An $SE(3)$-based formulation of the shape servoing problem

Louis Dehaybe, Olivier Brüls
Multibody and Mechatronic Systems Lab
University of Liège
Liège, Belgium
{louis.dehaybe, o.bruls} @uliege.be

***Abstract*—The art of controlling the shape of deformable objects autonomously, a.k.a. shape servoing, remains a challenging task for robots today. While some current works devoted attention to improving controllers, we propose to reconsider the formulation of the problem itself, under the theory of Lie groups. This results in a new representation of the state of the object and leads to a new definition of the shape error. In return, results obtained from simulations demonstrate that the approach can achieve better performance for large deformations tasks. In addition, mobile frame definitions reduce the non-linearity of the problem. Importantly, this approach is not tied to the algorithm developed here and can therefore be easily extended to other types of controllers.**

I. Introduction

The robotic manipulation of deformable objects (DO) questions many assumptions that are taken for granted when manipulating rigid ones. In particular, DOs have in theory an infinite number of degrees of freedom (DOF), while the robotic manipulator controls a few ones only; the problem is therefore highly underactuated [1, 2]. This makes the shape servoing task, i.e. deforming the soft object as desired, considerably hard. In order to solve these challenges, one can distinguish [3] between model-based techniques, whether precomputed [4, 5] or learned [6, 7], and model-free ones [8, 9].

Among model-free techniques, many works rely on 3D points as feedback feature [10, 11, 12]. Instead, the approach presented thereafter distinguishes itself by relying on Lie groups theory. It has found widespread applications in the study of rigid bodies in robotics [13, 14]. It has also been extended to the context of flexible multibody dynamics [15], but to the best of our knowledge, has not yet been employed for the robotic manipulation of DOs. In this line, we reformulate the core of the control algorithm, from which several interesting properties are obtained.

In particular, a new way of representing the shape is provided. Instead of using contours [16] or points, we use mobile frames that are defined directly from the mesh of the DO. As a result, the shape error is also redefined such that it couples position and rotation components. This representation contains rich information, in the sense that for a given number of dimensions, the range of achievable tasks is increased.

In addition, the movement of the feature frame(s) and the gripper(s) are defined in mobile frames. Consequently, non-linearities induced by rigid body motions are filtered out. Thus, the shape Jacobian remains constant under finite rotations, which is not the case of classical approaches. The obtained solution is also independent upon the choice of any global coordinate system. In the next section, we describe a classical approach to the shape servoing problem, in one of its most simple form (i.e. without all the specificities and improvements made by previous research), and we show its adaptation in section III.

II. Background

A common approach [10, 11, 12] is to approximate the configuration of the DO by the position of a subset of N "feature points" that compose it. The problem is then generally posed as follows: let $\mathbf{y} = \left[\mathbf{y}_1^T \, \dots \, \mathbf{y}_N^T\right]^T \in \mathbb{R}^{3N}$ be the vector gathering the current position of each of these points, $\mathbf{y}_i$ being the ith one. Let $\mathbf{y}_d$ be their desired position. It is also assumed that M independent grippers, each having 6 DOF, grasp the deformable object rigidly. The pose of gripper j is noted $\mathbf{z}_j \in \mathbb{R}^6$ and gathers the three position and the three rotation parameters. All the $\mathbf{z}_j$'s are stacked in the vector $\mathbf{z} = \left[\mathbf{z}_1^T \, \dots \, \mathbf{z}_M^T\right]^T \in \mathbb{R}^{6M}$

Then, if the grippers move slowly enough such that the system can be considered to be quasi-static at each instant t, there should exist a nonlinear function $\mathcal{F}(t) : \mathbb{R}^{6M} \rightarrow \mathbb{R}^{3N}$ that maps a configuration of the grippers to the configuration of the feature points, i.e. $\mathbf{y}(t) = \mathcal{F}(\mathbf{z}(t))$. Since $\mathcal{F}(\mathbf{z}(t))$ is a priori unknown, one solution is to linearize around the current configuration: $\frac{\partial \mathbf{y}}{\partial t} = \frac{\partial \mathcal{F}(\mathbf{z})}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial t}$, which leads to

$$\delta \mathbf{y} = \mathbf{J}(\mathbf{z}) \, \delta \mathbf{z} \tag{1}$$

where $\mathbf{J} = \frac{\partial \mathcal{F}(\mathbf{z})}{\partial \mathbf{z}} = \frac{\partial \mathbf{y}}{\partial \mathbf{z}}$ is the so-called shape Jacobian, that relates the motion of the grippers to that of the feature points, and δ indicates an infinitesimal variation. Estimating correctly $\mathbf{J}$ is a key ingredient in this approach. One way, employed by [10, 17], is to observe the evolution of $\mathbf{y}$ and $\mathbf{z}$ and iteratively update $\mathbf{J}$ using the Broyden update rule:

$$\mathbf{J}^t = \mathbf{J}^{t-1} + \alpha \frac{\Delta \mathbf{y}^t - \mathbf{J}^{t-1} \Delta \mathbf{z}^t}{(\Delta \mathbf{z}^t)^T (\Delta \mathbf{z}^t)} (\Delta \mathbf{z}^t)^T \tag{2}$$

where $0 < \alpha \leq 1$ is a parameter controlling the responsiveness of the estimation of $\mathbf{J}$. The infinitesimal variations in (1) are approximated by small but finite variations, i.e. $\Delta\mathbf{y}^t = \mathbf{y}^t - \mathbf{y}^{t-1}$, $\Delta\mathbf{z}^t = \mathbf{z}^t - \mathbf{z}^{t-1}$, the superscript t indicating the time instant. Defining the error between the current and the desired configuration of the feature points as $\mathbf{e} = \mathbf{y}_d - \mathbf{y}$, a simple controller can be designed:

$$\Delta\mathbf{z} = \mathbf{K}\,\mathbf{J}^{\dagger}\mathbf{e} \tag{3}$$

with $\mathbf{K}$ a small positive diagonal gain matrix and $\mathbf{J}^{\dagger}$ the pseudoinverse of $\mathbf{J}$.

One can notice that in the preceding, the motion of the feature points and the grippers are defined in an inertial frame. As a consequence, under a rigid body rotation of the DO, both $\Delta\mathbf{y}$ and $\Delta\mathbf{z}$ would vary in a nonlinear way, and following (2), $\mathbf{J}$ would too.

III. Method

In this section, we modify the control scheme presented above by incorporating concepts of Lie groups theory.

A. Notation and definition of the concepts

The transformation between a frame {O} and a frame {A} can be represented by elements of the special euclidean group SE(3), i.e. the group of homogeneous transformation matrices $\mathbf{H}_{OA} = \begin{bmatrix} \mathbf{R}_{OA} & \mathbf{x}_{OA} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \in SE(3)$ where $\mathbf{x}_{AB} \in \mathbb{R}^3$ is a position vector that represents the origin of frame {A} expressed in frame {O}, and $\mathbf{R}_{OA} \in SO(3)$ (the special orthogonoal group) is a 3×3 rotation matrix that represents the orientation of {A} with respect to {O}, expressed in {O}. An element of $SE(3)$ can also be thought of as an operator that applies a transformation to a frame:

$$\mathbf{H}_{OB} = \mathbf{H}_{OA}\mathbf{H}_r \Leftrightarrow \mathbf{H}_r = (\mathbf{H}_{OA})^{-1}\mathbf{H}_{OB} \tag{4}$$

In this case, $\mathbf{H}_r$ is the relative transformation between $\{A\}$ and $\{B\}$, and the fact that it right multiplies $\mathbf{H}_{OA}$ implies that it is expressed in $\{A\}$. An enjoyable property of this representation is that $\mathbf{H}_r$ is invariant under rigid body motions. Indeed, if we apply an arbitrary transformation $\mathbf{H}_c$ to both $\mathbf{H}_{OA}$ and $\mathbf{H}_{OB}$, we obtain $\mathbf{H}_{OA'} = \mathbf{H}_c\mathbf{H}_{OA}$ and $\mathbf{H}_{OB'} = \mathbf{H}_c\mathbf{H}_{OB}$. The relative transformation between $\{A'\}$ and $\{B'\}$ is thus $(\mathbf{H}_{OA'})^{-1}\mathbf{H}_{OB'}$, which is equal to $(\mathbf{H}_{OA})^{-1}\mathbf{H}_{OB} = \mathbf{H}_r$.

We build on this advantage to adapt (1) to (3) as follows. It is assumed that the DO can be represented by a mesh composed of triangles. Among these, a subset of size N is tied to a desired configuration; they are called "feature faces". Their selection depends on the task objective. In practice, each feature face is attributed a Darboux frame $\{P\}$ defined from the position x_i, x_j, x_k of its three vertices : $t_1 = x_{ji}$, $n = t_1 \times x_{ki}$, $t_2 = n \times t_1$, with $x_{ji} = x_j - x_i/\|x_j - x_i\|$ and $x_{ki} = x_k - x_i/\|x_k - x_i\|$. The origin of the frame, $\mathbf{x}_P$, is taken as the center of the triangle. The transformation is therefore

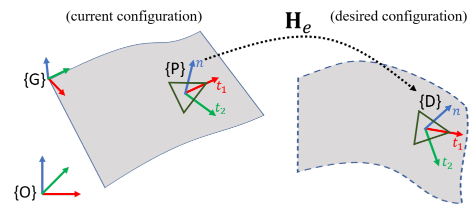

Fig. 1. Definition of the frames

defined as $\mathbf{H}_P = \begin{bmatrix} \mathbf{R}_P & \mathbf{x}_P \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}$, with $\mathbf{R}_P = \begin{bmatrix} t_1 & t_2 & n \end{bmatrix}$. Likewise, we build the frame transformation $\mathbf{H}_D$ from the same three vertices on the deformed configuration of the DO. Finally, $\mathbf{H}_G$ is simply the pose of the gripper (known from the forward kinematics). The situation is illustrated in fig. 1.

Now, let us consider that the grippers and the feature faces' frames vary with time. An infinitesimal variation of an element of $SE(3)$ is written $\delta\mathbf{H}_{OA} = \mathbf{H}_{OA}\,\widetilde{\delta\boldsymbol{\pi}}_A^A$, where $\delta\boldsymbol{\pi}_A^A \in \mathbb{R}^6$ is an infinitesimal motion of frame A (indicated by the subscript) expressed in A (indicated by the superscript), i.e. the local frame representation. $\widetilde{\delta\boldsymbol{\pi}}$ belongs to the Lie algebra of $SE(3)$, noted $\mathfrak{se}(3)$, which is a linear space. It is therefore isomorphic to $\mathbb{R}^6$ through the map $\mathbb{R}^6 \rightarrow \mathfrak{se}(3) : \delta\boldsymbol{\pi} \rightarrow \widetilde{\delta\boldsymbol{\pi}}$. From now on, only the local frame representation will be used, and the superscript is therefore dropped.

B. Adaptation of the method

With these definitions in mind, (1) is reformulated as:

$$\delta\boldsymbol{\pi}_p = \mathbf{J}\,\delta\boldsymbol{\pi}_g \tag{5}$$

with $\delta\boldsymbol{\pi}_p = \begin{bmatrix}\delta\boldsymbol{\pi}_{P_1}^T \dots \delta\boldsymbol{\pi}_{P_N}^T\end{bmatrix}^T \in \mathbb{R}^{6N}$, $\delta\boldsymbol{\pi}_g = \begin{bmatrix}\delta\boldsymbol{\pi}_{G_1}^T \dots \delta\boldsymbol{\pi}_{G_M}^T\end{bmatrix}^T \in \mathbb{R}^{6M}$, and $J \in \mathbb{R}^{6N \times 6M}$.

Accordingly, the method is reshaped as follows. First, an initialisation of $\mathbf{J}$ is required. Several methods exist. In this work, each DOF of the gripper is perturbed by a small amount, one at a time, and the column of $\mathbf{J}$ corresponding to the perturbed DOF is obtained using standard finite differences.

Next, we adapt the Broyden update rule to estimate $\mathbf{J}$ as:

$$\mathbf{J}^t = \mathbf{J}^{t-1} + \alpha\frac{\Delta\mathbf{u}_p - \mathbf{J}^t\Delta\mathbf{u}_g}{(\Delta\mathbf{u}_g)^T(\Delta\mathbf{u}_g)}(\Delta\mathbf{u}_g)^T \tag{6}$$

where we replace infinitesimal motions δ by small but finite variations Δ, such that $\Delta\mathbf{u}_p = \begin{bmatrix}\Delta\mathbf{u}_{P_1}^T \dots \Delta\mathbf{u}_{P_N}^T\end{bmatrix}^T \in \mathbb{R}^{6N}$, with $\widetilde{\Delta\mathbf{u}}_{P_i} = \log(\mathbf{H}_{P_i^{t-1}}^{-1}\mathbf{H}_{P_i^t})$. The subscript i indicates the feature frame considered, while t is the time instant. log refers to the logarithmic map (see more details in [15]) that maps an element of the Lie group to the Lie algebra. $\Delta\mathbf{u}_g \in \mathbb{R}^{6M}$ is defined analogously, replacing feature frames $\{P\}$ by gripper frames $\{G\}$. Importantly, one can highlight that the argument given to log takes a similar form as $\mathbf{H}_r$ in (4), and for the same reasons, it enjoys the invariance property with respect to any rigid body motion. Due to (6), since both $\Delta\boldsymbol{\pi}_p$ and $\Delta\boldsymbol{\pi}_g$ are

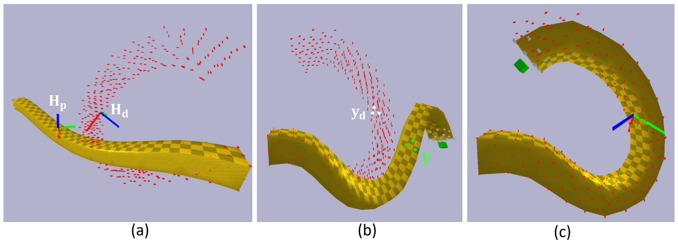

Fig. 2. (a) Start state (b) End state using method 1 and (c) using method 2

coordinate-independent and insensitive to superimposed frame transformations, so will be $\mathbf{J}$.

Then, this new framework also leads to a redefinition of the shape error. For the i^{th} feature, it is defined as $\mathbf{H}_{e_i} = \mathbf{H}_{P_i D_i} = \mathbf{H}_{P_i}^{-1}\mathbf{H}_{D_i}$ and its Lie algebra counterpart is

$$\widetilde{\Delta \mathbf{u}}_{e_i} = \log(\mathbf{H}_{e_i}) \tag{7}$$

$\widetilde{\Delta \mathbf{u}}_{e_i}$ is thus expressed in the mobile frame $\{P\}$. It is important to notice the introduction of the rotation in the error (since it is contained in the transformation matrices), and its coupling to position variables via $SE(3)$. This modification is precisely at the origin of the difference between the trajectory followed by the grippers in this method, compared to the former one that uses positions only to define the error.

Consequently, the control law (3) needs to be adapted as follows:

$$\Delta \mathbf{u}_g = \mathbf{K}\, \mathbf{J}^{\dagger} \Delta \mathbf{u}_e \tag{8}$$

where $\Delta \mathbf{u}_e = \left[\Delta \mathbf{u}_{e_1}^T \ldots \Delta \mathbf{u}_{e_N}^T\right]^T$. The output is therefore directly obtained in the coordinate system of the gripper.

Lastly, for each gripper i, we can use either position control: $\mathbf{H}_{G_i^{t+1}} = \mathbf{H}_{G_i^t}\ \exp(\widetilde{\Delta \mathbf{u}_{G_i}})$ where exp is the exponential map (see [15]), or velocity control by simply dividing the desired motion increment by the time step size h: $\mathbf{v}_g = \frac{1}{h}(\Delta \mathbf{u}_g)$

IV. Experiments and Results

In what follows, position-based servoing (PBS) refers to the method presented in section II, while SE(3)-based servoing (SE3BS) refers to the adapted one of section III. To demonstrate the difference between both, several experiments have been performed in simulation (only one result is shown for conciseness). The simulator, Pybullet [18], has been chosen for its ability to simulate soft bodies. In particular, a beam-like object is modelled as an hyperelastic material that follows a Neo-Hookean law. We emphasize that the controller has no access to the DO model. Without hurting the generality of the method, the gripper is considered as ”floating”, i.e. it is not attached to a robot. It is represented by the green cube (see fig. 2). The white cubes are the grasped nodes. The other side of the beam has been clamped, and the only external force is due to gravity. To obtain the target shape, in practice, the grippers are manually moved and the position of all the vertices of the deformed mesh are recorded. This ensures that the target is reachable from the starting configuration. The gain matrix is $\mathbf{K} = 0.02\,\mathbf{I}_6$ and $\alpha = 0.6$. The beam starts fully extended (fig. 2a), and the recorded shape is as represented by the red dots. Note that they are shown only for visualization purposes. Indeed, in SE3BS, the algorithm only knows the state of one feature frame (fig. 2c) that it uses to calculate the shape error (hence, $P = 1$). In PBS, only the position of the 3 white dots ($P = 3$) is used (fig. 2b). To ensure a fair comparison between methods, these three points are the same as those used to build the Darboux frame in method 2.

The results of such a task show the added value of working with frame transformations to represent the shape, and the advantage of adding rotation components in the error. Indeed, during the whole deformation, twists of the beam are avoided as much as possible by SE3BS because it degrades the rotational part of the error. The trajectory is consequently smoother and more straightforward. Starting from 2.03, $\|\Delta \mathbf{u}_e\| < 0.03$ is reached (the current and target frame coincide in fig. 2c) after a simulation time of 20.7s. In contrast, after the same time, PBS does not manage to bring the points to the desired locations, lowering $\|\mathbf{e}\|$ from 1.31 to 1.2 only. At the beginning of the task, twisting the beam improves the reduction in position error, but it leads the controller in a deadlock near the end (i.e. the final state shown), from which it can not recover. Notice also that the second method reduces a little bit the dimension of the problem ($\Delta \mathbf{u}_e \in \mathbb{R}^6$ against $\mathbf{y} \in \mathbb{R}^9$).

The methods have been tested in other setups. In general, for simple tasks where the deformation involves translations mostly, both achieve similar performance. However, in many cases, it allows to succeed in high deformation tasks that PBS can not execute properly because of deadlocks such as the aforementioned one.

Finally, we tested the case were the beam was subjected to a rigid body rotation while held by a gripper at both ends. It was observed, as expected, that the components of $\mathbf{J}$ varied non linearly while in the case of SE3BS, it remained constant. Therefore, it confirms that (6) only captures changes in the shape of the object, as it is supposed to.

V. Conclusion and Outlook

In this work, a Lie group approach is used to tackle the shape servoing problem. It involves a new representation of the DO which couples translation and rotation components. Starting from a typical controller used in this field, we adapt the method according to these considerations. Some experiments have been carried out in simulation and showed that this alternative is more appropriate to execute certain high deformation tasks. We emphasize that the framework developed here is not specific to the controller used in this work, nor to the Broyden update rule. Hence, future works will be devoted to the integration of this theory into more efficient shape Jacobian estimators and controllers. Finally, experiments on a real setup will be proposed to validate the method.

Acknowledgment

This work was supported by the Interreg VA programme Euregio Meuse-Rhine (European Fund for Regional Development), within the AACOMA project.

REFERENCES

[1] J. Zhu, A. Cherubini, C. Dune, D. Navarro-Alarcon, F. Alambeigi, D. Berenson, F. Ficuciello, K. Harada, J. Kober, X. Li *et al.*, “Challenges and outlook in robotic manipulation of deformable objects,” *IEEE Robotics and Automation Magazine*, 2021.

[2] J. Das, *Robotic control of deformable continua and objects therein*. Vanderbilt University, 2010.

[3] R. Herguedas, G. López-Nicolás, R. Aragüés, and C. Sagüés, “Survey on multi-robot manipulation of deformable objects,” in *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2019, pp. 977–984.

[4] F. Ficuciello, A. Migliozzi, E. Coevoet, A. Petit, and C. Duriez, “Fem-based deformation control for dexterous manipulation of 3d soft objects,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4007–4013.

[5] J. Das and N. Sarkar, “Robust shape control of deformable objects using model-based techniques,” *Advanced Robotics*, vol. 25, no. 16, pp. 2099–2123, 2011.

[6] C. Shin, P. W. Ferguson, S. A. Pedram, J. Ma, E. P. Dutson, and J. Rosen, “Autonomous tissue manipulation via surgical robot using learning based model predictive control,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3875–3881.

[7] Z. Hu, P. Sun, and J. Pan, “Three-dimensional deformable object manipulation using fast online gaussian process regression,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 979–986, 2018.

[8] D. Berenson, “Manipulation of deformable objects without modeling and simulating deformation,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4525–4532.

[9] M. Aranda, J. A. C. Ramon, Y. Mezouar, A. Bartoli, and E. Özgür, “Monocular visual shape tracking and servoing for isometrically deforming objects,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 7542–7549.

[10] F. Alambeigi, Z. Wang, R. Hegeman, Y.-H. Liu, and M. Armand, “A robust data-driven approach for online learning and manipulation of unmodeled 3-d heterogeneous compliant objects,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4140–4147, 2018.

[11] M. Shetab-Bushehri, M. Aranda, Y. Mezouar, and E. Ozgur, “As-rigid-as-possible shape servoing,” *IEEE Robotics and Automation Letters*, 2022.

[12] D. McConachie and D. Berenson, “Estimating model utility for deformable object manipulation using multiarmed bandit methods,” *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 967–979, 2018.

[13] J. M. Selig, *Geometric fundamentals of robotics*. Springer, 2005, vol. 128.

[14] K. M. Lynch and F. C. Park, *Modern robotics*. Cambridge University Press, 2017.

[15] V. Sonneville, “A geometric local frame approach for flexible multibody systems,” Ph.D. dissertation, Université de Liège, Liège, Belgique, 2015.

[16] 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 Transactions on Robotics*, vol. 34, no. 1, pp. 272–279, 2017.

[17] J. Huang, Y. Cai, X. Chu, R. H. Taylor, and K. S. Au, “Non-fixed contact manipulation control framework for deformable objects with active contact adjustment,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2878–2885, 2021.

[18] E. Coumans and Y. Bai, “Pybullet, a python module for physics simulation for games, robotics and machine learning,” http://pybullet.org, 2016–2021.