Dual-Objective Reinforcement Learning with Novel Hamilton-Jacobi-Bellman Formulations

William Sharpless*[1], Dylan Hirsch*[1], Sander Tonkens[1], Nikhil Shinde[1], and Sylvia Herbert[1]

[1]University of California San Diego, *Equal contribution, wsharpless@ucsd.edu

***Abstract*—In this work, we extend recent advances that connect Hamilton-Jacobi (HJ) equations with RL to propose two novel value functions for dual-objective satisfaction. Namely, we address: (1) the Reach-Always-Avoid problem – of achieving distinct reward and penalty thresholds – and (2) the Reach-Reach problem – of achieving thresholds of two distinct rewards. In contrast with constrained Markov processes or temporal logic approaches, we are able to derive explicit, tractable Bellman forms in this context by decomposing the problems into reach, avoid, and reach-avoid problems to leverage the recent advances. Moreover, we leverage our analysis to propose a variation of Proximal Policy Optimization, dubbed (DO-HJ-PPO), to solve this class of problems and demonstrate that it bests other baselines in safe-arrival and multi-target tasks, providing a new perspective on constrained decision-making.**

I. Related Works

Many in learning and autonomy have considered balancing safety and liveness. A few particularly relevant topics are mentioned here. *Constrained Markov Decision Processes* (CMDPs) are a popular approach to transform constraints into a Lagrangian [1, 2, 3, 4, 5], however, this often requires intricate reward engineering and parameter tuning to balance the combined objective. Similarly, *Multi-Objective* RL solves the Pareto optimal solution of vector-valued rewards but is not focused on priority-scalarized problems [6, 7, 8]. *Goal-Conditioned* RL [9, 10, 11] and *Linear/Signal Temporal Logic* RL [12, 13, 14, 15, 16, 17] generally learn to solve multiple tasks at once, by augmenting the problem to a surrogate problem or automaton, but this is a notoriously challenging approach and there are often no guarantees for the relation between the surrogate and original problem. In this work we are able to derive explicit forms for dual-objective Bellman equations that yield the optimal policy by augmenting the state. This leads to direct approaches for learning the dual-objective values, which is proven to yield improved performance. To do this, we build on traditional dynamic-programming methods that solve Hamilton-Jacobi-Bellman (HJB) equations, and specifically those that connect HJB and RL theories [18, 19, 20, 21].

II. Problem Definitions

Consider a Markov decision process (MDP) $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, f \rangle$ consisting of finite state and action spaces $\mathcal{S}$ and $\mathcal{A}$, and *unknown* dynamics f that define the deterministic transition $s_{t+1} = f(s_t, a_t)$. Let an agent interact with the MDP by selecting an action with policy $\pi : \mathcal{S} \to \mathcal{A}$ to yield a state trajectory s_t^π, i.e. $s_{t+1}^\pi = f\left(s_t^\pi, \pi\left(s_t^\pi\right)\right)$.

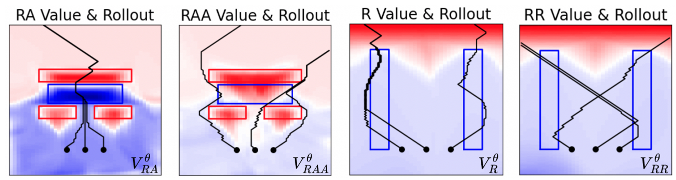

Fig. 1: **DDQN Demonstration of the RAA & RR Problems** We compare our novel formulations with previous HJ-RL formulations (RA & R) in a simple grid-world problem with Double-Deep Q Learning. The hazards are highlighted in red, the goal in blue, and trajectories in black (starting at the dot). In both models, the agents actions are limited to left and right or straight and the system flows upwards over time.

In this work, we consider the **Reach-Always-Avoid** (RAA) and **Reach-Reach** (RR) problems, which both involve balancing two objectives, which are each specified in terms involving a best reward or worst penalty encountered over time. For the RAA problem, let $r, p : \mathcal{S} \to \mathbb{R}$ be a reward to be maximized and a penalty to be minimized. For the RR problem, let $r_1, r_2 : \mathcal{S} \to \mathbb{R}$ be two distinct rewards to be maximized. The agent's overall objective is to maximize the *worst-case* outcome between the best-over-time reward and worst-over-time penalty (RAA) or the two best-over-time rewards (RR):

$$\text{(RAA)} \begin{cases} \text{maximize} & \min\left\{ \max_t r(s_t^\pi),\ \min_t q(s_t^\pi) \right\} \\ \text{s.t.} & s_{t+1}^\pi = f\left(s_t^\pi, \pi\left(s_t^\pi\right)\right), \\ & s_0^\pi = s, \end{cases}$$

$$\text{(RR)} \begin{cases} \text{maximize} & \min\left\{ \max_t r_1(s_t^\pi),\ \max_t r_2(s_t^\pi) \right\} \\ \text{s.t.} & s_{t+1}^\pi = f\left(s_t^\pi, \pi\left(s_t^\pi\right)\right), \\ & s_0^\pi = s, \end{cases}$$

where $q = -p$. Note that we have introduced this negated penalty q in order to establish a "higher-is-better" convention across subtasks as to compose them in an "and" fashion via the $\min$ operation. For conceptual ease, we recommend the reader still think of trying to minimize the largest-over-time penalty p rather than maximize the smallest-over-time q.

As the problem names suggest, these optimization problems are inspired by (but not limited to) tasks involving goal reaching and hazard avoidance. While these problems are thematically distinct, they are mathematically complementary. The respective values corresponding to a policy π are

$$V_{\text{RAA}}^\pi(s) = \min\left\{ \max_t r(s_t^\pi),\ \min_t q(s_t^\pi) \right\}$$
$$V_{\text{RR}}^\pi(s) = \min\left\{ \max_t r_1(s_t^\pi),\ \max_t r_2(s_t^\pi) \right\}.$$

Observe that the minimums and maximums over time make these values fundamentally different from more typical "sum-of-discounted-reward" objectives considered in RL [22]. Moreover, while both returns involve two objectives, these objectives are combined in worst-case fashion to ensure *dual-satisfaction*.

III. Reachability and Avoidability in RL

Prior works [18, 19] study the reach V_{R}^{π}, avoid V_{A}^{π}, and reach-avoid V_{RA}^{π} values, respectively defined by

$$
\begin{aligned}
V_{\text{R}}^{\pi}(s) &= \max_t r(s_t^{\pi}), \\
V_{\text{A}}^{\pi}(s) &= \min_t q(s_t^{\pi}), \\
V_{\text{RA}}^{\pi}(s) &= \max_t \min\left\{ r(s_t^{\pi}), \min_{\tau \leq t} q(s_{\tau}^{\pi}) \right\},
\end{aligned}
$$

for which special Bellman equations [18]. To put these value functions in context, assume the goal $\mathcal{G}$ is the set of states for which $r(s)$ is positive and the hazard $\mathcal{H}$ is the set of states for which $q(s)$ is non-positive. Then V_{R}^{π}, V_{A}^{π}, and V_{RA}^{π} are positive if and only if π causes the agent to eventually reach $\mathcal{G}$, to always avoid $\mathcal{H}$, and to reach $\mathcal{G}$ without hitting $\mathcal{H}$ prior to the reach time, respectively. The Reach-Avoid Bellman Equation (RABE), for example, takes the form [19]

$$
V_{\text{RA}}^{*}(s) = \min\left\{ \max\left\{ \max_{a \in \mathcal{A}} V_{\text{RA}}^{*}\left(f(s,a)\right), r(s) \right\}, q(s) \right\},
$$

and is associated with optimal policy $\pi_{\text{RA}}^{*}(s)$ (without the need for state augmentation, see Section A in the Supplementary Material). This formulation does not naturally induce a contraction, but may be discounted to induce contraction as in [19] by (implicitly) defining $V_{\text{RA}}^{\gamma}(z)$ for each $\gamma \in [0,1)$ via

$$
\begin{aligned}
V_{\text{RA}}^{\gamma}(s) = (1-\gamma)\min\{r(s), q(s)\} \quad + \\
\gamma \min\left\{ \max\left\{ \max_{a \in \mathcal{A}} V_{\text{RA}}^{\gamma}\left(f(s,a)\right), r(s) \right\}, q(s) \right\}.
\end{aligned}
$$

These prior value functions and corresponding Bellman equations have proven powerful for these simple reach/avoid/reach-avoid problem formulations. In this work, we generalize these results to the aforementioned broader class of problems.

IV. The need for augmenting states

Unlike for the reach/avoid/reach-aviod problems, the goal of choosing a policy $\pi : \mathcal{S} \to \mathcal{A}$ is inherently flawed for the RR and RAA problems without state augmentation. In considering multiple objectives over an infinite horizon, situations arise in which the optimal action depends on more than the current state, but rather the *history* the trajectory. An example clarifying the issue is shown in Figure 2.

To allow the agent to use relevant aspects of its history, we will henceforth consider an augmentation of the MDP with auxiliary variables. A theoretical result in the next section states that this choice of augmentation is sufficient in that no additional information (nor even the use of a stochastic policy) will be able to improve performance under the optimal (deterministic) policy.

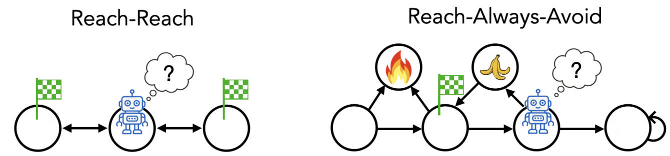

Fig. 2: **Examples where a Non-Augmented Policy is Flawed** In both MDPs, consider an agent with no memory. (Left) For a deterministic policy based on the current state, the agent can only achieve one target (RR), as the policy must associate the middle state with either of the two possible actions. (Right) In the RAA case, assume the robot must avoid the fire at all costs and would prefer to not encounter the peel, but will do so if needed. The optimal decision for the current state depends on state history, specifically on whether the robot has already reached the target state or not.

A. Augmentations

For the RAA problem, we consider an augmentation of the MDP defined by $\overline{\mathcal{M}} = \langle \overline{\mathcal{S}}, \mathcal{A}, \bar{f} \rangle$ with augmented states $\overline{\mathcal{S}} = \mathcal{S} \times \mathcal{Y} \times \mathcal{Z}$, where $\mathcal{Y} := \{r(s) \mid s \in \mathcal{S}\}$ and $\mathcal{Z} := \{q(s) \mid s \in \mathcal{S}\}$. Note the actions $\mathcal{A}$ remain unchanged. For any initial state s, let the augmented states be initialized as $y = r(s)$ and $z = q(s)$, and let the transition $\bar{f}$ of $\overline{\mathcal{M}}$ be given by

$$
\begin{aligned}
s_{t+1}^{\bar{\pi}} &= f\left(s_t^{\bar{\pi}}, \bar{\pi}\left(s_t^{\bar{\pi}}, y_t^{\bar{\pi}}, z_t^{\bar{\pi}}\right)\right), \\
y_{t+1}^{\bar{\pi}} &= \max\left\{ r\left(s_{t+1}^{\bar{\pi}}\right), y_t^{\bar{\pi}} \right\}, \\
z_{t+1}^{\bar{\pi}} &= \min\left\{ q\left(s_{t+1}^{\bar{\pi}}\right), z_t^{\bar{\pi}} \right\},
\end{aligned}
$$

so that y_t and z_t track the best reward and worst penalty yet seen. Hence, a policy $\bar{\pi} : \overline{\mathcal{S}} \to \mathcal{A}$ for $\overline{\mathcal{M}}$ may leverage information regarding the history of the trajectory.

For the RR problem, we augment the system similarly, except $\mathcal{Y} := \{r_1(s) \mid s \in \mathcal{S}\}$, $\mathcal{Z} := \{r_2(s) \mid s \in \mathcal{S}\}$, and

$$
\begin{aligned}
s_{t+1}^{\bar{\pi}} &= f\left(s_t^{\bar{\pi}}, \bar{\pi}\left(s_t^{\bar{\pi}}, y_t^{\bar{\pi}}, z_t^{\bar{\pi}}\right)\right), \\
y_{t+1}^{\bar{\pi}} &= \max\left\{ r_1\left(s_{t+1}^{\bar{\pi}}\right), y_t^{\bar{\pi}} \right\}, \\
z_{t+1}^{\bar{\pi}} &= \max\left\{ r_2\left(s_{t+1}^{\bar{\pi}}\right), z_t^{\bar{\pi}} \right\}.
\end{aligned}
$$

V. Optimal Policies for RAA and RR by Value Decomposition

We now discuss our first theoretical contributions.

A. Decomposition of RAA into avoid and reach-avoid problems

Our main theoretical result for the RAA problem shows that we can solve this problem by first solving the avoid problem corresponding to the negated penalty $q(s)$ to obtain the optimal value function $V_{\text{A}}^{*}(s)$ and then solving a reach-avoid problem with the negated penalty $q(s)$ and a modified reward $r_{\text{RAA}}(s)$.

Theorem 1. *For all initial states $s \in \mathcal{S}$,*

$$
\max_{\bar{\pi}} V_{\text{RAA}}^{\bar{\pi}}(s) = \max_{\pi} \max_t \min\left\{ r_{\text{RAA}}\left(s_t^{\pi}\right), \min_{\tau \leq t} q\left(s_{\tau}^{\pi}\right) \right\}, \tag{1}
$$

where $r_{\text{RAA}}(s) := \min\{r(s), V_{\text{A}}^{*}(s)\}$, with

$$
V_{\text{A}}^{*}(s) := \max_{\pi} \min_t q\left(s_t^{\pi}\right).
$$

Proof. See section A of the Supplementary Material. ∎

Corollary 1. *The value function* $V^*_{\text{RAA}}(s) := \max_{\bar{\pi}} V^{\bar{\pi}}_{\text{RAA}}(s)$ *satisfies the Bellman equation*

$$V^*_{\text{RAA}}(s) = \min\left\{\max\left\{\max_{a\in\mathcal{A}} V^*_{\text{RAA}}(f(s,a)), r_{\text{RAA}}(s)\right\}, q(s)\right\}.$$

B. Decomposition of the RR problem into three reach problems

Our main result for the RR problem shows that we can solve this problem by first solving two reach problems corresponding to the rewards $r_1(s)$ and $r_2(s)$ to obtain reach value functions $V^*_{\text{R1}}(s)$ and $V^*_{\text{R2}}(s)$, respectively. We then solve a third reach problem with a modified reward $r_{\text{RR}}(s)$.

Theorem 2. *For all initial states* $s \in \mathcal{S}$,

$$\max_{\bar{\pi}} V^{\bar{\pi}}_{\text{RAA}}(s) = \max_{\pi}\max_{t} r_{\text{RR}}(s^{\pi}_t), \tag{2}$$

where

$$r_{\text{RR}}(s) := \max\left\{\min\left\{r_1(s), V^*_{\text{R2}}(s)\right\}, \min\left\{r_2(s), V^*_{\text{R1}}(s)\right\}\right\},$$

with

$$V^*_{\text{R1}}(s) := \max_{\pi}\max_{t} r_1(s^{\pi}_t) \text{ and } V^*_{\text{R2}}(s) := \max_{\pi}\max_{t} r_2(s^{\pi}_t).$$

Proof. See section B of the Supplementary Material. □

Corollary 2. *The value function* $V^*_{\text{RR}}(s) := \max_{\bar{\pi}} V^{\bar{\pi}}_{\text{RR}}(s)$ *satisfies the Bellman equation*

$$V^*_{\text{RR}}(s) = \max\left\{\max_{a\in\mathcal{A}} V^*_{\text{RR}}(f(s,a)), r_{\text{RR}}(s)\right\}.$$

C. Optimality of the augmented problems

The following theoretical result demonstrates that the augmentation is optimal for the original problem in that no further historical information can be used to improve performance.

Theorem 3. *Let* $s \in \mathcal{S}$. *Then*

$$\max_{\bar{\pi}} V^{\bar{\pi}}_{\text{RAA}}(s) = \max_{a_0,a_1,\ldots} \min\left\{\max_{t} r(s_t), \min_{t} q(s_t)\right\},$$

and

$$\max_{\bar{\pi}} V^{\bar{\pi}}_{\text{RR}}(s) = \max_{a_0,a_1,\ldots} \min\left\{\max_{t} r_1(s_t), \max_{t} r_2(s_t)\right\}$$

where $s_{t+1} = f(s_t, a_t)$ *and* $s_0 = s$.

VI. DO-HJ-PPO: Solving RAA and RR with RL

In the previous sections, we demonstrated that the RAA and RR problems can be solved through decomposition of the values into reach, avoid, and reach-avoid problems. Because each of these problems has a corresponding Bellman relation (see [18, 19, 21]), these decompositions allow us to solve the dual-objective problems by minimally modifying existing RL methods. We make a few assumptions in the derivation that would limit performance and generalization, namely, the determinism of the values as well as access to the decomposed values (by solving them beforehand). In this section, we propose relaxations to the RR and RAA theory and devise a custom variant of Proximal Policy Optimization, **DO-HJ-PPO**, to solve this broader class of problems, and demonstrate its performance.

A. Stochastic Reach-Avoid Bellman Equation

It is well known that the most performative RL methods allow for stochastic learning. In [21], the Stochastic Reachability Bellman Equation (SRBE) is described for Reach problems and used to design a specialized PPO algorithm. In this section we proceed by closely following this work, modifying the SRBE into a Stochastic Reach-Avoid Bellman Equation (SRABE). Using Theorems 1 and 2, the SRBE and SRABE offer the necessary tools for designing a PPO variant for solving the RR and RAA problems.

We define $\tilde{V}^{\pi}_{\text{RAA}}$ to be the solution to the following Bellman equation (SRABE):

$$\tilde{V}^{\pi}_{\text{RAA}}(s) = \mathbb{E}_{a\sim\pi}\left[\min\left\{\max\left\{\tilde{V}^{\pi}_{\text{RAA}}(f(s,a)), r_{\text{RAA}}(s)\right\}, q(s)\right\}\right]$$

The corresponding action-value function is

$$\tilde{Q}^{\pi}_{\text{RAA}}(s,a) = \min\left\{\max\left\{\tilde{V}^{\pi}_{\text{RAA}}(f(s,a)), r_{\text{RAA}}(s)\right\}, q(s)\right\}.$$

We define a modification of the dynamics f involving an absorbing state s_∞ as follows:

$$f'(s,a) = \begin{cases} f(s,a) & q(f(s,a)) < \tilde{V}^{\pi}_{\text{RAA}}(s) < r_{\text{RAA}}(f(s,a)), \\ s_\infty & \text{otherwise.} \end{cases}$$

We then have the following proposition:

Proposition 1. *For each* $s \in \mathcal{S}$ *and every* $\theta \in \mathbb{R}^{n_p}$, *we have*

$$\nabla_\theta \tilde{V}^{\pi_\theta}_{\text{RAA}}(s) \propto \mathbb{E}_{s'\sim d'_{\pi}(s), a\sim\pi_\theta}\left[\tilde{Q}^{\pi_\theta}_{\text{RAA}}(s',a)\nabla_\theta \ln \pi_\theta(a|s')\right],$$

where $d'_\pi(s)$ *is the stationary distribution of the Markov Chain with transition function*

$$P(s'|s) = \sum_{a\in\mathcal{A}} \pi(a|s)\left[f'(s,\pi(a|s)) = s'\right],$$

with the bracketed term equal to 1 *if the proposition inside is true and* 0 *otherwise.*

Following [19], we then define the discounted value and action-value functions with $\gamma \in [0,1)$:

$$\begin{aligned}\tilde{V}^{\gamma,\pi}_{\text{RAA}}(s) &= (1-\gamma)\min\left\{r_{\text{RAA}}(s), q(s)\right\} \\ &\quad + \gamma\mathbb{E}_{a\sim\pi}\left[\min\left\{\max\left\{\tilde{V}^{\gamma,\pi}_{\text{RAA}}(f(s,a)), r_{\text{RAA}}(s)\right\}, q(s)\right\}\right], \\ \tilde{Q}^{\gamma,\pi}_{\text{RAA}}(s,a) &= (1-\gamma)\min\left\{r_{\text{RAA}}(s), q(s)\right\} \\ &\quad + \gamma\min\left\{\max\left\{\tilde{V}^{\gamma,\pi}_{\text{RAA}}(f(s,a)), r_{\text{RAA}}(s)\right\}, q(s)\right\}.\end{aligned}$$

The PPO advantage function is then given by $\tilde{A}^{\pi}_{\text{RAA}} = \tilde{Q}_{\text{RAA}} - \tilde{V}_{\text{RAA}}$ [23].

B. Algorithm

We introduce DO-HJ-PPO in Algorithm 1, a unified PPO-based algorithm for solving the Reach-Always-Avoid (RAA) and Reach-Reach (RR) problems, which builds on the SRABE and SRBE formulations with minimal modifications to the standard PPO framework. Note that in the case of a reset while

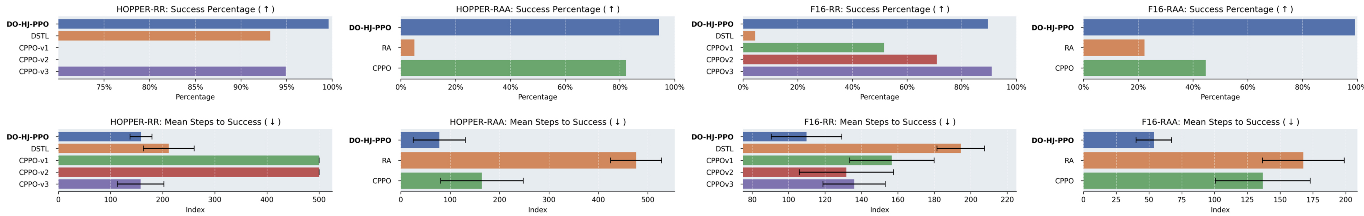

Fig. 3: **Algorithm Comparisons for RR and RAA tasks**. We evaluate our method and relevant baselines on 1,000 trajectories for both the Reach-Reach (RR) and Reach-Always-Avoid (RAA) problems for Hopper and F16 environments. In the RR tasks, we compare against a decomposed version of the problem (DSTL) and several variants of CPPO. Our method consistently reaches both target regions with a higher success rate and fewer steps on average. Notably, CPPOv1 and CPPOv2 fail to achieve any successful trajectories in the RR task, whereas CPPOv3 shows improved—but still limited—performance. For the RAA tasks, we compare our approach against Constrained PPO (CPPO) and standard reach-avoid baselines. Our method achieves a higher success rate while requiring a lower average number of steps to reach success.

Algorithm 1 : DO-HJ-PPO (Actor-Critic)

Define *Composed* Actor and Critic $\tilde{Q}$
Define *Decomposed* Actor(s) and Critic(s) $\tilde{Q}_i$
for $k = 0, 1, \ldots$ **do**
  for $t = 0$ to $T-1$ **do**
    Sample trajectories for $\tau_t : \{\hat{s}_t, a_t, \hat{s}_{t+1}\}$
    Define $\tilde{r}(s_t)$ with Decomposed Critics $\tilde{Q}_i(s_t)$ (Theorems 1 & 2)
    Composed Critic update:

$$\omega \leftarrow \omega - \beta_k \nabla_\omega \tilde{Q}(\tau_t) \cdot \left(\tilde{Q}(\tau_t) - B^\gamma[\tilde{Q};\tilde{r}](\tau_t)\right)$$

    Compute Bellman-GAE A^λ_{HJ} with B^γ
    (Standard) update Composed Actor
    Decomposed Critic update(s):

$$\omega \leftarrow \omega - \beta_k \nabla_\omega \tilde{Q}_i(\tau_t) \cdot \left(\tilde{Q}_i(\tau_t) - B^\gamma_i[\tilde{Q}_i](\tau_t)\right)$$

    Compute Bellman-GAE A^λ_i with B^γ_i
    (Standard) update Decomposed Actor(s)
  end for
end for
return parameter θ, ω

learning, augmented state variables are set to the initial values described in section IV.

In Algorithm 1, B^γ and B^γ_i represent the composed and decomposed Bellman updates for the users choice of problem (RAA or RR). The Bellman update $B^\gamma[\tilde{Q};\tilde{r}]$ differs for the RAA task and RR task, depending on the corresponding the reach, avoid, and reach-avoid decomposition. Beyond the modified Bellman updates, the learning proceeds in the nominal PPO fashion, in which the actively-learned decomposed values are used to define the composed value (see Supplementary).

VII. Experiments

We first demonstrate the theoretical results (Theorems 1 and 2) through a simple 2D grid-world experiment using Double Deep Q-Networks (DDQN) (Figure 1). Additional experimental details are provided in the Supplementary Material. On the left, we compare the optimal value functions learned under the classic RA formulation with those from the RAA setting. In the RA scenario, trajectories successfully avoid the obstacle but may terminate in regions from which future collisions are inevitable. On the right, we consider a similar environment but with two reward targets. Here, the RR formulation induces trajectories that visit both targets, unlike simple reach tasks in which the agent halts. These qualitative results highlight the behavioral distinctions induced by the RAA and RR objectives compared to their simpler counterparts.

To evaluate the method under more complex and less structured conditions, we extend our analysis to continuous control settings using our algorithm **DO-HJ-PPO**. Specifically, we apply DO-HJ-PPO to RAA and RR tasks in the Hopper and F16 environments. For the Hopper, two high targets and floor and wall obstacles are defined with respect to its head, and in the F16, the targets are defined by regions to fly through, while obstacles are defined by geofences which create a boxed flight corridor. We compare against both STL (DSTL) and contrained PPO (CPPO) baselines (see supplementary material). Empirically, DO-HJ-PPO performs equivalently at worst and more often at a significantly higher ability, scored in metrics of task success percentage and steps to achieve the task, indicating that DO-HJ-PPO more reliably and rapidly solves the given tasks. These results underscore the challenging nature of composing multiple objectives using traditional baselines while in contrast, our method provides a more robust and direct solution for handling such complex compositional tasks, with less required tuning.

VIII. Conclusion

In this brief, we introduced two novel Bellman formulations for new problems (RAA and RR) which generalize those in recent publications. We prove decomposition results for these problems that allow us to break them into simpler Bellman problems, which can then be composed to obtain the value functions and corresponding optimal policies. We use these results to design a PPO-based algorithm for practical solution of RAA and RR. More broadly, this work provides a road-map to extend the range of Bellman formulations that can be solved, via decomposing higher-level problems into lower-level ones. By solving the RAA and RR, we add two new ingredients to the list of solvable problems that can be leveraged toward this end.

REFERENCES

[1] Eitan Altman. *Constrained Markov decision processes: Stochastic modeling*. Routledge, Boca Raton, 13 December 2021.

[2] Joshua Achiam, David Held, Aviv Tamar, and P Abbeel. Constrained policy optimization. *ICML*, abs/1705.10528:22–31, 30 May 2017.

[3] Akifumi Wachi and Yanan Sui. Safe reinforcement learning in constrained Markov decision processes. *ICML*, 119:9797–9806, 12 July 2020.

[4] Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, and Alois Knoll. A review of safe reinforcement learning: Methods, theories, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(12):11216–11235, December 2024.

[5] Yinlam Chow, Aviv Tamar, Shie Mannor, and M Pavone. Risk-sensitive and robust decision-making: A CVaR optimization approach. *Neural Inf Process Syst*, abs/1506.02188, 6 June 2015.

[6] Marco A Wiering, Maikel Withagen, and Madalina M Drugan. Model-based multi-objective reinforcement learning. In *2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pages 1–6. IEEE, December 2014.

[7] Moffaert K Van and A Nowé. Multi-objective reinforcement learning using sets of Pareto dominating policies. *The Journal of Machine Learning Research*, 15(1):3483–3512, 2014.

[8] Xin-Qiang Cai, Pushi Zhang, Li Zhao, Jiang Bian, Masashi Sugiyama, and Ashley Llorens. Distributional Pareto-optimal multi-objective reinforcement learning. *Neural Inf Process Syst*, 36:15593–15613, 2023.

[9] Minghuan Liu, Menghui Zhu, and Weinan Zhang. Goal-conditioned reinforcement learning: Problems and solutions. *arXiv [cs.AI]*, 20 January 2022.

[10] Matthias Plappert, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, Jonas Schneider, Josh Tobin, Maciek Chociej, Peter Welinder, Vikash Kumar, and Wojciech Zaremba. Multi-goal reinforcement learning: Challenging robotics environments and request for research. *arXiv [cs.LG]*, 26 February 2018.

[11] Zhizhou Ren, Kefan Dong, Yuanshuo Zhou, Qiang Liu, and Jian Peng. Exploration via hindsight goal generation. *Neural Inf Process Syst*, 32:13464–13474, 1 June 2019.

[12] F Bacchus, Craig Boutilier, and Adam J Grove. Rewarding behaviors. In *Proceedings of the National Conference on Artificial Intelligence.*, pages 1160–1167. cs.toronto.edu, 4 August 1996.

[13] Fahiem Bacchus, Craig Boutilier, and Adam Grove. Structured solution methods for non-Markovian decision processes. In *AAAI/IAAI*, pages 112–117, 1997.

[14] S Thiebaux, C Gretton, J Slaney, D Price, and F Kabanza. Decision-theoretic planning with non-Markovian rewards. *J. Artif. Intell. Res.*, 25:17–74, 29 January 2006.

[15] Alberto Camacho, Oscar Chen, Scott Sanner, and Sheila McIlraith. Non-Markovian rewards expressed in LTL: Guiding search via reward shaping. *Proceedings of the International Symposium on Combinatorial Search*, 8(1):159–160, 1 September 2021.

[16] Rodrigo Toro Icarte, Toryn Q Klassen, R Valenzano, and Sheila A McIlraith. Using reward machines for high-level task specification and decomposition in reinforcement learning. *ICML*, 80:2112–2121, 3 July 2018.

[17] Alberto Camacho, Rodrigo Toro Icarte, Toryn Q Klassen, Richard Valenzano, and Sheila A McIlraith. LTL and beyond: Formal languages for reward function specification in reinforcement learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 6065–6073, California, 1 August 2019. International Joint Conferences on Artificial Intelligence Organization.

[18] Jaime F Fisac, Neil F Lugovoy, Vicenç Rubies-Royo, Shromona Ghosh, and Claire J Tomlin. Bridging hamilton-jacobi safety analysis and reinforcement learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8550–8556. IEEE, 2019.

[19] Kai-Chieh Hsu, Vicenç Rubies-Royo, Claire J. Tomlin, and Jaime F. Fisac. Safety and liveness guarantees through reach-avoid reinforcement learning. In *Proceedings of Robotics: Science and Systems*, Held Virtually, July 2021.

[20] Milan Ganai, Chiaki Hirayama, Ya-Chien Chang, and Sicun Gao. Learning stabilization control from observations by learning lyapunov-like proxy models. *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[21] Oswin So, Cheng Ge, and Chuchu Fan. Solving minimum-cost reach avoid using reinforcement learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[22] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018.

[23] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017.