Bi-Criteria Diverse Plan Selection via Beam Search Approximation

Shanhe Zhong[1], **Pouya Shati**[2,3], **Eldan Cohen**[1]

[1] Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Canada
[2] Department of Computer Science, University of Toronto, Toronto, Canada
[3] Vector Institute, Toronto, Canada
{jszhong, ecohen}@mie.utoronto.ca, pouya@cs.utonronto.ca

Abstract

Recent work on diverse planning has focused on a two-step setting where the first step consists of generating a large number of plans, and the second step consists of selecting a subset of plans that maximizes diversity. For the second step, previous work has focused on solving a combinatorial optimization problem for diverse subset selection that can be approximated using greedy search. In this work, we propose a flexible, bi-criteria framework for diverse plan selection. Our framework consists of optimizing both quality and diversity, generalizing previous work and providing flexibility to prioritize one objective over the other. We consider two quality and two diversity measures and show that greedy search guarantees an approximation with a constant ratio for certain configurations based on established results in the literature. To allow users to trade off additional computation for better solutions, we introduce a beam search approximation that generalizes the greedy search, and we provide approximation guarantees on the obtained solutions. Finally, we conduct extensive experiments that show that: (1) our flexible bi-criteria framework allows us to obtain solutions of better quality while still maintaining a high degree of diversity; (2) our beam search approximation obtains significant improvement in performance over greedy search and, for a large number of instances, is able to generate solutions that are equal to or better than those obtained by an exact MIP solver with a significantly higher runtime limit.

1 Introduction

Due to the expressive modelling capabilities inherent in automated planning, planners have become viable tools in various real-world applications, including robotics (Barrios et al. 2011; Karpas and Magazzeni 2020), logistics (Sousa and Tavares 2013; Pedersen and Krüger 2015), healthcare (Fuentetaja et al. 2020; Lindsay et al. 2022) and much more. Traditionally, the primary focus of planners has been on generating a single optimal or high-quality satisficing plan. However, applications in areas such as malware detection (Boddy et al. 2005), systems with unknown or partially known user preferences (Nguyen et al. 2012), automated analysis of streaming data (Riabov et al. 2015), and risk management (Sohrabi et al. 2018) underscore the necessity for a diverse set of plans. More broadly, many real-world scenarios can benefit from having a diverse set of plans. In dynamic and complex environments, a diverse set of plans enhances system robustness by offering alternative courses of action and increases the chance that one of the plans is acceptable to the user (Bloem 2015).

Recent work on diverse planning (Vadlamudi and Kambhampati 2016), including the state-of-the-art diverse planner proposed by Katz and Sohrabi (2020), focuses on a two-stage approach where a candidate set of satisficing plans is generated in the first step, and a diverse subset of k plans is selected in the second step. In this work, we focus on the second step of diverse plan selection. We extend existing approaches by providing a more flexible framework for defining objectives that consider both diversity and quality and by providing a more flexible approximation algorithm that allows users to trade off additional computation for better solutions. We make the following contributions:

- We propose a bi-criteria framework for diverse plan selection in two-stage diverse planning that considers both the diversity and the quality of solutions and generalizes previous work. By making the connection to previous work on diverse subset selection, we highlight key configurations of our approach for which a greedy search enjoys theoretical guarantees w.r.t. solution quality.
- To allow a flexible trade-off between solution quality and computational cost, we propose a beam search-based approximation for optimizing our bi-criteria framework, and we provide approximation guarantees on the obtained solutions.
- We present extensive experiments that show that: (i) our bi-criteria framework provides finer-grained control over the trade-off between diversity and quality and, in particular settings, can even provide gains in one criterion with little to no loss on the other; (ii) our beam search approximation is able to find better solutions compared to greedy search and can often obtain solutions that are equal to or better than those obtained by a MIP solver with a significantly higher runtime limit.

2 Background

In this work, we mostly follow the notations used by Katz and Sohrabi (2020). A SAS$^+$ planning task (Bäckström and Nebel 1995) is characterized by a tuple $\langle \mathcal{V}, \mathcal{A}, s_0, s^* \rangle$, where

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

$\mathcal{V} = \{v_1, ..., v_m\}$ is a set of *state variables*. A *state* is a complete assignment of values to the state variables, and a *partial state* can feature state variables with undefined value u^*. We write $s[v]$ to denote the value of the state variable v in a state s. A partial state p is said to be *consistent* with a state s if and only if for all $v \in \mathcal{V}$, $p[v] = s[v]$ unless $p[v] = u^*$. $\mathcal{A} = \{a_1, ..., a_n\}$ is a finite set of *actions*, where each action a is a pair $\langle pre(a), eff(a)\rangle$ of partial states called *preconditions* and *effects*. An action a is *applicable* in a state $s \in \mathcal{S}$ if and only if $pre(a)$ is consistent with s. Applying a changes the value of v to $eff(a)[v]$, for all v defined in $eff(a)$. The *cost* if an action a is an assignment defined by a mapping from action set $\mathcal{A}$ to the non-negative real numbers $C : \mathcal{A} \rightarrow \mathbb{R}_0^+$. The *cost* of an action sequence π, denoted by $C(\pi)$, is the sum of the costs of the action in it. s_0 is the *initial state* and s^* is the *goal state*. An action sequence $\pi = \langle a_1, ..., a_j\rangle$ is applicable in s if there exist states $s_0, \cdots, s_j$ such that (i) $s_0 = s$, and (ii) for each $1 \leq i \leq j$, a_i is applicable in s_{i-1} and $s_i = s_{i-1}[\![a_i]\!]$; we also say s_j is consistent with $s_0[\![\pi]\!]$. π is a *satisficing plan* if and only if π is applicable in s_0 and s_* is consistent with $s_0[\![\pi]\!]$.

2.1 Diverse Planning

A planner can be used for diverse planning if it can generate a diverse set of satisficing plans of a certain cardinality k, and several approaches were proposed to solve this problem. An early line of work (Coman and Munoz-Avila 2011; Nguyen et al. 2012; Roberts, Howe, and Ray 2014; Bloem 2015) has utilized a greedy strategy, similar to what was proposed in Hebrard et al. (2005), in the context of diverse planning. This strategy involves a process of generating a candidate plan, adding this plan to a solution set, and providing feedback to the planner so that it tries to find new plans that are distant from the current solution set. Following this process, subsequent candidates are then iteratively generated, but are only added to the solution set if the diversity criterion is satisfied. Vadlamudi and Kambhampati (2016) proposed a complete *two-stage optimization* approach where a large number of satisficing plans (that are of bounded quality) are generated in the first step as candidates and all k-sized combinations of these plans are explored in the second step.

Katz and Sohrabi (2020) adopted this two-stage optimization approach and proposed the *forbid iterative* (FI) planner that was shown to be empirically superior to many of the existing diverse planners in terms of coverage, solution quality, and diversity. The FI planner, in essence, iteratively reformulates the task once a plan is generated to forbid all previously generated plans from being generated again. This is proven to be a powerful approach to diverse planning as FI has obtained state-of-the-art performance. The second stage of the two-stage approach, which we call the *diverse plan selection* problem, involves selecting a subset of plans subject to a cardinality constraint k that maximizes diversity (Vadlamudi and Kambhampati 2016). Katz and Sohrabi (2020) tested and later proposed using mixed-integer programming (MIP) to obtain the set with optimal minimum pairwise diversity (Katz, Sohrabi, and Udrea 2022). However, solving the MIPs can be quite expensive as their MIP formulation features pairwise linear constraints which grow quadratically with the number of plans n in the candidate set. To provide a faster alternative, Katz and Sohrabi (2020) also proposed to use greedy search, a low-order polynomial-time search algorithm, for its simplicity and efficiency. Greedy search has shown surprisingly strong performance in terms of solution quality. In particular, experiments show that greedy search is able to achieve optimality around 60% of the time (Katz, Sohrabi, and Udrea 2022). The greedy search for diverse planning introduced by Katz and Sohrabi (2020) is provided in Algorithm 1.

Algorithm 1: Greedy Search for Diverse Plan Selection

Input: A set of generated plans Π, a set function modelling diversity $f(\cdot)$
Parameter: Cardinality constraint k
Output: $G \subset \Pi$ where $|G| = k$

1: $\Pi \leftarrow sort(\Pi, C(\cdot), ascending)$
2: $G \leftarrow \text{argmax}_{\{\pi_i,\pi_j\}\in\Pi} f(\{\pi_i, \pi_j\})$
3: **for** $i = 3, ..., k$ **do**
4: $\quad \pi \leftarrow \text{argmax}_{\pi\in\Pi\setminus G} f(G \cup \{\pi\})$
5: $\quad G \leftarrow G \cup \pi$
6: **end for**
7: **return** G

To evaluate the diversity of a set of plans, we first need a measure of similarity (or dissimilarity) between plans. In the context of diverse planning, several similarity measures have been utilized. These similarity measures include the *stability*, *state*, and the *uniqueness* (Katz and Sohrabi 2020). Uniqueness similarity (Roberts, Howe, and Ray 2014) measures whether a plan is a permutation or a subset of another plan that is already selected. It is defined as $\text{sim}_{\text{uniqueness}}(\Pi) = \sum_{\pi_n,\pi_m\in\Pi,\pi_n\neq\pi_m} u(\pi_n, \pi_m)$ where Π is a set of plans, and

$$u(\pi_n, \pi_m) = \begin{cases} 0, & \pi_n \subset \pi_m \\ 0, & \pi_n \setminus \pi_m = \emptyset \\ 1, & otherwise \end{cases}$$

The *stability similarity* (Fox et al. 2006; Coman and Munoz-Avila 2011) measures the ratio of actions that are shared between two plans. It is defined as $\text{sim}_{\text{stability}}(\pi, \pi') = \frac{|A(\pi)\cap A(\pi')|}{|A(\pi)\cup A(\pi')|}$, where $A(\pi)$ denotes the set of actions in π.

Suppose $(s_0, s_1, s_2, ..., s_n)$ and $(s'_0, s'_1, s'_2, ..., s'_{n'})$ are the sequences of states corresponding to plan π and plan π'. Assuming $n' \leq n$, the *state similarity* (Nguyen et al. 2012) is defined as $\text{sim}_{\text{state}}(\pi, \pi') = \frac{1}{n}\sum_{i=1}^{n'} \Delta(s_i, s'_i)$, where $\Delta(s_i, s'_i) = \frac{|s_i\cap s'_i|}{|s_i\cup s'_i|}$ is the similarity between two states.

We note that the original formulation for the stability similarity has been under an action set (Fox et al. 2006; Coman and Munoz-Avila 2011; Katz and Sohrabi 2020), but Katz, Sohrabi, and Udrea (2022) proposed it should be defined for action multi-sets. It follows the same definition except that $A(\pi)$ now denotes the multi-set of actions, allowing for repetitions. In this work, we mainly focus on the original definition for its computational efficiency.

2.2 Beam Search

Greedy search, which always selects the locally optimal decision, can often lead to sub-optimal solutions (Cohen and Beck 2019). In contrast, beam search is a generalization of greedy search that has been widely used in different tasks, most notably in neural sequence decoding (Cohen and Beck 2019, 2021), to mitigate the limitations of greedy search and find better solutions. At every step, beam search keeps a set of the most promising candidates (partial) solutions, with the cardinality of this set called *beam width* (or, alternatively, beam size). As a result, the beam search algorithm enables a trade-off between solution quality and extra computation that grows linearly with beam width. By keeping a larger number of candidate solutions at each step, the beam search explores a larger portion of the search space compared to greedy search, and consequently tends to yield superior solutions. Consider a beam search with a beam width of m. At every iteration, the beam search generates the successors for all partial solutions in the beam and selects the top-scoring m successors as the new beam for the next iteration. Let B_t denote the set of m partial solutions in the beam at iteration t, $B_t = \{B_t^1, B_t^2, ..., B_t^m\}$, where the beam is sorted base on a scoring function $f(\cdot)$ in descending order, i.e., $B_t^i \geq B_t^{i+1}$. Let B_t' denote the set of all successors of the beam of iteration $t-1$, $B_t' = \{B_{t-1}^i \cup \{\pi\}, \pi \in \Pi \setminus B_{t-1}\}$. The beam at iteration t is a then selected from B_t' as follows:

$$B_t = \underset{B_t^1,...,B_t^m \in B_t'}{\operatorname{argmax}} \sum_{i=1}^{m} f(B_t^i). \quad (1)$$

Note that beam search generalizes greedy search, as setting the beam width $m = 1$ will result in a greedy search.

3 Bi-Criteria Framework for Diverse and High-Quality Plan Selection

In this section, we present a framework for diverse plan selection that considers both the *diversity* of the selected set of plans, as well as their *quality* (represented by cost). Previous work on diverse plan selection has only considered plan costs in the first stage of generating plans, by enforcing bounded suboptimality on the generated plans (Katz, Sohrabi, and Udrea 2022). However, in such approaches, using strong bounds can limit the ability to generate a sufficiently diverse set of plans in the first stage. In contrast, using loose bounds can limit our ability to select a high-quality subset of plans in the second stage. Instead, we propose a bi-criteria optimization framework for plan selection that consists of a linear combination of quality and diversity measures, inspired by previous work on diverse subset selections (Borodin, Lee, and Ye 2012; Dasgupta, Kumar, and Ravi 2013). For a given set of plans S, we consider the following bi-criteria objective:

$$f(S) = \alpha g(S) + \beta h(S) \quad s.t. \; |S| = k \quad (2)$$

where $g(\cdot)$ is a quality measure, $h(\cdot)$ is a diversity measure, and the constraint $|S| = k$ is a cardinality constraint on the number of plans. We can control the trade-off between quality and diversity by tweaking $\frac{\alpha}{\beta}$. In particular, Eq. (2) generalizes previous work that focuses solely on diversity in the plan selection, which can be obtained by setting $\alpha = 0$. To our knowledge, this is the first work to consider a bi-criteria optimization framework for diverse plan selection.

By using a bi-criteria framework for diverse plan selection, we can relax the quality bound during the plan generation stage (although we note that our framework is useful in the presence of such bounds). In addition, as sets of longer plans can often lead to increased diversity but have lower quality, optimizing for both diversity and quality can help mitigate this problem.

3.1 Diversity Functions

We consider two well-known functions for measuring the diversity of a selected subset of items, based on a pairwise distance measure between plans, $d(\cdot,\cdot)$. In the context of diverse planning, stability similarity has a corresponding distance function computed by $d(\pi_i, \pi_j) = 1 - sim(\pi_i, \pi_j)$ with sim representing the stability similarity.

Previous work on diverse plan selection has utilized these distance measures to select a diverse subset of plans S of cardinality k, by maximizing the minimum distance between each pair of plans in the subset, denoted by h_{min}:

$$h_{min}(S) = \min_{\pi_i, \pi_j \in S} d(\pi_i, \pi_j)$$

This diversity function ensures all plans in the selected subset S are sufficiently diverse as $h_{min}(S)$ is a lower bound on their pairwise dissimilarities. In this work, we also consider another diversity function, the *sum of pairwise distances* h_{sum} (Borodin, Lee, and Ye 2012), defined as:

$$h_{sum}(S) = \sum_{\pi_i, \pi_j \in S} d(\pi_i, \pi_j).$$

Given the cardinality constraint, a solution that is optimal w.r.t. h_{sum} is also optimal w.r.t. the average pairwise distance (Nguyen et al. 2012) between plans.

For our theoretical guarantees (Section 4), we require that $d(\cdot,\cdot)$ is a *metric*, i.e., it is symmetric, non-negative, and satisfies the triangle inequality. Upon closer inspection, we observe that the stability distance is, in fact, a metric as it is a special case of *Jaccard Distance* (Markov and Larose 2007).[1]

3.2 Quality Functions

Our quality functions are based on the cost of the selected plans. In particular, we consider g_{min}, the *min plan cost* of a selected set of plans S:

$$g_{min}(S) = \min_{\pi \in S} c(\pi),$$

where $c(\pi)$ is the cost of plan π. In addition, we consider g_{sum}, the *sum of plan costs*:

$$g_{sum}(S) = \sum_{\pi \in S} c(\pi).$$

[1]The distance measure that arises from the multi-set definition is also known as the *generalized Jaccard distance*, which also satisfies the triangle inequality (Kosub 2019). The non-negativity and symmetry are trivial; this implies the plan distance based on the multi-set definition is also a metric distance function.

Similar to diversity, we scale the cost of plans between 0 and 1 by using min-max scaling. As we are maximizing the bi-criteria objective, we compute the complement to one such that the lowest-cost plans get the highest score of 1:

$$c(\pi) = 1 - \frac{C(\pi) - \min_{\pi \in \Pi}(C(\pi))}{\max_{\pi \in \Pi}(C(\pi)) - \min_{\pi \in \Pi}(C(\pi))}$$

where Π is the set of plans generated in the first stage of diverse planning. In this way, we can maximize the quality function to optimize plan costs. It should be noted that g_{sum} is *modular*, and therefore also *submodular*, as well as monotonic. For a set function $l : 2^{\Pi} \to \mathbb{R}$ defined on a finite set Π, submodularity refers to a diminishing return, which can be defined as: $l(S \cup \{\pi\}) - l(S) \geq l(S' \cup \{\pi\}) - l(S')$, for all $S \subset S' \subset \Pi$ and $\pi \in \Pi \setminus S'$ (Buchbinder and Feldman 2018). A set function is modular when equality appears in place of the inequality (Wu, Zhang, and Du 2019). Monotonicity, on the other hand, implies a set function is non-decreasing (Krause and Golovin 2014).

4 Efficient Approximation Algorithms

We now discuss efficient approximations for the proposed bi-criteria optimization framework in Eq. (2). In Section 4.1, we review existing approximation guarantees on the performance of greedy search for select combinations of diversity and quality measures in the proposed framework. In Section 4.2, we propose a beam search algorithm for the bi-criteria framework that generalizes the greedy search and offers a more flexible trade-off between computational cost and optimization performance. Then, in Section 4.3, we provide approximation guarantees for the proposed beam search. The existing and the new theoretical results for the greedy search and beam search approximations are summarized in Table 1.

4.1 Greedy Search for Diverse Plan Selection

Previous work on diverse subset selection has found that for some combinations of diversity and quality functions, a solution produced by a greedy search enjoys theoretical guarantees w.r.t. its objective value. In particular, Borodin, Lee, and Ye (2012) showed that optimizing h_{sum}, either on its own or in conjunction with a nonnegative monotone submodular quality function, using greedy search achieves a $\frac{1}{2}$-approximation of the optimal solution. Further, Dasgupta, Kumar, and Ravi (2013) showed that optimizing h_{min} via greedy search leads to a $\frac{1}{2}$-approximation, while optimizing h_{min} in conjunction with a nonnegative monotone submodular quality function via greedy search leads to a $\frac{1}{4}$-approximation.

While previous work on diverse plan selection (Katz and Sohrabi 2020; Katz, Sohrabi, and Udrea 2022) has considered using a greedy search to approximate h_{min}, it was not established that such approximation, when using pairwise distances based on stability similarity, provides theoretical guarantees following Dasgupta, Kumar, and Ravi (2013).

4.2 Beam Search for Diverse Plan Selection

While greedy search enjoys theoretical guarantees, it can, and often does, lead to suboptimal solutions. Further, greedy search does not provide any mechanism for trading off additional computation time for better solutions. Instead, we propose to use beam search, a generalization of greedy that offers much more flexibility: by controlling the beam width m, a user can trade off additional computation (that grows linearly with m) for better solutions. In practice, beam search was found to outperform greedy search in a range of tasks (Yang, Huang, and Ma 2018; Meister, Vieira, and Cotterell 2020; Cohen and Beck 2021). However, previous work has not investigated the use of beam search for the proposed optimization framework in Eq. (2). In particular, it is not known if beam search also enjoys theoretical guarantees and how it compares to greedy search when used to optimize Eq. (2).

Algorithm 2 describes the beam search procedure for optimizing Eq. (2). In adapting the standard beam search algorithm for diverse plan selection, several adjustments are needed to align it with the unique characteristics of this setting. Specifically, when the objective functions depend on the pairwise diversity, it is imperative to initialize the algorithm with a pair of plans. This step addresses the ambiguity that arises from selecting a singular plan at the outset. Moreover, we found that the diversity and quality functions for diverse plan selection often lead to multiple solutions with similar objectives. To avoid bias towards specific structures, we employ a random tie-breaking mechanism.

Algorithm 2: Beam Search for Diverse Plan Selection

Input: A set of generated plans Π, a set-based bi-criteria objective function $f(\cdot)$
Parameter: Cardinality constraint k, Beam width m
Output: $S' \subset \Pi$, $|S'| = k$

1: $B \leftarrow \text{argmax}_{b_1,\ldots,b_m \in \Pi, |b_i|=2} \sum_{i=1}^{m} f(b_i)$
2: **for** $i = 3, \ldots, k$ **do**
3: $\quad B' = successors(B)$
4: $\quad B \leftarrow \text{argmax}_{b_1,\ldots,b_m \in B'} \sum_{i=1}^{m} f(b_i)$. $\{$Break ties randomly.$\}$
5: **end for**
6: $S' \leftarrow \text{argmax}_{b \in B} f(b)$
7: **return** S'

4.3 Approximation Guarantees for Beam Search

In this section, we establish approximation guarantees[2] on beam search for diverse plan selection, as summarized in Table 1. In particular, we extend the approximation guarantees on greedy search from Borodin, Lee, and Ye (2012) and Dasgupta, Kumar, and Ravi (2013) to the beam search setting and introduce additional results for criteria involving g_{min}. All approximation ratios are shown for $k \geq 2$ as it is trivial that beam search can optimize any f for $k = 1$ due to the search space being equal to the solution space.

We note that the approximation guarantees for greedy search do not trivially extend to beam search. Although beam search tends to outperform greedy search, the solutions produced by beam search are *not* guaranteed to dom-

[2]All proofs appear in the supplementary material (Zhong, Shati, and Cohen 2024).

Criteria	Quality	Diversity	Bounds$_G$	Bounds$_B$	Origin$_G$	Origin$_B$
(1)	g_{sum}	$\emptyset$	1.000	1.000	Proposition 1	
(2)	g_{min}	$\emptyset$	1.000	1.000	Proposition 1	
(3)	$\emptyset$	h_{min}	0.500	0.500	(Dasgupta, Kumar, and Ravi 2013)	Theorem 1
(4)	$\emptyset$	h_{sum}	0.500	0.500	(Borodin, Lee, and Ye 2012)	Theorem 2
(5)	g_{sum}	h_{min}	0.250	0.250	(Dasgupta, Kumar, and Ravi 2013)	Corollary 1
(6)	g_{sum}	h_{sum}	0.500	0.500	(Borodin, Lee, and Ye 2012)	Theorem 2
(7)	g_{min}	h_{min}	0.250	0.250	Corollary 1	
(8)	g_{min}	h_{sum}	0.250	0.250	Corollary 1	

Table 1: The approximation ratios of different combinations of quality and diversity functions for both greedy search (G) and beam search (B).

inate solutions produced by greedy search. To demonstrate this, Examples 1 and Example 2 in the supplementary material (Zhong, Shati, and Cohen 2024) show that beam search may end up with worse solutions than greedy search.

To prove the approximation guarantees of beam search on different diverse plan selection objectives, we present Lemma 1.

Lemma 1. *The top solution in each iteration of the beam search procedure, has an equal or higher objective value compared to the best expansion of the top solution from last iteration.*

$$f(B_t^1) \geq f(B_{t-1}^1 \cup \{\pi\}), \forall \pi \in \Pi \setminus B_{t-1}.$$

For criteria (1) and (2) in Table 1 that are based solely on a quality function, namely $f(S) = g_{min}(S)$ and $f(S) = g_{sum}(S)$, it follows from Proposition 1 that we can find at least one optimal solution with beam search optimizing $f(S) = g_{sum}(S)$.

Proposition 1. *Beam search that operates on the objective function $f(S) = g_{sum}(S)$ finds the optimal solution in both $g_{sum}(S) = \sum_{\pi_i \in S} c(\pi_i)$ and $g_{min}(S) = \min_{\pi_i \in S} c(\pi_i)$.*

For criteria (3) in Table 1 that consists solely of the diversity function h_{min}, i.e., $f(S) = h_{min}(S)$, it follows from Theorem 1 that we achieve an approximation ratio of 1/2 to the optimum with one run of beam search optimizing the diversity function.

Theorem 1. *For a diverse plan selection task of size k from the set Π, beam search obtains a $1/2$-approximation to objective function $f(S) = h_{min}(S)$, where $h_{min}(S) = \min_{\pi_i, \pi_j \in S} d(\pi_i, \pi_j)$ and d is a metric distance function.*

For a bi-criteria optimization involving the diversity function h_{sum} and a submodular, monotonic, and non-negative quality function, such as g_{sum} (criteria (6) in Table 1), it follows from Theorem 2 that an approximation ratio of 1/2 can be achieved by optimizing a proxy objective $f(S) = \frac{\alpha}{2} g(S) + \beta h_{sum}(S)$. Further, solely optimizing h_{sum} (criteria (4) in Table 1) is a special case of Theorem 2 where $\alpha g(S) = 0$, hence retaining the 1/2 approximation ratio.

Theorem 2. *For a diverse plan selection task of size k from the set Π, beam search obtains a $1/2$-approximation to the function $f(S) = \alpha g(S) + \beta h_{sum}(S)$ for all monotone, submodular, and non-negative $g(S)$ and $\alpha, \beta \in \mathbb{R}$ where $h_{sum}(S) = \sum_{\pi_i, \pi_j \in S} d(\pi_i, \pi_j)$ and d is a metric distance function.*

For any bi-criteria objective $f(S) = \alpha g_{min}(S) + \beta h(S)$ where $h \in \{h_{min}, h_{sum}\}$ (criteria (7) and (8) in Table 1), since g_{min} is not monotone, we can no longer apply Theorem 2 to obtain a 1/2 approximation ratio. Similarly for a bi-criteria objective with $h = h_{min}$ (criteria (5) and (7) in Table 1), Theorem 1 does not apply. It instead follows from Corollary 1 that we can obtain a weaker approximation of 1/4 with 2 runs of beam search, each optimizing one of the quality and diversity criteria separately. In practice, the performance can be further improved by a third run of beam search that directly optimizes the objective $f(\cdot)$ and selects solutions with the highest objective, retaining the approximation guarantees.

Corollary 1. *The best solution from two beam search runs, one optimizing $g(s)$ and the other optimizing $h(s)$, both guaranteeing an approximation ratio of at least $1/2$, is guaranteed to obtain a $1/4$-approximation to the objective function $f(S) = \alpha g(S) + \beta h(S)$ with $\alpha, \beta \in \mathbb{R}$.*

5 Experimental Results

In this section, we perform experimental evaluation on the effectiveness of the proposed bi-criteria optimization framework and the approximation quality of beam search.

5.1 Experimental Setup

Our experiments are conducted on a 16-core Intel i7-13700K CPU clocked at 3.40GHz. The benchmark set utilized for our experiments is from the International Planning Competition (IPC). We utilized the sym-k loopless planner (von Tschammer, Mattmüller, and Speck 2022) to generate plans for these instances with a time and memory limit of 30 minutes and 4GB, respectively. The limit on the number of generated plans in stage 1 was set to 10,000 in the generation process. Out of the 1,797 instances, the planner was able to generate at least 1,000 plans for 1,025 instances. As a result, the benchmark set consists of these 1,025 tasks. Then, to counteract some of the scalability issues of the MIP, we randomly sample 1,000 plans from the plan pool of every instance with more than 1,000 plans. We also set $k = 5$.

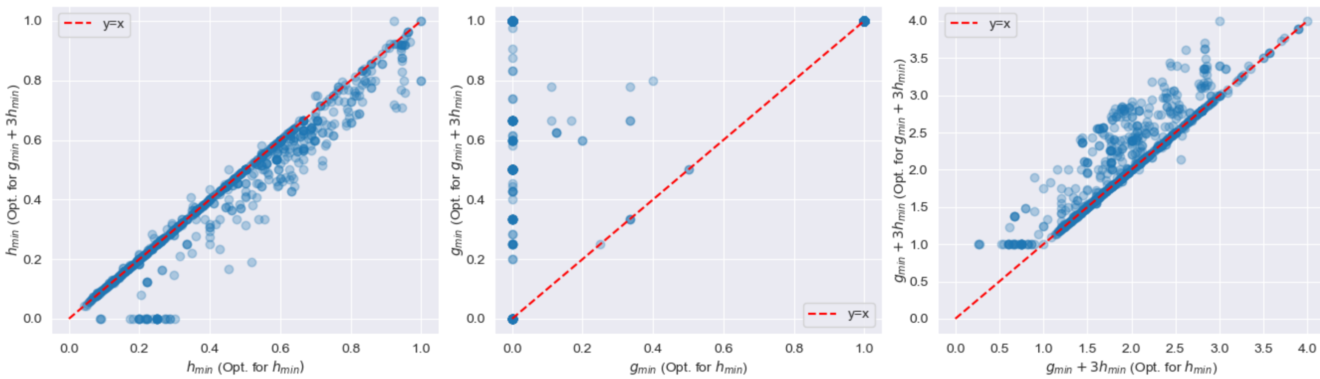

Figure 1: Plots demonstrating the beam search ($m = 128$) performance of optimizing for h_{min} vs. optimizing for $\alpha g_{min} + \beta h_{min}$ with $\frac{\alpha}{\beta} = \frac{1}{3}$.

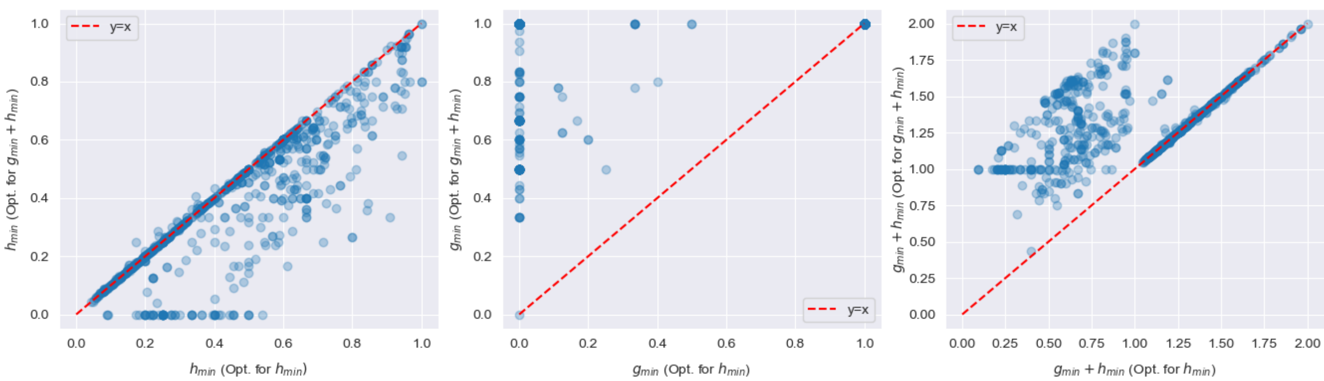

Figure 2: Plots demonstrating the beam search ($m = 128$) performance of optimizing for h_{min} vs. optimizing for $\alpha g_{min} + \beta h_{min}$ with $\frac{\alpha}{\beta} = 1$.

The search algorithms are implemented in Python, and we calculate the pairwise distances lazily to improve memory efficiency. The MIP models are implemented in Gurobi Optimizer v10.0.3 through its Python interface.[3] The MIP experiments are performed with a time and memory limit of 30 minutes and 4GB, respectively. As we employ random tie-breaking in our beam search, we run our experiments five times across different random seeds.

5.2 Results on the Bi-Criteria Framework

In Figure 1, we compare two settings: (1) optimizing solely for diversity using h_{min}; (2) bi-criteria optimization of h_{min} and g_{min} with a preference for diversity ($\alpha/\beta = 1/3$). In both cases, we use a beam search with a beam width of 128. Each point in the graphs represents the obtained solutions for one problem by directly optimizing for diversity (x-axis) and by optimizing for both diversity and quality (y-axis). We present results for one of the random seeds, however we observed similar trends across the different seeds.

[3]The MIP models are provided in the supplementary material (Zhong, Shati, and Cohen 2024).

In Figure 1 (left), we analyze the h_{min} value of the obtained solutions by the two approaches. Naturally, optimizing solely for h_{min} leads to better h_{min} values; however, the gain is only about 8%: a mean h_{min} of 0.435 compared to 0.403 with the bi-criteria framework. In fact, in only 294 out of the 1025 instances, we observed improved h_{min} when solely optimizing h_{min}. In Figure 1 (center), we compare the g_{min} values of the obtained solutions. As expected, the bi-criteria approach is able to obtain better g_{min} values. Out of the 1025 instances, 497 instances saw an improvement in g_{min} and 528 instances saw ties (of them, 501 had already obtained an optimal g_{min}, so there was no room for improvement). On average, the solutions from the bi-criteria framework have a g_{min} of 0.867, whereas optimizing for diversity leads to an average g_{min} of 0.601. Finally, in Figure 1 (right), the combination of marginally worse diversity and much higher quality resulted in a higher sum, indicating the effectiveness of the bi-criteria framework. In particular, optimizing for the bi-criteria framework leads to an 8.92% improvement on average for $g_{min}+3h_{min}$ (2.076 vs. 1.906).

Next, we perform a similar analysis with a different weighting scheme of quality and diversity. In Figure 2, we

Alg.	h_{min}			h_{sum}			$g_{min} + 3h_{min}$			$g_{sum} + 3h_{sum}$		
	Obj.	$\geq$ MIP	Time	Obj.	$\geq$ MIP	Time	Obj.	$\geq$ MIP	Time	Obj.	$\geq$ MIP	Time
G	0.953	584	0.13	0.997	440	0.13	0.963	576	0.55	0.988	380	0.58
B_2	0.956	592	0.25	0.999	473	0.24	0.971	632	1.64	0.990	437	0.70
B_4	0.963	621	0.43	1.001	527	0.42	0.974	657	2.03	0.991	490	0.88
B_8	0.967	660	0.79	1.003	566	0.76	0.979	682	2.85	0.992	509	1.25
B_{16}	0.972	680	1.54	1.004	626	1.47	0.983	714	4.57	0.993	563	2.00
B_{32}	0.975	691	3.09	1.005	659	2.97	0.985	736	8.20	0.994	591	3.58
B_{64}	0.980	729	6.47	1.006	688	6.23	0.989	768	16.38	0.994	616	7.06
B_{128}	0.982	741	14.41	1.007	709	13.92	0.991	795	36.57	0.995	631	15.27
MIP	1.000	1025	186.46	1.000	1025	395.16	1.000	1025	440.57	1.000	1025	591.26

Table 2: Results for the beam search approximation across different beam widths, objectives, and five random seeds. Objective values are averaged over five random seeds and normalized relative to MIP. The reported runtime is also averaged over five random seeds and measured in seconds.

compare the outcomes of directly optimizing diversity using h_{min} against the bi-criteria optimization of g_{min} and h_{min} ($\alpha/\beta = 1$). Figure 2 (left) shows, as expected, a greater decline in diversity under the bi-criteria framework with an average difference of approximately 15% (0.369 vs. 0.435). Figure 2 (center) shows the bi-criteria framework improved the g_{min} values of the solutions to a larger extent due to increasing the weight assigned to the quality function. Specifically, we observe an improvement of more than 55% (0.936 vs. 0.601) in average g_{min}. Finally, Figure 2 (right) shows the difference in the combined $g_{min} + h_{min}$ where the bi-criteria optimization resulted in an approximate 26% average increase in the combined metric (1.305 vs 1.036). Overall, we observe this weighting scheme ($\alpha/\beta = 1$) resulted in a set of less diverse plans with higher quality, on average.

We also analyzed the effectiveness of breaking ties in favor of lower-costed plans when optimizing solely for diversity using h_{min} with a beam width of 128. In this setting, we observe marginally better g_{sum} of 3.330 (random) compared to 3.346 (cost). For g_{min}, we observe a smaller improvement from 0.601 (random) to 0.603 (cost). However, the average h_{min} value actually degrades slightly from 0.435 (random) to 0.431 (cost). This indicates that breaking ties based on cost when optimizing solely for diversity has a limited impact compared to our bi-criteria framework that optimizes for both diversity and quality.

Finally, we also ran similar experiments for bi-criteria optimization of g_{sum} and h_{sum} with $(\alpha/\beta = 1/3, 1)$ We report these results in the supplementary material (Zhong, Shati, and Cohen 2024).

These experimental results demonstrate the flexibility of the bi-criteria framework; by including quality measures in the optimization and selecting the weighting scheme, users can explore the potential trade-offs between diversity and quality that lead to a better combination.

5.3 Results on Beam Search Approximation

In this section, we evaluate the performance of our beam search approximation and its ability to provide a flexible trade-off between solution quality and computation. To do so, we analyze the performance of beam search with beam widths ranging from 2 to 128 and compare the results to a greedy search, as well as to exact MIP. Table 2 shows the empirical performance of the different algorithms on the benchmark set, with G and B_i representing greedy search and beam search with beam width i, respectively. Specifically, we present mean relative performance compared to MIP (i.e., by dividing the average objective obtained for each instance by the objective obtained by MIP), the number of instances for which each approximation obtained solutions equal or better compared to MIP, and the mean runtime of each approach (excluding the time spent to construct the MIP). It should be noted that many MIPs (e.g., for h_{min}) ran out of memory (709 out of 1025), some ran out of time (36 out of 1025), and only a small portion of the instances was solved to optimality (280 out of 1025). We observe that beam search consistently improves its performance as the beam width grows and outperforms greedy search in all configurations starting with a beam width of 2. For example, when optimizing h_{min}, beam search with a width of 128 finds equal or better solutions compared to MIP in 741 out of the 1025 instances. In contrast, greedy search does so only 584 out of the 1025 instances, which is consistent with Katz, Sohrabi, and Udrea's (2022) reported results. In addition, the mean objective value for solutions obtained by beam search with a beam width of 128 was 98.2% of the mean objective value for MIP. Similar trends are observed for the additional three objectives we analyzed, h_{sum}, $g_{min}+3h_{min}$ and $g_{sum} + 3h_{sum}$. We note that these objectives that are based on our framework in Section 3 were not studied before in the context of diverse plan selection.

Table 2 also reports the average runtime for each approach. We note that, as expected, the average runtime of beam search grows approximately linearly with beam width. For example, beam search with a beam width of 128 required between 2.58%-8.30% of MIP runtime, while beam width of 32 required between 0.06%-1.86% of MIP runtime.

To better understand the gains of beam search for different beam widths, we focus on the 441 instances where the greedy search (i.e., beam search with a beam width of 1) fails

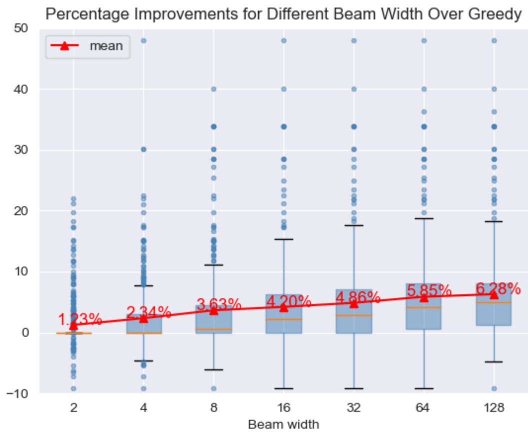

Figure 3: Relative improvement for different beam widths compared to greedy search.

to find equivalent solutions to MIP on the h_{min} objective. Figure 3 shows the empirical distribution, across problem instances, of relative solution objective value for beam search with different beam widths normalized by solution objective value from greedy search. We can see that, on average (the red line), beam search improves over greedy search, from a 1.23% improvement for a beam width of 2 up to a 6.28% improvement with a beam width of 128. In fact, for beam widths of 8 to 128, we observe improvements in objective values on more than 50% of the instances as indicated by the median (the orange lines). Moreover, for beam widths of 64 and 128, we observe improvements in more than 75% of the instances. For some instances, we observe improvements as large as 50%-80% (note that the y-axis is only presented in the range of -10% to 50% for clarity). However, we observe that for some problems beam search obtains worse solutions compared to greedy search, with up to 9.10% - 13.07% lower objective values. This is consistent with our examples, confirming that beam search is not *guaranteed* to outperform greedy search. We can also observe the gains in average objective value are diminishing as the beam widths are on an exponential scale. On average, beam search with a beam width of 128 gets around a 6.28% improvement over greedy search, whereas a beam width of 32 would have already given us a 4.86% improvement. This indicates a solid incentive to balance search effort (controlled by beam width) and performance.

The experimental results show that our beam search approximation provides a useful way of trading off additional computation for better solutions, while maintaining theoretical guarantees on the obtained solutions.

6 Discussions and Future Work

The results of the experiments demonstrate the flexibility provided by our bi-criteria framework in trading off diversity and quality in the selected subset of plans. They also demonstrate the flexibility provided by our beam search approximation in trading off additional computation for better solutions. We note that, to our knowledge, this is the first work to consider beam search for the criteria in Table 1 and provide approximation guarantees on the obtained solution. We expect that our beam search approximation can be adopted in additional tasks where bi-criteria optimization of diverse subset selection is used and has so far been optimized using greedy search.

Our work raises several interesting directions for future work. Further investigation can identify or propose additional quality and diversity functions that can be efficiently optimized by beam search, to extend our optimization framework. In addition, it remains an open question whether the theoretical results regarding our beam search approximation can be further improved by obtaining tighter bounds. Investigating techniques to efficiently parallelize beam search can allow us to utilize larger beam widths (thus obtaining better solutions) without incurring linear growth in runtime. In addition, beam search can used as an anytime algorithm (Zhang 1998; Cohen and Beck 2021) by iteratively restarting the search with a higher beam width, providing improved solutions over time. Finally, investigating exact optimization approaches, including novel MIP, Constraint Programming, or MaxSAT formulations, for the proposed optimization framework is an interesting direction for future work.

7 Conclusion

In this work, we focus on the problem of diverse plan selection, as part of the two-stage paradigm for diverse planning that underlies the recent state-of-the-art approaches. We introduce a bi-criteria optimization framework, that takes into account both diversity and quality, providing the user with a higher degree of flexibility and generalizing previous work that focused solely on diversity. In addition, we present a beam search approximation that allows the user to trade off additional computation for solution quality and present new approximation guarantees on the obtained solutions. Our experiments establish the main benefits of our framework: our flexible bi-criteria framework enables us to obtain better quality solutions with little to no loss on diversity, and our beam search approximation enables us to obtain significantly better solutions compared to greedy search by using a higher beam width.

Acknowledgments

We thank the anonymous reviewers whose valuable feedback helped improve the final paper. The authors gratefully acknowledge funding from the Natural Sciences and Engineering Research Council of Canada.

References

Bäckström, C.; and Nebel, B. 1995. Complexity results for SAS+ planning. *Computational Intelligence*, 11(4): 625–655.

Barrios, E. Q.; Olaya, Á. G.; Millán, D. B.; and Rebollo, F. F. 2011. Control of autonomous mobile robots with automated planning. *Journal of Physical Agents*, 5(1): 3–13.

Bloem, M. J. 2015. *Optimization and analytics for air traffic management*. Stanford University.

Boddy, M. S.; Gohde, J.; Haigh, T.; and Harp, S. A. 2005. Course of Action Generation for Cyber Security Using Classical Planning. In *ICAPS*, 12–21.

Borodin, A.; Lee, H. C.; and Ye, Y. 2012. Max-sum diversification, monotone submodular functions and dynamic updates. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGAI symposium on Principles of Database Systems*, 155–166.

Buchbinder, N.; and Feldman, M. 2018. Submodular functions maximization problems. In *Handbook of approximation algorithms and metaheuristics*, 753–788. Chapman and Hall/CRC.

Cohen, E.; and Beck, C. 2019. Empirical analysis of beam search performance degradation in neural sequence models. In *International Conference on Machine Learning*, 1290–1299. PMLR.

Cohen, E.; and Beck, J. C. 2021. Heavy-Tails and Randomized Restarting Beam Search in Goal-Oriented Neural Sequence Decoding. In *Integration of Constraint Programming, Artificial Intelligence, and Operations Research: 18th International Conference, CPAIOR 2021, Vienna, Austria, July 5–8, 2021, Proceedings 18*, 115–132. Springer.

Coman, A.; and Munoz-Avila, H. 2011. Generating diverse plans using quantitative and qualitative plan distance metrics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 946–951.

Dasgupta, A.; Kumar, R.; and Ravi, S. 2013. Summarization Through Submodularity and Dispersion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1014–1022.

Fox, M.; Gerevini, A.; Long, D.; and Serina, I. 2006. Plan stability: replanning versus plan repair. In *Proceedings of the Sixteenth International Conference on International Conference on Automated Planning and Scheduling*, 212–221.

Fuentetaja, R.; García-Olaya, A.; García, J.; González, J. C.; and Fernández, F. 2020. An Automated Planning Model for HRI: Use Cases on Social Assistive Robotics. *Sensors*, 20(22): 6520.

Hebrard, E.; Hnich, B.; O'Sullivan, B.; and Walsh, T. 2005. Finding diverse and similar solutions in constraint programming. In *AAAI*, 372–377.

Karpas, E.; and Magazzeni, D. 2020. Automated planning for robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 3: 417–439.

Katz, M.; and Sohrabi, S. 2020. Reshaping diverse planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9892–9899.

Katz, M.; Sohrabi, S.; and Udrea, O. 2022. Bounding Quality in Diverse Planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9805–9812.

Kosub, S. 2019. A note on the triangle inequality for the Jaccard distance. *Pattern Recognition Letters*, 120: 36–38.

Krause, A.; and Golovin, D. 2014. Submodular function maximization. *Tractability*, 3(71-104): 3.

Lindsay, A.; Ramirez-Duque, A.; Petrick, R.; and Foster, M. E. 2022. A Socially Assistive Robot using Automated Planning in a Paediatric Clinical Setting. *arXiv preprint arXiv:2210.09753*.

Markov, Z.; and Larose, D. T. 2007. *Data mining the Web: uncovering patterns in Web content, structure, and usage*. John Wiley & Sons.

Meister, C.; Vieira, T.; and Cotterell, R. 2020. If beam search is the answer, what was the question? *CoRR*, abs/2010.02650.

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.

Pedersen, M. R.; and Krüger, V. 2015. Automated planning of industrial logistics on a skill-equipped robot. In *IROS 2015 workshop Task Planning for Intelligent Robots in Service and Manufacturing*. Citeseer.

Riabov, A.; Sohrabi, S.; Sow, D.; Turaga, D.; Udrea, O.; and Vu, L. 2015. Planning-based reasoning for automated large-scale data analysis. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 25, 282–290.

Roberts, M.; Howe, A.; and Ray, I. 2014. Evaluating diversity in classical planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 24, 253–261.

Sohrabi, S.; Riabov, A.; Katz, M.; and Udrea, O. 2018. An AI planning solution to scenario generation for enterprise risk management. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Sousa, A.; and Tavares, J. 2013. Toward automated planning algorithms applied to production and logistics. *IFAC Proceedings Volumes*, 46(24): 165–170.

Vadlamudi, S. G.; and Kambhampati, S. 2016. A combinatorial search perspective on diverse solution generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

von Tschammer, J.; Mattmüller, R.; and Speck, D. 2022. Loopless top-k planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 32, 380–384.

Wu, W.-L.; Zhang, Z.; and Du, D.-Z. 2019. Set function optimization. *Journal of the Operations Research Society of China*, 7(2): 183–193.

Yang, Y.; Huang, L.; and Ma, M. 2018. Breaking the Beam Search Curse: A Study of (Re-)Scoring Methods and Stopping Criteria for Neural Machine Translation. *CoRR*, abs/1808.09582.

Zhang, W. 1998. Complete anytime beam search. In *AAAI/IAAI*, 425–430.

Zhong, S.; Shati, P.; and Cohen, E. 2024. Appendices to "Bi-Criteria Diverse Plan Selection via Beam Search Approximation", published at SoCS2024.