Robotic Guidance of a Surgical Tool Under Constrained Motions

Bassem Dahroug, Brahim Tamadazte, Patrick Rougeot, Nicolas Andreff

Robotic Guidance of a Surgical Tool under Constrained Motions

Bassem Dahroug[1], Brahim Tamadazte[2], Patrick Rougeot[1], and Nicolas Andreff[1]

***Abstract*—This article formulates a generic representation of a path-following controller operating under contained motion, which was developed in the context of surgical robotics. It reports two types of constrained motion: i) Bilateral Constrained Motion, also called Remote Centre Motion (RCM), and ii) Unilaterally Constrained Motion (UCM). To deal with the constrained motion of a surgical tool when performing a path-following task inside a cavity, we investigated methods that combine the two tasks operating in a hierarchical manner. The proposed solutions were successfully evaluated, first on our simulator that mimics realistic conditions of middle ear surgery, then on an experimental platform. Different validation scenarios were carried out experimentally to assess quantitatively and qualitatively each developed approach. Although ultimate precision was not the goal of this work, our concept is validated with enough accuracy ($\leq 100\ \mu m$) for the ear surgery.**

***Index Terms*—Medical Robotics, Constrained Motion, Path Following, Visual Servoing, Middle Ear.**

I. INTRODUCTION

Surgical robots are gaining more popularity due to their advantages for both the patient and the physician [1]–[3]. It is particularly valid for MIS (Minimally-Invasive Surgery) approaches. For instance, a keyhole surgery [4] performs incision around $10\ mm$. It is tiny compared to the larger incisions needed in laparotomy (open surgery). Another situation is where the surgical instruments could be inserted through a natural orifice (e.g., mouth, nasal clefts, urethra, anus) to reach the targeted organ. In both cases, the entry space (i.e., the incision hole or the natural orifice) restricts the surgical tool motion, consequently the surgeon's hands and the robot carrying the instrument [5].

This article mainly discusses two types of constrained motion that result directly from MIS procedures:

- *Remote Centre Motion* (RCM), also known as *fulcrum effect*, implies the incision hole has almost the same diameter as that of the surgical tool [6];
- *Unilaterally Centre Motion* (UCM) implies the incision diameter size is bigger than that of the tool, offering more freedom for the tool motion, for instance [7].

The first type of motion was initially achieved by designing a particular robotic structure that imposes the constrained motion mechanically [6], [8]. The RCM dictates that the centre line of the surgical tool is always coincident with the centre point of the incision orifice (trocar point). Consequently, the linear movement of the tool is restricted along two axes. The main advantage of RCM mechanisms is to reduce the risk of damaging the trocar wall because their kinematic structures ensure the pivoting motion. Their modest controller is also easy to implement. However, this kind of mechanism is restricted to a unique configuration and cannot provide enough flexibility for shifting the location of the trocar point.

An alternative solution proposes a software RCM for overcoming the previous problem by guiding a general-purpose robot with the advantage of being flexible enough for achieving complex tasks [9]. This solution is convenient for diverse medical applications (e.g., laparoscopic and eye surgeries). However, the RCM approach is not the best choice for other surgery types (e.g., ear, nose, mouth, and knee arthroscopy). In latter cases, the orifice diameter is generally bigger than the tool diameter. Consequently, the RCM controller imposes too strong limitations on the tool motion. Indeed, the RCM is a mathematical equality constraint (i.e., the distance between the tool body and the centre point of the incision orifice must be equal to zero). As such, RCM motion can be named a bilaterally constrained motion. On the contrary, UCM is a weaker restriction since the unilateral constraints are inequality equations (i.e., the latter distance could be greater or less than zero) [11].

In the literature, the term forbidden-region virtual fixtures [12] is used for collaboration tasks where the user can either manipulate a robotic device [13] or telemanipulate a master device [14], [15]. These fixtures could be defined as geometric forms [7], [16] or vector field [18] around the tool. Then a kinematic control [7] or dynamic one [16], [18] is applied to guide the robot during the desired task.

The work described in this article extends a first approach proposed in [17] which allows controlling a straight rigid instrument during the path-following task while respecting an RCM-type constraint. The contributions of this article are therefore to address both RCM and UCM constraints while performing a 3D path-following task in a hierarchical manner. As well, the new controller is designed to operate with either a straight or curved rigid instrument. The proposed methods were evaluated in a realistic simulator which includes surgical tools (straight or curved) and a digital twin mimicking the middle ear cavity. We also designed an experimental set-up which reproduces the simulator in a pre-clinical context, i.e. the instruments and the middle ear cavity are at a 1:1 scale. Moreover, the paths used in the validation are similar to a clinical procedure during a mechanical cholesteatoma resection. Different validation scenarios were proposed to

This work was supported by the Inserm ROBOT Project: ITMO Cancer no 17CP068-00.

[1] are with FEMTO-ST Institute, Supmicrotech-ENSMM, Université de Franche-Comté, CNRS, Besançon, France.

[2] is with Sorbonne Université, CNRS UMR 7222, INSERM U1150, ISIR, F-75005, Paris, France. brahim.tamadazte@cnrs.fr

assess the methods demonstrating good results in terms of behaviour and accuracy.

II. MEDICAL MOTIVATIONS

A. Treated Disease

The work discussed in this article represents a part of a long-term project. It deals with the development of a robotic system that is dedicated to cholesteatoma surgery. The system will aim to achieve an MIS within the middle ear cavity by passing through the external ear canal or an incision orifice made on the mastoid portion.

Cholesteatoma is a frequent disease that invades the middle ear. It infects the middle ear by introducing abnormal skin (lesional tissue) in the middle ear cavity. The most common explanation [19] is due to the immigration of the epidermal cells, which are the cells type in the external ear canal, and cover up the mucosa of the middle ear cavity, as shown in Fig. 1. These cells gradually proliferate within the temporal bone and destroy the adjacent bony structures.

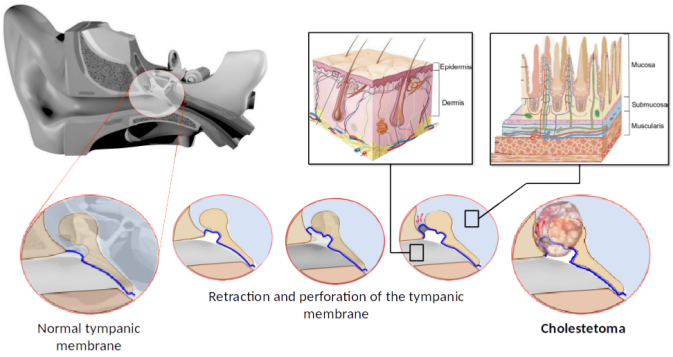

Fig. 1. Evolution of cholesteatoma disease within the middle ear, which is located behind the tympanic membrane.

The evolution of cholesteatoma is life-threatening in the long run. Different complications can occur, the destruction of the ossicular chain, facial paralysis, labyrinthitis, etc. It can notice the irreversible effects that cholesteatoma can cause in a patient. The only solution for cholesteatoma treatment is surgical intervention.

B. Current Surgical Procedure

As claimed above, the only treatment for cholesteatoma is a surgical procedure. It aims to eradicate all cholesteatoma tissue and reconstruct the anatomy of the middle ear [20].

For reaching the middle-ear cavity, the surgeon often drills the temporal bone behind the auricular, as shown in Fig. 2. This surgical procedure is called *mastoidectomy* where the surgeon maintains the wall of the external ear canal. This technique creates an incision that forms a triangular (around $40 \times 40 \times 30\ mm$) with a depth of about $30\ mm$ The latter procedure can also become more invasive by sacrificing the posterior portion of the external ear canal (i.e., *canal-wall-down*). Furthermore, although the surgical orifice is relatively large, the surgical procedure remains complex and requires high expertise and dexterity from the surgeon. Also, even with

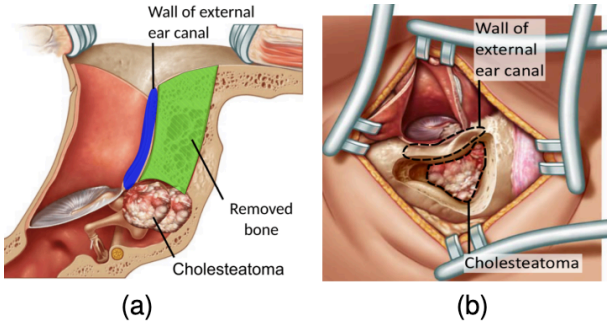

Fig. 2. Mastoidectomy procedure with canal-wall-up indicates that the external ear canal is preserved. (a) the side view of the mastoidectomy tunnel and (b) the top view of the mastoidectomy tunnel.

an experienced clinician in the cholesteatoma case, the clinical outcomes remain unsatisfactory in terms of effectiveness. Indeed, there is a high risk that the cholesteatoma could regrow a few months after the surgical intervention. It occurs due to residual cholesteatoma cells. Consequently, 10 to 40% of patients perform more than one surgery to get definitively over this disease [21].

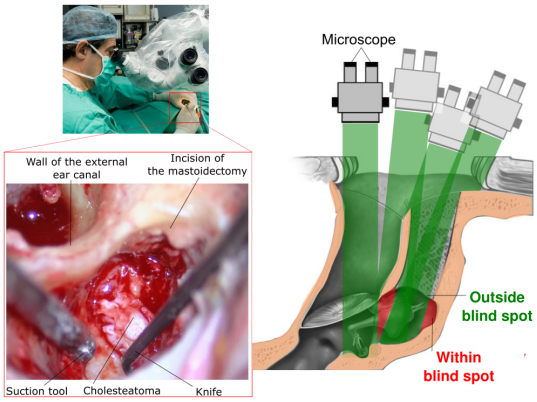

Fig. 3. Conceptual scheme to demonstrate the "blind spot" during the cholesteatoma surgery.

Due to the complexity of the temporal bone cavity, the surgeon mainly faces numerous difficulties during the procedure (Fig. 3): i) lack of ergonomy of the tools; ii) limited field of view of the oto-microscope (the surgeon cannot visualize the lateral regions hidden in the middle ear cavity) and iii) access with the conventional rigid instruments requires considerable expertise to handle. Therefore, it is increasingly important to overcome the previous problems and evolve this procedure towards less invasive. It implies reducing the incision orifice size, improving the cholesteatoma resection efficiency, and avoiding the current high surgical recurrence rate for this kind of surgery.

III. METHODOLOGY

This section begins by presenting a brief summary of the new surgical protocol associated with the robotic system. After that, it discusses the hierarchical controller for managing simultaneously the various tasks. It then explains separately the path-following, the RCM, and the UCM controllers.

A. Towards a New Surgical Protocol

In collaboration with expert surgeons, we have attempted to set up a new and more efficient surgical protocol reported in [2]. Firstly, the idea is to make cholesteatoma surgery less invasive compared to the traditional one. Thus, a macro-micro robotic system should pass through a millimetric incision made behind the ear (in the mastoid portion) to access the middle ear cavity [22], [23]. Secondly, cholesteatoma surgery needs to be more efficient by eliminating residual cases. This second objective can be accomplished by removing a large part of the cholesteatoma tissue using rigid miniature mechanical resection tools. After that, a bendable actuated tool [24] could be used to guide a laser fibre. This fibre carbonizes the residual cholesteatoma (resulting from the mechanical resection phase) [25], [26]. Both mechanical resection and laser ablation should be performable either in automatic or semi-automatic mode. While the mechanical resection does not require high accuracy, the laser ablation requires higher precision since the residual cholesteatoma cells can be a few tens of micrometres in size. Therefore, the contributions of robotics and vision-based control are essential for this task. In this work, we investigated the use of path-following control schemes under constrained motion (due to the incision orifice) to carry out the motions requested by cholesteatoma removal (i.e., mechanical resection and laser ablation).

B. Background

1) Task Hierarchical Controller: A surgical procedure can be considered as a set of sequential or overlapping sub-tasks. The hierarchical methods ensure the execution of several tasks simultaneously. Consequently, the required tasks do not enter into conflict [27], [28]. In the case of cholesteatoma surgery, various sub-tasks can be involved during the procedure, such as constraint enforcement (RCM or UCM) and ablation tools for the pathological tissues. Therefore, these sub-tasks must be carried out according to a defined hierarchical scheme.

To express a controller that manages simultaneous sub-tasks, let us start by assuming that a generic sub-task ($\dot{\mathbf{e}}_i \in \mathbb{R}^{m_i}$) given by:

$$\dot{\mathbf{e}}_i = \mathbf{L}_i \ {}^e\underline{\mathbf{v}}_e, \quad \text{where i=1,2,...,j} \tag{1}$$

where ${}^e\underline{\mathbf{v}}_e \in se(3)$ is the tool twist velocity to be computed in the end-effector frame $\mathcal{F}_e$, and $\mathbf{L}_i \in \mathbb{R}^{m_i \times n}$ is the interaction matrix which relates the vector ${}^e\underline{\mathbf{v}}_e$ to the error $\dot{\mathbf{e}}_i$.

The inverse solution of (1) is not guaranteed since the interaction matrix $\mathbf{L}_i$ could be non-square, and the matrix rank is locally deficient. Thanks to the least-square method, an approximate solution can be found by minimizing $\|\dot{\mathbf{e}}_i - \mathbf{L}_i \ {}^e\underline{\mathbf{v}}_e\|$ over ${}^e\underline{\mathbf{v}}_e$, and using numerical procedures (such as QR or SVD). The formal result of it can be simply written as ${}^e\underline{\mathbf{v}}_e = \mathbf{L}_i^{\dagger}\dot{\mathbf{e}}_i$, where $\mathbf{L}_i^{\dagger}$ is the pseudo-inverse of $\mathbf{L}_i$. If $\mathbf{L}_i$ does not have full-rank then it has at least one singular vector $\mathbf{z}_1$, located in its null-space ($\mathbf{L}_i\mathbf{z}_1 = \mathbf{0}$). The vector $\mathbf{z}_1$ is also described as the null-space of $\mathbf{e}_i$, because any twist vector parallel to $\mathbf{z}_1$ will leave $\mathbf{e}_i$ unchanged. Therefore, the projection gradient general form [29] is given by:

$${}^e\underline{\mathbf{v}}_e = \mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1 + (\mathbf{I} - \mathbf{L}_1^{\dagger}\mathbf{L}_1)\mathbf{z}_1 \tag{2}$$

In order to define $\mathbf{z}_1$, let us first consider a secondary sub-task $\dot{\mathbf{e}}_2 = \mathbf{L}_2 \ {}^e\underline{\mathbf{v}}_e$. Since the control vector must include the first sub-task, (2) is injected in the latter expression, resulting in:

$$\begin{aligned}\dot{\mathbf{e}}_2 &= \mathbf{L}_2\left(\mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1 + (\mathbf{I} - \mathbf{L}_1^{\dagger}\mathbf{L}_1)\mathbf{z}_1\right) \\ &= \mathbf{L}_2\mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1 + \underbrace{\mathbf{L}_2(\mathbf{I} - \mathbf{L}_1^{\dagger}\mathbf{L}_1)}_{\tilde{\mathbf{L}}_2}\mathbf{z}_1\end{aligned} \tag{3}$$

Thereby, $\mathbf{z}_1$ is deduced as:

$$\mathbf{z}_1 = \tilde{\mathbf{L}}_2^{\dagger}(\dot{\mathbf{e}}_2 - \mathbf{L}_2\mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1) + (\mathbf{I} - \tilde{\mathbf{L}}_2^{\dagger}\tilde{\mathbf{L}}_2)\mathbf{z}_2 \tag{4}$$

with another criterion vector $\mathbf{z}_2$ which is projected in the null-space of the secondary sub-task.

By introducing (4) in (2), a recursive form of the projection gradient is obtained as:

$$\begin{aligned}{}^e\underline{\mathbf{v}}_e &= \mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1 + (\mathbf{I} - \mathbf{L}_1^{\dagger}\mathbf{L}_1)\left(\tilde{\mathbf{L}}_2^{\dagger}(\dot{\mathbf{e}}_2 - \mathbf{L}_2\mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1) + (\mathbf{I} - \tilde{\mathbf{L}}_2^{\dagger}\tilde{\mathbf{L}}_2)\mathbf{z}_2\right) \\ &= \mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1 + (\mathbf{I} - \mathbf{L}_1^{\dagger}\mathbf{L}_1)\tilde{\mathbf{L}}_2^{\dagger}(\dot{\mathbf{e}}_2 - \mathbf{L}_2\mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1) \\ &\quad + (\mathbf{I} - \mathbf{L}_1^{\dagger}\mathbf{L}_1)(\mathbf{I} - \tilde{\mathbf{L}}_2^{\dagger}\tilde{\mathbf{L}}_2)\mathbf{z}_2\end{aligned} \tag{5}$$

The right-hand side of the previous equation can further be simplified as [30]

$${}^e\underline{\mathbf{v}}_e = \mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1 + \tilde{\mathbf{L}}_2^{\dagger}(\dot{\mathbf{e}}_2 - \mathbf{L}_2\mathbf{L}_1^{\dagger}\dot{\mathbf{e}}_1) \tag{6}$$

The latter equation finds a solution to satisfy both sub-tasks $\dot{\mathbf{e}}_1$ and $\dot{\mathbf{e}}_2$. It also ensures a form of hierarchy/priority between them. The analytical expression of each sub-task with its $\mathbf{L}_i$ is presented in the coming sections.

2) 3D Visual Servoing: This section is dedicated to describing how to control the tool-tip for regulating its position and orientation with respect to a reference frame, e.g., the orifice frame $\mathcal{F}_r$. This task is applied when the tool locates outside the incision orifice, and its pose must be adjusted with respect to the orifice before it starts another task inside the orifice.

To do this, a traditional 3D position-based visual servoing [31] is applied. The feature vector $\mathbf{s} = ({}^r\mathbf{t}_t, \theta \ {}^r\mathbf{u}_t)$ is defined as the pose vector which describes the tool-tip frame $\mathcal{F}_t$ with respect to the orifice frame $\mathcal{F}_r$. This vector gathers the translation $\mathbf{t}$ of the tool-tip and its rotation $\theta\mathbf{u}$ in the form of angle/axis parameterization. The desired feature vector $\mathbf{s}^* = (\mathbf{0}, \mathbf{0})$ is set to a zero vector since it is required to make coincident the frame $\mathcal{F}_t$ with $\mathcal{F}_r$. Thus, the approach task error $\mathbf{e}_{app}$ is deduced as the difference between the current features vector and the desired one, i.e.,

$$\mathbf{e}_{app} = \mathbf{s} - \mathbf{s}^* \tag{7}$$

The time variation of the latter error is related to the spatial velocity of the tool-tip ${}^t\underline{\mathbf{v}}_t$ by the interaction matrix $\mathbf{L}_{3D} \in \mathbb{R}^{6\times 6}$ as:

$$\dot{\mathbf{e}}_{app} = \mathbf{L}_{3D} \ {}^t\underline{\mathbf{v}}_t \tag{8}$$

where ${}^t\underline{\mathbf{v}}_t = ({}^t\mathbf{v}_t, {}^t\omega)$ gathers the instantaneous linear and angular velocities of the tool-tip.

Since the desired feature vector equals to $\mathbf{0}_{6\times 1}$, then the interaction matrix $\mathbf{L}_{3D}$ is determined by:

$$\mathbf{L}_{3D} = \begin{bmatrix} -\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{L}_{\theta\mathbf{u}} \end{bmatrix} \tag{9}$$

where $\mathbf{I}_{3\times3}$ is a 3×3 identity matrix, $\mathbf{0}_{3\times3}$ is a 3×3 zero matrix, and $\mathbf{L}_{\theta\mathbf{u}}$ is given by [32]:

$$\mathbf{L}_{\theta\mathbf{u}} = \mathbf{I}_{3\times3} - \frac{\theta}{2}[\mathbf{u}]_\times + \left(1 - \frac{\operatorname{sinc}\theta}{\operatorname{sinc}^2\frac{\theta}{2}}\right)[\mathbf{u}]_\times^2 \tag{10}$$

in which $\operatorname{sinc} x$ is the cardinal sinus.

Finally, the spatial velocity ${}^t\underline{\mathbf{v}}_t$ is determined for ensuring an exponential decoupled decrease of the error (i.e., $\dot{\mathbf{e}} = -\lambda\mathbf{e}$) as:

$${}^t\underline{\mathbf{v}}_t = -\gamma\mathbf{L}_{3D}^{-1}\mathbf{e}_{app} \tag{11}$$

where γ is a positive gain coefficient, and $\mathbf{L}_{3D}^{-1}$ is the inverse of the interaction matrix since it is square and has a closed-form inverse [32].

The command velocity of the robot end-effector ${}^e\underline{\mathbf{v}}_e = {}^e\mathbf{V}_t\,{}^t\underline{\mathbf{v}}_t$ is deduced by the following twist matrix:

$${}^e\mathbf{V}_t = \begin{bmatrix} {}^e\mathbf{R}_t & [{}^e\mathbf{t}_t]_\times\,{}^e\mathbf{R}_t \\ \mathbf{0}_{3\times3} & {}^e\mathbf{R}_t \end{bmatrix} \tag{12}$$

since the tool body is rigid and the transformation between the end-effector frame $\mathcal{F}_e$ and the tool-tip frame $\mathcal{F}_t$ is fixed. Finally, the controller stability was demonstrated in [32] to be globally exponentially stable.

3) 3D Path-Following: This section introduces the 3D path-following controller formulation for which a prelude work was published in [17]. In this first work, it was shown that modelling a surgical procedure as a path-following problem is very relevant and justified. Among the advantages raised of using a such controller is, for instance, the separation between i) the geometric curve (desired path $\mathcal{S}_p$) which is planned by the surgeon based on pre-operative images, and ii) the advance speed (v_{tis}) of the tool-tip along the desired path which is controlled by the surgeon during the operation [33]. In this manner, the collaboration surgeon/robot ensures that the robot guides the tool along the path while the surgeon controls the robot's progression without planning the robot's velocity direction.

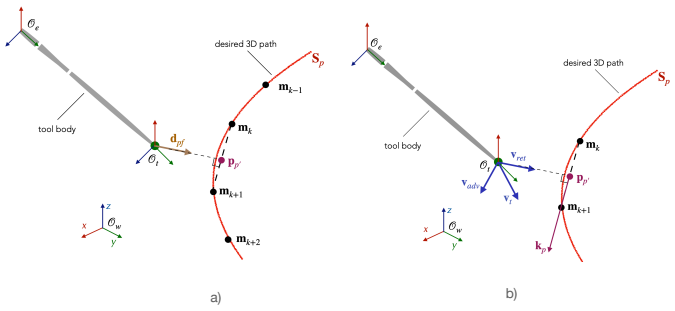

Fig. 4. (a) orthogonal projection of the tool-tip onto a geometric curve and (b) representation of the different velocities involved in the path-following controller.

Fig. 4(a) depicts the surgical instrument and its reference frames with respect to the desired path $\mathcal{S}_p$. By projecting the tool-tip $\mathcal{O}_t$ onto the reference path, the resultant orthogonal distance $\mathbf{d}_{pf}$ is considered as the error (i.e., lateral deviation) which must be controlled to zero. Therefore, the 3D vector distance between the tool-tip $\mathcal{O}_t$ and the projection point $\mathbf{p}_{p'}$ calculated as:

$$\mathbf{d}_{pf} = \mathcal{O}_t - \mathbf{p}_{p'} \tag{13}$$

In order to express the command velocity, the time-derivative of (13) provides the tool-tip velocity $\mathbf{v}_t$ as discussed in [17]:

$$\dot{\mathbf{d}}_{pf} = \left(\mathbf{I}_{3\times3} - \frac{\mathbf{k}_p\mathbf{k}_p^\top}{1 - \mathbf{d}_{pf}^\top\left(\mathbf{C}_p(s_p)\times\mathbf{k}_p\right)}\right)\mathbf{v}_t \tag{14}$$

where $\mathbf{C}_p(s_p)$ is the path curvature in the function of the path curve length, $\mathbf{k}_p$ is the unit-vector of the instantaneous tangential vector (Fig. 4).

At this stage, it requires choosing the adequate velocity of the tool-tip $\mathbf{v}_t$ (15) to ensure that the lateral error $\mathbf{d}_{pf}$ is regulated to zero while progressing along the path. An intuitive solution consists of decomposing the control velocity into two orthogonal components (Fig. 4(b)): i) the advance velocity ($\mathbf{v}_{adv}$) along the path, and ii) the return velocity ($\mathbf{v}_{ret}$) for regulating the tool deviation from the reference path. This can be formulated as follows:

$$\mathbf{v}_t = \underbrace{\alpha\mathbf{k}_p}_{\mathbf{v}_{adv}} + \underbrace{\beta\mathbf{d}_{pf}}_{\mathbf{v}_{ret}} \tag{15}$$

The tuning coefficients of the controller α and β allow adjusting the priority between the advance and return velocities, respectively. Besides that, the controller stability demonstrated in [17] shows that α should be a positive scalar while β must be a negative scalar to ensure the system stability.

The choice of these gain factors can be imposed by a function of a constant velocity $v_{tis} > 0$ that depends on the interaction between the surgical tool and the tissue. Therefore, (15) yields:

$$\underbrace{v_{tis}^2}_{=\|\mathbf{v}_t\|^2} = \alpha^2\underbrace{\|\mathbf{k}_p\|^2}_{=1} + \underbrace{\beta^2\|\mathbf{d}_{pf}\|^2}_{=\|\mathbf{v}_{ret}\|^2} \tag{16}$$

The gain factor α is thus determined as:

$$\alpha = \begin{cases} \sqrt{v_{tis}^2 - \|\mathbf{v}_{ret}\|^2} & \|\mathbf{v}_{ret}\|^2 < v_{tis}^2 \\ 0 & \|\mathbf{v}_{ret}\|^2 > v_{tis}^2 \end{cases} \tag{17}$$

If the tool is not far from the reference path, the first condition in (17) is selected. Otherwise, the priority is returning the tool-tip to the reference path, and the advance velocity is null (i.e., a second condition in (17)).

In our previous work [17], we proposed a constant value for the gain factor β. However, for better behaviour of the path-following controller, we investigated in this paper an adaptive gain β as a function of the curvature of the path formulated as follows:

$$\beta = \beta'\left(1 + sign\left(\mathbf{d}_{pf}^\top\left(\mathbf{C}_p(s_p)\times\mathbf{k}_p\right)\right)\left(1 - e^{\gamma_c\|\mathbf{C}_p(s_p)\|}\right)\right) \tag{18}$$

where β' is a negative gain for returning to the path, $sign(\bullet)$ is a sign function to determine the direction along the reference path, and γ_c is a negative gain for sensing the amount of path curvature.

The ratio between the gain factors (i.e., v_{tis} and β') forms an acceptable error band around the reference path. For instance, if β' is higher than v_{tis}, then the error band will be small. On the contrary, in the case where v_{tis} is bigger than β', then

the error band will be large since the priority is to advance along the reference path. The effect of this ratio is presented in section IV.

Furthermore, the control velocity of the tool-tip (15) could be represented with respect to any desired frame. Note that if the end-effector frame is selected, then the end-effector twist velocity ${}^e\underline{\mathbf{v}}_e$ is related to the linear velocity of the tool-tip ${}^e\mathbf{v}_t$ as:

$$ {}^e\mathbf{v}_t = \underbrace{\begin{bmatrix}\mathbf{I}_{3\times3} & -[{}^e\mathbf{et}]_\times\end{bmatrix}}_{\mathbf{L}_{pf}\in\mathbb{R}^{3\times6}} \underbrace{\begin{bmatrix} {}^e\mathbf{v}_e \\ {}^e\omega_e \end{bmatrix}}_{{}^e\underline{\mathbf{v}}_e} \tag{19} $$

whereby $[{}^e\mathbf{et}]_\times$ is the skew-symmetric matrix associated to the vector ${}^e\mathbf{et}$, and $\mathbf{L}_{pf}$ is the interaction matrix related to the path-following task.

Finally, the control velocity for the path-following task is deduced as:

$$ {}^e\underline{\mathbf{v}}_e = \mathbf{L}_{pf}^{\dagger}\ {}^e\mathbf{v}_t \tag{20} $$

C. Bilateral Constrained Motion Controller

As claimed above, the robot should perform the surgical task under the constraints of the incision point. This section begins with a description of RCM (bilateral constraints), which imposes that the centre-line of tool body $\mathcal{S}_t$ should be coincident with the point $\mathcal{O}_r$. Simultaneously, the tool-tip must follow the desired path inside the incision orifice.

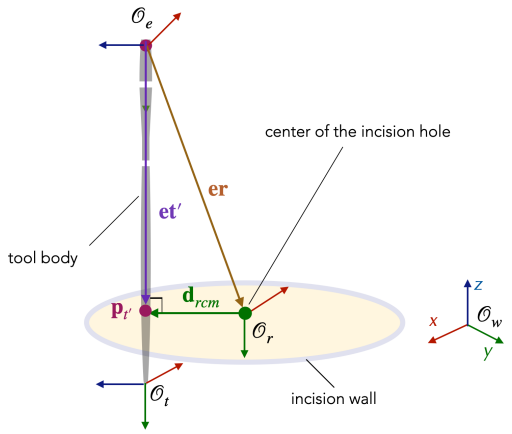

Fig. 5. Geometric scheme of the bilateral linear error $\mathbf{d}_{rcm}$.

Fig. 5 shows a straight tool which is located far from the centre-point of the incision orifice $\mathcal{O}_r$. The previous works [7], [17] built the controller based on the angular error between the vectors $\mathbf{et}'$ and $\mathbf{er}$ while the proposed controller in this section is based on the linear error $\mathbf{d}_{rcm}$. This new choice offers the controller to become independent of the tool shape. Let us imagine that the tool-tip position in Fig. 5 is fixed in space, but its length can change. In the case of angular error, when the tool length increases, the error reduces its value. However, the linear error stays constant when the tool length changes. Therefore, the new choice grants better numerical computing.

The error $\mathbf{d}_{rcm}$ is deduced by the orthogonal projection of the point $\mathcal{O}_r$ onto the tool body $\mathcal{S}_t$. The point $\mathbf{p}_{t'}$ is the resultant from the latter projection that is calculated as follows:

$$ {}^e\mathbf{p}_{t'} = {}^e\mathbf{u}_{et}\ {}^e\mathbf{u}_{et}^{\top}\ {}^e\mathbf{er} \tag{21} $$

whereby ${}^e\mathbf{u}_{et}$ is the unit vector of $\mathbf{et}$ expressed in $\mathcal{F}_e$, and ${}^e\mathbf{er}$ represents the vector between both points $\mathcal{O}_e$ and $\mathcal{O}_r$ which is expressed in $\mathcal{F}_e$.

In case the surgical tool is curved, the point $\mathbf{p}_{t'}$ is determined by discretizing the tool body. Then the closest point to the tool body is located. After that, the orthogonal projection is performed with respect to this point and the previous one on the tool centre-line. Thus, the error $\mathbf{d}_{rcm}$ is deduced as:

$$ \mathbf{d}_{rcm} = {}^e\mathcal{O}_r - {}^e\mathbf{p}_{t'} \tag{22} $$

The controller task consists of finding the spatial velocity of the robot end-effector ${}^e\underline{\mathbf{v}}_e$ for eliminating the rate-of-change of the bilateral linear error $\mathbf{d}_{rcm}$. Thereby, the time-derivative of the latter equation results in:

$$ \dot{\mathbf{d}}_{rcm} = {}^e\mathbf{v}_r - {}^e\mathbf{v}_{t'} \tag{23} $$

where ${}^e\mathbf{v}_{t'}$ is the linear velocity of the projected point $\mathbf{p}_{t'}$ along the tool body, and ${}^e\mathbf{v}_r$ is the linear velocity of the trocar point described in $\mathcal{F}_e$.

Indeed, the velocity of the projected point depends on the movement of the tool body with respect to the trocar point. Hence, this velocity is computed as [7]:

$$ {}^e\mathbf{v}_{t'} = \frac{{}^e\mathbf{k}_t\ {}^e\mathbf{k}_t^T}{1+\mathbf{d}_{rcm}^T(\mathbf{C}_t(s_t)\times\ {}^e\mathbf{k}_t)}\ {}^e\mathbf{v}_r \tag{24} $$

whereby $\mathbf{C}_t(s_t)$ is the tool curvature in the function of its arc length, and ${}^e\mathbf{k}_t$ is the instantaneous tangential unit-vector onto the tool curve/shape.

Since the calculation is done from the perspective of the end-effector frame $\mathcal{F}_e$, it implies that this frame is fixed, and the other ones are dynamic with respect to it. Consequently, the incision orifice virtually moves, and its linear velocity ${}^e\mathbf{v}_r$ is related to the spatial velocity of the robot end-effector thanks to the following formula:

$$ {}^e\mathbf{v}_r = \underbrace{\begin{bmatrix}\mathbf{I}_{3\times3} & -[{}^e\mathcal{O}_r]_\times\end{bmatrix}}_{\mathbf{L}_r\in\mathbb{R}^{3\times6}}\ {}^e\underline{\mathbf{v}}_e \tag{25} $$

By injecting the latter equation in (24) then the resultant in (23), the time-derivative of the error $\mathbf{d}_{rcm}$ is:

$$ \dot{\mathbf{d}}_{rcm} = \underbrace{\left[\mathbf{I}_3 - \frac{{}^e\mathbf{k}_t\ {}^e\mathbf{k}_t^T}{1+\mathbf{d}_{rcm}^T(\mathbf{C}_t(s_t)\times\ {}^e\mathbf{k}_t)}\right]\begin{bmatrix}\mathbf{I}_{3\times3} & -[{}^e\mathcal{O}_r]_\times\end{bmatrix}}_{\mathbf{L}_{rcm}\in\mathbb{R}^{3\times6}}\ {}^e\underline{\mathbf{v}}_e \tag{26} $$

where $\mathbf{L}_{rcm}$ is the interaction matrix which relates between the end-effector velocity ${}^e\underline{\mathbf{v}}_e$ and the rate-of-change of the error $\mathbf{d}_{rcm}$.

Furthermore, a linearized proportional controller is applied to reduce the bilateral linear error in an exponential decay form. It defines the control velocity of the end-effector as:

$$ {}^e\underline{\mathbf{v}}_e = -\lambda\ \mathbf{L}_{rcm}^{\dagger}\ \mathbf{d}_{rcm} \tag{27} $$

whereby λ is a positive gain which allows tuning the rate of exponential decay, and $\mathbf{L}_{rcm}^{\dagger}$ is the pseudo-inverse of the interaction matrix $\mathbf{L}_{rcm}$.

Finally, the RCM task can be combined as the highest priority with the path-following task as the secondary criterion.

The hierarchical controller deduces the control velocity, by replacing the equations (27) and (20) in equation (6), as:

$$ {}^{e}\underline{\mathbf{v}}_{e} = -\lambda \mathbf{L}_{rcm}^{\dagger}\mathbf{d}_{rcm} + \tilde{\mathbf{L}}_{pf}^{\dagger}\left({}^{e}\mathbf{v}_{t} + \lambda \mathbf{L}_{pf}\mathbf{L}_{rcm}^{\dagger}\mathbf{d}_{rcm}\right), \quad (28) $$

with,

$$ \tilde{\mathbf{L}}_{pf} = \mathbf{L}_{pf}\left(\mathbf{I} \; - \; \mathbf{L}_{rcm}^{\dagger}\mathbf{L}_{rcm}\right) \quad (29) $$

In the opposite case, the hierarchical controller sets the path-following task (20) as the highest priority while the RCM task (27) as the secondary one. The control velocity is deduced from equation (6) as:

$$ {}^{e}\underline{\mathbf{v}}_{e} = \mathbf{L}_{pf}^{\dagger} \; {}^{e}\mathbf{v}_{t} - \tilde{\mathbf{L}}_{rcm}^{\dagger}\left(\lambda \mathbf{d}_{rcm} + \mathbf{L}_{rcm}\mathbf{L}_{pf}^{\dagger} \; {}^{e}\mathbf{v}_{t}\right), \quad (30) $$

with

$$ \tilde{\mathbf{L}}_{rcm} = \mathbf{L}_{rcm}\left(\mathbf{I} \; - \; \mathbf{L}_{pf}^{\dagger}\mathbf{L}_{pf}\right) \quad (31) $$

D. Unilaterally Constrained Motion Controller

This section continues with the design of the path-following controller under unilateral constraints. Notice that the UCM task assumes the incision orifice is larger than the tool diameter. Consequently, it imposes on the tool-tip to follow the incision/ablation path while the tool body is free to move within the incision orifice as long as it does not damage the orifice wall. Therefore, the formulation of the previous section needs to extend to satisfy the unilateral constraints.

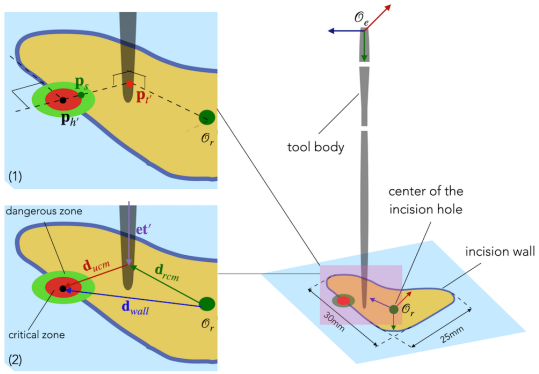

Fig. 6. Geometric modelling of the unilateral linear error $\mathbf{d}_{ucm}$.

Fig. 6(1) shows how the point $\mathbf{p}_{t'}$ is orthogonally projected onto the orifice wall in order to determine the closest point $\mathbf{p}_{h'}$ on the orifice wall $\mathcal{S}_h$. The distance between the latter two points forms the vector error $\mathbf{d}_{ucm}$ which can be defined as (left image (2) of Fig. 6)

$$ \mathbf{d}_{ucm} = \underbrace{{}^{e}\mathbf{t'r}}_{=\mathbf{d}_{rcm}} - \underbrace{{}^{e}\mathbf{h'r}}_{=\mathbf{d}_{wall}} \quad (32) $$

The objective now is how to maintain the value of the error $\mathbf{d}_{ucm}$ greater or equal to zero. For security issues, three regions are defined around the projected point $\mathbf{p}_{h'}$, as shown in the left image of Fig. 6:

- *critical zone* (dark red circle) which its border is defined by a minimal distance d_{min};
- *dangerous zone* (light green circle) which its border is defined by a maximal distance d_{max}; and
- *safe zone* which is the remaining region outside the danger zone.

When the Euclidean norm $\|\mathbf{d}_{ucm}\|$ is larger than the "dangerous" distance d_{max}, the tool can follow the reference path without any constraints since its location is in the safe zone. However, an admittance control is activated, which is composed of a virtual damper μ_{obs}, when the tool body passes the dangerous zone border. Indeed, the admittance control imposes unilateral constraint towards the safe point $\mathbf{p}_s$ by generating a compensation velocity in the opposite direction to the orifice wall.

By differentiating (32) with respect to time for deducing the velocity twist of the end-effector, it becomes equal to:

$$ \begin{aligned} \dot{\mathbf{d}}_{ucm} &= \underbrace{\left({}^{e}\mathbf{v}_{r} - {}^{e}\mathbf{v}_{t'}\right)}_{\dot{\mathbf{d}}_{rcm}} - \underbrace{\left({}^{e}\mathbf{v}_{r} - {}^{e}\mathbf{v}_{h'}\right)}_{\dot{\mathbf{d}}_{wall}} \\ &= {}^{e}\mathbf{v}_{h'} - {}^{e}\mathbf{v}_{t'} \end{aligned} \quad (33) $$

The velocity of the projected point $\mathbf{p}_{h'}$ is deduced in the same way as equation (24):

$$ {}^{e}\mathbf{v}_{h'} = \frac{{}^{e}\mathbf{k}_{h} \; {}^{e}\mathbf{k}_{h}^{T}}{1 + \mathbf{d}_{ucm}^{T}\left(\mathbf{C}_{h}(s_h) \times {}^{e}\mathbf{k}_{h}\right)} \; {}^{e}\mathbf{v}_{t'} \quad (34) $$

where $\mathbf{C}_h(s_h)$ is the orifice curvature in the function of its arc length, and ${}^{e}\mathbf{k}_h$ is the instantaneous tangential unit-vector onto the orifice curve. In another perspective, the latter equation describes how the projection of the point $\mathbf{p}_{t'}$ onto the geometric curve of the orifice wall $\mathcal{S}_h$ evolves with time.

The velocity ${}^{e}\mathbf{v}_{t'}$ is deduced by combining equations (24) and (25):

$$ {}^{e}\mathbf{v}_{t'} = \underbrace{\frac{{}^{e}\mathbf{k}_{t} \; {}^{e}\mathbf{k}_{t}^{T}}{1 + \mathbf{d}_{rcm}^{T}\left(\mathbf{C}_{t}(s_t) \times {}^{e}\mathbf{k}_{t}\right)}\left[\mathbf{I}_{3\times 3} \quad -\left[{}^{e}\mathcal{O}_{r}\right]_{\times}\right]}_{\mathbf{L}_{v_{t'}} \in \mathbb{R}^{3\times 6}} \; {}^{e}\underline{\mathbf{v}}_{e} \quad (35) $$

Replacing (34) and (35) in (33) yields:

$$ \dot{\mathbf{d}}_{ucm} = \underbrace{\left(\frac{{}^{e}\mathbf{k}_{h} \; {}^{e}\mathbf{k}_{h}^{T}}{1 + \mathbf{d}_{ucm}^{T}\left(\mathbf{C}_{h}(s_h) \times {}^{e}\mathbf{k}_{h}\right)} - \mathbf{I}_{3\times 3}\right)\mathbf{L}_{v_{t'}}}_{\mathbf{L}_{ucm} \in \mathbb{R}^{3\times 6}} \; {}^{e}\underline{\mathbf{v}}_{e} \quad (36) $$

whereas $\mathbf{L}_{ucm}$ is the interaction matrix that relates the twist end-effector with the rate of change of the error $\mathbf{d}_{ucm}$.

Thereby, the control velocity of the UCM task is defined as:

$$ {}^{e}\underline{\mathbf{v}}_{e} = -\mu_{obs}\lambda \mathbf{L}_{ucm}^{\dagger}\mathbf{d}_{ucm} \quad (37) $$

The damping coefficient μ_{obs} changes following a sigmoid function that depends on the vector $\mathbf{d}_{ucm}$. It means that the gain μ_{obs} reaches its minimal value when $\mathbf{d}_{ucm}$ is higher than the safe distance d_{max}, where the tool location is in the dangerous zone. However, μ_{obs} gradually increases until it reaches its maximal value when $\mathbf{d}_{ucm}$ is smaller than the critical distance d_{min}, where the tool location in the critical zone. This behaviour is modelled as:

$$ \mu_{obs} = \frac{\sigma_{max}}{1 + e^{\left(\sigma_{step}\left(\|\mathbf{d}_{ucm}\| - \sigma_{min}\right)\right)}} \quad (38) $$

where σ_{max}, σ_{min} and σ_{step} are tunable parameters for modifying the sigmoid form.

Finally, the path-following task can be combined as the highest priority with the UCM task as the secondary criterion. The hierarchical controller deduces the control velocity, by replacing the equations (37) and (20) in equation (6), as:

$$ {}^{e}\underline{\mathbf{v}}_{e} = \mathbf{L}_{pf}^{\dagger}\ {}^{e}\mathbf{v}_{t} - \tilde{\mathbf{L}}_{ucm}^{\dagger}\left(\mu_{obs}\lambda\mathbf{d}_{ucm} + \mathbf{L}_{ucm}\mathbf{L}_{pf}^{\dagger}\ {}^{e}\mathbf{v}_{t}\right), \quad (39) $$

with

$$ \tilde{\mathbf{L}}_{ucm} = \mathbf{L}_{ucm}\left(\mathbf{I} - \mathbf{L}_{pf}^{\dagger}\mathbf{L}_{pf}\right) \quad (40) $$

IV. VALIDATION

This section discusses the qualitative and quantitative evaluation of the proposed methods and materials. The developed controllers were first tested using our simulator framework and then in an experimental set-up that takes up the various components of the simulator.

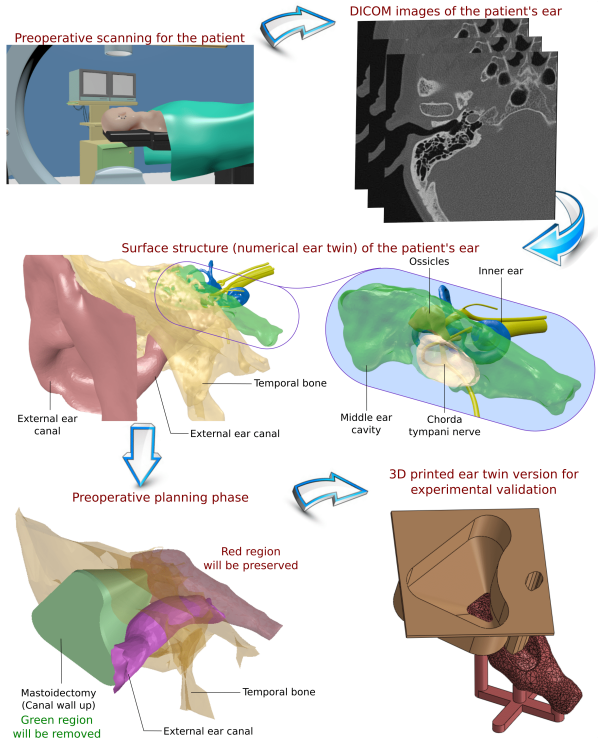

Fig. 7. The performed steps to achieve a digital and physical model of the middle ear cavity.

A. Implementation Details

This part begins by converting the patient's ear to its digital twin and then its 3D-printed twin. The first step to accomplish this work is the scan of the patient's ear during the preoperative phase for getting DICOM (Digital Imaging and Communications in Medicine) images, as depicted in Fig. 7. The DICOM images are handled by the software *3D Slicer* which converts these images to a 3D surface model after a segmentation process. The 3D Slicer software exports the segmentation results as *STL* files for each anatomical structure. Afterwards, the software *MeshLab* treats the STL files for smoothing the surface and reducing the number of vertices and faces to cut down the final STL file size. This step produces the digital twin of the patient's ear. The numerical model is then printed in 3D to be used during the experimental validation. The planning of the incision (reference) path is done manually, as well as the orifice wall, using Solidworks.

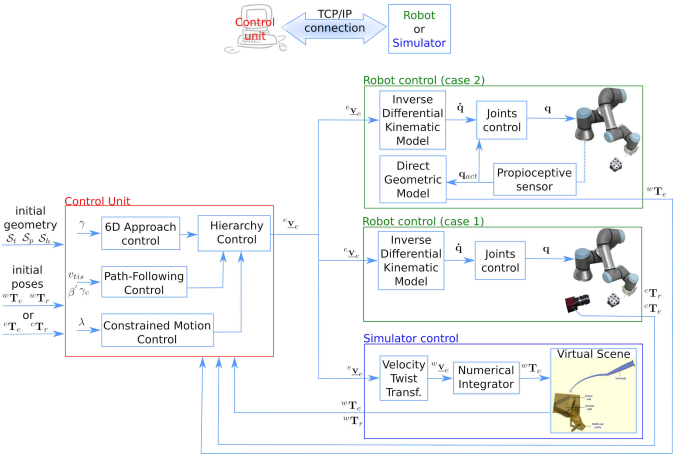

Fig. 8. Block diagram of the TCP/IP communication between the client (proposed controller) and the server (simulator or robot) or vice-versa.

We wanted to implement the proposed controller in the cartesian/operation space to test it quickly on different robotic structures (i.e., open or closed loop) which has a 6-DoF. Therefore, the null space projection was selected to find the solution. Using constrained optimisation in the joint space would certainly be more useful in the case of a (hyper-)redundant robot [34], but this is out of the scope of this article, which focuses on the tool task.

Figure 8 presents the proposed control architecture with a TCP/IP communication. This block diagram allows easy interchangeability between the real system (robot) and its digital twin (simulator). The latter figure (the red block at the left-hand side) also shows that the implemented controller is firstly initialized with the end-effector and the incision orifice poses, ${}^{\star}\mathbf{T}_{e}$ and ${}^{\star}\mathbf{T}_{r}$ respectively. These poses must be described in the same frame (e.g., the world frame $\mathcal{F}_{w}$ or the camera $\mathcal{F}_{c}$). Indeed, the tool geometry $\mathcal{S}_{t}$ is defined with respect to the end-effector frame $\mathcal{F}_{e}$ while the reference path $\mathcal{S}_{p}$ and the orifice wall $\mathcal{S}_{h}$ are described in the incision orifice frame $\mathcal{F}_{r}$. Furthermore, the controllers should be initialized by the different gain coefficients before the control loop starts.

The hierarchy controller arranges throughout the control loop the priority between the different tasks (i.e., the approach task, the path-following task, and the RCM/UCM constraints). Indeed, the control loop is mainly divided into three phases:

- the *outside phase*: the tool corrects its initial pose with respect to the incision orifice. This stage applies the approach task for regulating: i) the tool-tip position to the point located before the orifice centre point, and ii) the tool-tip rotation as the rotation of the orifice reference frame. This task is performed to ensure some security for the next phase;
- the *transition phase*: the tool-tip passes the centre point of the incision orifice. The RCM controller could oscillate

when the trocar point is close to the tool-tip. These oscillations are generated because the controller computes large angular motion due to the lever phenomena for compensating the angular error. Thus, the trocar point is virtually moved to the first point on the reference path. Consequently, the tool body can rotate about this new point. This virtual trocar point moves towards the orifice frame while the tool-tip advances along the reference path;
- the *inside phase*: the tool-tip follows the desired path while the tool body is constrained by the orifice wall or the orifice centre point.

Therefore, the output of this block is the spatial velocity of the end-effector expressed in its frame (${}^{e}\underline{\mathbf{v}}_{e}$) while its inputs are the instantaneous poses of the end-effector and the incision orifice (${}^{\star}\mathbf{T}_{e}$ and ${}^{\star}\mathbf{T}_{r}$). The question now is: what is the observation frame?

In the simulator case (the blue block at the right-hand side of Fig. 8), it is straightforward since the user initializes the poses with respect to the world frame $\mathcal{F}_w$ of the virtual scene. Thus, the spatial velocity ${}^{e}\underline{\mathbf{v}}_{e}$ is transformed to ${}^{w}\underline{\mathbf{v}}_{e}$ then it is integrated over the sample time T_e to deduce the new pose of the end-effector. Consequently, the tool pose is updated in the virtual scene, and this new pose is sent back to the control unit block for computing a new iteration. There are two options for designing the control architect in the experimental case. The first one consists of using an exteroceptive sensor (e.g., a camera) for estimating the required poses. This option is depicted in the green block of Fig. 8 named *Robot control (case 1)*. The input of this block is the spatial velocity ${}^{e}\underline{\mathbf{v}}_{e}$ that is transformed to deduce the angular velocity of each joint $\dot{\mathbf{q}}$ with the help of the inverse differential kinematic model to move mechanical structure of the robot. This motion is observed from the camera frame $\mathcal{F}_c$ in order to estimate the new pose of the end-effector and that of the orifice. These poses are the output of this block which are sent back to the control unit block for calculating a new iteration. However, this option is uneasy for implementation since it needs a particular setup to accurately track both the end-effector and the orifice. The second option is more fundamental than the first one. It is also presented in the green block of Fig. 8 named *Robot control (case 2)*. It uses the proprioceptive sensors of the robot and its forward geometric model to estimate the end-effector pose. Despite that, this option requires performing a registration process between the robot and the orifice before the control loop. After that, the robot works blindly, and the user assumes that the orifice does not move during the control loop.

The simulator is implemented in C++. It uses *Eigen* library for linear algebra and *PCL* (Point Cloud Library) for visualizing the STL parts and converting them to point clouds. This conversion is done to initialize the collision detection that is accomplished by *VCollide* library. Finally, *ViSP* library is used for manipulating the camera images throughout the experimental work.

B. Numerical Validation

A numerical simulator was developed, as the first step, to validate the functioning of the diverse methods before physical implementation. It simulates the geometric motion of the surgical tool through the incision orifice and the middle ear cavity. The software interchangeability of the simulator and the physical set-up allowed us also to tune the controller parameters before the experimental validation. To do this, three scenarios for the demonstration are carried out:

- *scenario 1* performs the path-following task without any constraint applied to the tool motion. It demonstrates the effect of the gain coefficients v_{tis} and β in (16) and (18), respectively, on the performance of the path-following controller;
- *scenario 2* performs the path-following task with RCM constraints. It mimics the drilling of a minimal invasive tunnel (i.e., conical tunnel) through the mastoid portion to reach the middle ear cavity;
- *scenario 3* assumes the surgeon performed a standard mastoidectomy. It mimics an inspection/resection task performed under the UCM constraints.

1) Simulation of the path-following task without constraints: Throughout this first trial, the value of $v_{tis} = 4\ mm/s$ in (16) remains constant during all tests. Besides that, the same reference path is tested during this trial, and it is defined as a spiral curve.

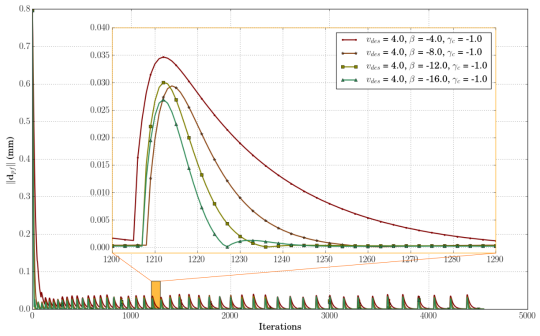

Fig. 9. The effect of the ratio between v_{des} and β' on the path-following error $\mathbf{d}_{pf}$ with a zoom and magnification on the orange region.

The first group of tests keeps the value of γ_c in (18) constant while decreasing the value of β' which its value varies from -4 to -16. Figure 9 shows the influence of the gain coefficient β' on the path-following error $\mathbf{d}_{pf}$. Indeed, this error is computed as in (13). The ripples appearing in this figure represent the linear error between the projected point $\mathbf{p}_{t'}$ and the closest point on the reference path $\mathbf{p}_{p'}$. An orange rectangle appeared in this figure for zooming on one of these ripples. One can observe that the error reduced as designed exponentially. It can be also highlighted that the best ratio between β' and v_{tis} should be greater than -2 (the saddle-brown line with star markers), and less than or equal -3 (the olive line with square markers). If the ratio is less than or equal to -1, the controller response is relatively slow, and there is a steady-state error (the maroon line with round markers in Fig. 9). On the opposite, if the ratio is higher than

or equal to -4, the system begins to oscillate (having over-shoots). However, the controller reduces the error faster than the previous cases (the sea-green line with triangular markers in Fig. 9).

The second group of tests chose a constant ratio -2 while decreasing the value of γ_c from -2 to -16. This group shows that the best value of γ_c is to be near to β'. If γ_c is higher than β', the system begins to have over-shoots, but it reduces faster the path-following error.

Finally, the objective of this trial is not only the choice of the value of a certain coefficient which allows having good behaviour (speed and stability) of the controller but rather the ratio between these coefficients. In other words, reach a compromise between the speed of the path execution and the degree of oscillations that can be tolerated. For instance, in the context of middle ear surgery, during the approaching phase where the instrument moves from an initial position outside the ear cavity to the inside, since it is safe for the patient, can be allowed limited oscillations and then performed rapidly. On the other hand, the resection task can be performed less rapidly but with precision and safety (without oscillations).

2) Simulation of a robotic drilling task under RCM constraint: The surgeon perforates manually until now the mastoid portion in the temporal bone for reaching the middle ear cavity. The resultant mastoidectomy orifice is invasive. Thereby, a less invasive tunnel is proposed in this trial. Besides that, the drilling procedure becomes automated so that the surgeon can concentrate on other essential tasks. Indeed, this drilling procedure is achieved by merging the approach task, the 3D path-following task, and the RCM task.

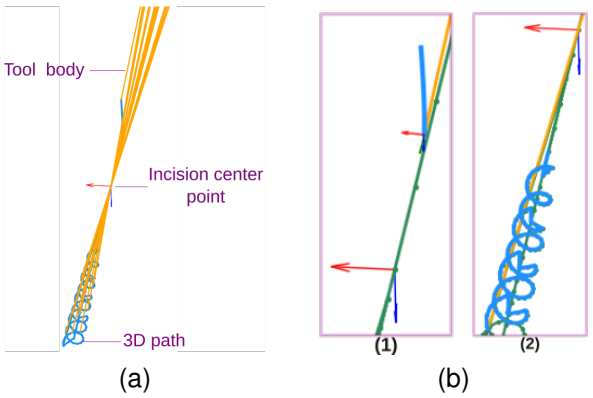

Fig. 10. Numerical validation of the 3D path-following under an RCM constraint (see Multimedia Extension). (a) The tool position with respect to the desired path. (b) The sequence of zoom images during the tool motion.

Figure 10 depicts the tool motion throughout the drilling procedure, where Fig. 10(a) draws the tool geometry and its poses at different instances (orange straight lines). It also shows the drilling path defined as a combination of spiral and linear portions (sea-green dotted line). One can view that the tool body is always coincident with the orifice's centre point. Subplot (b1) shows the path done by the tool-tip (dodger-blue line) to accomplish the outside phase by i) approaching the point located before the orifice centre point, and ii) regulating the rotation of the tool-tip frame to as that of the orifice reference frame. Figure 10(b) depicts an instantaneous zoom on the tool pose during the inside phase to show the RCM effect.

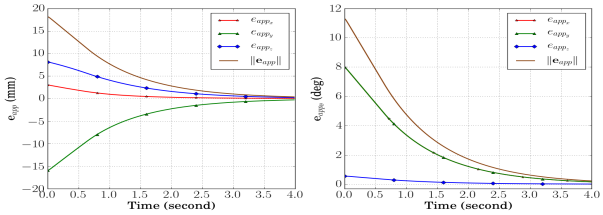

Fig. 11. The approach task error $\mathbf{e}_{app}$, where the left column is the linear error and the right column represents the angular error.

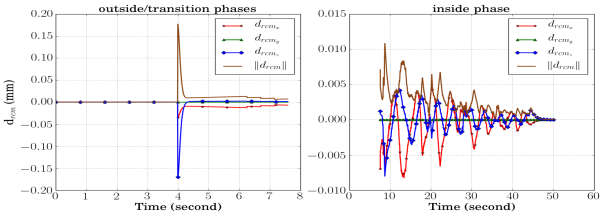

Fig. 12. The RCM task error $\mathbf{d}_{rcm}$, where the left column shows the error evolution during the transition phase while the right column presents the error during the inside phase.

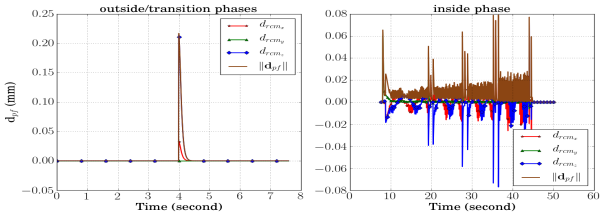

Fig. 13. The path-following task error $\mathbf{d}_{pf}$, where the left column shows the error evolution during the transition phase while the right column presents the error during the inside phase.

The approach task error $\mathbf{e}_{app}$ computed in (11) is shown in Fig. 11 which depicts the linear errors in the left and the angular errors in the right. It can be underlined that both the linear and angular errors decrease exponentially as expected. At the end of the approach task, the transition phase starts. The task-hierarchical controller becomes active, and it arranges the path-following task as the highest priority while the RCM task is the second one. The errors of these tasks presented on the left of Fig. 13 and Fig. 12 which are obtained from (13) and (22) for the path-following and RCM errors, respectively. One can observe a peak appearing around 4 $seconds$, it is due to the initial error when the controller becomes activated. Then, it attenuates the error until it attains stability.

After the previous period, the inside phase starts where the hierarchical controller modifies the priority by setting the RCM task as the highest one while the path-following is the secondary one. The RCM task error $\mathbf{d}_{rcm}$ was computed as 0.002 ± 0.002 mm (mean error $\pm$ STD (STandard Deviation) error), as shown in the right of Fig. 12, while the path-following error $\mathbf{d}_{pf}$ was 0.008 ± 0.009 mm, as shown in the

right of Fig. 13. Note that the gain values used for this trail were equal to $\lambda = 1$, $\gamma = 1$, $v_{tis} = 4\ mm/s$, $\beta' = -10$, $\gamma_c = -10$ and $T_e = 0.008\ s$.

3) Simulation of an ablation/excision surgical task under UCM constraint: In this trial, the incision orifice size is larger than the instrument diameter. The tool (a curved one) is consequently subject to the UCM for providing more freedom to the tool movements inside the incision orifice. This behaviour is shown in Fig. 14(a) where the orifice wall is represented by the red surface. The desired 3D path is thus composed of a linear portion to reach the middle ear cavity and a spiral curve to simulate the required surgical task. This selected path can reach some regions where a straight tool cannot attain (see Multimedia Extension).

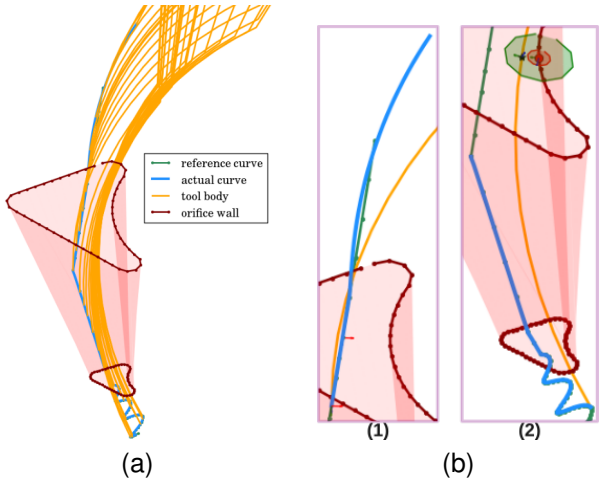

Fig. 14. Numerical validation of the 3D path-following under a UCM constraint (see Multimedia Extension). (a) The tool position with respect to the desired path. (b) The sequence of zoom images during the tool motion.

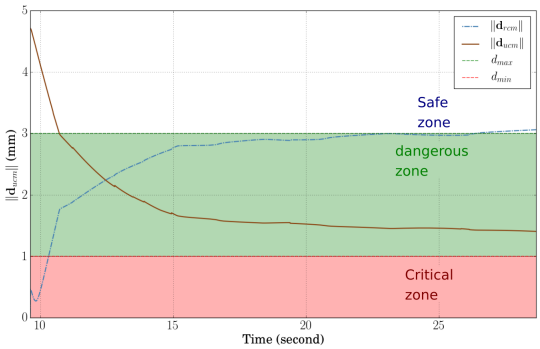

Fig. 15. The UCM task error $\mathbf{d}_{ucm}$ during the inside phase alongside the error $\mathbf{d}_{rcm}$.

Figure 14(b)(1) indicates the path done by the tool during the outside phase. It also presents an instantaneous pose of the tool body throughout the transition phase. As explained in the previous trial, the proposed controller executes the same tasks over these two phases. Figure 14(b)(2) presents the tool motion during the inside phase, where the dangerous and critical zones are represented by the green and red circles, respectively. The centre point of these circles corresponds to the point $\mathbf{p}_{h'}$ obtained by projecting $\mathbf{p}_{t'}$ onto the orifice wall $\mathcal{S}_h$.

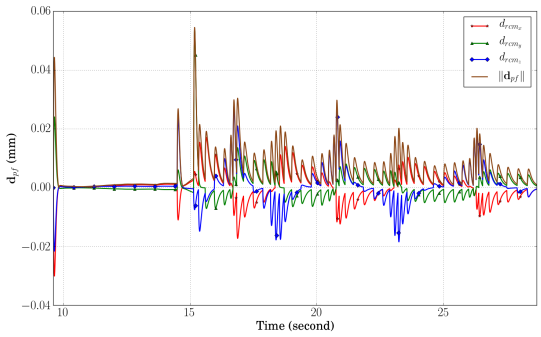

Fig. 16. The path-following task error $\mathbf{d}_{pf}$ during the inside phase.

Throughout the inside phase, the hierarchical controller combines the UCM task with the path-following task as described in (39). Figure 15 shows the UCM task error $\mathbf{d}_{ucm}$ which is deduced as in (32). It also presents the boundaries of the critical and dangerous zones. One can observe that the error $\mathbf{d}_{ucm}$ begins with a considerable value, compared to the error $\mathbf{d}_{rcm}$, since the previous phase delivers the tool to the centre point of the incision orifice. Then, the error $\mathbf{d}_{ucm}$ reduced, while the error $\mathbf{d}_{rcm}$ increased because the tool approached the incision wall to follow the reference path. However, the error $\mathbf{d}_{ucm}$ did not exceed the d_{min}, which implies the tool body did not enter the critical zone.

Figure 16 presents a path-following error $\mathbf{d}_{pf}$ during the inside phase of $0.005 \pm 0.006\ mm$ with the following gain values: $\lambda = 0.8$, $\gamma = 0.8$, $v_{tis} = 4\ mm/s$, $\beta' = -10$, $\gamma_c = -10$ and $T_e = 0.008\ s$.

C. Experimental Validation

This part is devoted to the physical implementation of the blocks *Robot control* that is shown in Fig. 8. Its physical correspondence is presented in Fig. 17. The robotic work cell in the latter figure consists of:

- a 6 DoFs serial robot from *Universal Robot* (UR3) with $\pm 0.03\ mm$ pose repeatability. It communicates with the proposed controller via TCP/IP for receiving the command velocity of the end-effector. It also sends the end-effector pose to the controller if the block *Robot control (case 1)* is required to be executed;
- a monocular camera from *Guppy* (with an image resolution of $640 \times 420\ pixels$) and an optical objective lens from *Computar* with distortion (model MLM3X-MP) are used for the control purpose;
- two visualization cameras provide other views for recording multimedia videos.

The digital twin of the ear model shown previously in Fig. 7 is modified for implementing its 3D printed twin shown in Fig. 17(b).

The trials of this part have the objective to evaluate the performance of the path-following controller under constraints. Therefore, a curved tool follows the same planned path, one time under the RCM constraint and the second time under the UCM constraint.

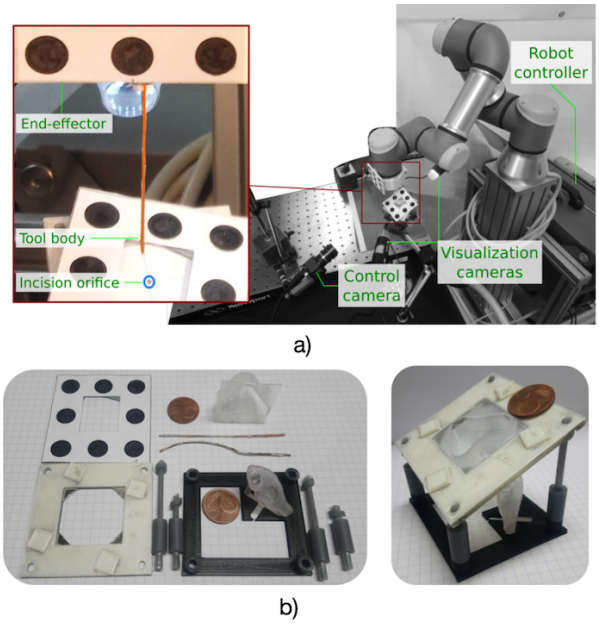

Fig. 17. Overview of the experimental setup. (a) the robotic system and (b) the printed model of the middle ear cavity.

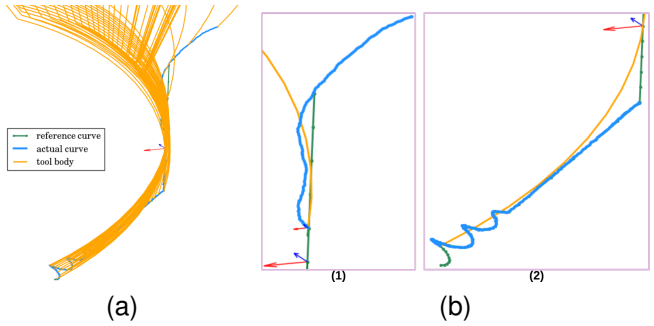

Fig. 18. Experimental validation of the 3D path-following under an RCM constraint (see Multimedia Extension). (a) The tool position with respect to the desired path. (b) The sequence of zoom images during the tool motion.

1) Path-following under RCM constraint: Figure 18 presents the desired path (sea-green dotted line), the resultant motion of the curved tool (orange line), and the path done by the tool-tip (dodger-blue line). One can observe in Fig. 18(b)(1) that the tool approaches the incision orifice by executing the controller given in (11). The approach task error $\mathbf{e}_{app}$ computed from (7). Figure 19 presents the latter error and it converges toward zero by the end of this phase.

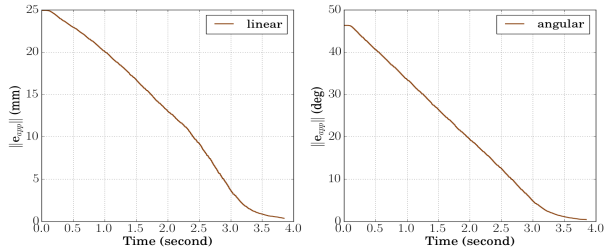

Fig. 19. The approach task error $\mathbf{e}_{app}$, where the left column is the linear error and the right column represents the angular error.

Afterwards, the transition phase starts so that the tool passes the centre point of the incision orifice, as explained previously. The hierarchical controller (30) arranges the path-following task as the highest priority while the RCM task is the second one. This behaviour is demonstrated in the left column of Fig. 20-21, where the hierarchical controller has been activated around $4\ s$. One can visualize that the RCM task error $\mathbf{d}_{rcm}$ has some steps due to the movements of the virtual trocar point while the path-following error $\mathbf{d}_{pf}$ maintained its value around zero.

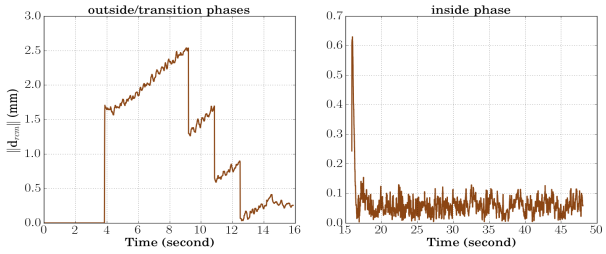

Fig. 20. The RCM task error $\mathbf{d}_{rcm}$, where the left column shows the error evolution during the outside/transition phases while the right column presents the error during the inside phase.

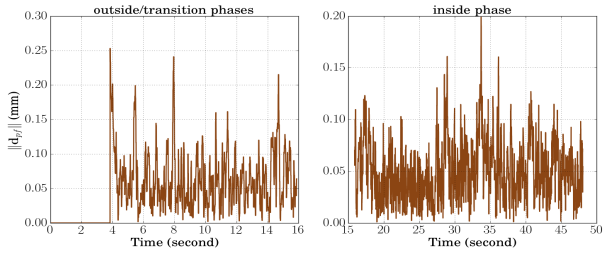

Fig. 21. The path-following task error $\mathbf{d}_{pf}$, where the left column shows the error evolution during the outside/transition phases while the right column presents the error during the inside phase.

When the tool passes the centre point of the incision orifice, the inside phase begins. The hierarchical controller (28) modifies its priorities by setting the RCM task as the highest one and the path-following as the second one. The system performances during the inside phase are shown in the right columns of Fig. 20-21. During this phase, the RCM task error $\mathbf{d}_{rcm}$ measured as $0.06 \pm 0.05\ mm$ (mean error $\pm$ standard deviation (STD) error) while the path-following error $\mathbf{d}_{pf}$ was $0.05 \pm 0.03\ mm$.

An exteroceptive sensor is used to close the control loop, as presented in Fig. 8 by the block *Robot control (case 2)*. Besides that, the gain values used in this experiment were equal to $\lambda = 1$, $\gamma = 1$, $v_{tis} = 0.5\ mm/s$, $\beta' = -1.25$, $\gamma_c = -10$ and $T_e = 0.008\ s$. Another trial was conducted for testing the block *Robot control (case 1)* by using the proprioceptive sensor in the control loop. The system performances are better than the exteroceptive test (see test 2 in Table I). The errors $\mathbf{d}_{rcm}$ and $\mathbf{d}_{pf}$ are reduced to almost half. It implies that our vision system needed amelioration in terms of accuracy.

From the surgeon's perspective, it is required to target the residual cells of cholesteatoma. It implies that the robot should detect/remove a human cell whose size is around $0.1\ mm$.

The proposed controller reached the requirements since the error $\mathbf{d}_{pf}$ is smaller than the human cell size. Besides that, the surgical tool does not damage the entry orifice (patient's head).

By increasing the tool velocity $v_{tis} = 2\ mm/s$ and maintaining the same ratio $\beta'/v_{tis} = -2$, the system performances deteriorated as expected. The errors $\mathbf{d}_{rcm}$ and $\mathbf{d}_{pf}$ are almost increased by half (see tests 2 and 4 in Table I). Therefore, the choice of the gain coefficients affects the system's performance.

2) Path-following under UCM constraint: This second trial assumes the same conditions as the previous one. It involves the same curved tool and the desired path. However, this trial imposed a unilateral constraint on the tool motion. Consequently, the tool can leave the centre point of the incision orifice and move near the orifice wall. Figure 22(b)(1) of the latter figure shows the path done by the tool-tip during the outside/transition phases, while Fig. 22(b)(2) presents the tool-tip path during the inside phase. The dangerous and critical regions are presented by the green and red circles in the latter sub-figure.

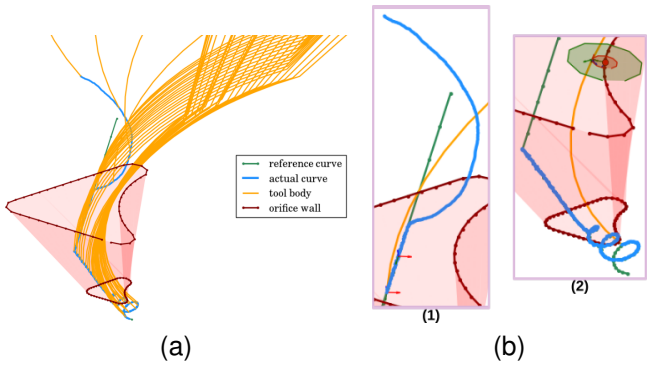

Fig. 22. Experimental validation of the 3D path-following under a UCM constraint (see Multimedia Extension). (a) The tool motion during the different phases. (b) The sequence of zoom images during the tool motion.

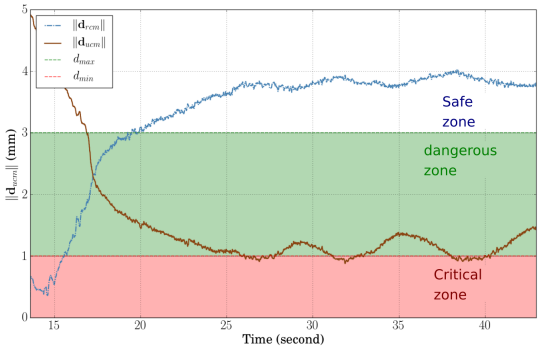

Fig. 23. The UCM task error $\mathbf{d}_{ucm}$ during the inside phase.

Throughout the inside phase, the hierarchical controller arranges the different tasks as explained in Section III-D. The highest priority is the path-following task when the tool is located in the safe zone. However, the highest priority changes to the UCM task when the tool body passes the danger zone. The system performances are shown in Fig. 23 and Fig. 24. One can observe from the UCM task error $\mathbf{d}_{ucm}$ (Fig. 23) that the tool body is maintained in the dangerous zone since the error $\mathbf{d}_{ucm}$ changes its value between d_{max} and d_{min}. However, there are some oscillations occurred compared to the simulation case in Fig. 15. These oscillations may appear due to the displacement of the ear model as shown in the Multimedia Extension.

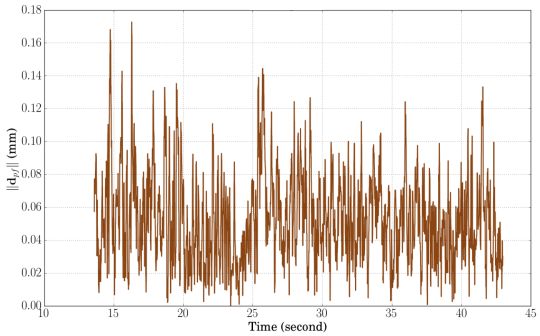

Fig. 24. The path-following task error $\mathbf{d}_{pf}$ during the inside phase.

Besides that, the path-following error $\mathbf{d}_{pf}$ (Fig. 24) was $0.05 \pm 0.03\ mm$ (mean error $\pm$ STD error) and its median error was $0.05\ mm$. An exteroceptive sensor is used as the feedback sensor and the gain values used for this second trial are $\lambda = 1$, $\gamma = 1$, $v_{tis} = 0.5\ mm/s$, $\beta' = -1.25$, $\gamma_c = -10$ and $T_e = 0.008\ s$.

The error $\mathbf{d}_{pf}$ of this trial remains almost the same as the previous trial. It implies that the UCM constraint does not affect the path-following error. It is essential in practice to ensure that the surgical tool follows the desired path independently from the applied constraints. Besides that, this trail shows that the tool movement increased along the x and z axes until the orifice wall.

In practice, the surgeon performs a mastoidectomy procedure to reach the middle ear cavity. Therefore, this article presented a UCM controller for guiding a rigid tool through this orifice type. Besides that, this article showed a less invasive procedure achieved by imposing an RCM on the same rigid instrument. However, the latter method decreases the reachable region within the middle ear cavity. Therefore, it will be required to determine this reachable region and explore the option of a flexible tool under actuation.

V. CONCLUSION AND FUTURE WORK

This article discussed the design of an original controller for guiding a rigid instrument (straight or curved) under constrained motions such as RCM or UCM. The proposed methods gave a generic formulation, in the same controller, of two tasks: i) the constrained motion (RCM or UCM), and ii) a revisited 3D path-following scheme by increasing the sensitivity to the path complexity. To manage the achievement of two or more tasks without conflicts, we also implemented a task-prioritizing paradigm. Consequently, the developed control scheme can be integrated easily with various robotic systems without an accurate knowledge of the robot's inverse kinematics.

TABLE I
SUMMARY OF DIFFERENT TRIALS ACHIEVED WITH THE CURVED TOOL DURING THE EXPERIMENTAL TESTS. RESULTS OBTAINED WITH THE FOLLOWING PARAMETERS: $\lambda = 1$, $v_{tis} = 0,5\ mm/s$, AND $T_e = 0,008\ s$. THE MEAN ERROR $\mu(\|\mathbf{e}\|)$ IS GIVEN IN mm.

N°	constraint	feedback	type of error	mean ($\|\mathbf{e}\|$) $\pm$ STD	value of β'
1	RCM	exteroceptive	d_{rcm} d_{pf}	0.06±0.05 0.05±0.02	-1.25
2	RCM	exteroceptive	d_{rcm} d_{pf}	0.15±0.06 0.08±0.05	-5
3	RCM	proprioceptive	d_{rcm} d_{pf}	0.02±0.05 0.02±0.01	-1.25
4	RCM	proprioceptive	d_{rcm} d_{pf}	0.03±0.08 0.03±0.02	-5
5	UCM	exteroceptive	d_{rcm} d_{pf}	3.30±0.93 0.05±0.03	-1.25
6	UCM	exteroceptive	d_{rcm} d_{pf}	3.30±0.93 0.09±0.06	-5
7	UCM	proprioceptive	d_{rcm} d_{pf}	2.74±0.77 0.02±0.01	-1.25
8	UCM	proprioceptive	d_{rcm} d_{pf}	2.69±0.67 0.03±0.02	-5

Experimental validation was also successfully conducted using a 6-DoF robotic system. The obtained results are promising in terms of behaviour and precision. These performances, even if they meet the specifications of the targeted middle ear surgery, can be improved or adapted for others surgeries that require more precision. The pose estimation of the tool-tip was performed using a geometric model of the instrument which can be a potential source of error. It would be interesting to investigate another method for estimating the tool's shape and pose, for instance using a depth camera.

The forthcoming work will implement the discussed methods in a clinical context using a realistic phantom and a human cadaver. Besides that, a force control could be added to increase the robot's sensitivity to its environment and increase the level of security.

REFERENCES

[1] L. Zorn, F. Nageotte, Zanne, and al., "A novel telemanipulated robotic assistant for surgical endoscopy: Preclinical application to ESD," *IEEE T. on Biomed. Eng.*, vol. 65, pp. 797–808, 2018.

[2] B. Dahroug, B. Tamadazte, L. Tavernier, and al., "Review on otological robotic systems: Toward micro-robot assisted cholesteatoma surgery," *IEEE R. in Biomed. Eng.*, vol. 11, pp. 125–142, 2018.

[3] J. Troccaz, G. Dagnino, and G.-Z. Yang, "Frontiers of medical robotics: From concept to systems to clinical translation," *Ann. R. of Biomed. Eng.*, vol. 21, pp. 193–218, 2019.

[4] H. Lee, B. Cheon, M. Hwang, *et al.*, "A master manipulator with a remote-center-of-motion kinematic structure for a minimally invasive robotic surgical system," *The Int. J. of Med. Rob. and Comp. Ass. Surg.*, vol. 14, p. e1865, 2018.

[5] S. A. Bowyer, B. L. Davies, *et al.*, "Active constraints/virtual fixtures: A survey," *IEEE T. on Rob.*, vol. 30, pp. 138–157, 2014.

[6] S. Aksungur, "Remote center of motion (rcm) mechanisms for surgical operations," *Int. J. of App. Math., Elec. and Comp.*, pp. 119–126, 2015.

[7] B. Dahroug, B. Tamadazte, and N. Andreff, "Unilaterally constrained motion of a curved surgical tool," *Robotica*, pp. 1940 – 1962, 2020.

[8] C.-H. Kuo and J. S. Dai, "Robotics for minimally invasive surgery: a historical review from the perspective of kinematics," in *Int. Symp.on Hist. of Mach. and Mech.*, pp. 337–354, 2009.

[9] E. M. Boctor, R. J. Webster III, H. Mathieu, *et al.*, "Virtual remote center of motion control for needle placement robots," *Comp. Aid. Sur.*, vol. 9, pp. 175–183, 2004.

[10] T. Osa, C. Staub, and A. Knoll, "Framework of automatic robot surgery system using visual servoing," in *IEEE/RSJ Int. Conf. on Intel. Rob. and Sys.*, pp. 1837–1842, 2010.

[11] O. Kanoun, F. Lamiraux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE T. on Rob.*, vol. 27, pp. 785–792, 2011.

[12] J. J. Abbott, P. Marayong, and A. M. Okamura, "Haptic virtual fixtures for robot-assisted manipulation," in *Rob. Res.*, pp. 49–64, 2007.

[13] B. C. Becker, R. A. MacLachlan, L. A. Lobes, *et al.* , "Vision-based control of a handheld surgical micromanipulator with virtual fixtures," *IEEE T. on Rob.*, vol. 29, pp. 674–683, 2013.

[14] M. Selvaggio, G. A. Fontanelli, F. Ficuciello, *et al.*, "Passive virtual fixtures adaptation in minimally invasive robotic surgery," *IEEE Rob. and Auto. Let.*, vol. 3, pp. 3129–3136, 2018.

[15] So, J. H., Tamadazte, B., Marturi, N., *et al.*, "Dual-scale robotic solution for middle ear surgery". *Int. Conf. on Rob. and Auto.*, pp. 1784-1790, 2022.

[16] Q. Zheng, Y. He, X. Qi, P. Zhang, *et al.*, "Safety tracking motion control based on forbidden virtual fixtures in robot-assisted nasal surgery," *IEEE Access*, vol. 6, pp. 44905–44916, 2018.

[17] B. Dahroug, B. Tamadazte, and N. Andreff, "Visual servoing controller for time-invariant 3d path following with remote centre of motion constraint," in *IEEE Int. Conf. on Rob. and Auto.*, pp. 3612–3618, 2017.

[18] M. M. Marinho, H. Ishida, K. Harada, *et al.*, "Virtual fixture assistance for suturing in robot-aided pediatric endoscopic surgery," *IEEE Rob. and Auto. Let.*, vol. 5, pp. 524–531, 2020.

[19] E. Olszewska, M. Wagner, M. Bernal-Sprekelsen, *et al.*, "Etiopathogenesis of cholesteatoma," *Eu. Arch. of Oto-Rhino-Laryn. and H. & N.*, vol. 261, no. 1, pp. 6–24, 2004.

[20] H. Hildmann and H. Sudhoff, *Middle Ear Surgery*. Springer Berlin Heidelberg, 2006.

[21] J. E. A. P. d. Aquino, N. A. Cruz Filho, *et al.*, "Epidemiology of middle ear and mastoid cholesteatomas: study of 1146 cases," *Braz J. of Otorh.*, vol. 77, pp. 341–347, 2011.

[22] J.-H. So, B. Tamadazte, and J. Szewczyk, "Micro/macro-scale robotic approach for middle ear surgery," *IEEE T. on Med. Rob. and Bion.*, vol. 2, pp. 533–536, 2020.

[23] So, J. H., Sobucki, S., Szewczyk, *et al.*, "Shared control schemes for middle ear surgery". *Frontiers in Robotics and AI*, vol. 9, 2022.

[24] A. Swarup, K. W. Eastwood, P. Francis, *et al.*, "Design, prototype development and pre-clinical validation of a novel instrument with a compliant steerable tip to facilitate endoscopic ear surgery," *J. of Med. Eng. & Tech.*, vol. 45,pp. 22–34, 2021.

[25] K. Lau, M. Stavrakas, M. Yardley, *et al.*, "Lasers in cholesteatoma surgery: A systematic review," *Ear, Nose & Throat J.*, pp. 94S–99S, 2021.

[26] So, J. H., Szewczyk, J., & Tamadazte, B., "Automatic laser steering for middle ear surgery", *IEEE/RSJ Int. Conf. on Intel. Rob. and Sys.*, pp. 394-400, 2022.

[27] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," *J. of Intel. and Rob. Sys.*, vol. 3, pp. 201–212, 1990.

[28] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The Int. J. of Rob. and Res.*, vol. 33, pp. 1006–1028, 2014.

[29] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *The Int. J. of Rob. and Res.*, vol. 6, pp. 3–15, 1987.

[30] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *The Int. J. of Rob. and Res.*, vol. 4, pp. 109–117, 1985.

[31] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *IEEE Rob. & Auto. Mag.*, vol. 13, pp. 82–90, 2006.

[32] E. Malis, F. Chaumette, and S. Boudet, "2 1/2 d visual servoing," *IEEE T. on Rob. and Auto.*, vol. 15, pp. 238–250, 1999.

[33] Seon, J.-A., and Tamadazte, B. and and N. Andreff, "Decoupling Path Following and Velocity Profile in Vision-Guided Laser Steering," *IEEE T. on Rob.*, vol. 31, pp. 280-289, 2015.

[34] F. Flacco, A. De Luca and O. Khatib, "Control of Redundant Robots Under Hard Joint Constraints: Saturation in the Null Space," *IEEE T. on Rob.*, vol. 31, pp. 637-654, 2015.