

Terrain-Aware Model Predictive Control of Heterogeneous Bipedal and Aerial Robot Coordination for Search and Rescue Tasks

Abdulaziz Shamsah^{1,2}, Jesse Jiang³, Ziwon Yoon¹, Samuel Coogan³, and Ye Zhao¹

Abstract—Humanoid robots offer significant advantages for search and rescue tasks, thanks to their capability to traverse rough terrains and perform transportation tasks. In this study, we present a task and motion planning framework for search and rescue operations using a heterogeneous robot team composed of humanoids and aerial robots. We propose a terrain-aware Model Predictive Controller (MPC) that incorporates terrain elevation gradients learned using Gaussian processes (GP). This terrain-aware MPC generates safe navigation paths for the bipedal robots to traverse rough terrain while minimizing terrain slopes, and it directs the quadrotors to perform aerial search and mapping tasks. The rescue subjects’ locations are estimated by a target belief GP, which is updated online during the map exploration. A high-level planner for task allocation is designed by encoding the navigation tasks using syntactically cosafe Linear Temporal Logic (scLTL), and a consensus-based algorithm is designed for task assignment of individual robots. We evaluate the efficacy of our planning framework in simulation in an uncertain environment with various terrains and random rescue subject placements.

I. INTRODUCTION

Humanoid robots demonstrate unique capabilities to traverse over rough and sloped terrains [1]–[5] while performing manipulation and payload transportation tasks [6], [7]. Such capabilities are essential for search and rescue missions. However, humanoids suffer from locomotion instability in extreme and uncertain environments, limited field of view for searching, and inferior agility compared to quadrotors [8], [9]. To this end, we propose a multi-robot task and motion planning framework (see Fig. 1) that leverages the capabilities of teamed-up bipedal and aerial robots for search and rescue missions. To our knowledge, this work presents the first framework that incorporates humanoids as part of a heterogeneous robot team in search and rescue tasks.

For humanoid locomotion, numerous studies have investigated navigation on sloped terrains [1]–[3], [10], often focusing on the avoidance of sloped terrains or finding planar footholds. The work in [1], closely related to the work we propose here, uses a piecewise linear terrain approximation for computing foot placements for a linear inverted pendulum model [11]. In [1], terrain information is assumed to be known or approximated by the operator, whereas in our work we employ a Gaussian process (GP) [12], [13] map

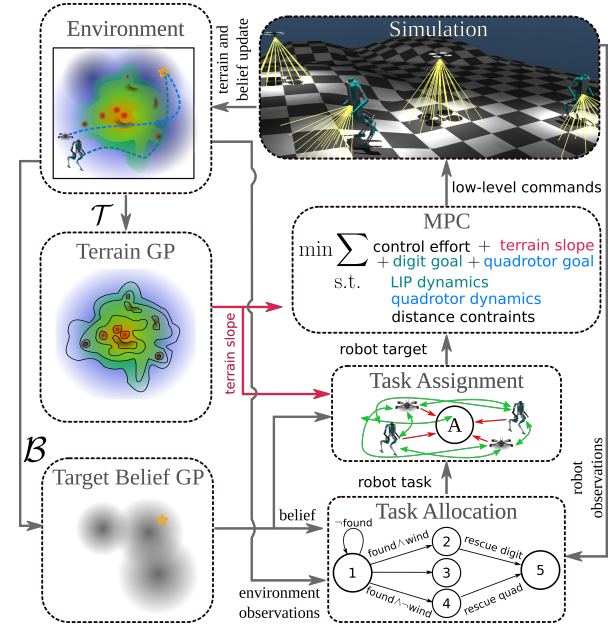


Fig. 1: Overall block diagram of the proposed heterogeneous bipedal and aerial robot planner for search and rescue in terrain-uncertain environments.

to learn terrain elevation. Given this, our framework enables the Model Predictive Controller (MPC) to compute optimal paths that allow the humanoid to traverse sloped terrain while minimizing terrain slopes in a cost function. Such capability of traversing sloped terrain generalizes our previous works on integrating terrain GPs with locomotion, which have focused on avoiding uncertain or elevated terrains [14], [15].

Multi-robot task allocation (MRTA) has been widely explored in the community using centralized [16] or decentralized [17], [18] communication. Such an MRTA problem is commonly solved based on three main methods: market-based, behavior-based, and optimization-based approaches [16], [17], [19]. To the best of our knowledge, there have been very few works leveraging the heterogeneous robot capabilities of humanoids and drones [20]–[22]. In the proposed heterogeneous robot teaming problem, task allocation requires matching of robot capabilities with assigned tasks, which makes market-based approaches ideal [23]. Therefore, we use consensus-based methods [18], [24], [25], as they strike a balance between optimality and efficiency. As opposed to a discretized task space, where conflict may arise from assigning the same task to multiple robots [18], our task requires coordinated exploration of a continuous map. To this end, we propose a new conflict resolution scheme tailored towards removing redundant map exploration.

Formal control methods such as Linear Temporal Logic

¹Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30313, USA. (e-mail: ashamsah3@gatech.edu, zyoona6@gatech.edu, ye.zhao@me.gatech.edu)

²Mechanical Engineering Department, College of Engineering and Petroleum, Kuwait University, PO Box 5969, Safat, 13060, Kuwait

³School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: jjiang@gatech.edu, sam.coogan@gatech.edu).

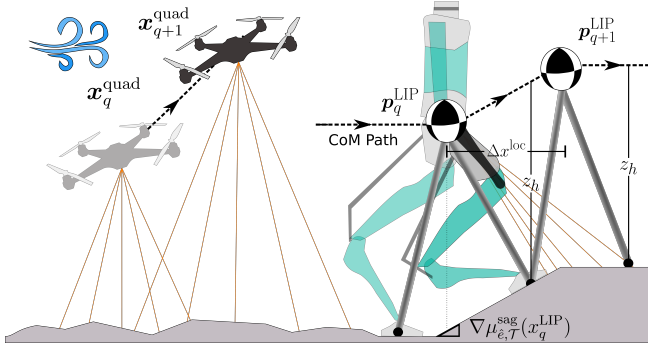


Fig. 2: Illustration of the LIP model for bipedal locomotion and the quadrotor model for two consecutive discrete states. The orange lines indicate the range sensors attached to the robots and an illustration of the corresponding local field of view for measuring the terrain data.

(LTL) and its variants have been commonly used to formally specify multi-robot tasks [26]–[28]. A major advantage of LTL-based planning approaches is that they give formal guarantees on task satisfaction. However, most LTL planning works consider scenarios in which the targeted locations are static and known *a priori*. For scenarios in which target locations must be dynamically updated at runtime, such as what we consider in this work, the literature [29] uses an exploration policy in concert with the task planner.

In this study, we present a hierarchical approach to bipedal and aerial multi-robot motion planning for search and rescue (see Fig. 1), integrating a task allocation planner at the high level (Sec V-B), consensus-based task assignment module at the middle level (Sec VI), and a multi-robot terrain-aware MPC at the low level (Sec. IV). The core contributions are as follows: (1) Integration of a terrain GP into an MPC that solves the optimal paths for the multi-robot team while maximizing the traversability for the bipedal robots, (2) Consensus-based task assignment module that assigns targets to the heterogeneous multi-robot team to match the capabilities of each robot, and (3) an scLTL task allocation module that dynamically allocates high-level tasks to the robot team based on current low-level robot observations of the environment and the progress of task completion.

II. PROBLEM FORMULATION

A. Task and Environment Setup

We task a team of bipedal robots and quadrotors with a search and rescue mission in an environment with partially observable and uneven terrain. The task is to search for subjects and deliver them to a safe location known *a priori*.

We assume that an initial set of terrain elevation data points $\mathcal{T} = \{(\mathbf{p}_1, e_1), (\mathbf{p}_2, e_2), \dots, (\mathbf{p}_n, e_n)\}$ is known *a priori*, where $\mathbf{p} = (x, y)$ is the 2-D position of the data point and e is the corresponding terrain elevation. We have initial belief points $\mathcal{B} = \{(\mathbf{p}_1^s, c_1^s), (\mathbf{p}_2^s, c_2^s), \dots, (\mathbf{p}_m^s, c_m^s)\}$ corresponding to the 2-D Euclidean positions of the subject location $\mathbf{p}^s = (x^s, y^s)$ and the corresponding belief confidence level c^s that the subject is located at this point.

B. Bipedal Robot Walking Model

We leverage a locomotion-specific reduced order model (ROM) to describe our Digit bipedal robot dynamics in this

study. The ROM used to design the 3-D walking motion is the linear inverted pendulum (LIP) model [11]. For the LIP model we assume that each walking step has a fixed time duration D [1], [30]–[32]. Then, we build our model on the discrete sagittal dynamics¹ ($\Delta x_q^{\text{loc}}, v_q^{\text{loc}}$), where x_q^{loc} is the CoM position at the beginning of the q^{th} step, $\Delta x_q^{\text{loc}} = x_{q+1}^{\text{loc}} - x_q^{\text{loc}}$ is the local sagittal CoM position difference between two consecutive walking steps², and v_q^{loc} is the sagittal velocity at the local coordinate for the q^{th} walking step, similar to [32]. The 3-D LIP dynamics become:

$$x_{q+1}^{\text{LIP}} = x_q^{\text{LIP}} + \Delta x_q^{\text{loc}}(u_q^f) \cos(\theta_q) \quad (1a)$$

$$y_{q+1}^{\text{LIP}} = y_q^{\text{LIP}} + \Delta x_q^{\text{loc}}(u_q^f) \sin(\theta_q) \quad (1b)$$

$$z_{q+1}^{\text{LIP}} = z_q^{\text{LIP}} + \nabla \mu_{e,T}^{\text{sag}}(x_q^{\text{LIP}}) \Delta x_q^{\text{loc}} \quad (1c)$$

$$v_{q+1}^{\text{loc}} = \cosh(\beta D) v_q^{\text{loc}} - \beta \sinh(\beta D) u_q^f \quad (1d)$$

$$\theta_{q+1}^{\text{LIP}} = \theta_q + u_q^{\Delta\theta} \quad (1e)$$

where u_q^f is the sagittal foot position relative to the CoM, $u_q^{\Delta\theta}$ is the heading angle change, $\beta = \sqrt{g/z_h}$, where g is the gravitational constant, z_h is the constant CoM height at apex state³ (see Fig. 2), $\nabla \mu_{e,T}^{\text{sag}}(x_q^{\text{LIP}})$ is the terrain slope at the current step (see Fig. 2), and θ is the heading. For notational simplicity, (1) will hereafter be referred to as:

$$\mathbf{x}_{q+1}^{\text{LIP}} = \Phi^{\text{LIP}}(\mathbf{x}_q^{\text{LIP}}, \mathbf{u}_q^{\text{LIP}}) \quad (2)$$

where $\mathbf{x}_q^{\text{LIP}} = (p_q^{\text{LIP}}, v_q^{\text{loc}}, \theta_q^{\text{LIP}})$ and $p_q^{\text{LIP}} = (x_q^{\text{LIP}}, y_q^{\text{LIP}}, z_q^{\text{LIP}})$ is the 3-D location in the world coordinate at step q . The control input is $\mathbf{u}_q^{\text{LIP}} = (u_q^f, u_q^{\Delta\theta})$. A detailed derivation of (1) is in [32].

C. Quadrotor Model

We use a 10-D quadrotor model assuming near-hover conditions (*i.e.*, small pitch and roll angles) [33], [34] and consider the discrete dynamics as:

$$\mathbf{x}_{q+1}^{\text{quad}} = \Phi^{\text{quad}}(\mathbf{x}_q^{\text{quad}}, \mathbf{u}_q^{\text{quad}}) \quad (3)$$

where $\mathbf{x}_q^{\text{quad}} = (p_q^{\text{quad}}, v_q^{\text{quad}}, \theta_q^{\text{quad}}, \omega_q)$, $p_q^{\text{quad}} = (x_q^{\text{quad}}, y_q^{\text{quad}}, z_q^{\text{quad}})$ denotes the position at the q^{th} time step, $v_q^{\text{quad}} = (v_{x,q}^{\text{quad}}, v_{y,q}^{\text{quad}}, v_{z,q}^{\text{quad}})$ are velocities, $\theta_q^{\text{quad}} = (\theta_x^{\text{quad}}, \theta_y^{\text{quad}})$ are pitch and roll, and $\omega_q = (\omega_x, \omega_y, \omega_z)$ are pitch and roll rates. The control input $\mathbf{u}_q^{\text{quad}} = (\alpha_{x,q}, \alpha_{y,q}, \alpha_{z,q})$ is the desired pitch and roll ($\alpha_{x,q}, \alpha_{y,q}$) and the vertical thrust $\alpha_{z,q}$. The quadrotor dynamics are discretized with the same step time D as the LIP model. We refer to [34] for a detailed description of the dynamics.

D. Problem Statement

Problem II.1. *Given the bipedal robot and quadrotor dynamics in Sec. II, along with terrain elevation data $\mathcal{T}$ and belief points $\mathcal{B}$, design a task allocation and assignment*

¹The lateral dynamics are only considered in the ALIP model at the low level since they are periodic with a constant desired lateral foot placement.

²The robot model used in our study represent step-by-step dynamics, *i.e.*, $\mathbf{x}_q^{\text{LIP}}$ and $\mathbf{x}_{q+1}^{\text{LIP}}$ are the CoM state at the foot contact switching instant of two consecutive walking steps.

³Apex state is the state when the CoM is directly on top of the foot.

algorithm to search for and rescue the subjects S_i , $\forall i \in K$. Synthesize an optimal motion plan which leverages the distinct capabilities of the robots while ensuring safety.

III. GAUSSIAN PROCESS LEARNING OF TERRAIN AND TARGET BELIEF

A. Gaussian Processes

To learn the terrain and belief uncertainties present in our environment, we use Gaussian process (GP) regression:

Definition III.1 (Gaussian Process Regression). *Gaussian Process (GP) regression models a function $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ as a distribution with covariance $\kappa : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}_{>0}$. Assume a dataset of m samples $\mathcal{D} = \{(\xi^j, \lambda_i^j)\}_{j \in \{1, \dots, m\}}$, where $\xi^j \in \mathbb{R}^n$ is the input and λ_i^j is an observation of $g_i(\xi^j)$ under Gaussian noise with variance $\sigma_{\nu_i}^2$. Let $K \in \mathbb{R}^{m \times m}$ be a kernel matrix defined elementwise by $K_{j\ell} = \kappa(\xi^j, \xi^\ell)$ and for $\xi \in \mathbb{R}^n$, let $k(\xi) = [\kappa(\xi, \xi^1) \ \kappa(\xi, \xi^2) \ \dots \ \kappa(\xi, \xi^m)]^T \in \mathbb{R}^m$. Then, the predictive distribution of g_i at a test point ξ is Gaussian with mean $\mu_{g_i, \mathcal{D}}$ and variance $\sigma_{g_i, \mathcal{D}}^2$ given by*

$$\begin{aligned} \mu_{g_i, \mathcal{D}}(\xi) &= k(\xi)^T (K + \sigma_{\nu_i}^2 I_m)^{-1} \Lambda \\ \sigma_{g_i, \mathcal{D}}^2(\xi) &= \kappa(\xi, \xi) - k(\xi)^T (K + \sigma_{\nu_i}^2 I_m)^{-1} k(\xi), \end{aligned}$$

where I_m is the identity and $\Lambda = [\lambda_i^1 \ \lambda_i^2 \ \dots \ \lambda_i^m]^T$.

In this work, we use the radial basis function (RBF) kernel

$$\kappa(\xi^i, \xi^j) = \sigma_f^2 \exp(-\|\xi^i - \xi^j\|^2 / 2\ell^2), \quad (4)$$

where σ_f^2 and ℓ are the signal variance and lengthscale hyperparameters, respectively. We will also use the partial derivative [35] with respect to $\xi_{i'}^i$, the dimension i' of ξ^i :

$$\frac{\partial \kappa(\xi^i, \xi^j)}{\partial \xi_{i'}^i} = \frac{-\sigma_f^2}{\ell^2} (\xi_{i'}^i - \xi_{i'}^j) \kappa(\xi^i, \xi^j). \quad (5)$$

B. Terrain GP for Locomotion Safety

We use the terrain elevation data $\mathcal{T} = \{(\mathbf{p}_i, e_i)\}_{i=1}^n$ to train a terrain GP $\hat{e}(\mathbf{p})$ for which the input is a global location (x, y) and the output is the predicted terrain height at that location, $\hat{e}$. Given $\hat{e}(\mathbf{p})$ and assuming no sensor noise, the mean of the terrain elevation at some test point $\mathbf{p}^t$ is

$$\mu_{\hat{e}, \mathcal{T}}(\mathbf{p}^t) = k(\mathbf{p}^t)^T K^{-1} E, \quad (6)$$

where $E = [e_1 \ e_2 \ \dots \ e_n]^T$ is the elevation data. As described in [12, Chap. 9], given the terrain elevation data $\mathcal{T}$ we can construct a slope GP by taking the gradients of (6) with respect to the test point in the x and y dimensions:

$$\nabla \mu_{\hat{e}, \mathcal{T}, \{x, y\}}(\mathbf{p}^t) = \nabla_{\mathbf{p}^t, \{x, y\}} k(\mathbf{p}^t)^T K^{-1} E, \quad (7)$$

where $\nabla_{\mathbf{p}^t, \{x, y\}} k(\mathbf{p}^t)$ is a vector of the partial derivative of $k(\mathbf{p}^t)$ with respect to $\mathbf{p}^t$, constructed by applying (5) elementwise with respect to either the x or y dimension.

We then transform the gradients to the sagittal and lateral directions relative to the current robot heading:

$$\begin{bmatrix} \nabla \mu_{\hat{e}, \mathcal{T}}^{\text{sag}}, \nabla \mu_{\hat{e}, \mathcal{T}}^{\text{lat}} \end{bmatrix}^T = R(\theta^{\text{LIP}}) \cdot [\nabla \mu_{\hat{e}, \mathcal{T}, x}, \nabla \mu_{\hat{e}, \mathcal{T}, y}]^T \quad (8)$$

where $R(\theta^{\text{LIP}})$ is a rotation matrix based on the LIP modeling heading angle. In Sec. IV, (8) will be minimized in our proposed MPC to encourage the walking robot to choose a path with low terrain slopes.

C. Belief GP for Search and Rescue Task

We use the initial belief data $\mathcal{B} = \{(\mathbf{p}_i^s, c_i^s)\}_{i=1}^m$ to generate training points $\hat{\mathcal{B}} = \{\hat{\mathcal{B}}_i\}_{i=1}^m$, where $\hat{\mathcal{B}}_i$ is a set of l points sampled from the normal distribution $\{\mathcal{N}(\mathbf{p}_i^s, c_i^{s^2})\}$. We then train a belief GP $\hat{b}(\mathbf{p})$ for which the input is a global potential location (x, y) of the subject and the output is the predicted belief at that location, $\hat{b}$. We solve for the mean of the belief value at point $\mathbf{p}^t$ as $\mu_{\hat{b}, \mathcal{B}}(\mathbf{p}^t) = k(\mathbf{p}^t)^T K^{-1} \hat{\mathcal{B}}$.

IV. MULTI-ROBOT TERRAIN-AWARE MPC

In this section, we design a terrain-aware motion planner for the heterogeneous multi-robot team. Compared to our previous works on integrating terrain GPs with locomotion planners, the objective of this planner is not to avoid terrains with high terrain uncertainty [14] or elevations [15]. Instead, we aim to generate safe paths for the bipedal robot to traverse rough terrain while minimizing the lateral slopes of the planned paths using gradient-based trajectory optimization.

To this end, we integrate lateral slopes derived from the terrain GP in (8) into the cost function in our proposed MPC framework. The terrain GP learns a smooth, differentiable, and continuous approximation of the rough terrain. This makes it an ideal candidate for gradient-based motion planning methods as proposed here:

$$\begin{aligned} \min_{X, U} \quad & \sum_{q=0}^{N-1} J(\mathbf{x}_q^{\text{LIP}}, \mathbf{x}_q^{\text{quad}}, \mathbf{u}_q^{\text{LIP}}, \mathbf{u}_q^{\text{quad}}) + \nabla \mu_{\hat{e}, \mathcal{T}}^{\text{lat}}(\mathbf{p}_q^{\text{LIP}})^2 \\ \text{s.t.} \quad & \mathbf{x}_{q+1}^{\text{LIP}} = \Phi^{\text{LIP}}(\mathbf{x}_q^{\text{LIP}}, \mathbf{u}_q^{\text{LIP}}) \\ & \mathbf{x}_{q+1}^{\text{quad}} = \Phi^{\text{quad}}(\mathbf{x}_q^{\text{quad}}, \mathbf{u}_q^{\text{quad}}) \\ & \mathbf{x}_0^{\text{LIP}} = \mathbf{x}_{\text{init}}^{\text{LIP}}, (\mathbf{x}_q^{\text{LIP}}, \mathbf{u}_q^{\text{LIP}}) \in \mathcal{XU}^{\text{LIP}} \\ & \mathbf{x}_0^{\text{quad}} = \mathbf{x}_{\text{init}}^{\text{quad}}, (\mathbf{x}_q^{\text{quad}}, \mathbf{u}_q^{\text{quad}}) \in \mathcal{XU}^{\text{quad}} \\ & d_l - \epsilon \leq d(\mathbf{p}_q^{\text{LIP1}}, \mathbf{p}_q^{\text{LIP2}}) \leq d_u + \epsilon \end{aligned}$$

where $J(\mathbf{x}, \mathbf{u})$ is a cost that penalizes the distance between the current CoM state of each robot and its corresponding target location and $\nabla \mu_{\hat{e}, \mathcal{T}}^{\text{lat}}(\mathbf{p}_q^{\text{LIP}})^2$ is used to minimize the lateral slope of the path that the LIP model takes to reach the target location. X is the state vector of all the robots $(\mathbf{x}^{\text{LIP}}, \mathbf{x}^{\text{quad}})^4$, and U is the control vector $(\mathbf{u}^{\text{LIP}}, \mathbf{u}^{\text{quad}})$. $\mathcal{XU}^{\text{LIP}}$ and $\mathcal{XU}^{\text{quad}}$ are linear constraints on the states and control inputs for the robots. For the legged robots not to collide with each other, we enforce a distance constraint between the robots, where $d(\mathbf{p}_q^{\text{LIP1}}, \mathbf{p}_q^{\text{LIP2}})$ is the Euclidean distance between the two robots, and (d_l, d_u) are upper and lower bounds, respectively, on the allowed distance⁵. This MPC is used as the underlying motion planner for all the tasks detailed in Sec. VI. The MPC formulation is general

⁴Note that the superscripts $(\cdot)^{\text{LIP}}$ and $(\cdot)^{\text{quad}}$ refer to both humanoid LIP models and both quadrupeds unless otherwise noted.

⁵The upper bound constraint is only active during bipedal robot rescue tasks (see Sec. VI-B).

and can be generalized to simultaneously solve the paths for a team of multiple humanoids and quadrotors.

V. TASK PLANNER

A. Task Specification

In this work, we use the semantics of syntactically cosafe Linear Temporal Logic (scLTL) to define tasks.

Definition V.1 (Syntactically co-safe LTL [36, Def. 2.3]). A syntactically co-safe linear temporal logic (scLTL) formula ϕ over a set of observations O is recursively defined as

$$\phi = \top \mid o \mid \neg o \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \bigcirc \phi \mid \phi_1 \mathcal{U} \phi_2 \mid \Diamond \phi \mid \phi_1 \rightarrow \phi_2$$

where $o \in O$ is an observation and ϕ , ϕ_1 , and ϕ_2 are scLTL formulas. We define the next operator $\bigcirc$ as meaning that ϕ will be satisfied in the next state transition, the until operator $\mathcal{U}$ as meaning that the system satisfies ϕ_1 until it satisfies ϕ_2 , the eventually operator $\Diamond$ as $\top \mathcal{U} \phi$, and the implication operator $\rightarrow$ as $\neg \phi_1 \vee \phi_2$.

Satisfaction of scLTL is checked with finite state automata.

Definition V.2 (Finite State Automaton [36]). A finite state automaton (FSA) is a tuple $\mathcal{A} = (S, s_0, O, \delta, F)$, where

- S is a finite set of states,
- $s_0 \in S$ is the initial state,
- O is the input alphabet,
- $\delta : S \times O \rightarrow S$ is a transition function, and
- $F \subseteq S$ is the set of accepting (final) states.

We define the observation space O as a combination of robot- and environment-centric observations.

Definition V.3 (Robot-centric Observations). Robot-centric observations O_R solely depend on actions taken by robotic agents, where we have:

- O_{R1} : Found subject n ,
- O_{R2} : Rescued subject n with a quadcopter,
- O_{R3} : Rescued subject n with a bipedal robot,

Definition V.4 (Environment-centric Observations). Environment-centric observations O_E depend on the state of an environmental condition, where we have:

- O_{E1} : Wind - quadcopter unable to rescue a subject
- O_{E2} : Untraversable terrain - bipedal robot unable to rescue a subject

We define an individual scLTL specification for each subject requiring a rescue. Thus, for a scenario with N subjects we concurrently use N FSAs.

B. Multi-robot Task Allocation

We now detail a high-level planning and task allocation policy. We first define robot tasks.

Definition V.5 (Robot Task). A robot task is a tuple $\Gamma_i = (\underline{n}, \bar{n}, \text{NAME}, \text{ROBOT})$, where $\underline{n}, \bar{n}$ are lower and upper bounds, respectively, on the number of agents required for the task, NAME denotes the task name, and ROBOT denotes the types of robots eligible for the task. In this work, we have the set Γ of robot tasks

- (Search) $\Gamma_1: (1, \infty, \text{SEARCH}, \{\text{QUADROTOR}, \text{BIPEDAL}\})$,
- (Quadrotor rescued) $\Gamma_2: (1, 1, \text{RESCUE}, \text{QUADROTOR})$,
- (Bipedal robot rescued) $\Gamma_3: (2, 2, \text{RESCUE}, \text{BIPEDAL})$,
- (Mapping terrain) $\Gamma_4: (1, 1, \text{MAPPING}, \text{QUADROTOR})$.

Then, we create a mapping $\mathcal{M} : O_R \rightarrow \Gamma$ from robot-centric observations to a set of robot tasks. In this work, we use the following mappings

$$O_{R1} \rightarrow \Gamma_1, O_{R2} \rightarrow \Gamma_2, O_{R3} \rightarrow \{\Gamma_3, \Gamma_4\}$$

We now detail the entire planning framework. We first define a scLTL specification for each mission subject and generate a corresponding FSA. We assume that at each state in any FSA, there is only one robot-centric observation available. For each FSA representing a subject, we query the robot-centric observation available at the current state and map it to its corresponding robot tasks. These are then sent to a task assignment algorithm detailed in Section VI. The agents execute their assignments until a robot-centric observation is returned from any of the agents. The corresponding FSA is then updated to a new state based on the robot- and environment-centric observations. Finally, tasks are reassigned for all agents.

VI. SEARCH AND RESCUE TASK ASSIGNMENT

The proposed search and rescue problem consists of three subtasks (search, rescue, and mapping), as discussed in the previous section, that guide the robot team in successfully transporting the subjects to a safe location R , known *a priori*. We now detail the metrics used to assign these subtasks.

A. Search Task

For the search task, we choose a consensus-based auction algorithm [18] as it balances computational efficiency and solution quality (*i.e.*, optimality) effectively, particularly when considering heterogeneous robot capabilities [23]. Auction-based methods, although computationally efficient, can lead to suboptimal task allocations, especially in scenarios with complex dependencies and path conflicts [16]. Consensus-based methods leverage the initial efficiency of auction-based allocation to quickly assign tasks based on overall scores, and then employ a consensus phase to iteratively resolve task conflicts [18]. In our case, task conflict arises when the planned paths for robots to complete their assigned task intersect, resulting in redundant environmental exploration. Here, instead of using a task bundle similar to the traditional consensus-based methods [18], made up of all the intermediate targets explored along the path, we use line segment intersection to indicate task conflict.

1) *Auction Phase*: In this phase each robot "bids" on the available tasks (targets in the environment to search) based on its own capabilities [16]. We consider a set of T uniformly distributed candidate target points $\{t_i \in \mathbb{R}^2\}_{i=1}^T$ as shown by the $\star$ in Fig. 3(a). In the proposed scenario, the robots score the targets based on the belief value, the traversability, and the time it takes to reach the target.

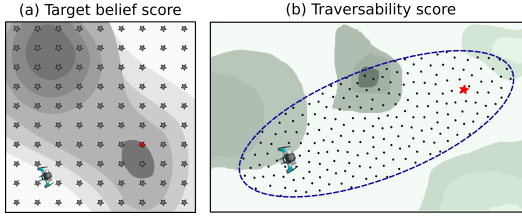


Fig. 3: (a) shows the candidate target points $\mathbf{t}$ as $\star$, based on the belief GP of the environment shown as the gray-white gradients. (b) shows the sample points for the calculation of the traversability score $\mathbf{p}_i^{\text{ellipse}}$, $i \in M$, as the black dots inside the ellipse. The ellipse focal points are the current position of the robot and the candidate target point shown as a red $\star$. The terrain GP of the environment is shown by the green-white gradient.

a) *Belief score* $\mathcal{S}_b$: represents the probability that the subject is located at a specific point. This score is derived from the belief GP model $\hat{b}(\mathbf{p})$, which predicts the likelihood of the subject's presence based on observed data. Mathematically, the belief score for a target $\mathbf{t}_i$ is given by the maximum upper confidence bound:

$$\mathcal{S}_b(\mathbf{t}_i) = \mu_{\hat{b},\mathcal{B}}(\mathbf{t}_i) + \alpha_b \sigma_{\hat{b},\mathcal{B}}(\mathbf{t}_i), \forall i \in [1, T] \quad (10)$$

where $\sigma_{\hat{b},\mathcal{B}}(\mathbf{t}_i)$ is the variance of the belief and α_b is a parameter that balances the mean and variance.

b) *Traversability score* ($\mathcal{S}_t$): evaluates the ease of traversing the terrain between the robot's current position and the target point. The traversability is evaluated based on the upper confidence bound of the terrain slope estimation at the sampled points. The work in [37] uses GP integrals to evaluate the traversability between two points. While that method provides a continuous evaluation, it is computationally expensive. As the proposed motion planner in Sec. IV will solve for an optimal path with high traversability (i.e., minimum lateral slopes), a continuous estimation of the slopes is not required, and sampling the maximum upper confidence bound of a set of randomly selected discrete points is sufficient to estimate the traversability score.

To compute the traversability score, we first need an initial estimation of the path. We can use a heuristic such as the straight-line path [37] or any predefined path planning algorithm. In this work, we sample points in an ellipse shape $\mathbf{p}_i^{\text{ellipse}}$, $i \in M$, where M is the number of sampled points in the ellipse, with its focal points being the current position of the robot and a candidate target point $\mathbf{t}_i$ (see Fig. 3(b)). This shape models the potential traversable area more realistically, as the legged robot will not always move along a straight line to the target point due to the potential rough terrain it encounters. We solve for the mean of the slope prediction for each point inside the ellipse by evaluating the GP $\nabla \mu_{\hat{e},\mathcal{T},\{x,y\}}(\mathbf{p}_i^{\text{ellipse}})$, $i \in M$. The traversability score is then calculated as:

$$\mathcal{S}_t(\mathbf{t}_i) = -\mu_{\text{ellipse}}(\mathbf{t}_i) - \alpha_t \sigma_{\text{ellipse}}(\mathbf{t}_i), \forall i \in [1, T] \quad (11)$$

where $\mu_{\text{ellipse}}(\mathbf{t}_i)$ and $\sigma_{\text{ellipse}}(\mathbf{t}_i)$ represent the average and the standard deviation of the M slopes inside the ellipse for each candidate target point $\mathbf{t}_i$. The parameter α_t balances the mean and variance.

c) *Time score* $\mathcal{S}_d$: accounts for the distance between the robot's current position and the target point, scaled by the robot's maximum velocity. This score ensures that a target location closer to the robot is prioritized. The time score is calculated as:

$$\mathcal{S}_d(\mathbf{t}_i) = -\|\mathbf{t}_i - \mathbf{p}_0\|/v_{\max} \quad (12)$$

where $\mathbf{p}_0$ is the robot's current position, and $v_{\max}$ is the robot's maximum velocity.

The total score $\mathcal{S}_{\text{total}}$ for each point is a weighted sum of the individual scores:

$$\mathcal{S}_{\text{total}}(\mathbf{t}_i) = w_b \mathcal{S}_b(\mathbf{t}_i) + w_t \mathcal{S}_t(\mathbf{t}_i) + w_d \mathcal{S}_d(\mathbf{t}_i) \quad (13)$$

where w_b , w_t , and w_d are the nonnegative weights assigned to the belief, traversability, and time scores, respectively. These weights can be tailored based on the task priorities.

2) *Consensus Phase*: Solely relying on the auction phase for task allocations will potentially lead to task conflicts between the robots, where individual robots have the same targets or their paths to their target locations explore the same intermediate points. In the consensus phase, we propose a method to resolve such conflicts, by iterating through the T target locations of the two conflicting robots and selecting the targets that lead to the highest sum of scores for both robots without conflicts. As an example, consider a scenario in which $\mathbf{p}^{\text{LIP}}$ has a target $\mathbf{t}_1^{\text{LIP}}$ and $\mathbf{p}^{\text{quad}}$ has a target $\mathbf{t}_1^{\text{quad}}$, and they are in conflict. The conflict resolution solves for the optimal targets $(\mathbf{t}_i^{\text{LIP}}, \mathbf{t}_j^{\text{quad}})$ such that the sum $(\mathcal{S}_{\text{total}}(\mathbf{t}_i^{\text{LIP}}) + \mathcal{S}_{\text{total}}(\mathbf{t}_j^{\text{quad}}))$ is maximized and no conflict exists for the newly assigned targets.

B. Rescue Task

The rescue task consists of transferring the subjects S_1 and S_2 to the rescue location R . In the targeted scenario we set that a quadrotor can rescue S_1 , while S_2 requires two legged robots for the rescue mission.

1) *Rescue of S_1 with A Quadrotor*: Once S_1 is found, the target location for the nearest quadrotor (e.g., q_2) is set to reach S_1 . Once a quadrotor reaches S_1 , the target location for q_2 is set to R . While q_2 is rescuing S_2 , q_1 is continuously searching for S_2 with the bipedal robot team.

2) *Rescue of S_2 with Two Bipedal Robots*: Once S_2 is found both legged robot targets are set to S_2 . The legged robots are tasked to maintain a specific distance between them and carry S_2 to R . In this phase of the task, the available quadrotors (e.g. quadrotor that is not rescuing S_1) continuously map the terrain around the two bipeds.

C. Mapping Task

All robots update the terrain GP based on current measurements of the surrounding environment during the searching task. However, during the rescue phase, the quadrotors play a critical role in mapping the potential path that the legged robots will take. The quadrotors are tasked to survey the terrain by orbiting around the bipedal robots:

$$\mathbf{t}^{\text{quad}} = \left(x_{q+1}^{\text{LIP}} + r \cdot \cos\left(\frac{2 \cdot s}{\pi}\right), y_{q+1}^{\text{LIP}} + r \cdot \sin\left(\frac{2 \cdot s}{\pi}\right) \right)$$

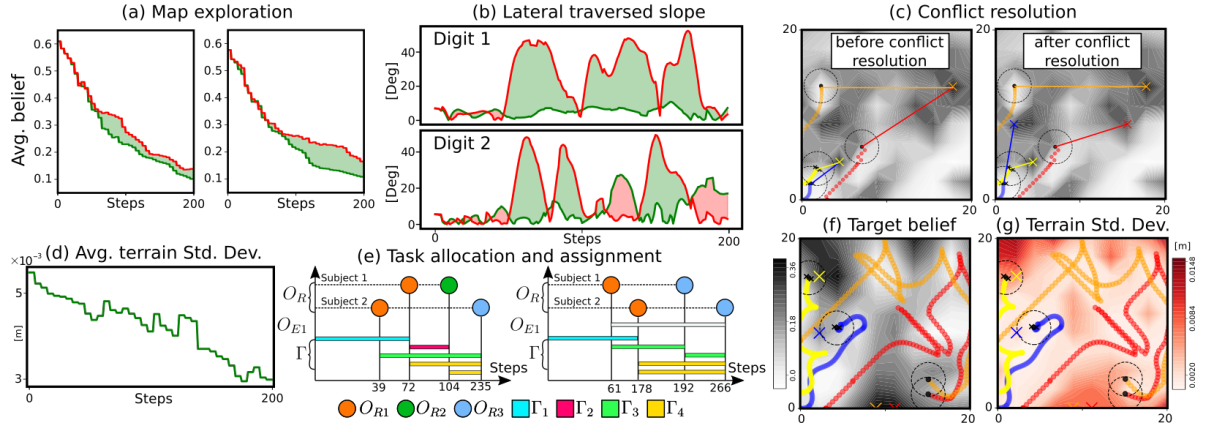


Fig. 4: Planning results: (a) shows the average belief value for two different trials with randomly selected initial belief conditions $\mathcal{B}$. The red line is trial without conflict resolution and the green line is with conflict resolution. (b) shows the lateral slopes traversed by both Digits. The red line is a trial without traversability score and slope minimization in the MPC and the green line is with traversability score and slope minimization. (c) shows conflict resolution when the auction phase outputs the same targets for the robot team, where the straight lines connect robots to their target points, (d) shows the average standard deviation of the terrain GP, (e) shows the task allocation and assignment results of two different runs with and without wind O_{E1} , (f) shows the belief value, and (g) shows that the standard deviation of the terrain estimation decreases as the robots explore the environment.

TABLE I: Parameter Values of Our Results

Parameter	Value	Parameter	Value
Environment	$20 \times 20 \text{ m}^2$	R	$(19, 19) \text{ m}$
N	10	D	0.4 s
d_l	0.9 m	d_u	2 m
ϵ	0.2 m	α_b	3
T	100	M	15
α_t	1	w_b	3
w_d^{quad}	0.5	w_d^{LIP}	1
w_t^{LIP}	1	$v_{\text{max}}^{\text{quad}}$	1.5 m/s
$v_{\text{max}}^{\text{LIP}}$	0.4 m/s	r	3

where r is the radius of the circle around the biped CoM state and s is the step increment.

VII. IMPLEMENTATION AND RESULTS

In this section, we will present the results of the proposed framework, in particular, the searching task and the terrain-aware MPC to minimize lateral slope for humanoid robot trajectories⁶. Details of the implementation parameters are shown in Table I.

A. Task Specification

Here we show the scLTL specifications used for the search and rescue task.

Subject 1: This specification indicates that subject 1 can only be rescued by the quadrotor when there is no wind at the location ($\neg O_{E1}$) and can be rescued by the bipedal robot when wind exists and the terrain is traversable ($O_{E1} \wedge \neg O_{E2}$):

$$\Diamond O_{R1} \wedge ([O_{R1} \wedge \neg O_{E1}] \rightarrow \bigcirc O_{R2}) \\ \vee ([O_{R1} \wedge O_{E1} \wedge \neg O_{E2}] \rightarrow \bigcirc O_{R3})$$

Subject 2: This specification indicates that subject 2 can only be rescued by the bipedal robots:

$$\Diamond O_{R1} \wedge ([O_{R1} \wedge \neg O_{E2}] \rightarrow \bigcirc O_{R3})$$

Fig. 4(e) depicts the progress through the scLTL specifications along with the corresponding task sequences for two sample runs of the mission.

⁶For rescue and mapping task please see the video submission.

B. Map Exploration and Conflict Resolution

We demonstrate the map exploration performance by inspecting the average target belief (Fig. 4(a)) of the environment with and without conflict resolution (Fig. 4(c)). As shown in Fig. 4(a), in two different trials with different random initial target belief information $\mathcal{B}$, the average belief value of the environment decreases as the robots explore the environment. We specifically show that with conflict resolution (green line) the average belief reduces at a faster rate than without conflict resolution (red line). Fig. 4(f), shows the belief value in the environment, and specifically, the reduction of belief uncertainty as the robots navigate through the environment.

C. Traversability and Terrain-Aware MPC

In Fig. 4(b), we show the result of integrating the terrain GP into the MPC to solve for paths with minimal lateral slopes, as well as adding the traversability score into the task allocation. Our proposed framework produces paths with low lateral slopes (green line) compared to the lateral slopes in the paths produced when not including terrain slope into the MPC and the traversability score in the total score (red line). Traversing paths with lower lateral slopes increases the locomotion safety of the humanoids as lateral stability is critical to maintaining balance [1], [4]. In Fig. 4(d) we show the standard deviation of the terrain GP, indicating the uncertainty reduction as the robots explore the environment. The average standard deviation decreased by $\approx 45\%$ in 200 steps. Fig. 4(g) shows that the standard deviation decreases along the explored paths.

VIII. CONCLUSION

This study presents a terrain-aware MPC for heterogeneous bipedal and aerial multi-robot teams conducting search and rescue missions in uncertain environments. Future work will focus on (i) expanding the task set to the full semantics of scLTL beyond the search and rescue domain, and (ii) potential hardware implementations like those shown in our previous work of [38].

REFERENCES

- [1] G. Gibson, O. Dosunmu-Ogunbi, Y. Gong, and J. Grizzle, "Terrain-adaptive, alip-based bipedal locomotion controller via model predictive control and virtual constraints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2022, pp. 6724–6731.
- [2] S. McCrory, B. Mishra, R. Griffin, J. Pratt, and H. E. Sevil, "Bipedal navigation planning over rough terrain using traversability models," in *SoutheastCon 2023*. IEEE, 2023, pp. 89–95.
- [3] J.-K. Huang and J. W. Grizzle, "Efficient anytime clf reactive planning system for a bipedal robot on undulating terrain," *IEEE Transactions on Robotics*, 2023.
- [4] Z. Gu, Y. Zhao, Y. Chen, R. Guo, J. K. Leestma, G. S. Sawicki, and Y. Zhao, "Robust-locomotion-by-logic: Perturbation-resilient bipedal locomotion via signal temporal logic guided model predictive control," *arXiv preprint arXiv:2403.15993*, 2024.
- [5] A. Shamsah, Z. Gu, J. Warnke, S. Hutchinson, and Y. Zhao, "Integrated task and motion planning for safe legged navigation in partially observable environments," *IEEE Transactions on Robotics*, pp. 1–22, 2023.
- [6] M. Murooka, K. Chappellet, A. Tanguy, M. Benallegue, I. Kumagai, M. Morisawa, F. Kanehiro, and A. Kheddar, "Humanoid locomanipulations pattern generation and stabilization control," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5597–5604, 2021.
- [7] I. Radosavovic, T. Xiao, B. Zhang, T. Darrell, J. Malik, and K. Sreenath, "Real-world humanoid locomotion with reinforcement learning," *Science Robotics*, vol. 9, no. 89, p. eadi9579, 2024.
- [8] E. Kaufmann, L. Bauersfeld, A. Loquercio, M. Müller, V. Koltun, and D. Scaramuzza, "Champion-level drone racing using deep reinforcement learning," *Nature*, vol. 620, no. 7976, pp. 982–987, 2023.
- [9] C. Papachristos, S. Khatkhat, and K. Alexis, "Uncertainty-aware receding horizon exploration and mapping using aerial robots," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 4568–4575.
- [10] Y. Zhao, Y. Li, L. Sentis, U. Topcu, and J. Liu, "Reactive task and motion planning for robust whole-body dynamic locomotion in constrained environments," *The International Journal of Robotics Research*, vol. 41, no. 8, pp. 812–847, 2022.
- [11] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, vol. 1. IEEE, 2001, pp. 239–246.
- [12] C. K. I. Williams and C. E. Rasmussen, "Gaussian processes for regression," in *Advances in neural information processing systems 8*. MIT press, 1996, pp. 514–520.
- [13] J. Jiang, Y. Zhao, and S. Coogan, "Safe learning for uncertainty-aware planning via interval mdp abstraction," *IEEE Control Systems Letters*, vol. 6, pp. 2641–2646, 2022.
- [14] J. Jiang, S. Coogan, and Y. Zhao, "Abstraction-based planning for uncertainty-aware legged navigation," *IEEE Open Journal of Control Systems*, 2023.
- [15] K. Muenprasitvej, J. Jiang, A. Shamsah, S. Coogan, and Y. Zhao, "Bipedal safe navigation over uncertain rough terrain: Unifying terrain mapping and locomotion stability," *arXiv preprint arXiv:2403.16356*, 2024.
- [16] F. Quinton, C. Grand, and C. Lesire, "Market approaches to the multi-robot task allocation problem: a survey," *Journal of Intelligent & Robotic Systems*, vol. 107, no. 2, p. 29, 2023.
- [17] H. Chakraborty, F. Guérin, E. Leclercq, and D. Lefebvre, "Optimization techniques for multi-robot task allocation problems: Review on the state-of-the-art," *Robotics and Autonomous Systems*, p. 104492, 2023.
- [18] H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," *IEEE transactions on robotics*, vol. 25, no. 4, pp. 912–926, 2009.
- [19] A. Khamis, A. Hussein, and A. Elmogy, "Multi-robot task allocation: A review of the state-of-the-art," *Cooperative robots and sensor networks 2015*, pp. 31–51, 2015.
- [20] J. Parker, E. Nunes, J. Godoy, and M. Gini, "Exploiting spatial locality and heterogeneity of agents for search and rescue teamwork," *Journal of Field Robotics*, vol. 33, no. 7, pp. 877–900, 2016.
- [21] B. Fu, W. Smith, D. M. Rizzo, M. Castanier, M. Ghaffari, and K. Barton, "Robust task scheduling for heterogeneous robot teams under capability uncertainty," *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 1087–1105, 2022.
- [22] G. Silano, T. Baca, R. Penicka, D. Liuzza, and M. Saska, "Power line inspection tasks with multi-aerial robot systems via signal temporal logic specifications," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 4169–4176, 2021.
- [23] M. Badreldin, A. Hussein, and A. Khamis, "A comparative study between optimization and market-based approaches to multi-robot task allocation," *Advances in Artificial Intelligence*, vol. 2013, no. 1, p. 256524, 2013.
- [24] X. Chen, P. Zhang, G. Du, and F. Li, "A distributed method for dynamic multi-robot task allocation problems with critical time constraints," *Robotics and Autonomous Systems*, vol. 118, pp. 31–46, 2019.
- [25] F. Ye, J. Chen, Q. Sun, Y. Tian, and T. Jiang, "Decentralized task allocation for heterogeneous multi-uav system with task coupling constraints," *The Journal of supercomputing*, vol. 77, pp. 111–132, 2021.
- [26] C. Menghi, S. Garcia, P. Pelliccione, and J. Tumova, "Multi-robot ltl planning under uncertainty," in *International Symposium on Formal Methods*. Springer, 2018, pp. 399–417.
- [27] P. Schillinger, M. Bürger, and D. V. Dimarogonas, "Multi-objective search for optimal multi-robot planning with finite ltl specifications and resource constraints," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 768–774.
- [28] M. Kloetzer and C. Mahulea, "Multi-robot path planning for syntactically co-safe ltl specifications," in *2016 13th International Workshop on Discrete Event Systems (WODES)*, 2016, pp. 452–458.
- [29] K. Grover, F. S. Barbosa, J. Tumova, and J. Kretinsky, "Semantic abstraction-guided motion planning for sltl missions in unknown environments," in *Robotics: Science and Systems*, 2021.
- [30] K. S. Narkhede, A. M. Kulkarni, D. A. Thanki, and I. Poulakakis, "A sequential mpc approach to reactive planning for bipedal robots using safe corridors in highly cluttered environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 831–11 838, 2022.
- [31] S. Teng, Y. Gong, J. W. Grizzle, and M. Ghaffari, "Toward safety-aware informative motion planning for legged robots," *arXiv preprint arXiv:2103.14252*, 2021.
- [32] A. Shamsah, K. Agarwal, N. Katta, A. Raju, S. Kousik, and Y. Zhao, "Socially acceptable bipedal robot navigation via social zonotope network model predictive control," *arXiv preprint arXiv:2406.17151*, 2024.
- [33] L. K. Chung, W. Jung, C. Kong, and S. Kousik, "Goal-reaching trajectory design near danger with piecewise affine reach-avoid computation," *arXiv preprint arXiv:2402.15604*, 2024.
- [34] M. Chen, S. L. Herbert, H. Hu, Y. Pu, J. F. Fisac, S. Bansal, S. Han, and C. J. Tomlin, "Fastrack: a modular framework for real-time motion planning and guaranteed safe tracking," *IEEE Transactions on Automatic Control*, vol. 66, no. 12, pp. 5861–5876, 2021.
- [35] J. E. Johnson, V. Laparra, A. Pérez-Suay, M. D. Mahecha, and G. Camps-Valls, "Kernel methods and their derivatives: Concept and perspectives for the earth system sciences," *Plos one*, vol. 15, no. 10, p. e0235885, 2020.
- [36] C. Belta, B. Yordanov, and E. Gözl, *Formal Methods for Discrete-Time Dynamical Systems*, ser. Studies in Systems, Decision and Control. Springer International Publishing, 2017.
- [37] M. Quann, L. Ojeda, W. Smith, D. Rizzo, M. Castanier, and K. Barton, "An energy-efficient method for multi-robot reconnaissance in an unknown environment," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 2279–2284.
- [38] M. E. Cao, X. Ni, J. Warnke, Y. Han, S. Coogan, and Y. Zhao, "Leveraging heterogeneous capabilities in multi-agent systems for environmental conflict resolution," in *IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2022.