

Multi-Objective Search: Algorithms, Applications, and Emerging Directions

Oren Salzman¹, Carlos Hernández Ulloa², Ariel Felner³, Sven Koenig⁴

¹Technion - Israel Institute of Technology

²Universidad San Sebastian

³Ben-Gurion University

⁴University of California, Irvine

osalzman@cs.technion.ac.il, carlos.hernandez@uss.cl, felner@bgu.ac.il, sven.koenig@uci.edu

Abstract

Multi-objective search (MOS) has emerged as a unifying framework for planning and decision-making problems where multiple, often conflicting, criteria must be balanced. While the problem has been studied for decades, recent years have seen renewed interest in the topic across AI applications such as robotics, transportation, and operations research, reflecting the reality that real-world systems rarely optimize a single measure. This paper surveys developments in MOS while highlighting cross-disciplinary opportunities, and outlines open challenges that define the emerging frontier of MOS research.

1 Introduction

Multi-objective Search (MOS) problems are pervasive in real-world settings where decision makers must balance several, often conflicting, objectives. For example, in route finding applications, we are interested in simultaneously minimizing both travel time and fuel consumption, or distance and toll costs. In many such cases, improvements in one objective cannot be achieved without hinderring another objective, making the search for well-balanced solutions both challenging and essential.

When a decision maker can articulate how much loss in one objective is acceptable for a given gain in another, all objectives can be turned into one scalar value by, e.g., optimizing a weighted sum or another order-preserving (monotone) aggregation. Then, the resulting problem can be solved by any standard single-objective algorithm. This aggregation approach, however, presupposes reliable *a priori* information about acceptable trade-off for the decision maker, which is often not available to the algorithm.

An alternative approach to addressing the multidimensional trade-off is to use MOS algorithms that compute the best attainable trade-offs wherein no objective can be improved without degrading at least one other objective. This set can then be presented to the decision maker for an a posteriori preference articulation and final choice.

While being a decades-old problem (Vincke 1976; Hansen 1980; Clímaco and Pascoal 2012; Current and Marsh 1993; Skriver 2000; Tarapata 2007; Ulungu and

Teghem 1991), in recent years, the study of MOS has attracted growing attention across multiple research communities. Dedicated workshops and tutorials addressing complex, often conflicting, objectives have been featured in mainstream AI venues (e.g., AAAI [2024], IJCAI [2023; 2025], AAMAS [2024], ICAPS [2024], ECAI [2025] and SoCS [2023]), and in robotics and machine-learning venues (e.g., RSS [2025] and NeurIPS [2024]). Related developments are also emerging in operations research (OR), transportation science, and evolutionary computation, where multi-objective optimization has a long tradition but is now being revisited with modern heuristic search, reinforcement learning, and hybrid approaches. This convergence of interests reflects a shared recognition that real-world decision-making rarely optimizes a single criterion, and that principled multi-objective reasoning is essential for building intelligent, robust, and adaptable systems.

Scope. In this paper, we highlight recent advances in the field in terms of problem variants, algorithms, applications and emerging directions. It is by no means a comprehensive literature review but an attempt to provide an accessible starting point for any researcher interested in the field.

Here, we focus on the setting of multi-objective *search*. However, we also highlight extensions and variants such as those that include uncertainty. Importantly, due to lack of space, we maintain a high-level description of approaches and refer the reader to (Salzman et al. 2023) for a technical overview of recent MOS advances.

2 Problem Setting & Variants

2.1 Notation

Boldface font indicates vectors, lower-case and upper-case symbols indicate elements and sets, respectively. The notation p_i will be used to denote the i 'th component of \mathbf{p} . The addition of two d -dimensional vectors \mathbf{p} and \mathbf{q} and the multiplication of a real-valued scalar k and a d -dimensional vector \mathbf{p} are defined as $\mathbf{p} + \mathbf{q} = (p_1 + q_1, \dots, p_d + q_d)$ and $k\mathbf{p} = (kp_1, \dots, kp_d)$, respectively.

Let \mathbf{p} and \mathbf{q} be d -dimensional vectors. For a minimization problem, we say that \mathbf{p} *dominates* \mathbf{q} and denote this as $\mathbf{p} \preceq \mathbf{q}$ if $\forall i, p_i \leq q_i$. We say that \mathbf{p} is *lexicographically smaller* than \mathbf{q} and denote this as $\mathbf{p} \prec_{\text{lex}} \mathbf{q}$ if $p_k < q_k$ for the first index k s.t. $p_k \neq q_k$. Finally, let \mathbf{p} and \mathbf{q} be two

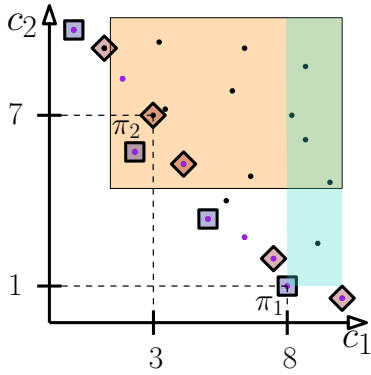


Figure 1: Visualization of key MOS concepts for the special case of a bi-objective problem. Solutions on and not on the PF are visualized as purple and black dots, respectively. Visualization of all solutions dominated and approximately dominated by solutions π_1 and π_2 are visualized by turquoise and orange regions respectively. Example for sets of solutions that approximate the PF which lie and which do not lie on the PF are depicted with purple squares and red diamonds, respectively. Finally, the dominance factor $\text{DF}(\pi_1, \pi_2)$ in this example is $\max(\max(8/3 - 1, 0), \max(1/7 - 1, 0)) = 5/3$.

d -dimensional vectors and let ε be another d -dimensional vector such that $\forall i \varepsilon_i \geq 0$. We say that \mathbf{p} *approximately dominates* \mathbf{q} with an *approximation factor* ε and denote this as $\mathbf{p} \preceq_{\varepsilon} \mathbf{q}$ if $\forall i, p_i \leq (1 + \varepsilon_i) \cdot q_i$.

2.2 MOS—Problem definition and variants

In most variants of a MOS problem a directed graph $G = (V, E)$ where each edge $e \in E$ has a nonnegative cost vector $\mathbf{c}(e) \in \mathbb{R}^d$ where $d > 0$ is the number of objectives. For the specific cases where $d = 1$ and $d = 2$, we refer to the problem as *single-objective* and *bi-objective*, respectively. For a path $\pi = \langle v_1, \dots, v_k \rangle$ where $(v_i, v_{i+1}) \in E$, its cost $\mathbf{c}(\pi)$ is the sum of the edge costs. That is, $\mathbf{c}(\pi) = \sum_i \mathbf{c}(v_i, v_{i+1})$. Given start and target vertices $s, t \in V$, a path from s to t is called a *solution*. A solution is *Pareto-optimal* iff its cost is not dominated by any other solution. See Fig. 1.

Exact MOS. In the basic MOS problem, we are given vertices $s, t \in V$ and the goal is to compute the set Π^* of Pareto-optimal solutions, also known as the *Pareto front* (PF) (Salzman et al. 2023). Importantly, computing Π^* is NP-hard (Serafini 1987) as its cardinality may be exponential in $|V|$ (Ehrgott 2005; Breugem, Dollevoet, and van den Heuvel 2017). Even determining whether a path belongs to Π^* is NP-hard (Papadimitriou and Yannakakis 2000).

In certain settings, we would like to compute the PF from a source $s \in V$ to *every* other vertex $v \in V$ (see, e.g., (Martins 1984; de las Casas, Sedeño-Noda, and Borndörfer 2021; Kurbanov, Cuchý, and Vokřínek 2023)) or from *any* vertex $u \in V$ to *every* other vertex $v \in V$ (see, e.g., (Zhang et al. 2023b; Cuchý, Vokřínek, and Jakob 2024)).

Approximate MOS. In real-world settings, we are often not interested in the entire PF as it may be too large to

present to decision makers (for example, there may be thousands of solutions in the PF of large road networks (Ren et al. 2025)). Thus, we are often interested in computing a bounded approximation of Π^* . Here, we are given an approximation factor ε . The ε -*approximate PF* Π_{ε}^* is a set of solutions such that $\forall \pi \in \Pi^*, \exists \pi' \in \Pi_{\varepsilon}^* \text{ s.t. } \mathbf{c}(\pi') \preceq_{\varepsilon} \mathbf{c}(\pi)$. Namely, every solution in Π^* is approximately dominated by some solution in Π_{ε}^* . Importantly, (i) the ε -approximate Pareto-optimal solution set is not necessarily unique and (ii) some variants of this definition require that $\Pi_{\varepsilon}^* \subseteq \Pi^*$ while others don't (See Fig. 1). Alternatively, some problem formulations seek a small representative set of solutions in Π^* (Rivera, Baier, and Hernández 2022) (without any formal definition of “small”).

Anytime MOS. Many applications benefit from obtaining a subset $\tilde{\Pi}^*$ of Π^* as fast as possible. As more time is available to the algorithm, additional solutions from $\Pi^* \setminus \tilde{\Pi}^*$ are added to $\tilde{\Pi}^*$. The algorithm terminates when either (i) the decision maker or the algorithm that uses the solutions terminates the algorithm or (ii) the entire PF has been returned (i.e., $\tilde{\Pi}^* = \Pi^*$). Formally, we define the *dominance factor* of a solution π over another solution π' as

$$\text{DF}(\pi, \pi') = \max_{i=1,2,\dots,N} \left(\max \left\{ \frac{c_i(\pi)}{c_i(\pi')} - 1 \right\}, 0 \right),$$

which measures how “good” π approximates π' . $\text{DF}(\pi, \pi')$ encodes the smallest ε -value that satisfies $\pi \preceq_{(\varepsilon, \dots, \varepsilon)} \pi'$ (See Fig. 1). For a set of solutions Π , we define the *approximation error* of Π as

$$\text{ERR}(\Pi) = \max_{\pi' \in \Pi^*} \left\{ \min_{\pi \in \Pi} \text{DF}(\pi, \pi') \right\},$$

which, roughly speaking, measure the solution in Π^* that is “least” approximated by any solution in Π .

Now, to measure the performance of an anytime MOS algorithm, we typically wish to minimize the *Area Under the Curve* (AUC) of the approximation error formally defined as $\text{AUC} = \int_0^{t^{\text{limit}}} \text{ERR}(\Pi(t)) dt$, where t^{limit} is the runtime limit and $(\Pi(t))$ is solution set returned at time t .

Incremental & Dynamic MOS. When the MOS problem is applied in an online fashion (i.e., planning is interleaved with taking actions) and the query is updated (either because the target is updated or because the environment’s representation is updated), one may want to avoid calling an algorithm from scratch and instead reuse previous search efforts. Incremental multi-objective graph search algorithms (see, e.g., (Ren et al. 2022b)) reuse previous searches to speed up subsequent exact or approximate MOS searches.

Similarly, in MOS applications such as flight planning, dynamic traffic roadmaps, and telecommunication and data networks, the underlying graph changes over time since either its structure (edges, nodes) or the cost functions (weights, travel times, risks, etc.) evolve. In contrast to the incremental setting, here we are given the dynamics before planning begins and need to account for the temporal changes. This was only recently formulated by de las Casas et al. (2021). For additional details, see also the recent work by Shovan, Khanda, and Das (2025).

2.3 Beyond MOS

While MOS assumes a deterministic model, many real problems demand richer models. To capture stochasticity, or general optimization beyond paths, several extensions have been studied. Each extends MOS along a different axis while keeping Pareto optimality central. We briefly describe each model, highlighting the similarities and differences compared to MOS.

Handling uncertainty. Recall that a MOS problem is defined using a graph $G = (V, E)$ together with edge costs \mathbf{c} which present a deterministic model. A *Multi-objective Stochastic Shortest Path* (MOSSP) problem extends the MOS framework by introducing probabilistic transitions between states (Rojiers and Whiteson 2017). Formally, we are given a graph $G = (V, E)$ together with edge costs \mathbf{c} and a transition probability distribution over successor vertices. A policy μ maps vertices to successor edges, inducing a distribution over paths from the start s to the target t . The cost of a policy is defined as the expected cumulative cost vector across objectives. As in MOS, policies are compared using dominance: a policy μ_1 dominates μ_2 if it has no worse expected cost in every objective and is strictly better in at least one. The goal is to compute a coverage PF of policies.

A Multi-objective Markov Decision Process (MOMDP) generalizes MOSSP by adopting an MDP formalism. In contrast to MOSSP which focuses on reaching a target in a stochastic graph with vector costs, MOMDP allow arbitrary horizon settings (e.g., finite, infinite, discounted) and sequential decision-making under uncertainty, not just reaching a target vertex.

Learning. *Multi-objective Reinforcement Learning* (MORL) extends the MOS framework to settings where the agent interacts with an environment through repeated trial-and-error learning rather than having an explicit model of the state-transition dynamics (Hayes et al. 2022; Felten, Talbi, and Danoy 2024). Formally, MORL is defined over the same structure as an MOMDP, $M = (S, A, P, \mathbf{c})$, with state space S , action space A , transition function P , and vector-valued cost or reward function $\mathbf{c}(s, a) \in \mathbb{R}^d$. However, unlike MOMDPs, in MORL the transition probabilities and reward distributions are not assumed to be known a priori. Instead, the agent learns a policy $\mu : S \rightarrow A$ through experience, typically by interacting with the environment and receiving vector-valued feedback.

MORL generalizes MOS in that it seeks Pareto-optimal solutions across multiple objectives, but unlike MOS, it does not assume a static graph with deterministic edges. Relative to MOSSP and MOMDP, MORL replaces planning with learning: instead of computing Pareto-optimal policies from a known model of uncertainty, the agent must discover them through exploration and approximation. Thus, MORL inherits the challenges of both reinforcement learning (e.g., exploration-exploitation trade-offs, function approximation) and MOS (e.g., dominance checks). The goal in MORL remains to approximate the Pareto set of policies, but learning algorithms must balance sample efficiency, preference sensitivity, and scalability in high-dimensional state and objective spaces.

Multi-objective Optimization. In this paper, we focus on MOS which can be seen as an instance of the more general *multi-objective optimization* (MOO) problem, (see, e.g., (Branke et al. 2008; Miettinen 2012; Rojiers and Whiteson 2017; Hwang and Masud 2012; Emmerich and Deutz 2018)). It is important to highlight the similarities and differences between the two fields.

MOO is the most general formulation of problems in which several, possibly conflicting, objectives must be optimized simultaneously. Formally, given a feasible set \mathcal{X} and objective functions $f_i : \mathcal{X} \rightarrow \mathbb{R}$ for $i = 1, \dots, d$, the problem is to find the set of non-dominated solutions

$$\mathcal{X}^* = \{x \in \mathcal{X} \mid \nexists y \in \mathcal{X}, \mathbf{f}(y) \preceq \mathbf{f}(x), f(y) \neq f(x)\},$$

where $\mathbf{f}(x) = (f_1(x), \dots, f_d(x))$. Here, \mathcal{X}^* corresponds to the *Pareto set* whose image in \mathbb{R}^d is the PF.

Relative to MOS, MOO generalizes the underlying domain: whereas MOS is defined over paths in a deterministic graph with additive vector costs, MOO is agnostic to the structure of the feasible set and can capture continuous, combinatorial, or black-box domains. Compared to stochastic settings such as MOSSP and MOMDP, MOO does not necessarily assume probabilistic dynamics or sequential decision processes; instead, it focuses purely on the optimization of static or offline-defined objectives. In contrast to MORL, MOO assumes direct access to the objective functions rather than learning them through interaction. In this sense, MOO serves as the broad umbrella under which MOS, MOSSP, MOMDP, and MORL can be seen as structured subclasses with additional constraints on the representation of \mathcal{X} , the dynamics of decision-making, and the information available to the algorithm.

From an algorithmic point of view, in contrast to MOS, which builds upon search algorithms, MOO typically builds upon local and global optimization methods such as genetic algorithms (Deb et al. 2002; Deb and Jain 2013; Zhang and Li 2007), particle swarm optimization (Coello and Lechuga 2002), and simulated annealing (Li and Landa-Silva 2011).

3 Algorithmic Advances

In this section we outline recent algorithmic advances in MOS. We start with a brief historical overview and continue to outline tools, techniques and algorithms that advances the state-of-the-art in MOS. We conclude with a brief description of advances in generalizations of MOS (Sec. 2.3) that have close ties to MOS algorithms

3.1 Brief historical overview

Early work on MOS established the algorithmic framework which is the basis of most modern algorithms (Hansen 1980; Martins 1984; Warburton 1987). For efficiently computing Π^* , two notable approaches emerged. The first generalizes the label-correcting paradigm to the multi-objective setting (Guerriero and Musmanno 2001). Label-correcting is an iterative shortest-path method that repeatedly updates tentative distance labels of vertices whenever a shorter path is found, allowing multiple updates per vertex until no further improvements are possible. The second generalizes the

celebrated A* algorithm (Hart, Nilsson, and Raphael 1968) which we detail next as most of the recent advancements fall under this category.

A notable contribution was the work by Stewart et al. (1991), who introduced Multi-Objective A* (MOA*). MOA* served as the foundation to multiple extensions (see, e.g., (Madow and De La Cruz 2005, 2010)) which differ in which information is contained in the nodes, how nodes are ordered in the priority queue and how dominance checks are implemented and when they are performed (upon generation or upon expansion). A key insight that dramatically improved the efficiency of these algorithms was to order the nodes in the priority queue in increasing lexicographic order and apply the notion of *dimensionality reduction* (Pulido, Madow, and Pérez-de-la Cruz 2015). See (Salzman et al. 2023) for an overview of the approach. The resulting algorithm based on this idea was termed NAMOA-dr¹.

Early approaches (Warburton 1987; Perny and Spanjaard 2008; Tsaggouris and Zaroliagis 2009; Breugem, Dollevoet, and van den Heuvel 2017) to approximating Π^* focused on Fully Polynomial Time Approximation Schemes² (FPTAS) (Vazirani 2001). Unfortunately, running these approaches on moderately-sized graphs (i.e., with roughly 10,000 vertices) is often impractical (Breugem, Dollevoet, and van den Heuvel 2017).

3.2 Algorithmic advances in MOS

Exact approaches. In recent years, several algorithms dramatically improved the efficiency of exact MOS algorithms (see, e.g., (de las Casas et al. 2023; de las Casas, Sedeño-Noda, and Borndörfer 2021; Ahmadi et al. 2021; Ren et al. 2025)). Notable examples that reduce the computational complexity of key operations include (i) BOA* (Hernández et al. 2023) which adapted and simplified NAMOA-dr for the bi-objective setting performing dominance checks in $O(1)$, and (ii) recent work (Zhang et al. 2024b; Ren et al. 2022a) which improves node indexing and data structures for dominance checks of two and three objectives to yield dramatic speedups. Finally, recent work (Ahmadi et al. 2024) considered the more general set of graphs with negative edges.

Approximate approaches. Goldin and Salzman (2021) suggested PPA*, an extension of BOA* that introduced new pruning techniques to efficiently compute an ε -approximate PF Π_ε^* for the bi-objective setting. PPA* was later generalized by Zhang et al. (2022) who suggested the A*pex algorithm which allows to compute Π_ε^* for any number of objectives. The efficiency of A*pex stems from the observation that paths whose cost is very similar can be grouped in an efficient manner allowing to dramatically prune the PF. A*pex was later used to exploit correlation of edge costs (Halle et al. 2025), develop anytime MOS algorithms (Zhang et al. 2024c) and more (Zhang et al. 2024a), see also Sec. 4.

¹Here, ‘dr’ stands for dimensionality reduction.

²An FPTAS is an approximation scheme whose time complexity is polynomial in the input size and also polynomial in $1/\varepsilon$ where ε is the approximation factor.

Parallelization. While there has been some research on parallelizing MOS algorithms (see, e.g., (Sanders and Madow 2013; Erb, Kobitzsch, and Sanders 2014; Medrano and Church 2015)), this research direction has been largely unexplored (Salzman et al. 2023). Two notable exceptions include (i) The work by Ahmadi et al. (2025) who explored permutations of objective orderings in parallel while sharing bounds to collapse subproblems, achieving near-linear speedups on many-core systems and (ii) the work by Hernández et al. (2024) who suggest an approach to compute set dominance checks or SDC (a key procedure, which dominates the running time of many state-of-the-art MOS algorithms) in parallel. They exploit vectorized the operations offered by “Single Instruction/Multiple Data” (SIMD) instructions to perform SDC on ubiquitous consumer CPUs thereby dramatically improving the runtime of existing MOS algorithms.

Theory. There have not been many recent theoretical advances. de las Casas et al. (2021) suggested an FPTAS for the new setting of dynamic MOS in which edges cost can change online. Skyler et al. (2024) extended theory from single-objective search (SOS) to MOS which characterizes the set of vertices and search nodes that any unidirectional search algorithm must expand to prove the optimality of the solution. Specifically, they introduce a classification of vertices into *must-expand*, *maybe-expand*, and *never-expand* categories. The notable difference between SOS and MOS is that vertices must be expanded to (i) prove that any path in a PF is Pareto-optimal (these are called optimality vertices) and (ii) ensure that there are no more solution costs that are not represented in the PF (these are called completeness vertices). Completeness vertices have no analogy in SOS.

Heuristics. Key to the success of heuristic search in general, and heuristic MOS in particular, is the ability to incorporate domain knowledge using heuristics that guide the algorithm. Almost all MOS algorithms use the “ideal point heuristic” $\mathbf{h}_{\text{ideal}}$, which combines a set of d single-objective heuristics h_1, \dots, h_d . Here, $h_i : V \rightarrow \mathbb{R}_{\geq 0}$ corresponds to the shortest path from each vertex according to the i ’th objective and $\forall v \in V \mathbf{h}_{\text{ideal}}(v) := (h_1(v), \dots, h_d(v))$.

However, in contrast to SOS, in the general case of MOS, the heuristic value of a vertex v is not a single cost vector, but a *set of cost vectors* (see (Madow and De La Cruz 2010) for original definition and (Salzman et al. 2023) for an in-depth discussion). While such heuristics, called Multi-Value Heuristics (MVH) are much more informative, the overhead of computing and using MVHs in MOS algorithms can be large and the total runtime is often larger than when using $\mathbf{h}_{\text{ideal}}$ (Geißer et al. 2022).

A notable example where MVHs are used is the recent work by Zhang et al. (2023a), which generalize Differential Heuristics (DHs) (Goldberg and Harrelson 2005), a class of memory-based heuristics for SOS, to bi-objective search, resulting in Bi-Objective Differential Heuristics (BO-DHs). They propose several techniques to reduce the memory usage and computational overhead of BO-DHs, demonstrating reductions in runtime of a bi-objective search algorithm by up to an order of magnitude.

3.3 Algorithmic advances in MOS extensions

While there has been many advances in MOSSP, MOMDP and MOO algorithms, which are not the focus of this paper, here we mention work that is closely related to MOS. Recent work by Chen, Trevizan, and Thiébaux (2023) suggests adapting heuristic-search algorithms (which are the foundation of MOS algorithms) for MOSSP. This is done by extending (single-objective) stochastic shortest-path algorithms, such as LAO* (Hansen and Zilberstein 2001) and LRTDP (Bonet and Geffner 2001), to the multi-objective setting. They also study how to guide their algorithms with domain-independent heuristics to account for the probabilistic and multi-objective features of the problem.

4 MOS as an Algorithmic Toolbox

Recently the algorithmic toolbox developed for MOS has also proven useful in other domains. Some, are new variants of MOS while others are seemingly unrelated optimization problems where MOS approaches have been useful.

Multi-objective minimum spanning tree. The Multi-Objective Minimum Spanning Tree (MO-MST) problem generalizes the classical MST problem to settings where edges are labeled with cost vectors. Instead of a single spanning tree with minimal total weight, the goal is to identify a Pareto set of spanning trees that represent undominated trade-offs among objectives. However, unlike MST, for which there are polynomial time algorithms that solve it, MO-MST is NP-hard (Fernandes et al. 2020). MO-MST is important for communication networks, where spanning trees must balance latency, bandwidth and resilience, and in transport and logistics, where constructing infrastructure with multiple cost criteria is essential (see, e.g., (Levin and Nuriakhmetov 2011)). MO-MST algorithms borrow heavily from MOS techniques (see, e.g., (Sourd and Spanjaard 2008; Fernandes et al. 2020; de las Casas, Sedeño-Noda, and Borndörfer 2025)).

MOS with objective aggregation. In many real-world problems with multiple objectives, the objectives interact in a complex manner, leading to problem formulations that do not allow out-of-the-box usage of MOS algorithms (Fu et al. 2023; Slutsky et al. 2021; Axelrod, Kaelbling, and Lozano-Pérez 2018). Roughly speaking, this is because the search algorithms needs to treat differently paths that are and that are not solutions. For example, in robot inspection planning (Fu et al. 2023; Alpert et al. 2025), a robot is required to view as many points of interest (POI) as possible using an on-board sensor while minimizing path length. The two objectives which define a solution π are the number of POIs viewed along π and the length of π . However, every path that is not a solution must keep track of which POI was viewed, essentially defining a binary objective for each POI. This is because two paths to the same vertex that viewed different POIs cannot dominate one another as their final bi-objective cost depends on which POIs will be viewed in the future.

This creates a mismatch between objectives at intermediate nodes, which we term *hidden objectives*, and objectives at solution nodes, which we term *solution objectives*. The

relation between solution objectives and hidden objectives is captured via some method of *objective aggregation* (Peer et al. 2025). Returning to our inspection-planning example, there is one hidden objective that corresponds to each POI as well as one for path length and there are two solution objectives corresponding to number of POIs viewed and path length. Here objective aggregation is done by adding all (binary) cost values of POI hidden objectives. We call such problems *MOS with objective aggregation* (MOS-OA). Importantly, MOS-OA algorithms can naturally employ the MOS algorithmic toolbox. Indeed, early versions of A*_{pex} were developed in the context of MOS-OA (Fu et al. 2023).

Multi-objective Multi-Agent Path Finding. The Multi-Agent Path Finding (MAPF) problem (Stern et al. 2019) involves finding non-colliding paths for multiple agents from their start locations to their respective target locations in a shared environment. The primary goal is to optimize a metric such as the sum of travel time of all agents or the makespan (i.e., task completion time). The Multi-Objective MAPF (MO-MAPF) problem extends the MAPF problem to multiple, often conflicting, optimization criteria such as makespan, energy consumption, safety margin, or fairness among agents. The result is not a single plan but a PF of MAPF plans, each representing a different trade-off. Recent algorithms (see, e.g., (Ren, Rathinam, and Choset 2021b,a, 2023; Wang et al. 2024)) integrate MAPF and MOS to obtain scalable algorithms for this purpose.

Constrained shortest path. In the Constrained Shortest-Path problem (CSP) (Storandt 2012) we are interested in computing a shortest path subject to some constraints (e.g., limited energy consumption for an autonomous agent). This setting was generalized by Skyler et al. (2022) who consider the setting where we need to find a solution which belongs to Π^* whose costs are below given upper bounds on each objective. Later Zhang et al. (2024a) considered a similar setting but where we need to find a solution which belongs to Π_ϵ^* for some $\epsilon > 0$.

k -Shortest simple path. In the k -Shortest Simple Path (k -SSP) problem, we are given a graph $G = (V, E)$ with regular (scalar) edge costs. Given start and target vertices $s, t \in V$ and a parameter k , we are tasked to compute the k shortest paths between s and t . While this is a single-objective problem, recently de las Casas et al. (2025) have shown that the 2-SSP can be solved by a reduction to a bi-objective search problem.

5 Emerging Applications

We briefly review several diverse domains where MOS and its variants have been recently used. This showcases the applicability of MOS despite its relative simplicity when compared to the richer models reviewed in Sec. 2.3.

Automated design & synthesis. MOS has been applied to design problems in chemistry, biology, and engineering. One example is retrosynthesis planning in computational chemistry, which is the problem of finding reaction sequences that produce a target molecule. Lai et al. (2025) consider multiple objectives and, by searching for un-dominated synthesis

routes, they were able to present several candidate pathways to a human chemist to evaluate and choose from. Similarly, MOS found applications in drug discovery and generative design. For example, Southiratn et al. (2025) suggested a bi-objective search algorithm for generating molecular structures that balance affinity to two proteins while also satisfying drug-like property constraints. The result is a set of novel molecular candidates with high predicted efficacy and acceptable pharmacological profiles, which traditional single-objective or scalarized approaches would have likely missed.

Multi-modal journey planning. Multi-modal journey planning determines routes combining different transport modes (Bast et al. 2016), which inherently involves multi-objective optimization such as time, cost and comfort. These methods (see e.g., (Potthoff and Sauer 2022b,a)) build upon MOS algorithms to make queries tractable at metropolitan scale. Many real-world uses of such algorithms have recently been documented. For example, OpenTripPlanner 2 is an open-source multi-modal journey planner for public transportation in combination with bicycling, walking, and mobility services such as bike share and ride hailing. It has been deployed nationwide in Norway and Finland. In Portland (Oregon), it provides about 40,000 trip plans on a typical weekday (OTP 2025).

Robotics. In robotics, multiple objectives often need to be simultaneously balanced (e.g., cost, energy and safety) In Sec. 4 we discussed robot inspection planning in the context of MOS-OA. Another example is autonomous vehicle (AV) planning using rulebooks (Slutsky et al. 2021; Censi et al. 2019; Halder and Althoff 2025; Penlington, Zanardi, and Frazzoli 2024), where the system must generate a trajectory that complies with a set of potentially conflicting traffic rules. Consider, for instance, Singapore’s Final Theory of Driving that requires (i) maintaining at least a one-meter gap when passing a parked vehicle and (ii) prohibits crossing a solid double white lane divider. When an AV encounters a car improperly parked along such a divider, it may be impossible to satisfy both requirements simultaneously. Fortunately, requirements often form in a hierarchy—e.g., avoiding a collision is more important than keeping safety margin from parked vehicles and than maintaining lane. Rulebooks are a systematic way to address such settings. Here, a rule corresponds to an objective and a rulebook defines a hierarchy that induces a partial order. For example rule r_1 (avoiding collision) is more important than rules r_2, r_3 (maintaining safety distance and lane) but rules r_2, r_3 are incomparable. This generalizes MOS which is a “flat” hierarchy where no objective (rule) is more critical than any other one.

6 Open Challenges and Opportunities

Despite the progress reviewed in this paper, several fundamental challenges remain open. In contrast to Salzman et al. (2023) who discuss technical challenges that are the foundations for advancing MOS algorithms, here we focus on challenges and opportunities that will increase the *impact* of MOS.

Scalability and dimensionality. Most existing algorithms scale poorly when the number of objectives grows beyond two or three. Approximate and bounded suboptimal MOS algorithms partially address this issue, but there is no consensus on how to effectively navigate high-dimensional cost spaces which may be essential in real-world applications.

Dynamic and uncertain environments. Real-world deployment increasingly requires algorithms that adapt to changing graphs or stochastic models. While recent works study dynamic MOS, MOSSP and MOMDP, current algorithms mostly remain theoretical or are limited to small instance sizes. Developing practical, general-purpose dynamic MOS algorithms is an interesting research opportunity.

Preference elicitation and user modeling. In many applications, decision makers cannot provide trade-offs upfront. Integrating preference elicitation into the search process—by interactively presenting Pareto-optimal candidates and learning from user choices—remains an underexplored yet impactful research direction. Combining MOS algorithms with methods from preference learning and human-in-the-loop AI is another research opportunity.

Cross-fertilization between research communities. Important opportunities exist at the interface of MOS and other AI subfields. In reinforcement learning, MORL is rediscovering many algorithmic ideas from MOS; conversely, MOS can benefit from policy-gradient and distributional methods. In many domains such as robotics, large-scale transport systems and OR, multiple objectives are prevalent but existing MOS formulations need to be adapted to be applied effectively.

Benchmarks. Classical MOS benchmarks focus on road networks and grid worlds, while robotics emphasizes motion-planning roadmaps, and reinforcement learning relies on synthetic MO-MDPs. This fragmentation hampers comparison across research communities.

A standardized benchmark suite that spans classical MOS, stochastic and dynamic settings, and application-inspired domains (such as transportation, robotics and chemistry) would be a major step forward. Beyond static datasets, benchmarks should include interactive tasks for preference elicitation and evaluation metrics that reflect both efficiency and effectiveness. The community would benefit from a shared repository of graphs, environments, and evaluation protocols to foster reproducibility and comparability.

7 Conclusion

MOS has rapidly expanded from a niche research topic to a broad principle that influences many disciplines and applications and studied by multiple communities. On the algorithmic side, there have been significant improvements in exact search, new approximate and parallel algorithms, and theoretical insights. On the applications side, numerous communities have started to formulate their problems in terms of trade-offs between different metrics and adopt MOS methods to handle these trade-offs. While MOS was the focus of this paper, there should be more cross fertilization between different multi-objective optimization approaches.

Acknowledgments

This research was supported by Grant No. 2021643 from the United States-Israel Binational Science Foundation (BSF).

References

- AAAI. 2024. Tutorial on Recent Advances in Multi-Objective Search. <https://sites.google.com/usc.edu/aaai24-mos-tutorial/home>.
- Ahmadi, S.; Sturtevant, N. R.; Harabor, D.; and Jalili, M. 2024. Exact Multi-objective Path Finding with Negative Weights. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 11–19.
- Ahmadi, S.; Sturtevant, N. R.; Raith, A.; Harabor, D.; and Jalili, M. 2025. Parallelizing Multi-objective A* Search. arXiv:2503.10075.
- Ahmadi, S.; Tack, G.; Harabor, D.; and Kilby, P. 2021. Bi-Objective Search with Bi-Directional A*. In *European Symposium on Algorithms (ESA)*, volume 204 of *LIPICs*, 3:1–3:15.
- Alpert, S. D.; Solovey, K.; Klein, I.; and Salzman, O. 2025. Inspection Planning Under Execution Uncertainty. *IEEE Trans. Robotics*, 41: 2406–2423.
- Axelrod, B.; Kaelbling, L. P.; and Lozano-Pérez, T. 2018. Provably safe robot navigation with obstacle uncertainty. *The International Journal of Robotics Research (IJRR)*, 37(13-14): 1760–1774.
- Bast, H.; Dellling, D.; Goldberg, A. V.; Müller-Hannemann, M.; Pajor, T.; Sanders, P.; Wagner, D.; and Werneck, R. F. 2016. Route Planning in Transportation Networks. In Klieemann, L.; and Sanders, P., eds., *Algorithm Engineering - Selected Results and Surveys*, volume 9220 of *Lecture Notes in Computer Science*, 19–80.
- Bonet, B.; and Geffner, H. 2001. Planning as heuristic search. *Artificial intelligence*, 129(1-2): 5–33.
- Branke, J.; Deb, K.; Miettinen, K.; and Slowiński, R. 2008. *Multiobjective optimization: Interactive and evolutionary approaches*, volume 5252. Springer Science & Business Media.
- Breugem, T.; Dollevoet, T.; and van den Heuvel, W. 2017. Analysis of FPTASes for the multi-objective shortest path problem. *Computers & Operations Research*, 78: 44–58.
- Censi, A.; Slutsky, K.; Wongpiromsarn, T.; Yerhov, D. S.; Pendleton, S.; Fu, J. G. M.; and Frazzoli, E. 2019. Liability, Ethics, and Culture-Aware Behavior Specification using Rulebooks. In *IEEE International Conference on Robotics and Automation (ICRA)*, 8536–8542.
- Chen, D. Z.; Trevizan, F.; and Thiébaux, S. 2023. Heuristic Search for Multi-Objective Probabilistic Planning. In *Association for the Advancement of Artificial Intelligence (AAAI)*.
- Clímaco, J.; and Pascoal, M. 2012. Multicriteria path and tree problems: discussion on exact algorithms and applications. *Int. Trans. in OR*, 19(1-2): 63–98.
- Coello, C. C.; and Lechuga, M. S. 2002. MOPSO: A proposal for multiple objective particle swarm optimization. In *Congress on Evolutionary Computation (CEC)*, volume 2, 1051–1056.
- Cuchý, M.; Vokřínek, J.; and Jakob, M. 2024. Multi-Objective Electric Vehicle Route and Charging Planning with Contraction Hierarchies. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 114–122.
- Current, J.; and Marsh, M. 1993. Multiobjective transportation network design and routing problems: Taxonomy and annotation. *Eur. J. of OR*, 65(1): 4–19.
- de las Casas, P. M.; Borndörfer, R.; Kraus, L.; and Sedeño-Noda, A. 2021. An FPTAS for Dynamic Multiobjective Shortest Path Problems. *Algorithms*, 14(2): 43.
- de las Casas, P. M.; Kraus, L.; Sedeño-Noda, A.; and Borndörfer, R. 2023. Targeted multiobjective Dijkstra algorithm. *Networks*, 82(3): 277–298.
- de las Casas, P. M.; Sedeño-Noda, A.; and Borndörfer, R. 2021. An Improved Multiobjective Shortest Path Algorithm. *Comput. Oper. Res.*, 135: 105424.
- de las Casas, P. M.; Sedeño-Noda, A.; and Borndörfer, R. 2025. New Dynamic Programming algorithm for the Multi-objective Minimum Spanning Tree problem. *Comput. Oper. Res.*, 173: 106852.
- de las Casas, P. M.; Sedeño-Noda, A.; Borndörfer, R.; and Huneshagen, M. 2025. K-shortest simple paths using bi-objective path search. *Math. Program. Comput.*, 17(2): 349–384.
- Deb, K.; and Jain, H. 2013. An evolutionary many-objective optimization algorithm using reference-point-based non-dominated sorting approach, part I: solving problems with box constraints. *IEEE transactions on evolutionary computation*, 18(4): 577–601.
- Deb, K.; Pratap, A.; Agarwal, S.; and Meyarivan, T. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2): 182–197.
- ECAI. 2025. Multi-Objective Decision Making Workshop. <https://modem2025.vub.ac.be/>.
- Ehrgott, M. 2005. *Multicriteria Optimization (2nd ed.)*. Springer.
- Emmerich, M.; and Deutz, A. 2018. A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Natural computing*, 17(3): 585–609.
- Erb, S.; Kobitzsch, M.; and Sanders, P. 2014. Parallel bi-objective shortest paths using weight-balanced b-trees with bulk updates. In *SEA*, 111–122.
- Felten, F.; Talbi, E.-G.; and Danoy, G. 2024. Multi-Objective Reinforcement Learning Based on Decomposition: A Taxonomy and Framework. *Journal of Artificial Intelligence Research*, 79.
- Fernandes, I. F.; Goldberg, E. F. G.; Maia, S. M.; and Goldberg, M. C. 2020. Empirical study of exact algorithms for the multi-objective spanning tree. *Computational Optimization and Applications*, 75(2): 561–605.
- Fu, M.; Kuntz, A.; Salzman, O.; and Alterovitz, R. 2023. Asymptotically optimal inspection planning via efficient

- near-optimal search on sampled roadmaps. *The International Journal of Robotics Research (IJRR)*, 42(4-5): 150–175.
- Geißer, F.; Haslum, P.; Thiébaux, S.; and Trevizan, F. 2022. Admissible Heuristics for Multi-Objective Planning. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 100–109.
- Goldberg, A. V.; and Harrelson, C. 2005. Computing the shortest path: A search meets graph theory. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, volume 5, 156–165.
- Goldin, B.; and Salzman, O. 2021. Approximate Bi-Criteria Search by Efficient Representation of Subsets of the Pareto-Optimal Frontier. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 149–158.
- Guerriero, F.; and Musmanno, R. 2001. Label correcting methods to solve multicriteria shortest path problems. *J. Opt. Theory App.*, 111(3): 589–613.
- Halder, P.; and Althoff, M. 2025. Sampling-Based Motion Planning with Preordered Objectives. In *2025 IEEE Intelligent Vehicles Symposium (IV)*, 125–131.
- Halle, Y.; Felner, A.; Koenig, S.; and Salzman, O. 2025. A Preprocessing Framework for Efficient Approximate Bi-Objective Shortest-Path Computation in the Presence of Correlated Objectives. In *Symposium on Combinatorial Search (SoCS)*, 65–73.
- Hansen, E. A.; and Zilberstein, S. 2001. LAO^{*}: A heuristic search algorithm that finds solutions with loops. *Artificial intelligence*, 129(1-2): 35–62.
- Hansen, P. 1980. Bicriterion Path Problems. In *Multiple Criteria Decision Making: Theory and Application*, 109–127.
- Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2): 100–107.
- Hayes, C. F.; Rădulescu, R.; Bargiacchi, E.; Källström, J.; Macfarlane, M.; Reymond, M.; Verstraeten, T.; Zintgraf, L. M.; Dazeley, R.; Heintz, F.; Howley, E.; Irissappane, A. A.; Mannion, P.; Nowé, A.; Ramos, G.; Restelli, M.; Vamplew, P.; and Roijers, D. M. 2022. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 36(1).
- Hernández, C.; Yeoh, W.; Baier, J. A.; Zhang, H.; Suazo, L.; Koenig, S.; and Salzman, O. 2023. Simple and efficient bi-objective search algorithms via fast dominance checks. *Artificial intelligence*, 314: 103807.
- Hernández, C.; Zhang, H.; Koenig, S.; Felner, A.; and Salzman, O. 2024. Efficient Set Dominance Checks in Multi-Objective Shortest-Path Algorithms via Vectorized Operations. In *Symposium on Combinatorial Search (SoCS)*, 208–212.
- Hwang, C.-L.; and Masud, A. S. 2012. *Multiple objective decision making—methods and applications: a state-of-the-art survey*, volume 164. Springer Science & Business Media.
- ICAPS. 2024. Finding Multiple Plans for Classical Planning Problems. https://icaps24.icaps-conference.org/program/tutorials/2024_t02_finding_multiple_plans/.
- IJCAI. 2023. First International Workshop on Search and Planning with Complex Objectives. <http://idmlab.org/wiki/complex-objective/>.
- IJCAI. 2025. Gradient-Based Multi-Objective Deep Learning. https://gradnexus.github.io/IJCAI25_tutorial/.
- Kurbanov, T.; Cuchý, M.; and Vokřínek, J. 2023. Fast One-to-Many Multicriteria Shortest Path Search. *IEEE Trans. Intell. Transp. Syst.*, 24(10): 10410–10419.
- Lai, H.; Kannas, C.; Hassen, A. K.; Granqvist, E.; Westerland, A. M.; Clevert, D.-A.; Preuss, M.; and Genheden, S. 2025. Multi-objective synthesis planning by means of Monte Carlo Tree search. *Artificial Intelligence in the Life Sciences*, 7: 100130.
- Levin, M. S.; and Nuriakhmetov, R. I. 2011. Multicriteria Steiner Tree Problem for Communication Network. *CoRR*, abs/1102.2524.
- Li, H.; and Landa-Silva, D. 2011. An adaptive evolutionary multi-objective approach based on simulated annealing. *Evolutionary computation*, 19(4): 561–595.
- Mandow, L.; and De La Cruz, J. L. P. 2005. A new approach to multiobjective A* search. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, 218–223.
- Mandow, L.; and De La Cruz, J. L. P. 2010. Multiobjective A* Search with Consistent Heuristics. *Journal of the ACM*, 57(5): 1–25.
- Martins, E. Q. V. 1984. On a Multicriteria Shortest Path Problem. *European Journal of Operational Research*, 16(2): 236–245.
- Medrano, F. A.; and Church, R. L. 2015. A parallel computing framework for finding the supported solutions to a biobjective network optimization problem. *J. Multi-Criteria Decis. Anal.*, 22(5-6): 244–259.
- Miettinen, K. 2012. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media.
- NeurIPS. 2024. Pluralistic Alignment workshop. <https://pluralistic-alignment.github.io/>.
- OTP. 2025. OpenTripPlanner Deployments Worldwide. <https://docs.opentripplanner.org/en/latest/Deployments/>.
- Papadimitriou, C. H.; and Yannakakis, M. 2000. On the approximability of trade-offs and optimal access of web sources. In *EEE Symposium on Foundations of Computer Science (FOCS)*, 86–92.
- Peer, H.; Weiss, E.; Alterovitz, R.; and Salzman, O. 2025. Generalizing Multi-Objective Search via Objective-Aggregation Functions. *CoRR*, abs/2509.22085.
- Penlington, M.; Zanardi, A.; and Frazzoli, E. 2024. Optimization of Rulebooks via Asymptotically Representing Lexicographic Hierarchies for Autonomous Vehicles. arXiv:2409.11199.
- Perny, P.; and Spanjaard, O. 2008. Near Admissible Algorithms for Multiobjective Search. In *European Conference on Artificial Intelligence (ECAI)*, 490–494.

- Potthoff, M.; and Sauer, J. 2022a. Efficient Algorithms for Fully Multimodal Journey Planning. In *Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS)*, volume 106, 14:1–14:15.
- Potthoff, M.; and Sauer, J. 2022b. Fast Multimodal Journey Planning for Three Criteria. In *Symposium on Algorithm Engineering and Experiments (ALENEX)*, 145–157.
- Pulido, F.-J.; Mandow, L.; and Pérez-de-la Cruz, J.-L. 2015. Dimensionality reduction in multiobjective shortest path search. *Computers & Operations Research*, 64: 60–70.
- Ren, L.; Han, B.; Ma, H. M.; Huang, Z.; Wang, Y.; and Chen, X. 2022a. Enhanced Multi-Objective A* Using Balanced Binary Search Trees. arXiv:2202.08992.
- Ren, Z.; Hernández, C.; Likhachev, M.; Felner, A.; Koenig, S.; Salzman, O.; Rathinam, S.; and Choset, H. 2025. EMOA*: A framework for search-based multi-objective path planning. *Artificial Intelligence*, 339: 104260.
- Ren, Z.; Rathinam, S.; and Choset, H. 2021a. Multi-objective Conflict-based Search for Multi-agent Path Finding. In *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE.
- Ren, Z.; Rathinam, S.; and Choset, H. 2021b. Subdimensional Expansion for Multi-Objective Multi-Agent Path Finding. *IEEE Robotics and Automation Letters*, 6(4): 7153–7160.
- Ren, Z.; Rathinam, S.; and Choset, H. 2023. A Conflict-Based Search Framework for Multiobjective Multiagent Path Finding. *IEEE transactions on automation science and engineering*, 20(2): 1262–1274.
- Ren, Z.; Rathinam, S.; Likhachev, M.; and Choset, H. 2022b. Multi-Objective Path-Based D* Lite. *IEEE Robotics and Automation Letters*, 7(2): 3318–3325.
- Rivera, N.; Baier, J. A.; and Hernández, C. 2022. Subset Approximation of Pareto Regions with Bi-objective A*. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 10345–10352.
- Rojers, D. M.; and Whiteson, S. 2017. *Multi-Objective Decision Making*, volume 12 of *Synthesis Lectures on Artificial Intelligence and Machine Learning*. Morgan & Claypool Publishers. ISBN 9781627058648.
- RSS. 2025. Workshop on Multi-Objective Optimization and Planning in Robotics. <https://sites.google.com/view/moors25/home>.
- Salzman, O.; Felner, A.; Hernández, C.; Zhang, H.; Chan, S.; and Koenig, S. 2023. Heuristic-Search Approaches for the Multi-Objective Shortest-Path Problem: Progress and Research Opportunities. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, 6759–6768.
- Sanders, P.; and Mandow, L. 2013. Parallel label-setting multi-objective shortest path search. In *Int. Symp. on Parallel and Distributed Processing*, 215–224.
- Serafini, P. 1987. Some considerations about computational complexity for multi objective combinatorial problems. In *Recent advances and historical development of vector optimization*, 222–232. Springer.
- Shovan, S. M.; Khanda, A.; and Das, S. K. 2025. Parallel Multi Objective Shortest Path Update Algorithm in Large Dynamic Networks. *IEEE Trans. Parallel Distributed Syst.*, 36(5): 932–944.
- Skriver, A. J. 2000. A classification of bicriterion shortest path (BSP) algorithms. *Asia Pacific J. of Oper. Res.*, 17(2): 199–212.
- Skyler, S.; Atzmon, D.; Felner, A.; Salzman, O.; Zhang, H.; Koenig, S.; Yeoh, W.; and Ulloa, C. H. 2022. Bounded-cost bi-objective heuristic search. In *Symposium on Combinatorial Search (SoCS)*, volume 15, 239–243.
- Skyler, S.; Shperberg, S. S.; Atzmon, D.; Felner, A.; Salzman, O.; Chan, S.; Zhang, H.; Koenig, S.; Yeoh, W.; and Ulloa, C. H. 2024. Theoretical Study on Multi-objective Heuristic Search. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, 7021–7028.
- Slutsky, K.; Yershov, D. S.; Wongpiromsarn, T.; and Frazzoli, E. 2021. Hierarchical Multiobjective Shortest Path Problems. In *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, volume 17, 261–276.
- SOCS. 2023. Master Class On Clashing Wavefronts in Pareto Search. <https://socs23.search-conference.org/program/master-classes>.
- Sourd, F.; and Spanjaard, O. 2008. A Multiobjective Branch-and-Bound Framework: Application to the Biobjective Spanning Tree Problem. *INFORMS J. Comput.*, 20(3): 472–484.
- Southirathn, T.; Koo, B.; Lu, Y.; and Kim, S. 2025. CombiMOTS: Combinatorial Multi-Objective Tree Search for Dual-Target Molecule Generation. In *International Conference on Machine Learning (ICML)*.
- Stern, R.; Sturtevant, N. R.; Felner, A.; Koenig, S.; Ma, H.; Walker, T. T.; Li, J.; Atzmon, D.; Cohen, L.; Kumar, T. K. S.; Barták, R.; and Boyarski, E. 2019. Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks. In *Symposium on Combinatorial Search (SoCS)*, 151–158.
- Stewart, B. S.; and White III, C. C. 1991. Multiobjective A*. *Journal of the ACM (JACM)*, 38(4): 775–814.
- Storandt, S. 2012. Route planning for bicycles—exact constrained shortest paths made practical via contraction hierarchy. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 234–242.
- Tarapata, Z. M. 2007. Selected multicriteria shortest path problems: An analysis of complexity, models and adaptation of standard algorithms. *Int. J. Appl. Math. Comput. Sci.*, 17(2).
- Tsaggouris, G.; and Zaroliagis, C. D. 2009. Multiobjective Optimization: Improved FPTAS for Shortest Paths and Non-Linear Objectives with Applications. *Theory of Com. Sys.*, 45(1): 162–186.
- Ulungu, E.; and Teghem, J. 1991. Multi-objective shortest path problem: A survey. In *Workshop on Multicriteria Decision Making: Methods–Algorithms–Applications*, 176–188.
- Vazirani, V. V. 2001. *Approximation algorithms*. Springer.

- Vincke, P. 1976. A New Approach to Multiple Criteria Decision-Making. In *Multiple Criteria Decision Making*, 341–350.
- Wang, F.; Zhang, H.; Koenig, S.; and Li, J. 2024. Efficient Approximate Search for Multi-Objective Multi-Agent Path Finding. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 613–622.
- Warburton, A. 1987. Approximation of Pareto Optima in Multiple-Objective Shortest-Path Problems. *Operations Research*, 35(1): 70–79.
- Zhang, H.; Salzman, O.; Felner, A.; Kumar, T. K. S.; and Koenig, S. 2024a. Bounded-Suboptimal Weight-Constrained Shortest-Path Search via Efficient Representation of Paths. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 680–688.
- Zhang, H.; Salzman, O.; Felner, A.; Kumar, T. K. S.; Skyler, S.; Ulloa, C. H.; and Koenig, S. 2023a. Towards Effective Multi-Valued Heuristics for Bi-objective Shortest-Path Algorithms via Differential Heuristics. In *Symposium on Combinatorial Search (SoCS)*, 101–109.
- Zhang, H.; Salzman, O.; Felner, A.; Kumar, T. K. S.; Ulloa, C. H.; and Koenig, S. 2023b. Efficient Multi-Query Bi-Objective Search via Contraction Hierarchies. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 452–461.
- Zhang, H.; Salzman, O.; Felner, A.; Kumar, T. K. S.; Ulloa, C. H.; and Koenig, S. 2024b. Speeding Up Dominance Checks in Multi-Objective Search: New Techniques and Data Structures. In *Symposium on Combinatorial Search (SoCS)*, 228–232.
- Zhang, H.; Salzman, O.; Felner, A.; Ulloa, C. H.; and Koenig, S. 2024c. A-A*pex: Efficient Anytime Approximate Multi-Objective Search. In *Symposium on Combinatorial Search (SoCS)*.
- Zhang, H.; Salzman, O.; Kumar, T. S.; Felner, A.; Ulloa, C. H.; and Koenig, S. 2022. A*pex: Efficient approximate multi-objective search on graphs. In *International Conference on Automated Planning and Scheduling (ICAPS)*, volume 32, 394–403.
- Zhang, Q.; and Li, H. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6): 712–731.