Cooperative Control of Dual-Arm Concentric Tube Continuum Robots

Hanna Jiamei Zhang, Sven Lilge, M. Taha Chikhaoui, Jessica Burgner-Kahrs

Cooperative Control of Dual-Arm Concentric Tube Continuum Robots

Hanna Jiamei Zhang[1], Sven Lilge[1] *Student Member, IEEE*, M. Taha Chikhaoui[2], *Member, IEEE*, and Jessica Burgner-Kahrs[1], *Senior Member, IEEE*

***Abstract*— Thanks to their small size, high dexterity and compliance, concentric tube continuum robots (CTCRs) are well suited for tasks in constrained and tortuous environments such as minimally invasive surgery. Many real-world tasks often require collaboration of multiple manipulators necessitating dual-arm (DA) robotic systems. Leveraging the benefits of CTCRs in a dual-arm configuration, i.e. DA-CTCR, has great potential for applications in such areas. A hierarchy-based control framework is proposed to efficiently control and coordinate motions in a semi-autonomous manner for DA-CTCRs. This framework is validated in various assistive/motion coordination control schemes in simulation and on a real-robot prototype. Our results demonstrate that a DA-CTCR can effectively be controlled when considering the autonomous execution of multiple functionally meaningful tasks aggregated in a prioritized hierarchy. The performance of the task prioritization was clear on the real robot where, for a relative positioning task prioritized over that of a single arm, respective mean errors of 0.38mm and 0.7mm were observed.**

I. Introduction

Concentric tube continuum robots (CTCRs see Fig. 1) are continuum manipulators composed of a flexible backbone leading to elastic deformations that can be described by continuous tangent vectors [1]. CTCRs are composed of two or more precurved elastic tubes of different lengths and curvatures that are inserted into one another. Each tube is independently actuated through translation and rotation at its base. Elastic and mechanical interactions between the tubes during actuation generate a tentacle-like motion. Their diameters of usually less than 2mm make them well suited for minimally invasive surgery. Their inherent compliance and dexterity allows them to work with and around sensitive structures.

One of the difficulties in control of continuum robots stems from the fact that the mapping from actuator to task space is not intuitive. The curvilinear shape of CTCR resulting from non-linear interactions of the elastic tubes, can induce large changes of the robot shape when small end-effector movements are desired. It is unclear to the average operator how tube translations, rotations, and curvatures interact to get a desired state of the robot (i.e. body shape or a relative coordination with another arm/structure). When using an inversion

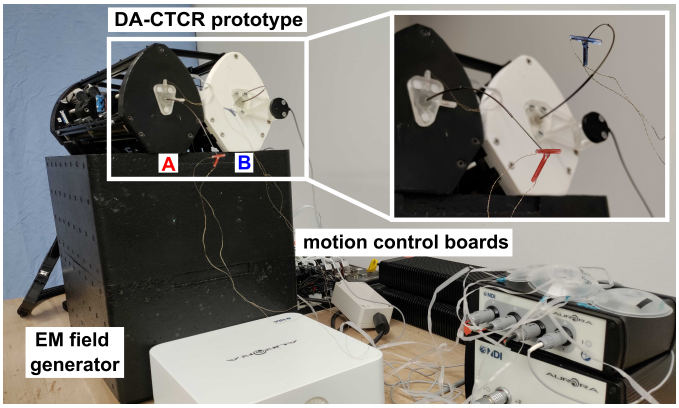

Fig. 1: Experimental setup depicting the two actuation units each with respective three-tube CTCRs. Electromagnetic tracking coils, providing real time pose information, are attached to each robot's end-effector. They are colored red and blue for robot A and B respectively (upper right corner).

of the forward kinematic mapping, desired commands during operation may lead to physically infeasible configurations due to non-invertibility of the forward mapping and resulting robot singularities. These issues are only amplified when considering the control of multiple CTCRs, an example being the dual-arm CTCR (DA-CTCR) which has a common use case in minimally invasive robot assisted surgery. To lessen the cognitive strain on the operator, "following" schemes have been proposed in which the operator is in full teleoperative control of one arm while the other(s) follow/react to the primary arm's motion in an autonomous manner [2]. This work aims to push this concept further specifically for multi-CTCR systems through the introduction of a unified control framework consisting of a variety of prioritized tasks.

A. Related Work

A kinematically redundant manipulator possesses more degrees of freedom than those needed to execute its task. This affords a robot with an increased level of dexterity that may be used to, at the end, avoid singularities, joint limits, and workspace obstacles, as well as to minimize joint torques, energy, or optimize other suitable performance criteria. To do so, the kinematic redundancy of the robot has to be resolved. Siciliano et al. [3] summarizes methods for the kinematic control of redundant serial robot manipulators, including methods based on the robot's differential kinematics.

Local inversion of the differential kinematics is the most common approach to control redundant robots [4]. The main

[1]Continuum Robotics Laboratory, Department of Mathematical & Computational Sciences, University of Toronto, Mississauga, ON L5L 1C6, Canada hannajiamei.zhang@mail.utoronto.ca

[2] Univ. Grenoble Alpes, CNRS, UMR 5525, VetAgro Sup, Grenoble INP, TIMC, 38000 Grenoble, France.

This research was partly funded by the CNRS-University of Toronto Joint Research Programme, the Natural Sciences and Engineering Research Council of Canada (NSERC) [funding reference RGPIN-2019-04846], and ANR-11-LABX-0004-01.

advantages are the simplicity of task design, the possibility of integrating sensor based-feedback, and the easy integration of multiple tasks with execution priorities [5]. Such prioritized controllers guarantee a hierarchy in the execution of multiple ordered tasks, i.e. higher priority tasks will always be executed to the best capability of the system without being affected by lower priority tasks. Such a hierarchy can be achieved through iterative nullspace projection, QR decomposition, quadratic programming, or using the task priority matrix [3], among other techniques/approaches.

Autonomous control of DA-CTCRs is much less explored in the literature compared to multi-arm serial robots. The studies that do exist for multi-arm CTCR only evaluate teleoperation paradigms (level 0 autonomy) with direct position control of each individual CTCR end-effector independent of the other [6], [7]. There exists a large body of control research into redundancy resolution and the dual-arm paradigm for serial-arm robots summarized in [3], [5], however due to the unique morphology of CTCRs it is not guaranteed for such methods to work when applied directly. Furthermore, such work is sparsely investigated in the literature [8]. To this point, in recent work, leveraging the redundancy of DA-CTCRs using nullspace projections for hierarchical task execution has shown success for maintaining a constant distance between both arms during motion [2], as well as for avoiding body collisions [9], however such investigations have been limited only to simulation. Further developments in this area can prove valuable in automating certain (repetitive, low risk, predictable) tasks which unloads some of the cognitive and physical burden off operators and, in surgical settings, have the potential to reduce the procedure duration, trauma, and expense [10]. Furthermore, with respect to all applications of DA-CTCR, automating some tasks can ensure their consistency in execution, as well as save the time and concentration of operators for more complex tasks.

B. Open Challenges & Contributions

While redundancy resolution for the execution of multiple tasks shows promise in DA-CTCRs, its exploration is limited to a few simple examples shown in simulation. Furthermore, consideration and demonstration of task formulations specific to DA-CTCR is lacking as much of the control work in this space translates tasks designed for serial arm manipulators to continuum manipulators. In the proposed DA-CTCR configuration, shown in Fig. 1, various types of autonomous motion coordination between the two end-effectors can be significantly helpful in surgical environments that go beyond the simple end-effector position control.

Building upon our previous work [2], this paper expands on the differential inverse kinematics control framework scheme to facilitate a wider variety of tasks, which can be executed simultaneously with priority through iterative nullspace projections. This will facilitate the more customized control of DA-CTCRs to better suit different operator gestures/actions. Validation of the control approach and task formulations is shown in simulation and on a real robot prototype. In particular, these additional tasks aim to improve the kinematic performance of the DA-CTCR, control the overall shape to avoid body collisions, and control the end-effectors towards desired triangulation configurations.

II. Control Scheme

In order to control a single CTCR's end-effector, a Jacobian $\mathbf{J}$ based technique is employed. For redundant manipulators, a vector $\nabla\boldsymbol{\eta}(\mathbf{q})$ can be projected on the nullspace of the Jacobian to fulfill some non-primary task(s) through a certain robot configuration (robot tube translations and rotations) $\mathbf{q} \in \mathbb{R}^{2n}$ where n is the number of tubes. The nullspace can be computed iteratively to obtain multiple priority levels resulting in a hierarchical control.

A. Modeling

The control framework is model-based and uses the kinematic modeling of a single continuum arm from [11] and the dual-arm robot kinematics derived from those of the serial arm [12], both summarized in [2]. The two arms of the DA-CTCR are referred to as A and B. The Jacobians for the individual 3-tube CTCRs used in this work are $\mathbf{J}_{A/B} \in \mathbb{R}^{6\times6}$ and the corresponding dual-arm relative Jacobian is $\mathbf{J}_D \in \mathbb{R}^{6\times12}$. When performing *single-arm* position or orientation control, only the first or last three rows of the Jacobian respectively are used and referred to as $\mathbf{J}_{A/Bp}$ or $\mathbf{J}_{A/Br} \in \mathbb{R}^{3\times6}$. Similarly when performing *relative* position and orientation control, only the first or last three rows respectively are used and referred to as $\mathbf{J}_{Dp}$ or $\mathbf{J}_{Dr} \in \mathbb{R}^{3\times12}$. This work demonstrates and describes only single and relative arm position control however the framework can be extended to achieve orientation and full pose control. Note that full pose control for CTCRs in general is challenging as end-effector position and orientation are highly coupled.

B. Task Priority Formulation

Task prioritization is achieved by performing a recursive projection on the nullspace of the prior task Jacobian and uses the augmented Jacobian as follows.

$$\mathbf{P}_k = \mathbf{P}_{k-1} - (\mathbf{J}_k\mathbf{P}_{k-1})^\dagger(\mathbf{J}_k\mathbf{P}_{k-1}), \quad \mathbf{P}_0 = \mathbf{I} \tag{1}$$

$$\nabla\boldsymbol{\eta}_k(\mathbf{q}) = \begin{cases} \mathbf{J}_k^\dagger\boldsymbol{\epsilon}_k(\mathbf{q}), & \text{for tasks errors} \\ \nabla\mathbf{f}_k(\mathbf{q}), & \text{for task gradients} \end{cases} \tag{2}$$

$$\dot{\mathbf{q}}_D = \sum_{k=1}^{t} \lambda_k\mathbf{P}_{k-1}\nabla\boldsymbol{\eta}_k(\mathbf{q}) \tag{3}$$

In Eq. ((1)), $\mathbf{P}_k$ is a projection to the nullspace, $\mathbf{J}_k$ is the Jacobian matrix, $\mathbf{J}_k^\dagger$ refers to the pseudo-inverse of $\mathbf{J}_k$ computed as $\mathbf{J}_k^\dagger = \mathbf{J}_k^T(\mathbf{J}_k\mathbf{J}_k^T)^{-1}$, and the subscript k refers to the task where $k = 1$ is the highest priority task and $k \in [1, \ldots, t]$ where t is the total number of priority levels and tasks (i.e. one task per priority level). The joint velocities $\dot{\mathbf{q}}_D$ for both arms A and B, can be found as a weighted sum (with weights λ_k) of the differential kinematics as per Eq. (3), where the contributions for various tasks (Eq. (2)) are prioritized through the corresponding $\mathbf{P}_k$ projection. Tasks $\nabla\boldsymbol{\eta}_k(\mathbf{q})$ can either be expressed using errors $\boldsymbol{\epsilon}_k(\mathbf{q})$

or gradients $\nabla \mathbf{f}_k(\mathbf{q})$. In the latter, it is desirable that task functions, $\mathbf{f}(\mathbf{q})$, be minimized or maximized, correspondingly one would travel in the direction of the positive or negative task gradient.

III. DA-CTCR Task Formulation and Definition

This section outlines the proposed control tasks for the DA-CTCR. These particular tasks were selected for the useful capabilities they enable for manipulation of the environment, relative orienting of the two arms, as well as keeping the system in a well conditioned state (i.e. collision-free, able to execute commands effectively). All following tasks refer to the entire DA-CTCR system so $\mathbf{q}_D \in \mathbb{R}^{12}$ is shortened to $\mathbf{q}$. Furthermore, the tasks gradients $\nabla \boldsymbol{\eta}$ are computed using finite differences, a local method, with perturbations of size 10^{-6}.

A. Single-Arm and Relative End-Effector Positioning

Single and relative end-effector position errors $\boldsymbol{\epsilon}_{Ap}$ and $\boldsymbol{\epsilon}_{Dp} \in \mathbb{R}^3$ are expressed using a hybrid representation in which translation and rotation are decoupled. $\dot{\boldsymbol{X}}_A$ is the trajectory feed-forward term of robot A, i.e. difference between current and last end effector position. Eqs. (4) and (5) represent the dual-arm following and single arm path tracking tasks respectively.

$$\nabla \boldsymbol{\eta}_{Dp}(\mathbf{q}_D) = \boldsymbol{J}^{\dagger}_{Dp} \boldsymbol{\epsilon}_{Dp} \tag{4}$$

$$\nabla \boldsymbol{\eta}_{Ap}(\mathbf{q}_A) = \begin{Bmatrix} \begin{bmatrix} \boldsymbol{J}^{\dagger}_{Ap} \left[\lambda_{Ap} \boldsymbol{\epsilon}_{Ap} + \dot{\boldsymbol{X}}_A \right] \\ \mathbf{0}_{6\times 1} \end{bmatrix} \end{Bmatrix} \tag{5}$$

B. Dual-Arm Manipulability Maximization

A measure of manipulability [13] describes how well-conditioned (i.e. how far from a singularity) a manipulator is to achieve any arbitrary velocity. Functionally, increasing the manipulability measure of a manipulator during operation means improving the velocity performance of the DA-CTCR and keeping the robot well away from singularities. This task aims to maximize the manipulability measure $\mathbf{f}_M(\mathbf{q}) = \sqrt{\det(\mathbf{J}(\mathbf{q})\mathbf{J}(\mathbf{q})^T)}$ for the entire DA-CTCR system. Manipulability maximization can be achieved using the task: $\nabla \boldsymbol{\eta}_M(\mathbf{q}) = +\nabla \mathbf{f}_M(\mathbf{q})$.

C. Body Collision Avoidance

The DA-CTCR is susceptible to self-collisions between the two CTCRs. In order to avoid these states during operation, this task aims to maximize the distance between the two CTCR bodies. The discretization of robots A and B's backbone shapes allows us to define the backbone segments $\mathbf{S}_A$ and $\mathbf{S}_B$ for robot A and B, respectively. The distances between all elements in $\mathbf{S}_A$ and $\mathbf{S}_B$ are computed as $d_{u,v}$ according to [9] where u and v index specific backbone segments in $\mathbf{S}_A$ and $\mathbf{S}_B$, respectively. The minimum distance *among all* computed distances $d_{u,v}$ is defined as the task function for body collision avoidance, i.e. $\mathbf{f}_S(\mathbf{q}) = \min(d_{u,v}), \quad \forall u \in \mathbf{S}_A$ and $\forall v \in \mathbf{S}_B$. This is also known as the closest distance to collision which is desirable to maximize using the task: $\nabla \boldsymbol{\eta}_S(\mathbf{q}) = +\nabla \mathbf{f}_S(\mathbf{q})$.

D. Triangulation of End-Effectors

Lines-of-sight (LoS) are rays with tangent curvature at, and emanating from, the end-effector tip. The directions of such rays are extracted from both robots A and B's end-effector pose $\mathbf{T}_{A/B}$ and are denoted $\mathbf{z}_{A/B}$. Triangulation is defined as a desired relative orientation of the two end-effectors such that the LoS of the two arms meet at a certain point in space *or* that is a desired triangulation angle achieved between the LoS.

1) Triangulation to a Specified Angle: To triangulate both arms according to [2] means for the LoS of both end-effectors to orient to a desired *triangulation angle* γ, defined as $\gamma = \arccos \frac{\mathbf{z}_A \cdot \mathbf{z}_B}{||\mathbf{z}_A|| \times ||\mathbf{z}_B||}$ (not necessarily requiring the LoS to intersect at a point). The resulting task is to minimize the error between the current and desired triangulation angle and is defined as $\mathbf{f}_{Ta}(\mathbf{q}) = |\gamma_{des} - \gamma_{cur}|$. This task is achieved by including the task gradient term in the control: $\nabla \boldsymbol{\eta}_{Ta}(\mathbf{q}) = -\nabla \mathbf{f}_{Ta}(\mathbf{q})$.

2) Triangulation to a Specified Point: In some interaction cases, it is more meaningful to command both robots to "point" to a shared location. One example is a scenario where one arm is working in an area under teleoperative control and the other holds a camera that could be controlled to automatically point at the tip of the other to visualize. A *triangulation point* $\mathbf{L}$ in free space can be defined by the operator. Consider $\mathbf{r}_{A/B}$ to define the start of the ray (i.e. the end-effector position) and $\mathbf{z}_{A/B}$ the direction. The shortest distance d between each ray from both CTCR arms and the desired triangulation point is computed by projecting the line between $\mathbf{L}$ and $\mathbf{r}_{A/B}$ onto $\mathbf{z}_{A/B}$ direction vector/pointing direction to get the parameterized position $\mathbf{t}_{A/B} = \frac{(\mathbf{L} - \mathbf{r}_{A/B}) \cdot \mathbf{z}}{\mathbf{z} \cdot \mathbf{z}}$. The projected position starting from $\mathbf{t}_{A/B}$ is then computed to get the shortest distance between the LoS and the triangulation point as: $d_{A/B} = \begin{cases} ||\mathbf{L} - \mathbf{r}_{A/B}||, & \text{if } \mathbf{t}_{A/B} \leq 0 \\ ||\mathbf{L} - (\mathbf{r}_{A/B} + \mathbf{t}_{A/B} \times \mathbf{z}_{A/B})||, & \text{otherwise} \end{cases}$

The task function is the sum of the shortest distances between the LoS and the triangulation point $\mathbf{f}_{Tp}(\mathbf{q}) = \mathbf{d}_A + \mathbf{d}_B$. In minimizing these distances, using: $\nabla \boldsymbol{\eta}_{Tp}(\mathbf{q}) = -\nabla \mathbf{f}_{Tp}(\mathbf{q})$, this task is achieved.

IV. Control Design

To evaluate the tasks defined in Section III. for the single and dual-arm CTCR, different forms of a task-priority controller are constructed using the presented framework. These control modules are outlined below using the conventions established above from Eqs. (1), (2), and (3):

1) **Follower:** Follow a path in space with one end-effector and have the other follow automatically.
   a) Relative position control ($k = 1$), Trajectory following of one arm ($k = 2$). $\dot{\mathbf{q}}_D = \lambda_D \nabla \boldsymbol{\eta}_{Dp}(\mathbf{q}_D) + \lambda_A \mathbf{P}_1 \nabla \boldsymbol{\eta}_{Ap}(\mathbf{q}_A)$
   b) 1a) with manipulability maximization ($k = 3$), i.e. adding a $+\lambda_M \mathbf{P}_2 \nabla \boldsymbol{\eta}_M(\mathbf{q})$ term.
2) **Triangulator:** Achieve a certain triangulation between the two end-effector arms.

TABLE I

Parameters of both arms of the simulated and real DA-CTCR setup.

Robot	Tube	$\emptyset_{out}$ / $\emptyset_{in}$	L_s	L_c	κ
Sim. 1)	1	1.21/0.67	430.6	55.7	0.017
	2	1.92/1.28	241.1	70.2	0.01
	3	2.74/2.08	82.4	68.6	0.005
Sim. 2)	1	1.21/0.67	150	30.7	0.048
	2	1.92/1.28	100	19.5	0.017
	3	2.74/2.08	50	23.8	0.014
Real	1	1.21/0.67	486.3‖430.6	0‖55.7	0‖0.017
	2	1.92/1.28	241.1	70.2	0.01
	3	2.74/2.08	82.4	68.6	0.005

L_s and L_c are each tube's straight and curved length, respectively. κ denotes the curvature of each tube, while $\emptyset_{out}$ / $\emptyset_{in}$ are its outer and inner diameters. All values are given in mm except κ in m^{-1}. Differing tube designs between arm A and B are separated by ‖.

TABLE II

Initial configurations for all simulated and real robot experiments.

config	1	2	3	4	5	6	7
α_1	0	10	0	20	-50	10	-10
α_2	0	50	0	20	-50	10	-10
α_3	0	-60	0	20	-60	10	-10
β_1	-377	-90	-220	-300	-230	-250	-210
β_2	-225*	-60	-165	-225	-175	-200	-160
β_3	-67	-37	-95	-145	-105	-120	-100

Tube translations β_i are in millimeters and rotations α_i are in degrees with respect to the zero position of the system being all tubes fully extended and curvatures aligned down. Negative β_i indicate the length of tube retracted relative to the fully extended zero position. The the tube index is i where $i = 0$ represents the innermost tube. The '*' marks a variation of the configuration that was used. Note, the above configurations are applied only to a single tube set, not to all robots in Table I.

a) Triangulation of both end-effectors to a specified triangulation point ($k = 1$). $\dot{\mathbf{q}}_D = \lambda_{Tp} \nabla \boldsymbol{\eta}_{Tp}(\mathbf{q})$
b) Triangulation of both end-effectors to a desired triangulation angle ($k = 1$). $\dot{\mathbf{q}}_D = \lambda_{Ta} \nabla \boldsymbol{\eta}_{Ta}(\mathbf{q})$
c) 2a) with body collision avoidance ($k = 2$), i.e. adding a $+\lambda_S \mathbf{P}_1 \nabla \boldsymbol{\eta}_S(\mathbf{q})$ term.

V. Evaluation in Simulation

In this section we apply the proposed methods to two simulated DA-CTCR prototypes with a 50mm separation between the bases to demonstrate functionality of tasks in the hierarchy and generalizability of the proposed framework. All methods were implemented in C++ on a standard PC with an Intel Core i7-7500U CPU @ 2.7GHz and 16GB of RAM. The update rate of the controller is chosen as 10ms.

A. Evaluation Scenarios

The Follower (Control 1a) and b)) was tested on a DA-CTCR with the Sim. 1) tubeset in Tab. I and starting with both robots in config. 1 in Tab. II. A variant of this initial configuration had an added translation to β_2 of -5mm to original (noted '*') to induce a poorly conditioned configuration (i.e. ill-conditioned, IC) for which the tubes are close to being pulled into one another, compared to the original (i.e. well-conditioned, WC). The control task involved following a lissajous curve path with arm A ($A = 5$mm, $B = 5$mm, $a = 1$, $b = 2$) that was discretized to 1600 points.

The Triangulator (Control 2b) and 2c)) was tested on the Sim. 2) tubeset in Tab. I, which are the triangulation-optimal tube parameters from [2], with robot A starting from config. 2 in Tab. II and same for robot B with the signs of the α_is flipped. The control task involved increasing the relative triangulation angle from the initial angle of 25° to 115° in 2° increments.

B. Results

The results of the Follower scenario with the WC initial configuration is shown in Fig. 2. The same scenario was run again starting from the IC initial configuration. In this case, without manipulability maximization 1a) the system hit its translation limits within the first hundred iterations whereas with 1b) the controller was able to avoid the translation limits and allow proper following of the entire lissajous path.

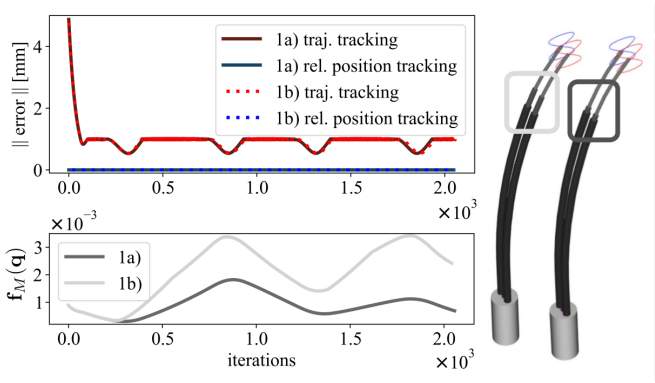

Fig. 2: Simulation results of the Follower following a lissajous curve path without 1a) and with 1b) manipulability maximization starting from a well-conditioned initial configuration (Left) and visualizations of the DA-CTCR in its end configuration for 1a) and 1b) (Right).

The results of the Triangulator scenario are shown in Fig. 3. Without the body collision task 2b), the robot is able to perform the task. This is due to the fact that the implemented simulator does not account for body interactions, thus in real life this would be an invalid motion. With it using 2c), as the desired triangulation angle is increased inducing a body collision, the robot is able to cross one arm over the other to avoid such a contact and still achieved the desired angle with little difference in error performance.

VI. Real DA-CTCR Robot Experiments

In this section we apply the proposed method to a DA-CTCR prototype with the real robot tube set in Tab. I to prove its applicability. The full setup is shown in Fig. 1.

A. Experimental Setup

1) Tracking System: An electromagnetic tracking system and sensors (Aurora v3, Nothern Digital Inc., ON, Canada) are used to measure the end-effectors' pose for closed-loop control. Doing so, we can measure position to sub-millimeter (RMS of 0.7mm) and orientation in sub-degree (RMS of 0.20°) ranges. Custom 6-DoF end-effector sensors were created from two 5-DoF sensors aligned in a T-shape that

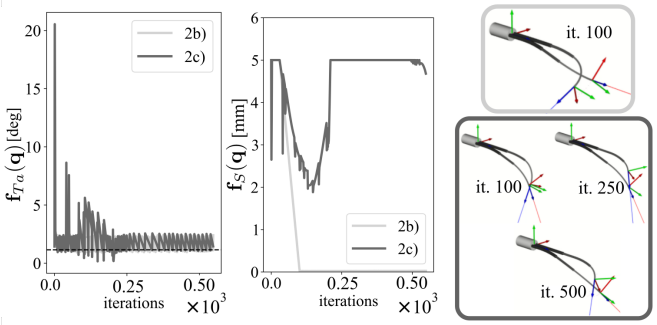

Fig. 3: Simulation results of the Triangulator increasing the relative triangulation angle without 2b) and with 2c) body collision avoidance (Left) and visualizations of the DA-CTCR during the manoeuver for 2b) and 2c) (Right).

can uniquely define the roll. They are fixated on the outside of the innermost end-effector tubes as shown in Fig. 1. As the tracking coils are lightweight, we neglect any impact on the robot's shape. For calibration with respect to each base frame, reference 6-DoF sensors were rigidly mounted to the base of robots A and B. This was also used to measure the rigid transformation between the robot bases. The update rate for Aurora measurements is 40 Hz. Due to asynchronous sensor updates and noise, 25 sequential measurements (each consisting of two end-effector pose measurements) each 25 milliseconds, were averaged and taken as the sensor feedback for a single control iteration. This limited real robot experiments to an update rate of 625ms.

2) Concentric Tube Continuum Robot: The DA-CTCR prototype used is shown in Fig. 1. Each single CTCR consists of three sliding carriages and six actuators (DCX 16 L, Maxon Motor AG, OW, Switzerland). A Motion Control Board (DMC-4163, Galil Motion Control, CA, USA) is used to control the motors of the actuation unit to the desired configuration values $\mathbf{q}$. Super-elastic pre-curved tubes made of Nickel-Titanium (Nitinol) (Young's modulus $E = 50{\times}10^9$ GPa, Poisson's ratio $\nu = 0.3$) are used. The geometric parameters of the tube sets used are listed in Tab. I.

B. Experiments

The selected test scenarios were chosen due to sensing limitations, i.e. collision avoidance requires complete shape measurements and measuring the manipulability directly is not feasible. The Follower controller 1a) was run on the real robot setup with the same lissajous path tested in simulation and a spatial circle path ($d = 5$mm) starting from config. 3 in Tab. II. To advance to the next point in the path, the controller needed to meet one of the two conditions: achieve a tracking error below 0.5mm or perform 10 control iterations. The Triangulator controller 2a) was tested on an example where the desired triangulation point followed a half-circle ($d = 5$mm) path with a sinusoidal oscillation (shown in Fig. 6) with the same advancement conditions as the Follower 1a) starting from config. 4 and 5 in Tab. II for robot A and B respectively. The Triangulator controller 2b) was tested on an example where the desired triangulation angle was varied in increments of 0.5° starting from config. 6 and 7 in Tab. II for robot A and B, respectively. To advance to the next point, the controller needed to meet one of the two conditions: achieve a triangulation error below 0.1° or perform 10 control iterations. All four scenarios were repeated for three trials.

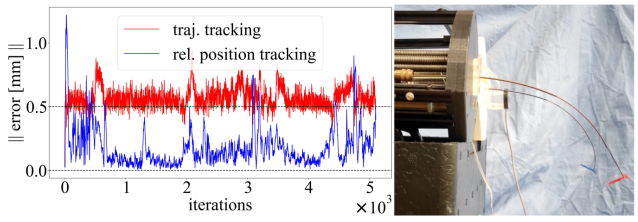

Fig. 4: Control results of the Follower 1a) following a lissajous curve on the real robot (Left). Real setup with an overlay of the simulation and lissajous path (Right).

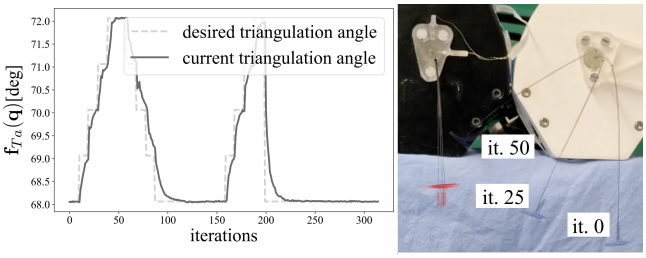

Fig. 5: Control results of the Triangulator 2b) on the real robot (Left) and the progression of the robot state in the first 50 control iterations (Right).

C. Results

One trial of the Follower 1a) on the lissajous curve path is shown in Fig. 4. Similarly, one trial using 2a), for achieving desired triangulation to a point in space, and 2b), for achieving desired triangulation angles, are shown in Figs. 5 and 6 respectively. All results suggest that the functional task was achieved and that the primary tasks were indeed prioritized over the secondary tasks as per the superior error performance of the primary task. Results from all trials are summarised in Tab. III.

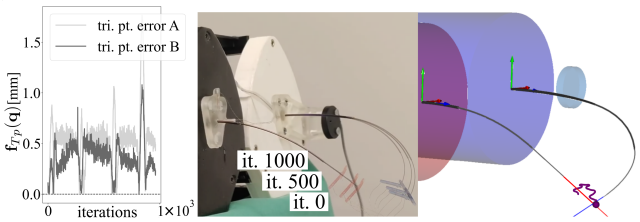

Fig. 6: Control results of the Triangulator 2a) on the real robot (Left), the progression of the robot state over the entire path (Center), the simulated robot with the designed triangulation point path and visualized robot lines-of-sight (Right).

TABLE III
Results of all control scenarios averaged across all trials, RMS ± StD shown

1a) lissajous [mm]	1a) circle [mm]	2a) [mm]	2b) [°]
AB: 0.38±0.27 A: 0.69±0.56	AB: 0.23±0.13 A: 0.54±0.05	A: 0.59±0.25 B: 0.40±0.16	0.005 ±0.008

AB: relative end-effector positioning error, A: trajectory tracking error with end-effector A.

VII. Discussion

The simulation results on the Follower when including manipulability show that the manipulability measure is improved throughout teleoperation (i.e. single-arm path following). Despite the task being maximization of the relative manipulability (which is difficult to visualize), this can also translate to the individual arms' manipulability as seen in Fig. 2 where in the final configuration, the tubes are more spaced apart with manipulability maximization compared to without. Practically, tending towards configurations with higher manipulability is favorable to avoid reaching the tube translation limits and maintaining acceptable dexterity of the arms during operation. This is further supported by the failure of the same path following scenario when the manipulator starts in an IC initial state without such a term in the control. Expected functional performance of the control tasks was observed over the triangulation simulation such that the body collision avoidance task prevented the system from a catastrophic collision and good error performance was observed in the triangulation angle tracking task. Over real robot trials, the Follower on the real robot performed quite well with sub-millimeter accuracy and the prioritization of the primary task of maintaining a fixed relative distance was evident in the significantly lower error value (by 0.31mm), for the primary task over the secondary one.

Overall, the experiments demonstrate that using finite differences to compute the task gradients is effective for a variety of tasks (i.e. manipulability maximization, body collision avoidance, and triangulation angle/point) within this framework. This suggests that the controller is versatile and can include a variety of secondary tasks. This includes optimizing other quantitative performance indices during operation, which could further improve upon the control such as stability and dexterity measures, kinematic isotropy/conditioning index, task compatibility, etc. A limitation to this approach is in the iterative nullspace projection technique which can only be exploited for a limited number of projections depending on the conditioning of the manipulator and the nature of the control task in higher priority levels. Furthermore, despite the promising performance of finite differences demonstrated in this work, there is still the limitation of it being a local approximation method which is only accurate within a small neighborhood for which it was computed. This could make it particularly sensitive to noise.

VIII. Conclusions

In this work, multiple control tasks for DA-CTCR were formulated and tested within a set of proposed control modules. The performance of the proposed modules on representative control scenarios demonstrate both the validity of the formulated tasks, as well as fundamental principles for multi-task controller design (task selection, placement in the hierarchy, and selected redundancy resolution scheme). The main contributions of this work are the integration of various common techniques into a unified DA-CTCR control framework and experimental validation on a robotic system.

The presented hierarchy modules enable the operator to maintain continuous control of the system while the robot provides certain assistance in the form of lower level/sub tasks performed in the background (ex. maximizing manipulability, avoiding body collisions, etc.). The framework presented in this work allows users to form different variants of the control resulting in control versatility. During robot operation, the operator can switch between modules as per the desired task at hand. This falls in the category of level 1 autonomy for CTCRs which remains an active area of research [8] within the community.

References

[1] A. W. Mahoney, H. B. Gilbert, and R. J. Webster III, "A review of concentric tube robots: modeling, control, design, planning, and sensing," The Encyclopedia of Medical Robotics: Volume 1 Minimally Invasive Surgical Robotics, pp. 181–202, 2019.

[2] M. T. Chikhaoui, J. Granna, J. Starke, and J. Burgner-Kahrs, "Toward motion coordination control and design optimization for dual-arm concentric tube continuum robots," IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 1793–1800, 2018.

[3] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," vol. 3. 1990 Kluwer Academic Publisher, 1990, pp. 201–222.

[4] B. Siciliano, O. Khatib, and K. Torsten, Springer handbook of robotics. Springer., 2016.

[5] B. Siciliano and J. . E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in Fifth International Conference on Advanced Robotics, Robots in Unstructured Environments, 1991, pp. 1211–1216 vol.2.

[6] R. J. Hendrick, C. R. Mitchell, S. D. Herrell, and I. Robert J. Webster, "Hand-held transendoscopic robotic manipulators: A transurethral laser prostate surgery case study," The International Journal of Robotics Research, vol. 34, no. 13, pp. 1559–1572, 2015, pMID: 27570361.

[7] J. Burgner, P. J. Swaney, D. C. Rucker, H. B. Gilbert, S. T. Nill, P. T. Russell, K. D. Weaver, and R. J. Webster, "A bimanual teleoperated system for endonasal skull base surgery," in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011, pp. 2517–2523.

[8] M. T. Chikhaoui and J. Burgner-Kahrs, "Control of continuum robots for medical applications: State of the art," pp. 1–11, 2018.

[9] S. Sabetian, T. Looi, E. D. Diller, and J. Drake, "Self-collision detection and avoidance for dual-arm concentric tube robots," IEEE Robotics and Automation Letters, pp. 1–1, 2019.

[10] K. Shamaei, Y. Che, A. Murali, S. Sen, S. Patil, K. Goldberg, and A. M. Okamura, "A paced shared-control teleoperated architecture for supervised automation of multilateral surgical tasks," in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2015, pp. 1434–1439.

[11] D. C. Rucker and R. J. Webster, "Computing jacobians and compliance matrices for externally loaded continuum robots," in IEEE International Conference on Robotics and Automation, 2011, pp. 945–950.

[12] R. S. Jamisola, P. S. Kormushev, R. G. Roberts, and D. G. Caldwell, "Task-space modular dynamics for dual-arms expressed through a relative jacobian," Journal of Intelligent & Robotic Systems, vol. 83, no. 2, pp. 205–218, 2016.

[13] T. Yoshikawa, "Manipulability of robotic mechanisms," The International Journal of Robotics Research, vol. 4, no. 2, pp. 3–9, 1985.