

Operation of a haptic interface for offline programming of welding robots by applying a spring-damper model

Angel Sanchez-Diaz, Ulises Zaldivar-Colado, J. Alfonso Pamanes-Garcia & Xiomara Zaldivar-Colado

To cite this article: Angel Sanchez-Diaz, Ulises Zaldivar-Colado, J. Alfonso Pamanes-Garcia & Xiomara Zaldivar-Colado (2019) Operation of a haptic interface for offline programming of welding robots by applying a spring-damper model, International Journal of Computer Integrated Manufacturing, 32:11, 1098-1116, DOI: [10.1080/0951192X.2019.1686177](https://doi.org/10.1080/0951192X.2019.1686177)

To link to this article: <https://doi.org/10.1080/0951192X.2019.1686177>



Published online: 06 Nov 2019.



Submit your article to this journal [↗](#)



View related articles [↗](#)



View Crossmark data [↗](#)



Citing articles: 2 View citing articles [↗](#)

ARTICLE



Operation of a haptic interface for offline programming of welding robots by applying a spring-damper model

Angel Sanchez-Diaz^a, Ulises Zaldivar-Colado^b, J. Alfonso Pamanes-Garcia^c and Xiomara Zaldivar-Colado^a

^aFaculty of Computing, Autonomous University of Sinaloa, Culiacan, Mexico; ^bFaculty of Computing, Autonomous University of Sinaloa, Mazatlan, Mexico; ^cLa Laguna Institute of Technology, National Technological Institute of Mexico, Torreon, Mexico

ABSTRACT

This paper studies a system of a virtual welding torch employed during the teaching process of welding tasks for offline programming of robotic manipulators. The torch is manipulated by the user in a virtual environment applying a haptic interface. An approach based on a spring-damper model is proposed to compute the force to be felt by the user during the manipulation of the interface. The force is such that more realistic sensations are perceived by the hand's user if a collision occurs of the torch in the virtual environment. A suitable strategy is employed to represent the virtual torch by two coupled models with spring-damper systems (SDS). Such models are termed kinematic virtual torch (KVT) and dynamic virtual torch (DVT). Thus, in the studied system, the DVT follows the geometric coordinates of the KVT during a manipulation of the torch avoiding penetration of other objects in the scene. Careful analyses are accomplished on manipulation tasks using different SDS configurations in a case study in order to show the efficacy of the proposed approach. As a result, a desired welding task is properly programmed for a robot in an easy and accurate way.

ARTICLE HISTORY

Received 26 February 2019
Accepted 17 October 2019

KEYWORDS

Robotic welding; path planning; haptics; offline programming; virtual reality; dynamic behaviour

1. Introduction

For the sake of competitiveness of modern enterprises in the global metal-mechanical sector, the robotisation of industrial welding tasks has become nowadays an essential issue. It is estimated that 25% of all industrial robots currently operating in the world are used for welding tasks (Karabegović and Mirza 2018). Nevertheless, in spite of the interesting features of modern technology, the classic offline programming process of robots for welding tasks remains as one of the greatest challenges that constrain a more comprehensive development of such systems (Larkin et al. 2018)

Consequently, some alternative approaches have been explored to simplify the programming process and do it more appealing for the user. One of these approaches includes the usage of a haptic interface (HI) to allow the user interacts easier with objects in a virtual environment (VE). Indeed, such a device makes the programmer feels on his hand the contact of the robot with objects in the VE when collisions occur in such a world (Burdea 1999). Unfortunately, a typical HI doesn't contribute to inspire an optical realism in a virtual scene because the visual interpenetration of objects cannot be avoided by using such a device. Evidently, the

interpenetration of objects doesn't occur in a real action. Providentially, the application of dynamic behaviour to solid objects in a VE can clarify the problem allowing more realistic simulations.

In this paper, an intuitive approach for the path teaching process required to define tasks for welding robots within a VE is presented. The present work proposes an offline approach for simplifying welding robot programming by using a HI. In this way, user specifies the desired motions of the robot within a VE on a PC. Since welding requires a high degree of accuracy, user can define accurate via-points from the motion of a virtual torch which is manipulated with a HI. Furthermore, a flexible planning of welding paths can be achieved, enabling non-experts to use it and reduce complexity and time of the robot programming issues.

As a relevant feature of our system, a spring-damper model-based technique was incorporated in such a way that the torch is represented in the virtual scene by two 3D models coupled by spring-damper systems (SDS): a kinematic virtual torch (KVT) and a dynamic virtual torch (DVT). Analogous notions were previously applied by Borst and Indugula (2005) in a system to move a virtual hand in a VE. When the user operates the HI, then the DVT follows the pose (position and orientation)

of the KVT. With the use of the DVT the manoeuvre of the torch in the VE makes the user feel as in an online programming environment. The DVT is a 3D rendering model in the virtual scene that emulates the real behaviour of the torch during its operation. On the other hand, the KVT is also a 3D rendering model but 'off-screen' (nonvisible), used to calculate the force that will be eventually transmitted to the user via the HI.

The paper is organised as follows. In Section 2 a literature review is presented. The overall framework architecture of the proposed system is presented in Section 3. Then, the virtual torch modelling is showed in Section 4 and the application of the spring-damper model in the torch is described in Section 5. The experiments with different SDS configurations in a case study are exposed in Section 6. Finally, the conclusion of the paper is presented.

2. Related works

The programming of a welding robot by applying typical procedures often requires an important amount of time of the programmer by using the robot. During this time, the robot is unproductive and consequently, a programming session becomes expensive. Accordingly, significant research efforts have been addressed in research organisations to develop simple and inexpensive programming systems for welding robots. A typical method of programming a robot is based on an online process that uses a teach-pendant attached to the robot's control. The programmer activates buttons of the teach-pendant for moving the robot and guiding the torch in such a way that this tool touches a sample of the desired path-points. Then, the corresponding taught torch poses are saved and computationally processed to finally obtain the whole desired sequence of robot's motions. Usually, this kind of programming involves manual repetitions, which are risky, slow and harmful for the production process (Pan et al. 2012). On the other hand, there are those researches oriented to offline programming (OLP) systems based in a VE of a workstation where a 3D model of the robot is incorporated. In this environment, the user specifies by simulation the required welding paths without interfering in the production process (Dong, Li, and Teng 2007). This graphic simulation is carried out within a VE structured by 3D

objects that are modelled in computer-aided design (CAD) software packages.

Supported by one of the packages, Carvalho, Siqueira, and Absi-Alfaro (1998) developed a program that extracts the data from a DXF file (AutoCAD® standard file format), this file includes all the information that specify the welding process. The simulation runs in the environment of this software. The same methodology is used by Pires, Godinho, and Ferreira (2004). Mitsi et al. (2005) presented a procedure for the automatic generation of the robot programming used in welding operations. In this work, the graphical simulation of the robot, torch, workpiece and workstation is achieved with SolidWorks™. The poses of the torch during a welding task are managed via keyboard and by using a dialogue box.

Within the CAD application called Autodesk Inventor® robot paths can be specified. A system like this was demonstrated by Neto and Mendes (2013) which uses line primitives for teaching a robotic system to do some paths. In the research of Ferreira et al. (2017), the welding path points are extracted from the CAD workpiece model, and the pose of the part to be welded is coming from a machine vision system. A personalised workbench was created in FreeCAD, a free and open source CAD 3D software. Soron and Kalaykov (2007) and Wu et al. (2015) introduced systems where the welding paths are specified directly on the CAD model of the workpiece and the poses of the torch are defined along the paths. The CAD models are collected in a robotic library.

By considering the advantages of CAD packages, the development of systems in which welding path planning is carried out has been previously detailed. However, the programming process associates with these systems require expert users with great skills in the use of CAD software (Larkin et al. 2016). Also, it requires people with broad knowledge in 3D environments in order to determine the information of the stitches that adjust a welding seam (Erdős et al. 2016).

In spite of attractive features of OLP systems, the interaction of the user with a VE could be not friendly enough if standard hardware interfaces (like a mouse or a joystick) are applied. Indeed, orientation and location of 3D objects in such environments could be a complex issue. In this way, virtual reality has been developed in the last few years because of other interesting technologies, e.g. – Lozano-Quilis

et al. (2013); Almajano et al. (2015) and Gorecky, Khamis, and Mura (2017) -. One of them is haptics, which applies the sense of touch to human communication with computer systems. Applications of haptics have been developed in many fields, e.g. in the aerospace industry, medicine, service, education, training and entertainment (El Saddik 2007).

The features of a first stage of an OLP system for arc welding tasks were presented by Alba et al. (2012). In this software, called WEROP (WElding Robots Offline Programming), the coordinates of a desired pose of the robot torch in a VE had to be explicitly specified by the user in a dialogue box. Unfortunately, this is an unfavourable procedure in an industrial environment. A new version of the WEROP system was later obtained. Zaldivar-Colado et al. (2014) presented the main characteristics of this work. In this, the benefits of using a HI are highlighted compared to the original functions presented by the first version of the WEROP system. A case study was shown in which the effectiveness of this experimental procedure was demonstrated, this was achieved because of the easy and instinctive interaction of the user in a VE. Although very little visual realism was seen in the scenes of the virtual world. In fact, some interpenetrations were observed between the torch and the other objects in the scene.

Sanchez-Diaz et al. (2017) presented a case study in which a HI was applied to the teaching of a welding task and the simulation of robot motion by achieving the task was performed in a VE. Moreover, a strategy was implemented by Sanchez-Diaz et al. (2018) based on the dynamic behaviour of a virtual torch used in an application for OLP of welding robots. The spring-damper model was applied for calculating the force in a haptic-feedback environment. In such a work, however, the coupling analysis between the involved torch models was not carried out.

Makris et al. (2014), Tsarouchi et al. (2016) and Blankemeyer et al. (2018) presented methodologies to make easier the robot programming. Since an intuitive and more natural interaction user-robot is carried out, a reduced amount of time is required for programming. However, the user must employ the robot in the workstation for a programming session and, consequently, it is not productive during this time. Also, it should be observed that the gestures are not always applicable to industrial environments, due to the lighting conditions and the dynamically

changing environment and only pick-and-place tasks are considered as application of the proposed approaches. Compared with these ones, in welding tasks, more constraints are imposed to the mobility of the end-effector. Indeed, more accurate poses of the torch are required on via-points in order to keep the electric arc and avoid collisions with plates to be welded.

By taking into account these issues, the method proposed here allows the user to get a faster and more accurate location of the virtual torch on via-points during the programming process of welding tasks. This paper studies a novel approach for the path teaching process required to specify tasks to welding robots within a VE. In our system, a spring-damper model-based technique is incorporated in such a way that the user feels more realistic sensations during the manoeuvres of the HI.

3. Framework architecture

The hardware used consists of a computer with an Intel® i5 processor with 2 cores, 2.30 GHz, 7.84 GB of RAM, a NVIDIA® GeForce GTX 950M video card with 2 GB of RAM. For user interaction with the VE, it is proposed to employ the Geomagic® Touch™ haptic interface (see Figure 1), produced by 3D Systems® Inc. Its technical specifications are given by the manufacturer company (3D Systems I 2015). It is a manipulandum of 6 degrees of freedom with a pencil-like end-effector that can take arbitrary orientations into its workspace.

Overall software architecture is shown in Figure 2, which is based on the virtual assembly system named VEDAP-II (Zaldivar-Colado 2009). For 3D visualisation



Figure 1. Geomagic® Touch™ haptic interface.

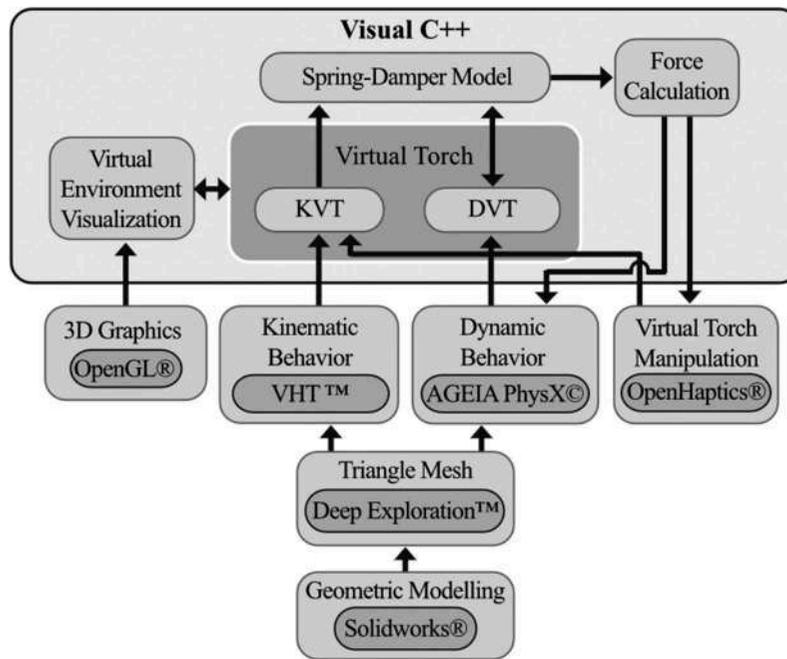


Figure 2. Software architecture.

purposes, the OpenGL[®] graphic rendering libraries are used in Visual C++ programming platform. OpenGL[®] is an open three-dimensional graphic library that is independent of the hardware and operating system, and can be obtained for free on all major current architectures, this makes it accessible for intermediate users of industrial robots. It has rich graphics capabilities and can render 3D graphics with excellent texture mapping, special effects and powerful real-time visualisation performance, which facilitates simulation issues.

Geometric modelling of the virtual torch – and the other objects – is realised by using the SolidWorks[™] CAD software. The 3D model generated in this software is converted into a triangle mesh format through the Deep Exploration[™] package. Virtual torch manipulation within the scene is achieved through the HI operation and its programming is carried out by using the OpenHaptics[®] software. This program has the HI kinematics models, so, HI end-effector geometric coordinates can be obtained, that is, its pose in 3D space. These coordinates are applied as a transformation to the virtual torch model, whose movements are linked to the HI.

AGEIA PhysX[©] software development kit is used for the dynamic behaviour of the virtual torch and the detection of collisions between the 3D models in the VE. In order to provide the user with the feeling of ‘touch’ when the torch is in contact with other objects

in the scene, the spring-damper model is used to calculate the force to send to the HI. This software uses the triangle mesh of the 3D torch model to create the DVT. Likewise, Virtual Hand Toolkit[™] (VHT) takes this mesh to generate the KVT model and manage its kinematic behaviour.

One of the main drawbacks in a VE is the lack of physical properties of the objects that make it up, e.g. weight, inertia and friction, like real-life objects. Consequently, a VE cannot physically restrict the movement of objects as it would be in a real environment. This causes an unrealistic movement of a virtual object and visually distracts the user when this one ‘passes’ through other virtual objects. Visual feedback can be improved by keeping the virtual object model out of the objects that it contacts. The use of dynamic behaviour in a VE allows a simulation more similar to reality, by avoiding the visual interpenetration between its objects.

Dynamic behaviour is obtained by the integration of a physics-based engine, which is a computer program that simulates Newtonian physics models to predict what will happen in the real world in a specific situation. Physics-based engine called AGEIA PhysX[©] is a framework for real-time physical simulation, published and supported by NVIDIA[®] Corp. This engine provides an integrated library for simulation based on laws of physics and supports the simulation of rigid bodies and soft objects. Also, this

package has a great simulation performance (Mondesire et al. 2016).

Furthermore, with the aid of the above engine, the construction of the physics and collision models, and the simulation of physical interactions of the objects during virtual path planning of welding tasks are simplified. Then, the present system is achieved because of the relevant collision detection approaches, numerical computation library and dynamics simulation schemes to significantly reduce the complexity of and effort in software development (Choi, Chan, and Pang 2012).

In the interest of obtaining a torch with dynamic behaviour in the VE, a technique based on the spring-damper model is used, which implies the creation of a KVT and a DVT. These models show an artificial coupling between them through the use of virtual springs-dampers (Borst and Indugula 2005), in such a way that during the torch motion, the DVT follows the KVT pose (Garbaya and Zaldivar-Colado 2007).

With the integration of OpenGL® graphic rendering libraries within Visual C++, the VE was created. The aim of rendering is to provide a two-dimensional view of an object in three dimensions, that is, it transforms an object designed in 3D and calculates a representation as realistic as possible in the two dimensions of the screen. Figure 3 shows the VE from the user's perspective where the global reference frame called Σ_G is indicated. The welding torch, a worktable and two plates to be welded as a Tee joint located on the table are considered as objects in the VE.

4. Virtual torch modelling

In order to carry out the planning of welding paths, user must manipulate the virtual torch via the HI. It has been suggested that user has to operate the HI by moving its

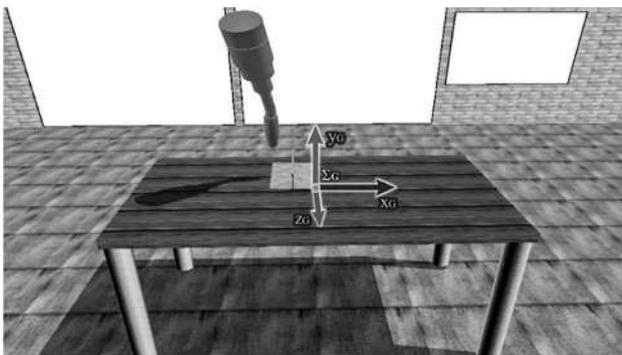


Figure 3. Virtual environment.

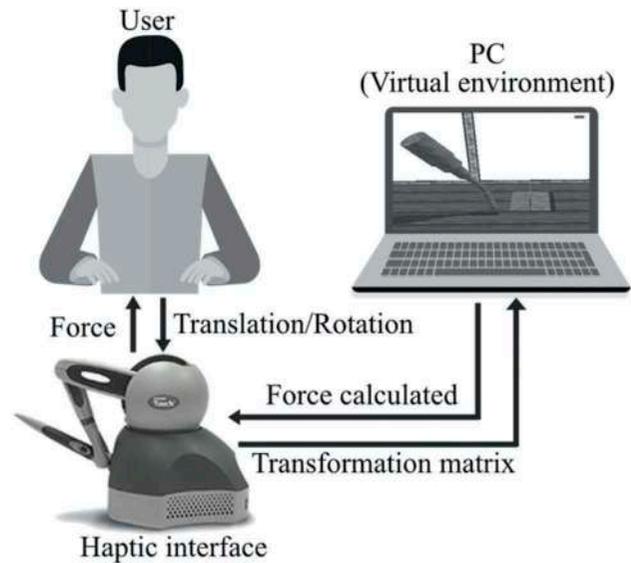


Figure 4. Scheme of the process for locating the torch in the VE through the HI.

end-effector with his hand to locate the welding torch in the VE. The movement of this virtual tool is achieved in real time during the HI manipulation. The manoeuvres executed by user are based on his visual perception of the VE displayed on the computer. Due to the DVT physical modelling made in AGEIA PhysX®, any collision of it can be detected by the system. Then, the control signals are sent to the HI in such a way that the user feels the impact of the DVT instantly. Thus, a movement rectification of the interface should guide the user to the correct location of its end-effector. In Figure 4 such process is schematised.

The torch model used is Abirob® Series A 500 22° of the brand Abicor Binzel, whose overall dimensions were obtained from the technical sheet provided by the manufacturer and these are indicated in the 3D model shown in Figure 5. It is required that the torch model has the features of a triangle mesh to be exported to the VE. So, the process to achieve this export begins at the modelling of the 3D torch in SolidWorks™ (see Figure 6(a)). It was assigned the frame Σ_T at its tip (Figure 7). This model is exported

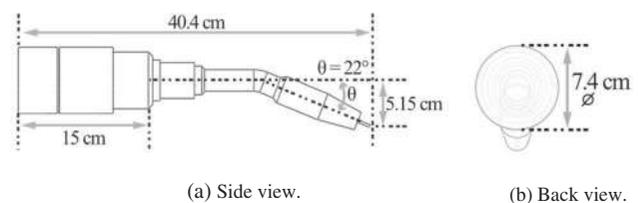


Figure 5. Dimensions of the 3D model of the torch.

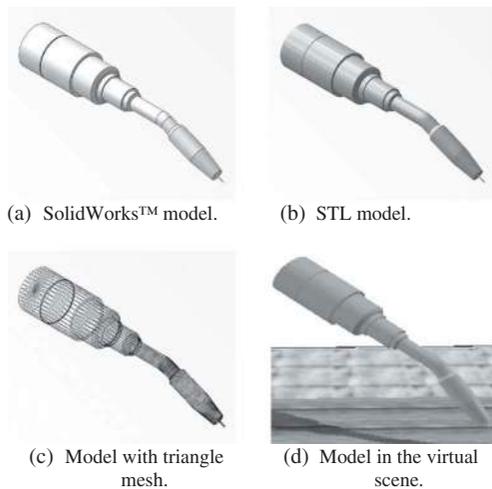


Figure 6. Modelling of the virtual torch.

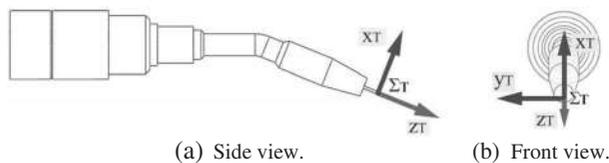


Figure 7. Frame of reference of the 3D model of the torch.

to a format called STL or stereolithograph, which is a purely geometric format (see Figure 6(b)). Triangle mesh data are generated by using the Deep Exploration™ package (see Figure 7(c)). Finally, the torch is drawn with the OpenGL® functions in the VE (see Figure 6(d)).

5. Realistic manipulation of the torch

In order to give the user a feeling of realism, the dynamic behaviour is applied, so, a coupling approach by spring-damper systems (SDS) between the KVT and the DVT has been adopted.

5.1. Kinematic virtual torch

In the program, the KVT is controlled with OpenHaptics® functions within Visual C++, which allows an intercommunication between the HI and the VE. Indeed, an OpenHaptics® function gives the transformation matrix of the HI end-effector with respect to Σ_G and it is used to calculate the KVT pose within the VE. Another function that declares an infinite loop is used to refresh the graphics on the screen. Within this loop the scene is redrawn.

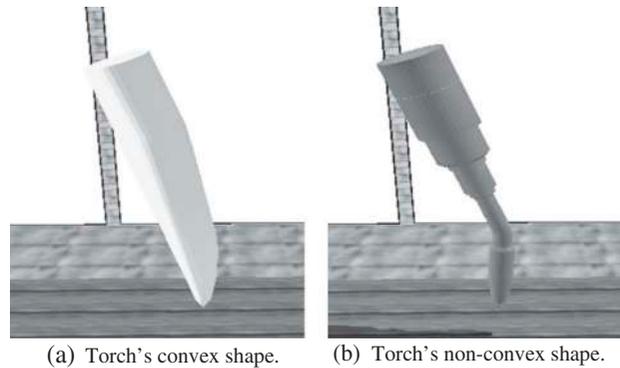


Figure 8. Rendering of the DVT and the KVT.

Likewise, in each iteration of this loop, the KVT pose is calculated based on the above matrix. In such a way, the movements that the user makes with the HI are reflected directly in the KVT.

VHT functions were used to render the KVT in the VE, which is constituted like a convex form (see Figure 8(a)). During the manipulation of the virtual torch, the KVT is 'off-screen', so it's also called nonvisible torch.

5.2. Dynamic virtual torch

The DVT is supported by AGEIA PhysX®, whose functions allow the detection of non-convex objects collisions. So, the DVT doesn't penetrate other objects in the VE. This is the main feature of dynamic behaviour. Besides, OpenGL® triangle mesh rendering is used to visualise the torch's non-convex shape (see Figure 8(b)).

When the user moves the KVT, the DVT tends to follow it by imitating real torch behaviour during the path planning phase. In the process for locating the torch in the VE, user only watches the DVT, which is also named visible torch (Youtube 2019a).

5.3. Virtual coupling

In order to obtain realistic behaviour during the contact between the virtual torch and the other objects in the environment, the spring-damper model was adopted based on a virtual coupling between the KVT and the DVT. This technique was originally proposed by Colgate, Stanley, and Brown (1995). This approach proposes to interconnect the KVT and the DVT with SDS. Likewise, the calculation of the contact force is made by using the difference of positions of both torches (see Section 5.7), as in the model developed by Borst and Indugula (2006).

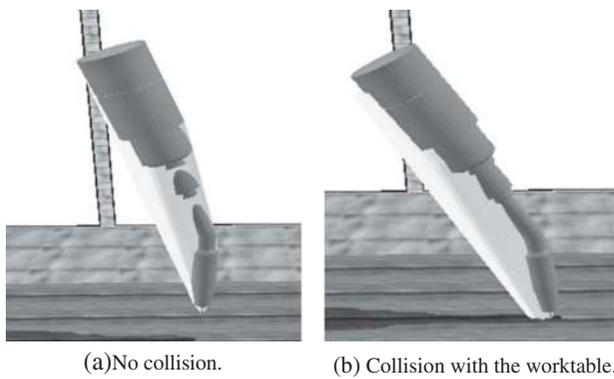


Figure 9. Coupling of the DVD and the KVT.

Before welding path planning takes place, both torches have the same pose. They seem to be one model of the torch. The DVT follows the KVT and reproduces the same pose without visually penetrating the other objects in the VE. The KVT rendering is aimed to make it stand out visually from the DVT. So, both torches can be identified by user when they have the same pose (see Figure 9(a)). The overlap of both torches remains as long as there is no contact of the DVT with any object. However, when a DVT collision occurs, the KVT penetrates the object and the DVT keeps 'attached' to the object (see Figure 9(b)) (Youtube 2019b).

5.4. Torch SDS configurations

A SDS is made up of a spring-damper and a reference mass at the DVT with its corresponding reference at the KVT. The springer-damper can be a linear springer-damper (LSD) or a torsional springer-damper (TSD). So, three torches with different SDS configurations were proposed. One linear SDS (LSDS) and one torsional SDS (TSDS) both at its centre of mass (CM) were assigned

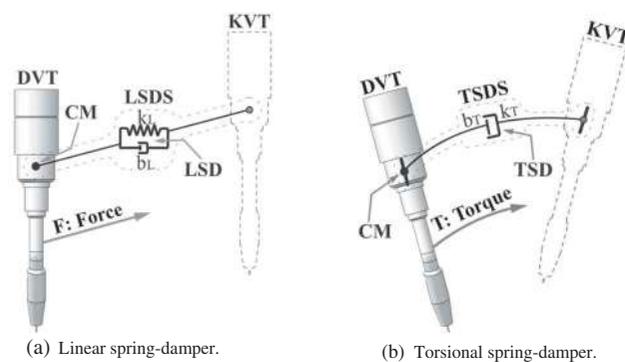


Figure 10. Torch with C_1 configuration.

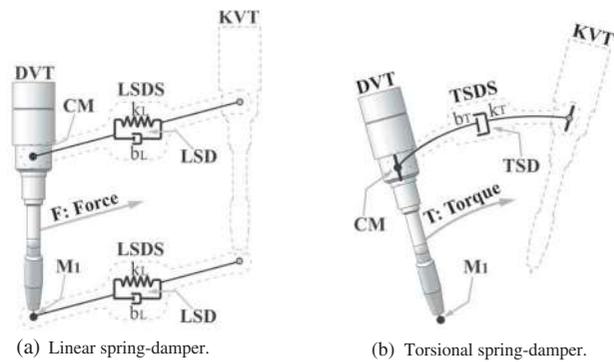


Figure 11. Torch with C_2 configuration.

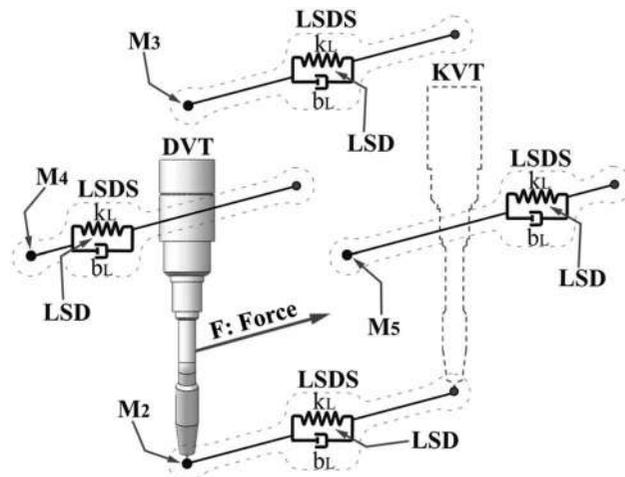


Figure 12. Torch with C_3 configuration.

at the first torch. This SDS assignment is called C_1 (see Figure 10). One LSDS was incorporated to the second torch at its tip (Σ_T) in relation to C_1 . This SDS configuration is called C_2 (see Figure 11). Finally, the third torch uses four LSDS: one at its tip, other is located near its base and two of them were placed at its sides. This SDS allocation is called C_3 (see Figure 12). Although in C_3 there is no TSDS, it is intended that the combined forces of the four LSD allow a DVT rotational motion.

A strategy was used in which once the force/torque is calculated, the resulting component will be applied to the corresponding reference mass at the DVT. In this way, the DVT is 'pushed' towards the KVT, by causing both torches have the same pose eventually (Zaldívar-Colado et al. 2009). In C_1 , CM is used as the only reference mass. Likewise, the mass M_1 and CM are used as reference masses in C_2 . In C_3 , the masses M_2 , M_3 , M_4 and M_5 are included at the DVT for the same purpose. Particularly, in this configuration, the masses are not physically attached to the DVT, but they keep a fixed position with respect to Σ_T during the DVT motion.

The CM position with respect to Σ_T is calculated by the physics-based engine, resulting $rx_{CM} = -6.39$ cm, $ry_{CM} = 0.00$ cm and $rz_{CM} = -28.91$ cm. In C_1 and C_2 this reference mass position was not changed during the manipulation tests. Also, once the M_1 position was established, it was not modified during these experiments with C_2 . Finally in C_3 , the M_2 position was not adjusted throughout the tests. The masses location in C_1 and C_2 are presented in Figures 16 and 17 and their positions are indicated in Table 3.

5.5. Linear spring-damper

Each LSD produces a translation force called linear component F :

$$F = k_L(p_k - p_d) - b_L(v_d - v_k) \quad (1)$$

where F is the force applied to the DVT in order to follow the KVT by executing a linear movement; k_L is the linear spring constant; b_L is the linear damper constant; p_d is the position of the mass attached to the DVT and p_k is the position of its KVT corresponding reference – both with respect to Σ_G . v_d and v_k are the velocities of the DVT and the KVT, respectively, and these are calculated as follows:

$$v_d = \frac{p_d^t - p_d^{t-1}}{\Delta t} \text{ and } v_k = \frac{p_k^t - p_k^{t-1}}{\Delta t} \quad (2)$$

where p_d^t, p_k^t are the current positions of the DVT and the KVT, respectively; p_d^{t-1}, p_k^{t-1} are the previous positions of the DVT and the KVT, respectively; Δt is the elapsed time between the position at t and at $(t-1)$.

5.6. Torsional spring-damper

The TSD produces a rotation force called torque T , which consists of two elements: the restoring torque (T_R) and the damper torque (T_D). The transformations calculated here are based on quaternion coordinate system. By starting from q_d and q_k represent the current orientation of the DVT and the KVT, respectively. The rotational difference between both torches is calculated, with respect to Σ_G :

$$q = q_d^* q_k \quad (3)$$

Then, the torque is computed by:

$$T_R = k_T(Ang(q)) \quad (4)$$

where T_R is the magnitude of the torque applied to the DVT about the rotation axis: $Axis(q)$ extracted from q ; k_T is the torsional spring constant and $Ang(q)$ is the rotation angle extracted from q . To calculate T_D , it is necessary to get the angular velocity of both torches with respect to Σ_G .

$$Ang(q_{wk}) = \frac{Ang(q_k^* q_k^{t-1})}{\Delta t} \quad (5)$$

$$Axis(q_{wk}) = Axis(q_k^* q_k^{t-1}) \quad (6)$$

where $Ang(q_{wk})$ is the rotation angle of the KVT angular velocity; $Axis(q_{wk})$ is the rotation axis of the KVT angular velocity; q_{wk} is the quaternion representing the KVT angular velocity; q_k^* is the conjugated quaternion of the KVT current orientation; q_k^{t-1} is the quaternion representing the KVT previous orientation and Δt is the elapsed time between $(t-1)$ and t situations. If the above formulation is used to obtain q_{wd} , then:

$$Ang(q_{wd}) = \frac{Ang(q_d^* q_d^{t-1})}{\Delta t} \quad (7)$$

$$Axis(q_{wd}) = Axis(q_d^* q_d^{t-1}) \quad (8)$$

The damper torque is obtained by:

$$T_D = -b_T(Ang(q_{wd} q_{wk}^*)) \quad (9)$$

where T_D is the torque that is applied to the DVT with respect to the rotation axis: $Axis(q_{wd} q_{wk}^*)$; b_T is the torsional damper constant and $Ang(q_{wd} q_{wk}^*)$ is the DVT relative velocity with respect to the KVT.

5.7. Contact force sensation

In order to calculate the force to be transmitted to the user via the HI, during the virtual torch manipulation, the difference of positions of the tips of both torches is required. The above is expressed:

$$F_H = \alpha(p_k - p_d) \quad (10)$$

where F_H is the force sent to the HI; α is the force resolution, $1 \leq \alpha \leq 3$ and p_k, p_d are the tip positions of the KVT and the DVT, respectively. The value F_H is a Cartesian coordinated vector, which represents the force dimension that user feels in his hand via the HI. The force value sent to the Geomagic® Touch™ during the contact between the DVT and any object in the environment is represented by a vector leaving from the torch tip (see Figure 13). If the DVT doesn't have

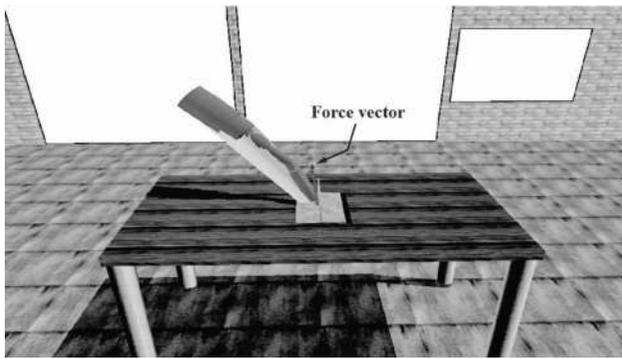


Figure 13. Force indicated during the torch manipulation.

any contact with the other objects, the DVT and the KVT are fully overlapped. In this case, there is no contact force:

$$(p_k - p_d) = 0 \therefore F_H = 0 \quad (11)$$

6. Experiments

Virtual coupling aims at a smaller distance between the DVT and the KVT, as well as a minimum difference between the angular values of them. Also, a stable and smooth movement of the DVT should also be observed. In order to find the best possible virtual coupling, three parameters of the SDS configurations were considered:

- Spring-damper constants.
- Position of the masses of the DVT.
- Masses of the DVT.

The experimental study aims at evaluating the values of position and orientation of both torches at the virtual coupling in the VE.

6.1. Case study

Virtual coupling between the KVT and the DVT with the three SDS configurations has been verified and validated in two types of experiments: the first of them, X_1 , it consists of executing an automatic path with the virtual torch and in the second one, X_2 , user performs a path with the virtual torch via the HI. The expected behaviour is that the DVT follows the KVT as close as possible. At experiment type X_1 , an automatic linear path is programmed where the KVT presents a constant velocity. During the path course, the positions of the tip and the centre of mass of both torches as

well as their angular values are recorded in each time instant – with respect to Σ_G -. This path starts at point P_1 and it ends at point P_2 . KVT Cartesian coordinates at point P_1 and at point P_2 are $Px_1 = -15$ cm, $Py_1 = 10$ cm, $Pz_1 = -15$ cm and $Px_2 = 25$ cm, $Py_2 = 10$ cm, $Pz_2 = 15$ cm, respectively. Likewise, KVT Bryant angles at P_1 and P_2 with respect to Σ_G are $\lambda_1 = 120^\circ$, $\mu_1 = 20^\circ$, $\nu_1 = 50^\circ$ and $\lambda_2 = 30^\circ$, $\mu_2 = 70^\circ$, $\nu_2 = 140^\circ$, respectively. A linear interpolation of these values is made in an interval of 2 s.

Experiment type X_2 consists of moving the KVT from P_1 to P_2 , passing through P_K (see Figure 14). The position values of both the tip and the centre of mass for each torch, as well as the orientation values of both torches, are recorded during the path at each instant of time – with respect to Σ_G -. The interpolation process is performed within a period of 2 s. Ten subjects all of them undergraduate students without prior experience with virtual reality systems participated to this experiment. They are all males, right-handed and aged between 18 and 23 years of age. The same group of subjects participated in the three torch's experimental conditions – C_1 , C_2 and C_3 – to carry out the manipulation of the torch through the HI (see Figure 15). In Table 1 are presented the configurations of the experiments.

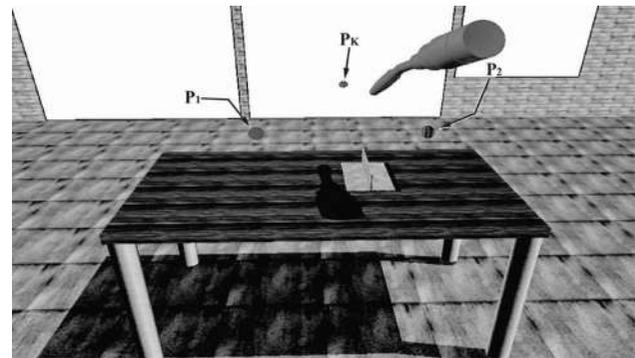


Figure 14. Virtual scene during experiments.



Figure 15. Subject manipulates the virtual torch via the HI.

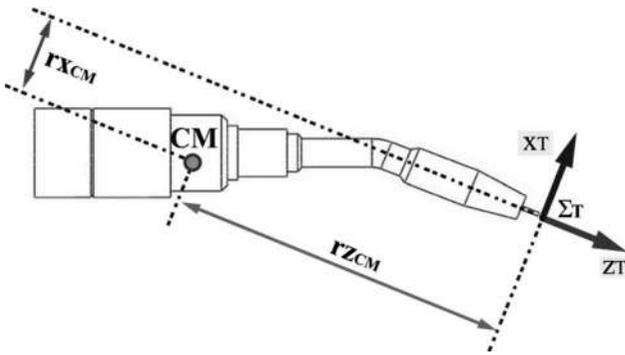


Figure 16. C_1 masses distribution (Side view).

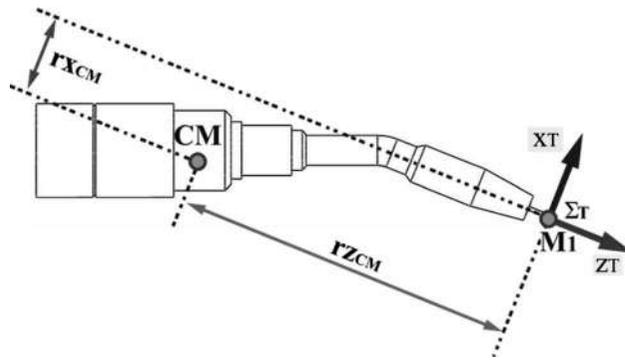


Figure 17. C_2 masses distribution (Side view).

6.2. Results and data analysis

The values of the constants of each spring-damper, both linear – k_L , b_L – and torsional – k_T , b_T – were obtained by performing manipulation tests of the virtual torch, in which they were adjusted with the aim of achieving the best possible virtual coupling.

Since in C_1 one LSDS and one TSDS were used, a set of linear constants and a set of torsional constants were obtained. In C_2 , two LSDS and one TSDS were used, thus two sets of linear constants and a set of torsional constants were acquired. Moreover, in C_3 there are four set of linear constants in this regard. These values are shown in Table 2. The positions of the masses attached to the torch in C_3 were eventually modified too, where experimentally different values were tested until to obtain

Table 1. Experiment configurations.

Experiment ID	SDS configuration	Type
1 (E_1)	C_1	X_1
2 (E_2)	C_2	X_1
3 (E_3)	C_3	X_1
4 (E_4)	C_1	X_2
5 (E_5)	C_2	X_2
6 (E_6)	C_3	X_2

X_1 – Automatic motion of the torch

X_2 – Manipulation of the torch with the HI

Table 2. Spring-damper constants.

SDS configuration	Mass	k_L	b_L	k_T	b_T
C_1	CM	24,682	12,145	3,478	1,126
C_2	M_1	107,871	39,913	-	-
	CM	10,827	6,821	3,064	626
C_3	M_2	263,863	72,769	-	-
	M_3	51,423	14,059	-	-
	M_4	473	128	-	-
	M_5	473	128	-	-

Units:

k_L : kg/s^2 ; b_L : kg/s^2

k_T : Nm/rad ; b_T : Nm/rad

Table 3. Positions of the masses with respect to Σ_T (in cm).

SDS configuration	Mass	rx_i	ry_i	rz_i
C_1	CM	-6.39	0.00	-28.90
C_2	M_1	0.00	0.00	0.00
	CM	-6.39	0.00	-28.90
C_3	M_2	0.00	0.00	0.00
	M_3	-13.87	0.00	-47.35
	M_4	-6.39	39.19	-28.90
	M_5	-6.39	-39.19	-28.90

a consistent virtual coupling in the trials performed. The C_3 masses distribution obtained is shown in Figure 18 and their positions along with those of C_1 and C_2 are listed in Table 3. Another variable considered was the value of each of the masses attached to the torch in order to acquire a steady virtual coupling and a stable torch during manipulation tests. According to these tests, CM mass value is zero in all cases. Table 4 shows the values obtained in this regard. For the calculation of the force to send to the HI, the coefficient α is empirically determined carrying out manipulation trials by adjusting the value of α until a steady force sensation is obtained, resulting $\alpha = 1.15$.

In order to evaluate the virtual coupling for the three SDS configurations, the data collected from E_1 , E_2 and E_3 were compared with each other, as well as the averages of the data collected in the 10 tests for each experiment type X_2 – E_4 , E_5 and E_6 – (see Tables 5 and 6). The minimum values of parameters are shown in bold for each type of experiment. In Table 5, an overall weighing coefficient has been calculated by the sum of their four parameters. Moreover, an average weighing coefficient also has been calculated by adding AADT and ADDCM. Besides, in Table 6 an overall weighing coefficient has been calculated by adding MAD and AAD. The minimum values of these coefficients are also shown in bold. Further, in Tables 7-9 are listed the detailed data obtained in E_4 , E_5 and E_6 respectively. In each of these tables, an overall weighing coefficient has also been calculated by the sum of all the parameters of each subject. The minimum overall weighing values are

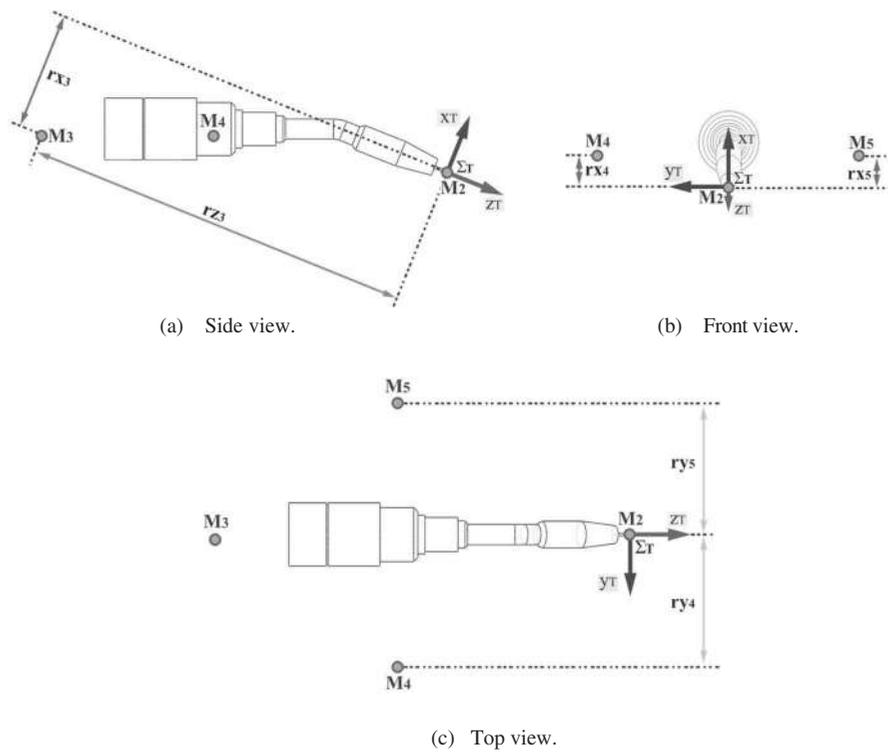


Figure 18. C_3 masses distribution.

shown in bold and the maximum overall weighing values have been underlined.

In E_1 , the result indicates a significant difference in both position and orientation between the two torches (see Figure 19). Only in MDDCM the results with this SDS configuration show the best performance within the configurations tested in experiments type X_1 – 2.69 cm -. MDDT is 7.81 cm; ADDT and ADDCM are 3.66 cm and 0.52 cm, respectively. Finally, MAD is 10.07° and AAD during the motion was 4.38°. The results collected in E_2

Table 4. Values of the masses (in kg).

SDS configuration	Mass	Value
C_1	CM	0.00
C_2	M_1	2,924.01
	CM	0.00
C_3	M_2	5,059.31
	M_3	1,338.95
	M_4	9.14
	M_5	9.14

show a significant improvement in the virtual coupling performance with respect to E_1 . The lines representing

Table 5. Distance differences between the KVT and the DVT (in cm).

Parameter	X_1			X_2		
	E_1	E_2	E_3	E_4	E_5	E_6
Max. Distance Difference (Tips) – MDDT -	7.81	1.13	1.09	3.03	0.40	0.45
Max. Distance Difference (Centres of Mass) – MDDCM -	2.69	3.59	3.16	1.15	1.34	1.20
Average Distance Difference (Tips) – ADDT -	3.66	0.16	0.18	0.97	0.13	0.14
Average Distance Difference (Centres of mass) – ADDCM -	0.52	0.74	0.48	0.29	0.39	0.34
Overall weighing	14.68	5.62	4.91	5.44	2.26	2.13
Average weighing	4.18	0.90	0.66	1.26	0.52	0.48

Table 6. Angular differences between the KVT and the DVT (in degrees).

Parameter	X_1			X_2		
	E_1	E_2	E_3	E_4	E_5	E_6
Max. Angular Difference – MAD -	10.07	4.00	3.53	3.27	1.63	2.01
Average Angular Difference – AAD -	4.38	1.07	0.71	1.11	0.52	0.50
Overall weighing	14.45	5.07	4.24	4.38	2.15	2.51

Table 7. Data obtained in E_4 .

Parameter	Subject									
	1	2	3	4	5	6	7	8	9	10
MDDT (in <i>cm</i>)	3.37	2.40	2.33	4.23	2.62	3.56	1.85	2.19	5.61	2.12
MDDCM (in <i>cm</i>)	1.45	1.69	0.84	1.18	1.03	1.14	1.08	1.45	1.21	0.46
ADDT (in <i>cm</i>)	1.01	1.00	0.73	0.95	0.81	1.28	0.73	0.86	1.55	0.75
ADDCM (in <i>cm</i>)	0.30	0.36	0.29	0.26	0.28	0.31	0.23	0.36	0.30	0.18
MAD (in <i>degrees</i>)	3.03	2.32	2.88	5.05	2.58	3.92	2.21	2.02	6.73	2.00
AAD (in <i>degrees</i>)	1.20	0.98	0.89	1.19	1.00	1.40	0.85	1.00	1.82	0.80
Overall weighing	10.36	8.75	7.96	12.86	8.32	11.61	6.95	7.88	17.22	6.31

Table 8. Data obtained in E_5 .

Parameter	Subject									
	1	2	3	4	5	6	7	8	9	10
MDDT (in <i>cm</i>)	0.32	0.35	0.26	0.33	0.66	0.53	0.37	0.39	0.54	0.26
MDDCM (in <i>cm</i>)	1.37	1.01	0.92	1.27	2.03	1.36	1.04	1.28	2.13	1.01
ADDT (in <i>cm</i>)	0.11	0.12	0.10	0.12	0.17	0.16	0.11	0.17	0.19	0.10
ADDCM (in <i>cm</i>)	0.39	0.33	0.24	0.39	0.45	0.51	0.30	0.44	0.45	0.35
MAD (in <i>degrees</i>)	1.69	1.46	0.98	1.74	2.91	1.70	1.13	1.47	1.66	1.52
AAD (in <i>degrees</i>)	0.54	0.53	0.38	0.63	0.53	0.68	0.38	0.54	0.49	0.46
Overall weighing	4.42	3.80	2.88	4.48	<u>6.75</u>	4.94	3.33	4.29	5.46	3.70

Table 9. Data obtained in E_6 .

Parameter	Subject									
	1	2	3	4	5	6	7	8	9	10
MDDT (in <i>cm</i>)	0.39	0.51	0.54	0.58	0.40	0.51	0.49	0.39	0.41	0.26
MDDCM (in <i>cm</i>)	1.18	1.03	1.06	1.37	0.94	2.63	1.32	0.94	0.80	0.66
ADDT (in <i>cm</i>)	0.13	0.15	0.14	0.16	0.14	0.12	0.15	0.17	0.13	0.08
ADDCM (in <i>cm</i>)	0.39	0.33	0.34	0.41	0.32	0.37	0.38	0.35	0.27	0.24
MAD (in <i>degrees</i>)	4.29	2.04	1.35	1.62	1.12	3.47	1.97	2.09	1.00	1.12
AAD (in <i>degrees</i>)	0.87	0.51	0.51	0.50	0.36	0.63	0.49	0.51	0.32	0.32
Overall weighing	7.25	4.57	3.94	4.64	3.28	<u>7.73</u>	4.80	4.45	2.93	2.68

the KVT position and the DVT position are overlapped in relation to their tips and their centres of mass. Only at the beginning of the motion, there is a small position difference (see Figure 20(a)). In terms of orientation, the lines depicting the KVT orientation and the DVT orientation are overlapped most of the time. There are only small angular differences at the beginning and at the end of the motion (see Figure 20(b)). MDDT and MDDCM are 1.13 cm and 3.59 cm, respectively. Also, the value 0.16 cm represents the best ADDT within experiments type X_1 . ADDCM is 0.74 cm. In angular values, MAD is 4.00° and AAD is 1.07° . Additionally, the best overall results in experiments type X_1 were found in E_3 . There is also an overlap of the lines that represent the KVT position and the DVT position (see Figure 21(a)). With regard to the angular values, there are small angular differences at the beginning and at the end of the motion (see Figure 21(b)). So that, MDDT is 1.09 cm and MDDCM is 3.16 cm. ADDT is 0.18 cm and ADDCM is

0.48 cm. In the angular differences, MAD is 3.53° and the AAD minimum value in experiments type X_1 is 0.71° .

By taking into account the overall weighing coefficients of X_1 experiments in Table 5, automatic motion of the torch with C_3 has the best result in virtual coupling – 4.91 cm -. Furthermore, by considering the average weighing coefficients of the same table, the programmed path of the torch with C_3 also showed the best performance in virtual coupling – 0.66 cm -. Besides, the overall weighing coefficients and AAD in Table 6 indicate that during the automatic motion of the torch, C_3 also presented the minimum values in angular differences between the KVT and the DVT – 4.24° -.

In experiments type X_2 , the values are lower with respect to X_1 experiments because of the change of the torch's orientation made by the users is not as great as in the automatic motion. From the data of the 10 tests in experiments type X_2 , the global averages for each parameter is obtained. So, the worst overall

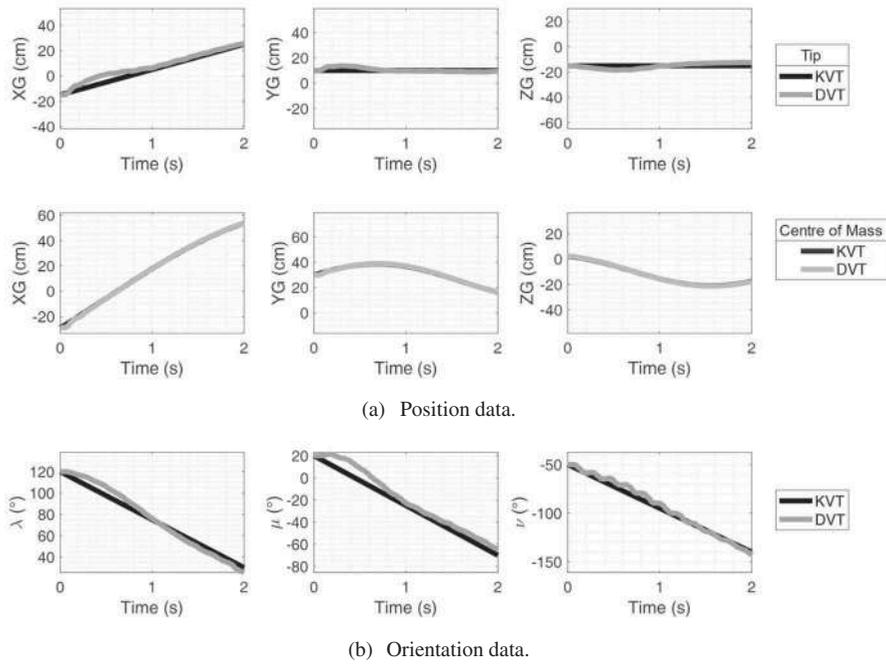


Figure 19. Results in E_1 .

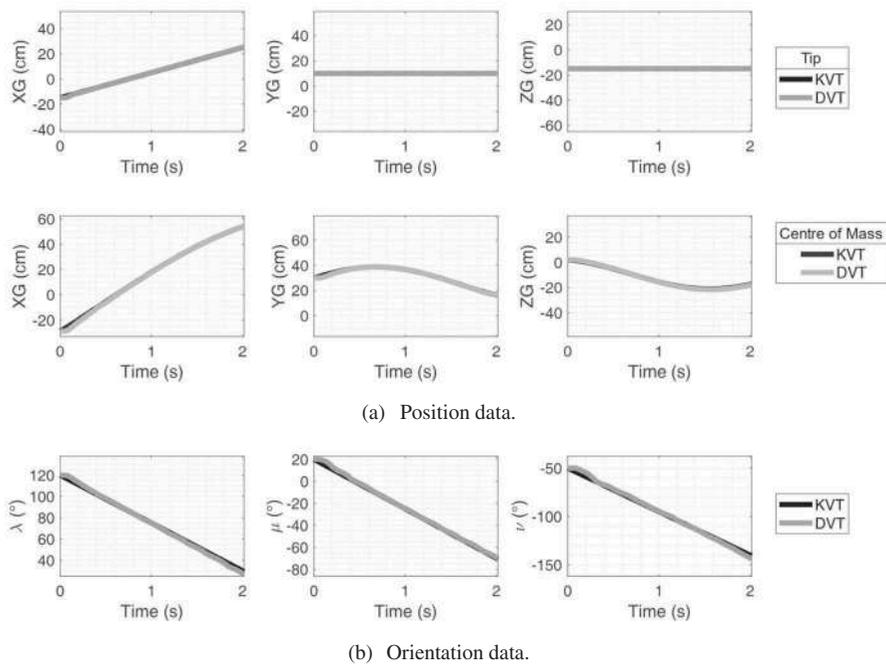


Figure 20. Results in E_2 .

results within experiments type X_2 are observed in E_4 . MDDT is 3.03 cm and MDDCM is 1.15 cm. The averages show that ADDT is 0.97 cm and ADDCM is 0.29 cm. Additionally, MAD is 3.27° and AAD is 1.11°. Data collected in E_5 represent a better virtual coupling overall performance within X_2 experiments. Thereby, MDDT is 0.40 cm and MDDCM is 1.34 cm. ADDT is

0.13 cm and ADDCM is 0.39 cm. Likewise, MAD is 1.63° and AAD is 0.52°. Moreover, in E_6 the behaviour of both torches was similar to the one presented in E_5 . Here, MDDT is 0.45 cm and MDDCM is 1.20 cm. ADDT is 0.14 cm and ADDCM is 0.34 cm. Finally, MAD is 2.01° and AAD is 0.50°. The worst and best results in X_2 experiments have been selected by considering the

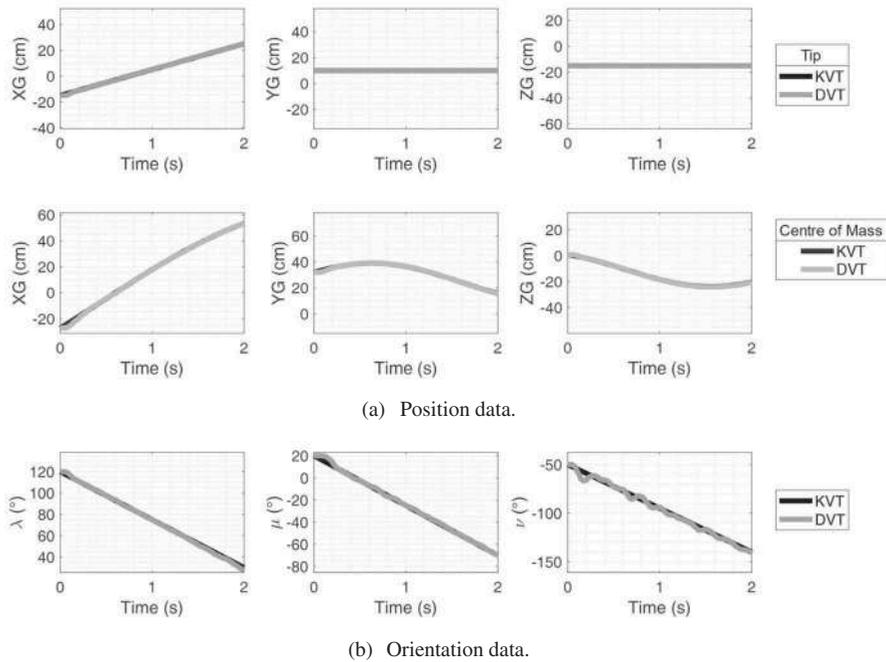


Figure 21. Results in E_3 .

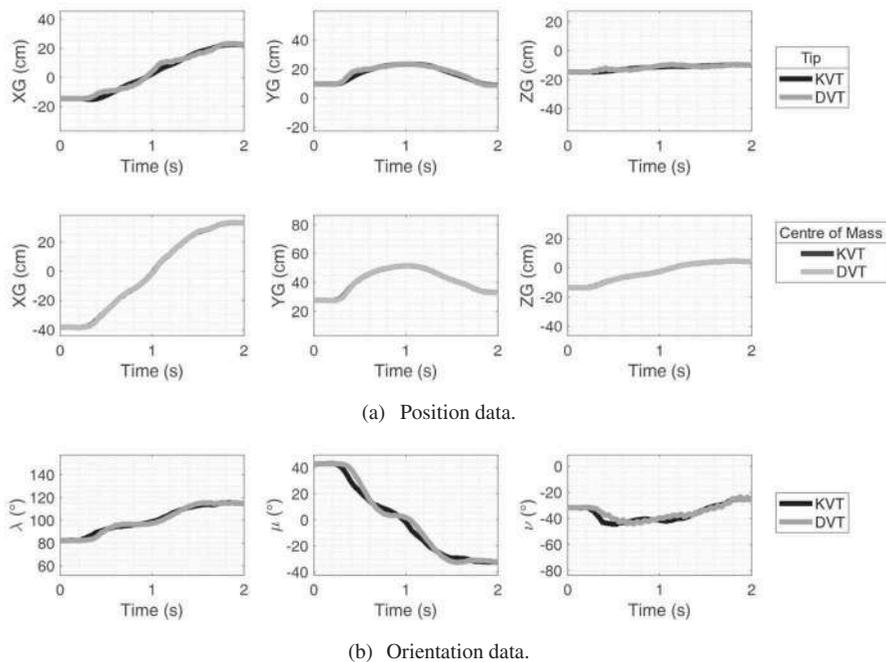


Figure 22. Worst results in E_4 (Subject 9).

values of their overall weighing coefficients. From Figure 22–27 show these results.

In order to determine which SDS configuration showed the best virtual coupling performance in X_2 experiments, the overall weighing coefficients of Table 5 were considered. So, when the torch was manipulated by using C_3 , the overall distance difference had the

minimum value – 2.13 cm -. Likewise, by looking at the average weighing coefficients, C_3 has the best SDS configuration when the users manipulate the torch with the HI – 0.48 cm -. On the other hand, in Table 6, C_2 presented the best virtual coupling overall performance by taking account their overall weighing coefficients – 2.15° -. Nevertheless, if we consider only AAD, the torch with

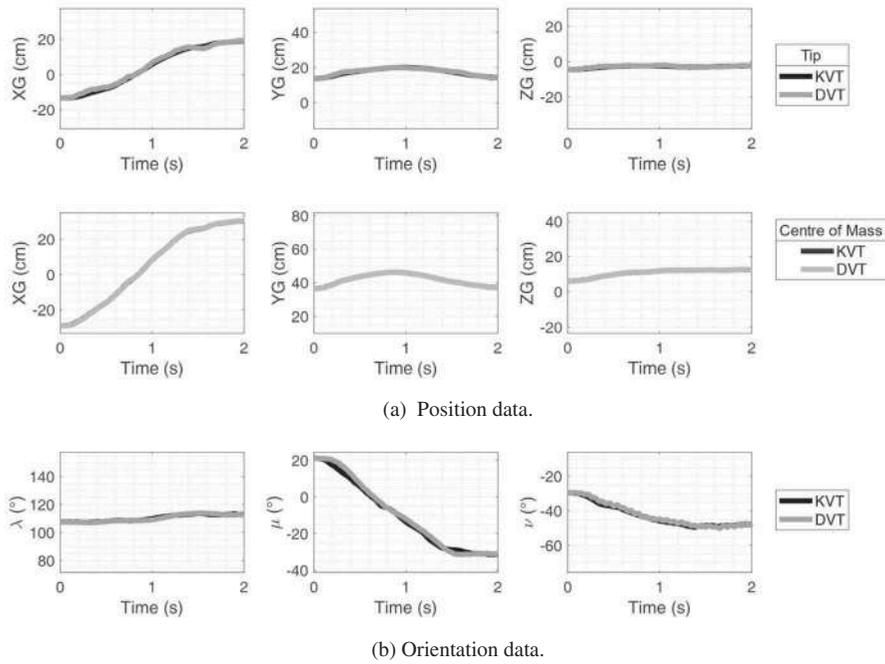


Figure 23. Best results in E_4 (Subject 10).

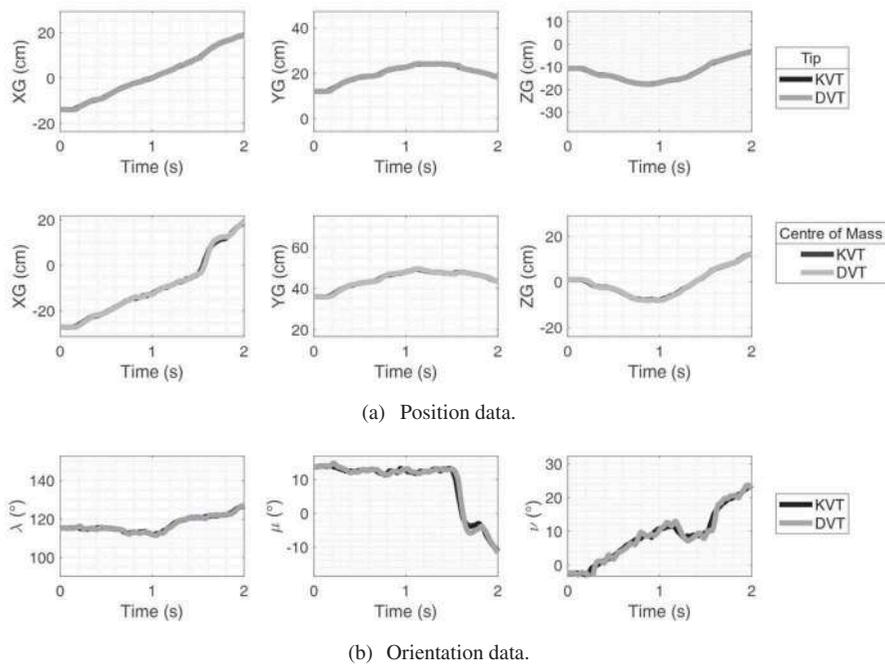


Figure 24. Worst results in E_5 (Subject 5).

C_3 showed the average minimum value -0.50° – in angular difference between the KVT and the DVT.

In summary, C_1 shows only the best overall virtual coupling when it refers to the centres of mass. This happens because according to this configuration only force is applied to the centre of mass of the torch without interfering any other force. In addition, the

result shows an improvement in virtual coupling with the use of C_3 compared with the use of C_1 and C_2 in the virtual torch automatic motion – X_1 experiment -. Thus, the use of only LSDS causes a better virtual coupling performance when the torch’s velocity is constant. On the other hand, when the users manipulate the virtual torch by using the HI – X_2 experiment

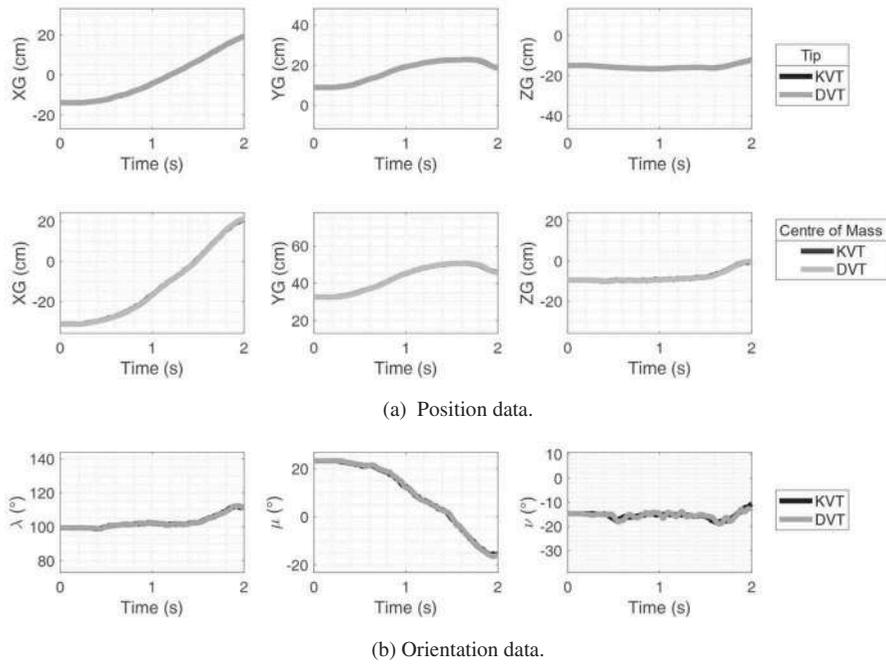


Figure 25. Best results in E_5 (Subject 3).

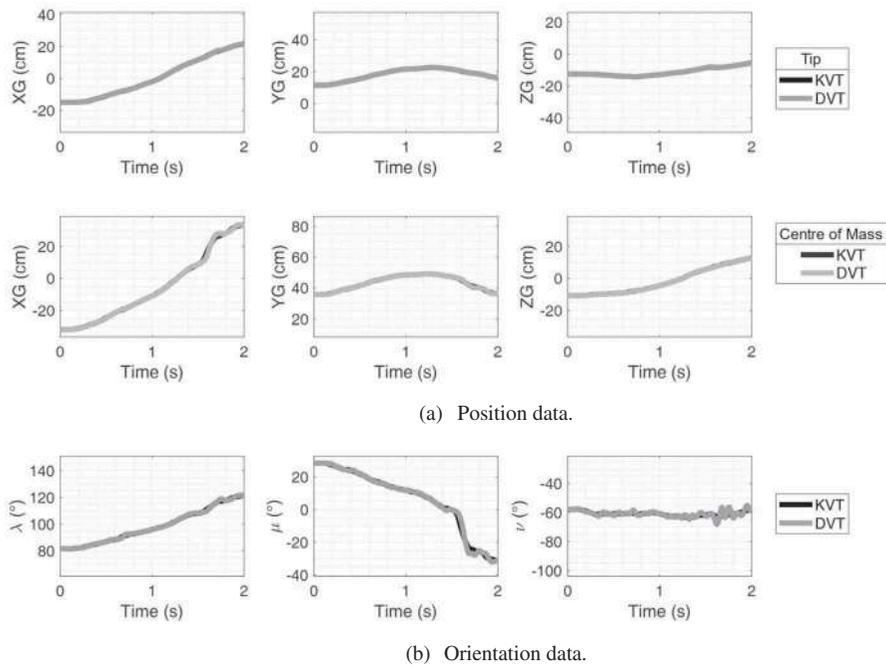


Figure 26. Worst results in E_6 (Subject 6).

-, C_2 shows better overall results in angular differences between the KVT and the DVT within the tested configurations. Therefore, since the user naturally moves the torch with a non-constant velocity, C_3 decreases the virtual coupling performance with respect to C_2 . However, by using C_3 , the minimum distance difference between both torches when the user performs

a path by manipulating the virtual torch with the HI was found.

7. Conclusions and future work

A novel and intuitive approach for the offline programming of welding robots has been presented.

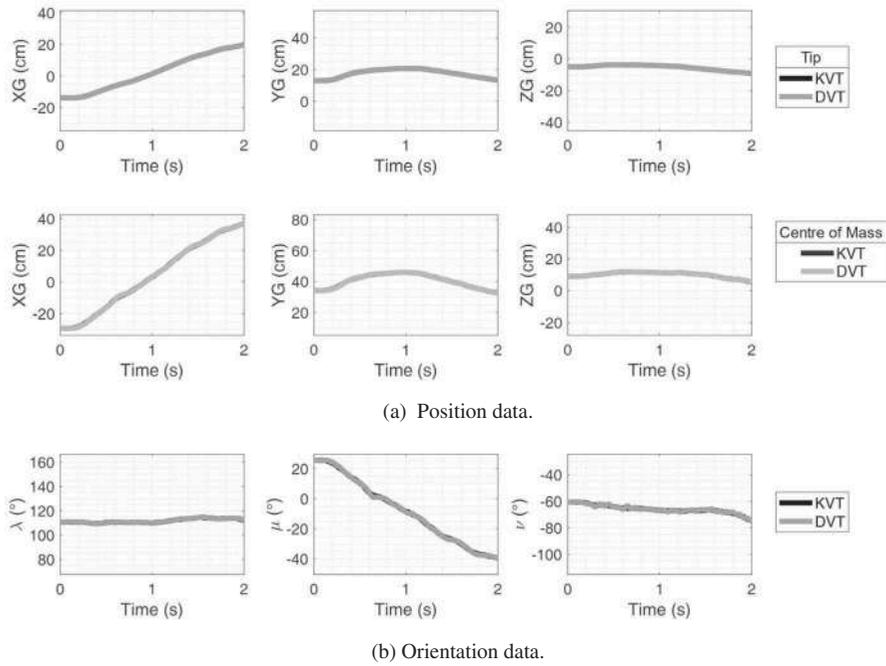


Figure 27. Best results in E_6 (Subject 10).

The user has to manipulate a welding torch through a HI to define tasks for welding robots within a VE. A technique is used in which the virtual torch is represented by two coupled models based on the spring-damper model. The difference of positions of the two models of torches allows to calculate the force to send to the HI in such a way that the hand's user feels in real time the impact of the virtual torch. So, a rectification of the manipulandum's motion should accomplish the user until the right location of the end-effector. Thanks to this approach, the feeling of force for the user makes easier the specification of the desired motion of the torch by allowing a more accurate placement of the welding points, making sure that they are placed on the plates to be welded and not 'at the air' or inside them.

The HI is used to define the set of torch poses involved in a global welding path, which is constituted of linear specific paths that the robot must follow to carry out the programmed tasks. The operational coordinates (position and orientation) of these poses will be used for solving the inverse kinematics, so the generated data are processed to obtain the complete required motions of the robot and user can visualise them in a simulation of the execution of the tasks within the VE. Mathematical tools are available in the software to compute the optimal relative placement robot/task in such a way that the accessibility to the task and the best

kinematic performance of the manipulator be gotten. Such tools were developed based on methods proposed by Pamanes-García, Cuan-Durón, and Zegloul (2008), Pámanes, Wenger, and Zapata (2002) and Zegloul and Pamanes-García (1993). Even if these tools are available in the software, its features and application are not discussed in this work because this one is focused on the application of the HI.

Even if modelling of objects is carried out by using accurate dimensions, and enough precision has to be attained in their manufacturing process, the location of the objects in the real workstation could be not exact. In such a case, the actual environment will not match with the modelled virtual world. In order to take into account, the location errors of the real objects and to obtain more precise interaction with such parts, a calibration process should be completed to relocate the virtual objects before to start the execution of the task. As a result, a more accurate task will be achieved.

In this paper, a comparative study for three SDS configurations for a virtual torch was made. Also, virtual masses in particular coordinates of the torch by depending of the configuration were attached. With all the configurations tested, the torch showed a stable behaviour during the manipulation tests. Nevertheless, experimental results showed that the torch's configurations with two or more SDS obtained

a better virtual coupling between the KVT and the DVT. Indeed, the combination of LSDS and TSDS – C_2 – and only by using LSDS – C_3 –, both facilitated this coupling, but with the use of C_3 , a better overall virtual coupling is obtained. If the torch model changed, the spring-damper constants, the position of the masses of the DVT and their values should be updated for configuring the SDS in order to find the best coupling virtual possible.

This work presented a stabilisation study for the virtual torch and a virtual coupling analysis between the KVT and the DVT, but different experiments may be interesting to carry out, as the user performance evaluation. Also, precision comparative experiments among the torch presented in this work and a torch without dynamic behaviour will also be considered.

The proposed approach can be applied in the industrial environment with some considerations. So far, only the programming of simple linear welding tasks can be programmed. Likewise, the different conditions found in the actual workstation with respect to the VE must be considered. Besides, it is necessary to define suitable arc welding parameters to get a good quality of the welded pieces. Certainly, the final quality of such pieces depends on both an appropriate welding process and the best kinematic performance of the manipulator.

Acknowledgment

This work was supported by the Autonomous University of Sinaloa, the National Technological Institute of Mexico (TNM) / La Laguna Institute of Technology, and the National Council of Science and Technology (CONACyT) of Mexico.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

- 3D Systems I. 2015. *OpenHaptics Toolkit - Programmer's Guide Version 3.3.0*, 71. Rock Hill, SC, USA: 3D Systems, Inc.
- Alba, O., A. Urquiza, A. Pámanes, J. Pámanes, M. Nieto. 2012. "Sistema de Programación Fuera de Línea Para Robots de Soldadura. Etapa de Planificación de Movimientos." *Proc. of the XVIII International Congress of SOMIM (Mexican Society of Mechanical Engineers)*, 961–970. Salamanca, México
- Almajano, P., M. L. Sanchez, I. Rodriguez, and E. Mayas. 2015. "Including Conversational Agents into Structured Hybrid 3D Virtual Environments." *IEEE Latin America Transactions* 13: 523–531. doi:10.1109/TLA.2015.7055574.
- Blankemeyer, S., R. Wiemann, L. Posniak, C. Pregizer, and A. Raatz. 2018. "Intuitive Robot Programming Using Augmented Reality." *Procedia CIRP* 76: 155–160. doi:10.1016/j.procir.2018.02.028.
- Borst, C. W., and A. P. Indugula. 2005. "IEEE Proceedings. VR 2005." *Virtual Reality, 2005*, 91–98. Bonn, Germany: IEEE. doi:10.1007/s00253-004-1818-9
- Borst, C. W., and A. P. Indugula. 2006. "A Spring Model for Whole-hand Virtual Grasping." *Presence: Teleoperators & Virtual Environments* 15: 47–61. doi:10.1162/pres.2006.15.1.47.
- Burdea, G. C. 1999. "Invited Review: The Synergy between Virtual Reality and Robotics." *IEEE Transactions on Robotics and Automation* 15: 400–410. doi:10.1109/70.768174.
- Carvalho, G., M. Siqueira, and S. Absi-Alfaro. 1998. "Off-line Programming of Flexible Welding Manufacturing Cells." *Journal of Materials Processing Technology* 78: 24–28. doi:10.1016/S0924-0136(97)00458-5.
- Choi, K.-S., S.-H. Chan, and W.-M. Pang. 2012. "Virtual Suturing Simulation Based on Commodity Physics Engine for Medical Learning." *Journal of Medical Systems* 36: 1781–1793. doi:10.1007/s10916-010-9638-1.
- Colgate, J. E., M. C. Stanley, and J. M. Brown. 1995. "Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems." *Human Robot Interaction and Cooperative Robots. Vol. 3* 140–145. Pittsburgh, PA, USA: IEEE.
- Dong, W., H. Li, and X. Teng. 2007. "Mechatronics and Automation, 2007. ICMA 2007." *International Conference on 2007*, 763–768. Harbin, China: IEEE.
- El Saddik, A. 2007. "The Potential of Haptics Technologies." *IEEE Instrumentation & Measurement Magazine*. Vol. 10. 10–17. Clifton, NJ, USA.
- Erdős, G., C. Kardos, Z. Kemény, A. Kovács, and J. Váncza. 2016. "Process Planning and Offline Programming for Robotic Remote Laser Welding Systems." *International Journal of Computer Integrated Manufacturing* 29: 1287–1306. doi:10.1080/0951192X.2015.1033753.
- Ferreira, L. A., Y. L. Figueira, I. F. Iglesias, and M. Á. Souto. 2017. "Offline CAD-based Robot Programming and Welding Parametrization of a Flexible and Adaptive Robotic Cell Using Enriched CAD/CAM System for Shipbuilding." *Procedia Manufacturing* 11: 215–223. doi:10.1016/j.promfg.2017.07.228.
- Garbaya, S., and U. Zaldivar-Colado. 2007. "The Affect of Contact Force Sensations on User Performance in Virtual Assembly Tasks." *Virtual Reality* 11: 287–299. doi:10.1007/s10055-007-0075-5.
- Gorecky, D., M. Khamis, and K. Mura. 2017. "Introduction and Establishment of Virtual Training in the Factory of the Future." *International Journal of Computer Integrated Manufacturing* 30: 182–190.
- Karabegović, I., and R. Mirza. 2018. "International Conference." *New Technologies, Development and Applications*, 3–17. Sarajevo, Bosnia and Herzegovina: Springer.

- Larkin, N., A. Short, Z. Pan, and S. van Duin. 2016. "Automatic program generation for welding robots from CAD." *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)2016*, 560–565. Banff, AB, Canada: IEEE.
- Larkin, N., A. Short, Z. Pan, and S. van Duin. 2018. "Automated Programming for Robotic Welding." In *Transactions on Intelligent Welding Manufacturing*, edited by S. Chen, Y. Shan and Z. Feng, 48–59. Singapore: Springer.
- Lozano-Quilis, J.-A., H. Gil-Gomez, J.-A. Gil-Gómez, S. Albiol-Perez, G. Palacios. 2013. *7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops2013*, 366–369. Venice, Italy: IEEE.
- Makris, S., P. Tsarouchi, D. Surdilovic, and J. Krüger. 2014. "Intuitive Dual Arm Robot Programming for Assembly Operations." *CIRP Annals* 63: 13–16. doi:10.1016/j.cirp.2014.03.017.
- Mitsi, S., K-D. Bouzakis, G. Mansour, D. Sagris, and G. Maliaris. 2005. "Off-line Programming Of an Industrial Robot for Manufacturing." *The International Journal of Advanced Manufacturing Technology* 26: 262–67.
- Mondesire, S. C., D. B. M. J. Stevens, S. Zielinski, and G. A. Martin. 2016. *Physics Engine Benchmarking in Three-Dimensional Virtual World Simulation*, 5–8. Virginia Beach, VA, USA: MODSIM World.
- Neto, P., and N. Mendes. 2013. "Direct Off-line Robot Programming via a Common CAD Package." *Robotics and Autonomous Systems* 61: 896–910. doi:10.1016/j.robot.2013.02.005.
- Pámanes, G. J., P. Wenger, and D. J. Zapata. 2002. *Motion Planning of Redundant Robotic Manipulators for Specified Trajectory Tasks*, 203–212. Dordrecht, Netherlands: Advances in Robot Kinematics Kluwer Academic Publishers.
- Pamanes-García, J., E. Cuan-Durón, and S. Zeghloul. 2008. *Single and Multi-objective Optimization of Path Placement for Redundant Robotic Manipulators*, 231–257. Ciudad de México, México: Revista INGENIERÍA Investigación y Tecnología IX.
- Pan, Z., J. Polden, N. Larkin, S. Van Duin, and J. Norrish. 2012. "Recent Progress on Programming Methods for Industrial Robots." *Robotics and Computer-integrated Manufacturing* 28: 87–94. doi:10.1016/j.rcim.2011.08.004.
- Pires, J. N., T. Godinho, and P. Ferreira. 2004. "CAD Interface for Automatic Robot Welding Programming." *Industrial Robot: An International Journal* 31: 71–76. doi:10.1108/01439910410512028.
- Sanchez-Diaz, A., J. A. Pamanes, U. Zaldivar-Colado, and X. Zaldivar-Colado. 2017. *XIX Congreso Mexicano de Robotica*. Mazatlan, Sin. Mex: AMRob.
- Sanchez-Diaz, A., U. Zaldivar-Colado, J. A. Pamanes, and X. Zaldivar-Colado. 2018. *XX Congreso Mexicano de Robotica*. Ensenada, B.C. Mex: AMRob.
- Soron, M., and I. Kalaykov. 2007. "Control & Automation, 2007. MED'07." *Mediterranean Conference on2007*, 1–5. Athens, Greece: IEEE.
- Tsarouchi, P., A. Athanasatos, S. Makris, X. Chatzigeorgiou, and G. Chryssoulouris. 2016. "High Level Robot Programming Using Body and Hand Gestures." *Procedia CIRP* 55: 1–5. doi:10.1016/j.procir.2016.09.020.
- Wu, H., H. Deng, C. Yang, Y. Guan, H. Zhang, and H. Li. 2015. "Robotics and Biomimetics (ROBIO)." *2015 IEEE International Conference on 2015*, 1711–1716. Zhuhai, China: IEEE.
- Youtube. 2019a. Accessed 15 January 2019. <https://youtu.be/DFhJV2um52s>
- Youtube. 2019b. Accessed 15 January 2019. <https://youtu.be/82eRn657Tp0>
- Zaldivar-Colado, U. 2009. *Planification d'Assemblage en Environnement Virtuel*. France: Université de Versailles Saint Quentin-en-Yvelines.
- Zaldivar-Colado, U., L. Arias, J. A. Pámanes, M. Nieto, R. Ávila-Rondón, and J. P. Nieto. 2014. "Programación Fuera de Línea de Robots de Soldadura en el Paquete WEROP Usando una Interfaz Háptica." *Chapter of the Book Avances De La Ingeniería Mecánica: Investigación Y Aplicaciones, ISBN: 978-607-96746-0-1*, edited by M. Trujillo and H. Plascencia, 786–793. *Chapter A3-182*. Querétaro, México: Sociedad Mexicana de Ingeniería Mecánica (SOMIM).
- Zaldívar-Colado, U., D. Murillo, X. Zaldívar, E. Osuna, and V. Nalda. 2009. "Interaction Technique for Virtual Robot Stabilization with Dynamic Behavior." *Congreso Internacional de Ingeniería Electrónica, Biomédica, Computación e Informática*, 335–349. Guadalajara, México.
- Zeghloul, S., and J.-A. Pamanes-Garcia. 1993. "Multi-criteria Optimal Placement of Robots in Constrained Environments." *Robotica* 11: 105–110. doi:10.1017/S0263574700019202.