



**HAL**  
open science

## **A multi-trainee architecture for haptic hands-on training**

Angel Ricardo Licona Rodriguez, Arnaud Lelevé, Minh Tu Pham, Damien Eberard

### ► To cite this version:

Angel Ricardo Licona Rodriguez, Arnaud Lelevé, Minh Tu Pham, Damien Eberard. A multi-trainee architecture for haptic hands-on training. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019), Nov 2019, Macao, China. 10.1109/IROS40897.2019.8967771 . hal-02176416

**HAL Id: hal-02176416**

**<https://hal.science/hal-02176416>**

Submitted on 31 Jul 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A multi-trainee architecture for haptic hands-on training

A.R. Licona, A. Leleve<sup>1</sup>, M.T. Pham and D. Eberard

**Abstract**—Haptic simulators are used in many education domains to improve hands-on training performance. Many of these simulators are used in the medical domain, where it is particularly important to be trained, guided by an expert. However, the majority of haptic hands-on training devices are usable by only one user at the same time, either the trainer or the trainee. Dual-user approaches have enhanced this situation by allowing both trainer and trainee to use the same simulator to introduce guided training into the simulation. In this paper, we introduce a multi-trainee architecture that expands further a former dual-user hands-on haptic training solution designed according to an energetic approach. This system will permit several trainees to be involved in the same simulation at the same time, guided by a single trainer. It paves the way to the multi-trainee hands-on training on haptic simulators.

## I. INTRODUCTION

Simulators are used during medical training as the trainees must improve their skills before practicing on a real patient. Classical simulators include the use of manikins or corpses. Nowadays, modern systems allow for training via immersion in virtual environments. The advantages that these simulators offer versus classic ones are: trainees can repeat the procedure several times without getting short of supplies (which is the case with corpses) and get quantitative (versus qualitative) feedback to determine what skills they need to improve [1].

Medical students need kinesthetic feedback to get significant learning, especially for complex gestures in delicate tasks [2]. Kinesthesia is the ability to sense the movement and position of the body limbs. Thanks to it, haptic training can be performed, leading to a more kinesthetic focused training. This is why, in recent years, the use of haptic systems in medical training has been rising [3].

The majority of haptic training systems in the literature only propose a dual-user configuration which is intended to be used by only one trainer and one trainee [4]. A haptic training system supporting several trainees at the same time will allow a single trainer to train several trainees at the same time. With this approach, the students can benefit from more time to practice on complex tasks instead of waiting for their turn. This is an important advantage as medical students are generally numerous and haptic simulators very little available.

Authors are with Ampère Laboratory, INSA, universit  de Lyon, Villeurbanne, 69621, France

<sup>1</sup> arnaud.leleve@insa-lyon.fr

The authors would like to express their gratitude to the Consejo Nacional de Ciencia y Tecnologia (CONACYT) in Mexico for providing financial support to the PhD student.

In this paper, we introduce a multi-trainee architecture which will allow training, on a same haptic training simulator, multiple trainees at the same time. Also, one trainee can present his progress to the other users (the trainer and the other trainees). For this purpose, we expanded the ESC architecture introduced in [5] to an  $n$ -user architecture. This architecture, designed through an energetic approach, has the interesting property to be scalable.

The outline of this paper is as follows: in section II, we recall the main characteristics of dual-user architectures used in a hands-on training context and we explain why existing solutions do not suit to an extension of the number of users. In section III, we detail how we extend the ESC dual-user architecture to a multi-trainee system. Section IV details its experimental validation. Finally, in section V, we discuss the results obtained with this new architecture.

## II. HAPTIC SIMULATORS: FROM DUAL TO MULTI-USERS

### A. Dual-user Architectures

Most of the multi-user systems in the literature are dual-user based. They extend the teleoperation systems where a single user controls a remote slave device (robot) which interacts with its environment. A dual-user haptic training simulator is a system where two users can simultaneously manipulate, each one, their haptic device (masters), to perform a guided training on a (real or virtual) slave device. Both master devices are interconnected with the slave using a controller which synchronizes the three device motions. Figure 1 shows a system with one trainer, one trainee and one slave. Master 1 is the device used by the trainer and Master 2 is the device used by the trainee. A dominance factor  $\alpha$  is set to decide which user controls the slave, while the other is guided with some haptic feedback (see [6] for instance).  $\alpha$  leads to three interaction scenarios during the training:

**Demonstration mode**  $\alpha = 1$ : the trainer has all the dominance over the slave. He teaches a gesture without any disturbance from the trainee’s motions. He gets some haptic feedback from the interaction between the slave and its environment.

**Training mode** ( $0 < \alpha < 1$ ): the trainer and the trainee cooperate to perform a shared slave motion. In this mode, the trainee performs a task and is potentially corrected by the trainer in real-time. **Evaluation mode** ( $\alpha = 0$ ): the trainee has all the dominance over the slave. His gesture is evaluated by the trainer without any disturbance from the trainer’s motions. He gets some haptic feedback from the interaction between the slave and its environment.

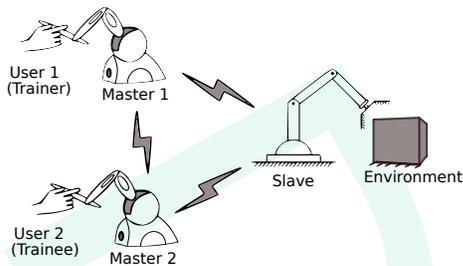


Fig. 1: Dual user haptic system

### B. Multi-trainee architecture

Figure 2 illustrates the concept of multi-trainee architecture. In this system, a trainer (using Master 1) can guide every trainee at the same time or evaluate the gestures of one of the trainees. Each trainee is assigned to each remaining master device. All the Master devices receive haptic feedback from the slave regardless who controls it. A complementary dominance factor is set to determine which trainee controls the slave while the others are only guided with some haptic feedback.

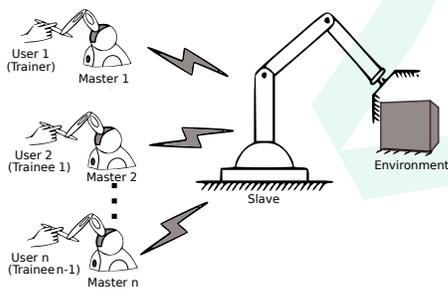


Fig. 2: Multi-trainee system

### C. State of Art

Khademian *et al.* have first studied multilateral architecture (Multi-Master/Single-Slave: MM/SS) for training purposes, while previous works on these systems were intended for collaboration purposes. They studied the transparency issues at first in [6], with two users and 3 degree-of-freedom planar devices. In 2011, they enhanced the control to guarantee the stability of this system against uncertainties in the environment and the user's dynamics, with the help of Llewellyn's unconditional stability criterion [7]. An experimental user study was carried out to assess the effectiveness of the proposed architecture in terms of transparency in a training context. More recently, Zakerimanesh *et al.* proposed, in [8], to link the master devices together and then to link them with the slave, generating a trilateral connection. They took into account the variable delays due to the connections. Simulations validate the correct behavior of this architecture, but at this time, it has not been experimentally confirmed. In all these works, the trilateral connection between the masters and the slave intrinsically prevents the addition of more haptic systems as the interconnection number would raise squared.

In [9], Khademian *et al.* introduced a criterium for unconditional stability of Multi-Master/Multi-Slave teleoperation systems, depending on the multi-port network parameters and the port terminations. However, the proposed robust stability analysis framework is examined only on their previous dual-user shared control architecture.

In [10], a MM/SS teleoperated framework is proposed to offer the opportunity of cooperative task performance to multiple operators. It can also be used for training purpose. The passivity of the system is investigated, taking into account time-varying communication delays (but with a assumption that they are all equal, which is not restricting as, in training applications, users are generally located in the same site). An impedance-based control methodology is developed to provide haptic feedback to every user. Nevertheless, the experiments show that when a user (such as the trainer) has the full control on the slave, the other users are not guided by this user and they get haptic feedback whatever their device's position. This behavior may not be efficient for the training as the trainees will not be able to synchronize position and force in their gesture learning.

This bibliographical study highlights the lack of advanced MM/SS frameworks likely to be adapted to multiple trainees. Thus, the next section introduces a solution to respond to the need of multi-trainee training systems.

## III. MULTI-TRAINEE HAPTIC TRAINING ARCHITECTURE

### A. ESC Architecture

To this purpose, we propose to extend the aforementioned ESC dual-user haptic training architecture introduced at first in [11] and expanded to  $N$  degrees of freedom (DOF) [5]. This architecture corresponds to the one displayed in figure but with only one trainee. In this figure, the control of only one joint is represented. In this case, the control for all the DOF is performed by duplicating this architecture joint by joint, independently. It is possible as every haptic device is identical and so has the same kinematics.

The dominance sharing between the users is performed through a dominance factor  $\alpha$  (first introduced in [12]) which is mainly set at 0 and 1 for aforementioned demonstration and evaluation modes, as it is difficult for both users to precisely dose their efforts when the control of the slave is shared between both users. In these modes, only one user pilots the slave (the leader user) while the other (the follower user) observes the gestures of the first one: his haptic device guides him on the trajectory of the leader user. When  $\alpha = 1$  (resp. 0), the trainer is the leader (resp. follower) while the trainee is the follower (resp. leader).

The ESC architecture is built upon the passive interconnection ( $D_{m1}$ ,  $D_{m2}$ , and  $D_s$ ) of passive components (IPC controllers). Connectors redirect the power signals ( $\theta, T$ ) between device controllers according to  $\alpha$ . Whatever user and environment, the passivity is guaranteed by a Time-Domain Passivity Controller (MFS in Fig. 2). More details about the design of ESC is available in [11] and [5].

The main property of this architecture is to provide to the trainee (in demonstration mode and respectively to the

trainer in evaluation mode) the ability to be guided by the other user while being provided with the same haptic feedback as the user who is manipulating the slave. This only necessitates that the guided user positions his device at the same position as the leading user, through a visual display on a monitor. This property has not been encountered in any other architecture so far.

### B. Expanding ESC to $n - 1$ trainees

Expanding the ESC architecture to an  $n - 1$  trainee system allows for the development of new simulators which support several trainees, receiving the same guidance at the same time. The main perspective is to provide one device for one trainer (Master 1) and  $n - 1$  devices for the  $n - 1$  trainees (Master 2 to Master  $n$ ). The challenge lies in how to preserve the passivity and the transparency quality (the aptitude to make users feel like they are manipulating directly the object, not through a master-slave system) of an architecture which significantly grows.

During the expansion of the ESC architecture, the node  $D_{m1}$  remains unaltered, as it is intended that Master 1 will always be the trainer's device. For the remaining ones, it is necessary to duplicate the node  $D_{m2}$  for each new trainee (it becomes  $D_{m2..n}$ ), with its corresponding IPC controller. It is also necessary to update  $D_{m2..n}$  and  $D_s$  in a way that permits to select only one trainee in evaluation mode. Figure 3 shows the expanded version of the ESC architecture.

To preserve the lossless properties of  $D_s$ , we updated it by duplicating the factors  $\beta_1$  and  $\beta_2$ , introduced in [11] to avoid the appearance of  $\alpha^2$  terms in the relations between exchanged angular velocities and torques. The relations between  $\alpha$  and  $\beta_1, \beta_2$  remain defined as

$$\beta_1 = \begin{cases} \alpha, & \alpha = 1, 0 \\ 1, & 0 < \alpha < 1 \end{cases} \quad \beta_2 = \begin{cases} \alpha, & \alpha = 1, 0 \\ 0, & 0 < \alpha < 1 \end{cases} \quad (1)$$

The expansion of node  $D_s$  is shown in equations 2 and 3:

$$\begin{bmatrix} \dot{\theta}_{s1} \\ \dot{\theta}_{s2} \\ \vdots \\ \dot{\theta}_{sn} \\ T_{rs} \end{bmatrix} = D_s \begin{bmatrix} T_{s1} \\ T_{s2} \\ \vdots \\ T_{sn} \\ \dot{\theta}_{rs} \end{bmatrix} \quad (2)$$

$$D_s = \begin{bmatrix} 0 & 0 & \cdots & 0 & \beta_1 \\ 0 & 0 & \cdots & 0 & 1 - \beta_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 - \beta_2 \\ -\beta_1 & \beta_2 - 1 & \cdots & \beta_2 - 1 & 0 \end{bmatrix} \quad (3)$$

where  $D_s$  remains a skew-symmetric matrix (which ensures lossless interconnection between its input-output energetic ports),  $\dot{\theta}_{s1..n}$  are the angular velocity outputs from the slave towards all master IPC respectively,  $T_{s1..n}$  are the torques transmitted by each master IPC to the slave IPC, and  $T_{rs}$  is the torque transmitted to the slave IPC, taking into account the dominance factor.

The  $\alpha$  factor still determines the dominance level between the leader and the follower, knowing that each one can be the trainer or one of the trainees. As it is desirable that only one trainee interacts at the same time with the trainer and the slave in evaluation mode (while the remaining trainee devices only receive feedback from the slave), the selection of the active trainee is manually established in real-time by the trainer, to indicate which one, out of the trainees, gets the dominance. Therefore, a complementary trainee selection mechanism has been introduced in conjunction with  $\alpha$ :  $\Delta \in \{2..n\}$ . This discrete variable establishes which of the trainee devices interacts with the slave and the trainer. Top right part of Figure 2 illustrates how the new hierarchy of dominance works. To implement  $\Delta$  in the ESC architecture, we split it in  $(n - 1)$  boolean variables  $\delta_i$  whose value is provided in equation 4. This way, we could integrate them into each  $D_{m i}$  node ( $i \in \{2..n\}$ ).

$$\delta_i = \begin{cases} 1 & \text{if } \Delta = i \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

When  $\delta_i = 0$  the master device  $i$  only receives haptic feedback and is positioned according to the slave device, whatever value of  $\alpha$ . When  $\delta_i = 1$ , the master device  $i$  has a dominance established by  $\alpha$ . Equations 5 and 6 show the resulting skew-symmetric matrices for  $D_{m i}$  nodes.

$$\begin{bmatrix} \dot{\theta}_{ri} \\ T_{si} \\ T_{sfi} \end{bmatrix} = D_{m i} \begin{bmatrix} T_{ri} \\ \dot{\theta}_{si} \\ \dot{\theta}_{sfi} \end{bmatrix} \quad (5)$$

$$D_{m i} = \begin{bmatrix} 0 & x_{1i} & x_{2i} \\ -x_{1i} & 0 & 0 \\ -x_{2i} & 0 & 0 \end{bmatrix} \quad (6)$$

where  $\dot{\theta}_{sfi}$  is the angular velocity injected by the MFS # $i$  into the trainee's IPC # $i$ ,  $T_{sfi}$  is the torque provided by the trainee's IPC # $i$  to the MFS # $i$ ,  $\dot{\theta}_{ri}$  is the angular velocity transmitted to the trainee's IPC # $i$ , affected by the dominance computation on  $\dot{\theta}_{si}$  and  $\dot{\theta}_{sfi}$ , and  $x_{1i}$  and  $x_{2i}$  are defined in equations 7 and 8:

$$x_{1i} = (1 - \alpha)\delta_i \quad (7)$$

$$x_{2i} = 1 + \delta_i(\alpha - 1) \quad (8)$$

Note that  $D_{m i}$  is expressed for each trainee's device  $i$ . Their number is not limited but it permits to provide the dominance to only one trainee at a time. This extended architecture remains composed of passive components (out of the MFS) interconnected by lossless (and so passive) nodes. The passivity (and so its stability) of the whole system is then still depending on the behavior of the MFS. Therefore, we extended the aforementioned Time-Domain Passivity Controller accordingly. Equation 9 shows how the energy generated by each MFS  $E_{pi}$  is computed.

$$E_{pi}(t) = \int_{u=0}^{u=t} \alpha(u) \delta_i(u) T_{ri}(u) \dot{\theta}_{rsi}(u) du \quad (9)$$

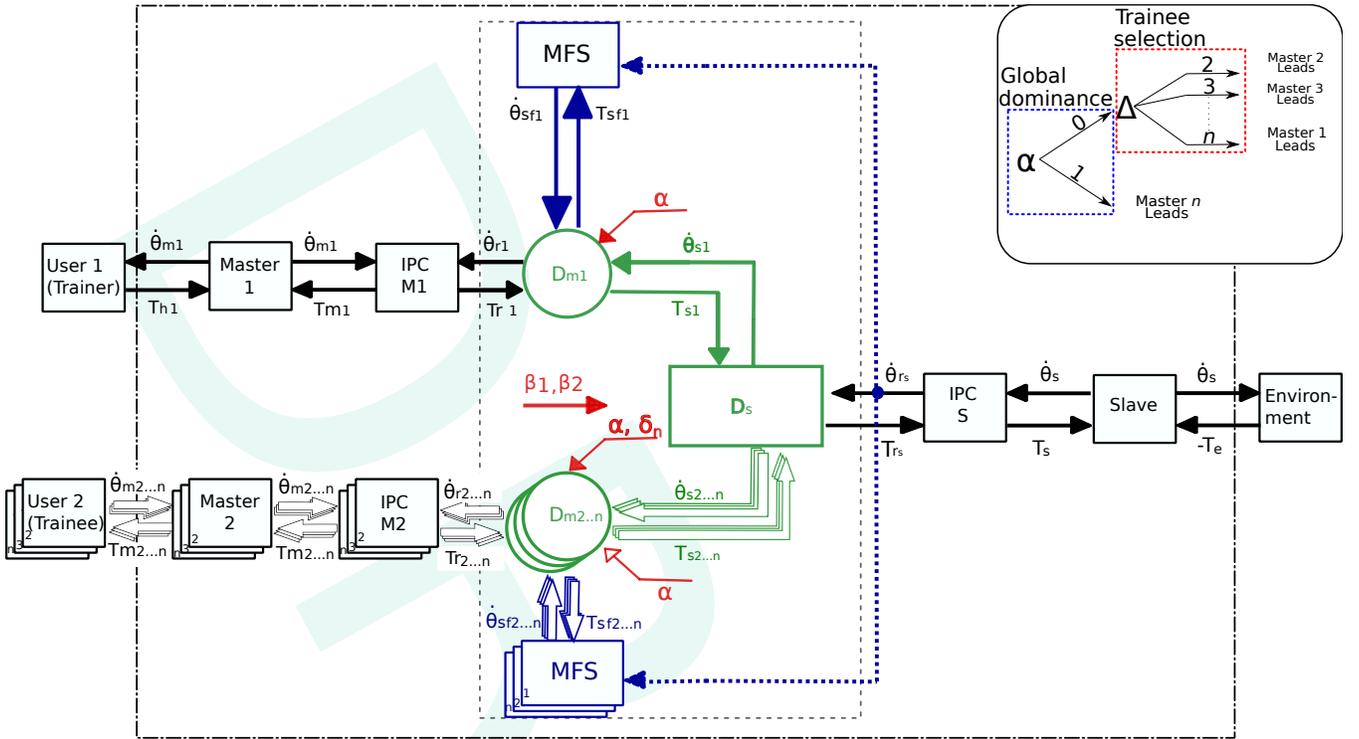


Fig. 3: ESC expanded to  $n$  devices

The passivity controller uses  $E_{pi}$  to determinate the real-time value of  $\dot{\theta}_{sfi}$  in equation 10, to ensure the preservation of passivity and the haptic feedback as long as the passivity is not compromised [5].

$$\dot{\theta}_{sfi}(t) = \begin{cases} \dot{\theta}_{rs}(t) & \text{if } E_{pi}(t) \leq 0 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

This approach is more conservative than the one used in the ESC dual-user architecture. Indeed, in the initial architecture, the global energy generated by both MFS was evaluated. In this extended version, we monitor the energy generated independently by each MFS. This is a cautious design choice which may be discussed.

#### IV. EXPERIMENTAL RESULTS

To validate this extended architecture, experiments were performed. The tests took place with two trainees, so three master haptic devices (M1, M2, and M3), and one slave (S). Due to the lack of available devices, out of these four haptic devices, only three were real Geomagic Touch 3D™ which propose six DOF (only three are actuated) and a maximum force of 3 N for each axis. The remaining one was a virtual model of the same kind which model was derived from [15]. In this section, Master 1 will be called M1v to recall that it represents a virtual device. We also could have chosen a virtual slave and three real masters. We preferred this configuration as a real slave provides more realistic force feedback in interaction phases. To show that the change of dominance between trainees does not disturb the controller, we chose to set a virtual trainer's device and real trainees' ones.

The control of every device and the simulation of the virtual one were performed by a single Mathworks Matlab Simulink® program, run on a PC (with an Intel® Core™ i7-6700 CPU clocked at 3.4GHz, an NVIDIA NVS 315 graphics board and 16Gb RAM, running with Microsoft® Windows 2010). The software library introduced in [16] was enhanced to communicate between the three real haptic devices and Matlab/Simulink. The devices' kinematic and dynamic parameters can be found in [17]. The torques indicated in this paper are retrieved from the haptic device library functions.

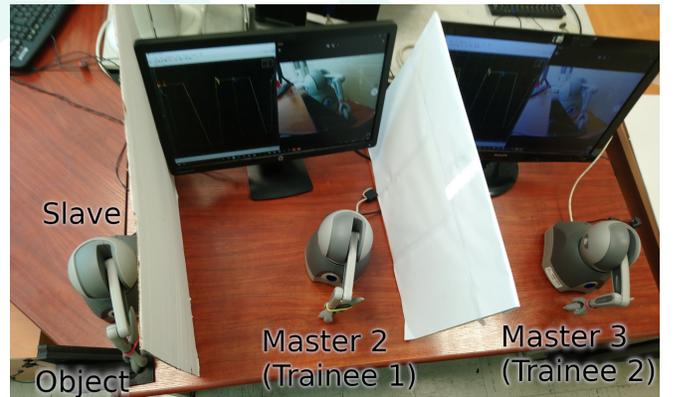


Fig. 4: Experimental setup

The purpose of these experiments was to check that position tracking and then force tracking between the leading user device, the slave and the followers' devices were correctly performed. Indeed, the user who gets the dominance controls

the slave's motion, which follows his/her gestures. The slave feeds back the interaction forces between its tip and its environment, towards every user. The follower users' devices reproduce the slave motion, to guide their respective user.

#### A. Position tracking

This experiment was performed to validate the position tracking between all devices, in free motion (no obstacle), and whatever value of the trainee selector  $\Delta$ .

Experiments validating the behavior of the system on dominance factor  $\alpha$  update have also been performed and provide the same results. All the follower devices must follow the leader's path, which is the one who has major dominance. For this experiment, we put the system in evaluation mode, so  $\alpha = 0$ : the trainer is following. During the task, the value of  $\Delta$  is changed from 1 to 2 at  $t_{\Delta} = 5$  s to verify that the change of leading trainee device does not affect negatively the tracking task. In this experiment, the leading trainees are asked to perform a helical path because it allows us to demonstrate the position tracking in all the degrees of freedom. Figure 5 shows the path performed in the cartesian space. At the beginning  $\Delta = 1$  so M2 is the leader device. At  $t = t_{\Delta}$ , the dominance changes ( $\Delta = 2$ ) and M3 becomes the leader. As the two follower devices and the slave were following very close trajectories, no glitch or oscillation are visible on  $\Delta$  change. The only difference before and after this moment is that there has been an exchange in the leading between M2 and M3.

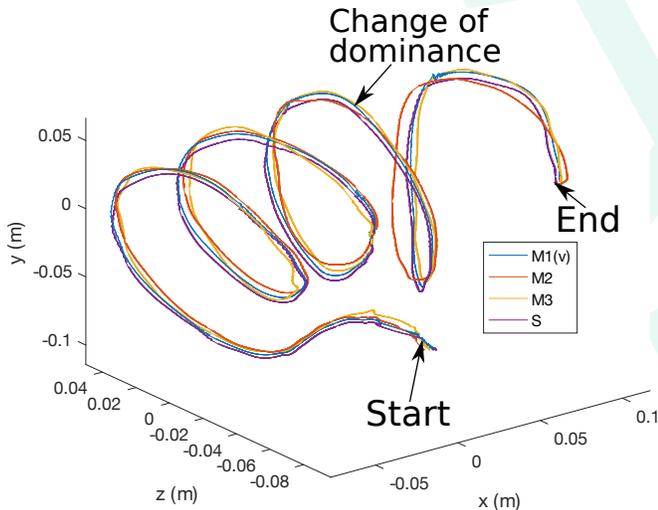


Fig. 5: Position tracking: helical path in free motion

Figure 6 shows the path tracking for each cartesian axis. In this experiment, M1v does not (virtually) apply any resisting force on his device, which reproduces a trainer who perfectly agrees with the trainees' gestures. This results in an artificially very good position tracking with the slave and leading trainee's devices, as visible in Table I. It confirms that every device trajectory is not disturbed by the change of dominance at  $t = t_{\Delta}$ .

To evaluate the performance of the position tracking, the Root Mean Square (RMS) tracking errors between the

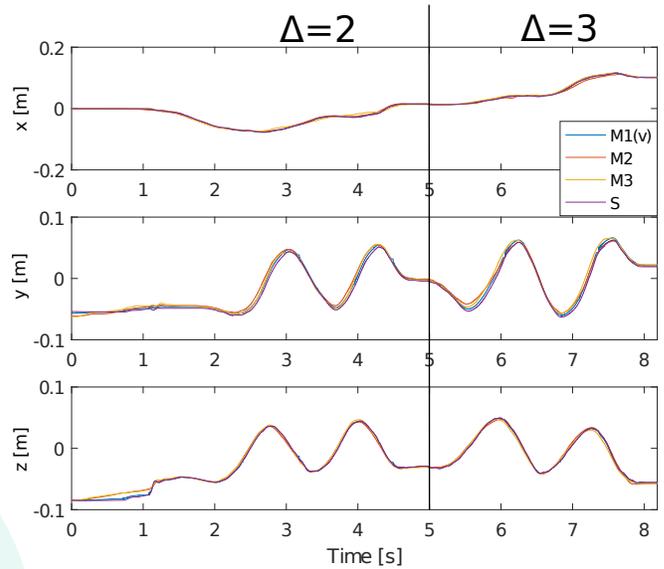


Fig. 6: Position tracking: devices' trajectories with  $\alpha = 0$

cartesian positions of the devices were computed, before and after the change of  $\Delta$ . For instance, the tracking error between M2 and S during the first period was:

$$\varepsilon_{M2-S}^{\Delta=1} = \sqrt{\frac{1}{k_{\Delta}} \sum_{k=1}^{k=k_{\Delta}} (\Delta P_{M2-S}[k])^2} \quad (11)$$

where  $k_{\Delta}$  is such as  $t[k_{\Delta}] = t_{\Delta}$  and

$$\begin{aligned} \Delta P_{M2-S}[k]^2 = & (x_{M2}[k] - x_S[k])^2 + (y_{M2}[k] - y_S[k])^2 \\ & + (z_{M2}[k] - z_S[k])^2 \end{aligned} \quad (12)$$

Table I shows these values when master devices M2 and M3 are the respective leaders. This table highlights that the RMS errors between the leader device and the other devices are smaller than between other devices. Typically, the error between M1v and M2 in the first part and then between M1v and M2 in the second part are very low (less than 3 mm) as the algorithm of M1 does not resist to the guiding motion. It is confirmed by the value of the M1v-S error which is around 2 mm in both cases. Also, the error between M2 and S (when M2 is the leader for  $t \leq t_{\Delta}$ ) is smaller than when M2 is a follower ( $t > t_{\Delta}$ ), and it is the same for M3. This is logical as the position errors between the slave and follower devices depend on the reaction of the users who may voluntarily resist to the guidance of their device and artificially raise this error. The cartesian RMS errors between each leader and the slave is around 4 mm. With this kind of off-the-shelf low bracket haptic device, we estimate that this is a good result.

This experiment showed that this extended version of the ESC architecture presents the same level of quality in position tracking as its dual-user version in evaluation mode when the trainer agrees with the proposed gesture. Complementary experiments have been performed in training mode

Error $\varepsilon$ [mm]	$\Delta = 2$ ( $t < 5$ s)	$\Delta = 3$ ( $t \geq 5$ s)
	M2 Leader	M3 Leader
M1(v) - S	1.8	1.8
M2 - M1(v)	<b>2.7</b>	4.9
M2 - S	<b>4.2</b>	5.1
M3 - M1(v)	3.5	<b>1.4</b>
M3 - M2	2.5	5.1
M3 - S	4.5	<b>3.5</b>

TABLE I: Cartesian position RMS error in free motion

and evaluation mode with Adaptive Authority Adjustment (AAA) introduced in [14] which automatically changes  $\alpha$  so that the trainer quickly gets back the authority on the slave when his gesture moves away from the one of the leading trainee because he disagrees with him. Due to lack of space, these experiments will be published in another paper.

### B. Force tracking

Once position tracking has been validated, we checked that the force tracking was also as accurate as in dual user configuration. In this experiment, leading trainees are asked to palpate a surface and the following users follow to feel the same force feedback. Two small "overshoots" are visible on the slave position in Fig. 7. It seems that they are due to the lack of rigidity of the arm in extension, at the moment of landing and taking off the surface. This test was performed similarly with  $\Delta$  changing from 1 to 2 during the task.

In the beginning,  $\Delta = 1$  and M2 controls the slave, which collides with a surface located at the articular position  $\theta_1 = -0.7$  rad of joint 1 (rotation around the vertical axis). Figure 7 shows this angular position and the corresponding torque. In this experiment,  $t_{\Delta} = 25$  s. We arbitrarily chose to present angular positions and torques instead of cartesian positions and forces as the slave collides a surface in an orthoradial direction, and the angular positions of master devices and the slave one are a little different, which would complicate the representation in a cartesian frame.

When the slave contacts the wall, the leader user applies some effort on the leader device, which positions its device beyond the position of the surface and permits him to get haptic feedback from the slave. This phenomenon depends on the stiffness of the surface, the power of the haptic device and the controllers' gains. The more they are elevated, the less the leader master device will go beyond the real surface position and the transparency will be better. This effect is due to a compromise between stability and transparency met in every master-slave system [18]. The leader user gets force feedback which corresponds to the interaction force between the slave and the surface (red and purple torque plots overlap). During this interaction, every follower device is guided near the position of the surface, when  $3 \text{ s} < t < 8 \text{ s}$ . To feel the same force feedback as the leader user, follower users are asked to position their device at the same position as the leader master (displayed to each follower on their screen). They move their device during the period  $8 \text{ s} < t < 13 \text{ s}$ . When the position of their device matches the leader's position, they get the same force feedback (when  $13 \text{ s} < t < 20 \text{ s}$ ).

Then, the dominance is changed to  $\Delta = 2$ , M3 becomes the leader and reiterates the same procedure.

Table II shows the torque errors during the periods where every master device are located at the same position as the leader to get the same force feedback from the slave. This table clearly shows that the leader is provided very good force feedback from the slave (the values are under the precision level) in both periods. Note that these values are estimated and provided by the haptic library software, not from force sensors. Concerning M1(v), to make it react as a "standard" trainee, motion towards the position of the leader has been artificially generated for it to behave realistically. The torque error between M1(v) and the other devices should thus be considered as the best performance reachable. The worst error with the leaders is  $5.10^{-3}$  N.m, corresponding to less than 1% (and 1% maximum between M1 and the slave). This proves that the perfect positioning of a follower device provides perfect force feedback.

Note that during all the period where each device is a leader (not only during the interaction), the relative torque error is around 5%. This is due to the transient modes when the slave tool gets into and leaves contact with the surface. These brutal changes of state disturb a little the force feedback control: they introduce some noise visible on torque plots at these moments, but this globally corresponds to less than 5%. Also, the users do not feel these disturbances in practice as they are filtered by the dynamics of the devices.

During the first (resp. second) phase, M3 (resp. M2) does not position exactly as M2 (resp. M3), with a relative error of 2%. It results in a force feedback relative error around 25%. This error amplification may seem problematic as it is difficult to position exactly one's device as the leader one only from plots on a screen. On the one hand, a better alignment graphic tool should help minimize this error, and on the other hand, one has to take into account the Just Noticeable Difference of force that users can detect. In [19], a JND of 10% has been experimentally determined for forces detected by the index finger with forces from 2 to 5 N. This corresponds to a JND force magnitude of 0.5 N. Assuming it applies in our case, an error of 25% on 5 N corresponds to a force magnitude error of 1.25 N which should be detectable by a human. This means that very precise positioning of the follower device is necessary to render good force feedback for the followers. More detailed experiments should be performed to determine this assumption in our case.

Error $\varepsilon$ [N.m]	$\Delta = 2$ ( $13 \text{ s} < t < 19 \text{ s}$ )	$\Delta = 3$ ( $35 \text{ s} < t < 42 \text{ s}$ )
	M2 Leader	M3 Leader
M1 - S	<b><math>2.10^{-3}</math></b>	<b><math>5.10^{-3}</math></b>
M2 - M1	<b><math>3.10^{-3}</math></b>	$114.10^{-3}$
M2 - M3	$128.10^{-3}$	$115.10^{-3}$
M2 - S	<b>0</b>	$115.10^{-3}$
M3 - M1	$128.10^{-3}$	<b><math>5.10^{-3}</math></b>
M3 - S	$128.10^{-3}$	<b>0</b>

TABLE II: Torque RMS error in contact

To conclude, this extended version of the ESC architecture also presents the same level of quality in terms of force track-

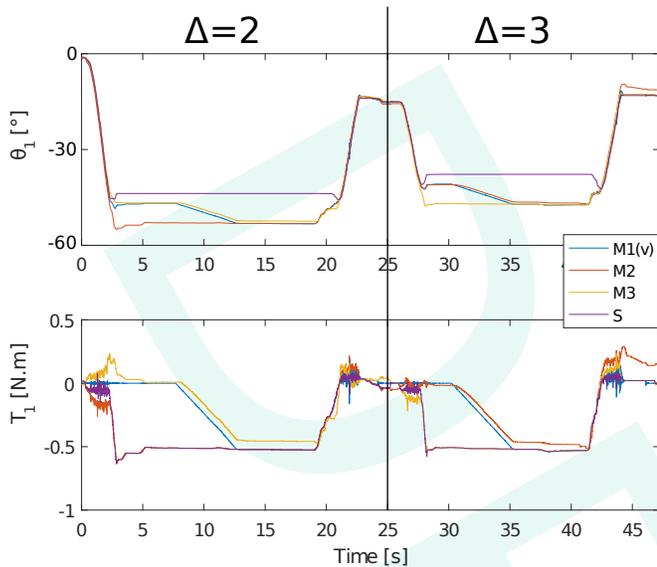


Fig. 7: Force tracking: angular positions and torques for joint 1

ing as its dual-user version. Some complementary experiments should be performed to determine whether the relative errors in the force feedback are physiologically detectable. As for the position tracking evaluation, other experiments have been performed with AAA. An enhancement of the graphic positioning help function should help users get even better precision in the force feedback.

## V. CONCLUSION

Providing haptic training systems supporting several trainees at the same time will allow for more efficient hands-on training sessions. To achieve this, we introduced in this paper an  $(n-1)$ -trainee architecture, based on the expansion of the previously proposed ESC dual-user architecture. We extended its architecture to manage the new simultaneous trainees and updated the passivity controller accordingly. The energetic approach used to design ESC presents good scalability which permitted to extend it without having to redesign it from scratch. Experiments with two real master devices, a virtual master and a real slave highlighted the correct behavior of the architecture concerning position and force tracking between leader users, slave, and follower users. The brutal changes of dominance between the two trainees did not introduce any disturbance. The performance of this new architecture looks appropriate for this kind of low bracket hardware. It still may be enhanced with high end off-the-shelf haptic devices and a better graphic positioning help function for follower users.

As future work, other combinations with several trainers are envisaged and the validation of AAA in this multi-trainee configuration. Also, the adaptation of this architecture to be used with a slave with different kinematics than the masters should permit to use a real slave robot (versus a haptic interface), as in [5] for the dual-user architecture. Also, we envisage considering the delays generated by communication

protocols when some of the users are located in remote locations.

## REFERENCES

- [1] C. Sutherland, K. Hashtrudi-Zaad, R. Sellens, P. Abolmaesumi, and P. Mousavi, "An augmented reality haptic training simulator for spinal needle procedures," in *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 3009–3018, 2013.
- [2] L. Panait, E. Akkary, R. L. Bell, K. E. Roberts, S. J. Dudrick, and A. J. Duffy, "The Role of Haptic Feedback in Laparoscopic Simulation Training," *Journal of Surgical Research*, vol. 156, no. 2, pp. 312–316, 2009.
- [3] T. R. Coles, D. Meglan, and N. W. John, "The role of haptics in medical training simulators: A survey of the state of the art," *IEEE Transactions on Haptics*, vol. 4, no. 1, pp. 51–66, 2011.
- [4] N. Vaughan, V. N. Dubey, T. W. Wainwright, and R. G. Middleton, "A review of virtual reality based training simulators for orthopaedic surgery," *Medical Engineering & Physics*, vol. 38, no. 2, pp. 59–71, 2016.
- [5] A. Licon, F. Liu, A. Lelevé, D. Eberard, and M. T. Pham, "Collaborative hands-on training on haptic simulators," in *Proceedings of the 2018 7th International Conference on Mechatronics and Control Engineering (ICMCE 2018)*, (Amsterdam, Netherlands), Nov 2018.
- [6] B. Khademan and K. Hashtrudi-Zaad, "Performance issues in collaborative haptic training," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 2007)*, pp. 3257–3262, April 2007.
- [7] B. Khademan and K. Hashtrudi-Zaad, "Shared control architectures for haptic training: Performance and coupled stability analysis," *The International Journal of Robotics Research*, vol. 30, no. 13, pp. 1627–1642, 2011.
- [8] A. Zakerimaneh, F. Hashemzadeh, and A. R. Ghiasi, "Dual-user nonlinear teleoperation subjected to varying time delay and bounded inputs," *ISA Transactions*, vol. 68, pp. 33–47, 2017.
- [9] B. Khademan and K. Hashtrudi-Zaad, "A framework for unconditional stability analysis of multimaster/multislave teleoperation systems," *IEEE Transactions on Robotics*, vol. 29, pp. 684–694, June 2013.
- [10] M. Shahbazi, S. F. Atashzar, H. A. Talebi, and R. V. Patel, "Novel cooperative teleoperation framework: Multi-master - single-slave system," *IEEE/ASME Transactions on Mechatronics*, vol. 20, pp. 1668–1679, Aug 2015.
- [11] F. Liu, A. Lelevé, D. Eberard, and T. Redarce, "A dual-user teleoperation system with adaptive authority adjustment for haptic training," in *Mechanisms and Machine Science*, vol. 39, pp. 165–177, 2016.
- [12] B. Khademan and K. Hashtrudi-Zaad, "Dual-user teleoperation systems: New multilateral shared control architecture and kinesthetic performance measures," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 5, pp. 895–906, 2012.
- [13] S. Stramigioli, *Modeling and IPC Control of Interactive Mechanical Systems - a Coordinate-Free Approach*. Springer, 1 ed., 2008.
- [14] F. Liu, A. Lelevé, T. Redarce, and D. Eberard, "A dual-user teleoperation system with adaptive authority adjustment for haptic training," in *Proc. of the 4th International Workshop on Medical and Service Robots (MESROB 2015)*, (Nantes, France), July 2015.
- [15] M. Tavakoli, R. V. Patel, M. Moallem, and A. Aziminejad, *Haptics For Teleoperated Surgical Robotic Systems*. World Scientific, 2008.
- [16] C. I. Aldana, E. N. no, L. B. nez, and E. Romero, "Operational space consensus of multiple heterogeneous robots without velocity measurements," *Journal of the Franklin Institute*, vol. 351, no. 3, pp. 1517–1539, 2014.
- [17] T. Sansanayuth, I. Nilkhamhang, and K. Tungpimolrat, "Teleoperation with inverse dynamics control for phantom omni haptic device," in *Proc. of SICE Annual Conference (SICE)*, pp. 2121–2126, Aug. 2012.
- [18] C. Zilles and J. Salisbury, "A constraint-based god-object method for haptic display," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 95)*, vol. 3, pp. 146–151, 1995.
- [19] M. S. Raghu Prasad, S. Purswani, and M. Manivannan, "Force JND for right index finger using contra lateral force matching paradigm," in *Proc of Conference ICoRD'13* (A. Chakrabarti and R. V. Prakash, eds.), (India), pp. 365–375, Springer India, 2013.