2013 21st Mediterranean Conference on Control & Automation (MED)
Platanias-Chania, Crete, Greece, June 25-28, 2013

Towards time-optimal exploration and control by an autonomous robot

Vladislav Nenchev and Jörg Raisch

Abstract— **In this paper, we address the problem of an autonomous robotic vehicle collecting a finite but unknown number of objects with non-negligible masses and unknown locations in a restricted area and moving them to a particular spot in minimum time. An adaptive certainty-equivalent navigation and control policy is introduced based on a pick-up and an exploration/drop-off mode. While the input signal in pick-up mode is easily obtained in real time, complete exploration and drop-off corresponds to a hybrid optimal control problem (OCP) with exponential complexity in the finitely discretized space. We propose a trajectory planning algorithm by restricting the motion of the robot to a finite weighted graph. Further, we describe a discrete-time approximation of the hybrid OCP and compare both approaches with respect to computational complexity and accuracy.**

I. Introduction

In dynamic vehicle routing problems critical information is acquired during mission execution and the path needs to be replanned online [1]. Exploration needs to be performed in a structured way to ensure complete coverage of (un)known regions, while minimizing a performance criterion (e.g. time or energy). Typical applications thereof are automated cleaning and coating, field harvesting or demining, lawn mowing etc.

The single depot vehicle routing problem corresponding to computing the shortest closed tour from a depot visiting a given number of delivery/loading points is a well defined optimization problem. In contrast, finding a finite number of objects with non-negligible masses in a restricted area and delivering them to a particular spot, while taking into account the change in dynamics caused upon object pick-up and drop-off and minimizing the overall time, represents a hybrid OCP with limited knowledge. The supervisory control framework introduced in [2] addresses a closely related setup based on random exploration and optimization of a quantitative high-level language measure instead of "physical" costs. Combined hybrid identification of a-priori unknown environmental constraints (e.g. borders) and single goal directed agent control has been investigated in [3]. Learning the partially unknown behavior of elements in the environment has been integrated efficiently in an optimal control scheme for a surveillance and request servicing task [4].

In this paper we address minimum time exploration and control for a point vehicle with fourth order dynamics by an adaptive certainty-equivalent feedback policy based on a pick-up and an exploration/drop-off mode. Assuming an unknown number of objects, the exploration/drop-off costs will be of significant importance for the overall assignment time. Complete coverage of a bounded space can be achieved by a finite waypoint parametrization and traversal, related to a traveling salesman problem (TSP). The time-optimal hybrid OCP of visiting K points by a vehicle was studied in [5], [6]. Due to the inherent complexity of waypoint sequencing, the exact solution is only feasible for small problem instants. TSP-like problems for double integrators were investigated both in worst-case and stochastic setups in [7].

The input signal in pick-up mode can be easily computed in real time. Time-optimal exploration/drop-off for a finitely parametrized unexplored region corresponds to a search over possible waypoint and discrete state sequences. Optimal control of autonomous piecewise affine switched systems has been addressed by a two-stage iterative approach and by dynamic programming providing a state feedback control law in [8]. The exponential autonomous switching problem for a partitioned state space has been tackled by employing necessary optimality conditions from the hybrid maximum principle (as an extension of Pontryagin's work) to compute an approximation of the cost-to-go [9]. Alternatively, complex trajectory planning can be achieved by concatinating a finite number of parametrized primitives that encode simple and stereotypical motions [10], [11], [12]. As a remedy for complexity, we restrict the motion of the robot to a finite set of suitable maneuvers between neighboring waypoints (similar to [13]). The resulting restricted OCP is related to a generalized TSP on a finite directed weighted graph. The shortest path problem can be efficiently solved by state-of-the-art branch and bound methods. The proposed scheme resembles hierarchical control methods based on discrete abstractions [14], [15]. We also describe a standard discrete-time approximation of the hybrid OCP and compare both solutions in a numerical simulation setup. We start with a definition of the hybrid OCP.

II. Problem statement

Consider a mobile robot that has to find a collection of static point objects $O = \{o_1, \dots, o_d\}$, where d is finite but unknown, in a compact domain $Y \subset \mathbb{R}^2$, and move them to a marked safe spot (depot) $y_s \in Y$ in minimum time. An object $o_l, l \in \{1, \dots, d\}$ is characterized by its unknown initial position $y_0^l \in Y$ and its known mass $m^l \in \mathbb{R}$. Assume that the robot has accurate localization in a fixed reference

V. Nenchev and J. Raisch are with *Technische Universität Berlin*, Control Systems Group, Einsteinufer 17, Sekr. EN 11, 10587 Berlin, Germany. Corresponding email: *nenchev@control.tu-berlin.de*.

J. Raisch is with *Max Planck Institute for Dynamics of Complex Technical Systems*, Systems and Control Theory Group, Sandtorstr. 1, 39106 Magdeburg, Germany.

978-1-4799-0997-1/13/$31.00 ©2013 IEEE

frame and dynamics given by

$$\dot{x}(t) = f_q(x,u) = \underbrace{\begin{pmatrix} 0_2 & I_2 \\ 0_2 & -\frac{\mu}{m_q} I_2 \end{pmatrix}}_{A_q} x(t) + \underbrace{\begin{pmatrix} 0_2 \\ \frac{1}{m_q} I_2 \end{pmatrix}}_{B_q} u(t),$$
$$y(t) = \begin{pmatrix} I_2 & 0_2 \end{pmatrix} x(t), \|u(t)\|_2 \leq u_{\max}, \tag{1}$$

where 0_2 denotes a zero matrix of size 2×2 and I_2 the 2-dimensional identity matrix, μ is proportional to the friction coefficient and $u_{\max}$ is the maximum input force. Let the objects currently carried by the robot form a set $O_q \subseteq O$. Then, the current overall mass is given by $m_q = m_\emptyset + \sum_l m^l, o_l \in O_q$, where $m_\emptyset$ stands for the nominal mass of the robot. An object pick-up is carried out with zero velocity at any y_0^l, accompanied by a change of the overall mass. Another switch is enforced at the depot y_s upon dropping all objects resulting in a reset to nominal dynamics. Let the robot's sensing range with perfect sensor performance around its current position $y(t)$ be restricted to a square

$$\omega_{y(t)} = \{z \in \mathbb{R}^2 | \begin{pmatrix} -1 & 0 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \end{pmatrix} z \leq \begin{pmatrix} -1 & 0 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \end{pmatrix} y(t) + \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \rho\}, \tag{2}$$

where $\rho \in \mathbb{R}^+$. The region covered along the trajectory $y|_{[0,t]}$ is $\Omega|_{[0,t]}$. The detection of an object o_l corresponds to an uncontrollable event σ_l, triggered when $y_0^l \in \Omega|_{[0,t]}, y_0^l \notin \Omega|_{[0,\tau]}, \tau < t$. The role of $\Sigma = \{\sigma_1, \ldots, \sigma_d\}$ is solely to provide discrete measurement information to the control policy described in Section III. The corresponding model is given as follows.

Definition 1: The system is modeled as a hybrid automaton, i.e., a 9-tuple $H^c = (Q, X, U, F, E, \text{Inv}, G, R, \text{Init})$ with

- $Q \subseteq 2^O$ – the finite set of discrete states;
- $X \subset \mathbb{R}^4$ – the continuous state space;
- $U \subset \mathbb{R}^2$ – the continuous input space;
- $F = \{f_q\}_{q \in Q}$ – the collection of vector fields $f_q : X \times U \to X$ describing the robot's dynamics, where $f_\emptyset$ represents the nominal dynamics;
- $E = E_1 \cup E_2 \subseteq (Q \times Q) \cup (Q \times \emptyset)$ – the set of discrete state transitions, where $e_1 \in E_1$ denotes a pick-up switching with
$$E_1 = \{(q_1, q_2) | q_2 = q_1 \cup \{o_l\}, o_l \in O \setminus q_1\},$$
and $e_2 \in E_2$ a drop-off switching with
$$E_2 = \{(q_1, \emptyset) | q_1 \neq \emptyset\};$$
- $\text{Inv} : Q \to 2^X$ – the invariant map with
$$\text{Inv}(q) = \{x \in X | x \neq [y_0^{l\,\prime}\ 0']'\} \wedge \{x \in X | x \neq [y_s'\ 0']'\},$$
where y_0^l is the initial position of any object not picked up so far;
- $G : E \to 2^X$ – the guard map
$$G(e_1 = (q_1, q_2 = q_1 \cup \{o_l\})) = \{x \in X | x = [y_0^{l\,\prime}\ 0']'\},$$
$$G(e_2) = \{x \in X | x = [y_s'\ 0']'\}$$
where $e_1 \in E_1, e_2 \in E_2$ and y_0^l is the position of the object being picked up in the event e_1;
- $R : E \times X \to X$ – the reset map with $R(e, x) = x, \forall (e, x) \in E \times G(e)$;
- $\text{Init} = (x_0, \emptyset)$ – the initial state.

The hybrid OCP is to achieve the following in minimum time: determine the initial positions y_0^l of all objects (exploration) and drive the robot and all objects to the depot (control). Due to limited sight and the lack of a priori knowledge of y_0^l, an optimal solution is intractable. Thus, we propose an adaptive certainty-equivalent feedback control policy that guarantees complete exploration and provides a suboptimal solution by a heuristic decomposition of the hybrid OCP.

III. CONTROL POLICY

We suggest a bimodal adaptive feedback control policy consisting of a pick-up and an exploration/drop-off mode for the hybrid OCP as depicted in Fig. 1. If the robot is in exploration/drop-off mode and detects an object $o_l \in O$ at time t corresponding to the occurrence of the event σ_l, this triggers an instant transition to pick-up mode.

A. Pick-up mode

In pick-up mode, the robot moves to the location of the detected object y_0^l as fast as possible, corresponding to the solution of the following two point boundary value problem (TPBVP).

Subproblem 1:

$$\min_u J_l, \text{ where } J_l = T_l,$$
$$\text{s.t. } (1), x(t) = [y(t)'\ \dot{y}(t)']', x(t + T_l) = [y_0^{l\,\prime}\ 0']'.$$

If the robot is in pick-up mode after σ_l has occurred and detects other objects $o_{l_1}, l_1 \in \{1, \ldots, d\} \setminus l$, it remains in this mode until o_l is picked up and all o_{l_1} are put in a temporary queue set $O_p \subseteq O$. We adopt a heuristic policy where the robot collects all $o_{l_1} \in O_p$ in a last-in-first-out manner. In the case of simultaneous object detection, a random pick-up order is chosen. If $O_p = \emptyset$ and o_l has been picked up, the robot switches to exploration/drop-off mode.

B. Exploration/drop-off mode

The problem in this step is complete coverage of the remaining unexplored region in minimal expected time t_f, including a drop-off of the currently carried objects at the depot. To guarantee that all objects are discovered, the robot needs to explore the whole previously uncovered space $Y \setminus \Omega|_{[0,t]}$. We suggest a complete regular discretization of the latter by the finite set of waypoints $W = \{w^{(1)}, \ldots, w^{(K)}\}$ with $\cup_k \omega_{w^{(k)}} \supseteq Y \setminus \Omega|_{[0,t]}, k \in \{1, \ldots, K\}$. Then, coverage represents a constrained multi-point boundary value problem (MPBVP) over W given as follows.

Subproblem 2:

$$\min_u J_e, \text{ where } J_e = \tau_f + \sum_{i=0}^{K-1} \int_{t_i}^{t_{i+1}} dt \text{ with } \tau_f = \int_{t_K}^{t_f} dt,$$

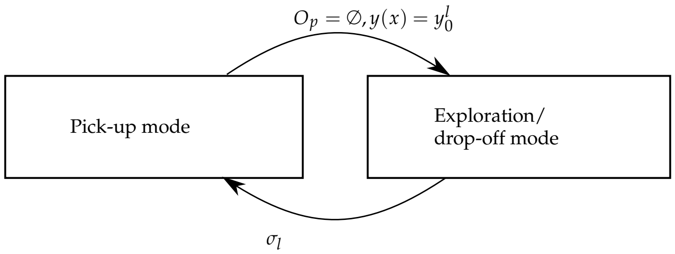


Fig. 1. The proposed bimodal adaptive feedback control policy for an arbitrary object o_l.

$t_0 = t$ denotes the starting time instant, $t_i, i > 0$ is the time when $w^{(i)} \in W$ is reached for the first time and t_f is the final time when the robot arrives at the depot. In exploration/drop-off mode the hybrid system evolves in a subset of its discrete state set, namely $Q_0 = \{q(t_0), \emptyset\} \subset Q$. The initial state is given by $(x_0, q_0) = ([y_0' \; \dot{y}_0']', q(t_0))$ and the final state by $(x_f, q_f) = ([y_s' \; 0']', \emptyset)$. The drop-off corresponding to the discrete state switching occurs at the unknown time t_s with $x(t_s) = [y_s' \; 0']'$, where $t_s \in (t_0, t_f]$.

While in pick-up mode the control can be determined by solving Subproblem 1 upon object detection in real time, solving Subproblem 2 is computationally demanding due to the differential (inertia), geometric (limited sensor range) and switching (discrete states) constraints. Thus, computing an exploration/drop-off solution based on the current overall mass m_q and the remaining unexplored space will be the focus of the following sections.

IV. Restricted exploration and drop-off

To reduce the planning domain, the mobile robot's motion is restricted to a finite number of time-optimal elementary maneuvers. Trajectory plans can be obtained by concatenating compatible maneuvers.

A. Elementary maneuvers

In contrast to a finite input discretization often employed in motion planning approaches, let the velocity of the mobile robot be constrained to a finite set $\mathcal{V}$ at all waypoints. Pairs of a waypoint $w^{(k)} \in W$ and a velocity $v^{(n)} \in \mathcal{V}, n \in \{1, \ldots, N\}$ will be referred to as admissible waypoint configurations $(w^{(k)}, v^{(n)})$. Due to the desired performance goal, define the velocity set as

$$\mathcal{V} = \left\{ \begin{array}{ccc} (\tilde{v}, \tilde{v})', & (\hat{v}, 0)', & (\tilde{v}, -\tilde{v})', \\ (0, \hat{v})', & (0, 0)', & (0, -\hat{v})', \\ (-\tilde{v}, \tilde{v})', & (-\hat{v}, 0)', & (-\tilde{v}, -\tilde{v})' \end{array} \right\}, \quad (3)$$

with $\tilde{v} = \dfrac{\hat{v}}{\sqrt{2}}$ and $0 < \hat{v} < \dfrac{u_{\max}}{\mu}$.

Naturally, the cardinality of $\mathcal{V}$ has a great impact on both the accuracy and complexity of the restricted solution compared to the non-restricted one. While a larger set $\mathcal{V}$ provides better accuracy, it also leads to an increased complexity. Choosing an appropriate discretization for the velocity and a "good" $\hat{v}$ depends on the particular problem setup.

An elementary maneuver is the time-optimal transition between an initial waypoint configuration $x^{(i)} = (w^{(k_i)}, v^{(n_i)})$ and a final waypoint configuration $x^{(j)} = (w^{(k_j)}, v^{(n_j)})$, where $k_{i,j} \in \{1, \ldots, K\}, n_{i,j} \in \{1, \ldots, N\}$, $w^{(k_i)}, w^{(k_j)} \in W, v^{(n_i)}, v^{(n_j)} \in \mathcal{V}$ for system (1) in discrete state $\tilde{q} \in Q_0$ with duration $d_{i,j}^{\tilde{q}}$. A maneuver can be computed offline by solving the corresponding minimum time TPBVP either by direct or indirect optimization methods. As the set of waypoint configurations $W \times \mathcal{V}$ is finite, all possible maneuvers form a finite set $\mathcal{M}$. Two maneuvers $(w^{(k_i)}, v^{(n_i)}), (w^{(k_j)}, v^{(n_j)})$ and $(w^{(\bar{k}_i)}, v^{(\bar{n}_i)}), (w^{(\bar{k}_j)}, v^{(\bar{n}_j)})$ can be concatinated, if $w^{(k_j)} = w^{(\bar{k}_i)}$ and $v^{(n_j)} = v^{(\bar{n}_i)}$, i.e., the final waypoint configuration of the first maneuver is equal to the initial waypoint configuration of the following maneuver. Compatible maneuver sequences can be captured by a graph.

B. Restricted discrete search

The restricted motion of the robot in discrete state q_0 can be represented by a directed weighted graph $H^{q_0} = (\mathcal{X}^{q_0}, \mathcal{M}^{q_0}, D^{q_0})$ with

- $\mathcal{X}^{q_0} = \xi_0 \cup (W \times \mathcal{V} \times \{q_0\}) \cup \xi_f^{q_0}$ – the set of nodes representing possible waypoint configurations, where W is the waypoint set and $\mathcal{V}$ the velocity set as given in (3), $\xi_0 := (x_0, q_0)$ the initial and $\xi_f^{q_0} := ([y_s' \; 0']', q_0)$ the final node;
- $\mathcal{M}^{q_0} \subset \mathcal{X}^{q_0} \times \mathcal{X}^{q_0} = M_0^{q_0} \cup M_1^{q_0} \cup M_2^{q_0}$ – the set of edges representing admissible maneuvers. In particular, $M_0^{q_0}$ is the set of maneuvers between the initial node ξ_0 and an arbitrary waypoint configuration $\xi_i^{q_0} := ((w^{(k_i)}, v^{(n_i)}), q_0) \in \mathcal{X}^{q_0}$, i.e.,
$$M_0^{q_0} = \{(\xi_0, \xi_i^{q_0})\}. \quad (4)$$
The set $M_1^{q_0}$ contains edges between nodes denoting different waypoint configurations, i.e.,
$$M_1^{q_0} = \{(\xi_i^{q_0}, \xi_j^{q_0}\}, \quad (5)$$
with $\xi_j^{q_0} := ((w^{(k_j)}, v^{(n_j)}), q_0) \in \mathcal{X}^{q_0}, k_i \neq k_j$. Finally, $M_j^{q_0}$ are maneuvers connecting arbitrary nodes $\xi_i^{q_0}$ and the final node, i.e.,
$$M_2^{q_0} = \{(\xi_i^{q_0}, \xi_f^{q_0})\}; \quad (6)$$
- $D^{q_0} : \mathcal{M}^{q_0} \to \mathbb{R}^+$ associating maneuvers with their corresponding durations.

Similarly, the motion graph in discrete state $\emptyset$ is $H^{\emptyset} = (\mathcal{X}^{\emptyset}, \mathcal{M}^{\emptyset}, D^{\emptyset})$ with $\mathcal{X}^{\emptyset} = (W \times \mathcal{V} \times \{\emptyset\}) \cup \xi_f^{\emptyset}$, where $\xi_f^{\emptyset} := ([y_s' \; 0']', \emptyset)$ is the initial and final node, $\mathcal{M}^{\emptyset} = M_0^{\emptyset} \cup M_1^{\emptyset} \cup M_2^{\emptyset}$, where $M_0^{\emptyset}$ is the set of edges from the initial to arbitrary waypoint configurations $\xi_i^{\emptyset} := ((w^{(k_i)}, v^{(n_i)}), \emptyset) \in \mathcal{X}^{\emptyset}$, i.e., $M_0^{\emptyset} = \{(\xi_f^{\emptyset}, \xi_i^{\emptyset})\}$. $M_1^{\emptyset}$ is the set of edges between waypoint configurations defined analogously to (5), and $M_2^{\emptyset}$ is the set of maneuvers from arbitrary waypoint configurations to the final node, i.e., $M_2^{\emptyset} = \{(\xi_i^{\emptyset}, \xi_f^{\emptyset})\}$, and $D^{\emptyset} : \mathcal{M}^{\emptyset} \to \mathbb{R}^+$ associates costs (durations) to maneuvers.

To decribe the overall motion, both graphs are combined to a graph $H = (\mathcal{X}, \mathcal{M}, D)$ with $\mathcal{X} = \mathcal{X}^{q_0} \cup \mathcal{X}^{\emptyset}$, $\mathcal{M} =$

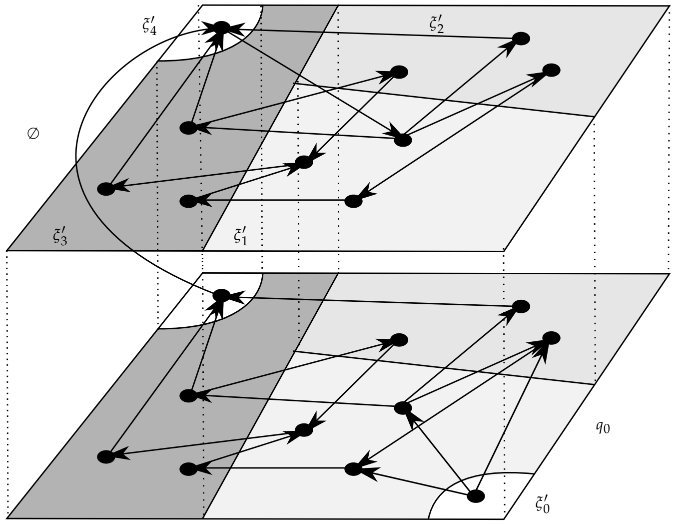


Fig. 2. Exemplary structure of H for $|W| = |\mathcal{V}| = 3$. Each node subset ξ'_k contains nodes on both layers, denoted by corresponding gray schades.

$\mathcal{M}^{q_0} \cup \mathcal{M}^{\emptyset} \cup m_s$, where $m_s = (\xi_f^{q_0}, \xi_f^{\emptyset})$ is the connecting edge between the final node of H^{q_0} and the initial node of $H^{\emptyset}$, denoting the discrete state switch at the depot, and D capturing the edge costs. Let $\mathcal{X}$ be partitioned into subsets $\xi'_k, k \in \{0, \dots, K+1\}$ with $\mathcal{X}' = \{\xi'_0, \dots, \xi'_{K+1}\}$, where $\xi'_0 = \{\xi_0\}$ is the initial state, the set $\xi'_k = (\{w^{(k)}\} \times \mathcal{V} \times \{q_0\}) \cup (\{w^{(k)}\} \times \mathcal{V} \times \{\emptyset\})$ contains all configurations in waypoint $w^{(k)}$ and $\xi'_{K+1} = \{\xi_f^{q_0}, \xi_f^{\emptyset}\}$ is the final node set. The solution of the restricted OCP is the shortest path in H starting in ξ'_0 that visits at least one node in every ξ'_k and ends in ξ'_{K+1}. An exemplary structure of H for $|W| = |\mathcal{V}| = 3$ is depicted in Fig. 2. The lower layer in H represents the possible motion of the robot in discrete state q_0. If there are waypoint sets ξ'_k that have not been visited upon reaching $\xi_f^{q_0}$, the graph traversal continues on the upper layer corresponding to the discrete state $\emptyset$ and ends at $\xi_f^{\emptyset}$. Because of the potentially large size of $\mathcal{X}$, the discrete search problem was solved by a mixed-integer program (MIP).

C. Implementation

The search is related to generalized traveling salesman problems often emerging in vehicle routing. Mathematical programming provides a suitable framework for tackling such problems (e.g. [16]). In particular, branch-and-bound methods, based on computing upper and lower bounds at every search node and pruning branches when the lower bound of the cost-to-go exceeds any computed upper bound at the current tree level, have proven to be very efficient.

To simplify notation, relabel waypoint configurations in $\mathcal{X}^{q_0}$ and $\mathcal{X}^{\emptyset}$ identically by their corresponding indices, i.e.,

$$\left.\begin{aligned} \xi_i^{q_0} &= ((w^{(k_i)}, v^{(n_i)}), q_0) \\ \xi_i^{\emptyset} &= ((w^{(k_i)}, v^{(n_i)}), \emptyset) \end{aligned}\right\} \to i \in \{1, \dots, \underbrace{KN}_{\tilde{n}}\}$$

and, further, assign

$$\begin{aligned} \xi_f^{q_0}, \xi_f^{\emptyset} &\to \tilde{n}+1, \\ \xi_0^{q_0} &\to \tilde{n}+2. \end{aligned}$$

Let $D^{q_0} \in \mathbb{R}^{(\tilde{n}+2)\times(\tilde{n}+2)}$ and $D^{\emptyset} \in \mathbb{R}^{(\tilde{n}+1)\times(\tilde{n}+1)}$ be the precedence matrices of the graphs H^{q_0} and $H^{\emptyset}$, respectively, i.e., $d_{i,j}^{q_0}$ is the cost associated with the edge $(i,j) \in \mathcal{M}^{q_0}$ and $d_{i,j}^{\emptyset}$ is the weight of edge $(i,j) \in \mathcal{M}^{\emptyset}$. The edge m_s denoting the transition from H^{q_0} to $H^{\emptyset}$ has zero duration, while nonexistent edges in H correspond to a very large number. The overall cost of a path in H is the sum of the weights of the edges it contains, given by the functional

$$J_{\mathcal{M}} = \sum_{i=1}^{\tilde{n}+2}\sum_{j=1}^{\tilde{n}+2} d_{i,j}^{q_0} b_{i,j}^{q_0} + \sum_{i=1}^{\tilde{n}+1}\sum_{j=1}^{\tilde{n}+1} d_{i,j}^{\emptyset} b_{i,j}^{\emptyset} \tag{7}$$

with variables $b_{i,j}^{q_0}, b_{i,j}^{\emptyset} \in \mathbb{N}_0$ denoting the number of traversals of every edge in $\mathcal{M}^{q_0}$ and $\mathcal{M}^{\emptyset}$, respectively.

A path is subject to the edge constraints

$$\sum_{i=1}^{\tilde{n}+1} b_{\tilde{n}+2,i}^{q_0} = 1 \tag{8}$$

guaranteeing a single transition from the initial node. To guarantee complete coverage, we require that in each set of nodes ξ'_k, there exists a node i with at least one outgoing edge to another node $j \in \mathcal{X} \backslash \xi'_k$, and, that there exists at least one incoming edge from any node in a different partition to a node in ξ'_k, corresponding to

$$\begin{aligned} \sum_{i \in \xi'_k} \sum_{j \in \mathcal{X} \backslash \xi'_k} (b_{i,j}^{q_0} + b_{i,j}^{\emptyset}) \geq 1, \\ \sum_{i \in \mathcal{X} \backslash \xi'_k} \sum_{j \in \xi'_k} (b_{i,j}^{q_0} + b_{i,j}^{\emptyset}) \geq 1. \end{aligned} \tag{9}$$

The entering and the leaving node of a partition ξ'_k need to be identical to provide a connected solution, captured by

$$\forall j \in \xi'_k, \sum_{i \in \mathcal{X} \backslash \xi'_k} b_{i,j}^{q_0} = \sum_{i \in \mathcal{X} \backslash \xi'_k} b_{j,i}^{q_0}, \sum_{i \in \mathcal{X} \backslash \xi'_k} b_{i,j}^{\emptyset} = \sum_{i \in \mathcal{X} \backslash \xi'_k} b_{j,i}^{\emptyset}. \tag{10}$$

The path in the subgraph H^{q_0} ends at the final node with

$$b_{\tilde{n}+2,\tilde{n}+1}^{q_0} + \sum_{i=1}^{\tilde{n}} b_{i,\tilde{n}+1}^{q_0} = 1. \tag{11}$$

Any path in the subgraph $H^{\emptyset}$ starts and ends at $\tilde{n}+1$, i.e.,

$$\sum_{i=1}^{\tilde{n}} b_{\tilde{n}+1,i}^{\emptyset} = \sum_{i=1}^{\tilde{n}} b_{i,\tilde{n}+1}^{\emptyset} \tag{12}$$

restricting the number of outgoing and incoming edges to be equal.

The overall number of nodes in H is $2(KN+1)+1$ and finding the shortest path is NP-hard. However, due to the relaxation (of the classical generalized TSP) that edges can be traversed more than once and the relatively sparse graph structure, the problem can be solved efficiently.

For comparison we provide a standard discrete-time approximation of the non-restricted hybrid OCP in the following section.

V. Discrete-time approximation

Assuming equidistant time discretization, the exploration and drop-off problem can be tackled by a linear MIP over a pre-specified optimization horizon N_{max}. As we drop the velocity discretization (3) assumed for the restricted OCP, we expect that for a sufficiently fine step size the resulting optimal cost will, in general, be lower than the cost of the restricted OCP, at the price of a significantly higher computational complexity.

A. Formulation

Let $i, j \in \{1, N_{\text{max}}-1\}$ denote time instants and $x_i = x(i)$ in the following. The intermediate depot visit at time t_s can be captured by a boolean vector b_s with

$$b_{s_i} = \begin{cases} 1, & \text{if } x_i = [y_s' \ 0']' \wedge b_{s_j} = 0, \forall j < i, \\ 0, & \text{else.} \end{cases} \tag{13}$$

Let the dynamics (1) be exactly discretized with sampling time t_{sam} leading to linear difference equations with matrices $\bar{A}_q$ and $\bar{B}_q$. For the discrete states q_0 and $\emptyset$ the switch of the dynamics can be realized by the constraints

$$\forall i, x_{i+1} = \begin{cases} \bar{A}_{q_0} x_i - \bar{B}_{q_0} u_i, & \text{if } b_{s_j} = 0, \forall j \le i, \\ \bar{A}_{\emptyset} x_i - \bar{B}_{\emptyset} u_i, & \text{else,} \end{cases} \tag{14}$$

implying that the system evolves in discrete state q_0 until $b_{s_j} = 1$, when it switches to and remains in $\emptyset$ thereafter. Similar to [17], the nonlinear input constraints can be approximated by a regular C-sided polygon, leading to

$$\forall i, u_{i,1} \sin\left(\frac{2\pi c}{C}\right) + u_{i,2} \cos\left(\frac{2\pi c}{C}\right) \le u_{\text{max}}, \tag{15}$$

$\forall c \in [1, \ldots, C]$. Traversing a waypoint $w^{(k)}$ can be captured by K binary vectors with

$$b_{k_i} = \begin{cases} 1, & \text{if } y(x_i) = w^{(k)}, \\ 0, & \text{else,} \end{cases} \tag{16}$$

and complete coverage is provided when every $w^{(k)}$ is visited at least once, corresponding to

$$\sum_{i=1}^{N_{\text{max}}-1} b_{k_i} \ge 1, \forall k. \tag{17}$$

Similarly, the final time constraint can be captured by a boolean vector

$$b_{f_i} = \begin{cases} 1, & \text{if } x_i = x_f \wedge \sum_{j=1}^{i-1} b_{k_j} \ge 1, \forall k, \\ 0, & \text{else,} \end{cases} \tag{18}$$

reflecting that all waypoints need to be visited at least once before ending at the depot.

B. Optimization

The approximate hybrid OCP can be formulated as

$$\begin{aligned} \min_u \ J_d &= \sum_{i=1}^{N_{\text{max}-1}} i b_{f_i}, \\ &\text{s.t. (13)} - \text{(18)}, \end{aligned} \tag{19}$$

where the cost is the minimum time index. The constraints (13), (14), (16) and (18) can be implemented by big-M formulations, commonly employed in MIP, where the corresponding relaxation coefficients should be chosen according to the setup.

Obtaining a solution of (19) requires solving a MPBVP for every possible waypoint sequence. Even by employing state-of-the-art branch-and-bound methods, the worst case computational burden is more than polynomial in both the optimization span length N_{max} and the number of waypoints $|W|$.

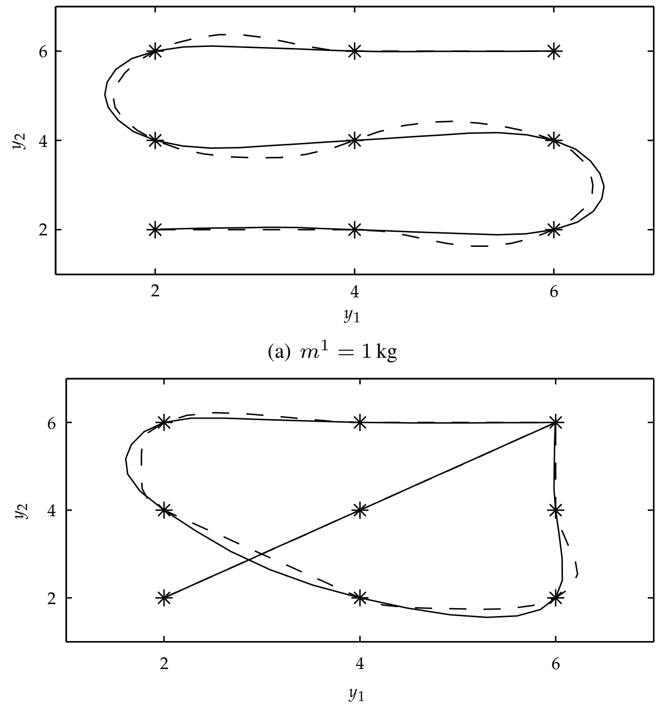
In the next section we compare the solution of the restricted OCP with the non-restricted OCP in a simple numerical setup.

VI. Numerical example

Consider a robot with dynamics (1), parameters $m_\emptyset = 3\,\text{kg}$, $\mu = 0.8\,\text{kg/s}$, $u_{\text{max}} = 0.3\,\text{kg m/s}^2$ and $t_{\text{sam}} = 1\,\text{s}$ and sight region (2) with $\rho = 1$. The proposed exploration/drop-off approaches are applied to a setup with $|W| = 8$ previously uncovered waypoints and an object being picked up in $x_0 = [y_0' \ 0']'$. The mass of the object is $m^1 = 1\,\text{kg}$ in the first and $m^2 = 3\,\text{kg}$ in the second simulation. For the restricted OCP $\hat{v}$ in (3) is set to the maximum velocity that can be reached upon a transition of the vehicle from an arbitrary waypoint with zero velocity to its nearest neighboring waypoint.

The simulation was done with the commercial solver CPLEX using YALMIP [18] as an interface in MATLAB on an AMD Athlon 64 X2 2GHz processor with 3 GB RAM. By exploiting the symmetry of the chosen setup, computing the relevant maneuvers offline by direct shooting with t_{sam} took approximately 1 min per discrete state $q \in Q$. The robot's paths for the restricted and the non-restricted approximate OCP are shown for two different masses in Fig. 3. The corresponding durations for the different masses and the mean computation times over 10 simulations in each case are summarized in Table I.

Note that the optimal waypoint sequence depends on the overall mass of the robot. For light objects exploration is finished without an intermediate drop-off, while the optimal solution when carrying heavy objects includes an intermediate depot visit. The qualitative and quantitative results of both methods are similar, while the computational expense of the maneuver-based method is smaller by an order of magnitude of 2. The advantage of the restricted OCP grows with the problem size. Due to the MIP formulation a good suboptimal solution guess is provided after a few seconds even for large problems.



(a) $m^1 = 1\,\text{kg}$

(b) $m^2 = 3\,\text{kg}$

Fig. 3. Non-restricted (solid) and maneuver-based (dashed) robot paths for an 8-waypoint (*) setup with initial position $y_0 = w^{(1)} = [2\ 2]'$ and depot $y_s = [6\ 6]'$ for different masses m^i.

TABLE I

PATH DURATIONS FOR DIFFERENT MASSES m_o WITH AVERAGE COMPUTATION TIMES t^{COMP}.

m^i [kg]	$J_d\,[s]$	$t_d^{\text{comp}}\,[s]$	$J_{\mathcal{M}}\,[s]$	$t_{\mathcal{M}}^{\text{comp}}\,[s]$
1	51	3092	64	40
3	59	3101	75	47

VII. CONCLUSIONS

We addressed the hybrid OCP of finding, collecting and securing a finite number of mass objects by an adaptive certainty-equivalent feedback control policy consisting of a pick-up and an exploration/drop-off mode. To assure coverage of the area, a suitable number of waypoints is selected. Constraining the motion of the robot in the exploration/drop-off mode to suitable maneuvers results in a restricted OCP that can be solved by graph search. A discrete-time approximation of the non-restricted OCP has been provided for comparison. Both approaches have been implemented by MIP and solved by branch-and-bound methods. The computational advantage of the maneuver-based method at the price of a comparably small performance loss has been shown in a simple numerical setup. The acquired maneuver sequence can also be used as an admissible initial guess for post-optimization.

ACKNOWLEDGMENTS

The authors would like to thank Marc Toussaint and Oliver Brock for inspiring discussions in the initial phase of the work.

REFERENCES

[1] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. Smith, "Dynamic vehicle routing for robotic systems," *Proceedings of the IEEE*, vol. 99, pp. 1482–1504, 2011.

[2] X. Wang, G. Mallapragada, and A. Ray, "Language-measure-based supervisory control of a mobile robot," in *Proceedings of the 2005 American Control Conference*, june 2005, pp. 4897 – 4902 vol. 7.

[3] D. Austin, "Simultaneous identification and control of a hybrid dynamic model for a mobile robot," in *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. 4, 2000, pp. 3138 – 3143.

[4] Y. Chen, J. Tumova, and C. Belta, "Ltl robot motion control based on automata learning of environmental dynamics," in *Proc. of ICRA*, 2012, pp. 5177–5182.

[5] M. Buss, M. Glocker, M. Hardt, O. von Stryk, R. Bulirsch, and G. Schmidt, "Nonlinear hybrid dynamical systems : modelling, optimal control, and applications," *Lecture notes in control and information sciences*, vol. 279, pp. 311–335, 2002.

[6] M. Glocker and O. von Stryk, "Hybrid optimal control of motorized traveling salesmen and beyond," in *Proceedings of the 15th IFAC World Congress On Automatic Control*, 2002, pp. 21–26.

[7] K. Savla, F. Bullo, and E. Frazzoli, "Traveling salesperson problems for a double integrator," *Automatic Control, IEEE Transactions on*, vol. 54, no. 4, pp. 788 –793, april 2009.

[8] C. Seatzu, D. Corona, A. Giua, and A. Bemporad, "Optimal control of continuous-time switched affine systems," *IEEE Transactions on Automatic Control*, vol. 51, pp. 726–741, 2006.

[9] B. Passenberg, P. E. Caines, M. Sobotka, O. Stursberg, and M. Buss, "The minimum principle for hybrid systems with partitioned state space and unspecified discrete state sequence," in *Proceedings of the 49th IEEE Conference on Decision and Control*, 2010.

[10] E. Frazzoli and F. Bullo, "On quantization and optimal control of dynamical systems with symmetries," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 1, 2002, pp. 817 – 823 vol.1.

[11] A. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," in *Advances in neural information processing systems 15*. Cambridge, MA: MIT Press, 2003, pp. 1547–1554.

[12] K. Flaßkamp, S. Ober-Blöbaum, and M. Kobilarov, "Solving optimal control problems by exploiting inherent dynamical systems structures," *Journal of Nonlinear Science*, 2012.

[13] M. Pivtoraiko and A. Kelly, "Kinodynamic motion planning with state lattice motion primitives," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.

[14] T. Moor, J. Raisch, and J. Davoren, "Admissibility criteria for a hierarchical design of hybrid control systems." in *Proc. ADHS03 - IFAC Conference on Analysis and Design of Hybrid Systems*, 2003, pp. 389–394.

[15] H. Kress-Gazit, G. Fainekos, and G. Pappas, "Temporal-logic-based reactive mission and motion planning," *Robotics, IEEE Transactions on*, vol. 25, no. 6, pp. 1370 –1381, 2009.

[16] I. Kara and T. Bektas, "Integer linear programming formulation of the generalized vehicle routing problem," in *Proceedings of 5th EURO/INFORMS Joint International Meeting, Istanbul, Turkey*, 2003.

[17] A. Richards and J. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," in *Proceedings of the 2002 American Control Conference*, vol. 3, 2002, pp. 1936 – 1941 vol.3.

[18] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.