2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

A Unified Closed-Loop Motion Planning Approach for an I-AUV in Cluttered Environment with Localization Uncertainty

Huan Yu[1,2], Wenjie Lu[2], Dikai Liu[2]

***Abstract*— This paper presents a unified motion planning approach for an Intervention Autonomous Underwater Vehicle (I-AUV) in a cluttered environment with localization uncertainty. With the uncertainty being propagated by an information filter, a trajectory optimization problem closed by a Linear-Quadratic-Gaussian controller is formulated for a coupled design of optimal trajectory, localization, and control. Due to the presence of obstacles or complexity of the cluttered environment, a set of feasible initial I-AUV trajectories covering multiple homotopy classes are required by optimization solvers. Parameterized through polynomials, the initial base trajectories are from solving quasi-quadratic optimization problems that are linearly constrained by waypoints from RRTconnect, while the initial trajectories of the manipulator are generated by a null space saturation controller. Simulations on an I-AUV with a 3 DOF manipulator in cluttered underwater environments demonstrated that initial trajectories are generated efficiently and that optimal and collision-free I-AUV trajectories with low state uncertainty are obtained.**

***Index Terms*— Intervention AUV, Motion planning, Uncertainty minimization**

I. Introduction

Remotely operated underwater vehicles equipped with manipulators have been used in practical applications, such as shipwreck search and rescue, underwater structure maintenance, and biological sampling. However, their performance is limited by the skills of the operators. In recent years, development of underwater perception and navigation technologies, such as Sonar, Ultra-short Baseline and cameras, has made it possible to develop Intervention Autonomous Underwater Vehicles (I-AUVs) for enhanced performance. These systems have been developed and studied in many projects, including ALIVE [1], SAUVIM [2], TRIDENT [3], TRITON [4], and PANDORA [5].

Two-stage planning approaches have been used by researchers, where trajectory of the body (or base) is first planned and followed by trajectory of the manipulator. However, such approaches cannot provide optimal base and manipulator trajectories or might not be able to find feasible trajectories in a cluttered underwater environment. Therefore, unified motion planning for underwater mobile manipulator has drawn a lot of attention recently. As shown in [6], sampling-based methods (such as FMT*, RRT*) and optimization based approaches (such as CHOMP [7], STOMP [8]) all show their ability to plan feasible and local optimal paths.

[1]Huan Yu is with School of Automation, Beijing Institute of Technology, Beijing, China.

[2]Huan Yu, Wenjie Lu, Dikai Liu are with Centre for Autonomous Systems, University of Technology Sydney, Australia {Huan.Yu, Wenjie.Lu, Dikai.Liu}@uts.edu.au.

It might be computationally expensive to sample a large number of nodes in high-dimensional state space for considering I-AUV kinematic constraints directly. While being able to construct trajectories that satisfy kinematic constraints, CHOMP and STOMP are more computational intensive than sampling based methods [6]. Other studies have presented genetic algorithms, particle swarm optimization, and ant colonization for unified I-AUV motion planning [9], [10].

Nonetheless, different from a ground robot, the accuracy in localizing an I-AUV cannot be guaranteed due to the limits of the sensors and perception algorithms. The localization uncertainty is not considered in these motion planning approaches, thus the resultant trajectories might not be achievable. In addition, due to un-modeled motion noise and observation noise, the I-AUV state may deviate from the planned trajectory, making the planned trajectory unsuitable. In this case, replanning has to be conducted within a short time.

This paper seeks for efficient unified I-AUV motion planning under uncertainties (also known as planning in belief space). While such planning could be formulated as a Partially Observable Markov Decision Processes (POMDP) [11], state-of-art POMDP solvers suffer from the curse of dimensionality and history for planning in high-dimensional belief space. Feedback-based information roadmap (FIRM) proposed in [12] transfers POMDP into MDP by sampling in the belief space and generating a belief-graph. Although it breaks the curse of history, a huge number of nodes still have to be sampled in the belief space for acceptable optimality, resulting in intractable computational complexity in planning I-AUV motion.

Another popular formulation for belief space planning is based on Linear Quadratic Gaussian (LQG) or Receding Horizon Control (RHC) control, such as LQG-MP [13], iLQG [14], MLO [15], T-LQG [16], Covariance-free RHC [17]. Those methods define a stochastic control problem and utilize decoupled (LQG-MP, iLQG, Covariance-free RHC) or coupled (MLO, T-LQG) design of trajectory and control policy. Stochastic control based planning methods as mentioned in [16] usually are defined as a Nonlinear Programming Problem (NLP). Detailed comparison of those methods in computation complexity can be found in [16]. It shows that T-LQG has the lowest computational complexity $O(Kn^3)$ through Sequential Quadratic Programming (SQP), with n being state dimensions and K being horizon steps.

Similar to T-LQG, this paper adopts the separation principle of LQG controller, which provides a trajectory with best estimation performance. As opposed to a Kalman filter

in T-LQG [16], the proposed approach utilizes an information filter to propagate localization uncertainty. A Linear-Quadratic-Gaussian control problem in closed-loop feedback policies space considering Localization uncertainty (referred to as a L-LQG problem) is then formulated for a coupled design of optimal trajectory, localization, and control. This problem is further solved by SQP.

However, due to the presence of obstacles, all collision-free trajectories are grouped by homotopy classes. such that the trajectories can be transformed into each other within the same class by gradual variation without colliding obstacles [18]. Therefore to find an global optimal solution via SQP or other gradient based solvers, at least one candidate from each homotopy class is used as an initial trajectory in solving the L-LQG problems. Then, the solution with the optimal value among all resultant solutions is the global optimal trajectory.

This work proposes a novel efficient approach to generate feasible I-AUV initial trajectories. In addition, these initial trajectories are obtained from solving a reduced planning problem, which are generated by neglecting localization uncertainty from the L-LQG problems. The resultant initial trajectories are feasible and near-optimal and thus could be more efficient in solving the L-LQG problems.

Specifically, the initial trajectories of the base of an I-AUV are parameterized by polynomials in time and thus can be described as a linear combinations of parameters after the time horizon being discretized. Then a linearly constrained quasi-quadratic optimization problem is formulated and solved for base trajectories [19]. The constraints are the waypoints generated from RRTconnect [20] for covering multiple homotopy classes. After that, the initial manipulator trajectories are generated by using a Null Space Saturation (NSS) controller given the base trajectories [21]. Such decomposition is able to keep the property of linearity in parameters in representing trajectories, for the purpose of formulating a quasi-quadratic optimization problem, and yet still be able to find feasible and near-optimal initial trajectories for solving the L-LQG problem efficiently.

The remainder of this paper is organized as follows: we first present problem definition of I-AUV motion planning in Section II. Section III presents detailed description of the proposed I-AUV motion planning approach. After that, Section IV summarizes simulation results and analysis. At last, Section V gives our discussion and future works.

II. Problem Definition

I-AUV studied in this work is composed of a 6 Degree Of Freedom (DOF) free-floating base and a n DOF manipulator. Let $\mathbf{x} = [\mathbf{x}^B; \mathbf{x}^M] \in \mathbb{X} \subset \mathbb{R}^{6+n}$ denote the I-AUV state, including pose and attitude of base (B) and joint angles of the manipulator (M). The equations of motion in discrete time can be written as:

$$\begin{aligned} \mathbf{x}_{t+1} &= f(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\omega}_t) \\ \mathbf{z}_t &= h(\mathbf{x}_t, \boldsymbol{\nu}_t), \end{aligned} \tag{1}$$

where f and h are the motion and observation model. Let $\mathbf{u}_t = [\mathbf{u}_t^B; \mathbf{u}_t^M] \in \mathbb{U} \subset \mathbb{R}^{n_u}$ is the n_u-dimensional control input, $\mathbf{z}_t \in \mathbb{Z} \subset \mathbb{Z}^{n_z}$ is the n_z-dimensional observation model, $\boldsymbol{\omega}_t$ and $\boldsymbol{\nu}_t$ are zero mean Gaussian noises, i.e., $\boldsymbol{\omega}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\omega}_t})$, $\boldsymbol{\nu}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}_t})$.

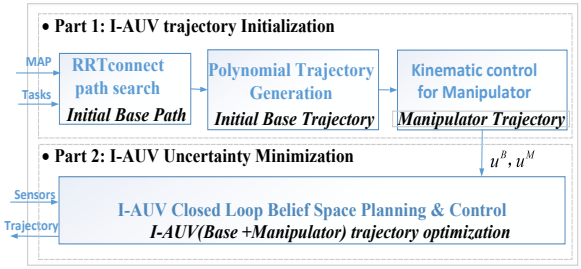


Fig. 1: A unified I-AUV motion planning Approach

Due to the partial and noisy observations and lack of accurate motion model, it might be difficult to obtain the accurate I-AUV state, resulting in the localization uncertainty. Therefore, the trajectory planning problem is built in a belief space. *Belief* $b[\mathbf{x}_t] \in \mathbb{B}$ is the conditional distribution of $\mathbf{x}_t$ and represented by a Gaussian distribution in belief space $\mathbb{B}$ [22]:

$$\begin{aligned} b[\mathbf{x}_t] &= p[\mathbf{x}_t | \mathbf{u}_0, ..., \mathbf{u}_{t-1}; \mathbf{z}_0, ..., \mathbf{z}_t] \ , \\ b[\mathbf{x}_{t+1}] &= \tau(b[\mathbf{x}_t], \mathbf{u}_t, \mathbf{z}_t) \ , \end{aligned} \tag{2}$$

where $b[\mathbf{x}_{t+1}] : \mathbb{X} \times \mathbb{Z} \times \mathbb{U} \to \mathbb{R}$ and $\tau : \mathbb{B} \times \mathbb{U} \times \mathbb{Z} \to \mathbb{B}$ is a belief propagation function. In this work, given maximum likelihood observation and Gaussian noise assumption, belief can be denoted by mean and covariance, and propagation of belief can be based on filtering method (like Kalman filter). In the remainder of this paper, let $\mathbf{b}_t$ denote $b[\mathbf{x}_t]$.

In this work, we define the following planning problem:

***Planning problem 1*:** Find the optimal control policy $\pi_t : \mathbb{B} \to \mathbb{U}$ for all $0 \leq t \leq K$ that minimizes the objective function J^π:

$$\begin{aligned} \min_{\pi} J^\pi &:= \mathbb{E}[\sum_{t=0}^{K-1} c_t^\pi(\mathbf{b}_t, \mathbf{u}_t) + c_K^\pi(\mathbf{b}_K)] \\ s.t.\ \mathbf{b}_{t+1} &= \tau(\mathbf{b}_t, \mathbf{u}_t, \mathbf{z}_{t+1}), \end{aligned} \tag{3}$$

where c_t^π is the instant cost at time step t, which could be a measure on I-AUV state uncertainty (denoted as $\mathcal{C}_{uncertainty,t}$), control effort (denoted as $\mathcal{C}_{control,t}$), and/or penalty from other constraints. Index K is the terminal time step and is fixed, which could also be a variable to be optimized [19], [23].

III. Unified I-AUV Motion Planning Approach

As shown in Fig. 1, the proposed approach consists of two parts (I-AUV trajectory initialization and unified I-AUV trajectory optimization). The first part provides collision-free initial I-AUV trajectories (control sequence $\mathbf{u}^B$ and $\mathbf{u}^M$) for solving L-LQG problems in the second part. In the first part, as opposed to being searched in $(6+n)$-dimension configuration space or higher dimensional state space, the initial I-AUV trajectories are decomposed into base and manipulator trajectories and are found through three steps: sampling, optimization and adaptation. The second part solves a unified L-LQG problem with the initial trajectories

from the first part. The construction of feedback gain matrix L_t in controller is solved in [16]. Note that replanning is triggered if the deviation between current I-AUV state and the planned state exceeds a given threshold. The algorithm is summarized in Algorithm 1.

A. Part 1: I-AUV Trajectory Initialization

In this part, as shown in Fig. 1, we proposed a hierarchy initial trajectories search method including three steps:

- First step: Waypoints search via RRTconnect. Thus we can obtain a set of paths (represented by waypoints) that cover multiple different homotopy classes.
- Second step: Base trajectory optimization. Parameterized by polynomials and written as a linear combination of parameters, base trajectories can be easily found by solving a quasi-quadratic optimization problem that are linearly constained by waypoints.
- Third step: Manipulator trajectory adaptation. Collision free initial manipulator trajectories are generated by using a NSS controller given the base motion.

1) Waypoints search via RRTconnect: A waypoint path in our work denotes the path connected by waypoints through straight lines. Waypoints are the nodes that the base tries to go through. Multiple initial base trajectories can be generated based on different waypoint paths. In our work, we choose RRTconnect as waypoint search method. To find the waypoints that represent different homotopy class trajectories, a rewire process is adopted, which reconnects the RRTconnect path by reducing the length of the path. A path is then represented by the set of waypoints composing a rewired path. The next subsection will show how to generate continuous-time base trajectories based on those waypoints.

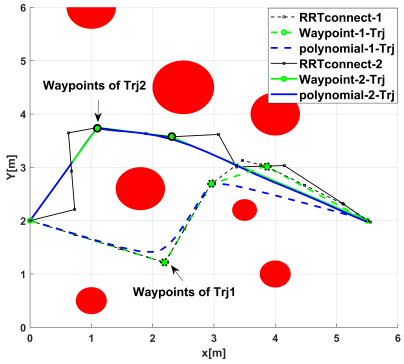


Fig. 2: Two initial base trajectories belong to different homotopy classes. The black curves denote RRTconnect path; the green curves denote the waypoint path based on RRTconnnect path; the blue curves are smooth and collision-free polynomial trajectories based on waypoints.

2) Base trajectory optimization: Continuous-time base trajectory optimization is based on polynomial trajectory representation [24], [25]. In this paper, only the translation coordinates of the base trajectory in $\mathbb{R}^3$ are planned. The trajectory of each coordinate axis is described by an Nth polynomial:

$$f^k(t) = a_0^k + a_1^k t + a_2^k t^2 + a_3^k t^3 + ... + a_N^k t^N \ , \quad (4)$$

and the polynomial coefficients of each axis is:

$$\mathbf{p}^k = [a_0^k \ a_1^k \ a_2^k \ ... \ a_N^k]^T, \qquad k = x, y, z \ . \quad (5)$$

As show in [19], the polynomial is described by end-derivative rather than the coefficients. The mapping of them is denoted as:

$$\mathbf{p} = \mathbf{A}^{-1}\mathbf{M}\begin{bmatrix}\mathbf{d_F}\\ \mathbf{d_P}\end{bmatrix} \ , \quad (6)$$

where $\mathbf{d_P}$ and $\mathbf{d_F}$ denote free and fixed blocks, respectively. $\mathbf{d_P}$ are the optimized variables and $\mathbf{d_F}$ are fixed parameters. The construction of matrices $\mathbf{A}$ and $\mathbf{M}$ is defined in [19], which maps the free and fixed parameters to the polynomial coefficients.

The base trajectories has to be smooth, collision-free, and more importantly close to the waypoints generated, as shown in Fig. 2. Therefore, the objective function is written as:

$$J_p = \min_{\mathbf{d_P}} \ \omega_s J_s + \omega_c J_c + \omega_w J_w \ , \quad (7)$$

The first term J_s is to minimize the jerk or snap of the trajectory. The second term J_c is to ensure a collison-free trajectory. Obstacles push the trajectory away, while waypoints have opposite effect, which attracts the base to them. Therefore, the last term J_w is to reduce the distance between the resultant trajectories and waypoints.

The definitions of J_s and J_c are the same as the ones in [26]. The term J_w is a line integral defined as below:

$$J_w = \int_S a(\boldsymbol{f}(t))ds \ , \quad (8)$$

$$a(d) = \begin{cases} d - \varepsilon_2 + 0.5 * (\varepsilon_2 - \varepsilon_1) & if \ d > \varepsilon_2 \\ \frac{1}{2(\varepsilon_2 - \varepsilon_1)}(d - \varepsilon_1)^2 & if \ \varepsilon_1 \le d \le \varepsilon_2 \\ 0 & if \ 0 < d < \varepsilon_1 \end{cases} \ , \quad (9)$$

where $a(\boldsymbol{f}(t))$ is the attractive force, d is the distance between the base and the nearest waypoint, ε_1 and ε_2 are two distance thresholds.

3) Manipulator trajectory adaptation: A collision-free manipulator trajectory is generated given a base trajectory via a NSS controller. In our work, the primary task for the control is the base trajectory tracking, the secondary task is collision avoidance of manipulator. For all tasks, we define :

$$\boldsymbol{\sigma}_x = \boldsymbol{\sigma}_x(\boldsymbol{\eta}, \boldsymbol{q}) \in \mathbb{R}^m \ , \dot{\boldsymbol{\sigma}}_x = \boldsymbol{J}_x\boldsymbol{\zeta} \quad (10)$$

where m is the dimension of the task, $\boldsymbol{J}_x$ is the corresponding Jacobian matrix, and $\boldsymbol{\zeta}$ is the system velocity [21].
The tracking error and its jacobian of the primary task is given by :

$$\begin{aligned} \boldsymbol{\sigma}_a &= \sqrt{(\boldsymbol{\eta}_{1d} - \boldsymbol{\eta}_1)^T(\boldsymbol{\eta}_{1d} - \boldsymbol{\eta}_1)} \in \mathbb{R}^1 \ , \\ \boldsymbol{J}_a &= -\frac{(\boldsymbol{\eta}_{1d} - \boldsymbol{\eta}_1)^T}{\|\boldsymbol{\eta}_{1d} - \boldsymbol{\eta}_1\|}\boldsymbol{J}_{pos1} \in \mathbb{R}^{1\times 6+n} \ . \end{aligned} \quad (11)$$

While the error and its jacobian of the secondary task is given by:

$$\begin{aligned}\boldsymbol{\sigma}_b &= \epsilon - \sqrt{(\boldsymbol{p}_{obst} - \boldsymbol{\eta}_{ee,1})^T(\boldsymbol{p}_{obst} - \boldsymbol{\eta}_{ee,1})} \in \mathbb{R}^1 \ , \\ \boldsymbol{J}_b &= -\frac{(\boldsymbol{p}_{obst} - \boldsymbol{\eta}_{ee,1})^T}{\|\boldsymbol{p}_{obst} - \boldsymbol{\eta}_{ee,1}\|}\boldsymbol{J}_{pos2} \in \mathbb{R}^{1\times 6+n} \ ,\end{aligned} \tag{12}$$

where $\boldsymbol{\eta}_{1d}$ is the desired position of I-AUV base, $\boldsymbol{J}_{pos1}$ and $\boldsymbol{J}_{pos2}$ are the corresponding Jacobian matrices, $\boldsymbol{p}_{obst}$ is the closest obstacle point of the end effector position $\boldsymbol{\eta}_{ee,1} \in \mathbb{R}^3$, ϵ is the safety distance.

Then, the NSS controller is given as

$$\zeta_r = \boldsymbol{J}_a^+(\boldsymbol{\eta},\boldsymbol{q})(\dot{\sigma}_{a,d} + \boldsymbol{k}_a\tilde{\sigma_a}) + \mathbf{N}_a\boldsymbol{J}_b^+(\boldsymbol{\eta},\boldsymbol{q})(\dot{\sigma}_{b,d} + \boldsymbol{k}_b\tilde{\sigma_b}) \ , \tag{13}$$

where $\boldsymbol{J}_a^+$ and $\boldsymbol{J}_b^+$ are pseudoinverse of the Jacobian matrices $\boldsymbol{J}_a$ and $\boldsymbol{J}_b$, $\mathbf{N}_a$ is the null-space projector. After that, the manipulator trajectory can be generated by the control through following integration:

$$\begin{aligned}\begin{bmatrix} \boldsymbol{\eta}_r(t) \\ \boldsymbol{q}_r(t) \end{bmatrix} &= \int_0^t \begin{bmatrix} \dot{\boldsymbol{\eta}}_r(t) \\ \dot{\boldsymbol{q}}_r(t) \end{bmatrix} d\sigma + \begin{bmatrix} \boldsymbol{\eta}(0) \\ \boldsymbol{q}(0) \end{bmatrix} \\ &= \int_0^t \boldsymbol{J}_k^+(\sigma)\zeta_r(\sigma)d\sigma + \begin{bmatrix} \boldsymbol{\eta}(0) \\ \boldsymbol{q}(0) \end{bmatrix} .\end{aligned} \tag{14}$$

.

B. Part 2: Trajectory-optimized LQG

Although LQG controller can be used to track a given trajectory, different nominal trajectory will produce different control effect due to the noise in motion control and state observation. To plan a trajectory with good control perfermance, the most intuitive way is first generate a lot of feasible trajectories, then find the best one based on the LQG control effect [13]. This way is called decoupled design of trajectory and control. In contrast, a coupled design of trajectory and estimator is proposed in [16], which optimize trajectory with good estimation performance.

Similar to [16], we first linearize the I-AUV system and then formulate a linear quadratic Gassuian system. Let an initial control sequence be denoted by $\{\mathbf{u}_t^p\}_{t=0}^{K-1}$, $\mathbf{x}_{t+1}^p = f(\mathbf{x}_t^p, \mathbf{u}_t^p, 0)$ for $0 \leq t \leq K-1$, where $\mathbf{x}_0^p = \hat{\mathbf{x}}_0$. Linearization of the nonlinear motion models (1) about the initial control and state sequence yeilds

$$\begin{aligned}\tilde{\mathbf{x}}_{t+1} &= \mathbf{A}_t^p\tilde{\mathbf{x}}_t + \mathbf{B}_t^p\tilde{\mathbf{u}}_t + \mathbf{G}_t^p\boldsymbol{\omega}_t \\ \tilde{\mathbf{z}}_t &= \mathbf{H}_t^p\tilde{\mathbf{x}}_t + \mathbf{M}_t^p\boldsymbol{\nu}_t\end{aligned} \tag{15}$$

where $\tilde{\mathbf{x}}_t := \mathbf{x}_t - \mathbf{x}_t^p$, $\tilde{\mathbf{u}}_t := \mathbf{u}_t - \mathbf{u}_t^p$, $\tilde{\mathbf{z}}_t := \mathbf{z}_t - \mathbf{z}_t^p$ are the state, control and observation error. In this work, we use canonical parameterization (an alternative representation) for Gaussian distribution, which is usually described by mean μ and covariance Σ. If we use canonical parameterization, Guassian distribution is described by information matrix Ω and information vector ξ. The equivalence between moments and canonical parameterization is: $\Omega = \Sigma^{-1}$, $\xi = \Sigma^{-1}\mu$. As shown in Eq. (16), propagation of the information matrix Ω is based on information filter [27], rather than Riccati equations in a Kalman filter.

Based on the canonical parameterization, trajectory optimization problem is written as:

***Planning problem 2*:** given a start belief $\mathbf{b}_0$, goal state $\mathbf{x}_g$ and its tolerance radius r_g, time horizon K, we define the following problem:

$$\begin{aligned}\min_{\mathbf{u}_{0:K-1}^p} & \sum_{t=1}^{K}[-logDet(\mathbf{W}_t\Omega_t\mathbf{W}_t^T) + (\mathbf{u}_{t-1}^p)^T\mathbf{W}_t^u\mathbf{u}_{t-1}^p] \\ s.t. \quad & \bar{\Omega}_t = (A_t\Omega_{t-1}^{-1}A_t^T + G_t^p\Sigma_{\boldsymbol{\omega}_t}(G_t)^T), \\ & \Omega_t = H_t^T(M_t^p\Sigma_{\boldsymbol{\nu}_t}(M_t^p)^T)^{-1}H_t + \bar{\Omega}_t, \\ & \Omega_0 = (\Sigma)_0^{-1}, \qquad \mathbf{x}_0^p = \mathbb{E}[\mathbf{b}_0(\mathbf{x})], \\ & \mathbf{x}_{t+1}^p = f(\mathbf{x}_t^p, \mathbf{u}_t^p, 0), \quad 0 \leqslant t \leqslant K-1, \\ & \|\mathbf{u}_t^p\|_2 < r_u, \qquad 0 < t < K, \\ & \|\mathbf{x}_K^p - \mathbf{x}_g\|_2 < r_g, \qquad \mathbf{x}_t \notin \mathcal{O}_{obstacle} \ ,\end{aligned} \tag{16}$$

where $\mathbf{W}_t$ and $\mathbf{W}_t^u = diag([\mathbf{W}_t^{u^B}, \mathbf{W}_t^{u^M}])$ are positive-definite weight matrices. Generally, it is easier to control manipulator than base, thus entries of matrix $\mathbf{W}_t^{u^B}$ are designed be to be larger than the ones in $\mathbf{W}_t^{u^M}$. The first two constraints in Eq. (16) are the information matrix propagation; the third and fourth constraints are initial conditions; the last three constraints are for confining state and control in their limits, respectively. The optimized solution is refer to as $\{\mathbf{u}_{0:K-1}^*, \mathbf{x}_{0:K}^*\}$.

C. Obstacle Cost Definition Virtual Boundary Expansion

To reduce computation complexity, we use a soft obstacle avoidance strategy. An Obstacle cost term $\mathcal{C}_{obs}$ is added to the cost function in Eq. (16). We also assume that the joint motion of manipulator is accurate (manipulator equipped with precise joint encoder). But the base pose uncertainty will be transferred to manipulator through forward kinematics. Therefore, there should be a safe collision avoidance strategy for the manipulator. In our work, we use similar obstacle cost definition used in [7], but we not only take obstacle cost into consideration, but also integrate motion and observation uncertainty of the base into the localization uncertainty of manipulator. The geometry of the I-AUV is first described by a sphere bounding technique as described in [28]. Let $S_j(j = 1, ...M)$ denote the set of spheres with the center (x_j, y_j, z_j) and the radius R_j. The position of the center of each sphere in workspace is $p(\mathbf{x}_k, S_j)$. For each step of the motion $\mathbf{x}_k$, the obstacle penalty function is

$$\mathcal{C}_{obs} = [c(d(\mathbf{x}_k, S_j))]|_{1\leq j\leq M} \ , \tag{17}$$

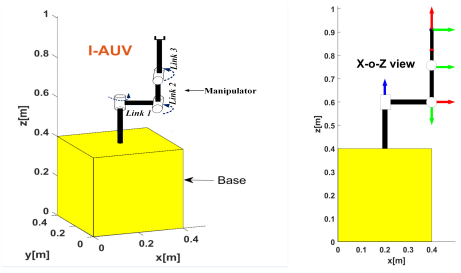
$$c(d) = \begin{cases} -d + \frac{1}{2}\varepsilon & if \ d < 0 \\ \frac{1}{2\varepsilon}(d-\varepsilon)^2\varepsilon & if \ 0 \leq d \leq \varepsilon \\ 0 & if \ d > \varepsilon \end{cases} \ , \tag{18}$$

where ε is the safety distance, d is the distance from center of S_j to the nearest obstacle surface. For motion safety, the obstacle space will be expanded according to the localization uncertainty of the system state. To avoid collision in 99% confidence level, the distance d will be enlarged into $d^{'} =$

$d + 2 \times 3.3682 \times \lambda_1(\Sigma)$, where λ_1 is major axes of 99% confidence ellipsoid [29].

IV. Simulations

In the section, simulations of motion planning for an I-AUV were conducted in MATLAB2016b and SQP was used to solve the L-LQG problem. The simulated I-AUV is illustrated in Fig. 3. It has a 3 link manipulator at the top of base. This I-AUV is designed to conduct cleaning, data collection or inspection on underwater infrastructures or algae growing on the underside of seaice in polar area [30]. The I-AUV generalized coordinates are defined as $\mathbf{x} = [\mathbf{P}^{B^T}, \mathbf{\Psi}^{B^T}, \boldsymbol{\theta}^{M^T}]^T$, where $\mathbf{P}^B = [x, y, z]^T$ denotes the position, $\mathbf{\Psi}^B = [\theta_{pitch}, \gamma_{roll}, \psi_{yaw}]^T$ is attitude. Its generalized velocities are denoted as $\dot{\mathbf{x}} = [\mathbf{v}^{B^T}, \boldsymbol{\omega}^{B^T}, \boldsymbol{\omega}^{M^T}]^T$, $\boldsymbol{\theta}^M = [\theta_1, \theta_2, \theta_3]^T$, $\boldsymbol{\omega}^M = [\omega_{\theta_1}, \omega_{\theta_2}, \omega_{\theta_3}]^T$. The control input is the desired velocities $\mathbf{u} = \dot{\mathbf{x}}$. Specifically, the kinematic equations of motion are: $\mathbf{x}_{k+1} = \mathbf{x}_k + \tau \mathbf{v}_k + M[\mathbf{P}_k] \cdot \boldsymbol{\omega}_t$, $\mathbf{z}_k = \mathbf{x}_k + N[\mathbf{P}_k] \cdot \boldsymbol{\nu}_t$, where τ is the size of discretized time step, $M[\mathbf{P}_k]$ and $N[\mathbf{P}_k]$ scales the motion and observation noise and it is proportional to the robot's state.



(a) Three-D view (b) X-o-Z view

Fig. 3: I-AUV model for simulation

Two simulation scenarios are considered to validate the proposed approach. In all cases, we assume that the observation noise varies in the y-axis: $N[\mathbf{P}_k] = 0.125 * (4.5 - y)$. This noise distribution also is referred to as Light-dark model [14]. In the first scenario, obstacles are all vertical cylinders and thus the manipulator is not required to move given a feasible base trajectory. While the second scenario involves both vertical and horizontal obstacles of cylinder shape, where the initial manipulator trajectory has to be adapted in order to avoid collision with the horizontal cylinder members. To clearly show the motion of I-AUV, an video of the simulations is available at `https://youtu.be/InHvNlPHUaA`.

A. Scenario One

Figure 4 shows the cluttered environment used in the simulation and the result from the proposed unified motion planning approach. Paths from RRTconnect and trajectories from algorithm Part I are referred to as Waypoint-path and polynomial-Trj, respectively. Fig. 4(b) and Fig. 4(c) show the local optimal trajectories (in yellow) obtained by algorithm Part II, given two different initial trajectories from two homotopy classes. The Waypoint-paths and polynomial-Trjs are marked by * and ·, respectively. In this case, the start configuration of the manipulator is the same as the goal, therefore one trival feasible solution is to keep the manipulator configuration in the whole trajectory and change the base pose. While our simulation result clearly shows that the configuration of manipulator changes continuously for optimal planning. The main reason is that to reduce state uncertainty, the I-AUV moved toward and spent more time in area with low observation noise (For example, in Fig. 4(b), the I-AUV spent more time around position $[3.5, 3.5, 1.2]^T$ than $[2, 2, 1.2]^T$) for better localization performance. In this case, by comparing the performance of two trajectories based on Eq. (16), the solution in Fig. 4(c) will be chosen as the planned trajectory for task execution and control.

Algorithm 1 Unified closed-loop I-AUV motion planning Approach

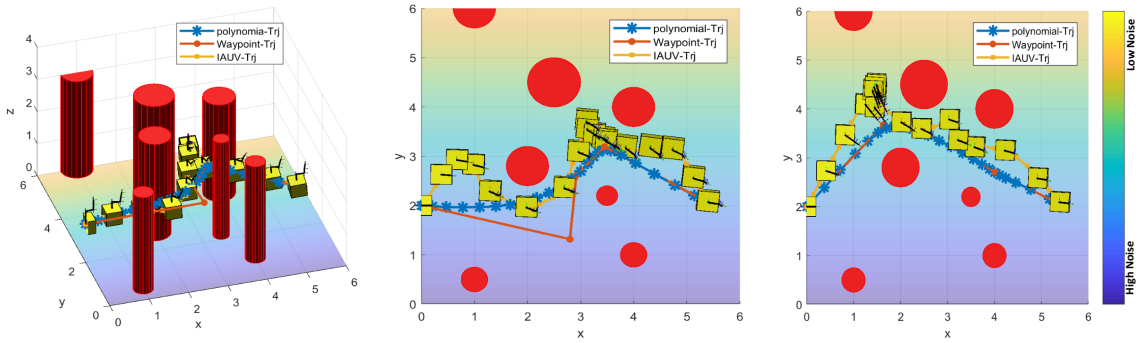
Input: Initial belief $\mathbf{b}_0$, goal state $\mathbf{x}_g$ and tolerance r_g, planning time horizon K, map $\mathcal{M}$

Output: Control sequence $\mathbf{u}_{0:K-1}$ for I-AUV task execution

while $\|\mathbf{x}_t - \mathbf{x}_g\|_2 > r_g$ **do**
  if $d(\mathbf{x}_t, \mathbf{x}_t^o) > \varepsilon$ or $t == 0$ or $t == K$ **then**
    Part 1: I-AUV trajectory initialization. Generate control sequences (such as velocity control or acceleration control) as the initial value for Part 2
    1. *Step1*: Search N waypoints paths from multiple homotopy classes as shown in Fig. 2;
    2. *Step2*: Slove optimization problem defined in Eq. (7) to generate N base polynomial trajectories;
    3. *Step3*: manipulator control sequences search for N base trajectories based on Eq. (13);
    Part 2: Uncertainty minimization
    1. Solve N nomial trajectories based on the problem definition in Eq. (16) and N initial I-AUV trajectories solved in Part 1;
    2. Compare cost of N nominal trajectories based on performance index defined in Eq. (16) and select the best one: $\{\mathbf{u}^*_{0:K-1}, \mathbf{x}^*_{0:K}\}$
  end
  else
    Policy Function: $\hat{\mathbf{x}}_t \leftarrow \mathbb{E}[\mathbf{b}_t]$,
    $\mathbf{u}_t \leftarrow -\mathbf{L}_t(\hat{\mathbf{x}}_t - \mathbf{x}^o_t) + \mathbf{u}^o_t$;
    Task Execution: $\mathbf{x}_{t+1} \leftarrow f(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\omega}_t)$
    State Observation: $\mathbf{z}_{t+1} \leftarrow h(\mathbf{x}_{t+1}, \boldsymbol{\nu}_{t+1})$;
    Belief Propagation: $\mathbf{b}_{t+1} \leftarrow \tau(\mathbf{b}_t, \mathbf{u}_t, \mathbf{z}_{t+1})$;
    $t \leftarrow t + 1$;
  end
end

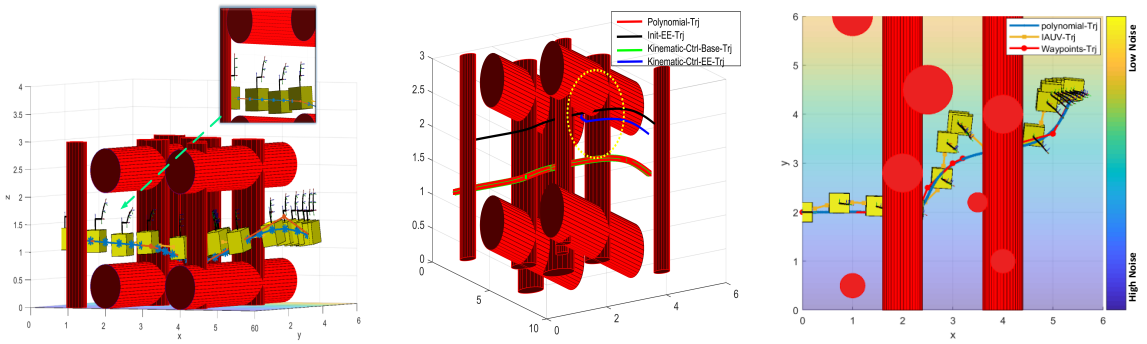
B. Scenario Two

In this scenario, the I-AUV must go through a narrow passage. Fig. 5 shows the manipulator trajectory adaptation by the NSS control for collision avoidance. It clearly shows if the manipulator pose remained the same as the initial state, the initial end-effector trajectory (Init-EE-Trj) would collide with the horizontal obstacle. After the kinematic control based trajectory search for manipulator, a collision-



(a) Case 1: I-AUV 3D planning trajectory (b) Case 1: Local minimal solution . (c) Case 1: Different local minimal solution

Fig. 4: Simulation result based on different initial trajectories. Yellow area has low level noise. (a) Three-D view of the planned trajectories, including waypoints trajectory, polynomial trajectory of base and uncertainty minimization result. (b) is X-o-Y view of (a). (b) and (c) show that the uncertainty minimization results are local optimal trajectories, which depend on the homotopy class of the initial trajectory. In this case, the start state: $\mathbf{x}_0 = [0; 2; 1.2; 0; 0; 0; 0; -\pi/2; 0]$, goal state: $\mathbf{x}_g = [5.5; 2; 1.2; 0; 0; 0; 0; -\pi/2; 0]$.



(a) Case 2: Safe I-AUV trajectory planning in cluttered scenario (b) Case 2: Kinematic control for obstacle avoidance of manipulator (c) Case 2: Planning in cluttered environment(X-o-Y view)

Fig. 5: I-AUV 3D planning trajectory in cluttered environment. (a) Three-D view of the planned trajectory. (b) Kinematic control result. The manipulator will fall down to avoid obstacle and base trajectory is well tracked. (c) shows the X-o-Y view of (a) and the motion of the first link can be seen clearly. In this case, start state: $\mathbf{x}_0 = [0; 2; 1.2; 0; 0; 0; 0; -\pi/2; 0]$, goal state: $\mathbf{x}_g = [5.5; 4.6; 1.2; 0; 0; 0; 0; -\pi/2; 0]$.

free manipulator trajectory was solved as shown in Fig. 5(b) (Kinematic-Ctrl-EE-Trj) and it shows that kinematic control keeps the original base trajectory unchanged. The area zoomed in in Fig. 5(a) shows that although initial collision cost is zero, the manipulator still moved down to guarantee safety due to obstacle expansion described in section III-C. While adjusting the base pose is another way to reduce collision cost, it is more cost-effective to move manipulator considering the cost of moving the base down.

V. Conclusions

This paper presents a unified motion planning approach in belief space for an I-AUV in cluttered environment. The proposed three-step feasible initial trajectory search method is able to find multiple sub-optimal solutions efficiently, through solving a linearly constrained quasi-quadratic optimization problem and using a null space saturation controller. Simulation results demonstrated that a collision-free optimal trajectory with low state uncertainties can be obtained by solving a LQG problem considering localization uncertainty minimization, via sequential quadratic programming. In our simulations, we only considered a simple observation and noise distribution model. More complex noise distribution will be analyzed in our future work. In addition to I-AUVs, the approach proposed in this paper also can be applied to other (ground or aerial) mobile manipulator systems [31], [32]. In the future, the proposed planning method will be tested on an I-AUV system developed at University of Technology Sydney.

Acknowledgement

This work was supported in part by the Australian Research Council (ARC) Linkage Project (LP150100935), the Roads and Maritime Services of NSW, the Centre for Autonomous Systems (CAS) at the University of Technology Sydney, and China Scholarship Council (CSC).

REFERENCES

[1] J. Evans, P. Redmond, C. Plakas, K. Hamilton, and D. Lane, "Autonomous docking for intervention-AUVs using sonar and video-based real-time 3d pose estimation," in *OCEANS 2003*, vol. 4, Sep. 2003, pp. 2201–2210 Vol.4.

[2] G. Marani, S. K. Choi, and J. Yuh, "Underwater autonomous manipulation for intervention missions AUVs," *Ocean Engineering*, vol. 36, no. 1, pp. 15–23, 2009.

[3] P. J. Sanz, P. Ridao, G. Oliver, C. Melchiorri, G. Casalino, C. Silvestre, Y. Petillot, and A. Turetta, "TRIDENT: A framework for autonomous underwater intervention missions with dexterous manipulation capabilities," *IFAC Proceedings Volumes*, vol. 43, no. 16, pp. 187–192, 2010.

[4] P. Cieslak, P. Ridao, and M. Giergiel, "Autonomous underwater panel operation by GIRONA500 UVMS: A practical approach to autonomous underwater manipulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 529–536.

[5] F. Maurelli, M. Carreras, J. Salvi, D. Lane, K. Kyriakopoulos, G. Karras, M. Fox, D. Long, P. Kormushev, and D. Caldwell, "The PANDORA project: A success story in AUV autonomy," in *OCEANS 2016*, April 2016, pp. 1–8.

[6] D. Youakim and P. Ridao, "Motion planning survey for autonomous mobile manipulators underwater manipulator case study," *Robotics and Autonomous Systems*, 2018.

[7] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "CHOMP: Gradient optimization techniques for efficient motion planning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 489–494.

[8] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal, "STOMP: Stochastic trajectory optimization for motion planning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 4569–4574.

[9] A. Alvarez, A. Caiti, and R. Onken, "Evolutionary path planning for autonomous underwater vehicles in a variable ocean," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 2, pp. 418–429, 2004.

[10] M. P. Aghababa, M. H. Amrollahi, and M. Borjkhani, "Application of GA, PSO, and ACO algorithms to path planning of autonomous underwater vehicles," *Journal of Marine Science and Application*, vol. 11, no. 3, pp. 378–386, 2012.

[11] G. Shani, J. Pineau, and R. Kaplow, "A survey of point-based POMDP solvers," *Autonomous Agents and Multi-Agent Systems*, vol. 27, no. 1, pp. 1–51, 2013.

[12] A.-A. Agha-Mohammadi, S. Chakravorty, and N. M. Amato, "FIRM: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements," *The International Journal of Robotics Research*, vol. 33, no. 2, pp. 268–304, 2014.

[13] J. Van Den Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 895–913, 2011.

[14] J. Van Den Berg, S. Patil, and R. Alterovitz, "Motion planning under uncertainty using iterative local optimization in belief space," *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1263–1278, 2012.

[15] R. Platt, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, "Belief space planning assuming maximum likelihood observations," in *Proceedings of Robotics: Science and Systems (RSS)*, June 2010.

[16] M. Rafieisakhaei, S. Chakravorty, and P. R. Kumar, "T-LQG: Closed-loop belief space planning via trajectory-optimized lqg," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 649–656.

[17] S. Patil, G. Kahn, M. Laskey, J. Schulman, K. Goldberg, and P. Abbeel, "Scaling up gaussian belief space planning through covariance-free trajectory optimization and automatic differentiation," in *Algorithmic foundations of robotics XI*. Springer, 2015, pp. 515–533.

[18] L. Palmieri, A. Rudenko, and K. O. Arras, "A fast random walk approach to find diverse paths for robot navigation," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 269–276, 2017.

[19] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Robotics Research*. Springer, 2016, pp. 649–666.

[20] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2000, pp. 995–1001.

[21] G. Antonelli, "Underwater robots, volume 96 of springer tracts in advanced robotics," 2014.

[22] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.

[23] F. Gao, W. Wu, Y. Lin, and S. Shen, "Online safe trajectory generation for quadrotors using fast marching method and bernstein basis polynomial," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 344–351.

[24] H. Seo, S. Kim, and H. J. Kim, "Locally optimal trajectory planning for aerial manipulation in constrained environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 1719–1724.

[25] H. Oleynikova, M. Burri, Z. Taylor, J. Nieto, R. Siegwart, and E. Galceran, "Continuous-time trajectory optimization for online uav replanning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 5332–5339.

[26] H. Oleynikova, Z. Taylor, R. Siegwart, and J. Nieto, "Safe local exploration for replanning in cluttered unknown environments for microaerial vehicles," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1474–1481, 2018.

[27] G. A. Terejanu, "Discrete kalman filter tutorial," *University at Buffalo, Department of Computer Science and Engineering, NY*, vol. 14260, 2013.

[28] J. Dong, M. Mukadam, F. Dellaert, and B. Boots, "Motion planning as probabilistic inference using gaussian processes and factor graphs." in *Robotics: Science and Systems*, vol. 12, 2016.

[29] B. Wang, W. Shi, and Z. Miao, "Confidence analysis of standard deviational ellipse and its extension into higher dimensional euclidean space," *PloS one*, vol. 10, no. 3, p. e0118537, 2015.

[30] J. Woolfrey, D. Liu, and M. Carmichael, "Kinematic control of an autonomous underwater vehicle-manipulator system (AUVMS) using autoregressive prediction of vehicle motion and model predictive control," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 4591–4596.

[31] V. Pilania and K. Gupta, "Mobile manipulator planning under uncertainty in unknown environments," *The International Journal of Robotics Research*, vol. 37, no. 2-3, pp. 316–339, 2018.

[32] M. Tognon, E. Cataldi, H. A. T. Chavez, G. Antonelli, J. Cortés, and A. Franchi, "Control-aware motion planning for task-constrained aerial manipulation," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2478–2484, 2018.