_2(s,a) + \cdots + w_nQ^*_n(s,a)$, may not maximise the composed reward function

$$r_c(s,a) = w_1r_1(s,a) + w_2r_2(s,a) + \cdots + w_nr_n(s,a)$$

due to the non-linearity introduced by the *max* operation in the Bellman optimality equation in MDPs:

$$\begin{aligned} Q^*(s,a) &= \mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r + \gamma V^*(s')] \\ &= \mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r + \gamma \max_{a'} Q^*(s',a')]. \end{aligned} \tag{10}$$

However, in linear MDPs [40, 41, 44, 39], which replace the *max* operation over Q-values by a soft-maximum V_Q with respect to the default policy (see equation 9), optimal (softmax) and near-optimal (additive) compositions are possible. The Bellman optimality equation for Q-values in linear MDPs is as follows:

$$\begin{aligned} Q^*(s,a) &= \mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r + \gamma V^*(s')] \\ &= \mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r + \gamma \mathbb{E}_{a\sim\pi_Q^{\mathcal{B}}}[Q(s',a')] - \tau D_{\mathrm{KL}}[\pi_Q^{\mathcal{B}} \| \pi^d](s')] \\ &= \mathbb{E}_{(s',r)\sim p(s',r|s,a)}[r + \gamma V_Q(s')]. \end{aligned} \tag{11}$$

Theorem 1 (Optimal Softmax Composition) *[44, 98]*

Let $Q^*_{i,\tau}(s,a)$ be the optimal entropy-regularized Q-functions for individual rewards $r_i(s,a)$.

Then the reward function for the composed task is given by the log-sum-exp (soft maximum) of the individual reward functions:

$$r_c(s,a) = \tau \log \left(\sum_{i=1}^{n} w_i \exp\left(\frac{r_i(s,a)}{\tau} \right) \right), \tag{12}$$

where τ is a temperature parameter.

The optimal Q-function $Q^*_{c,\tau}(s,a)$ for the composed task is equal to the composition of Q-values $Q_{comp,\tau}(s,a)$ given by:

$$Q^*_{c,\tau}(s,a) = Q_{comp,\tau}(s,a) = \tau \log \left(\sum_{i=1}^{n} w_i \exp\left(\frac{Q^*_{i,\tau}(s,a)}{\tau} \right) \right), \tag{13}$$

where $Q^*_{i,\tau}$ are the optimal Q-functions for the individual tasks.

The theorem for additive composition in Linear MDPs is provided in the Supplementary Methods 2.

4.4 Off-policy model-free learning algorithms in Linear MDPs

Model-free algorithms do not assume a probabilistic model about state transitions and rewards but instead learn value functions through reward prediction errors, which can be implemented by phasic dopamine signals in the striatum. We focus on online algorithms, such as soft Q-learning [48], which update values continuously during episodes rather than waiting until the end, unlike offline algorithms like Z-learning [41], a Monte Carlo control algorithm. We further also prefer off-policy algorithms like Soft Q-learning and our subsequent extensions.

Soft Q-learning (One-Step)

We adopt soft Q-learning and extend it from the maximum entropy formulation to a relative entropy formulation. The Q-value update equation is given by:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha \delta_t, \tag{14}$$

where α is the learning rate, and δ_t is the reward prediction error at timestep t, defined as:

$$\delta_t = r_{t+1} + \gamma V_Q(s_{t+1}) - Q(s_t,a_t), \tag{15}$$

where V_Q is given by equation 9. Deep RL implementations inspired by Mnih et al. [72] may use a separate target network (e.g., $\underline{Q}$, resulting in $V_{\underline{Q}}$) to construct the loss function, which we exclude here for simplicity.

Multi-step Soft Q-learning

This section presents novel update rules for multi-step extensions of soft Q-learning, where the agent learns from multiple steps rather than the most immediate step. Under the assumption that the state action values are approximately unchanging [28], we can write the update rule for the N-step soft Q learning and its extension with eligibility traces, soft Q(λ) using TD-errors.

When following the Boltzmann policy, the N-step soft Q-learning is simply,

$$Q_{t+n}(s_t, a_t) \leftarrow Q_{t+n-1}(s_t, a_t) + \alpha \left(\sum_{k=t}^{min(T-1,t+n-1)} \gamma^k \delta_k \right). \tag{16}$$

Where T is the time step at which the episode terminated, Q_{t+n} denotes Q-value accessed or updated at timestep $t+n$ and the TD-errors are defined as follows. For $k = t$, the TD-error is given by equation 15. For $k > t$, it includes the KL divergence term and is given by,

$$\delta_k = r_{k+1} - \tau \mathrm{KL}_k + \gamma V_Q(s_{k+1}) - V_Q(s_k) \tag{17}$$

However, if one is following a behavioural policy that is not Boltzmann, we need a truly off-policy update rule. If the agent has access to the behavioural policy, then it can use importance sampling (detailed derivation provided in Supplementary Methods 3). However, this can lead to higher variance in the updates and requires access to the behavioural policy. Therefore we derive an alternative method using Tree Backup which does not require knowing the behavioural policy. The update rule is as follows,

$$Q_{t+n}(s_t, a_t) \leftarrow Q_{t+n-1}(s_t, a_t) + \alpha \left(\sum_{k=t}^{min(T-1,t+n-1)} \delta_k \prod_{i=t+1}^{k} \gamma \pi_Q^{\mathcal{B}}(a_i|s_i) \right). \tag{18}$$

We next extend these methods to incorporate eligibility traces. Under Boltzmann policy, the Q-value update rule is,

$$Q_{t+1}(s, a) \leftarrow Q_t(s, a) + \alpha \delta_t e_t(s, a) \;\; \forall s, a \tag{19}$$

and eligibility traces are updated as follows (in the tabular setting),

$$e_t(s, a) = \begin{cases} \gamma \lambda e_{t-1}(s, a) + 1, & \text{if } (s, a) = (s_t, a_t), \\ \gamma \lambda e_{t-1}(s, a), & \text{otherwise,} \end{cases} \tag{20}$$

The TD-error (δ_t) is the same as equation 17 (except substitute k with t). Note, this algorithm is entirely online.

For a full off-policy Soft Q(λ), we build upon the Tree Backup approach. The Q-value update rule and the TD-errors remain the same, but the eligibility trace updates are adjusted to include the target policy $\pi_Q^{\mathcal{B}}$,

$$e_t(s, a) = \begin{cases} \gamma \lambda \pi_Q^{\mathcal{B}}(a_t|s_t) e_{t-1}(s, a) + 1, & \text{if } (s, a) = (s_t, a_t), \\ \gamma \lambda \pi_Q^{\mathcal{B}}(a_t|s_t) e_{t-1}(s, a), & \text{otherwise,} \end{cases} \tag{21}$$

All detailed derivations for N-step soft Q-learning and Soft Q(λ) are provided in Supplementary methods 3 and 4, respectively.

4.5 Relationship to the default representation (DR)

The optimal values $V^*(s)$ are calculated using the default representation as follows [43, 41, 39] (in a model-based fashion):

$$\exp(\mathbf{V}^*/\tau) = \mathbf{M}\mathbf{P} \exp(\mathbf{r}/\tau) \tag{22}$$

where, $\mathbf{V}^*$ is the vector of optimal values at nonterminal states, $\mathbf{r}$ is the vector of rewards at terminal states, $\mathbf{P}$ is the one-step transition probabilities $\mathbf{T}_{NT}$ from non-terminal states to terminal states under the default policy π^d and $\mathbf{M}$ is the DR matrix defined as $\mathbf{M} = (\mathrm{diag}(\exp(-\mathbf{r}_N)/\tau) - \mathbf{T}_{NN})^{-1}$, where $\mathbf{r}_N$ is vector of rewards at non-terminal states and $\mathbf{T}_{NN}$ is the one-step transition probabilities between non-terminal states under the default policy π^d.

Further, [43] extend and define a more general version of the DR matrix D over all states (not just terminal states), as $\mathbf{D} = (\text{diag}(\exp(-\mathbf{r}_A)/\tau) - \mathbf{T})^{-1}$, where $\mathbf{r}_A$ is vector of rewards over all states and $\mathbf{T}$ are transition probabilities over all states under the default policy π^d. Here, $\mathbf{M}$ is a sub-block of $\mathbf{D}$ and $\mathbf{T}_{NT}$ and $\mathbf{T}_{NN}$ are sub-blocks of matrix $\mathbf{T}$. We observe that in both cases, it requires storing and/or learning the one-step transitions $\mathbf{T}$ under the default policy over all states, resulting in a $S \times S$ memory cost, where S is the size of the state space. Therefore, the memory cost of the DR scales quadratically with the size of the state space.

We propose our model (composition of multiple soft Q-learning rules) is intuitively akin to a compressed DR tuned to only relevant reward dimensions and converges to the same. To show this, we decompose the reward vector at terminal states, by assuming it to be composed using the optimal softmax composition [44], $\mathbf{r} = \tau \log\left(\sum_{i=1}^{n} w_i \exp\left(\frac{\mathbf{r}_i}{\tau}\right)\right)$. Therefore, the vector of optimal values using the DR will be:

$$\begin{aligned}
\exp(\mathbf{V}^*/\tau) &= \mathbf{MP}\left(\sum_{i=1}^{N} w_i \exp\left(\frac{\mathbf{r}_i}{\tau}\right)\right) \\
&= \sum_{i=1}^{N} w_i \left(\mathbf{MP} \exp\left(\frac{\mathbf{r}_i}{\tau}\right)\right) \\
&= \sum_{i=1}^{N} w_i \exp(\mathbf{V}_i^*/\tau)
\end{aligned} \tag{23}$$

Here, the vector of optimal values for each reward decomposition can be computed using soft Q-learning and consumes memory cost of S (state-space size). Therefore, for N reward decompositions, our method consumes a memory cost of $S \times N$. We believe the necessary reward decompositions would often be much lesser than the total number of states, $N << S$. Therefore, our method scales linearly with the size of the state space, whilst achieving the same optimal values as the DR for a predefined reward basis.

4.6 Simulation parameters

Simulations in Fig. 2D,E and supplementary Fig. S2, S3 use learning rate $\alpha = 0.1$, discount rate $\gamma = 1$, temperature $\tau = 1$ (unless explicitly varied) and are averaged over 100 runs. Simulations on efficient learning (Fig. 3) and safe learning (Fig. 5) use $\alpha = 0.1$, $\gamma = 0.99$, $\tau = 0.5$ for all algorithms. Fig. 3 results were averaged over 30 runs. For Fig. 5, we ran the simulations for 10^5 steps and then averaged over the time points that the agent was closest to the novel object and inside the grey zone to capture retreat start. In the safe learning experiment, we used a Bayesian approach to infer the belief state b_t, representing the probability of a "threatened" context c_t ($c_t = 1$ for "threatened" and $c_t = 0$ for "not threatened"). The belief state dynamically adjusted the weights $w_{\text{threatened}} = b_t$ for $Q_{\text{threatened}}$ and $w_{\text{not-threatened}} = 1 - b_t$ for $Q_{\text{not-threatened}}$.

Observations o_t were modelled as samples from a Bernoulli distribution, with likelihoods:

$$p(o_t = 1|c_t = 1) = 0.9, \quad p(o_t = 0|c_t = 1) = 0.1, \tag{24}$$

$$p(o_t = 1|c_t = 0) = 0.1, \quad p(o_t = 0|c_t = 0) = 0.9. \tag{25}$$

We assume the true context c_t depends on the agent's position. If the agent was in the vicinity of the novel object (demarcated as the grey states), the true context was $c_t = 1$ ("threatened"); otherwise, $c_t = 0$ ("not threatened"). Using these observations, the posterior distribution over c_t was modelled using a Beta distribution with parameters α_t and β_t. These parameters were updated iteratively with a decay term $\zeta = 0.1$ as follows:

$$\alpha_t = (1 - \zeta)\alpha_{t-1} + o_t, \quad \beta_t = (1 - \zeta)\beta_{t-1} + (1 - o_t). \tag{26}$$

Finally, the belief state b_t was computed as the mean of the Beta posterior distribution:

$$b_t = \frac{\alpha_t}{\alpha_t + \beta_t}. \tag{27}$$

The stable learning simulations slowly update the default policy, given by a delta rule [43, 34] upon taking action a in state s: $\hat{\pi}^d(a|s) \leftarrow \pi^d(a|s) + \alpha_d(1 - \pi^d(a|s))$ followed by normalising $\hat{\pi}^d(\cdot|s)$ to get the updated default policy $\pi^d(\cdot|s)$. Fig. 6, 7 use $\gamma = 0.99$, $\alpha = 0.1$ and $\alpha_d = 0.001$ (τ and λ mentioned in Figure captions, wherever applicable) and results are averaged over 10 runs.

Author Contributions

PM: Conceptualization, Investigation, Formal Analysis, Software, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing. BS: Funding Acquisition, Supervision, Writing – Review & Editing.

Acknowledgments

An earlier extended abstract of this work was presented at RLDM 2025, and the authors thank the two anonymous reviewers at RLDM 2025, Thomas Akam and Chris Summerfield for their helpful suggestions on how to make this work more impactful. Authors thank the funders: Wellcome Trust (214251/Z/18/Z, 203139/Z/16/Z and 203139/A/16/Z), IITP (MSIT 2019-0-01371) and JSPS (22H04998). This research was also partly supported by the NIHR Oxford Health Biomedical Research Centre (NIHR203316). The views expressed are those of the author(s) and not necessarily those of the NIHR or the Department of Health and Social Care. For the purpose of open access, the authors have applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

Code and Data availability

Code for all simulations is available at https://github.com/PranavMahajan25/OptCompMultValues.git. No data was collected during this study.

References

[1] Wolfram Schultz, Peter Dayan, and P Read Montague. A neural substrate of prediction and reward. *Science*, 275 (5306):1593–1599, 1997.

[2] Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

[3] Yael Niv and Geoffrey Schoenbaum. Dialogues on prediction errors. *Trends in cognitive sciences*, 12(7):265–272, 2008.

[4] Paul W Glimcher. Understanding dopamine and reinforcement learning: the dopamine reward prediction error hypothesis. *Proceedings of the National Academy of Sciences*, 108(supplement_3):15647–15654, 2011.

[5] Neir Eshel, Michael Bukwich, Vinod Rao, Vivian Hemmelder, Ju Tian, and Naoshige Uchida. Arithmetic and local circuitry underlying dopamine prediction errors. *Nature*, 525(7568):243–246, 2015.

[6] Mitsuko Watabe-Uchida, Neir Eshel, and Naoshige Uchida. Neural circuitry of reward prediction error. *Annual review of neuroscience*, 40(1):373–394, 2017.

[7] Zhewei Zhang, Kauê M Costa, Angela J Langdon, and Geoffrey Schoenbaum. The devilish details affecting tdrl models in dopamine research. *Trends in Cognitive Sciences*, 2025.

[8] Zack Dulberg, Rachit Dubey, Isabel M Berwian, and Jonathan D Cohen. Having multiple selves helps learning agents explore and adapt in complex changing worlds. *Proceedings of the National Academy of Sciences*, 120 (28):e2221180120, 2023.

[9] Beren Millidge, Yuhang Song, Armin Lak, Mark E Walton, and Rafal Bogacz. Reward bases: A simple mechanism for adaptive acquisition of multiple reward types. *PLOS Computational Biology*, 20(11):e1012580, 2024.

[10] Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993.

[11] Samuel J Gershman. The successor representation: its computational logic and neural substrates. *Journal of Neuroscience*, 38(33):7193–7200, 2018.

[12] Matthew PH Gardner, Geoffrey Schoenbaum, and Samuel J Gershman. Rethinking dopamine as generalized prediction error. *Proceedings of the Royal Society B*, 285(1891):20181645, 2018.

[13] Sindhu Padakandla, Prabuchandran KJ, and Shalabh Bhatnagar. Reinforcement learning algorithm for non-stationary environments. *Applied Intelligence*, 50(11):3590–3606, 2020.

[14] Mike JF Robinson and Kent C Berridge. Instant transformation of learned repulsion into motivational "wanting". *Current Biology*, 23(4):282–289, 2013.

[15] Mehdi Keramati and Boris Gutkin. Homeostatic reinforcement learning for integrating reward collection and physiological stability. *Elife*, 3:e04811, 2014.

[16] Ethan B Richman, Nicole Ticea, William E Allen, Karl Deisseroth, and Liqun Luo. Neural landscape diffusion resolves conflicts between needs across time. *Nature*, 623(7987):571–579, 2023.

[17] Momchil S Tomov, Eric Schulz, and Samuel J Gershman. Multi-task reinforcement learning in humans. *Nature Human Behaviour*, 5(6):764–773, 2021.

[18] Wolfram Schultz. Multiple reward signals in the brain. *Nature reviews neuroscience*, 1(3):199–207, 2000.

[19] Regina M Carelli, Stephanie G Ijames, and Alison J Crumling. Evidence that separate neural circuits in the nucleus accumbens encode cocaine versus "natural"(water and food) reward. *Journal of Neuroscience*, 20(11): 4255–4266, 2000.

[20] Kazuki Enomoto, Naoyuki Matsumoto, Sadamu Nakai, Takemasa Satoh, Tatsuo K Sato, Yasumasa Ueda, Hitoshi Inokawa, Masahiko Haruno, and Minoru Kimura. Dopamine neurons learn to encode the long-term value of multiple future rewards. *Proceedings of the National Academy of Sciences*, 108(37):15462–15467, 2011.

[21] Armin Lak, William R Stauffer, and Wolfram Schultz. Dopamine prediction error responses integrate subjective value from different reward dimensions. *Proceedings of the National Academy of Sciences*, 111(6):2343–2348, 2014.

[22] Yuji K Takahashi, Hannah M Batchelor, Bing Liu, Akash Khanna, Marisela Morales, and Geoffrey Schoenbaum. Dopamine neurons respond to errors in the prediction of sensory features of expected rewards. *Neuron*, 95(6): 1395–1405, 2017.

[23] Mitsuko Watabe-Uchida and Naoshige Uchida. Multiple dopamine systems: weal and woe of dopamine. In *Cold Spring Harbor Symposia on Quantitative Biology*, volume 83, pages 83–95. Cold Spring Harbor Laboratory Press, 2018.

[24] Jing-Yu Chang, Patricia H Janak, and Donald J Woodward. Comparison of mesocorticolimbic neuronal responses during cocaine and heroin self-administration in freely moving rats. *Journal of Neuroscience*, 18(8):3098–3115, 1998.

[25] Benedicte M Babayan, Naoshige Uchida, and Samuel J Gershman. Belief state representation in the dopamine system. *Nature communications*, 9(1):1891, 2018.

[26] Samuel J Gershman and Naoshige Uchida. Believing in dopamine. *Nature Reviews Neuroscience*, 20(11): 703–714, 2019.

[27] Yuji K Takahashi, Thomas A Stalnaker, Lauren E Mueller, Sevan K Harootonian, Angela J Langdon, and Geoffrey Schoenbaum. Dopaminergic prediction errors in the ventral tegmental area reflect a multithreaded predictive model. *Nature Neuroscience*, 26(5):830–839, 2023.

[28] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[29] William Menegas, Benedicte M Babayan, Naoshige Uchida, and Mitsuko Watabe-Uchida. Opposite initialization to novel cues in dopamine signaling in ventral and posterior striatum in mice. *elife*, 6:e21886, 2017.

[30] William Menegas, Korleki Akiti, Ryunosuke Amo, Naoshige Uchida, and Mitsuko Watabe-Uchida. Dopamine neurons projecting to the posterior striatum reinforce avoidance of threatening stimuli. *Nature neuroscience*, 21 (10):1421–1430, 2018.

[31] Korleki Akiti, Iku Tsutsui-Kimura, Yudi Xie, Alexander Mathis, Jeffrey E Markowitz, Rockwell Anyoha, Sandeep Robert Datta, Mackenzie Weygandt Mathis, Naoshige Uchida, and Mitsuko Watabe-Uchida. Striatal dopamine explains novelty-induced behavioral dynamics and individual variability in threat prediction. *Neuron*, 110(22):3789–3804, 2022.

[32] Iku Tsutsui-Kimura, Zhiyu Melissa Tian, Ryunosuke Amo, Yizhou Zhuo, Yulong Li, Malcolm G Campbell, Naoshige Uchida, and Mitsuko Watabe-Uchida. Dopamine in the tail of the striatum facilitates avoidance in threat–reward conflicts. *Nature Neuroscience*, pages 1–16, 2025.

[33] Francesca Greenstreet, Hernando Martinez Vergara, Yvonne Johansson, Sthitapranjya Pati, Laura Schwarz, Stephen C Lenzi, Jesse P Geerts, Matthew Wisdom, Alina Gubanova, Lars B Rollik, et al. Dopaminergic action prediction errors serve as a value-free teaching signal. *Nature*, pages 1–10, 2025.

[34] Kevin J Miller, Amitai Shenhav, and Elliot A Ludvig. Habits without values. *Psychological review*, 126(2):292, 2019.

[35] Jack Lindsey and Ashok Litwin-Kumar. Action-modulated midbrain dopamine activity arises from distributed control policies. *arXiv preprint arXiv:2207.00636*, 2022.

[36] Rafal Bogacz. Dopamine role in learning and action inference. *Elife*, 9, 2020.

[37] Isobel Green, Ryunosuke Amo, and Mitsuko Watabe-Uchida. Shifting attention to orient or avoid: a unifying account of the tail of the striatum and its dopaminergic inputs. *Current Opinion in Behavioral Sciences*, 59: 101441, 2024.

[38] Richard Bellman. On the theory of dynamic programming. *Proceedings of the national Academy of Sciences*, 38 (8):716–719, 1952.

[39] Emanuel Todorov. Efficient computation of optimal actions. *Proceedings of the national academy of sciences*, 106(28):11478–11483, 2009.

[40] Hilbert J Kappen. Linear theory for control of nonlinear stochastic systems. *Physical review letters*, 95(20): 200201, 2005.

[41] Emanuel Todorov. Linearly-solvable markov decision problems. *Advances in neural information processing systems*, 19, 2006.

[42] Krishnamurthy Dvijotham and Emanuel Todorov. A unifying framework for linearly solvable control. *arXiv preprint arXiv:1202.3715*, 2012.

[43] Payam Piray and Nathaniel D Daw. Linear reinforcement learning in planning, grid fields, and cognitive control. *Nature communications*, 12(1):4942, 2021.

[44] Emanuel Todorov. Compositionality of optimal control laws. *Advances in neural information processing systems*, 22, 2009.

[45] David Marr. *Vision: A computational investigation into the human representation and processing of visual information*. MIT press, 2010.

[46] Samuel J Gershman. Origin of perseveration in the trade-off between reward and complexity. *Cognition*, 204: 104394, 2020.

[47] E Thorndike. Biological lectures from the marine laboratory at woods' holl, usa, for 1899. *Nature*, 62:411, 1900.

[48] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International conference on machine learning*, pages 1352–1361. PMLR, 2017.

[49] Brian D Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University, 2010.

[50] John Schulman, Xi Chen, and Pieter Abbeel. Equivalence between policy gradients and soft q-learning. *arXiv preprint arXiv:1704.06440*, 2017.

[51] Pranav Mahajan, Peter Dayan, and Ben Seymour. Homeostasis after injury: How intertwined inference and control underpin post-injury pain and behaviour. *bioRxiv*, pages 2025–02, 2025.

[52] Stefan Elfwing and Ben Seymour. Parallel reward and punishment control in humans and robots: Safe reinforcement learning using the maxpain algorithm. In *2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 140–147. IEEE, 2017.

[53] Pranav Mahajan, Shuangyi Tong, Sang Wan Lee, and Ben Seymour. Balancing safety and efficiency in human decision making. *bioRxiv*, pages 2024–01, 2024.

[54] Zack Dulberg and Jonathan D Cohen. On the duality of pain and pleasure processing: Why two dimensions of valence may be better than one. *bioRxiv*, pages 2025–01, 2025.

[55] Sham Kakade and Peter Dayan. Dopamine: generalization and bonuses. *Neural Networks*, 15(4-6):549–559, 2002.

[56] Peter Dayan. "liking" as an early and editable draft of long-run affective value. *PLoS Biology*, 20(1):e3001476, 2022.

[57] Stuart J Russell and Andrew Zimdars. Q-decomposition for reinforcement learning agents. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 656–663, 2003.

[58] Tuomas Haarnoja, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. Composable deep reinforcement learning for robotic manipulation. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 6244–6251. IEEE, 2018.

[59] Evan M Russek, Ida Momennejad, Matthew M Botvinick, Samuel J Gershman, and Nathaniel D Daw. Predictive representations can link model-based reinforcement learning to model-free mechanisms. *PLoS computational biology*, 13(9):e1005768, 2017.

[60] Eric Wiewiora. Potential-based shaping and q-value initialization are equivalent. *Journal of Artificial Intelligence Research*, 19:205–208, 2003.

[61] Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pages 278–287, 1999.

[62] John G Mikhael and Rafal Bogacz. Learning reward uncertainty in the basal ganglia. *PLoS computational biology*, 12(9):e1005062, 2016.

[63] Rachel S Lee, Yotam Sagiv, Ben Engelhard, Ilana B Witten, and Nathaniel D Daw. A feature-specific prediction error model explains dopaminergic heterogeneity. *Nature Neuroscience*, pages 1–13, 2024.

[64] Peter Dayan and Geoffrey E Hinton. Feudal reinforcement learning. *Advances in neural information processing systems*, 5, 1992.

[65] Thomas G Dietterich. Hierarchical reinforcement learning with the maxq value function decomposition. *Journal of artificial intelligence research*, 13:227–303, 2000.

[66] Ronald Parr and Stuart Russell. Reinforcement learning with hierarchies of machines. *Advances in neural information processing systems*, 10, 1997.

[67] Maaike MH van Swieten and Rafal Bogacz. Modeling the effects of motivation on choice and learning in the basal ganglia. *PLoS Computational Biology*, 16(5):e1007465, 2020.

[68] Jack Lindsey, Jeffrey E Markowitz, Winthrop F Gillis, Sandeep Robert Datta, and Ashok Litwin-Kumar. Dynamics of striatal action selection and reinforcement learning. *bioRxiv*, 2024.

[69] Lucas Lehnert, Stefanie Tellex, and Michael L Littman. Advantages and limitations of using successor features for transfer in reinforcement learning. *arXiv preprint arXiv:1708.00102*, 2017.

[70] Brian F Sadacca, Joshua L Jones, and Geoffrey Schoenbaum. Midbrain dopamine neurons compute inferred and cached value prediction errors in a common framework. *elife*, 5:e13665, 2016.

[71] Timothy A Krausz, Alison E Comrie, Ari E Kahn, Loren M Frank, Nathaniel D Daw, and Joshua D Berke. Dual credit assignment processes underlie dopamine signals in a complex spatial environment. *Neuron*, 111(21): 3465–3478, 2023.

[72] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[73] Masayuki Matsumoto and Okihide Hikosaka. Two types of dopamine neuron distinctly convey positive and negative motivational signals. *Nature*, 459(7248):837–841, 2009.

[74] Tomas Ljungberg, Paul Apicella, and Wolfram Schultz. Responses of monkey dopamine neurons during learning of behavioral reactions. *Journal of neurophysiology*, 67(1):145–163, 1992.

[75] Jon C Horvitz, Tripp Stewart, and Barry L Jacobs. Burst activity of ventral tegmental dopamine neurons is elicited by sensory stimuli in the awake cat. *Brain research*, 759(2):251–258, 1997.

[76] Mark W Howe, Patrick L Tierney, Stefan G Sandberg, Paul EM Phillips, and Ann M Graybiel. Prolonged dopamine signalling in striatum signals proximity and value of distant rewards. *nature*, 500(7464):575–579, 2013.

[77] HyungGoo R Kim, Athar N Malik, John G Mikhael, Pol Bech, Iku Tsutsui-Kimura, Fangmiao Sun, Yajun Zhang, Yulong Li, Mitsuko Watabe-Uchida, Samuel J Gershman, et al. A unified framework for dopamine signals across timescales. *Cell*, 183(6):1600–1616, 2020.

[78] John G Mikhael, HyungGoo R Kim, Naoshige Uchida, and Samuel J Gershman. The role of state uncertainty in the dynamics of dopamine. *Current Biology*, 32(5):1077–1087, 2022.

[79] Ryunosuke Amo, Sara Matias, Akihiro Yamanaka, Kenji F Tanaka, Naoshige Uchida, and Mitsuko Watabe-Uchida. A gradual temporal shift of dopamine responses mirrors the progression of temporal difference error in machine learning. *Nature neuroscience*, 25(8):1082–1092, 2022.

[80] Ryunosuke Amo. Prediction error in dopamine neurons during associative learning. *Neuroscience Research*, 199:12–20, 2024.

[81] Ronald A Howard and James E Matheson. Risk-sensitive markov decision processes. *Management science*, 18 (7):356–369, 1972.

[82] Vivek S Borkar. Q-learning for risk-sensitive control. *Mathematics of operations research*, 27(2):294–311, 2002.

[83] Will Dabney, Zeb Kurth-Nelson, Naoshige Uchida, Clara Kwon Starkweather, Demis Hassabis, Rémi Munos, and Matthew Botvinick. A distributional code for value in dopamine-based reinforcement learning. *Nature*, 577 (7792):671–675, 2020.

[84] Pablo Tano, Peter Dayan, and Alexandre Pouget. A local temporal difference code for distributional reinforcement learning. *Advances in neural information processing systems*, 33:13662–13673, 2020.

[85] Margarida Sousa, Pawel Bujalski, Bruno F Cruz, Kenway Louie, Daniel C McNamee, and Joseph J Paton. A multidimensional distributional map of future reward in dopamine neurons. *Nature*, pages 1–9, 2025.

[86] Paul Masset, Pablo Tano, HyungGoo R Kim, Athar N Malik, Alexandre Pouget, and Naoshige Uchida. Multi-timescale reinforcement learning in the brain. *Nature*, pages 1–9, 2025.

[87] Maite Azcorra, Zachary Gaertner, Connor Davidson, Qianzi He, Hailey Kim, Shivathmihai Nagappan, Cooper K Hayes, Charu Ramakrishnan, Lief Fenno, Yoon Seok Kim, et al. Unique functional responses differentially map onto genetic subtypes of dopamine neurons. *Nature neuroscience*, 26(10):1762–1774, 2023.

[88] Gioele La Manno, Daniel Gyllborg, Simone Codeluppi, Kaneyasu Nishimura, Carmen Salto, Amit Zeisel, Lars E Borm, Simon RW Stott, Enrique M Toledo, J Carlos Villaescusa, et al. Molecular diversity of midbrain development in mouse, human, and stem cells. *Cell*, 167(2):566–580, 2016.

[89] Jean-Francois Poulin, Jian Zou, Janelle Drouin-Ouellet, Kwang-Youn A Kim, Francesca Cicchetti, and Rajeshwar B Awatramani. Defining midbrain dopaminergic neuron diversity by single-cell gene expression profiling. *Cell reports*, 9(3):930–943, 2014.

[90] Jean-Francois Poulin, Giuliana Caronia, Caitlyn Hofer, Qiaoling Cui, Brandon Helm, Charu Ramakrishnan, C Savio Chan, Daniel A Dombeck, Karl Deisseroth, and Rajeshwar Awatramani. Mapping projections of molecularly defined dopamine neuron subtypes using intersectional genetic approaches. *Nature neuroscience*, 21 (9):1260–1271, 2018.

[91] Anne GE Collins and Michael J Frank. Opponent actor learning (opal): modeling interactive effects of striatal dopamine on reinforcement learning and choice incentive. *Psychological review*, 121(3):337, 2014.

[92] Peter R Zambetti, Bryan P Schuessler, Bryce E Lecamp, Andrew Shin, Eun Joo Kim, and Jeansok J Kim. Ecological analysis of pavlovian fear conditioning in rats. *Communications Biology*, 5(1):1–11, 2022.

[93] Tingke Shen and Peter Dayan. Risking your tail: Modeling individual differences in risk-sensitive exploration using bayes adaptive markov decision processes. *eLife*, 13, 2024.

[94] Jackson J Cone, Samantha M Fortin, Jenna A McHenry, Garret D Stuber, James E McCutcheon, and Mitchell F Roitman. Physiological state gates acquisition and expression of mesolimbic reward prediction signals. *Proceedings of the National Academy of Sciences*, 113(7):1943–1948, 2016.

[95] A Dickinson and B Balleine. The role of learning in motivation in gallistel cr (ed.), steven's handbook of experimental psychology: Learning, motivation and emotion (vol. 3, pp. 497–534), 2002.

[96] Genela Morris, Alon Nevet, David Arkadir, Eilon Vaadia, and Hagai Bergman. Midbrain dopamine neurons encode decisions for future action. *Nature neuroscience*, 9(8):1057–1063, 2006.

[97] Yael Niv, Nathaniel D Daw, and Peter Dayan. Choice values. *Nature neuroscience*, 9(8):987–988, 2006.

[98] Benjamin Van Niekerk, Steven James, Adam Earle, and Benjamin Rosman. Composing value functions in reinforcement learning. In *International conference on machine learning*, pages 6401–6409. PMLR, 2019.

[99] Doina Precup. Eligibility traces for off-policy policy evaluation. *Computer Science Department Faculty Publication Series*, page 80, 2000.

[100] Kristopher De Asis, J Hernandez-Garcia, G Holland, and Richard Sutton. Multi-step reinforcement learning: A unifying algorithm. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Supplementary Methods

SM 1: How does the Boltzmann policy achieve the stochastic Bellman optimal policy in Linear MDPs?

We here aim to provide an intuitive explanation for known results. Consider the KL divergence between any policy π and the Boltzmann policy $\pi_Q^{\mathcal{B}}$ under some Q-values.

$$\begin{aligned} D_{\mathrm{KL}}(\pi(\cdot|s)\|\pi_Q^{\mathcal{B}}(\cdot|s)) &= \mathbb{E}_{a\sim\pi}[\log\pi(a|s) - \log\pi_Q^{\mathcal{B}}(a|s)] \\ &= \mathbb{E}_{a\sim\pi}[\log\pi(a|s) - \log\pi^d(a|s) - Q(s,a)/\tau + \log\mathbb{E}_{a\sim\pi^d}[Q(s,a)/\tau]] \\ &= D_{\mathrm{KL}}[\pi\|\pi^d](s) - \mathbb{E}_{a\sim\pi}[Q(s,a)/\tau] + \log\mathbb{E}_{a\sim\pi^d}[Q(s,a)/\tau] \end{aligned} \tag{28}$$

We can rearrange this equation and multiply by τ to get V_π (as per equation 5) on the left-hand side.

$$\mathbb{E}_{a\sim\pi}[Q(s,a)] - \tau D_{\mathrm{KL}}[\pi\|\pi^d](s) = \tau\log\mathbb{E}_{a\sim\pi^d}[Q(s,a)/\tau] - \tau D_{\mathrm{KL}}[\pi\|\pi_Q^{\mathcal{B}}](s) \tag{29}$$

Here, we can see that the left-hand side of the equation (i.e. $V_\pi(s)$)) is maximised with respect to π, during generalised policy iteration (GPI), when the KL term on the right-hand side is minimized (as the other term does not depend on π), and $D_{\mathrm{KL}}[\pi\|\pi_Q^{\mathcal{B}}](s)$ is minimized at $\pi = \pi_Q^{\mathcal{B}}$. After each GPI, as $D_{\mathrm{KL}}[\pi\|\pi_Q^{\mathcal{B}}](s)$ approaches zero at $\pi = \pi_Q^{\mathcal{B}}$, we observe that left-hand side of the equation is the "soft" Bellman value function $V_Q(s)$. This shows that for fixed Q-values, the Boltzmann policy is the stochastic greedy policy that maximises value. Under optimal Q-values, this greedy policy can lead to the Bellman optimal policy in linear MDPs.

During the generalised policy evaluation (GPE), these optimal Q-values can be learnt using any reinforcement learning algorithm with convergence guarantees. Repeating these generalised policy updates (GPE+GPI) will lead to the optimal policy π^* will be given by the Boltzmann policy $\pi_{Q^*}^{\mathcal{B}}$. This concludes our intuitive explanation.

SM 2: Theorem for additive composition in Linear MDPs

Theorem 2 (Additive Composition) *[58, 98]*

Let $Q^*_{1,\tau}(s,a)$ and $Q^*_{2,\tau}(s,a)$ be the optimal entropy-regularized Q-functions for two tasks with rewards $r_1(s,a)$ and $r_2(s,a)$.

Then the reward function for the composed task aimed to ensure both objectives are given by the average of the individual reward functions:

$$r_c(s,a) = \frac{r_1(s,a) + r_2(s,a)}{2}.$$

Let the composition of Q-values $Q_{comp,\tau}(s,a)$ be:

$$Q_{comp,\tau}(s,a) = \frac{Q^*_{1,\tau}(s,a) + Q^*_{2,\tau}(s,a)}{2}.$$

The optimal Q-function $Q^*_{c,\tau}(s,a)$ for the composed task is bounded by:

$$Q_{comp,\tau}(s,a) \geq Q^*_{c,\tau}(s,a) \geq Q_{comp,\tau}(s,a) - C^*_\tau(s,a),$$

where C^*_τ is a fixed point of

$$C^*_\tau = \tau\mathbb{E}_{s'\sim\rho(s,a)}\left[D_{\frac{1}{2}}(\pi_1^*(s)\|\pi_2^*(s)) + \max_{a'} C(s',a')\right],$$

where $\pi_i^*(s)$ is the optimal Boltzmann policy for task i, and $D_{\frac{1}{2}}(\cdot\|\cdot)$ is the Rényi divergence of order $\frac{1}{2}$.

SM 3: Novel derivations extending Soft Q-learning to N-step soft Q-learning

In this section, we provide a detailed derivation of how soft Q-learning can be extended to N-step soft Q-learning. We will first begin with the on-policy setting, under the special case of Boltzmann policy (the stochastic optimal policy) and then extend it to a fully off-policy algorithm.

N-step Soft Q-learning (on-policy with Boltzmann policy)

N-step soft Q-learning incorporates multiple future rewards and KL penalties for deviating from the default policy, starting from the second time step onward.

The N-step return at time t, after taking an action a_t in state s_t is defined as:

$$G_{t:t+n} \doteq r_{t+1} + \gamma(r_{t+2} - \tau \mathrm{KL}_{t+1}) + \gamma^2(r_{t+3} - \tau \mathrm{KL}_{t+2}) + \ldots + \gamma^{n-1}(r_{t+n} - \tau \mathrm{KL}_{t+n-1}) + \gamma^n V_Q(s_{t+n}), \quad (30)$$

Note that the KL penalty terms appear only from the second timestep onward, as the cost of deviating from the default policy affects subsequent actions. If the episode terminates at timestep T, which can be less than $t+n$, then we will see next that the summation of TD-errors is appropriately truncated to $min(T-1, t+n-1)$.

We can rewrite $G_{t:t+n}$ in terms of the temporal difference (TD) error δ, by adding and subtracting $\gamma V_Q(s_{t+1})$, $\gamma^2 V_Q(s_{t+2})$, $\gamma^3 V_Q(s_{t+3})$ and so on:

$$\begin{aligned} G_{t:t+n} = (r_{t+1} + \gamma V_Q(s_{t+1})) + \gamma(r_{t+2} - \tau \mathrm{KL}_{t+1} + \gamma V_Q(s_{t+2}) - V_Q(s_{t+1})) \\ + \ldots + \gamma^{n-1}(r_{t+n} - \tau \mathrm{KL}_{t+n-1} + \gamma V_Q(s_{t+n}) - V_Q(s_{t+n-1})). \end{aligned} \quad (31)$$

Simplifying, we obtain:

$$\begin{aligned} G_{t:t+n} &= (r_{t+1} + \gamma V_Q(s_{t+1})) + \sum_{k=t+1}^{min(T-1,t+n-1)} \gamma^k \delta_k \\ &= Q_{t-1}(s_t, a_t) + (r_{t+1} + \gamma V_Q(s_{t+1}) - (Q_{t-1}(s_t, a_t)) + \sum_{k=t+1}^{min(T-1,t+n-1)} \gamma^k \delta_k \\ &= Q_{t-1}(s_t, a_t) + \sum_{k=t}^{min(T-1,t+n-1)} \gamma^k \delta_k \end{aligned} \quad (32)$$

where the TD error δ_k at each timestep is given as follows.

If $k = t$, the same as soft Q-learning:

$$\delta_t = r_{t+1} + \gamma V_Q(s_{t+1}) - Q(s_t, a_t) \quad (33)$$

For $k \geq t$,

$$\delta_k = r_{k+1} - \tau \mathrm{KL}_k + \gamma V_Q(s_{k+1}) - V_Q(s_k) \quad (34)$$

The first TD error term, $\delta_t = r_{t+1} + \gamma V_Q(s_{t+1}) - Q(s_t, a_t)$, does not include the KL penalty since it doesn't depend on the action a_t which has already been chosen [49, 48, 50].

Thus, the N-step soft Q-learning update rule is defined as:

$$Q_{t+n}(s_t, a_t) \leftarrow Q_{t+n-1}(s_t, a_t) + \alpha \left(G_{t:t+n} - Q_{t+n-1}(s_t, a_t) \right), \quad (35)$$

where α is the learning rate. The subscripts denote the timestep in the episode when the Q-value was used or updated. Note that n-step returns for $n > 1$ involve future rewards and states that are not available at the time of transition from t to $t+1$. Thus, the first Q-update of state s_t is performed at timestep $t+n$ and not t.

If the approximate action-values are unchanging, i.e. $Q_{t-1}(s_t, a_t) \simeq Q_{t+n-1}(s_t, a_t)$ (similar to Exercise 7.11 in Sutton and Barto [28]), then we can substitute the expression for $G_{t:t+n}$ to get:

$$Q_{t+n}(s_t, a_t) \leftarrow Q_{t+n-1}(s_t, a_t) + \alpha \left(\sum_{k=t}^{min(T-1,t+n-1)} \gamma^k \delta_k \right). \tag{36}$$

If the approximate action values are changing, then we will have an additional term of $Q_{t-1}(s_t, a_t) - Q_{t+n-1}(s_t, a_t)$ in the update.

N-step Soft Q-learning (off-policy with importance sampling)

We can now extend this to an off-policy algorithm that learns the Boltzmann policy ($\pi_Q^{\mathcal{B}}$) as the target policy while collecting data under any behavioural policy b. Considering that soft Q-learning is akin to expected SARSA for relative-entropy regularised objective, this derivation is similar to the N-step expected SARSA derivation [28].

We define the importance sampling ratio as follows (T is the last time step of the episode),

$$\rho_{t:h} = \prod_{k=t}^{min(h,T-1)} \frac{\pi_Q^{\mathcal{B}}(a_k|s_k)}{b(a_k|s_k)} \tag{37}$$

Now the update from the previous subsection can be replaced with its off-policy form,

$$Q_{t+n}(s_t, a_t) \leftarrow Q_{t+n-1}(s_t, a_t) + \alpha \rho_{t+1:t+n-1} \left(G_{t:t+n} - Q_{t+n-1}(s_t, a_t) \right), \tag{38}$$

$$Q_{t+n}(s_t, a_t) \leftarrow Q_{t+n-1}(s_t, a_t) + \alpha \rho_{t+1:t+n-1} \left(\sum_{k=t}^{t+n-1} \gamma^k \delta_k \right). \tag{39}$$

where, δ_{t+k} is defined as per equations 33 and 34. Note, we use $\rho_{t+1:t+n-1}$ and not $\rho_{t+1:t+n}$ as in any N-step expected SARSA such as this one, all possible actions are taken into account in the last state; the one actually taken has no effect and does not have to be corrected for [28, Page 150]. One can further write this recursively using per-decision importance sampling [28, 99], but it is not essential to our derivations.

N-step Soft Q-learning (off-policy with Tree Backup)

We next present N-step Soft Q-learning using the Tree Backup algorithm. N-step soft Q-learning with importance sampling only uses the expectation over actions in the last time step. Tree Backup instead uses it at every step. This provides the following advantages: (1) reduces the variance due to the importance sampling ratio, (2) an importance sampling ratio does not need to be computed, thus the behavioural policy b does not need to be stationary, Markov, or even known [100, 99].

We begin by writing the N-step return under the Boltzmann policy after taking action a_t in state s_t in the Tree Backup format. Note, this is the soft-Bellman optimal return regardless of the behavioural policy which chooses actions $a_t, a_{t+1}, a_{t+2}, ...$ leading to states $s_{t+1}, s_{t+2}, s_{t+3}, ...$ respectively.

$$G_{t:t+n} \doteq r_{t+1} + \gamma V_{\pi_Q^{\mathcal{B}}}(s_{t+1}) \tag{40}$$

Using equation 5, we get,

$$G_{t:t+n} \doteq r_{t+1} + \gamma \left(\sum_a \pi_Q^{\mathcal{B}}(a|s_{t+1}) Q_t(s_{t+1}, a) - \tau \mathrm{KL}_{t+1} \right) \tag{41}$$

We can now write it in Tree-Backup format,

$$
\begin{aligned}
G_{t:t+n} \doteq\ & r_{t+1} + \gamma \sum_{a \neq a_{t+1}} \pi_Q^{\mathcal{B}}(a|s_{t+1}) Q_t(s_{t+1}, a) - \gamma\tau \mathrm{KL}_{t+1} \\
& + \gamma \pi_Q^{\mathcal{B}}(a_{t+1}|s_{t+1}) \left(r_{t+2} - \tau \mathrm{KL}_{t+1} + \gamma \sum_{a \neq a_{t+2}} \pi_Q^{\mathcal{B}}(a|s_{t+2}) Q_{t+1}(s_{t+2}, a) - \gamma\tau \mathrm{KL}_{t+2} \right) \\
& + \gamma^2 \pi_Q^{\mathcal{B}}(a_{t+2}|s_{t+2}) \pi_Q^{\mathcal{B}}(a_{t+1}|s_{t+1}) \left(r_{t+3} - \tau \mathrm{KL}_{t+2} + \gamma \sum_{a \neq a_{t+3}} \pi_Q^{\mathcal{B}}(a|s_{t+3}) Q_{t+2}(s_{t+3}, a) - \gamma\tau \mathrm{KL}_{t+3} \right) \\
& + ... \\
& + \gamma^{n-1} \prod_{i=t+1}^{min(t+n-1,T-1)} \pi_Q^{\mathcal{B}}(a_i|s_i) \left(r_{t+n} - \mathrm{KL}_{t+n-1} + \gamma \sum_a \pi_Q^{\mathcal{B}}(a|s_{t+n}) Q_{t+n-1}(s_{t+n}, a) - \gamma\tau \mathrm{KL}_{t+n} \right)
\end{aligned} \tag{42}
$$

This is visualised as follows: The update is from the estimated action values of the leaf nodes of the tree. The action nodes in the interior, corresponding to the actual actions taken, do not participate. Each leaf node contributes to the target with a weight proportional to its probability of occurring under the target policy.

This can now be written recursively as,

$$
G_{t:t+n} \doteq r_{t+1} + \gamma \sum_{a \neq a_{t+1}} \pi_Q^{\mathcal{B}}(a|s_{t+1}) Q_t(s_{t+1}, a) + \gamma \pi_Q^{\mathcal{B}}(a_{t+1}|s_{t+1}) (G_{t+1:t+n} - \tau \mathrm{KL}_{t+1}) \tag{43}
$$

Alternatively, it can also be compactly written in terms of temporal difference errors, by using the following relation from equations 9 and 29:

$$
\begin{aligned}
\sum_{a \neq a_k} \pi_Q^{\mathcal{B}}(a|s_k) Q_{k-1}(s_k, a) &= \sum_a \pi_Q^{\mathcal{B}}(a|s_k) Q_{k-1}(s_k, a) - \pi_Q^{\mathcal{B}}(a_k|s_k) Q_{k-1}(s_k, a_k)) \\
&= V_Q(s_k) + \tau \mathrm{KL}_k - \pi_Q^{\mathcal{B}}(a_k|s_k) Q_{k-1}(s_k, a_k))
\end{aligned} \tag{44}
$$

By substituting this relation in equations 42, the $\tau \mathrm{KL}_k$ terms cancel out and we can write the Tree-Backup return in terms of TD-errors as follows:

$$
\begin{aligned}
G_{t:t+n} \doteq\ & r_{t+1} + \gamma \left(V_Q(s_{t+1}) - \pi_Q^{\mathcal{B}}(a_{t+1}|s_{t+1}) Q_t(s_{t+1}, a_{t+1}) \right) \\
& + \gamma \pi_Q^{\mathcal{B}}(a_{t+1}|s_{t+1}) \left(r_{t+2} - \tau \mathrm{KL}_{t+1} + \gamma V_Q(s_{t+2}) - \gamma \pi_Q^{\mathcal{B}}(a_{t+2}|s_{t+2}) Q_{t+1}(s_{t+2}, a_{t+2}) \right) \\
& + \gamma^2 \pi_Q^{\mathcal{B}}(a_{t+2}|s_{t+2}) \pi_Q^{\mathcal{B}}(a_{t+1}|s_{t+1}) \left(r_{t+3} - \tau \mathrm{KL}_{t+2} + \gamma V_Q(s_{t+3}) - \gamma \pi_Q^{\mathcal{B}}(a_{t+3}|s_{t+3}) Q_{t+2}(s_{t+3}, a_{t+3}) \right) \\
& + ... \\
& + \gamma^{n-1} \left[\prod_{i=t+1}^{min(t+n-1,T-1)} \pi_Q^{\mathcal{B}}(a_i|s_i) \right] \left(r_{t+n} - \mathrm{KL}_{t+n-1} + \gamma V_Q(s_{t+n}) - \gamma \pi_Q^{\mathcal{B}}(a_{t+n}|s_{t+n}) Q_{t+n-1}(s_{t+n}, a_{t+n}) \right)
\end{aligned} \tag{45}
$$

If we combine $r_{k+1} - \tau KL_k + V_Q(s_{k+1})$ with the last term of the (previous) k-th term, and add and subtract $Q(s_t, a_t)$ for the first term, then we have the following.

$$
G_{t:t+n} = Q_{t-1}(s_t, a_t) + \sum_{k=t}^{min(T-1,t+n-1)} \left[\delta_k \prod_{i=t+1}^{k} \gamma \pi_Q^{\mathcal{B}}(a_i|s_i) \right] - \gamma^n Q_{t+n-1}(s_{t+n}, a_{t+n}) \prod_{i=t+1}^{min(T-1,t+n-1)} \pi_Q^{\mathcal{B}}(a_i|s_i) \tag{46}
$$

If the $t+n-1 > T-1$, that is, the last state is terminal, then we can set the last Q-term to zero, and this expression simplifies to,

$$G_{t:t+n} = Q_{t-1}(s_t, a_t) + \sum_{k=t}^{T-1} \left[\delta_k \prod_{i=t+1}^{k} \gamma \pi_Q^{\mathcal{B}}(a_i|s_i) \right] \tag{47}$$

Again, if we assume the approximate Q-values are unchanging (similar to Exercise 7.11 in Sutton and Barto [28]), then this gives us our Q-update equation as follows,

$$Q_{t+n}(s_t, a_t) \leftarrow Q_{t+n-1}(s_t, a_t) + \alpha \left(\sum_{k=t}^{min(T-1,t+n-1)} \delta_k \prod_{i=t+1}^{k} \gamma \pi_Q^{\mathcal{B}}(a_i|s_i) \right). \tag{48}$$

where δ_k are defined as per equations 33 and 34. These updates lead to the estimation of off-policy multi-step returns under any behavioural policy, without knowing the behavioural policy.

Note, that if one starts the Tree Backup derivation with $V_Q(s_t+1)$ instead of $V_{\pi_Q^{\mathcal{B}}}(s_{t+1})$, then this leads to an alternate equivalent derivation in terms of the default policy instead of the Boltzmann policy (which requires calculating TD-errors under the default policy as well). We think this alternate derivation is less relevant as the agent is the target policy for the agent is the soft-Bellman optimal Boltzmann policy; therefore, we focus on the derivation in terms of the Boltzmann policy.

This concludes our novel derivations of off-policy N-step extensions of Soft Q-learning, using either importance sampling or Tree-Backup. One may further aspire to unify these two multi-step off-policy methods, as done in the standard RL setting by De Asis et al. [100], but it is not essential to the current work and is left as future work.

SM 4: Novel derivations extending N-step soft Q-learning to an elegant algorithm with eligibility traces

Soft Q(λ) (on-policy with Boltzmann policy)

Here, we build upon the N-step Soft Q-learning results to develop Soft Q(λ), a solution using eligibility traces.

We define a λ-return, which is the weighted summation of n-step returns [28].

$$G_t^{\lambda} = (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} G_{t:t+n} \tag{49}$$

To simplify the derivation, we define the Boltzmann backup operator following Schulman et al. [50],

$$\begin{aligned} [\mathcal{T}_{\pi_Q^{\mathcal{B}}} Q](s,a) &= \mathbb{E}_{(s',r)\sim p(s',r|s,a)} \left[r + \gamma\tau \log \mathbb{E}_{a'\sim\pi^d}[\exp(Q(s',a')/\tau)] \right] \\ &= \mathbb{E}_{(s',r)\sim p(s',r|s,a)} \left[r + \gamma V_Q(s')] \right] \end{aligned} \tag{50}$$

We can now define the SARSA(λ) version of this backup operator under the Boltzmann policy, $[\mathcal{T}_{\pi_{\mathcal{Q}}^{\mathcal{B}},\lambda} \mathcal{Q}](s,a)$, as follows.

$$G_t^{\lambda} = [\mathcal{T}_{\pi_Q^{\mathcal{B}},\lambda} Q] = (1-\lambda)(1 + \lambda \mathcal{T}_{\pi_Q^{\mathcal{B}}} + (\lambda \mathcal{T}_{\pi_Q^{\mathcal{B}}})^2 + ...) \mathcal{T}_{\pi_Q^{\mathcal{B}}} Q \tag{51}$$

Based on n-step methods, we can derive it to be,

$$G_t^{\lambda} = [\mathcal{T}_{\pi_Q^{\mathcal{B}},\lambda} Q](s,a) = Q(s,a) + \mathbb{E}\left[\sum_{k=t}^{\infty} (\gamma\lambda)^k \delta_k \right] \tag{52}$$

where,

$$\delta_k = r_{k+1} - \tau \mathrm{KL}_k + \gamma V_Q(s_{k+1}) - V_Q(s_k) \tag{53}$$

The update rule using G_t^λ , with a forward-view but offline algorithm is,

$$Q_{t+1}(s,a) \leftarrow Q_t(s,a) + \alpha(G_t^\lambda - Q_t(s,a)) \tag{54}$$

This can be approximated using a backwards view (SARSA(λ)-like) online algorithm under the Boltzmann policy, with eligibility traces (e_t) and the TD-errors as mentioned above in equation 53 (δ_t).

$$Q_{t+1}(s,a) \leftarrow Q_t(s,a) + \alpha\delta_t e_t(s,a) \;\; \forall s,a \tag{55}$$

and eligibility traces are updated as follows (in the tabular setting),

$$e_t(s,a) = \begin{cases} \gamma\lambda e_{t-1}(s,a) + 1, & \text{if } (s,a) = (s_t,a_t), \\ \gamma\lambda e_{t-1}(s,a), & \text{otherwise,} \end{cases} \tag{56}$$

Soft Q(λ) (off-policy with Tree Backup)

We next extend the algorithm to a full off-policy algorithm, developing upon the n-step method using the Tree Backup algorithm.

$$G_t^\lambda \approx Q(s,a) + \left[\sum_{k=t}^{\infty} \delta_k \sum_{i=t+1}^{k} \gamma_i \lambda_i \pi_Q^{\mathcal{B}}(a_i|s_i)\right] \tag{57}$$

Which gives us an online off-policy soft Q(λ) algorithm, similar to the previous one, but the eligibility trace update is adjusted with the target policy $\pi_Q^{\mathcal{B}}$,

$$Q_{t+1}(s,a) \leftarrow Q_t(s,a) + \alpha\delta_t e_t(s,a) \;\; \forall s,a \tag{58}$$

where,

$$\delta_t = r_{t+1} - \tau \text{KL}_t + \gamma V_Q(s_{t+1}) - V_Q(s_t) \tag{59}$$

and,

$$e_t(s,a) = \begin{cases} \gamma\lambda \pi_Q^{\mathcal{B}}(a_t|s_t) e_{t-1}(s,a) + 1, & \text{if } (s,a) = (s_t,a_t), \\ \gamma\lambda \pi_Q^{\mathcal{B}}(a_t|s_t) e_{t-1}(s,a), & \text{otherwise,} \end{cases} \tag{60}$$

This concludes our derivation of a basic online off-policy Soft Q(λ) algorithm. Such algorithms can be extended to (1) function approximation, (2) a more "true" online algorithm and (3) more stable algorithms following Chapter 12 in Sutton and Barto [28].

Supplementary Results and Figures

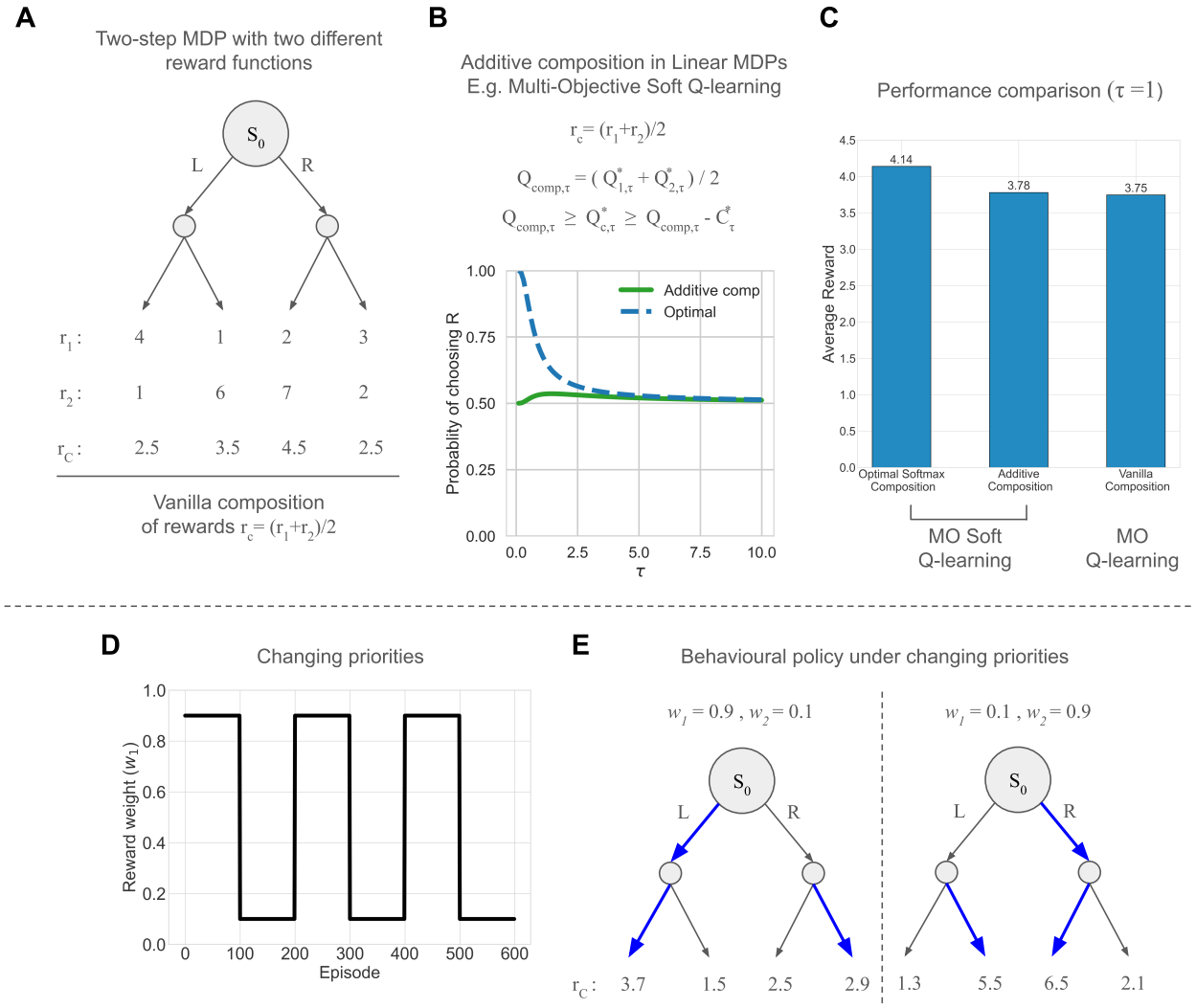

Figure S1: MO Q-learning comparison: (A) Two-step MDP with two (diverging) reward functions from the main text. (B) Action selection probabilities under additive composition in Linear MDPs, which perform better than additive composition in MDPs (standard RL) for $\tau > 0$. (C) Multi-objective (MO) soft Q-learning leads to better performance than MO Q-learning in this task. MO SARSA comparison: (D) Reward weight w_1, denoting change in priorities and $w_2 = 1 - w_1$. (E) Distinct behavioural policies under different priorities, which then affect the valuation of future states. Bold blue lines indicate the preferred action, and r_c in this figure is calculated under vanilla/additive composition of rewards.

Additional info accompanying Fig. S1: Haarnoja et al. [58] shows that simple additive composition of Q-functions (Q_{comp}) never overestimates Q_c^* by more than the divergence of the constituent policies. Here, constant C^* is the "value" of an adversarial policy that seeks to maximise this divergence (Theorem 2, Supplementary Methods 2). For $\tau = 0$, the behaviour of such additive composition resembles standard Q-learning (Fig. 2B), but, for $\tau > 0$, there is a weak bias towards the optimal action $a = R$, even with adversarially chosen rewards r_1, r_2 designed for maximal divergence. To our knowledge, equivalent performance guarantees are unavailable for standard multi-objective Q-learning. While mathematically distinct from a simple weighted sum, the weighted soft maximum approach empirically often yields a higher average weighted sum of rewards than alternatives in our simulations (supplementary Fig. S1). This advantage is not always guaranteed in all kinds of tasks and depends on τ and MDP utilities. However, these differences could be

quantified behaviourally by examining aggregate choices/consumption (adjusted for subjective utilities, [21]), especially under stable reward priorities.

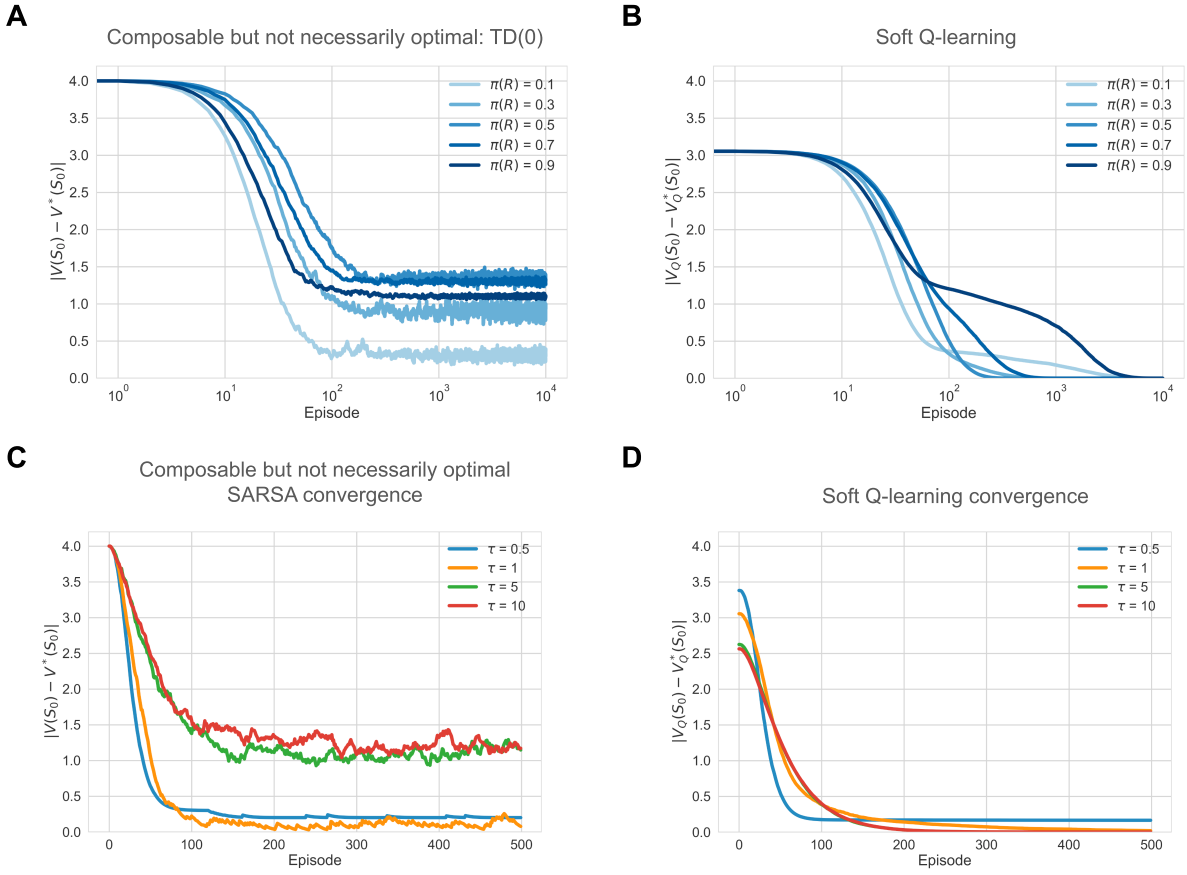

Figure S2: Demonstration of the off-policy learning of optimal values in linear MDP under sub-optimal trajectories (A & B) and under optimal control at different τ (C & D). (A) TD(0) learns the value of the policy used for data collection but fails to learn the optimal policy from the collected data; (B) Soft Q-learning learns the optimal value under any policy. We use random policies with static probabilities, $\pi(a = R) \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $\pi(a = L) = 1 - \pi(a = R)$. Other metaparameters are $\tau = 1$, learning rate = 0.1 and π^d as a uniform distribution, but the results are not affected by changing any of these. (C) SARSA (aka TD-control) learns the value of the exploring policy used for data collection, but fails to learn the optimal policy from the collected data. (D) Soft Q-learning learns the optimal value under any policy. Note, the plot has different values at episode 0, because the $V_Q^*(S_0)$ is dependent on τ. All plots were averaged over 30 runs, values were initialised to 0, and the learning rate was set to 0.1.

Figure S3: MO SARSA with state-dependent TD-errors, uses $\tilde{\delta}_i = w_i \delta_i$, same as the second model from Millidge et al. [9]. Thus the update is $\Delta Q_i = \alpha w_i \tilde{\delta}_i = \alpha w_i^2 \delta_i$ instead of simply $\Delta Q_i = \alpha \delta_i$. We find this does increase stability/reliability, but may not converge to the optimal values. MO SARSA with state-dependent updates, simply include it as $\Delta Q_i = \alpha w_i \delta_i$, similar to how it shows up in feature-based accounts. We find that this does not quite improve the stability. Neither can completely avoid the interference and unintended unlearning effect caused by the on-policy nature of MO SARSA.

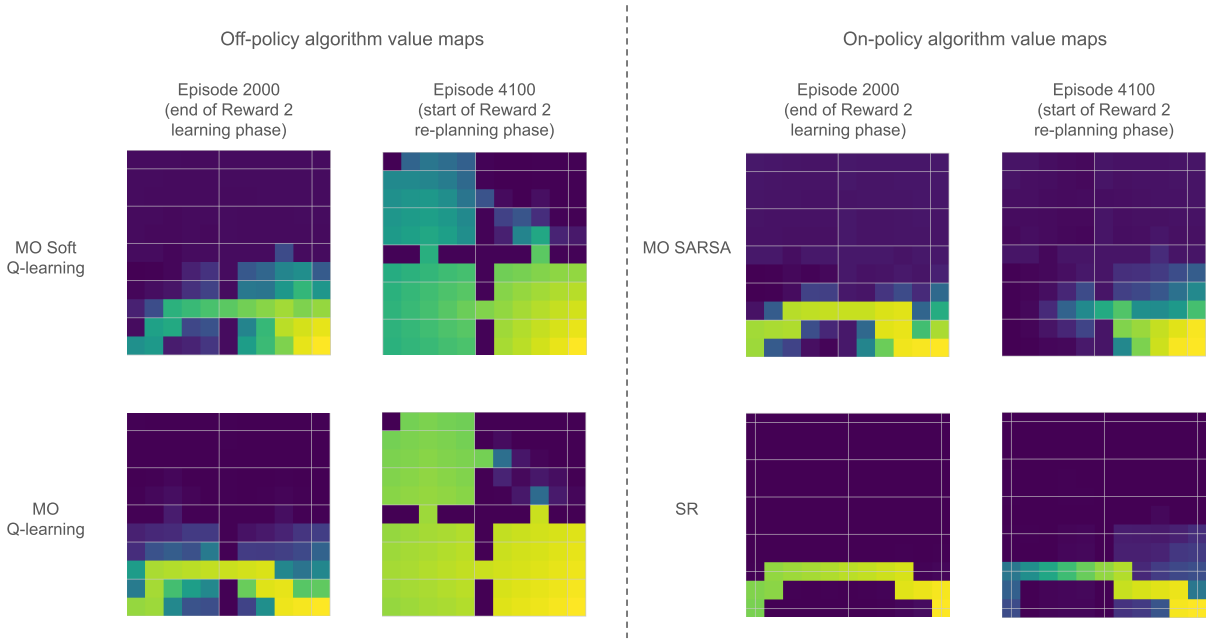

Figure S4: Differences in value propagation between multi-objective (MO) off-policy and on-policy algorithms, while exploring under different policies. Off-policy algorithms propagate optimal values throughout the environment, whereas on-policy algorithms do not. This further explains the efficiency of off-policy MO algorithms in re-planning to a previously experienced reward function.

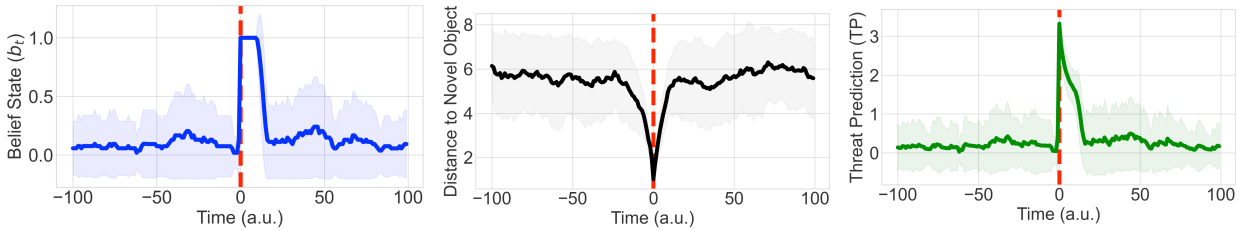

Figure S5: An additional experiment showing that gating the aversive initialisation with a belief state produces the desired temporal asymmetry in TP responses. (A) Belief state that acts like a switch turned on in the vicinity of the novel object and turned off randomly 10-20 steps after avoiding the object (B). Resultant distance to novel object showing the approach-retreat bout, and (C) The Threat-Prediction response on the retreat start, which decays along with the gradient of the value initialisation.

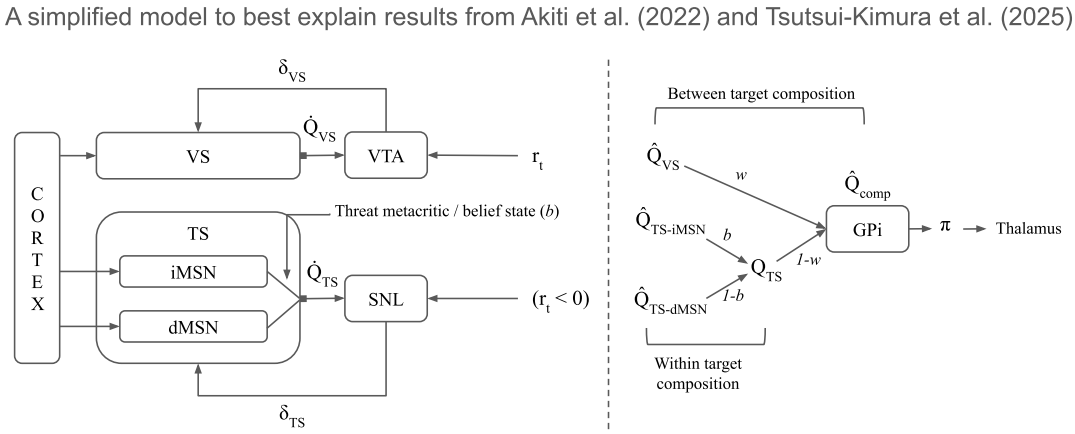

Figure S6: A simplified model to best explain results from Akiti et al. [31] and Tsutsui-Kimura et al. [32].

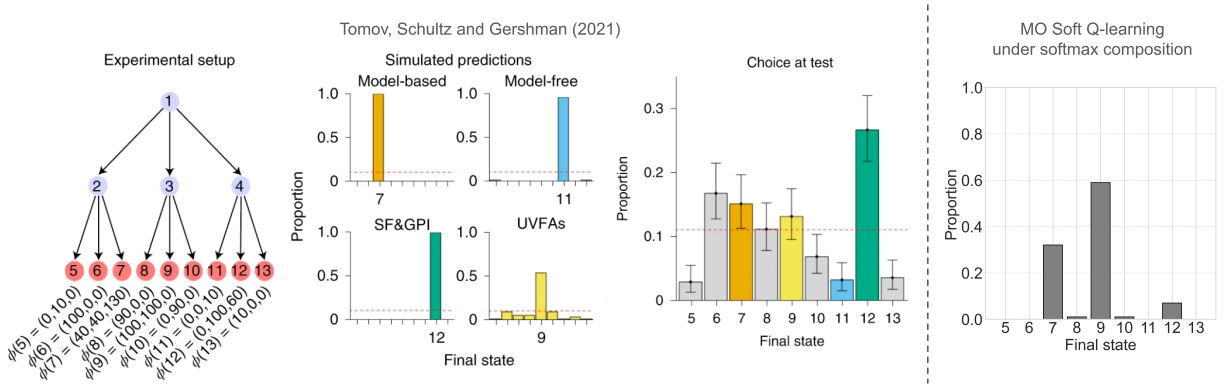

Figure S7: Soft maximum composition may not match human behaviour when they are explicitly asked to optimise the weighted sum of rewards across different modalities [17]. There are two possible explanations. First, like all multi-objective (MO) model-free RL, our model generalises in values, not task structure, whereas human behaviour in these experiments shows generalisation in task structure. Second, the policy under weighted softmax composition can deviate significantly from the policy optimising the weighted summation of rewards. The left side of the figure shows the two-stage MDP by Tomov et al. [17], where the participants were playing a medieval trading game. Different terminal states lead to different quantities of 3 resources (denoted by ϕ), and at the start of each episode, they get to know the price/cost of each resource, setting the reward weights, requiring them to maximise the weighted sum of these multiple rewards. First 100 episodes had different weights: $w_{train} = \{[1, -1, 0], [-1, 1, 0], [1, -2, 0], [-2, 1, 0]\}$ and then tested their responses on 101st episode on a novel weight combination $w_{test} = [1, 1, 1]$. They refined the experiment 3 times, and here we show the final iteration, which gathers support for the SF&GPI strategy, which predicts the final state 12 to be chosen on the 101st episode. However, it is important to note that only 60% of the participants managed to learn their tasks and produce average rewards greater than 0, and the rest were excluded. We find that the soft maximum composition of soft Q-learning results in favouring the final state 9 (same as UVFA, which generalises in values), and partly also 7 (same as MB) for $\tau < 1$ across all of their experiment iterations (here showing for the final one). Since weights cannot be negative in MO Soft Q-learning, the loss was included in the resource quantities (ϕ), composed with the absolute value of the weights for composition. Figures adapted from Tomov et al. [17] with permission from Springer Nature.