

# Dual-arm robotic manipulation of flexible cables

Jihong Zhu, Benjamin Navarro, Philippe Fraisse, André Crosnier and Andrea Cherubini

**Abstract**—Deforming a cable to a desired (reachable) shape is a trivial task for a human to do without even knowing the internal dynamics of the cable. This paper proposes a framework for cable shapes manipulation with multiple robot manipulators. The shape is parameterized by a Fourier series. A local deformation model of the cable is estimated on-line with the shape parameters. Using the deformation model, a velocity control law is applied on the robot to deform the cable into the desired shape. Experiments on a dual-arm manipulator are conducted to validate the framework.

## I. INTRODUCTION

The concept of deformation is defined as the transformation of a body from a reference configuration to a current configuration [1]. A configuration is a set containing the positions of all particles of the body. Deformations result from a stress field induced by forces. Much mainstream research on robot manipulation is placed on rigid objects. The deformation of these objects during manipulation is negligible or null.

A great deal of industrial/household/medical scenarios involve the manipulation of deformable flexible objects. To name a few: cable management in factories or at home [2], folding clothes [3], manipulation of organs and tissues in medical contexts [4].

Common flexible objects in our daily life are cables. Flexible cables (e.g., power cables, HDMI cables etc) are frequently used in industrial/household environments. Cables arrangement, pick and place and insertion are common tasks. These cables deform under forces acting on them. Humans can manipulate cables using two hands. Given a (reachable) desired cable shape, a human is able to deform the cable into such shape. This task is easy for a human to do without even knowing the internal dynamics of the cable (see Fig. 1). For robots, it still remains a challenge.

Probably the oldest literature on cable manipulation is the work of Hopcroft et al. [5]. In this work, a vision system was used to guide the robot and an abstract description of the cable configuration was developed. Later, Chen and Zhang [6] used a non-linear model to characterize the deformation of a flexible beam, and applied the model for manipulation. A more complex model of linear objects considering twist was developed in [7]; the stable shape of the object can be derived by minimizing the potential energy under geometric constraints. Nakagaki et al. extended the model in [7], and used it for cable insertion [8]. Kosuge et al. [9] modeled the static deformation of a sheet metal by Lagrange's equation based on a finite element model. Most of above

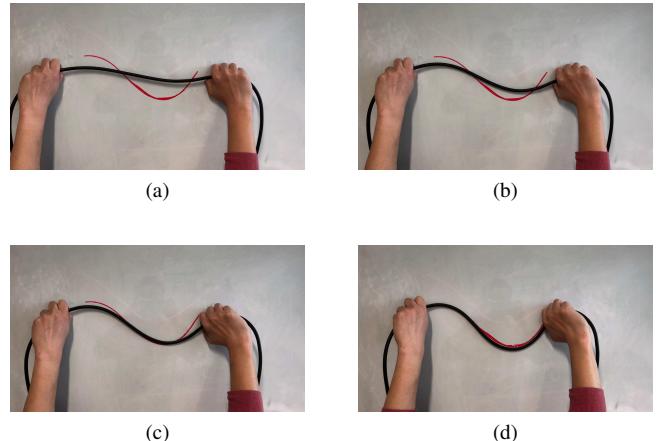


Fig. 1: Cable manipulation by humans, red color line marks the desired shape.

mentioned works require identifying the deformation model of the flexible cables/beams before manipulation. However, cables are usually of different thickness, and manufactured with distinct materials. For different cables, the deformation models generally vary. Therefore, these methods lack the robustness needed for them to be applied to general cables.

More recent works [10], [11], [12] adopted model-free methods to tackle the problem of manipulation of flexible objects. The control action was derived by an on-line deformation models constantly updated by visual feedbacks. In all three papers, deformations can only be characterized by point, angle and curvature displacement. A more general geometric representation of deformation of closed contour was proposed in [13] by truncated Fourier series.

In this work, we draw inspiration from [13], and extend the work to multiple robotic arms manipulation of an open contour (the shape of a flexible cable). In addition, we consider the rotational action of the manipulators while [13] considered only translations. The problem is more complex as we are dealing with more inputs. Furthermore by considering multiple manipulators and rotation, we are approaching manipulation capability of a human.

The objective of this work is to setup a framework for multi-arm robots to complete the task of deforming a flexible cable to a desired shape through vision-based control with a on-line estimated deformation model. Here, we consider only manipulation on a 2D plane, and leave 3D manipulation as future work.

The rest of the paper is organized as follows. In Sect. II, we formulate the task as a control problem. Sect. III describes

The authours are with the LIRMM, Université de Montpellier, CNRS, Montpellier, France. [firstname.lastname@lirmm.fr](mailto:firstname.lastname@lirmm.fr)

the methods to tackle the problem. In Sect. IV, a dual-arm robot demonstration is presented<sup>1</sup>. In Sect. V, we conclude.

## II. PROBLEM FORMULATION

Let us consider a multi-arm robot manipulating a cable on a 2D plane. The cable can be regarded as a system with unknown dynamics that accepts inputs from the robot. Each of the end-effectors applies 3 velocity inputs, respectively: two translation velocities in the manipulation plane,  $\dot{x}$  and  $\dot{y}$ , and one angular velocity  $\omega$  along the axis perpendicular to the manipulation plane. A dual-arm manipulation example is shown in Fig. 2. Assume there are  $M$  number of manipulators, the total number of inputs from the robot is:

$$\mathbf{r} = [\dot{x}_1 \ \dot{y}_1 \ \omega_1 \ \dots \ \dot{x}_M \ \dot{y}_M \ \omega_M]^T \in \mathbb{R}^{3M}. \quad (1)$$

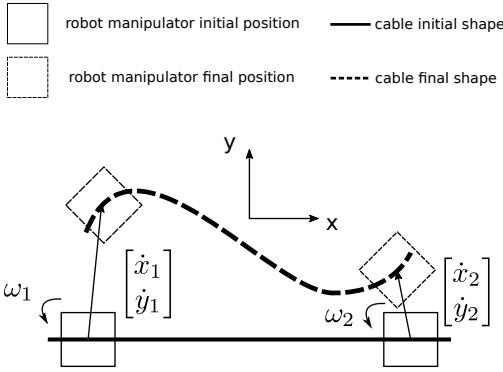


Fig. 2: Control inputs of a dual-arm example.

The shape of the cable is continuously observed by a static camera perpendicular to the manipulation plane. The cable shape on the camera image is represented as  $\mathbf{c} = [\mathbf{u}, \mathbf{v}]^T$ , where  $\mathbf{u}$  and  $\mathbf{v}$  are image coordinates of pixels sampled along the cable. We represent the desired cable shape by  $\mathbf{c}^*$ .

The problem is to use the control inputs  $\mathbf{r}$  to drive the cable from its initial shape  $\mathbf{c}_0$  to the desired shape  $\mathbf{c}^*$  with an on-line estimated deformation model by visual feedback.

## III. METHODS

To tackle the problem stated above, we first transform the cable shape into feature parameters other than  $\mathbf{u}$  and  $\mathbf{v}$ . After parameterizing the shape, we model locally the relationship between the robot motion and the changes in the shape parameters. Such model is referred to as a deformation model. The final step is to derive the control strategy to deform the cable into the desired shape based on such deformation model. In this section, we describe each sub-task sequentially.

<sup>1</sup>The video of the experiments can be found at [https://www.youtube.com/watch?v=DPl\\_d7lbL84](https://www.youtube.com/watch?v=DPl_d7lbL84)

### A. Shape feature parameters

The  $i^{th}$  sample  $c(i) = [u(i), v(i)]^T$ ,  $i = 1, 2, \dots, K$  can be approximated using Fourier series:

$$\begin{aligned} u(i) &= \sum_{j=1}^N [a_j \ b_j] \begin{bmatrix} \cos(j\rho_i) \\ \sin(j\rho_i) \end{bmatrix} + e \\ v(i) &= \sum_{j=1}^N [c_j \ d_j] \begin{bmatrix} \cos(j\rho_i) \\ \sin(j\rho_i) \end{bmatrix} + f, \end{aligned} \quad (2)$$

with

$$\rho_i = (i-1) \frac{\pi}{K},$$

$N \geq 1$  is the order of the Fourier series.

We denote  $\mathbf{s}$  to be the feature parameters characterize (2):

$$\mathbf{s} = [a_1 \ b_1 \ \dots \ a_N \ b_N \ e \ c_1 \ d_1 \ \dots \ c_N \ d_N \ f]^T \in \mathbb{R}^{4N+2}. \quad (3)$$

It will later be used in deformation model estimation and control. We will show how we solve for  $\mathbf{s}$  given image data.

We can rewrite (2) as:

$$c(i) = \begin{bmatrix} u(i) \\ v(i) \end{bmatrix} = \begin{bmatrix} \mathbf{F}(i) & \mathbf{0} \\ \mathbf{0} & \mathbf{F}(i) \end{bmatrix} \mathbf{s}. \quad (4)$$

In (4),  $\mathbf{F}(i)$  are the harmonics terms defined as:

$$\mathbf{F}(i) = [\cos \rho_i \ \sin \rho_i \ \dots \ \cos(N\rho_i) \ \sin(N\rho_i) \ 1] \in \mathbb{R}^{2N+1}.$$

Using all  $K$  samples in  $\mathbf{c}$ , we have:

$$\mathbf{c} = \mathbf{G}\mathbf{s},$$

with:

$$\mathbf{c} = [c(1)^T \ c(2)^T \ \dots \ c(K)^T]^T \in \mathbb{R}^{2K},$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{F}(1) & \mathbf{0} \\ \mathbf{0} & \mathbf{F}(1) \\ \vdots & \vdots \\ \mathbf{F}(K) & \mathbf{0} \\ \mathbf{0} & \mathbf{F}(K) \end{bmatrix} \in \mathbb{R}^{2K \times (4N+2)}.$$

Feature parameters of the shape can be solved by linear least squares:

$$\mathbf{s} = \mathbf{G}^\dagger \mathbf{c},$$

where  $\dagger$  represents the pseudo inverse:  $\mathbf{G}^\dagger = (\mathbf{G}^T \mathbf{G})^{-1} \mathbf{G}^T$ .

To ensure the pseudo inverse of  $\mathbf{G}$  exists we have:

$$\text{rank}(\mathbf{G}) \geq 4N + 2, \quad (5)$$

must be fulfilled. A necessary condition for (5) is:

$$2K \geq 4N + 2.$$

### B. Deformation model estimation

The feature parameters  $\mathbf{s}$  describes the cable shape. A small movement of the robot will produce a tiny change in the cable shape, hence on the feature parameters. From this observation, at a given operating point, we are able to linearize the deformation model as:

$$\delta\mathbf{s} = \mathbf{Q}\delta\mathbf{r}. \quad (6)$$

In (6)

$$\delta\mathbf{r} = \mathbf{r}\delta t \in \mathbb{R}^{3M}, \quad (7)$$

corresponds to the change in robot position with  $\delta t$  being the time interval and  $\mathbf{r}$  being the velocity inputs of the robot;  $\delta\mathbf{s} \in \mathbb{R}^{4N+2}$  is the change in feature parameters, and  $\mathbf{Q} \in \mathbb{R}^{(4N+2) \times 3M}$  is the local deformation matrix relating the two.

For the  $i^{\text{th}}$  element of  $\mathbf{s}$ , we can write:

$$\delta s_i = \delta\mathbf{r}^T \mathbf{q}_i,$$

where  $\mathbf{q}_i^T \in \mathbb{R}^{3M}$  is the  $i^{\text{th}}$  row of  $\mathbf{Q}$ .

To estimate  $\mathbf{q}_i$ , we denote the current time as  $t_m$ . Using a constant sampling period  $\delta t$ , within the time period  $(m-1)\delta t$ , we collect  $m$  consecutive data of  $\delta s_i$  and  $\delta\mathbf{r}^T$  while the robot is moving:

$$\boldsymbol{\sigma}_i = \begin{bmatrix} \delta s_i(t_1) \\ \delta s_i(t_2) \\ \vdots \\ \delta s_i(t_m) \end{bmatrix} \in \mathbb{R}^m, \mathbf{R} = \begin{bmatrix} \delta\mathbf{r}^T(t_1) \\ \delta\mathbf{r}^T(t_2) \\ \vdots \\ \delta\mathbf{r}^T(t_m) \end{bmatrix} \in \mathbb{R}^{m \times 3M},$$

where

$$t_k = t_m - (m-k)\delta t, \quad k = 1, 2, 3, \dots, m.$$

Using  $\mathbf{R}$  and  $\boldsymbol{\sigma}_i$  we have:

$$\boldsymbol{\sigma}_i = \mathbf{R}\mathbf{q}_i. \quad (8)$$

Then,  $\mathbf{q}_i$  can be estimated via:

$$\hat{\mathbf{q}}_i = (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \boldsymbol{\sigma}_i. \quad (9)$$

To successfully estimate  $\mathbf{Q}$ , the following conditions must be satisfied::

- 1)  $m \geq 3M$
- 2)  $\mathbf{R}$  must has full row rank.

The first condition ensures that in (8), there are equal or more equations than unknowns, so that  $\mathbf{q}_i$  can be solved using linear least squares. If the second condition is not satisfied,  $\mathbf{R}^T \mathbf{R}$  in (9) will be singular. In practice, this condition infers that within the time period  $(m-1)\delta t$ , it is necessary that the robot moves in all directions and rotate at least once. This makes prefect sense as if one or more component(s) of  $\mathbf{r}$  is/are not active during the whole time period, no relationship can be derived between that/those component(s) and the feature parameters.

To ensure that  $\mathbf{R}$  has full row rank, we can check the rank of  $\mathbf{R}$  before estimation of  $\mathbf{Q}$ . If  $\mathbf{R}$  does not fulfill the rank condition, we extend the time period until the rank condition is fulfilled. Last but not the least, the time period  $(m-1)\delta t$  should be as small as possible while satisfying both conditions.

### C. Shape servo-controller

Using the camera, we are able to observe the current parametric shape of the cable  $\mathbf{c}$ . Given the target shape  $\mathbf{c}^*$ , both  $\mathbf{c}$  and  $\mathbf{c}^*$  can be parameterized by feature parameters  $\mathbf{s}$  and  $\mathbf{s}^*$  given in (3). The differences between the current

shape  $\mathbf{c}$  and the desired shape  $\mathbf{c}^*$  can be characterized by the difference between the feature parameters of the two shapes:

$$\Delta\mathbf{s} = \mathbf{s}^* - \mathbf{s}.$$

Using the estimated deformation model:

$$\delta\mathbf{s} = \hat{\mathbf{Q}}\delta\mathbf{r},$$

we have then:

$$\delta\mathbf{r} = \lambda \hat{\mathbf{Q}}^\dagger \Delta\mathbf{s}. \quad (10)$$

As described in (7),  $\delta\mathbf{r}$  is a vector that consists of motion for end-effectors:

$$\delta\mathbf{r} = [\delta\mathbf{r}_1^T, \delta\mathbf{r}_2^T, \dots, \delta\mathbf{r}_M^T] \in \mathbb{R}^{3M},$$

where

$$\delta\mathbf{r}_i = \begin{bmatrix} \delta x_i \\ \delta y_i \\ \delta \omega_i \end{bmatrix} \in \mathbb{R}^3, \quad i = 1, 2, \dots, M, \quad (11)$$

with  $\delta \omega_i$  the end effector orientation (with regard to the vertical axis) controlled by  $\omega_i$ .

The translation vectors in (11)

$$\begin{bmatrix} \delta x_i \\ \delta y_i \end{bmatrix} \in \mathbb{R}^2, \quad i = 1, 2, \dots, M,$$

can be normalized to give the direction of translation of each end-effectors:

$$\mathbf{d}_i = \frac{1}{\sqrt{\delta x_i^2 + \delta y_i^2}} \begin{bmatrix} \delta x_i \\ \delta y_i \end{bmatrix} \in \mathbb{R}^2, \quad (12)$$

Based on (12), the velocity control command to be sent to each end-effectors can be formulated as:

$$\mathbf{r} = [\mathbf{r}_1 \ \mathbf{r}_2 \ \dots \ \mathbf{r}_M] \in \mathbb{R}^{3M},$$

where

$$\mathbf{r}_i = \begin{bmatrix} \lambda_1 \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \lambda_2 \end{bmatrix} \begin{bmatrix} \mathbf{d}_i \\ \omega_i \end{bmatrix} \in \mathbb{R}^3. \quad (13)$$

In (13),  $\lambda_1, \lambda_2$  are adjustable gains. These should be tuned to small values leading to a slow motion of the robot such that the assumptions of local deformation model holds.

### D. Iterative model adaptation

Since (6) calculates a local deformation model, throughout the manipulation,  $\mathbf{Q}$  needs to be updated.

At time  $t_N$ , we estimate the local deformation matrix  $\mathbf{Q}_N$ . Using the deformation matrix, we can derive the velocity control command  $\mathbf{r}_N$ . The control is applied for a  $\delta t$  period until  $t_{N+1}$ .

At time  $t_{N+1}$ , the cable is driven to a new shape with new feature parameters  $\mathbf{s}(t_{N+1})$ . Therefore, we have new set of data:

$$\begin{aligned} \delta\mathbf{s}(t_{N+1}) &= \mathbf{s}(t_{N+1}) - \mathbf{s}(t_N), \\ \delta\mathbf{r}(t_{N+1}) &= \mathbf{r}_{N+1} - \mathbf{r}_N \delta t. \end{aligned} \quad (14)$$

The data window is shifted from  $\{t_1 - t_N\}$  to  $\{t_2 - t_{N+1}\}$ , as we add in the newly obtained data (14), and remove the first data  $\delta\mathbf{s}(t_1)$  and  $\delta\mathbf{r}(t_1)$ . Using the new data window we can follow the same step and update  $\mathbf{Q}_N$  to be  $\mathbf{Q}_{N+1}$ .

## IV. EXPERIMENTAL VALIDATION

### A. Experimental setup

We use two lightweight KUKA LWR IV robots for the dual-arm setup. Hence,  $M = 2$  for the above algorithms.

We design two grippers to ensure manipulation of the cable without slipping (see Fig. 3). The blue gear is attached to the robot, the black cylinder represents the cable and the yellow gear is used to fastens the cable rigidly.

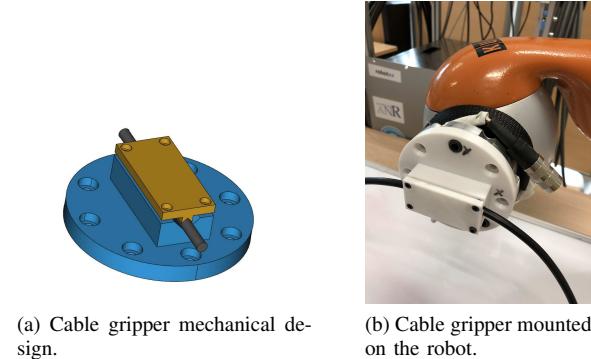


Fig. 3: The cable gripper.

The cable is gripped by two end-effectors. An Allied Vision® AVT camera is placed perpendicular to the manipulation plane to track the shape of the cable during manipulation. The image resolution is  $1936 \times 1456$ . A Linux-based 64-bits PC is used to process the image at a frame rate of 33.3Hz.

The overall hardware setup is showed in Fig. 4.

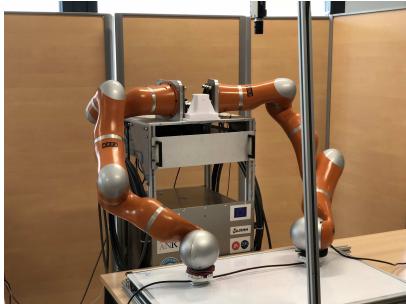


Fig. 4: The hardware setup for cable manipulation.

To test the robustness of the algorithm, we prepared two cables with different thickness. The diameter of the thin cable is 5.88 millimeters, and the thick cable is 7.66 millimeters.

### B. Image processing

Image processing is not the major part of this research, therefore we simplify the environment by putting a white board on the table to serve as the 2D plane and using black cables for the manipulation. The detection of the cable is achieved with OpenCV 3.2.0. Since camera is in a fixed position, Region of Interest (ROI) on the image for the manipulation is constant. The color image is converted into a grayscale image. Then by thresholding the grayscale image,

we can segment the cable from the image. The target shape is drawn on the white board with red color markers. To detect the target shape, we transform the image representation from RGB to HSV, and then apply thresholding to segment the red shape from the image. Fig. 5 shows the image processing under experimental conditions.

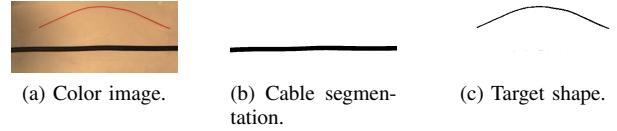


Fig. 5: Image processing.

### C. Experiments

The target shape feature parameters  $\mathbf{s}^*$  is calculated from the binary image of the target shape (Fig. 5c) at the start of the control scheme. The current shape of the cable is continuously observed. And the feature parameters of the cable is constantly updated to compute the local deformation model and the control inputs. We choose  $N = 2$  for Fourier series in (2). This value is chosen because: (i) cable shapes reachable by the dual-arm are not complex,  $N = 2$  can produce reasonable approximation of the shape while the  $N = 1$  sometimes result in a bad approximation, and (ii) larger  $N$  will result in more feature parameters of the shape. Since the inputs from the dual-arm robot is always 6, when  $N = 1$ , number of feature parameters is  $4N + 2 = 6$ , making the system fully actuated. When  $N > 1$ , with bigger  $N$ , the system becomes more and more under actuated, which increases the possibility of control being attracted to a local minimum [14], also (iii) increment  $N$  will increase the complexity of computation.

To initialize  $\mathbf{Q}$  estimation, the robot executes a small motion:

$$\delta \mathbf{r} = [-0.002, -0.002, 0.5, 0.002, -0.002, -0.5]. \quad (15)$$

The translation is expressed in meters and rotation in degrees in (15). This motion excites all components in  $\mathbf{r}$  and based on the algorithm proposed in Section III-B, an initial estimation of  $\mathbf{Q}$  can be computed.

The following evaluation metric is designed to monitor the performance of the controller:

$$\Phi = \frac{\|\mathbf{s}^* - \mathbf{s}\|}{\|\mathbf{s}^*\|}.$$

Lower  $\Phi$  indicates better matches between the cable shape and the target shape.

We test the algorithm on the thick cable with two different shapes. Figure 6 and 7 depicts the experiments. The red color line is the target shape. In each figure, the initial and final shape of the cable is presented in (a) and (d) respectively. Figure (b) and (c) shows the intermediate shapes while reaching the final shape.

Figure 8 and 9 show the evolution of the performance metric during the manipulation. The general trend of the

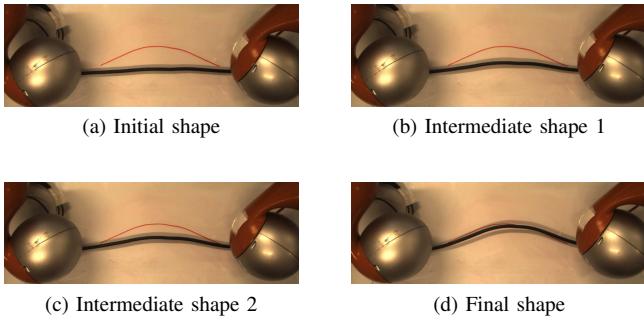


Fig. 6: Experiment 1 - Thick cable.

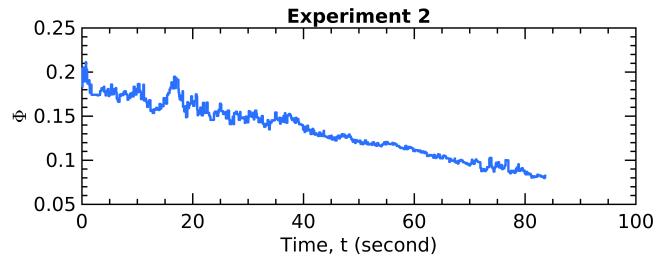


Fig. 9: Performance metric - Experiment 2.

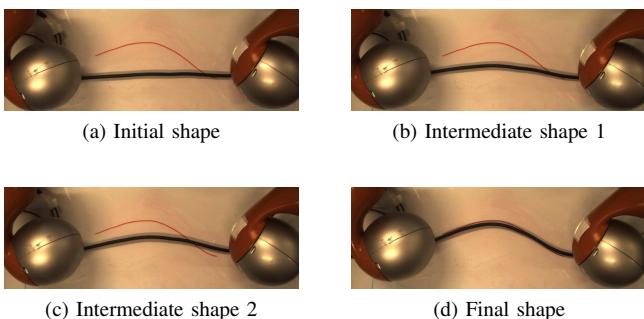


Fig. 7: Experiment 2 - Thick cable.

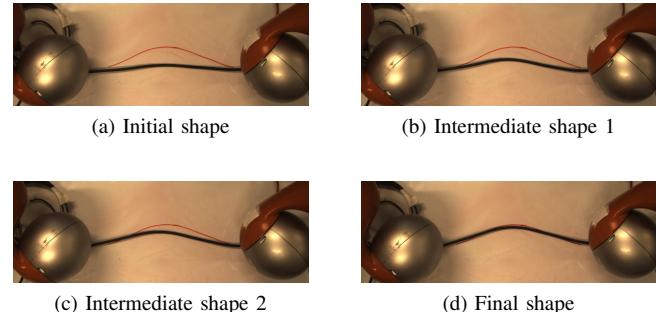


Fig. 10: Experiment 3 - Thin cable.

performance metric is decreasing and the final shape overlap with the target shape as shown in figure 6 and 7.

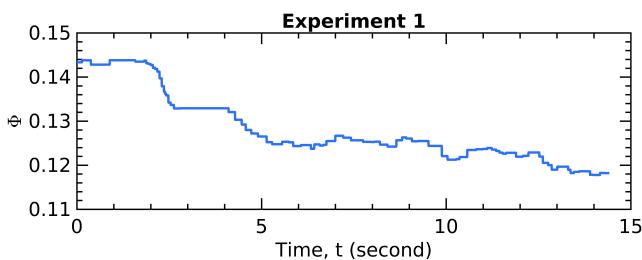


Fig. 8: Performance metric - Experiment 1.

To further test the robustness of our algorithm, we test it on a thinner cable. The shape servoing steps are shown in Fig. 10 and the evolution of the performance metric is shown in Fig. 11.

We try also to change the initial configuration of the cable. We slightly alter the light condition, the segmentation is robust to adapt. The manipulation result is shown in Fig. 12. It proves that the algorithm is invariant of the initial configuration of the cable.

The shape servoing tasks are carried out with no knowledge of the deformation characteristic of the cable. The algorithm is verified on two distinct cables to show robustness. In addition, no camera calibration is needed.

Figure 13 shows one failure in reaching the target shape. Figure 13a and 13b shows the initial shape and final shape of the cable respectively, and the red line depicts the target

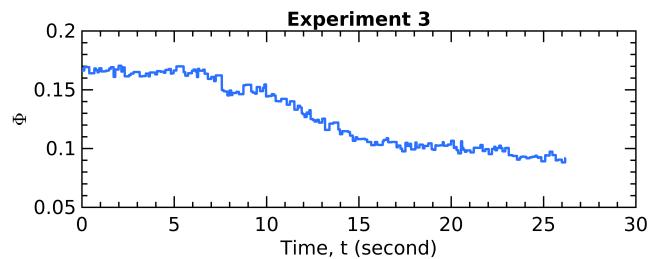


Fig. 11: Performance metric - Experiment 3.

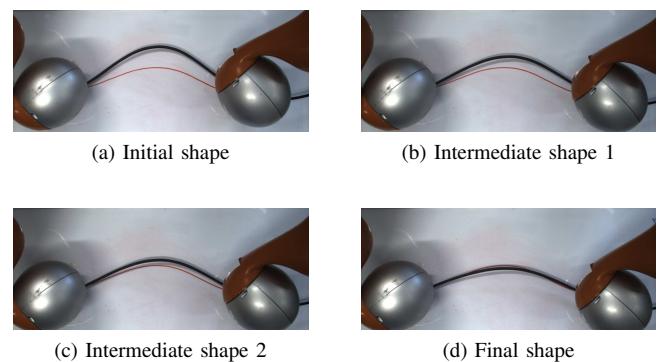


Fig. 12: Experiment 4 - Thin cable, different starting configuration.

shape. The robot goes back and force in a final shape and is not able to reach the target shape.

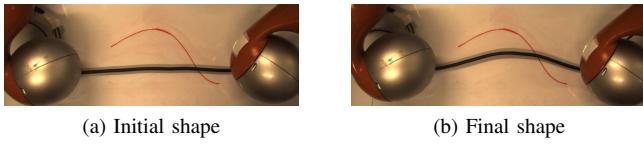


Fig. 13: Experiment 5 - Unreachable target.

In the previous experiments, the control is able to make the parameter differences  $\Delta s$  smaller, hence the cable approaches the target shape. For the failure case, in Fig. 14 we present the changes in the absolute value of parameter differences  $|\Delta s|$  for each feature parameter after one step of control around the final shape. We can observe that after the control, for 6 parameters, the differences decreases, for the other 4, the differences increases. This may due to under-actuation. With 6 inputs from the robot and 10 parameters to control, the system is under actuated. Hence, we can not ensure global asymptotic stability of the control [14].

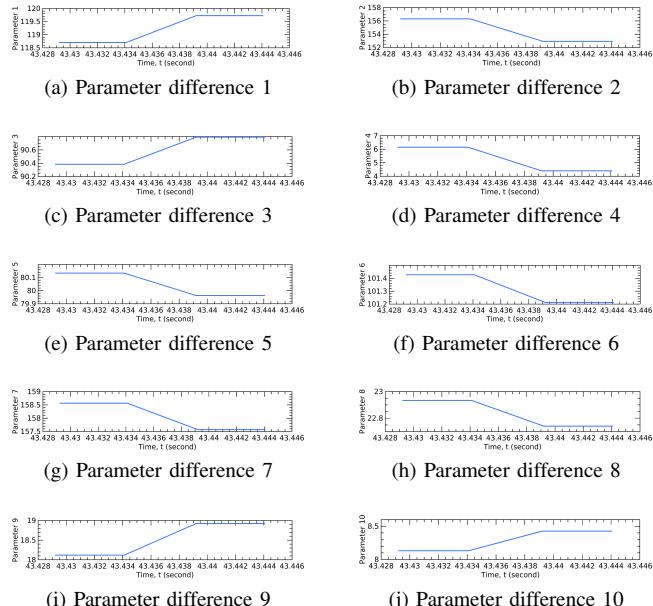


Fig. 14: Change of parameter differences after one step of control around the final shape

In addition, the deformation model is locally approximated, so if the target shape deviates too much from the initial shape, without path planning, the control will more likely converge to a local minimum. The approximation accuracy can also affect the convergence.

## V. CONCLUSIONS AND FUTURE WORK

In this paper, we adopt the model-free method proposed in [13], and extend the method to multiple robotic arms manipulation of open contours. The contour is characterized by a Fourier-based feature parameters. The parameters are

used to estimate a local deformation model on-line. Then the deformation model is used to compute control to drive the cable to the target shape. As the deformation model directly maps the robot motion to feature parameters, this method requires no camera calibration.

The under-actuation problem (highlighted in the last experiment) can be addressed by reducing the number of shape feature parameters. In future research, we intend to work on different representation of the shapes. We would also like to investigate path planning strategies to further enhance the convergence of the algorithm.

## ACKNOWLEDGEMENT

This work has received funding from the European Union Horizon 2020 research and innovation programme as part of the project VERSATILE under grant agreement No 731330.

## REFERENCES

- [1] C. Truesdell and W. Noll, in *The non-linear field theories of mechanics*. Springer, 2004.
- [2] J. R. White, P. E. Satterlee Jr, K. L. Walker, and H. W. Harvey, “Remotely controlled and/or powered mobile robot with cable management arrangement,” U.S. Patent 4 736 826, 1988.
- [3] S. Miller, J. Van Den Berg, M. Fritz, T. Darrell, K. Goldberg, and P. Abbeel, “A geometric approach to robotic laundry folding,” *Int. Journal of Robotics Research*, vol. 31, no. 2, pp. 249–267, 2012.
- [4] V. Mallapragada, N. Sarkar, and T. K. Podder, “Toward a robot-assisted breast intervention system,” *IEEE/ASME Trans. on Mechatronics*, vol. 16, no. 6, pp. 1011–1020, 2011.
- [5] J. E. Hopcroft, J. K. Kearney, and D. B. Krafft, “A case study of flexible object manipulation,” *Int. Journal of Robotics Research*, vol. 10, no. 1, pp. 41–50, 1991.
- [6] C. Chen and Y. F. Zheng, “Deformation identification and estimation of one-dimensional objects by vision sensors,” *Journal of Field Robotics*, vol. 9, no. 5, pp. 595–612, 1992.
- [7] H. Wakamatsu, S. Hirai, and K. Iwata, “Modeling of linear objects considering bend, twist, and extensional deformations,” in *IEEE Int. Conf. on Robotics and Automation*, vol. 1, 1995, pp. 433–438.
- [8] H. Nakagaki, K. Kitagi, T. Ogasawara, and H. Tsukune, “Study of insertion task of a flexible wire into a hole by using visual tracking observed by stereo vision,” in *IEEE Int. Conf. on Robotics and Automation*, vol. 4, 1996, pp. 3209–3214.
- [9] K. Kosuge, H. Yoshida, T. Fukuda, M. Sakai, and K. Kanitani, “Manipulation of a flexible object by dual manipulators,” in *IEEE Int. Conf. on Robotics and Automation*, vol. 1, 1995, pp. 318–323.
- [10] D. Navarro-Alarcon, Y. Liu, J. G. Romero, and P. Li, “Model-free visually servo deformation control of elastic objects by robot manipulators,” *IEEE Trans. on Robotics*, vol. 29, no. 6, pp. 1457–1468, 2013.
- [11] D. Navarro-Alarcon, Y. Liu, J. G. Romero, and P. Li, “On the visual deformation servoing of compliant objects: Uncalibrated control methods and experiments,” *Int. Journal of Robotics Research*, vol. 33, no. 11, pp. 1462–1480, 2014.
- [12] D. Navarro-Alarcon, H. M. Yip, Z. Wang, Y. Liu, F. Zhong, T. Zhang, and P. Li, “Automatic 3-D manipulation of soft objects by robotic arms with an adaptive deformation model,” *IEEE Trans. on Robotics*, vol. 32, no. 2, pp. 429–441, 2016.
- [13] D. Navarro-Alarcon and Y.-H. Liu, “Fourier-based shape servoing: A new feedback method to actively deform soft objects into desired 2-D image contours,” *IEEE Trans. on Robotics*, 2017.
- [14] S. Hutchinson and F. Chaumette, “Visual servo control, part I: Basic approaches,” *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.