Multiple Plans are Better than One: Diverse Stochastic Planning

Mahsa Ghasemi, [1*] Evan Scope Crafts, [2*] Bo Zhao, [2, 3] Ufuk Topcu [2, 4]
[1]Department of Electrical and Computer Engineering
[2]Oden Institute for Computational Engineering and Sciences
[3]Department of Biomedical Engineering
[4]Department of Aerospace Engineering and Engineering Mechanics
The University of Texas at Austin, Austin, Texas 78712, USA
{mahsa.ghasemi, escopec, bozhao, utopcu}@utexas.edu

Abstract

In planning problems, it is often challenging to fully model the desired specifications. In particular, in human-robot interaction, such difficulty may arise due to human's preferences that are either private or complex to model. Consequently, the resulting objective function can only partially capture the specifications and optimizing that may lead to poor performance with respect to the true specifications. Motivated by this challenge, we formulate a problem, called *diverse stochastic planning*, that aims to generate a set of representative — small and diverse — behaviors that are near-optimal with respect to the known objective. In particular, the problem aims to compute a set of diverse and near-optimal policies for systems modeled by a Markov decision process. We cast the problem as a constrained nonlinear optimization for which we propose a solution relying on the Frank-Wolfe method. We then prove that the proposed solution converges to a stationary point and demonstrate its efficacy in several planning problems.

Introduction

Solution diversity has value in numerous planning applications, including collaborative systems, reinforcement learning, and preference-based planning. In human groups and, more generally, animal groups, the so-called notion of behavioral diversity leads to the group members' heterogeneous behavior (Balch et al. 1997). This heterogeneity ensures that the members learn complementary skills, thus improving the group's overall performance. An agent learning a task in an unknown environment may benefit from inducing diversity in its decisions to explore the environment more efficiently. In planning with unknown preferences, one can use diversity to construct a set of behaviors that are suitable for different preferences.

Algorithms that use notions of diversity to address one or more of these applications are known as *quality diversity (QD)* algorithms (Pugh, Soros, and Stanley 2016). A key component of QD algorithms is a way to summarize the important properties of different solutions. This description, known as a *behavior characterization*, is used to define diversity-based metrics (Doncieux and Mouret 2013). Without proper behavior characterization, solutions with trivial differences can have high values of diversity as measured by the resulting metric.

Our work is motivated by planning in settings where, in addition to a known objective, there exist some unknown objectives. The unknown objectives may represent a human user or designer's preference, which is either private or complex to model. In these settings, we propose a QD-based approach to construct a "representative" — small and diverse — set of near-optimal policies with respect to the known objective and then present that to the human to select from according to their unknown objectives. This approach allows the human to have the ultimate control over the behavior, without requiring prior knowledge of the human's preferences.

Formally, we consider the multi-objective optimization problem of returning a set of feasible policies for an infinite horizon Markov decision process (MDP) that is both near-optimal and diverse. We define the optimality of a set of policies as the sum of each policy's expected average reward in the set. Diversity captures the representativeness of a set of policies. We characterize the behavior of policies using their state-action occupancy measures and quantify diversity by the sum of pairwise divergences between the state-action occupancy measures of the policies in the set.

A key element of our approach is the behavior characterization of policies using their state-action occupancy measures. This characterization is domain-independent and fully encapsulates the dynamics of a given policy. We use this characterization to define the diversity of a set of policies through the pairwise Jensen-Shannon divergences between the occupancy measures. We then define the objective as a linear combination of the sum of the policies' rewards and their diversity. By utilizing the dual of the average cost linear program, we recast our formulation as a constrained optimization problem. We then show that, due to the constraints' linearity, the problem can be solved efficiently using the Frank-Wolfe algorithm. We also prove that the algorithm is guaranteed to converge to a stationary point. Furthermore, in a series of simulations, we evaluate the proposed algorithm's performance and show its efficacy.

[*]These authors contributed equally to the manuscript.

Related Work

Research on the development of QD algorithms has occurred within two different communities. In the field of optimization, perspectives on evolution as a process that finds distinct niches for different species have motivated the use of diversity. Simultaneously, there has been significant interest in the use of diversity to provide high-quality solutions for unknown objectives within the planning community.

In the optimization community, recent interest in QD algorithms has been driven by the success of the Novelty Search algorithm (Lehman and Stanley 2008). The original Novelty Search algorithm eschews the use of notions of solution quality entirely; its sole goal is to find a set of solutions that are diverse with respect to some distance measure. Surprisingly, this approach is able to find solutions with better performance on difficult tasks, such as maze navigation, than algorithms relying on an objective function. This result has led to considerable interest in the development of new QD algorithms to address tasks that were previously considered to be too difficult. For an overview, see the work of Pugh, Soros, and Stanley (2016).

The type of behavior characterization used in these works varies and can be domain-dependent. For example, in navigation problems, diversity can be defined using Euclidean distances between points visited. Another approach, used by the popular MAP elites algorithm, is to assume that a domain-dependent behavior characterization is given (Mouret and Clune 2015). A promising area of research is the development of new approaches to behavior characterization (Gaier, Asteroth, and Mouret 2020).

The success of the Novelty Search and MAP elites algorithms has inspired the use of diversity in reinforcement learning, with the hope that diversity can help avoid poor local minima. Different methods of behavior characterization for policies have been used, including methods based on sequences of actions (Jackson and Daley 2019), state trajectories (Eysenbach et al. 2018), or diversity through determinants of actions in states (Parker-Holder et al. 2020). Similarly to our work, Parker-Holder et al. consider an explicit tradeoff between the quality and diversity of the policies. However, our approach differs in that we leverage knowledge of the system dynamics to characterize policies in a way that includes information about both the states visited and the policy actions, and to develop a solution algorithm with guaranteed convergence to a local minimum.

Behavior characterization has also been a key focus of QD-based work in the planning community. For example, in an approach similar to MAP elites, Myers and Lee (1999) and Myers (2006) assume that there is a meta-description of the planning domain. They then define an approach that obtains solutions that are diverse with respect to the meta-description. Another approach to behavior characterization is through the use of domain landmarks, which are disjunctive sets of propositions that plans must satisfy, such as a set of states that a trajectory must reach before the goal state (Hoffmann and Nebel 2001). If the set of landmarks can be computed, a greedy algorithm can be used to iteratively select landmarks from the set and find a plan that satisfies the landmark (e.g., reaches a certain state) (Bryce 2014). Behavior characterization based on the plan actions, as in the RL community, is also a common technique (Coman and Munoz-Avila 2011; Nguyen et al. 2012; Katz and Sohrabi 2020).

The way behavioral characterization and diversity metrics are incorporated into planning algorithms varies. In some cases, the problem is formulated as maximizing the diversity of the set of solutions (Coman and Munoz-Avila 2011; Hebrard et al. 2005), or as finding a set of solutions that satisfy a diversity threshold (Nguyen et al. 2012; Srivastava et al. 2007; Hebrard et al. 2005). In other cases, like our work, there exists both an unknown objective and a known objective, and the problem is formulated in terms of a tradeoff between the diversity of the solution set and the optimality of each of the candidate solutions (Coman and Munoz-Avila 2011; Katz and Sohrabi 2020; Petit and Trapp 2015). Our work is distinct from these approaches because we develop a new method for behavior characterization and consider a stochastic setting modeled as an MDP. In addition, unlike many QD-based planning algorithms, our approach does not rely on greedy strategies. While greedy algorithms have near-optimality guarantees in some settings, such as when the problem is submodular (Bach 2013), in general no such guarantee exists.

Problem Formulation

We now overview the required background related to Markov decision processes, occupancy measures, and divergence metrics. We then present the main problem as a nonlinear optimization problem over the space of occupancy measures.

Preliminaries

We consider systems whose behavior is modeled by a Markov decision process (MDP). An MDP is a tuple $M = (S, A, P, R)$, where S is a finite set of states, A is a finite set of actions, $P : S \times A \times S \to [0, 1]$ is a probabilistic transition function such that for all $s \in S$ and $a \in A$, $\sum_{s' \in S} P(s'|s, a) = 1$, and $R : S \times A \times S \to \mathbb{R}$ is a reward function.

A stationary stochastic policy π on an MDP is a mapping from the state space to a probability distribution over the actions, formally defined as $\pi : S \times A \to [0, 1]$. Here we consider only stationary stochastic policies and denote the set of all such policies as Π_{ss}.

We focus on the class of problems defined over MDPs that aim to maximize the long-run average reward. The long-run average reward of a policy π is

$$\mathbb{E}_{\tau \sim \pi} \left[\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} r(s_t, a_t) \right], \tag{1}$$

where the expectation is over all possible trajectory realizations from policy π, and s_t and a_t are time-indexed states and actions according to a trajectory τ. We assume that the MDP satisfies the weak accessibility (WA) condition (Bertsekas 1995).

The occupancy measure of a policy, $\rho^\pi(.,.)$, is defined as the distribution induced by the execution of that policy over the state-action pairs, asymptotically, i.e.,

$$\rho^{\pi}(s,a) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \Pr(s_t = s, a_t = a|\pi). \tag{2}$$

The long run behavior of a stationary stochastic policy can be represented using its corresponding occupancy measure. An optimal stationary stochastic policy is a policy that maximizes the long-run average reward. It has been shown that under the WA condition, an optimal policy π^* can be obtained by solving the Bellman equation, which can be reformulated as the dual form of a linear program (see Section 4.5 in Volume II of (Bertsekas 1995)),

$$\begin{aligned}
&\max_{\rho} \langle \rho, r\rangle \\
&\textit{subject to} \\
&\quad \sum_{a\in A} \rho(s,a) = \sum_{s'\in S}\sum_{a'\in A} P(s|s',a')\rho(s',a') \\
&\qquad\qquad \textit{for all} \quad s \in S, \\
&\quad \sum_{s\in S}\sum_{a\in A} \rho(s,a) = 1, \\
&\quad \rho(s,a) \geq 0 \qquad\qquad \textit{for all} \quad s\in S, a\in A
\end{aligned} \tag{3}$$

over occupancy measures ρ, where $\langle \rho, r\rangle$ denotes the inner product in the space of $S \times A$, i.e.,

$$\langle \rho, r\rangle = \sum_{s\in S}\sum_{a\in A} \rho(s,a) r(s,a).$$

In particular, an optimal policy π^* corresponding to the solution ρ^* from solving the above linear program can be computed by defining

$$\pi^*(s,a) = \frac{\rho^*(s,a)}{\sum_{a'\in A} \rho^*(s,a')}\ , \tag{4}$$

for all non-transient states, i.e., all states $s \in S$ such that $\sum_{a'\in A} \rho^*(s,a') \neq 0$. We note that the optimal policy corresponding to ρ^* is not uniquely defined as the choice of action in transient states does not affect the long-run behavior.

In our problem context, we seek to find a set of policies such that each policy in the set is near-optimal, and the set is representative of the diverse range of near-optimal behaviors. Specifically, we aim to find a small set of policies with cardinality $k \in \mathbb{N}$, and we use $\Pi_k \in \Pi_{ss}^k$ to denote a set of k stationary stochastic policies.

The state-action occupancy measures provide a natural and domain-independent way to characterize the behavior of policies and ensure diversity. In particular, by using a pairwise metric of the distance between occupancy measures, we can define a diversity metric for a set of policies. Given that the occupancy measures are probability distributions, a natural choice is the Jensen-Shannon divergence (Briët and Harremoës 2009). The Jensen–Shannon divergence between two probability distributions p and q is expressed as

$$\text{JSD}(p\|q) = \frac{1}{2}\text{KL}(p\|m) + \frac{1}{2}\text{KL}(q\|m), \tag{5}$$

where $m = 1/2(p+q)$ is the average distribution, and $\text{KL}(.,.)$ denotes the Kullback–Leibler divergence. The Kullback–Leibler divergence for two probability distributions p and m, over the same discrete probability space X, is defined as

$$\text{KL}(p\|m) = -\sum_{x\in X} p(x) \log\left(\frac{m(x)}{p(x)}\right). \tag{6}$$

We choose the Jensen-Shannon divergence over other probability distribution-based measures because it is symmetric and bounded between zero and one.

Problem Statement

We aim to design an algorithm that can provide a representative set of polices over an MDP that are near-optimal with respect to a known reward function. In particular, given the stated definitions, the objective is to construct k policies that cumulatively, have high reward and diversity. We define the cumulative reward of a set of policies Π_k as the sum of their individual accumulated rewards, i.e.,

$$R(\Pi_k) = \sum_{\pi\in\Pi_k} \mathbb{E}_{\tau\sim\pi}\left[\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T} r(s_t,a_t)\right], \tag{7}$$

and their cumulative diversity as the sum of the pairwise Jensen–Shannon divergences between their occupancy measures, i.e.,

$$D(\Pi_k) = \sum_{\substack{\pi_i,\pi_j\in\Pi_k \\ i<j}} \text{JSD}(\rho^{\pi_i}\|\rho^{\pi_j}). \tag{8}$$

Therefore, given an MDP M and a parameter k, the goal is to find a set of policies $\Pi_k \in \Pi_{ss}^k$ with high cumulative reward $R(\Pi_k)$ and high diversity $D(\Pi_k)$.

Remark 1. *It should be noted that one can choose a metric different than the linear combination used here to quantify the cumulative reward. The linear combination has been widely used in the related literature, partly due to its simplicity. Another important factor that makes this linear combination desirable in this work is the fact that it achieves a faster convergence rate. Later, we prove that the Lipschitz constant of the gradient of the objective function inversely affects the convergence rate. The proposed linear combination does not contribute to the Lipschitz constant, therefore enabling a faster convergence (at least theoretically). Nonetheless, our proposed approach is amenable to variants of the quality metric.*

Proposed Solution

Our problem statement defines a multi-objective optimization problem that aims to maximize a reward-based objective and a diversity-based objective. A standard method for tackling multi-objective problems is to linearly combine the objectives using judiciously chosen weights. To that end, we first note that the objectives should be independent of the cardinality of the solution set, i.e., the number of policies should not affect the quality of the solution. We address this point by normalizing the reward term by the number of policies, k, and the diversity term by the number of unique policy pairs, $k(k-1)/2$. Then, we can define the compound objective function as a linear combination of the normalized reward and diversity. The problem of finding Π_k^* can thus be cast as finding a solution to the following optimization problem:

$$\Pi_k^* = \operatorname*{arg\,max}_{\Pi_k \in \Pi_{ss}^k} \frac{1}{k} R(\Pi_k) + \frac{2\lambda}{k(k-1)} D(\Pi_k), \tag{9}$$

where λ is the tradeoff parameter that controls the relative weightings of the reward and diversity. Using the dual of the linear program for finding an optimal policy, we reformulate the above problem as

$$\begin{aligned}
&\max_{\rho_{1:k}} f(\rho_{1:k}) \\
&\textit{subject to} \\
&\quad \sum_{a \in A} \rho_i(s,a) = \sum_{s' \in S} \sum_{a' \in A} P(s|s',a') \rho_i(s',a') \\
&\qquad\qquad \textit{for all} \quad i \in [k], s \in S, \\
&\quad \sum_{s \in S} \sum_{a \in A} \rho_i(s,a) = 1 \qquad\qquad \textit{for all} \quad i \in [k], \\
&\quad \rho_i(s,a) \geq 0 \qquad\qquad \textit{for all} \quad i \in [k], s \in S, a \in A,
\end{aligned} \tag{10}$$

where

$$f(\rho_{1:k}) = \frac{1}{k} \sum_{i \in [k]} \langle \rho_i, r \rangle + \frac{2\lambda}{k(k-1)} \sum_{\substack{i,j \in [k] \\ i<j}} \mathrm{JSD}(\rho_i \| \rho_j),$$

$\rho_{1:k} = \{\rho_1, \rho_2, \ldots, \rho_k\}$ denotes the k occupancy measures corresponding to the k policies, and $[k] = \{1, 2, \ldots, k\}$.

The reformulated version is a constrained optimization problem with linear constraints and a nonlinear (and non-concave) objective function. In general, this problem does not have a unique global solution. For instance, any permutation of the k policies in an optimal solution will result in another optimal solution. Nonetheless, one can seek solution approaches that at least converge to stationary points.

Projected Gradient Ascent

The first optimization method that we consider is projected gradient ascent (PGA) (Boyd and Vandenberghe 2004). PGA iteratively applies a gradient update followed by a projection step. Let $\mathcal{P}(\tilde{\rho}, M)$ denote the projection operator projecting $\tilde{\rho}$ onto the space of feasible occupancy measures for the MDP M, i.e., it returns the solution to the optimization problem

$$\begin{aligned}
&\min_{\rho} \mathcal{D}(\rho, \tilde{\rho}) \\
&\textit{subject to} \\
&\quad \sum_{a \in A} \rho(s,a) = \sum_{s' \in S} \sum_{a' \in A} P(s|s',a') \rho(s',a') \\
&\qquad\qquad \textit{for all} \quad s \in S, \\
&\quad \sum_{s \in S} \sum_{a \in A} \rho(s,a) = 1, \\
&\quad \rho(s,a) \geq 0 \qquad\qquad \textit{for all} \quad s \in S, a \in A.
\end{aligned} \tag{11}$$

Algorithm 1 Diverse Stochastic Planning with Projected Gradient Ascent

```
Input: An MDP $M = (S, A, P, R)$, number of policies $k$, tradeoff parameter $\lambda$, step size $\eta^t$, maximum number of iterations $T$, convergence tolerance $\epsilon$
Output: Occupancy measures $\rho_1^T, \rho_2^T, \ldots, \rho_k^T$
Initialize $\rho_1^0, \ldots, \rho_k^0$ by randomly sampling from $\Delta_M$
for $t = 0, \ldots, T$ do
    for $i = 1, \ldots, k$ do
        Find $\rho_i^{t+1/2} \leftarrow \rho_i^t + \eta^t \nabla_{\rho_i} f(\rho_{1:k}^t)$
        Compute $\rho_i^{t+1} \leftarrow \mathcal{P}\left(\rho_i^{t+1/2}, M\right)$
    end for
    Find the gradient mapping $h^t := \frac{1}{\eta^t}(\rho_{1:k}^{t+1} - \rho_{1:k}^t)$
    if $h^t \leq \epsilon$ then
        Return $\rho_1^{t+1}, \rho_2^{t+1}, \ldots, \rho_k^{t+1}$
    end if
end for
Return $\rho_1^T, \rho_2^T, \ldots, \rho_k^T$
```

We choose the projection metric to be the ℓ_2-norm, i.e., $\mathcal{D}(\rho, \tilde{\rho}) := \|\rho - \tilde{\rho}\|_2^2$. The details of the PGA algorithm are outlined in Algorithm 1. Let Δ_M represent the space of feasible occupancy measures defined by the constraints in (10). We initialize the occupancy measures by first defining random policies for the given MDP, i.e., policies with random probability distributions over actions in each state. The algorithm terminates once the convergence criteria are met, e.g., the gradient mapping (Nesterov 2013) defined as

$$h^t := \frac{1}{\eta^t}(\rho_{1:k}^{t+1} - \rho_{1:k}^t)$$

hits a target threshold or the number of iterations exceeds a prespecified number.

It is worth noting that the projection step (11) is much simpler than the original constrained nonlinear optimization in (10) because it is a convex optimization problem, and hence, amenable to efficient solutions.

Frank-Wolfe Algorithm

Even though PGA can decouple the projection step for each policy, it still has to solve a convex optimization problem for each policy at each iteration. To avoid this complexity, we propose the use of the Frank-Wolfe (FW) algorithm (Frank and Wolfe 1956). Every iteration of FW aims to move toward a minimizer of the linear approximation of the original objective function at the current point. Due to this fact, it has gained popularity for optimization problems with structured constraint sets. In particular, the linearity of the constraints in (10) turns every iteration of FW into a linear optimization problem. We implement FW with adaptive step sizes (Lacoste-Julien 2016) and backtracking line search, as presented in Algorithm 2. At iteration t, the algorithm finds a feasible point s^t within the set of feasible policies, Δ_M, that minimizes the linear approximation of $f(\rho_{1:k})$ at the current point. Then, it moves in the direction of s^t by a step size

Algorithm 2 Diverse Stochastic Planning with Frank-Wolfe Algorithm

```
Input: An MDP $M = (S, A, P, R)$, number of policies $k$, tradeoff parameter $\lambda$, maximum number of iterations $T$, convergence tolerance $\epsilon$
Output: Occupancy measures $\rho_1^T, \rho_2^T, \dots, \rho_k^T$
Initialize $\rho_1^0, \dots, \rho_k^0$ by randomly sampling from $\Delta_M$
for $t = 0, \dots, T$ do
    Compute $s^t := \arg\max_{s \in \Delta} \langle s, \nabla_{\rho_{1:k}} f(\rho_{1:k}^t) \rangle$
    Find FW update direction $d^t := s^t - \rho_{1:k}^t$
    Find FW gap $g^t := \langle d^t, \nabla_{\rho_{1:k}} f(\rho_{1:k}^t) \rangle$
    if $g^t \leq \epsilon$ then
        Return $\rho_1^t, \rho_2^t, \dots, \rho_k^t$
    else
        Compute $\gamma^t = \arg\max_{\gamma \in [0,1]} f(\rho_{1:k}^t + \gamma d^t)$
    end if
    $\rho_{1:k}^{t+1} \leftarrow \rho_{1:k}^t + \gamma^t d^t$
end for
Return $\rho_1^T, \rho_2^T, \dots, \rho_k^T$
```

γ^t that is computed using a line search. We efficiently implement the line search using backtracking. The algorithm terminates once the FW gap defined as

$$g^t := \langle d^t, \nabla_{\rho_{1:k}} f(\rho_{1:k}^t) \rangle$$

falls below a given tolerance or the number of iterations reaches a prespecified number T.

Theoretical Guarantees

Next, we prove that by applying PGA and FW on a slightly revised problem one can establish non-asymptotic convergence rates to a stationary point.

Let $\Delta_{M,\delta} = \Delta_M \cap \{\rho_{1:k} | \rho_i \geq \delta, \forall i \in [k]\}$ for some $\delta > 0$ represent a restricted space for occupancy measures. In the next lemma, we prove that the gradient of the objective function $f(\rho_{1:k})$ is Lipschitz continuous over the restricted space $\Delta_{M,\delta}$.

Lemma 1. *Let $\delta \in (0, 1)$. The gradient of the objective function $f(\rho_{1:k})$ defined in* (10) *is L-Lipschitz over $\Delta_{M,\delta}$. That is,*

$$\|\nabla f(\rho_{1:k}) - \nabla f(\rho'_{1:k})\|_2 \leq L \|\rho_{1:k} - \rho'_{1:k}\|_2,$$
$$\forall\ \rho_{1:k}, \rho'_{1:k} \in \Delta_{M,\delta}, \quad L := \lambda \frac{1+\delta}{4\delta^2}.$$

Proof. First, we note that the linear term of $f(\rho_{1:k})$ does not contribute to the Lipschitzness. Moreover, the diversity term has been normalized. Therefore, if we can show that $\mathrm{JSD}(\rho_i \| \rho_j)$ has a L'-Lipschitz gradient, then we can conclude that $f(\rho_{1:k})$ has a $\lambda L'$-Lipschitz gradient. To show that $\mathrm{JSD}(\rho_i \| \rho_j)$ has Lipschitz gradient, we start by computing an entry of its Hessian $\nabla^2 \mathrm{JSD}(\rho_i \| \rho_j)$. Let $x = (s, a) \in X$ be an arbitrary state-action pair where $X = S \times A$. Then, we have

$$\mathrm{JSD}(\rho_i \| \rho_j) = \frac{1}{2} \sum_{x \in X} \rho_i(x) \log \frac{2\rho_i(x)}{\rho_i(x) + \rho_j(x)} + \frac{1}{2} \sum_{x \in X} \rho_j(x) \log \frac{2\rho_j(x)}{\rho_i(x) + \rho_j(x)}.$$

Taking the derivative with respect to an arbitrary x, we obtain

$$\frac{\partial \mathrm{JSD}(\rho_i \| \rho_j)}{\partial \rho_i(x)} = \frac{1}{2} \log \frac{2\rho_i(x)}{\rho_i(x) + \rho_j(x)}.$$

By some straightforward calculation, one can see that the Hessian is sparse. More specifically, it holds that

$$\frac{\partial^2 \mathrm{JSD}(\rho_i \| \rho_j)}{\partial \rho_i(x) \partial \rho_i(x)} = \frac{\rho_j(x)}{2\rho_i(x)(\rho_i(x) + \rho_j(x))},$$
$$\frac{\partial^2 \mathrm{JSD}(\rho_i \| \rho_j)}{\partial \rho_i(x') \partial \rho_i(x)} = 0,$$
$$\frac{\partial^2 \mathrm{JSD}(\rho_i \| \rho_j)}{\partial \rho_j(x) \partial \rho_i(x)} = -\frac{1}{2(\rho_i(x) + \rho_j(x))},$$
$$\frac{\partial^2 \mathrm{JSD}(\rho_i \| \rho_j)}{\partial \rho_j(x') \partial \rho_i(x)} = 0,$$

where $x \neq x'$. Therefore, given that only two entries of each column of the Hessian are nonzero, using the Gershgorin circle theorem (Horn and Johnson 2012), we can show that

$$\|\nabla^2 \mathrm{JSD}(\rho_i \| \rho_j)\|_2 \leq \left| \frac{\rho_j(x)}{2\rho_i(x)(\rho_i(x) + \rho_j(x))} \right| + \left| -\frac{1}{2(\rho_i(x) + \rho_j(x))} \right| \leq \frac{1}{4\delta^2} + \frac{1}{4\delta} = \frac{1+\delta}{4\delta^2}.$$

Hence, $\nabla \mathrm{JSD}(\rho_i \| \rho_j)$ is $(1+\delta)/(4\delta^2)$-Lipschitz and consequently, $\nabla f(\rho_{1:k})$ is $\lambda(1+\delta)/(4\delta^2)$-Lipschitz. ■

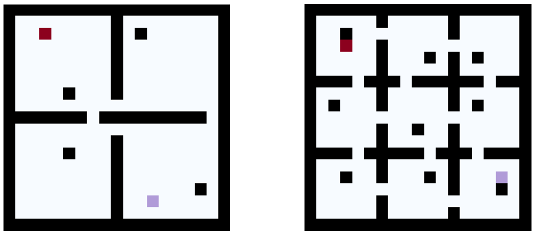
The following two theorems establish that since the gradient is Lipschitz, both PGA and FW are guaranteed to converge to a stationary point.

Theorem 1 (Based on Theorem 6.5 in (Lan 2020))**.** *Define the minimal gradient mapping of the PGA algorithm as $\|\tilde{h}\|_2 := \min_{0 \leq t \leq T} \|h^t\|_2$ encountered by the iterates during the algorithm until the T^{th} iteration. Suppose that the stepsizes $\{\eta^t\}$ in the PGA scheme are chosen such that $0 < \eta^t < 2/L$, where L is the Lipschitz constant of the gradient of $f(\rho_{1:k})$ on $\Delta_{M,\delta}$. Then it holds that*

$$\|\tilde{h}\|_2 \leq \sqrt{\frac{f_\delta^* - f(\rho_{1:k}^0)}{\sum_{t=0}^T \eta^t (1 - L\eta^t/2)}}, \tag{12}$$

where f_δ^* denotes the optimal solution of (10) *over the restricted domain. In particular, If $\eta^t = 1/L$, then*

$$\|\tilde{h}\|_2 \leq \sqrt{\frac{2L(f_\delta^* - f(\rho_{1:k}^0))}{T+1}}. \tag{13}$$



(a)

(b)

Figure 1: Example realizations of the (a) four-room and (b) nine-room grid worlds. Obstacles and walls are black, the initial state is red, and the goal state is purple.

Theorem 2 (Based on Theorem 1 in (Lacoste-Julien 2016))**.** *Define the minimal FW gap as* $\|\tilde{g}\|_2 := \min_{0\leq t\leq T} \|g^t\|_2$ *encountered by the iterates during the algorithm until the* T^{th} *iteration. Consider running the FW algorithm with the adaptive stepsize strategy specified in Line 11 of Algorithm 2. Then, it holds that*

$$\|\tilde{g}\|_2 \leq \frac{\max\{2(f^*_\delta - f(\rho^0_{1:k})), \operatorname{diam}(\Delta_{M,\delta})^2 L\}}{\sqrt{T+1}}, \qquad (14)$$

where $\operatorname{diam}(\Delta_{M,\delta})$ *is the diameter of* $\Delta_{M,\delta}$*,* L *is the Lipschitz constant of the gradient of* $f(\rho_{1:k})$*, and* f^*_δ *denotes the optimal solution of* (10) *over the restricted domain.*

Experiments and Results

We evaluate the performance of the proposed approach in a series of navigation problems as detailed next.

Grid World Design

The grid worlds are two-dimensional nineteen-by-nineteen rectangular grids. The state of the agent is its current position. The agent receives a large reward for reaching a defined goal state, after which it transitions back to an initial state. In all states other than the goal, the agent has five choices of actions that correspond to moving down, left, up, right, or stopping and staying in the same place. The 'stop' action is deterministic; for all other actions, the probability of transitioning to the desired successor state is given by a 'correct transition' hyper-parameter, denoted α. If the transition is not successful, the agent will transition to another neighboring state randomly. The environment is filled with obstacles and reaching an obstacle state results in a large penalty. We note that this design could represent, for example, the environment of a robot vacuum or guard robot.

We define two different grid world types. In the first, which we refer to as the four-room grid world, the environment consists of four eight-by-eight rooms arranged in a two-by-two grid. There is a reward of -200 for reaching states with walls and obstacles, a reward of 400 for reaching the goal, and a reward of -4 in all other states to shape the agent behavior. In the second grid world, there are nine total five-by-five rooms arranged in a three-by-three grid. We refer to this setup as the nine-room grid world. In this setup, there is a reward of -40 for reaching walls and obstacles, a reward of 200 for reaching the goal, and a reward of -1.2 in all other states. Both environments have walls separating adjacent rooms with a single door, represented by a hole in the wall, linking the rooms. There is a single additional obstacle placed within each room. In both worlds, the initial state is located in the top left room, and the goal state is located in the bottom right room.

Opt. Alg.	Reward/policy	Diversity	Runtime (s)
PGA	-39.02	0.34	1488.76
FW	13.24	0.50	26.90

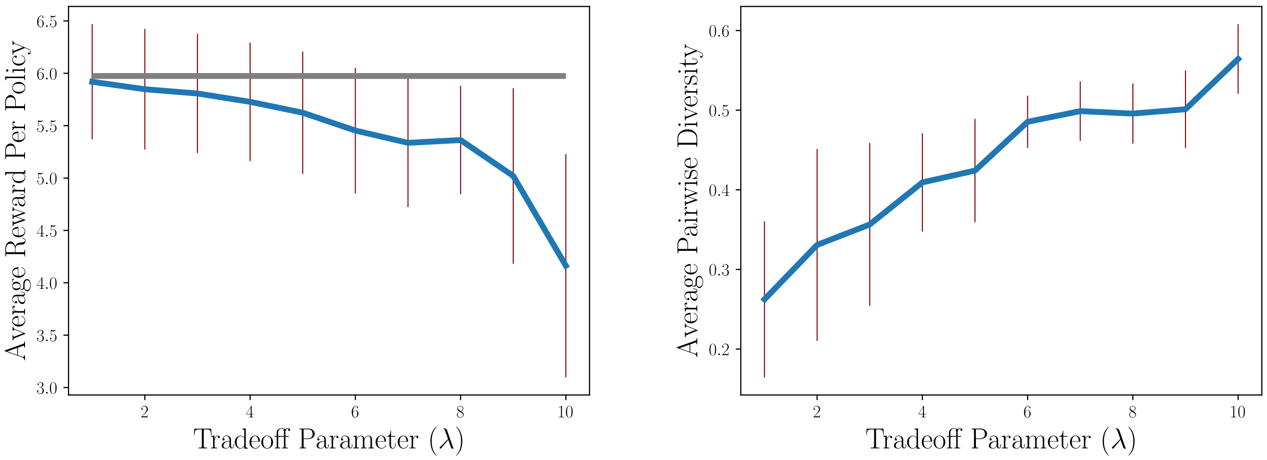
Figure 2: Performance of PGA and FW on the four-room grid world with $\lambda = 8$, $k = 2$, and $\alpha = .95$. Results are averaged over ten trials.

To evaluate the robustness of our proposed approach, we test the performance over a range of trials. In each trial, the location of the agent and the goal state within their respective rooms, the locations of the doors in each wall, and the locations of the obstacles within the rooms are randomized. Figure 1 shows a single trial grid world for both the four- and nine-room setups.

Frank-Wolfe and Projected Gradient Ascent

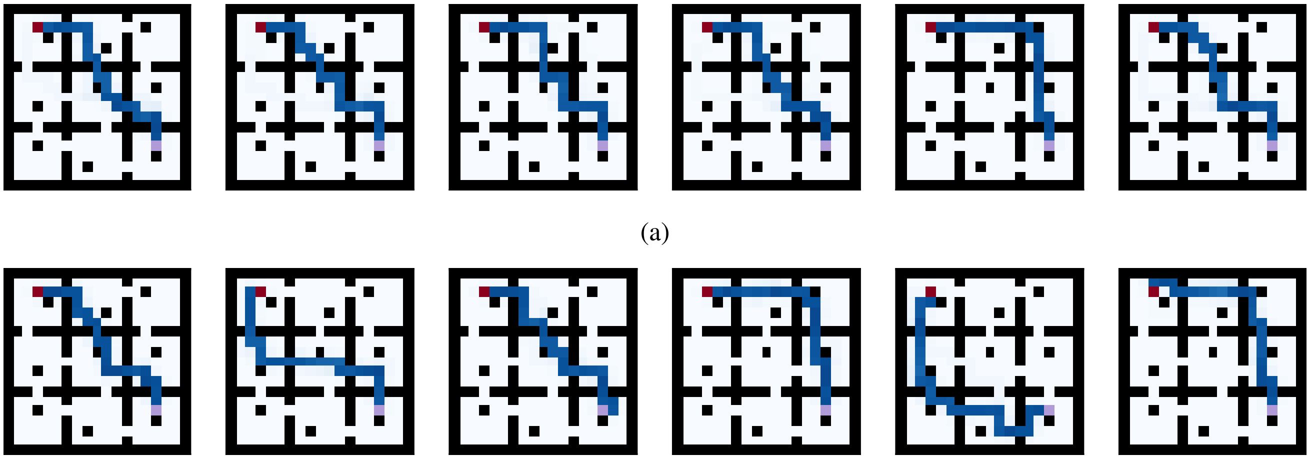
Here we compare the performance of the FW and PGA optimization algorithms. We terminate PGA when the maximum iteration number is reached or when the difference in norms between consecutive solutions falls below a tolerance threshold of .01. We use Sequential Least Squares Programming (Nocedal and Wright 2006) to solve the projection step for each iteration.. The Sequential Least Squares Programming algorithm terminates after ten iterations or when a stationarity condition is met. We implement the FW algorithm with a shrinkage factor of $\gamma = .5$ for the backtracking line search. The FW algorithm terminates when the Frank-Wolfe gap falls below a tolerance of .001 or when the maximum iteration number is reached. We set the maximum iteration number for both approaches as $T = 30$. The approaches were implemented in Python 3.8.8 on a Dell XPS 13 9370 computer with 16 GB of RAM and 1.6 GHz processors. To run these experiments see the code in the following repository: https://github.com/Evan1578/MDPdiversity.

We evaluate performance using the four-room grid world with the correct transition parameter $\alpha = .95$, the number of policies in the return set $k = 2$, and the tradeoff parameter $\lambda = 8$. Figure 2 shows the average performance over ten trials. FW is clearly superior to PGA in both performance and computational efficiency. This is because PGA involves solving a constrained least-squares optimization problem for each policy at each iteration to project the policies back onto the feasible space. Even small errors in the projection can considerably deteriorate the near-optimality and diversity of the policies. In contrast, FW only requires a linear program to be solved at each iteration. The solution to the linear program lies in the feasible space by construction and thus there are no issues with stability. We use FW as the optimization algorithm in the subsequent experiments.



(a) (b)

Figure 3: The reward and diversity of the policies found as a function of the tradeoff parameter for ten nine-room grid-world trials with $\alpha = .95$ and $k = 6$. (a) Mean and standard deviation of the average reward per policy. The reward for the optimal policy is displayed in grey. (b) Mean and standard deviation of the average pairwise diversity.



(a)

(b)

Figure 4: Sample state-occupancy maps from a nine-room grid-world trial with $k = 8$, $\alpha = .95$, and two different tradeoff parameter values. Obstacles and walls are black, the initial state is red, the goal state is purple, and the state-occupancy probability is blue. (a) The tradeoff parameter is two. (b) The tradeoff parameter is eight.

Role of the Tradeoff Parameter

The tradeoff parameter plays a crucial role in ensuring a proper balance between the near-optimality of the candidate solutions and the diversity of the set of solutions. Testing the performance as a function of the tradeoff parameter provides important insights into the performance and properties of our proposed approach. Here we evaluate the performance for a range of tradeoff parameters using the nine-room grid world. We fix the correct transition parameter as $\alpha = .95$, and set the number of policies in the return set as $k = 6$. This k value is the number of unique door combinations the agent can take to reach the goal without cycling or other undesirable behavior.

Figure 3 shows the average reward per policy and the average pairwise diversity over ten trials. As expected, the pairwise diversity shows a marked increase as a function of the tradeoff parameter. The average reward decreases slightly as the tradeoff parameter increases until it begins to fall sharply around $\lambda = 8$. This is the point where it becomes optimal in some trials to find policies that do not reach the goal but have maximal diversity. Up to this point, our approach is still able to find increasingly diverse near-optimal solutions.

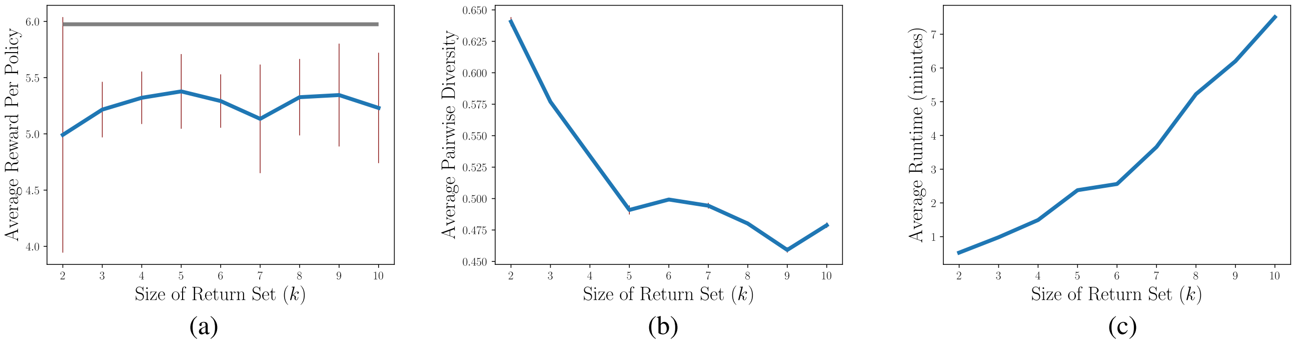


Figure 5: The reward and diversity of the policies found, and the computational efficiency of the optimization, as a function of the size of the return set for ten nine-room grid-world trials with $\alpha = .95$ and $\lambda = 8$. (a) Mean and standard deviation of the average reward per policy. The reward for the optimal policy is displayed in grey. (b) Mean and standard deviation of the average pairwise diversity. (c) Average runtime, in minutes, of the optimization.

This behavior can also be observed in Figure 4, which shows sample state-occupancy maps for a small range of lambda values and a single trial. The state-occupancy measure is the long-run expected probability of being in a given state s, i.e., $\rho(s) = \sum_{a \in A} \rho(s, a)$. With $\lambda = 2$, our approach finds several policies with nearly identical behavior. With $\lambda = 8$, the algorithm finds policies that utilize increasingly diverse strategies to reach the goal and traverse through many of the doors and rooms in the grid world. Note, however, that even with $\lambda = 8$, policies one and three and policies four and six have relatively similar behavior as they utilize the same door combinations to reach the goal. This can be explained by the fact that our approach finds only a local minimum in the loss landscape, and by the fact that even with high values of λ the configuration of the doors and obstacles can limit the number of meaningfully distinct near-optimal policies.

Finding More Policies

We show how varying the desired number of policies in the return set affects performance using the nine-room grid world. Here we set the value of the tradeoff parameter $\lambda = 8$ based on the results in the previous section and again set the correct transition parameter $\alpha = .95$. Figure 5 shows the average reward per policy, the average pairwise diversity of the policies, and the average runtime of the FW algorithm as a function of the size of the return set. It can be observed that the runtime of the optimization algorithm increases linearly with the size of the return set. Further, there is a marked decrease in the average pairwise diversity as the size of the return set grows. This provides further evidence that the environment provides a natural limit on the number of meaningfully diverse policies that can be obtained. Please see the extended version of this manuscript (Ghasemi et al. 2020) for a more detailed investigation of the effect of the size of the return set on performance, as well as an examination of the role of the correct transition parameter.

Conclusion and Future Work

In this work, we considered the problem of stochastic planning in situations where the objective function is known to be only partially specified. In this setting, we proposed generating a representative set of near-optimal policies with respect to the known objective. To that end, we formulated a nonlinear optimization problem that finds a small set of near-optimal and diverse policies. We showed that it is possible to efficiently solve the optimization problem using the Frank-Wolfe method and proved non-asymptotic convergence rates. We then compared the performance of the Frank-Wolfe method with projected gradient ascent and investigated the role of the hyperparameters using a series of navigation problems.

Our results show that the choice of the tradeoff parameter and the size of the return set play an important role in the performance of our approach. As the tradeoff parameter and the size of the return set increase, our approach is able to find increasing numbers of meaningfully distinct near-optimal policies up to a limit that is related to the structure of the environment. An interesting future extension of our approach would be investigating the utility of these near-optimal diverse strategies in generating effective collaboration between groups of autonomous agents.

Acknowledgments

This work was supported in part by NIH grant R00EB027181, DARPA grant D19AP00004, and AFRL grant FA9550-19-1-0169. We thank Abolfazl Hashemi and Yue Yu for their constructive discussion and also thank the reviewers for their valuable feedback.

References

Bach, F. 2013. *Learning with Submodular Functions: A Convex Optimization Perspective*, volume 6 of *Foundations and Trends in Machine Learning*. Hanover, MA: Now Publishers Inc.

Balch, T.; et al. 1997. Learning Roles: Behavioral Diversity in Robot Teams. In *AAAI Workshop on Multiagent Learning*.

Bertsekas, D. P. 1995. *Dynamic Programming and Optimal Control*, volume 1. Belmont, MA: Athena Scientific.

Boyd, S.; and Vandenberghe, L. 2004. *Convex Optimization*. Cambridge: Cambridge University Press.

Briët, J.; and Harremoës, P. 2009. Properties of Classical and Quantum Jensen-Shannon Divergence. *Physical Review A* 79(5): 052311.

Bryce, D. 2014. Landmark-Based Plan Distance Measures for Diverse Planning. In *Proceedings of International Conference on Automated Planning and Scheduling*, 56–64.

Coman, A.; and Munoz-Avila, H. 2011. Generating Diverse Plans Using Quantitative and Qualitative Plan Distance Metrics. In *AAAI*, 946–951.

Doncieux, S.; and Mouret, J.-B. 2013. Behavioral diversity with multiple behavioral distances. In *IEEE Congress on Evolutionary Computation*, 1427–1434.

Eysenbach, B.; Gupta, A.; Ibarz, J.; and Levine, S. 2018. Diversity Is All You Need: Learning Skills Without a Reward Function. *arXiv preprint arXiv:1802.06070* .

Frank, M.; and Wolfe, P. 1956. An Algorithm for Quadratic Programming. *Naval Research Logistics Quarterly* 3(1-2): 95–110.

Gaier, A.; Asteroth, A.; and Mouret, J.-B. 2020. Automating Representation Discovery with MAP-Elites. *arXiv preprint arXiv:2003.04389* .

Ghasemi, M.; Crafts, E. S.; Zhao, B.; and Topcu, U. 2020. Multiple Plans are Better than One: Diverse Stochastic Planning. *arXiv preprint arXiv:2012.15485* .

Hebrard, E.; Hnich, B.; O'Sullivan, B.; and Walsh, T. 2005. Finding Diverse and Similar Solutions in Constraint Programming. In *AAAI*, volume 5, 372–377.

Hoffmann, J.; and Nebel, B. 2001. The FF Planning System: Fast Plan Generation Through Heuristic Search. *Journal of Artificial Intelligence Research* 14: 253–302.

Horn, R. A.; and Johnson, C. R. 2012. *Matrix Analysis*. Cambridge: Cambridge University Press.

Jackson, E. C.; and Daley, M. 2019. Novelty Search for Deep Reinforcement Learning Policy Network Weights by Action Sequence Edit Metric Distance. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 173–174.

Katz, M.; and Sohrabi, S. 2020. Reshaping Diverse Planning. In *AAAI*, 9892–9899.

Lacoste-Julien, S. 2016. Convergence Rate of Frank-Wolfe for Non-convex Objectives. *arXiv preprint arXiv:1607.00345* .

Lan, G. 2020. *First-order and Stochastic Optimization Methods for Machine Learning*. Switzerland: Springer Nature.

Lehman, J.; and Stanley, K. O. 2008. Exploiting Open-Endedness to Solve Problems Through the Search for Novelty. In *ALIFE*, 329–336.

Mouret, J.-B.; and Clune, J. 2015. Illuminating Search Spaces by Mapping Elites. *arXiv preprint arXiv:1504.04909* .

Myers, K. L. 2006. Metatheoretic Plan Summarization and Comparison. In *International Conference on Automated Planning and Scheduling*, 182–192.

Myers, K. L.; and Lee, T. J. 1999. Generating Qualitatively Different Plans Through Metatheoretic Biases. In *AAAI/IAAI*, 570–576.

Nesterov, Y. 2013. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. New York: Springer Science & Business Media.

Nguyen, T. A.; Do, M.; Gerevini, A. E.; Serina, I.; Srivastava, B.; and Kambhampati, S. 2012. Generating Diverse Plans to Handle Unknown and Partially Known User Preferences. *Artificial Intelligence* 190: 1–31.

Nocedal, J.; and Wright, S. 2006. *Numerical Optimization*. New York: Springer Science & Business Media.

Parker-Holder, J.; Pacchiano, A.; Choromanski, K.; and Roberts, S. 2020. Effective Diversity in Population-Based Reinforcement Learning. *arXiv preprint arXiv:2002.00632* .

Petit, T.; and Trapp, A. C. 2015. Finding Diverse Solutions of High Quality to Constraint Optimization Problems. In *International Joint Conference on Artificial Intelligence*.

Pugh, J. K.; Soros, L. B.; and Stanley, K. O. 2016. Quality Diversity: A New Frontier for Evolutionary Computation. *Frontiers in Robotics and AI* 3: 40.

Srivastava, B.; Nguyen, T. A.; Gerevini, A.; Kambhampati, S.; Do, M. B.; and Serina, I. 2007. Domain Independent Approaches for Finding Diverse Plans. In *IJCAI*, 2016–2022.