Coupled Active Perception and Manipulation Planning for a Mobile Manipulator in Precision Agriculture Applications

Shuangyu Xie, Chengsong Hu, Di Wang, Joe Johnson, Muthukumar Bagavathiannan, and Dezhen Song

***Abstract*— A mobile manipulator often finds itself in an application where it needs to take a close-up view before performing a manipulation task. Named this as a coupled active perception and manipulation (CAPM) problem, we model the uncertainty in the perception process and devise a key state/task planning algorithm that considers reachability conditions jointly established from perception and manipulation task constraints. By minimizing expected energy usage in body key state planning while satisfying task constraints, our algorithm is able to find an energy-efficient trajectory with less body repositioning motion while ensuring the success of the task. We have implemented the algorithm and tested it in both simulation and physical experiments. The results have confirmed that our algorithm has a lower energy consumption compared to a two-stage decoupled approach, while still maintaining a success rate of 100% for the task.**

I. Introduction

In precision agriculture or various mobile manipulation applications, high-resolution scene maps are often inaccessible before the task due to the expensive construction cost or the non-static nature of the scene (e.g., target object moving [1] or growing [2]). Due to the lack of detailed information on the target object, a typical scenario arises in which the robot first needs to obtain a close-up view of the object of interest before planning the manipulation task. The close-up view provides high-resolution images, facilitating the precise recognition algorithm [3]. Based on the precise target information, the mobile manipulator can determine where and how the manipulation should proceed. The process of obtaining the close-up view is an active perception problem. Combining with manipulator task planning, it leads to a coupled active perception and manipulation (CAPM) problem because the probabilistic distribution of the perception result and the end configuration of the robot in active perception affect the subsequent manipulation problem.

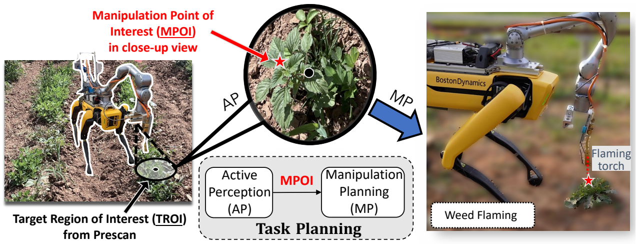

Manipulation Point of Interest (MPOI) in close-up view
AP
MP
BostonDynamics
Flaming torch
Active Perception (AP)
MPOI
Manipulation Planning (MP)
Task Planning
Weed Flaming
Target Region of Interest (TROI) from Prescan

Fig. 1: An illustration of CAPM problem scenario using the weed flaming application.

Fig. 1 illustrates the CAPM problem using weed removal as an application example where the task is to precisely burn down the biologic center of the weed in the field using the robot. The robot is a hand-eye mobile manipulator with a quadrupedal platform. The robot has a low-resolution prescan of the scene (similar to the background image on the left-hand side in Fig. 1) to start the task, but the prescan's resolution is insufficient to determine the desired flaming point. With the knowledge of the rough target region of interest (TROI), the robot first navigates towards it to get a close-up view to determine the flaming target position. We name this flaming position as the manipulation point of interest (MPOI), which is shown as the red star in the figure. Due to the uncertainty of MPOI, the robot has to adjust its next task correspondingly: directly perform the manipulation task (i.e. weed flaming) if the MPOI is within reach or reposition its body for the manipulation.

The research question is whether this CAPM task planning can be solved in an efficient manner. In this work, we formulate the CAPM problem and analyze four types of problem using reachability analysis by designing the constraints of the active perception and manipulation tasks. We model energy usage in the CAPM problem and propose an algorithm to minimize the expected energy usage by reducing the body reposition motion. We have implemented our algorithm and tested it in both simulation and weed-flaming physical experiments. The experimental results have confirmed that our algorithm has lower energy consumption compared to a two-stage decoupled approach, while still maintaining a success rate of 100% for the task.

II. Related Work

For our task planning, determining the close-up view can be achieved efficiently through active perception. Active perception, by definition, is an intelligent data acquisition process [4], which guides the robot to take intentional actions to perceive the required information [5]. Numerous active perception algorithms are developed for multiple purposes, such as alleviating ambiguity or occlusion issues in object recognition [6], [7], improving performance for UAV navigation [8], and multi-robot path planning [9]. The key challenge for active perception is to define the scene-related criteria [4] as feedback to planning and control (e.g., semantic characteristic [8], cross-frame scan overlap [10]). For the vision-based sensor, these criteria are achieved by adjusting the viewpoint

S. Xie, D. Wang, and D. Song are with Department of Computer Science and Engineering, Texas A&M University. D. Song is also with Department of Robotics, Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI) in Abu Dhabi, UAE. C. Hu, J. Johnson and M. Bagavathiannan are with Department of Soil and Crop Sciences, Texas A&M University. Corresponding author: Dezhen Song. Email: dezhen.song@mbzuai.ac.ae.

or the sensor field of view [11]. However, these works [6]–[8] focus solely on finding the best view without considering the subsequent task. We propose the next sufficient view condition for the active perception constraint as feedback for the robot state. This constraint can be integrated into the planning framework, allowing joint optimization for both active perception and manipulation tasks.

Mobile manipulators have been widely deployed for tasks in factories such as pick-up and delivery [12] or indoor applications such as opening doors [13]) due to their high dexterity and mobility. For long-horizon tasks with sequential nature like chores, the Task and Motion Planning (TAMP) method is designed to discretize the action space into symbolic action with continuous motion. We use a similar formulation by designing the action sequence first and then solving for the state parameters. When scene information is not fully provided, deterministic TAMP approaches face the issue of incorporating it into real-world applications. Recently, there has been progress in the description of the scene that contains uncertainty. Probabilistic modeling of the robot/target state is developed and integrated into the TAMP framework in [14], [15]. These methods have the advantage of dealing with generating sequential action for complicated task space, but they still passively receive the observation instead of actively planning for observation that improve system efficiency.

To reduce cost and labor dependency, robotic solutions are integrated with precision agriculture applications that include scene perception [3], [16]–[18], robot navigation [19], motion planning [20], and the deployment of aerial/ground platforms [21]–[23]. However, due to the nonstatic scene caused by the complex plant morphology, close-up-view observation should be considered with manipulation in task planning.

III. Problem Formulation

Our goal is to find an efficient action plan and state parameters for a mobile manipulator to tackle CAPM tasks using weed flaming as an example. To formulate our problem and focus on the key issues, we have the following assumptions.

A. Assumptions

a.1 The mobile platform is holonomic.

a.2 We do not consider obstacle avoidance in body motion planning because our weed removal robot stays on top of weeds and crops in agriculture fields.

a.3 The robot does not execute body and arm motion simultaneously to ensure the stability.

a.4 The energy usage of the arm motion is negligible compared to that of the body.

a.5 Camera resolution is fixed.

Assumptions a.1 and a.2 ensure that the shortest trajectory of base motion is always along the straight linear path from the current state to the goal state in each task. Therefore, the motion planning problem is reduced to a state planning problem. For our platform, the arm weight is less than 10% of the overall platform. This condition establishes assumption a.4, which will determine the energy model later in the paper.

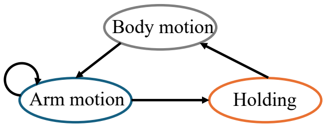


Fig. 2: Action transition diagram.

Motion Type	Task/Goal
Body motion	Navigation
Arm motion	Observation manipulation
Holding	Manipulation

TABLE I: Action/task table.

B. Input, Key States, and Output

1) Initial Input: Prior to the task, the initial state of the robot is given as input. Scene information is provided by a low-resolution field prescan. This prescan can be obtained by drone surveillance or cameras above ground. Let us define the field as the robot workspace $\mathcal{W} \subset \mathbb{R}^3$. With the low-resolution prescan, the object recognition algorithm can be applied to identify the TROI defined as $R_w^{\text{TROI}} \subset \mathcal{W}$ as illustrated in Fig. 1. Without loss of generality, we model TROI as a half-ball above the groundplane centered on $X_w = [x_w, y_w, 0]^\mathsf{T} \in \mathcal{W}$ and with radius r_w. It is possible that the workspace contains more than one TROI and the robot has to handle them one at a time. We focus only on the next TROI in this paper, which is defined as,

$$R_w^{\text{TROI}} = \{X = [x, y, z]^\mathsf{T} : \|X - X_w\|_2^2 \leq r_w, z \geq 0\} \subset \mathcal{W}.$$

R_w^{TROI} is a primary input of our problem.

Remark: X_w does not necessarily overlap with MPOI $\bar{X}_w \in \mathcal{W}$ because X_w is the geometrical center position observed from the low-resolution prescan and $\bar{X}_w$ is the biological center. MPOI can be anywhere within TROI, i.e. $\bar{X}_w \in R_w^{\text{TROI}}$. $\bar{X}_w$ cannot be obtained from the prescan data because it requires a close-up view image for recognition [3]. R_w^{TROI} guide initial body motion planning for active perception. Obtaining $\bar{X}_w$ efficiently is the active perception part of the problem, which is related to the notion of key states.

2) Key States: Our state representation includes both the environment states and the robot states. While R_w^{TROI} and $\bar{X}_w$ fully describe the static environment, the robot states are dynamic. Denote the discrete time set $\mathcal{K} = \{0, ..., k_{\max}\}$ as the time index set for the entire problem. Let us represent the state of the robot at time k as $\mathbf{x}_k := [\mathbf{x}_{e,k}, \mathbf{x}_{b,k}]^\mathsf{T}$. For $k \in \mathcal{K}$, $\mathbf{x}_k$ includes both the pose of the mobile platform (aka body) $\mathbf{x}_{b,k} \in \mathcal{X}_b \subset SE(3)$ and the manipulator (aka arm) states $\mathbf{x}_{e,k} \in \mathcal{X}_e \subset SE(3)$, where $\mathcal{X}_b$ and $\mathcal{X}_e$ are the state space of body and arm. Denote the 3D position and orientation of the body in quarternion as $X_{b,k}$ and $\mathbf{q}_{b,k}$. For the arm state, $\mathbf{x}_{e,k}$ is represented by the pose of the end effector rather than by the arm configuration in its joint space.

Key states refer to states acting as decision points that significantly alter the operation of the robot. We consider two types of key state. According to Assumption a.2 in Sec. III-A, the motion of the robot body and the motion of the arm are executed disjointly. The states that trigger the switching between motion types are one type of key state. Table. I and Fig. 2 explains the types of robot motion and the switching relationship between them. The other type of key states are the states associated with goal change even with the same motion type. For example, if the arm switches from an active perception task to a manipulation task, the state associated with the switch is also a key state. Table. I shows the correspondence between the type of motion and the goal.

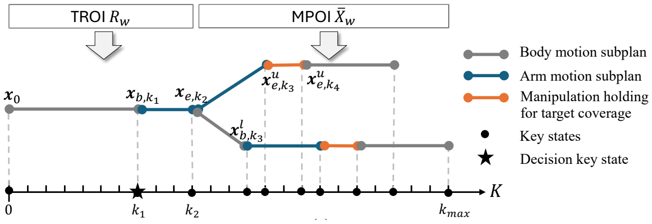


Fig. 3: Two possible key state operation sequences.

3) CAPM Operation Sequence: Fig. 3 illustrates two typical operation sequences in key states. Initially, the robot is in the starting state $\mathbf{x}_0$ and let $k_0 = 0$ since the first key state is the starting state. The robot moves its base to $\mathbf{x}_{b,k_1}$, which allows it to move closer to R_w^{TROI}. k_1 is the time index when the robot makes its first transition from body motion to arm motion to obtain a close-up observation of TROI. At time k_2, the arm reaches $\mathbf{x}_{e,k_2}$ to closely observe R_w^{TROI} and provides high-resolution images for the recognition algorithm to estimate MPOI $\bar{X}_w$.

Depending on $\bar{X}_w$ in relation to the pose of the robot, there are two possible subsequent operations depicted as upper and lower branches in Fig. 3. In the upper branch, the robot finds that it can perform task manipulation (i.e. weed flaming) without moving its body. Then it executes the arm motion to reach the flaming pose $\mathbf{x}^u_{e,k_3}$ (the subscript u means upper branch) and stay there until k_4 based on the flaming duration requirement before moving to the next target. In the lower branch, since the robot cannot reach $\bar{X}_w$, it must be repositioned to $\mathbf{x}^l_{b,k_3}$ (the subscript l means lower branch) before the flaming operation. The following action sequence is the same as that in the upper branch. Since the upper branch and the lower branch have different time lengths, we denote the time index for the ending state as k_{max}, which should also be the last key state. It is clear that the key states are non-deterministic because they are largely dependent on the inherently probabilistic perception result $\bar{X}_w$.

4) Output: Let us define the set of key states as $\{\mathbf{x}_{k_i} | k_i \in \mathcal{K}'\}$, where $\mathcal{K}' \subset \mathcal{K}$ is the set of key state time indexes. In fact, the output can be viewed as a sequence of task planning with each task goal represented as a key state. As stated in Assumption a.3 in Sec. III-A, our focus is not solving the full motion plan, but the generation of key states. It is worth noting that we cannot generate the entire sequence of key states at once with initial input because of the uncertainty in perception. In particular, as the output of active perception, the value of MPOI $\bar{X}_w$ observed in the key state of the arm $\mathbf{x}_{e,k_2}$ determines the follow-up key state choices. In addition, $\mathbf{x}_{e,k_2}$ depends on the previous body key state $\mathbf{x}_{b,k_1}$. Later, we will explain those dependencies as active perception and manipulation constraints. For now, this is sufficient for us to introduce the following problem definition.

C. Problem Definitions

Definition 1 (CAPM Definition): Given the initial state $\mathbf{x}_{k_0}$ and TROI R_w^{TROI}, sequentially plan for key states $\mathbf{x}_{k_1}, ..., \mathbf{x}_{k_{\text{max}}}$, where $k_1, ..., k_{\text{max}} \in \mathcal{K}'$, to obtain MPOI $\bar{X}_w$ to guide and execute the subsequent manipulation task with the minimum expected energy cost:

$$\mathbb{E}\left[\sum_{k_i \in \mathcal{K}'} c(\mathbf{x}_{k_{i-1}}, \mathbf{x}_{k_i})\right] \qquad (1)$$

where $\mathbb{E}(\cdot)$ is expectation function and $c(\cdot)$ is energy function that will be defined later in the algorithm section.

Note that obtaining MPOI $\bar{X}_w$ means that certain key states need to satisfy the perception condition. Meanwhile, the completion of the manipulation task means that the task execution condition should be satisfied. Therefore, the definition of the current problem is not complete and (1) is a constrained optimization problem with conditions introduced in the next section.

IV. Algorithm

Before we introduce our algorithm, it is important to explain the task constraints associated with active perception and manipulation.

A. Task Constraints

1) Active Perception Constraints: To obtain MPOI $\bar{X}_w$, the robot must observe TROI R_w fully in the field of view and be close enough so that the image can provide sufficient details. This leads to view coverage condition and target resolution condition. These two conditions, which are dubbed the next sufficient view condition as a whole, impose constraints on the pose of the arm $\mathbf{x}_{e,k}$ because the camera is mounted on the end effector. From now on, we drop k from the notation (e.g. use $\mathbf{x}_e$ instead of $\mathbf{x}_{e,k}$) for brevity. k can be added back if the temporal index becomes necessary. Denote TROI projection on the ground plane as $R_w^g = \{R_w^{\text{TROI}} | z = 0\}$.

View coverage condition: Keeping the entire R_w^g in the camera field of view guarantees that $\bar{X}_w$ is visible within the image. Let the camera image resolution be $u_{\text{max}} \times v_{\text{max}}$ pixels. The entire image can be represented as a set of pixels in a homogeneous coordinate $I := \{\mathbf{p} | \mathbf{p} = [u, v, 1]^\mathsf{T}, u \in [1, u_{\text{max}}], v \in [1, v_{\text{max}}]\}$. Since the camera image covers the ground plane in which the targets of interest lie, the relationship between the ground plane and the image plane can be characterized as a homography transformation $H_{\mathbf{x}_e}$ from the projective geometry. $H_{\mathbf{x}_e}$ is parameterized by the pose of the end effector $\mathbf{x}_e$. The camera field of view can be back-projected to $\mathcal{W}$ as R_w^{FoV}: $R_w^{\text{FoV}} = \{X | X = (H_{\mathbf{x}_e}^{-1})\mathbf{p},\ X \in \mathcal{W},\ \forall \mathbf{p} \in I\}$. View coverage condition is defined as a binary function:

$$\mathbb{1}_{\text{coverage}}(\mathbf{x}_e, R_w^{\text{TROI}}) = \begin{cases} 1, & \text{if } R_w^g \subset R_w^{\text{FoV}} \\ 0, & \text{otherwise.} \end{cases} \qquad (2)$$

Target resolution condition: It is important to ensure that the projection area of TROI R_w^g into the image coordinate system is large enough so that the object recognition algorithm can effectively detect MPOI $\bar{X}_w$. R_w^g is projected to R_I^{TROI} in the image coordinate system indicated by the subscript I: $R_I^{\text{TROI}} = \{\mathbf{p}|\mathbf{p} = H_{\mathbf{x}_e} X, \ \forall X \in R_w^g\}$. Let the area of R_I^{TROI} in the image coordinate system be Area(R_I^{TROI}). Define the area ratio of R_I^{TROI} in the entire image as the target resolution condition $h(\mathbf{x}_e, R_w^{\text{TROI}})$:

$$h(\mathbf{x}_e, R_w^{\text{TROI}}) = \frac{\text{Area}(R_I^{\text{TROI}})}{u_{\max} v_{\max}} \geq \delta \tag{3}$$

where δ is the threshold for the area ratio.

The overall active perception constraint is a combination of view coverage condition and target resolution condition. We name it as the next sufficient view condition (NSV) $\mathbb{1}_{\text{NSV}}(\mathbf{x}_e, R_w^{\text{TROI}})$ as follows,

$$\mathbb{1}_{\text{NSV}}(\mathbf{x}_e, R_w^{\text{TROI}}) = \begin{cases} 1, & \text{if } \mathbb{1}_{\text{coverage}}(\mathbf{x}_e, R_w^{\text{TROI}}) \bigwedge \left(h(\mathbf{x}_e, R_w^{\text{TROI}}) \geq \delta\right) \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

2) Manipulation Constraints: To execute the manipulation task, for example weed flaming, it is necessary that the manipulator maintains a certain pose for a given amount of time, which are the spatiotemporal manipulation constraints.

End-effector pose condition for manipulation (EPMC): This is the condition to determine whether the manipulator can reach the target but not too close according to the task requirement. Being too close may cause self-collision or inability to execute weed flaming task without damaging the robot. For a given MPOI $\bar{X}_w$ and the state of the end effector $\mathbf{x}_{e,k}$, we define EPMC as a binary function $\mathbb{1}_{\text{EPCM}}(\mathbf{x}_{e,k}, \bar{X}_w)$ at time $k \in \mathcal{K}_m$, where $\mathcal{K}_m \subset \mathcal{K}$ represents a discrete time set during the manipulation period. Recall that $\mathbf{x}_{e,k} \in SE(3)$ is the pose of the end effector, let $X_{e,k} \in \mathcal{W}$ be its position components, and let the orientation of the end effector point to the MPOI. Define $\varepsilon_{\min}$ and $\varepsilon_{\max}$ as the minimum and maximum distance thresholds for the end effector to be able to perform the task, respectively. EPCM is defined as

$$\mathbb{1}_{\text{EPCM}}(\mathbf{x}_{e,k}, \bar{X}_w) = \begin{cases} 1, & \text{if } \|X_{e,k} - \bar{X}_w\|_2^2 \in [\varepsilon_{\min}, \varepsilon_{\max}] \\ 0, & \text{Otherwise.} \end{cases} \tag{5}$$

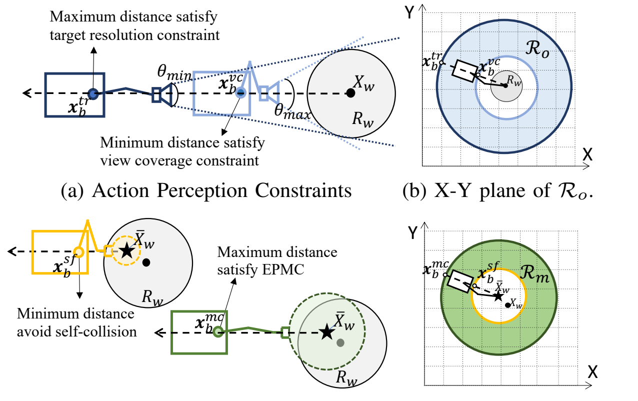
Manipulation temporal condition (MTC): In a weed flaming task, the manipulator is required to hold its position for a certain time ξ to ensure sufficient heat transfer. Such an MTC also exists in many other tasks. Recall that $\mathcal{K}_m$ is the time index set for the manipulator to maintain the pose $\mathbf{x}_e$. Let us define MTC $\mathbb{1}_{\text{MTC}}(\mathbf{x}_e, \bar{X}_w)$ as a binary function,

$$\mathbb{1}_{\text{MTC}}(\mathbf{x}_e, \bar{X}_w) = \begin{cases} 1, & \text{if } \mathbb{1}_{\text{EPCM}}(\mathbf{x}_e, \bar{X}_w) \wedge \left(|\mathcal{K}_m| \geq \xi\right) \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

where $|\cdot|$ is set cardinality.

B. Inverse Reachability Region Analysis for Task Constraints

Based on the inverse kinematics of the arm, the task constraints in Sec. IV-A can be used to determine the feasible body states. Denote the joint space of the arm as Θ. For a given body pose $\mathbf{x}_b \in \mathcal{X}_b$ and joint configuration, the forward kinematics function of the arm maps to the last link pose $\mathbf{x}_e \in \mathcal{X}_e$, $f : \Theta \times \mathcal{X}_b \to \mathcal{X}_e$ and the inverse kinematics function is $f^{-1} : \mathcal{X}_b \times \mathcal{X}_e \to \Theta$. The inverse kinematics function $f^{-1}(\mathbf{x}_b, \mathbf{x}_e)$ for a given end effector $\mathbf{x}_e$ and a body pose $\mathbf{x}_b$ may not necessarily have a solution. We use the term $\exists f^{-1}(\mathbf{x}_b, \mathbf{x}_e)$ to indicate the logical truth that there exists a solution.



(a) Action Perception Constraints (b) X-Y plane of $\mathcal{R}_o$.

(c) Manipulation Constraints (d) X-Y plane of $\mathcal{R}_m$.

Fig. 4: A visualization of feasible body pose sets $\mathcal{R}_o$ and $\mathcal{R}_m$ with color coded boundaries.

As shown in Fig. 4a, for a given TROI R_w^{TROI}, the robot body cannot be too far from it because it needs to satisfy (3) with a fixed resolution camera (Assumption a.5). At the farthest reachable point, the dark blue robot arm has to be fully extended to provide good coverage of R_w^{TROI}. Since the robot can approach R_w^{TROI} from any direction, this boundary condition leads to the dark blue outer circular boundary in Fig. 4b as a visualization in the X-Y plane. On the other hand, the robot body cannot be too close to R_w^{TROI} due to the coverage condition in (2). Being too close cannot maintain a full view of TROI. The boundary condition corresponds to the close-up robot configuration and the inner circular boundary in light blue in Figs. 4a and 4b, respectively. The region between the two boundaries is the feasible body pose set in the X-Y plane. Define $\mathcal{R}_o(R_w^{\text{TROI}})$ as the set of feasible body poses, we have

$$\mathcal{R}_o(R_w^{\text{TROI}}) = \{\mathbf{x}_b | \mathbb{1}_{\text{NSV}}(\mathbf{x}_e, R_w^{\text{TROI}}) \wedge \left(\exists f^{-1}(\mathbf{x}_b, \mathbf{x}_e)\right), \forall \mathbf{x}_e\}, \tag{7}$$

which shows as a blue donut shape co-centered with R_w^{TROI} in X-Y plane (Fig. 4b).

For a given MPOI $\bar{X}_w$, the body pose should be selected such that the arm poses can satisfy (5) during the task. Since (5) has both minimum and maximum distance thresholds, the set of reachable body poses $\mathcal{R}_m$ also has both inner and outer boundaries color coded in yellow and dark green, respectively, as shown in Figs. 4c and 4d. Therefore, the shape of $\mathcal{R}_m$ is also donut-like in the X-Y plane. Mathematically, $\mathcal{R}_m$ is defined as follows,

$$\mathcal{R}_m(\bar{X}_w) := \{\mathbf{x}_b | \mathbb{1}_{\text{MTC}}(\mathbf{x}_e, \bar{X}_w) \wedge \left(\exists f^{-1}(\mathbf{x}_b, \mathbf{x}_e)\right), \forall \mathbf{x}_e\}. \tag{8}$$

C. CAPM Problem Types in Mobile Manipulation

Defining $\mathcal{R}_o$ and $\mathcal{R}_m$ allows us to classify CAPM problems based on region relationships. Depending on robot sensing and actuation configurations and task requirements, we can classify the CAPM problem into four types, as illustrated in Fig. 5. Before discussing the four types, it is worth noting that $\mathcal{R}_o$ and $\mathcal{R}_m$ are not necessarily co-centered in the X-Y plane, because the former is centered at X_w, the center of TROI, and the latter is centered at MPOI $\bar{X}_w$.

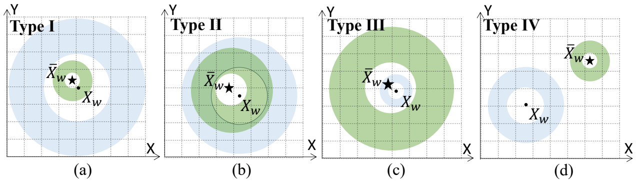


Fig. 5: CAPM problem types based on four different $\mathcal{R}_o$ (in lighter blue) and $\mathcal{R}_m$ (in darker green) relationships.

Type I: As shown in Fig. 5 (a), $\mathcal{R}_o$ is much larger than $\mathcal{R}_m$. In fact, $\mathcal{R}_m$ is completely enclosed by the inner boundary of $\mathcal{R}_o$ and $\mathcal{R}_o \cap \mathcal{R}_m = \emptyset$. This happens when the robot has a strong perception capability that enables the observation for complete TROI at a much further distance. In such problems, active perception and manipulation are decoupled and are essentially a two-step problem that can be solved independently.

Type II : For Type II (Fig. 5 (b)), $\mathcal{R}_o$ has a significant overlap with $\mathcal{R}_m$, but none can fully enclose the other, which means that both $\mathcal{R}_o$ and $\mathcal{R}_m$ are strict subsets of $\mathcal{R}_o \cup \mathcal{R}_m$. In such a coupled situation, if a robot can plan its body to be in $\mathcal{R}_o \cap \mathcal{R}_m$, then active perception and manipulation can be addressed simultaneously.

Type III: Fig. 5 (c) is the type opposite to Type I where $\mathcal{R}_m$ is much larger than $\mathcal{R}_o$. In fact, $\mathcal{R}_o$ is completely enclosed by the inner boundary of $\mathcal{R}_m$ and $\mathcal{R}_o \cap \mathcal{R}_m = \emptyset$. Such a situation occurs when the sensor is very nearsighted, similar to the endoscope camera in minimally invasive surgery [24]. After active perception, the robot needs to retreat for manipulation, which leads to separate solving of active perception and manipulation problem.

Type IV: Fig. 5 (d) shows the case where $\bar{X}_w \notin R_w^{\text{TROI}}$ and $\mathcal{R}_o \cap \mathcal{R}_m = \emptyset$. This means that the prescan is erroneous or is not up to date. Such a problem usually becomes a search problem like [14] instead of a CAPM problem.

From the analysis, Type II problems fit the nature of mobile manipulator with a common camera sensor, and this is where the CAPM problem actually matters.

D. Energy Cost Function

In order to solve the CAPM problem while minimizing the energy used, we model the energy cost for the task plan defined by piecewise motion (as illustrated in Fig. 3) using key states. For two key states $\mathbf{x}_{k_i}$, $\mathbf{x}_{k_j}$, the energy cost is a combination of a fixed initial energy cost for the mobile platform and the variable energy cost. Since we assume that the energy of arm movement is much lower than that of body movement, the arm energy usage is ignored in the cost function. The start energy cost is a fixed cost once movement occurs. We follow the method in [25] to define the distance metric between two states as

$$d(\mathbf{x}_{k_i}, \mathbf{x}_{k_j}) = \|X_{b,k_j} - X_{b,k_i}\|_2^2 + \beta(1 - \|\mathbf{q}_{b,k_j} \cdot \mathbf{q}_{b,k_i}\|),$$

where $\beta \geq 0$ and determined by specific robot setup. Denote the indicator function for body movement as follow:

$$\mathbb{1}_{\text{b-move}}(\mathbf{x}_{k_j}, \mathbf{x}_{k_i}) = \begin{cases} 1, & \text{if } d(\mathbf{x}_{k_i}, \mathbf{x}_{k_j}) > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

The overall energy cost function is defined by the initial energy cost(first term) and variable energy cost(second term) in the case where body only move once between $\mathbf{x}_{k_i}, \mathbf{x}_{k_j}$:

$$c(\mathbf{x}_{k_i}, \mathbf{x}_{k_j}) = \mathbb{1}_{\text{b-move}}(\mathbf{x}_{k_j}, \mathbf{x}_{k_i}) + \gamma d(\mathbf{x}_{k_i}, \mathbf{x}_{k_j}). \tag{10}$$

E. Coupled two-stage key states planning

With the constraint and energy cost introduced, we are ready to solve the CAPM problem. Intuitively, from the perspective of reducing energy cost, the overall planned key state sequence should contain fewer body movements. The ideal situation is that the robot body only needs to move once to satisfy the requirements of both active perception and manipulation tasks, as shown in the upper branch of Fig. 3. Through the analysis in Sec. IV-C, we know that it is possible for the Type II problem. However, for unknown $\bar{X}_w$, it is unlikely that this can be guaranteed.

From Sec. III-B.3, there are two key body states $\mathbf{x}_{b,k_1}$ and possible $\mathbf{x}_{b,k_3}$ that determine the energy use. These two become the key decision variable in our planning problem. However, they cannot be obtained in advance because $\bar{X}_w$ is unknown prior to close-up observation. The manipulation constraint cannot be evaluated beforehand. More specifically, $\mathbf{x}_{b,k_1}$ and MPOI jointly determine whether additional body movement is needed to achieve manipulation. Denote the probability that the body state for active perception $\mathbf{x}_{b,k_1}$ is also feasible for manipulation as $p(\mathbf{x}_{b,k_1})$. Since we know $\bar{X}_w \in R_w^{\text{TROI}}$, we can establish a probability distribution for the estimate of the target position $\hat{X}_w \sim \mathcal{N}(X_w, \Sigma_w)$ using the target region of interest described by the center X_w and the radius r_w. The probability $p(\mathbf{x}_{b,k_1})$ can be derived as

$$\begin{aligned} p(\mathbf{x}_{b,k_1}) &= p(\mathbf{x}_{b,k_1} \in R_m(\hat{X}_w) | \hat{X}_w) && (11) \\ &= \int p(\hat{X}_w) \mathbb{1}(\mathbf{x}_b, \hat{X}_w) d\hat{X}_w, && (12) \end{aligned}$$

where $\mathbb{1}(\mathbf{x}_b, \hat{X}_w) = \begin{cases} 1, & \text{if } \mathbf{x}_{b,k_1} \in R_m(\hat{X}_w) \\ 0, & \text{otherwise.} \end{cases}$

Except for $\mathbf{x}_{b,k_1}$, other states are deterministic given $\mathbf{x}_{b,k_1}$ and $\bar{X}_w$. The minimum energy CAPM problem formulation in (1) is reduced to the following:

$$\min_{\mathbf{x}_{b,k_1}, \mathbf{x}_{b,k_3}^l} \quad c_0 + p(\mathbf{x}_{b,k_1}) c_u + (1 - p(\mathbf{x}_{b,k_1})) c_l \tag{13}$$

$$\text{s.t.} \quad \mathbf{x}_{b,k_1} \in \mathcal{R}_o(R_w^{\text{TROI}}) \text{ and } \mathbf{x}_{b,k_3}^l \in \mathcal{R}_m(X_w), \tag{14}$$

where costs $c_0 = c(\mathbf{x}_{b,0}, \mathbf{x}_{b,k_1})$, $c_u = c(\mathbf{x}_{b,k_1}, \mathbf{x}_{b,k_{\text{max}}})$, and $c_l = c(\mathbf{x}_{b,k_1}, \mathbf{x}_{b,k_3}^l) + c(\mathbf{x}_{b,k_3}^l, \mathbf{x}_{b,k_{\text{max}}})$. (14) are region constraints specified in Sec. IV-B.

In addition, the state $\mathbf{x}_{b,k_{\text{max}}}$ is the final state in which the robot body should arrive. This state can be determined by the current target and the next target's TROI, e.g. the middle point between two target's center. If the current target is the last target, $\mathbf{x}_{b,k_{\max}}$ is the predefined end pose given by the user. By solving the above optimization, we obtain the key states for the robot base. Also, finding the best manipulator pose for a given base pose and intermediate states beyond key states is trivial, and we omit the details here. A viable method is proposed in [26]. After the robot executes the key states $\mathbf{x}_{b,k_1}$ and $\mathbf{x}_{e,k_2}$ to perform active perception, the precise MPOI $\bar{X}_w$ can be perceived. If the planned location for the next state $\mathbf{x}^l_{b,k_3}$ does not satisfy the EPMC constraint, re-planning is needed to find $\mathbf{x}^l_{b,k_3}$. The re-planning objective is to minimize c_l subject to $\mathbf{x}^l_{b,k_3} \in \mathcal{R}_m(\bar{X}_w)$.

V. Experiments

In the experiment, the mobile manipulator is a Boston Dynamic Spot Mini™ with a Unitree Z1™ arm. The Spot Mini™ with all attached accessories weighs 40 kg. Specifically, the Unitree Z1™ arm only weighs 4.0 kg, which consumes much less power than the base. We use IKFast [27] to compute the inverse kinematics for the task constraint regions $\mathcal{R}_o$ and $\mathcal{R}_m$. The experiment includes two parts: the simulation for the comparison of algorithm performance with naive algorithms and physical experiments.

A. Simulation

In the simulation, we introduce the baseline algorithms, the naive planners (Alg. a) and a variant of our algorithm, the decoupled version (Alg. b) to compare with the proposed algorithm (Alg. c) in Sec. IV-E. All algorithms minimize the expected energy cost given start/end state $\mathbf{x}_{k_0}$, $\mathbf{x}_{k_{\max}}$.

a. Deterministic planner. This planner considers the geometric center of TROI X_w as the MPOI, that is, assume $\bar{X}_w = X_w$ and solve $\min_{\mathbf{x}_{b,k_1}} c(\mathbf{x}_{b,k_0}, \mathbf{x}_{b,k_1}) + c(\mathbf{x}_{b,k_1}, \mathbf{x}_{b,k_{\max}})$, subject to $\mathbf{x}_{b,k_1} \in \mathcal{R}_m(X_w)$.
b. Decoupled active perception and manipulation planner. This planner has an action sequence of the lower branch in Fig. 3, without considering the possibility of the upper branch in the figure when solving the key states.
c. CAPM Planner that solves (13).

Other experiment parameters are chosen on the basis of typical field conditions. We consider the operating height of Spot Mini to be fixed at 0.8 m and region constraints are reduced to ring-shaped regions parameterized by the inner and outer circular boundary as in Figs. 4b and 4d. We randomly generate $N = 1000$ different TROI $X_w = (x_w, y_w, 0)$ within the $3 \times 3\text{m}^2$ workspace with radius $r_w \in [0.2\text{m}, 0.3\text{m}]$. The MPOI is sampled through $\bar{X}_w \sim \mathcal{N}(X_w, \Sigma_w)$ where $\Sigma_w = r_w \mathbf{I}_{2\times 2}$ the energy cost coefficient $\gamma = 2$. To simulate the scenario with different weed density, for each sample TROI we generate 5 different path lengths. The total number of scene instances is $5N$.

To compare the baseline and variants of our algorithm, we employ two performance metrics: success rate (%) and average energy cost (Avg(c)). Since our experiment is based on the application of weed flaming and flaming is a static manipulation without direct contact, the success of the task is defined as the pose of the manipulator that satisfies the EPMC constraint (5) with $\varepsilon_{\min} = 0.05$ and $\varepsilon_{\max} = 0.10$. The success rate is the ratio between the number of successful trials and the total number of trials $5N$. The result is listed in Table II. It is clear that although the baseline algorithm (Alg. a) achieves a lower energy cost without considering the active perception task, its success rate is the lowest due to the lack of MPOI information. The decoupled active perception and manipulation planner (Alg.b) successfully executes all tasks, but the energy cost is higher than that of our proposed CAPM planner (Alg.c). To compare the energy savings of the coupled version algorithm with the decoupled version, we define the energy saving coefficient as $\eta_{bc} = (\text{Avg}(c)_b - \text{Avg}(c)_c)/\text{Avg}(c)_c$ where subscripts a, b, and c refer to Alg.a, Alg.b, and Alg.c, respectively. It is clear that the CAPM algorithm works better when the distance traveled between targets is short (i.e., dense weed distribution). This is desirable.

TABLE II: Simulation results of the 3-algorithm comparison

		Avg(c) at different path length (m)				
Alg.	%	2.75	3.25	3.75	4.25	4.75
a	38	3.29	3.56	4.01	4.21	4.52
b	100	4.35	4.62	4.89	5.18	5.48
c	100	3.82	4.20	4.46	4.73	5.09
η_{bc}		**0.14**	0.10	0.10	0.09	0.08

B. Physical Experiment

We evaluate our system on the physical platform. The CAPM planner is able to efficiently and precisely burn down the weeds. More details are given in the video attachment.

VI. Conclusion and Future Work

We presented a CAPM problem where a mobile manipulator must obtain a close-up view before performing its manipulation task. Due to the fact that the perception results determine the follow-up manipulation, the proposed task planning approach can exploit the overlapping observation and manipulation reachable sets to reduce the overall expected energy usage while guaranteeing the task success rate. We implemented our CAPM algorithm and tested it in both simulation and physical experiments. The results confirmed our design and showed that our algorithm has a lower energy consumption compared to a typical two-stage decoupled approach while still maintaining a success rate for the task 100%. In the future, we will improve the proposed algorithm to handle multiple TROI and multiple mobile manipulator problems.

Acknowledgement

The authors would like to thank Dylan Shell, Jason O'Kane, Wenping Wang, Fengzhi Guo, Aaron Kingery, Yingtao Jiang, and Hojun Ji for their inputs and feedbacks.

References

[1] J. Qian, V. Chatrath, J. Yang, J. Servos, A. P. Schoellig, and S. L. Waslander, "Pocd: Probabilistic object-level change detection and volumetric mapping in semi-static scenes," *arXiv preprint arXiv:2205.01202*, 2022.

[2] J. Dong, J. G. Burnham, B. Boots, G. Rains, and F. Dellaert, "4d crop monitoring: Spatio-temporal reconstruction for agriculture," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 3878–3885.

[3] S. Xie, C. Hu, M. Bagavathiannan, and D. Song, "Toward robotic weed control: Detection of nutsedge weed in bermudagrass turf using inaccurate and insufficient training data," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7365–7372, 2021.

[4] R. Bajcsy, "Active perception," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 966–1005, 1988.

[5] R. Bajcsy, Y. Aloimonos, and J. K. Tsotsos, "Revisiting active perception," *Autonomous Robots*, vol. 42, pp. 177–196, 2018.

[6] N. Atanasov, B. Sankaran, J. Le Ny, G. J. Pappas, and K. Daniilidis, "Nonmyopic view planning for active object classification and pose estimation," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1078–1090, 2014.

[7] E. Safronov, N. Piga, M. Colledanchise, and L. Natale, "Active perception for ambiguous objects classification," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 4437–4444.

[8] L. Bartolomei, L. Teixeira, and M. Chli, "Semantic-aware active perception for uavs using deep reinforcement learning," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 3101–3108.

[9] G. Best, J. Faigl, and R. Fitch, "Multi-robot path planning for budgeted active perception with self-organising maps," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 3164–3171.

[10] M. Hegedus, K. Gupta, and M. Mehrandezh, "Towards an integrated autonomous data-driven grasping system with a mobile manipulator," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 1601–1607.

[11] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active vision," *International journal of computer vision*, vol. 1, pp. 333–356, 1988.

[12] S. Thakar, P. Rajendran, A. M. Kabir, and S. K. Gupta, "Manipulator motion planning for part pickup and transport operations from a moving base," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 1, pp. 191–206, 2022.

[13] M. Mittal, D. Hoeller, F. Farshidian, M. Hutter, and A. Garg, "Articulated object interaction in unknown scenes with whole-body mobile manipulation," 2022.

[14] A. Adu-Bredu, N. Devraj, P.-H. Lin, Z. Zeng, and O. C. Jenkins, "Probabilistic inference in planning for partially observable long horizon problems," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 3154–3161.

[15] C. R. Garrett, C. Paxton, T. Lozano-Pérez, L. P. Kaelbling, and D. Fox, "Online replanning in belief space for partially observable task and motion problems," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5678–5684.

[16] G. V. Nardari, A. Cohen, S. W. Chen, X. Liu, V. Arcot, R. A. F. Romero, and V. Kumar, "Place recognition in forests with urquhart tessellations," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 279–286, 2021.

[17] E. Marks, F. Magistri, and C. Stachniss, "Precise 3d reconstruction of plants from uav imagery combining bundle adjustment and template matching," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Philadelphia, May 2022, pp. 2259–2265.

[18] C. Hu, S. Xie, D. Song, J. A. Thomasson, R. G. H. IV, and M. Bagavathiannan, "Algorithm and system development for robotic micro-volume herbicide spray towards precision weed management," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 633–11 640, 2022.

[19] A. N. Sivakumar, S. Modi, M. V. Gasparino, C. Ellis, A. E. Baquero Velasquez, G. Chowdhary, and S. Gupta, "Learned visual navigation for under-canopy agricultural robots," in *Proc. Robot.: Sci. Syst.*, Virtual, July 2021.

[20] P. Maini, B. M. Gonultas, and V. Isler, "Online coverage planning for an autonomous weed mowing robot with curvature constraints," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 5445–5452, 2022.

[21] X. Liu, G. V. Nardari, F. C. Ojeda, Y. Tao, A. Zhou, T. Donnelly, C. Qu, S. W. Chen, R. A. F. Romero, C. J. Taylor, and V. Kumar, "Large-scale autonomous flight with real-time semantic slam under dense forest canopy," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 5512–5519, 2022.

[22] T. C. Thayer, S. Vougioukas, K. Goldberg, and S. Carpin, "Multirobot routing algorithms for robots operating in vineyards," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 3, pp. 1184–1194, July 2020.

[23] W. McAllister, J. Whitman, J. Varghese, A. Davis, and G. Chowdhary, "Agbots 3.0: Adaptive weed growth prediction for mechanical weeding agbots," *IEEE Trans. Robot.*, vol. 38, no. 1, pp. 556–568, 2022.

[24] A. Üneri, M. A. Balicki, J. Handa, P. Gehlbach, R. H. Taylor, and I. Iordachita, "New steady-hand eye robot with micro-force sensing for vitreoretinal surgery," in *2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2010, pp. 814–819.

[25] J. J. Kuffner, "Effective sampling and distance metrics for 3d rigid body path planning," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 4. IEEE, 2004, pp. 3993–3998.

[26] A. Mohammed, B. Schmidt, L. Wang, and L. Gao, "Minimizing energy consumption for robot arm movement," *Procedia CIRP*, vol. 25, pp. 400–405, 2014, 8th International Conference on Digital Enterprise Technology - DET 2014 Disruptive Innovation in Manufacturing Engineering towards the 4th Industrial Revolution.

[27] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, Carnegie Mellon University, Robotics Institute, August 2010.