Concerning optimization parameters and preferences

Nils Wilde and Dylan A. Shell

***Abstract*— This short paper's purpose is to explore some closely related, and we venture oft-conflated, concepts that arise naturally whenever optimization is applied to robotics problems. Using the concrete problem of a mobile robot choosing from among multiple paths that have differing characteristics, we examine the basic question of how one might express preferences in a manner useful to an optimizer. While it is all too common to have parameters and hyper-parameters of various sorts describing prioritizations, constraints, or weightings in practice—we ask: to what extent do such mechanisms actually succeed at expressing preferences? A few extremely elementary examples appear to indicate such techniques fall short. We suggest that, in fact, the fundamental question of how to reasonably specify preferences is still not entirely settled.**

I. Introduction

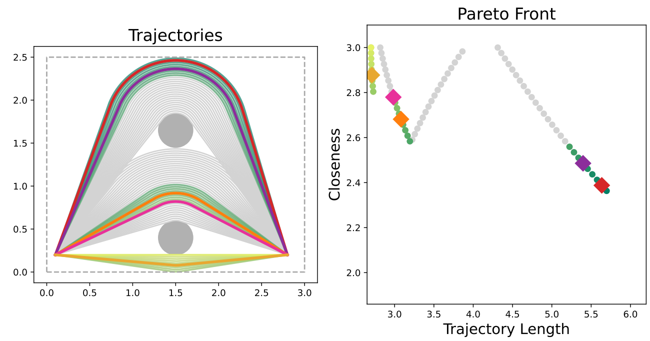
To produce reasonable behaviour, a robot should execute actions (for a mobile robot: choose trajectories) that minimize fuel, effort, or time while satisfying requirements like eventually reaching a goal, meeting deadlines, and maintaining safety (e.g., by avoiding collisions). Once cast in the formal language of optimization, these become objectives and constraints. Typically, robots must strike a compromise between the various competing objectives; furthermore, optima are oftentimes found to occur at constraint boundaries. The resulting behaviour, thus, may depend on the *exact* choice of constraint-defining constants and weighting multipliers.

A. Narration of the dilemma: constraints versus objectives

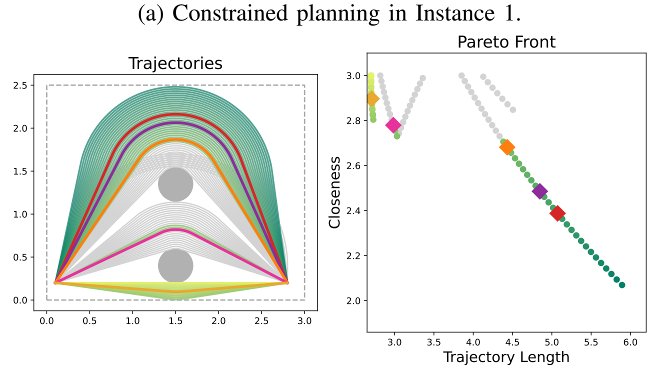
Concretely: consider the hard constraint that a robot ought to never collide with an obstacle. Owing to imprecision, uncertainty, and even the nerve-wracking observation of something passing close enough to risk the paintwork, one may naturally expect that some *safety margin* be employed. For instance we could require that the robot never come within $5\,\mathrm{cm}$ of any obstacle. Having an inflated obstacle as a strict constraint, the robot might find that it must do a detour, traversing a path twice as long because otherwise it would have passed within $4.95\,\mathrm{cm}$ of an obstacle.

It appears that softening the constraint may avoid such a compromise between competing demands. One approach is to downgrade distance-to-the-nearest-obstacle from a constraint to being expressed, somehow, within the objective instead. It might contribute as a penalty term or it might become a separate objective in the multi-objective setting:

- In the former case, there is a choice of a particular 'ramp' function (e.g., a Lorentz cone, or an exponential), and/or weighting parameters. Then, the decision that a $\leqslant 1\,\mathrm{mm}$ infraction to halve path length is justified depends on the form those weighting parameters take.



(a) Constrained planning in Instance 1.



(b) Constrained planning in Instance 2.

Fig. 1: Constrained optimization for capturing preferences. The ground set of trajectories is shown in grey, Pareto-optimal ones in shades of green. Highlighted colours are solutions corresponding to fixed constraints on closeness.

- In the multi-objective case, instead of directly obtaining some expression on the objective, a Pareto front is produced, and then an assessment made to determine whether $\leqslant 1\,\mathrm{mm}$ closer for a $50\,\%$ reduction in the path length would be desirable or not.

Typically, in the second case, a point on the Pareto front is identified. Since a weighting corresponding to that point may be obtained from it, we may be lead to wonder whether a multi-objective formulation is a better approach than tuning parameters. Is it more principled? More informed? Is it likely to do a better job of capturing (which we take to involve both eliciting and expressing) underlying human preferences?

B. The trouble with constraints: an illustrative example

Fig. 1 illustrates the effect of slight variations in the planning problem on optimization for fixed parameters. Here we consider a Dubins car navigating around two obstacles while minimizing trajectory length and closeness to the obstacles (the setup is adapted from [1]). In Fig. 1a we highlight five solutions, corresponding three qualitatively different behaviours that ought to be thought of as three classes of preferences: yellow for preference one, pink and orange for preference two, and purple and red for preference three. We may express these preferences using constraints on closeness while minimizing length. Now consider another instance of the same problem, shown in Fig. 1b, where the upper obstacle is moved vertically downwards only slightly. We notice that while some solutions remain qualitatively the same (yellow, pink and purple), others change: most notably, orange is no longer passing between obstacles, but moves around them as its constraint cannot be satisfied . The red solution is no longer at the extreme end of the trade-offs but has moved closer to center of the Pareto front.

In this first example, we did not specify what preference is intended to be encapsulated by these constraints. Yet, we observe that fixed constraints produce a variety of behaviours even under only small variations of planning instances.

C. *Preferences vs Parameters vs Pareto fronts*

As per the discussion in §I-A, instead of constraints, one might incorporate parameters to balance multiple contributions in a single objective. That leads to the question: If we form a single objective via parameters to weight different contributing factors, is the tuning of such parameters just the same as making a selection from a Pareto front?

We answer this question in the negative, suggesting that while making a selection from a Pareto front is closer to soliciting a preference, it is not exactly that either.

Firstly, is it always true that identifying the point on the Pareto front corresponds to a solution obtained under some parameter settings? They are equivalent either when the Pareto front is convex (a property of the problem), or by permitting a richer parameterization than merely linearly weighting. (Hence, we emphasized that the choice of 'ramp' function may also be an important knob to tune.)

Secondly, the Pareto front serves as a sort of visual summary: it allows assessment of the nearby behavior of the trade-offs involved. The user can decide *ex post facto* that $\pm 1\,\mathrm{mm}$ is not worth worrying about given the improvements that can be obtained by relaxing it. In other words, the Pareto front helps resolve uncertainty in the interplay of factors, and the selection can be made in context once one better understands the relative impact of changes. Notice that $\pm 1\,\mathrm{mm}$ may matter a great deal in some places (e.g., other than around the synthetic $5\,\mathrm{cm}$ margin). Further, relaxing $\pm 1\,\mathrm{mm}$ may not be justified in terms of path-length returns elsewhere either.

Thus, we see that as describing *preferences*, selections from the Pareto front are better informed. But, also, this is a fickle statement. Here are some reasons for concern:—

- Identifying a point on the Pareto front doesn't always uniquely identify a relative weighting of objectives.
- Even if a weighting/prioritization is identified (as in [2], [3], [4], [5]), is there any claim that this is reflective of generally preferred trades to be made for other problem instances?

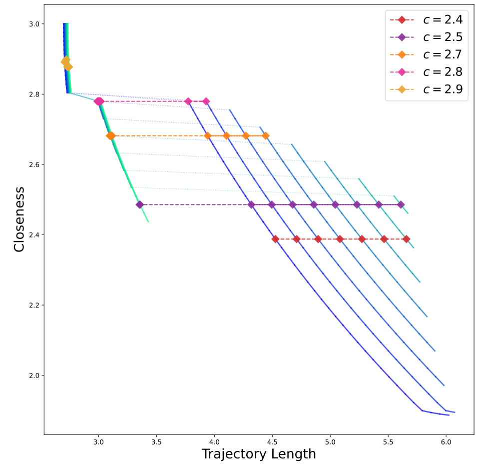


Fig. 2: Pareto fronts and solutions for constrained optimization over a family of Pareto fronts. Blue to green lines are the Pareto fronts across instances; highlighted colours are the solutions corresponding to fixed constraints on closeness.

- Preferences must satisfy some basic axioms (e.g., transitivity); why ought we to suppose those properties hold whenever selections are made from a Pareto front?

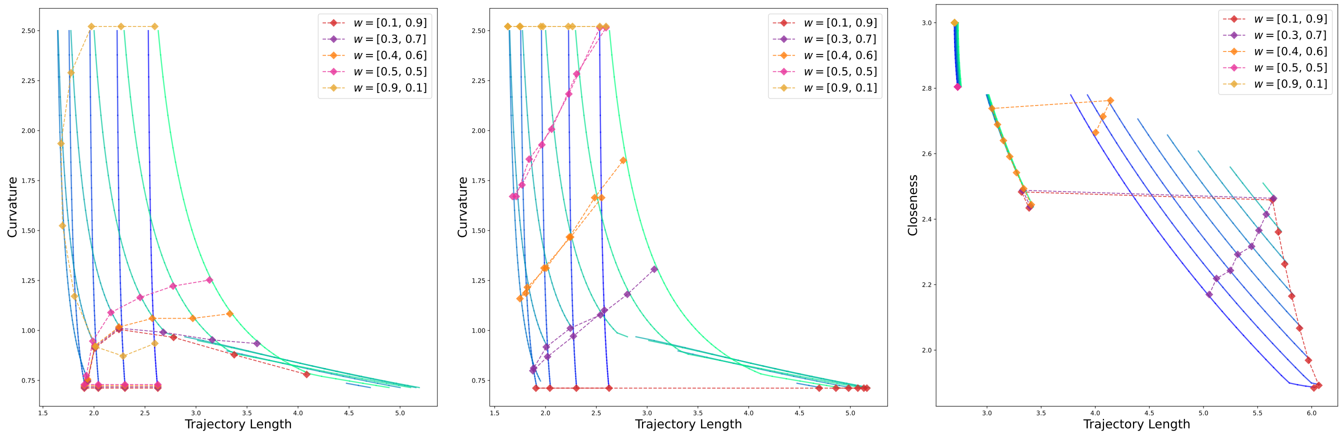
II. Case Study

Additional detail will help elaborate further on the preliminary arguments presented in the preceding section. To do so, in this section we examine two planning problems, showcasing how constraints and weights are limited in their ability to represent preferences. Our problems are simple path planning for a Dubins car. In Scenario A, a robot navigates around static obstacles (the setting that was illustrated in Fig. 1) seeking to minimize path length and closeness to the obstacles, the latter is measured the difference of the width of the environment and the closest distance to an obstacle along the path. In Scenario B, a robot navigates in free space and we seek to minimize path length and maximum curvature. For both scenarios, we generate a family of closely related instances. In Scenario A the start and goal positions remain fixed, as does the position of the lower obstacle, but the y position of the upper obstacle varies. In Scenario B, the start pose remains the same; the x position and orientation of the goal pose also remain constant, but the y position varies.

A. *Fixed constraints across instances*

We consider the setup from Fig. 1 and extend it to a broader set of instances of Scenario A. Fig. 2 shows the Pareto fronts for nine different instances, and highlights the solutions found by minimizing trajectory length under five fixed constraints on the closeness objective.

Overall, the constraints ensure that the same closeness values are attained across instances. Yet, the implications are quite different for the individual choices. Indeed, this is strikingly visible in Fig. 2. For $c = 2.9$ (yellow), the constraint is not tight and solutions only vary marginally. In contrast, the smallest constraint of $c = 2.4$ (orange) is not feasible in all instances. When seeking to express



(a) WS in Scenario B. (b) WM in Scenario B. (c) WM in Scenario A.

Fig. 3: Pareto fronts and solutions for weighted objetives over a family of Pareto fronts. Blue to green lines are the Pareto fronts across instances; highlighted colours are the solutions corresponding to fixed constraints on closeness.

preferences with constraints (unlike using constraints to enforce a hard safety requirement), this might be ill-specified. Even when a high clearance is desired, some compromise is likely more desirable than not returning any solution. In the three intermediate settings we observe large jumps—this corresponds to switching from moving around the obstacles to moving between them. In particular, $c = 2.8$ captures the case described in §I-A: Strictly adhering to the constraint can cause a long detour in some instance; a slightly higher value of 2.82 would already allow the robot to always take the same route across all instances.

In summary, constraints provide strict consistency for the value of closeness across instances. Yet, constraints are blind to close-by solutions. Choosing a constraint in a particular instance can fail to capture preferences as it entirely disregards solutions exceeding the constraints, irrespective of the prospective gains—either on the other objective, or even just in terms of finding a feasible solution.

B. Fixed objective weights across instances

To avoid the strictness of fixed constraints, motion planners often use scalar objective functions that balance cost terms [6]. We consider two scalarizations: The first is linear scalarization, or weighted sum (WS), which is widely used in robot planning [7]. The objective becomes $\min_P w_1 f_1(P) + w_2 f_2(P)$, where f_1 and f_2 are our objective functions and w_1, w_2 are tunable positive weights. Second, we consider Chebyshev scalarization, or weighted maximum (WM) [1], [8], which takes the form $\min_P \max(w_1 f_1(P), w_2 f_2(P))$. This approach has provable advantages as it finds all points on a Pareto front, while WS only obtains solutions on the convex parts of Pareto fronts [1].

a) Weighted sum objective: When using WS we are restricted to Scenario B, as the Scenario A has non-convex Pareto fronts where WS is ill-suited. Fig. 3a illustrates a family of Pareto fronts and the solutions for each of them across a set of five different weights in a WS objective.

We notice that solutions are no longer keeping one objective value constant across instances (as in Fig. 2). Instead, for each fixed weight the solutions adapt to the shape of the respective Pareto-front. Yet, this comes at two caveats.

First, the trends for each weight are unintuitive. Considering $w = [.9, .1]$ (yellow), there are solutions with approximately the same trajectory length but vastly different curvature. When using WS all solutions for a fixed weight attain the same *steepness*. Yet, individual weights do not show a consistent trend across instances with respect to the objective values attained, raising the question what preferences ought to be expressed with weights is WS formulations.

Second, we notice that solutions spread unevenly across individual Pareto fronts. In some instances, even large differences in weights lead to only minor or even no differences in the corresponding trajectories. Most notably, all weights except of $[.9, .1]$ become increasingly similar and eventually correspond to the same solution. This results in *preference indefiniteness*: selecting a solution in such a planning instance does not uniquely determine a weight. Instead, it remains ambiguous which weight captures an expressed preference, choosing a weight can be misaligned with the preference when moving to a different instance. For example, one might select weight $[.1, .9]$ (red), while $[.5, .5]$ (magenta) was an equally plausible choice. Yet, $[.1, .9]$ produces substantially longer trajectories in the other instances.

b) Weighted maximum objective: Next, we study the WM scalarization. To begin with, we continue to keep our attention on Scenario B, shown in Fig. 3b.

In contrast to WS, the different weights exhibit distinct characteristics across Pareto fronts: Each weight attains the same ratio of weighted objectives, i.e., ${w_1 f_1(P)}/{w_2 f_2(P)}$, leading to clear characteristics of the solutions for each weight. The problem of preference indefiniteness is seemingly much less prevalent. Yet, $w = [.9, .1]$ and $w = [.5, .5]$ attain the same solution for one Pareto front. While preference indefiniteness was more present for WS, there can remain a large ambiguity for weights considering unlucky instances.

However, the result change in Scenario A, shown in Fig. 3c. A key difference is that the Pareto fronts are non-convex and not continuous but rather exhibit large gaps. Some weights ($w = [.9, .1]$ and $w = [.5, .5]$) return the same respective solution across all instances, namely short paths close to the lower obstacle. The other weights ($w = [.4, .6]$, $w = [.3, .7]$ and $w = [.1, .9]$) still exhibit consistent and distinct trends across *some* of the instances, yet these do not resemble the linear trend so clearly visible in Scenario B. Further, the notion of preference indefiniteness resurfaces: the same three weights produce identical solution in several instances, but strongly different solutions in others.

In summary, fixed weights of cost functions achieve substantially different characteristics than fixed constraints. The points selected on each Pareto front can shift excessively. Further, the two examined scalarization methods differ: WS does not exhibit a clearly interpretable trend or notion of preference across instances and strongly suffers from ambiguities of weights. WM showed very structured trends is Scenario B, yet this did not continue in the more complex Scenario A, where the same challenges as for WS arise.

The case study shows that parametrized optimization techniques are highly sensitive to even small changes in the problem instance. It appears questionable that any of the three examined approaches has a notion of generalization that is suitable to express preferences for robot behaviour.

III. Implications and Open Problems

The preceding examples spur us to recapitulate oft-repeated suggestions that the multi-objective perspective has several strengths:

+ Firstly, during the process of seeking optima, multi-objective optimization correctly models situations in which no total order has been specified. Or, put more precisely, it avoids the mistake of imposing a total order where none had been expressed in the input. This might be understood as a form of unbiasedness.
+ Secondly, the multi-objective approach directly avoids the vexed issue of combining incommensurate quantities. In fact, cases involving incompatible or incomparable quantities highlight how *ad hoc* or ill-founded any such combination must be.
+ Third, in constructing a Pareto front, it enables visibility, not only of individual solutions but also of their relationships, thereby facilitating an informed selection. It is informed not just in the sense of being relative to other solutions, but—in doing this after establishing how the underlying problem constrains available solutions—it does this relative to what cannot be attained.

Adopting a multi-objective view and selecting a solution via the Pareto front has the apparency of expressing a preference, but it is not a panacea, either:

– While it appears that multi-objective optimization eschews the idea of ‘point solutions’ for running the whole gamut of the solution space, this can be quite misleading. Visualizing the Pareto front for one instance expresses nothing about other instances.
– One interpretation of the multi-objective approach is to delay making a decision until the local landscape is visible. But the geometry of the Pareto Frontier can be drastically altered by, for example, considering the $\log$ of some objective. In fact any co-monotonic transformation preserves the semantics of the optimization, but can drastically alter the geometry. This might be understood as a form of bias toward some particular projection or view of the inter-relationships.
– The concept of delay is appropriate if one is attempting to utilize more information to resolve uncertainty.[1] But the Pareto front provides a poor expression of uncertainty—distributional information might be clearer. The issue of generalization across multiple instances has been mentioned, but one might also ask about representing preferences across an ensemble of users. (Or even, joint information.)

Note that the considerations given above deal with how well-formed the optimization questions are, and have entirely put aside questions of tractability when employing multi-objective methods. One expects complexity increases (both computationally and also in actually providing a practical visualization of the Pareto front) in employing multi-objective approaches.

The above lead us to suggest the following open research questions for the community.

1) How might multi-objective optimization techniques be better informed by their downstream uses? For instance, the exploration phase concerned with the ‘lie of the land’ may demand different operation than during a selection of a single solution. Similarly, knowing that a single solution is sought with the specific aim to generalize along a certain dimension might alter the upstream operation of identifying candidate solutions. In this case, knowing the type of intended scalarization (e.g. a one-size-fits-most risk-based one), could be useful to the optimizer.
2) How might the Pareto front concept be altered to help visualize uncertainty? Ideally, it could show this explicitly. This should include aspects of weight non-uniqueness, per-instance variation, and potentially even preferences across user populations.
3) How might the Pareto front visualized be ‘scrubbed’ of arbitrary aspects of the problem? E.g., how does one remove scaling artifacts that drastically affect visual aspect ratios.
4) The interplay between decision-theoretic axioms that describe preferences (e.g., antisymmetry and transitivity) versus the selection of options from a Pareto front ought to be better understood. For instance, what selection criteria ensure such axioms will be met?
5) How might machine learning concepts for generalization (e.g., empirical risk minimization) be brought to bear on the problem of preference generalization?

[1] We refer the reader to the meme showing Ron Swanson, the Parks and Recreation character, declaring: “I don’t procrastinate. I intentionally wait until the last minute because then I will be older, and therefore wiser.”

References

[1] N. Wilde, S. L. Smith, and J. Alonso-Mora, “Scalarizing multi-objective robot planning problems using weighted maximization,” *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2503–2510, 2024.

[2] E. Biyik, M. Palan, N. C. Landolfi, D. P. Losey, and D. Sadigh, “Asking easy questions: A user-friendly approach to active reward learning,” in *Proc. of CoRL*, 2020, pp. 1177–1190.

[3] N. Wilde, D. Kulić, and S. L. Smith, “Active preference learning using maximum regret,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 10 952–10 959.

[4] H. J. Jeon, S. Milli, and A. D. Dragan, “Reward-rational (implicit) choice: A unifying formalism for reward learning,” in *Advances in Neural Information Processing Systems (NIPS)*, Dec. 2020.

[5] B. W. Knox, S. Hatgis-Kessell, S. Booth, S. Niekum, P. Stone, and A. Allievi, “Models of human preference for learning reward functions,” *Trans. Machine Learning Research*, 2024.

[6] M. Zucker, N. Ratliff, A. D. Dragan, M. Pivtoraiko, M. Klingensmith, C. M. Dellin, J. A. Bagnell, and S. S. Srinivasa, “Chomp: Covariant hamiltonian optimization for motion planning,” *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1164–1193, 2013.

[7] A. Botros, N. Wilde, A. Sadeghi, J. Alonso-Mora, and S. L. Smith, “Regret-based sampling of pareto fronts for multi-objective robot planning problems,” *IEEE Transactions on Robotics*, vol. 40, pp. 3778–3794, 2024.

[8] K. Kalavadia, S. Dutta, Y. V. Pant, and S. L. Smith, “Efficient multi-objective planning with weighted maximization using large neighbourhood search,” *arXiv preprint arXiv:2604.04826*, 2026.