Dynamic Multi-Objective Ergodic Path Planning Using Decomposition Methods

Abigail Breitfeld[1] and David Wettergreen[1]

Abstract— **Robots are often employed in hazardous or inaccessible environments, such as disaster sites, extraterrestrial terrains, agricultural fields, and ocean floors. Autonomous operation is crucial in these scenarios to reduce reliance on human operators and enable real-time decision-making. However, robots must balance multiple, often conflicting, objectives. These objectives are subject to change based on new data or evolving conditions. This paper presents a novel approach to dynamic multi-objective trajectory planning. The proposed method leverages the boundary intersection decomposition technique to adaptively plan trajectories that balance multiple evolving objectives. Our approach ensures efficient and effective exploration by continuously optimizing the trade-offs between changing objectives. We show that our method performs on average 34% better in terms of solution quality on the dynamic multi-objective trajectory planning problem as compared to prior work.**

I. Introduction

Robots are often used to explore environments that are dangerous or infeasible for humans to visit, such as disaster sites [1]–[3] and other planetary bodies [4], [5]. Robots also aid in monitoring phenomena like plant health for agricultural applications [6], [7] or seafloor mapping [8], [9].

In these scenarios, it is advantageous to operate autonomously, without time-intensive human decision-making. However, a robot should balance all factors that a human operator would: risk to the vehicle, power constraints, scientific information gain, or the importance of particular regions. All these objectives that must be balanced are subject to change over time or as the robot gains more information. For example, taking a measurement in one place may change the robot's belief about similar locations. Any algorithm should be able to adapt to this changing environment, as a human decision-maker would. In general, an autonomous trajectory planning algorithm should be able to effectively balance all objectives, even as those objectives change over time.

This work develops a novel method for addressing the problem of dynamic multi-objective trajectory planning. Given objectives that may change over time or with new measurements, we wish to efficiently plan the most effective trajectory for a robot to traverse. In order to achieve a good trade-off between objectives, we propose a method based on the boundary intersection decomposition approach [10], which ensures that even as objectives change over time, our method provides solutions that adapt well for every objective (see Figure 1). We show through experimentation on real-world data that our method consistently produces better quality solutions than prior work.

[1]Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA

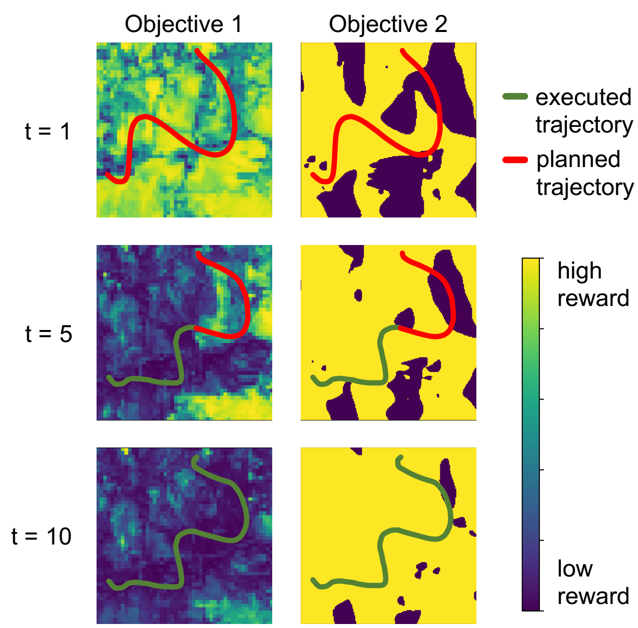

Fig. 1: Boundary intersection-based dynamic multi-objective ergodic search (BI-MO-ES) plans a trajectory which balances multiple objectives, adapting the trajectory over time as these objectives change. This method prioritizes finding a trajectory which gives a good trade-off between objectives.

II. Background and Related Work

A. Planning for Exploration and Exploitation

This work focuses on the exploration of objective maps. An objective map represents a prior belief about the spatial distribution of information or reward over an area. We wish to plan a trajectory which both explores the region and prioritizes visiting the areas corresponding to high reward.

1) Coverage and Informative Path Planning: In general, Coverage Path Planning (CPP) methods plan trajectories that visit all points in a given region [11]. CPP methods discretize the space and devise methods to exhaustively visit every location [12]–[17]. In contrast, an Informative Path Planning (IPP) method aims to maximize information gained over a trajectory [18]. IPP methods may not produce a trajectory that visits every location (covers the space), but instead will construct the most informative sequence of actions [18]–[24].

2) Ergodic Planning: We wish to consider both exploration of a region and exploitation of available information. Ergodic search strikes this balance between exploration and exploitation by planning trajectories such that the distribution of information seen by the robot matches the distribution of information in the objective map [25]. We adopt ergodic search as our path generation strategy in this work due to its balanced approach and straightforward implementation.

Ergodic search optimizes a trajectory such that the Fourier coefficients of the trajectory's *spatial time-average statistics* match the Fourier coefficients of the spatial distribution of the objective map. The *spatial time-average statistics* of the trajectory represent the proportion of time a robot spends at any state $x \in \mathcal{X}$ (where $\mathcal{X} \subset \mathbb{R}^{\mathrm{d}}$ is the d-dimensional search domain), and are defined as

$$C^t(x, \gamma(t)) = \frac{1}{t} \int_0^t \delta(x - \gamma(\tau)) d\tau, \tag{1}$$

where $\gamma(t)$ represents the trajectory over time t and δ is the Dirac delta function. The *ergodic metric* compares the Fourier coefficients of the spatial time-average statistics of the trajectory (c_k) with the Fourier coefficients of the spatial distribution of the objective map (ξ_k). This quantifies the difference between the information in the map and the information seen by the robot. The goal of ergodic search is to minimize the ergodic metric, such that the information seen by the robot matches the information map. The ergodic metric is defined as

$$\Phi(\gamma(t)) = \sum_{k=0}^{K} \lambda_k \left| c_k(\gamma(t)) - \xi_k \right|^2, \tag{2}$$

where λ_k are the weights assigned to each Fourier coefficient, and K is the number of Fourier bases chosen (we choose $K = 10$). For this work, $\lambda_k = \left(1 + \left\| \frac{k}{\pi} \right\|^2\right)^{-\frac{d+1}{2}}$, meaning more importance is placed on lower frequency Fourier components, which correspond to large-scale variations in the objective map.

The original ergodic search algorithm finds optimal controls that minimize the ergodic metric for a robot with defined dynamics [25]. However, the ergodic metric can be used to directly optimize the trajectory points in the spatial domain if the dynamics of a particular robot are not known or relevant.

B. Multi-Objective Optimization and Pareto Optimality

We wish to solve the multi-objective problem of exploration of multiple maps with a mobile robot. A multi-objective problem is defined in general as

$$\begin{aligned} &minimize \quad F(x) = [f_1(x), f_2(x), \ldots, f_M(x)] \\ &subject\ to \quad x \in \Omega \end{aligned} \tag{3}$$

where $x = [x_1, x_2, ..., x_N]$ represents a solution (decision vector), $f_m(x)$ represents the m^{th} objective function, M is the total number of objective functions, and Ω is the feasible decision space. We focus on planning a trajectory that covers multiple objective maps simultaneously. Thus, our solutions, or decision vectors x, are trajectories, and our objective functions $f_m(x)$ represent a coverage metric for each objective map (for example, the ergodic metric described above).

Multi-objective optimization typically focuses on finding *Pareto optimal* solutions. Pareto optimality refers to the concept of finding "non-dominated" solutions: solutions for which there exists no other solution that can perform better in one objective without performing worse in another. The set of all non-dominated solutions for a particular problem is called the *Pareto front* [26].

C. Decomposition of Static Multi-Objective Problems

A few approaches exist for transforming a multi-objective optimization problem into a set of scalar optimization problems. This process, called decomposition, collapses the multi-objective problem into a set of independent single-objective problems, which can then be minimized (or maximized) using any optimization method. It is advantageous to decompose the multi-objective problem because it allows for use of well-established and effective single-objective optimization strategies. We evaluate the weighted sum, Tchebychev, and boundary intersection methods.

1) Weighted Sum: The weighted sum approach solves a series of scalar optimization problems which are each a weighted sum of every objective function [26] [27]. This approach uses a weight vector $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$ where $\lambda_i \geq 0$ for all $i = 1, \ldots, m$. The weight vectors are normalized such that $\sum_{i=1}^{m} \lambda_i = 1$. Thus, the multi-objective problem becomes the following scalar optimization problem for every choice of $\boldsymbol{\lambda}$:

$$\begin{aligned} &minimize \quad g^{ws}(x|\boldsymbol{\lambda}) = \sum_{i=1}^{m} \lambda_i f_i(x) \\ &subject\ to \quad x \in \Omega \end{aligned} \tag{4}$$

2) Tchebychev: This approach uses the Tchebychev function to create a set of scalar optimization problems. The Tchebychev function penalizes the distance between a particular solution and a "reference point" for the problem, which represents an "ideal point" in objective space ($\min\{f_i(x)|x \in \Omega\}$ for all $i = 1, \ldots, m$) [26] [27]. This method again uses a set of normalized weight vectors $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$. The scalar optimization problem is then

$$\begin{aligned} &minimize \quad g^{te}(x|\boldsymbol{\lambda}, z^*) = \max_{i=1,\ldots,m} \{\lambda_i(|f_i(x) - z_i^*|)\} \\ &subject\ to \quad x \in \Omega \end{aligned} \tag{5}$$

where $z^* = (z_1^*, \ldots, z_m^*)^T$ is the reference point.

3) Boundary Intersection: The boundary intersection approach decomposes the multi-objective problem into a series of scalar optimization problems where the objective is to minimize the distance between a solution and a reference point in objective space, subject to the constraint that the solution lies along a particular line in objective space. This method again defines a series of weight vectors $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$, which represent different lines in objective space. We use the penalty-based boundary intersection formulation, which uses a soft constraint to encourage solutions to lie along the line defined by λ:

$$\begin{aligned} &minimize \quad g^{pbi}(x|\boldsymbol{\lambda}, z^*) = d_1 + \theta d_2 \\ &subject\ to \quad x \in \Omega \\ &\quad where \quad d_1 = \frac{||(F(x) - z^*)^T \boldsymbol{\lambda}||}{||\boldsymbol{\lambda}||} \\ &\qquad\qquad\quad d_2 = ||F(x) - (z^* + d_1 \boldsymbol{\lambda})||. \end{aligned} \tag{6}$$

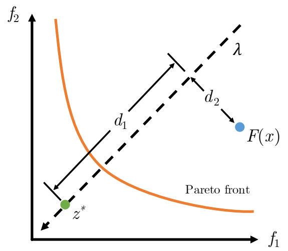

Fig. 2: Visual representation of penalty-based boundary intersection on a two objective problem. Modified from [27].

$F(x) = (f_1(x), \ldots, f_m(x))$ represents the solution in objective space, z^* is the reference point, θ is a user-defined penalty factor, and the values d_1 and d_2 are illustrated in Figure 2. Choosing an appropriate penalty factor θ is an important consideration in using the boundary intersection method [10]. For this work, we use $\theta = 100$, which was experimentally determined, evaluating by orders of magnitude.

Boundary intersection has significant advantages over both the weighted sum method and the Tchebychev method. In contrast to the weighted sum method, boundary intersection can model non-convex Pareto fronts [26]. Using the same set of weight vectors, the boundary intersection approach will generally give a more uniformly distributed set of solutions than the Tchebychev method [28]. Additionally, the weight vectors for boundary intersection have an interpretable meaning in objective space (see Figure 2), in contrast to the weight vectors used in the other two methods. Thus the weights for boundary intersection can be chosen in such a way that particular locations in objective space are "covered" by solutions. Given these advantages, we choose to incorporate the concepts of boundary intersection in our approach to the dynamic multi-objective optimization problem.

D. Dynamic Multi-Objective Optimization for Path Planning

Prior work ([27], [29]–[31]) utilizes the above decomposition methods for multi-objective trajectory planning in which the objectives do not change over time. However, our contribution is in *dynamic* multi-objective trajectory planning, which demands a modified approach. A dynamic multi-objective problem (DMOP), in which the objective functions can change over discrete time steps, is defined as

$$\begin{aligned} &\textit{minimize} \quad F(x,t) = [f_1(x,t), \ldots, f_M(x,t)] \\ &\textit{subject to} \quad x \in \Omega \end{aligned} \tag{7}$$

where t is the discrete time index and other variables are defined as in Equation 3. For our work, this means the objective maps that we wish to cover can change over time, due to measurements or changing environmental factors.

Methods for solving dynamic multi-objective problems differ in their approach from those solving static multi-objective problems. While static multi-objective problems attempt to model one unchanging Pareto front, dynamic multi-objective problems must constantly adapt to model a Pareto front that changes with time.

A few efforts have been made to apply dynamic multi-objective optimization to mobile robot path planning. Ajeil et al. use an algorithm inspired by echo-location of bats [32]. A similar work uses a different bio-inspired approach, modified aging ant colony optimization (AACO), to solve the trajectory planning problem in dynamic environments [33]. Gul et al. use yet another nature-inspired approach, the grey wolf optimizer-particle swarm optimization algorithm (PSO-GWO) [34]. These existing trajectory planning algorithms only address objectives of the path itself, like length, smoothness, and obstacle avoidance. Existing work only introduces a dynamic element in obstacles that can move, and do not consider other environmental factors that can change with time. There is a lack of work that focuses on mobile robot coverage of multiple dynamic objectives.

E. Dynamic Multi-Objective Ergodic Search

Our prior work presents a method for generating trajectories for a mobile robot that take into account multiple objective maps that change with time. The Dynamic Multi-Objective Ergodic Search (D-MO-ES) [30] method builds on static Multi-Objective Ergodic Search (MO-ES) [31] by predicting how best to construct a trajectory at each time step after changes occur in the environment, using relatively little prior information about how each objective map changes. However, this prediction method can degrade in effectiveness as objectives change significantly beyond what the method has examples for. D-MO-ES uses only a few examples to generate a prediction function, and once the maps change significantly from their original state, there is no guarantee that the prediction function remains accurate. Also, D-MO-ES needs example data in order to construct the prediction function, which is not feasible in many situations.

It is advantageous to use a method for dynamic multi-objective ergodic search that does not rely on prior data. In this work, we present a method for dynamic multi-objective ergodic search which uses the boundary intersection method to find a solution that balances each objective without the need for prior information.

III. Boundary Intersection-based Dynamic Multi-Objective Ergodic Search

Our approach to solving the problem of multi-objective ergodic search for dynamic information maps builds on the concept of boundary intersection decomposition. We design a method, boundary intersection-based dynamic multi-objective ergodic search (BI-MO-ES) that efficiently re-plans a trajectory each time an objective changes, while maintaining a Pareto optimal trade-off between each objective.

A. Ergodic Search Enhancements

We use ergodic search to plan trajectories to balance exploration and exploitation of information. We modify ergodic search for mobile robot exploration by optimizing directly for trajectory points in the spatial domain rather than optimizing for controls based on robot dynamics. This allows the function to generalize better to robots with different dynamics, and allows sample points along the trajectory to be more evenly distributed. To ensure even distribution, we add the following cost to the ergodic metric, which penalizes the distance between each neighboring trajectory point:

$$\text{gap cost} = \zeta \times \max_{0 \le n < N} \big((\gamma_x(n) - \gamma_x(n-1))^2 + (\gamma_y(n) - \gamma_y(n-1))^2 \big) \tag{8}$$

where $\zeta = 1$ is a user-chosen penalty factor. An additional term is also added to constrain the end point of the exploration, as it is often desirable that a robot stops at a particular location. We add a penalty on the distance from the end point to some desired point. Given a trajectory point $\gamma(n)$ and a desired point p, the penalty is

$$\text{point cost} = \alpha \times \big((\gamma_x(n) - p_x)^2 + (\gamma_y(n) - p_y)^2 \big) \tag{9}$$

where $\alpha = 10$ is a user-chosen penalty factor. Our method also uses a length budget to restrict the length of the trajectory, as a robot often will only have enough energy or time to traverse a particular distance. Given a budget length b, the cost on the trajectory length is calculated as follows:

$$\text{length cost} = \beta \times \Big| \sum_{n=0}^{N} (\gamma_x(n) - \gamma_x(n-1))^2 + (\gamma_y(n) - \gamma_y(n-1))^2) - b \Big| \tag{10}$$

where $\beta = 100$ is a user-chosen penalty factor. These additional terms ensure that the paths generated by our method and all baselines are the same length, have the same start and end location, and have similar spacing between trajectory points, allowing them to be compared fairly.

B. *Adapting Boundary Intersection for Dynamic Objectives*

Each time the environment changes, due to time-varying objectives or robot actions, our method will generate a single solution to the dynamic multi-objective problem. Although the boundary intersection method is designed to produce a *set* of possible trajectories for a *static* multi-objective problem, our work extends these concepts to the dynamic case. We make two innovations: 1) we adaptively determine a reference point as objectives change, and 2) we ensure a good trade-off between objectives by intelligently choosing where a solution lies in objective space.

1) Reference Point Calculation: Every time an objective changes, the "ideal point", or the best possible value of each objective, also changes. To produce a trajectory using boundary intersection, our method must calculate the reference point every time the environment changes, as follows: first, single-objective ergodic search is performed to obtain an optimal trajectory for each objective map individually. Performing single-objective ergodic search determines the best ergodic value possible to achieve on each individual objective, as considering an additional objective(s) will necessarily degrade performance in the other. The combination of the best ergodic values forms the reference point for these maps at this time step. Figure 3 visualizes these values.

For a three objective problem, for example, to obtain one solution from our method, we must plan four trajectories (three single-objective to find the reference point, one multi-objective to obtain the solution). This still results in a reduction in the number of solutions planned in contrast to approaches which attempt to cover the entire objective space.

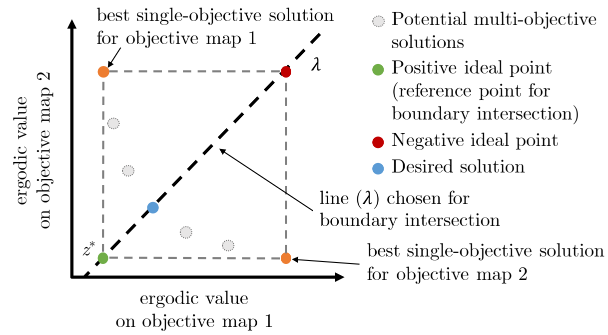

Fig. 3: Process for obtaining the reference point (z^*) and defining the desired solution line (λ) for our method. One single-objective solution is obtained for each objective, and the best objective values are combined to form the reference point. The desired solution line is chosen to lie in the middle of the space of possible solutions, ensuring a good trade-off.

2) Defining the Line in Objective Space: Each time objectives change, our method should generate a single solution to the multi-objective problem that effectively balances all objectives. Intuitively, a well-balanced solution will be one that lies close to the center of the Pareto front, or on the "knee" of the Pareto front curve. Knee points have been shown to be desirable solutions to multi-objective problems in the absence of other operator preferences, so they act as a good heuristic for quality solutions [35]–[39].

We use this concept of knee point solutions to inform our trajectory optimization. Using the boundary intersection method allows us to define a line in objective space (λ) that our desired solution should lie along. Our novel method defines a line that intersects the positive ideal point (best possible values of each objective, or the reference point) and the negative ideal point (worst values of each objective). This ensures the solution adapts to the changing environment, and will always lie in the "middle" of the feasible objective space, giving a good trade-off between each objective. An example for a two-objective case can be found in Figure 3, but this concept extends to more than two objectives. Defining this line in M-objective space to produce a high quality solution as objectives change is a novel contribution of our method, distinct from the normal use of boundary intersection, which plans a set of solutions for static objectives.

IV. Experiments

We perform three sets of experiments, each with different objectives, to test the utility of our approach. Trajectories are planned over 100 time steps (each dynamic objective changes every time step), with re-planning occurring every 10 time steps. We use 50 random start and goal positions to test performance over a variety of different scenarios.

A. *Objectives*

All our experiments use the entropy map formulation proposed by Candela et al., which models the uncertainty of

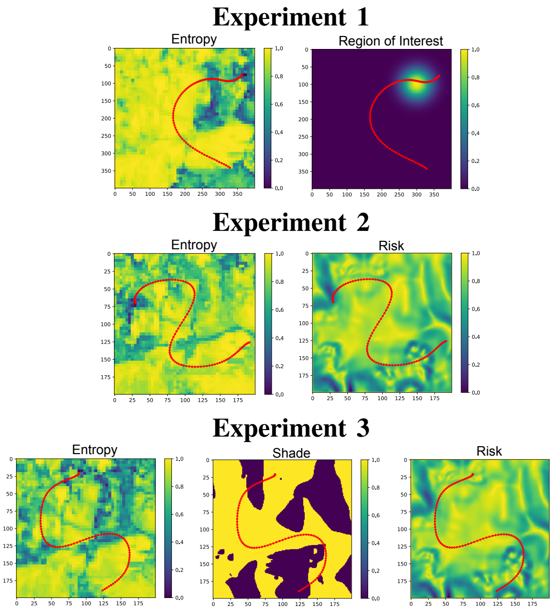

Fig. 4: Objective maps for all experiments with example planned trajectories. Entropy and shade are dynamic objectives while the region of interest map and risk are static.

scientific information over a region [40]. This dynamic map will change as the robot explores and collects geologic information. To evaluate performance on the entropy map, we measure the percent reduction in entropy over the trajectory. Our second dynamic objective is a map of shadowed regions over the same area as the entropy map. We assume a solar-powered robot should remain in sunny areas, so we evaluate performance based on the percentage of the trajectory in the sun. Our static objective maps are a simulated Gaussian map representing interest in a particular region, and a risk map constructed from the digital elevation model of the terrain, related to the slope at each pixel. Trajectory performance on both static objectives is evaluated based on the ergodic metric. Figure 4 shows which objectives are used for each of our three experiments.

B. Methods of Comparison

We compare our novel method with three other methods of dynamic multi-objective ergodic search: D-MO-ES ([30]), MO-ES + TOPSIS ([30], [31], bounded to planning five trajectories per replanning cycle), and the "equally weighted" method. MO-ES + TOPSIS plans a set of solutions at each time step and chooses the best one based on the TOPSIS method [41]. The equally weighted approach uses weighted sum decomposition to plan one solution each time step which weights each objective map equally.

C. Metrics of Evaluation

1) TOPSIS Closeness: There are several methods of choosing the "best" solution from a set of Pareto optimal solutions, including the technique for order of preference by similarity to ideal solution (TOPSIS) [41]. This technique is the most widely used and is an efficient and intuitive method of selecting a solution from a Pareto optimal set [42], [43].

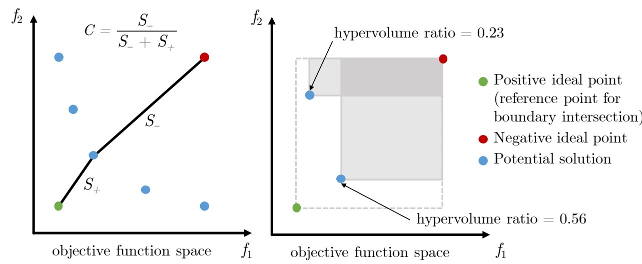

Fig. 5: Visualization of the TOPSIS closeness measure (left) and the hypervolume ratio metric (right).

The TOPSIS method chooses a solution that is closest to the positive ideal point (best possible value of each objective) and farthest from the negative ideal point (worst possible value of each objective) [41]. To do this, TOPSIS uses a "closeness" measure,

$$C = \frac{S_-}{S_- + S_+} \tag{11}$$

where S_- and S_+ are the Euclidean distances from the solution to the negative and positive ideal point, respectively (see Figure 5). The closeness metric is unit-less, so it can be used to compare two solutions with different ideal points. Since each time step in the dynamic multi-objective problem will have different ideal points, this metric allows us to compare trajectories across different time steps. A higher closeness measure indicates a better quality solution.

2) Hypervolume Ratio: Another measure of solution quality is the *hypervolume ratio*, which we have devised specifically for this work. The concept of hypervolume [44] is generally used to evaluate the quality of a *set* of solutions; a higher hypervolume means a solution set covers the objective space well. We adapt this concept to measure the quality of *one* solution, normalized based on the positive and negative ideal points for each time step. We measure the hypervolume (or area for two objectives) between a solution and the negative ideal point. We then measure the hypervolume between the positive ideal point and negative ideal point. The closer the ratio between these two values is to 1, the better the chosen solution is. A visual representation of this metric for two objectives can be found in Figure 5.

3) Planning Time: We measure the planning time of each trajectory to compare the efficiency between methods.

V. Results and Discussion

Results for all experiments can be found in Table I. Our method, boundary intersection multi-objective ergodic search (BI-MO-ES), performs on average 43% better in terms of the hypervolume ratio than the next best method, and 34% better in terms of TOPSIS closeness. We can calculate the p-values between all baselines and BI-MO-ES for these two metrics using the double sided Wilcoxon signed-rank test [45]. These p-values are all appreciably less than 0.01, meaning this result is statistically significant.

For the two-objective tests (Experiments 1 and 2), we see that in all metrics except one, BI-MO-ES outperforms all other methods tested. The weighted method outperforms our

Experiment 1: Two Objectives (Dynamic Entropy and Static Gaussian Map)

	% Reduction of Entropy ↑	Ergodic Metric for Gaussian Map ↓	Hypervolume Ratio ↑	Closeness to Ideal Point ↑	Time per Trajectory (s) ↓
Equally Weighted	**8.056** ± 3.150	0.356 ± 0.080	0.253 ± 0.374	0.420 ± 0.114	3.6 ± 2.3
MO-ES + TOPSIS	7.903 ± 3.587	0.149 ± 0.099	0.121 ± 0.147	0.603 ± 0.078	16.8 ± 5.3
D-MO-ES	8.027 ± 3.195	0.399 ± 0.081	0.049 ± 0.125	0.313 ± 0.156	**3.5** ± 2.1
BI-MO-ES	7.840 ± 4.041	**0.140** ± 0.036	**0.363** ± 0.102	**0.662** ± 0.086	12.1 ± 3.1

Experiment 2: Two Objectives (Dynamic Entropy and Static Risk)

	% Reduction of Entropy ↑	Ergodic Metric for Risk Map ↓	Hypervolume Ratio ↑	Closeness to Ideal Point ↑	Time per Trajectory (s) ↓
Equally Weighted	7.896 ± 3.899	0.033 ± 0.006	0.661 ± 0.466	0.536 ± 0.193	**3.3** ± 1.2
MO-ES + TOPSIS	7.911 ± 3.856	0.049 ± 0.010	0.605 ± 0.609	0.813 ± 0.128	16.7 ± 5.5
D-MO-ES	8.026 ± 3.327	0.042 ± 0.012	0.243 ± 0.479	0.560 ± 0.323	**3.3** ± 1.2
BI-MO-ES	**8.112** ± 3.674	**0.031** ± 0.007	**0.991** ± 0.149	**0.982** ± 0.051	14.2 ± 3.9

Experiment 3: Three Objectives (Dynamic Entropy, Dynamic Shade, and Static Risk)

	% Reduction of Entropy ↑	% of Trajectory in Sun ↑	Ergodic Metric for Risk Map ↓	Hypervolume Ratio ↑	Closeness to Ideal Point ↑	Time per Trajectory ↓
Equally Weighted	7.811 ± 4.588	**80.08** ± 9.297	**0.036** ± 0.008	0.338 ± 0.230	0.530 ± 0.207	**3.3** ± 1.2
MO-ES + TOPSIS	7.965 ± 3.624	78.94 ± 9.142	0.039 ± 0.011	0.398 ± 0.360	0.768 ± 0.130	11.6 ± 3.8
D-MO-ES	**7.997** ± 3.231	77.96 ± 9.312	0.049 ± 0.016	0.193 ± 0.331	0.558 ± 0.285	**3.3** ± 1.2
BI-MO-ES	7.965 ± 3.469	79.02 ± 8.753	0.038 ± 0.009	**0.538** ± 0.271	**0.793** ± 0.127	16.5 ± 5.0

TABLE I: Results from evaluation using both real and synthetic data. Our method is Boundary Intersection Multi-Objective Ergodic Search (BI-MO-ES). All values are averages taken over 50 trials of each method, ± standard deviations.

method in Experiment 1 in terms of entropy reduction, at the expense of performing poorly on the Gaussian map. In contrast, our method strikes a good balance between the two.

We also see that for Experiment 2, our method has an average hypervolume ratio and closeness measure very close to 1 (the highest possible value). The entropy and risk maps in this experiment are correlated, because they are derived from the same geologic features. That means there is not a large trade-off between trajectories that perform well on entropy versus those that perform well on risk. All the other methods tested (weighted, MO-ES + TOPSIS, and D-MO-ES) are based on the weighted sum method of decomposition. Thus we find empirically that our method, based on boundary intersection, is better at finding solutions close to the ideal point for multi-objective problems with correlated objectives than the weighted sum method.

For Experiment 3, our method does not outperform others when evaluated on individual objectives (entropy reduction, percent of trajectory in sun, and ergodic value of risk map). However, we see that BI-MO-ES has the best hypervolume ratio and closeness to the ideal point among all methods. Even though it does not win individually on any one objective, our method gives a solution that has the best trade-off between objectives, which is a higher quality solution overall.

As far as planning time, the BI-MO-ES plans a trajectory faster than MO-ES + TOPSIS on the two-objective experiments, but is consistently slower than the equally weighted and D-MO-ES methods. Because of the reference point calculation, BI-MO-ES performs $n+1$ trajectory optimizations: n optimizations to calculate the reference point, where n is the number of objectives, plus one to plan the solution trajectory. In contrast, the weighted method and D-MO-ES only plan one trajectory every step, so they are significantly faster. When using BI-MO-ES we incur a time penalty, but reap the benefits of a much better quality final solution. Future work will explore increasing the efficiency of the reference point calculation for BI-MO-ES.

VI. Conclusions

We show in this work that our boundary intersection-based ergodic search method for dynamic multi-objective trajectory optimization outperforms prior work in terms of several important metrics. Our method extends the concepts of decomposition of a multi-objective problem to the dynamic case, by intelligently optimizing for good quality solutions even as objectives change. Using our method, we consistently find a solution that is closer to the positive ideal point in objective space than other methods tested. This is important when planning for planetary rovers and other mobile robots, as we would like these robots to autonomously make intelligent movement decisions without the aid of human operators, under changing environmental conditions or with the collection of new information.

The BI-MO-ES method performs 34% better in terms of the TOPSIS closeness metric and 43% better in terms of hypervolume ratio than the next best approach. Much of the planning time increase for our method is due to the reference point calculation, which demands n trajectory optimizations for a problem with n objectives. In the future, it would be useful to formulate a method that eliminates the need to plan many reference trajectories in order to obtain a reference point, so this approach could produce high quality solutions more efficiently.

Acknowledgments

Thank you to Margaret Hansen and Ananya Rao for your ideas and suggestions on this work. This work was supported by NASA's LSITP program [80MSFC20C0008] and the AI Institute for Resilient Agriculture [NSF 024408B].

REFERENCES

[1] S. Grogan, R. Pellerin, and M. Gamache, "The use of unmanned aerial vehicles and drones in search and rescue operations–a survey," *Proceedings of the PROLOG*, 2018.

[2] D. S. Drew, "Multi-agent systems for search and rescue applications," *Current Robotics Reports*, vol. 2, no. 2, pp. 189–200, 2021.

[3] B. Shah and H. Choset, "Survey on urban search and rescue robots," *Journal of the Robotics Society of Japan*, vol. 22, no. 5, pp. 582–586, 2004.

[4] A. Rankin, M. Maimone, J. Biesiadecki, N. Patel, D. Levine, and O. Toupet, "Driving curiosity: Mars rover mobility trends during the first seven years," in *2020 IEEE Aerospace Conference*, 2020, pp. 1–19.

[5] J. Balaram and M. Golombek, "The ingenuity helicopter on the perseverance rover," *Space Science Reviews*, vol. 217, Jun. 2021.

[6] R. R Shamshiri, C. Weltzien, I. A. Hameed, *et al.*, "Research and development in agricultural robotics: A perspective of digital farming," 2018.

[7] S. Fountas, N. Mylonas, I. Malounas, E. Rodias, C. Hellmann Santos, and E. Pekkeriet, "Agricultural robotics for field operations," *Sensors*, vol. 20, no. 9, p. 2672, 2020.

[8] A. Candela, K. Edelson, M. Gierach, D. Thompson, G. Woodward, and D. Wettergreen, "Using remote sensing and in situ measurements for efficient mapping and optimal sampling of coral reefs," *Frontiers in Marine Science*, vol. 8, Sep. 2021.

[9] N. Cruz, N. Abreu, J. Almeida, *et al.*, "Cooperative deep water seafloor mapping with heterogeneous robotic platforms," in *OCEANS 2017-Anchorage*, IEEE, 2017, pp. 1–7.

[10] M. Ming, R. Wang, Y. Zha, and T. Zhang, "Pareto adaptive penalty-based boundary intersection method for multi-objective optimization," *Information Sciences*, vol. 414, pp. 158–174, 2017, ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2017.05.012.

[11] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013, ISSN: 0921-8890. DOI: https://doi.org/10.1016/j.robot.2013.09.004.

[12] J.-C. Latombe, *Robot motion planning*. Springer Science & Business Media, 2012, vol. 124.

[13] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," in *Field and service robotics*, Springer, 1998, pp. 203–209.

[14] E. U. Acar, H. Choset, A. A. Rizzi, P. N. Atkar, and D. Hull, "Morse decompositions for coverage tasks," *The International Journal of Robotics Research*, vol. 21, no. 4, pp. 331–344, 2002.

[15] A. Zelinsky, R. A. Jarvis, J. Byrne, S. Yuta, *et al.*, "Planning paths of complete coverage of an unstructured environment by a mobile robot," in *Proceedings of international conference on advanced robotics*, Citeseer, vol. 13, 1993, pp. 533–538.

[16] Y. Gabriely and E. Rimon, "Spiral-stc: An on-line coverage algorithm of grid environments by a mobile robot," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, IEEE, vol. 1, 2002, pp. 954–960.

[17] S. X. Yang and C. Luo, "A neural network approach to complete coverage path planning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 1, pp. 718–724, 2004.

[18] M. Popović, T. Vidal-Calleja, G. Hitz, *et al.*, "An informative path planning framework for uav-based terrain monitoring," en, *Autonomous Robots*, vol. 44, no. 6, pp. 889–911, Jul. 2020, ISSN: 1573-7527. DOI: 10.1007/s10514-020-09903-2.

[19] K. C. T. Vivaldini, V. Guizilini, M. D. C. Oliveira, T. H. Martinelli, D. F. Wolf, and F. Ramos, "Route planning for active classification with uavs," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 2563–2568. DOI: 10.1109/ICRA.2016.7487412.

[20] B. Charrow, S. Liu, V. Kumar, and N. Michael, "Information-theoretic mapping using cauchy-schwarz quadratic mutual information," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4791–4798. DOI: 10.1109/ICRA.2015.7139865.

[21] G. Hitz, E. Galceran, M.-È. Garneau, F. Pomerleau, and R. Siegwart, "Adaptive continuous-space informative path planning for online environmental monitoring," *Journal of Field Robotics*, vol. 34, no. 8, pp. 1427–1449, 2017.

[22] P. Lanillos, S. K. Gan, E. Besada-Portas, G. Pajares, and S. Sukkarieh, "Multi-uav target search using decentralized gradient-based negotiation with expected observation," *Information Sciences*, vol. 282, pp. 92–110, Oct. 2014, ISSN: 0020-0255. DOI: 10.1016/j.ins.2014.05.054.

[23] L. M. Miller, Y. Silverman, M. A. MacIver, and T. D. Murphey, "Ergodic exploration of distributed information," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 36–52, 2016. DOI: 10.1109/TRO.2015.2500441.

[24] E. Ayvali, H. Salman, and H. Choset, "Ergodic coverage in constrained environments using stochastic trajectory optimization," en, in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC: IEEE, Sep. 2017, pp. 5204–5210, ISBN: 978-1-5386-2682-5. DOI: 10.1109/IROS.2017.8206410. [Online]. Available: http://ieeexplore.ieee.org/document/8206410/.

[25] G. Mathew and I. Mezić, "Metrics for ergodicity and design of ergodic dynamics for multi-agent systems,"

Physica D: Nonlinear Phenomena, vol. 240, no. 4, pp. 432–442, 2011, ISSN: 0167-2789.

[26] K. Miettinen, *Nonlinear Multiobjective Optimization* (International Series in Operations Research & Management Science). Springer New York, NY, 1998, ISBN: 978-0-7923-8278-2. DOI: https://doi.org/10.1007/978-1-4615-5563-6.

[27] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, Dec. 2007, ISSN: 1941-0026. DOI: 10.1109/TEVC.2007.892759.

[28] I. Das and J. Dennis, "Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems," *SIAM Journal on Optimization*, vol. 8, Jul. 2000. DOI: 10.1137/S1052623496307510.

[29] D. Yi, M. Goodrich, and K. Seppi, "Morrf*: Sampling-based multi-objective motion planning," Jun. 2015.

[30] A. Rao, A. Breitfeld, A. Candela, B. Jensen, D. Wettergreen, and H. Choset, "Multi-objective ergodic search for dynamic information maps," in *Proceedings of (ICRA) International Conference on Robotics and Automation*, May 2023.

[31] Z. Ren, A. K. Srinivasan, H. Coffin, I. Abraham, and H. Choset, "A local optimization framework for multi-objective ergodic search," in *Robotics: Science and Systems XVIII*, Robotics: Science and Systems Foundation, Jun. 2022.

[32] F. H. Ajeil, I. K. Ibraheem, M. A. Sahib, and A. J. Humaidi, "Multi-objective path planning of an autonomous mobile robot using hybrid pso-mfb optimization algorithm," *Applied Soft Computing*, vol. 89, p. 106 076, 2020, ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2020.106076.

[33] F. H. Ajeil, I. K. Ibraheem, A. T. Azar, and A. J. Humaidi, "Grid-based mobile robot path planning using aging-based ant colony optimization algorithm in static and dynamic environments," *Sensors*, vol. 20, no. 7, p. 1880, 2020, ISSN: 1424-8220. DOI: 10.3390/s20071880.

[34] F. Gul, W. Rahiman, S. S. N. Alhady, A. Ali, I. Mir, and A. Jalil, "Meta-heuristic approach for solving multi-objective path planning for autonomous guided robot using pso–gwo optimization algorithm with evolutionary programming," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 7, pp. 7873–7890, 2020, ISSN: 1868-5137. DOI: 10.1007/s12652-020-02514-w.

[35] K. Deb and S. Gupta, "Understanding knee points in bicriteria problems and their implications as preferred solution principles," *Engineering Optimization*, vol. 43, no. 11, pp. 1175–1204, 2011. DOI: 10.1080/0305215X.2010.548863.

[36] J. Zou, Q. Li, S. Yang, H. Bai, and J. Zheng, "A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization," *Applied Soft Computing*, vol. 61, pp. 806–818, Dec. 2017, ISSN: 1568-4946. DOI: 10.1016/j.asoc.2017.08.004.

[37] X. Zhang, Y. Tian, and Y. Jin, "A knee point-driven evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, pp. 761–776, 2015. DOI: 10.1109/TEVC.2014.2378512.

[38] F. Zou, G. G. Yen, and L. Tang, "A knee-guided prediction approach for dynamic multi-objective optimization," *Information Sciences*, vol. 509, pp. 193–209, 2020, ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2019.09.016.

[39] F. Zou, G. G. Yen, L. Tang, and C. Wang, "A reinforcement learning approach for dynamic multi-objective optimization," *Information Sciences*, vol. 546, pp. 815–834, Feb. 2021, ISSN: 0020-0255. DOI: 10.1016/j.ins.2020.08.101.

[40] A. Candela Garza, "Bayesian models for science-driven robotic exploration," Ph.D. dissertation, Carnegie Mellon University, 2021.

[41] C. Hwang and K. Yoon, *Multiple Attribute Decision Making: Methods and Applications : a State-of-the-art Survey* (Lecture notes in economics and mathematical systems). Springer-Verlag, 1981, ISBN: 9783540105589.

[42] Z. Wang and G. Rangaiah, "Application and analysis of methods for selecting an optimal solution from the pareto-optimal front obtained by multi-objective optimization," *Industrial & Engineering Chemistry Research*, vol. 56, Dec. 2016.

[43] M. Behzadian, S. Khanmohammadi Otaghsara, M. Yazdani, and J. Ignatius, "A state-of the-art survey of topsis applications," *Expert Systems with Applications*, vol. 39, no. 17, pp. 13 051–13 069, 2012, ISSN: 0957-4174.

[44] A. P. Guerreiro, C. M. Fonseca, and L. Paquete, "The hypervolume indicator: Computational problems and algorithms," *ACM Comput. Surv.*, vol. 54, no. 6, Jun. 2021, ISSN: 0360-0300. DOI: 10.1145/3453474.

[45] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945, ISSN: 00994987.