



HAL
open science

Enhanced tracking wall: A real-time computing method for needle injection on haptic simulators

Ma Alamilla-Daniel, Richard Moreau, Redarce Tanneguy

► **To cite this version:**

Ma Alamilla-Daniel, Richard Moreau, Redarce Tanneguy. Enhanced tracking wall: A real-time computing method for needle injection on haptic simulators. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Oct 2020, Las Vegas, United States. 10.1109/IROS45743.2020.9341642 . hal-03068204

HAL Id: hal-03068204

<https://hal.science/hal-03068204>

Submitted on 15 Dec 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.

Enhanced tracking wall: A real-time computing method for needle injection on haptic simulators

Ma.Alamilla-Daniel¹, Richard Moreau¹ and Redarce Tanneguy¹

Abstract—Haptic simulators can help medical students to train and improve their skills before practicing with a real patient. However, the vast majority of needle insertion haptic simulators are based on sophisticated models that are accurate but highly demanding in computing resources. Most of them do not provide haptic feedback and/or are not suitable for haptic control due to their computing time. In this paper, we presented a new low computing consuming method that aims to provide a realistic needle insertion experience to the student. A description of the proposed solution is provided, and it is illustrated by experimental results to highlight its performance.

I. INTRODUCTION

Needle injection is one of the most used procedures that doctors must face day by day. In rheumatology, needle injection can be used to relieve pain in the joints, introduce drugs, and carry out biopsies. During this procedure, doctors have the possibility to use an echographic probe to follow the needle position inside the joint. The difficulty of this gesture is to handle both tools (needle and echographic probe) which requires a visual-motor coordination. To train, medical students can use corpses or manikins before practicing on real patients. However, this kind of training has several drawbacks such as:

- It is not customizable to represent different kinds of patients.
- It is not realistic enough from a haptic point of view.
- It is not repeatable.
- It does not allow to record the full gesture (position, force, etc.).

Thanks to technological advances, haptic simulators have become a way to help doctors to improve and gain medical's skills before practicing on real patients. These simulators allow working in different environments with force feedback to help immerse them in diverse situations. It also reduces the time spent on patients to complete the training and allows them to extrapolate their acquired skills to a real-life situation. However, haptic simulators still present some difficulties to simulate the different inherent stiffness of the body such as skin, fat, muscles and bones. Even though the involved forces can be calculated using biomechanical models, which are mostly based on FEM (Finite Element Method), their implementation in real-time can be limited due to the high computing resources needed to compute forces. [1].

In this paper, we introduce an enhancement of the previous "Tracking Wall" method introduced in [2]. This new method can compute the forces that act on the needle during the insertion into the tissue along the axial direction (cutting forces and attracted forces). This method is also implemented with a virtual fixture to increase the realism of the operation. We also present the results of the experimental validation of this method and our future perspectives for a more complete simulator.

II. STATE OF ART

To develop an injection simulator, the main issue is to give realistic force feedback of the tissue deformation to the final user. In the bibliography several methods and techniques are available. Most of them are based on the use of mathematical biomechanical models, which are high resource-demanding, and others only mimic in a basic way the feeling of deforming the patient's tissue.

With haptic devices, it is possible to create different virtual environments like objects/wall collision scenarios. This can be done by controlling the desired force applied by the device on the hand of the user, which is the simplest method called *direct method*. The forces and torques are generated to give the sensation of interaction by computing the distance between the tool and the surface [3]. In practice, this solution is not suitable, as chattering problems can occur on the tip of the tool due to the sudden change of applied force perturbing the haptic rendering.

One method to render forces in the virtual environment is using the *God-Object* algorithm, also known as a surface contact point, proxy point, or IHIP (Ideal Haptic Interface Point) [4], which is implemented in several haptic simulators [5]. This algorithm begins with the assumption that it is not possible to avoid that the user overpasses the virtual location of a surface in the real world. This is due to the in-exactitude of the human body to sensor small forces from the haptic device. To avoid unreal behaviour, the algorithm uses a virtual point called the God-Object, which is located at the edge of the virtual object. The forces are rendered using impedance control techniques based on the relative position of the god object against the real tool.

Another way to render the insertion forces is to implement Voigt model [6]. This method uses a virtual spring and a damper in parallel, which are very low computer consuming, but inaccurate.

The use of FEM is widely spread due to its accuracy in calculating the behavior of the tissue during puncture or deformation. One example is the method presented by

¹ All the members are with University of Lyon, INSA Lyon, CNRS Ampère, F-69621. Villeurbanne, France. richard.moreau@insa-lyon.fr

[7] where authors used a tetrahedral mesh to successfully compute the reaction forces in the needle while injecting. However, FEM is very well known for its highly demanding resources, which impedes by the moment, being applied in online estimations for haptic simulators.

In [8], authors proposed to use a statistical model based on acquired data from a measurement campaign. Their model includes parameters for several pathologies and patient's profiles. However, the model does not include data for intraarticular needle injection of the desired joint (knee, shoulder, wrist), which limits its use to our purpose.

In [9], it is proposed a model for needle insertion for surgical simulation. The authors proposed a first-order model that includes the pre-puncture, puncture and friction forces. They obtain the parameters from acquired data and compare the estimated force from their model with one obtained during its measurements. The force given by the model and the one capture resembles closely, but the problem with this method is the lack of consideration of several layers of tissue, as their parameters and model were calculated only works for a liver.

III. METHODOLOGY

In the case of needle insertion, it is necessary to know which forces act during the needle's injection and withdrawal, and how to compute them from the virtual environment to the haptic device. Considering that the needle shaft rubs against the tissue while the needle cuts the skin, three main forces are generated during the tissue piercing [10].

- 1) The cutting force, which acts on the tip of the needle in the axial direction. Its magnitude differs according to the various tissue layers and the shape of the needle tip.
- 2) The clamping force, which acts on the side of the needle shaft in the normal direction.
- 3) The friction force, which acts along the side of the needle in the axial direction.

In Fig.1, are shown the involved forces during the needle injection. In addition to these forces, another force is generated when the needle is withdrawn from the tissue, and this acts on the needle's opposite motion direction; this force is called attracted force [11] generated in the axial direction.

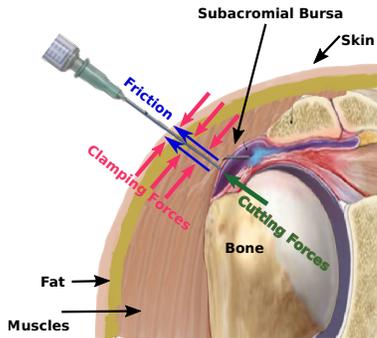


Fig. 1: Interaction forces during the needle insertion.

A. Cutting force and attracted force: Enhanced tracking wall

To reproduce the cutting force, we opted to use the Tracking Wall method (introduced in [2]) due to its low-resource consumption. This method is based on a virtual spring to compute the cutting forces. The difference with classic methods is that we avoid using any damper to dissipate the energy and substituted it with a second wall that ensures a constant force, and tracks the tool position during the displacement. On this occasion, the tracking wall method was improved to compute the cutting force during the needle insertion and the attracted force when it is withdrawn from the tissue. Also, it was modified to work from left to right (vice-versa) and from top to bottom (vice-versa) giving the possibility to set the virtual limb in any part of our workspace.

The tracking wall method helps us to achieve some objectives concerning the needle puncture, like the reproduction of different tissue stiffness, the ability to perform stops inside the virtual articulation, and the small rejection force generated by the muscles when the needle stops inside the joint.

As it is mentioned in [2], the method uses two virtual walls (Fig. 2). The first one, ω_0 is the established position of a fixed wall, when the tool collides with the skin (virtual environment). The second one, ω_n , is the position of a moving virtual wall that is updated during the tool displacement or when a stop occurs. In the case the needle is moving inside the joint, ω_n is updated by (1). This process is declared as "Update with movement" and considers if the tool enters from right to left (and vice-versa) or top to bottom (and vice-versa).

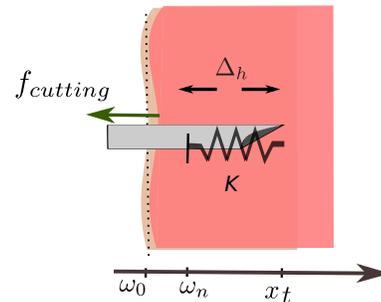


Fig. 2: Tracking wall representation.

$$\omega_n = \begin{cases} x_t - \Delta_{new} & \text{if } x_t > \omega_0 \\ x_t + \Delta_{new} & \text{if } x_t < \omega_0 \end{cases} \quad (1)$$

Where x_t is the current needle position, and Δ_{new} is a variable called "safety position", that computes the minimum distance required to reproduce the desired force, and also it represents the distance between the current point and ω_n while the tool is moving.

From this, Δ_{new} is calculated by

$$\Delta_{new} = \frac{f_d}{K_t} \quad (2)$$

Where f_d is the desired force, established by a force profile. K_t is a tuning parameter that allows establishing the value of the “safety position” and reducing the value of Δ_{new} , which should be calculated for each layer according to the desired forces.

When a stop occurs inside the virtual limb, as ω_n is behind the tool, it remains a minimum distance between the current point and ω_n which we called *difference distance* (Δ_h) and it is calculated by

$$\Delta_h = x_t - \omega_n \quad (3)$$

This distance should be reduced at each iteration to avoid having a too high rejection force when the user stops. Ideally $\Delta_h \rightarrow 0$ by the end of the last iteration, but we can also leave Δ_h with a small value, which will produce a small rejection of force that can be interpreted like the one the muscles exert on the needle. This rejection force can be set according to the properties of the involved muscles or part of the body. To update ω_n when a stop occurs, we use (4), and this process is called as “Update with stop”.

$$\omega_n = \begin{cases} m * i + \omega_{n-1} & \text{if } i \leq n \\ \omega_n & \text{if } i > n \end{cases} \quad (4)$$

where m is calculated by (5) and must remain unchanged during the rest of the process.

$$m = \frac{x_t - \omega_n}{n} \quad (5)$$

Where n is the total number of iterations to update ω_n , and i is the current number of iteration. Concerning the value n , this must be tuned according to the target hardware and programming language. It is suggested that the tracking wall method must be implemented in a real-time environment, so n can be set in seconds and designers can have a better manipulation of the wall update.

The purpose of this algorithm is to move the wall in front of the tool, so the next time the user moves the tool, he will face it again. To compute the cutting force ($f_{cutting}$) we use

$$f_{cutting} = K_t(\Delta_h) \quad (6)$$

As the attracted force is contrary to the cutting force, it is established that $f_{cutting} = -f_{attracted}$, so (6) can be used to calculate it by only applying the corresponding sign. The flow diagram on Fig.3 shows the condition established to compute the forces $f_{cutting}$ and $f_{attracted}$ depending on the tool’s direction. Only one of these forces is applied to the haptic system at each iteration.

Resuming it, the main advantages of the method tracking wall are:

- No biomechanical models are used. So, the method only requires low computing resources.
- Ability to render the different tissues of the body, by applying several successive tracking walls or by changing the desired forces in real-time
- Ability to configure the rejection force to provide realism when the needle is rejected by the tissue.

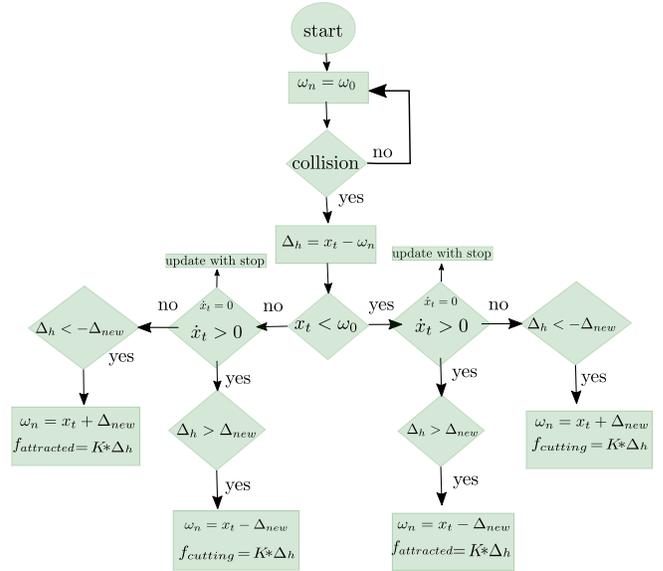


Fig. 3: Flow diagram to compute cutting force and attracted force.

B. Clamping force: Virtual fixture

As one of our objectives is to allow the doctor to choose the entry point on the virtual joint. The trajectory and the clamping forces need to be recalculated every time that the point of insertion is changed. It is thus not possible to establish a predefined trajectory for every case.

During a real injection, the muscles constraint the movement of the needle in angular and cartesian space, making difficult to change the desired trajectory. When the needle is piercing the skin, the user can keep its initial trajectory or he can change the needle’s orientation to reach the desired target. This change is limited by an angle α which allows a maximum angle of 45° concerning the needle main axis, forming a 3D cone (Fig.4). So, as the needle keeps going deeper and deeper $\alpha \rightarrow 0$, limiting the orientation until it can not be longer changed. α can be calculated by

$$\alpha = \tan^{-1}\left(\frac{n_L - d_t}{n_L + d_t}\right) \quad (7)$$

where n_L represents the needle’s length, and d_t the distance computed from the point where the needle pierced the tissue until the current point

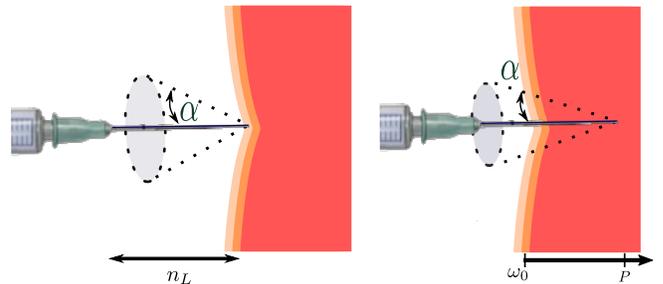


Fig. 4: Angle definition.

To limit the tilting of the needle established by α , an orientation control can be implemented on an actuated wrist.

To ensure that the needle follows the desired path once the orientation has been set, it is necessary to create a virtual guide that limits the needle's movement in the cartesian space. If the user tries to depart from this trajectory, a correctional force is exerted to keep it in. By computing the projection point onto the line, it is possible to create a virtual fixture [12].

This is made by creating a line using the insertion point of the needle (P_0), and the director vector (\vec{u}_P), which represents the *reference trajectory* and the sense of the needle. The current position of the needle is denoted by P and its projection onto the line is denoted as Q , that represents the desired point onto the line (Fig.5).

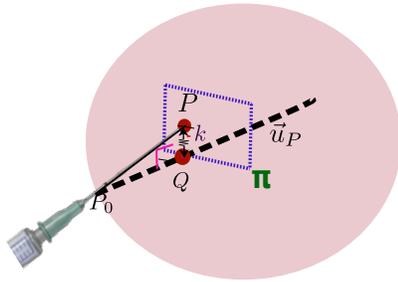


Fig. 5: Orthogonal projection of a point onto a line

To compute the projection of the point P the vector \vec{u}_P helps us to find an orthogonal plane π that fulfills the expression:

$$\pi \begin{cases} P \in \pi \\ \vec{u}_P \perp \pi \end{cases}$$

To find Q we calculate the orthogonal projection of a point onto the line. Ideally, if the difference between P and Q is minimal, no correctional force must be exerted. On the other hand, if the positions have a significant deviation, the correctional force will be exerted with a significant value to correct the needle's trajectory.

To calculate the correctional force, we apply a proportional control which helps to correct the user movement, once the tool quits from the desired trajectory (8). These forces represent the *clamping forces*.

$$f_c = k * e \quad (8)$$

where e is the euclidean distance of $\|QP\|$ and k is a tuning value that grant the restriction of the needle's movement, depending on the depth and the tissue stiffness.

IV. EXPERIMENTAL RESULTS

To test experimentally that the enhanced Tracking wall and the virtual fixture can render the haptic force from a needle injection, we make tests simulating an intraarticular injection

on a shoulder. For testing the forces, our test bench consists of a Virtuoso™ 6D Desktop haptic device, which offers up to 10N of peak force, set in a base with a syringe mock and support for the hand (Fig. 6a). To visualize the needle injection, we implement in CHAI3D the tracking wall and the virtual fixture (Fig. 6b). CHAI3D is a framework written in C++ that give us the required libraries and functions to implement a fully 3D environment and compute collisions between the needle and the virtual articulation [13].

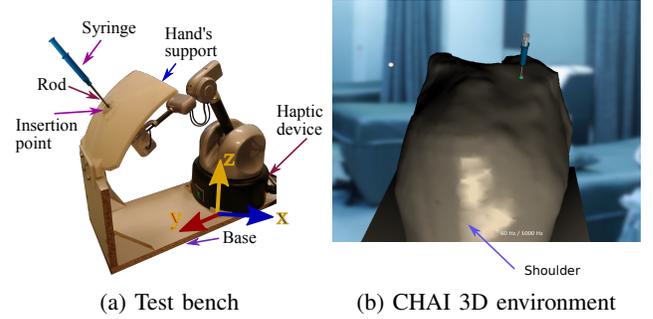


Fig. 6: Needle injection simulator

A. Cutting force and Attracted force

This section presents some tests implementing the enhanced tracking wall method to compute the cutting and the attracted forces. The objective is to corroborate that the proposed method can render the same forces that are generated in a real procedure.

For the first test, the method is tested in 1 DoF, where the objective is to compute the desired cutting force of 1N during the needle insertion, this value is chosen from the experiment performed by [14], and a desired attracted force of 0.5N is established empirically by our medical partner.

Fig. 7 shows the results in 1 DoF. In the first instance, the tool is in free motion (FM), it means that there is no contact with the virtual articulation, and hence there is no exerted force. From $t = 24s$ to $t = 27s$ the needle is inserted inside the virtual joint. ω_n begins to follow it, with a small Δ_h error to ensure the desired force. During this displacement, the cutting force increases until reaching the desired cutting force of 1N, while the speed remains negative. From $t = 27s$ to 29.5s an intentional stop occurs, and when the needle's speed is almost zero, ω_n updates its position until it reaches the needle's actual one. During ω_n updating, the force starts to decrease until zero, to avoid a too high rejection force during the stop. When the intentional stop ends, the needle begins to be withdrawn from the virtual body from $t = 29.5s$ to $t = 31.5s$. At this moment, an attracted force is computed with the chosen magnitude of 0.5N. Finally, the needle is completely withdrawn from the articulation to the free zone motion, where the user can choose another point of insertion.

The objective of the second test is to set a profile of six different cutting force values (based on the numbers of layers from an articular joint: skin, muscle, bursa, tendon, cartilage, and bone) in 3 DoF. The desired cutting forces are 1N (layer 1), 2N (layer 2), 1.5N (layer 3), 2.5N (layer 4), 3N (layer

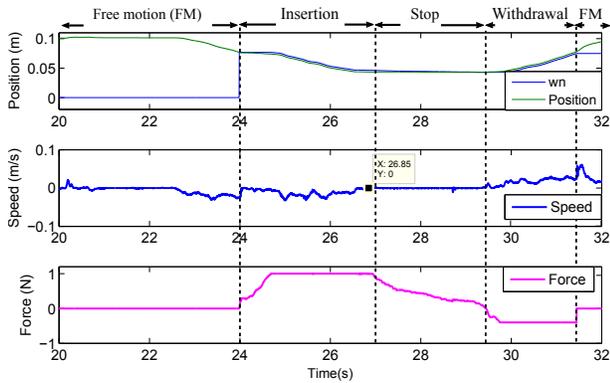


Fig. 7: Tracking wall - 1 DoF

5), 4N (layer 6), and a desired attracted force of 0.4N. The cutting forces are set based on [14] and by our medical partner.

Fig.8 presents the needle position and the tracking walls along each axis during the tool insertion and the tool withdrawal, these positions are computed respecting to the reference frame of the haptic device Fig.6a. As it can be appreciated, the wall tracks the needle displacement during the tool insertion and withdrawal, keeping a *difference distance* (between the needle tip position and ω_n) to ensure the desired forces. In this case, as it is only tuned one K_t for all the layers, it was proposed to set $K_t = 500\text{N/m}$ to reduce the *difference distance* as much as possible without perturbing the haptic response.

The cutting and attracted force are shown in Fig.9. The first force increases and decreases according to the stiffness of the layer ($t = 0\text{s}$ to $t = 0.75\text{s}$). As it is shown, the force response is almost instantaneous when this goes from 2N to 1.5N. This is due to the fact that Δ_{new} is smaller when the cutting force goes from a high force to a low one. Then when the needle is withdrawn from the virtual joint, the attracted force is generated ($t = 0.75\text{s}$ to $t = 1.5\text{s}$). In this case, the attracted is negative because it is opposite to the cutting force.

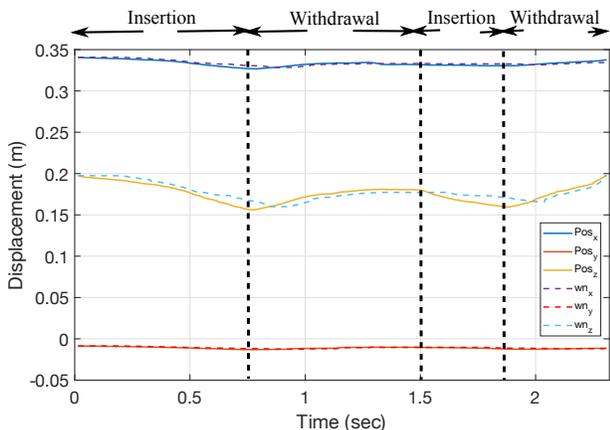


Fig. 8: Tracking wall - 3 Dof

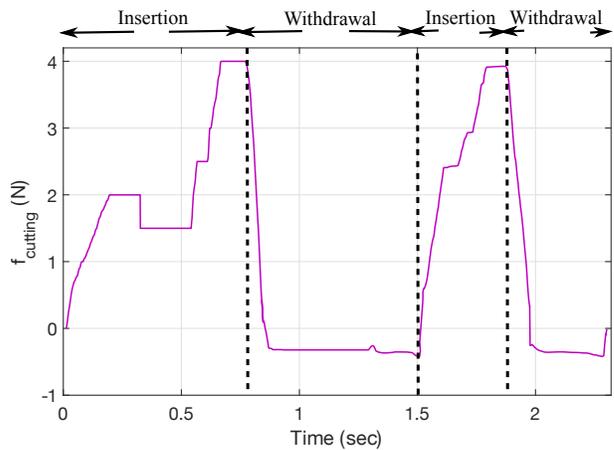


Fig. 9: Cutting forces and attracted forces - 3 Dof

B. Clamping Forces

This test is performed in 3 degrees of freedom, computing the virtual fixture previously described in section III-B.

Fig.10 shows the tool displacement (blue line) and the clamping force's behavior (pink arrows) during the needle insertion inside the virtual articulation. Fig. 11 shows the forces exerted by the simulator. In this test, the user chooses the point of insertion in the virtual environment (Start point), then he continues inserting the needle until point 1, but as he does not arrive at the desired target, the tool is withdrawn, but not totally. Following the previous path of 1, the user performed a change on the needle's trajectory at 2 and finally, he reached the desired target. After achieving the objective, the user withdraw the needle from the articulation in its totality.

Fig.10 shows that at the beginning of the test, the user almost follows the desired trajectory. Consequently, this provokes that the magnitude force is small (Fig.11). As the user keeps moving deeper, the clamping force starts to rise due to the user departing from the desired trajectory (Fig. 11). Then the user withdraw the needle to change the path and proceed to insert it again, but during the procedure, the user keeps trying to depart from the desired path, so the clamping forces are computed to correct his movements.

As can be seen, the user can perform a correctional change of trajectory limited by the maximum α angle. When this occurs the direction of the needle changes, the haptic device exerts a force preventing the user changes his trajectory, but as the forces are not large enough to stop him, the trajectory is changed. This behavior is also similar to a real injection task when a medical student tries to re-orient the needle into the tissue.

V. CONCLUSIONS

The enhanced tracking wall allows us to compute the cutting force when the needle pierces the tissue, and the attracted forces when the needle is withdrawn from it. The benefits of this method are that we avoid the use of any complex biomechanical model, and therefore the method

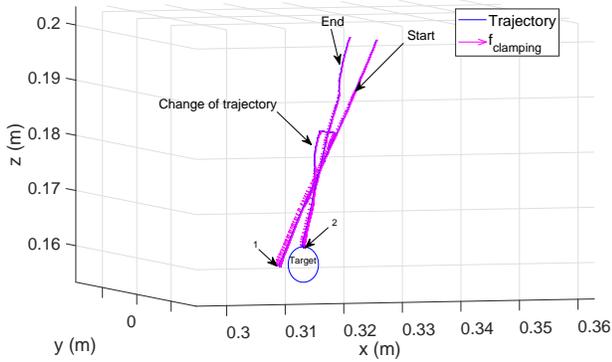


Fig. 10: Virtual fixture: representation of "Clamping forces"

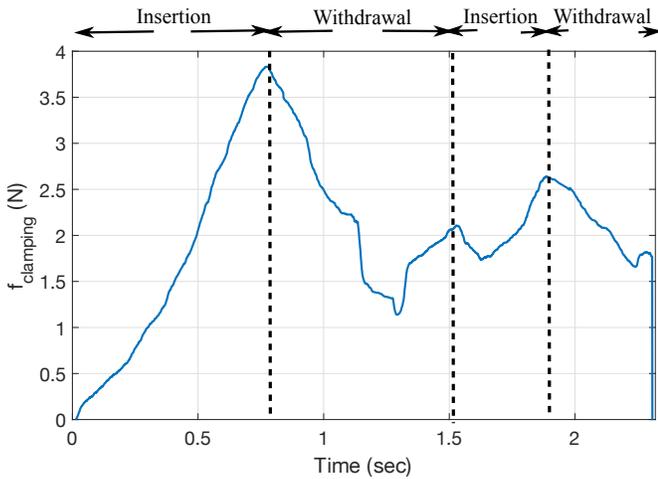


Fig. 11: "Clamping force", $|f_{clamping}|$

does not require high computing resources, allowing easy implementation on mid-range computing systems and real-time applications.

The clamping forces are computed from the projection point of the current needle's position on the desired trajectory. With this method the calculated normal forces imitates the ones that the surrounding muscles exert on the needle.

As future work, the enhanced tracking wall method will be tested by using pre-calculated values of a needle injection provided by a biomechanical model developed by one of our partners. The objective of this test is to create a more realistic simulator.

Another point to improve is the reproduction of the friction forces to increase the realism of the haptic feedback.

Finally, a campaign of measurements will be carried out in order to record medical doctors with different experiences (senior, residents, and novices). Their gestures will be analyzed in order to highlight their differences and to help novices to improve their gestures.

ACKNOWLEDGMENT

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 and we would like to thank the ANR (French National Research Agency) for financing SAMSEI project (ANR-11-IDFI-0034) under the supervision of Pr. X. Martin.

REFERENCES

- [1] Y. Kurita, H. Ohtsuka, K. Nagata, and T. Tsuji, "Haptic rendering of a needle insertion by enhancing the real force response of a base object," in *Haptics Symposium 2014, HAPTICS 2014*, pp. 357–360, 2014.
- [2] M. Alamilla D, R. Moreau, and T. Redarce, "A new method to render virtual walls for haptic systems: Tracking wall. Application to needle insertion simulation," in *ICVARS'19 Proceedings of the 2019 3rd International Conference on Virtual and Augmented Reality Simulations*, pp. 33–38, 2019.
- [3] F. Ryden and H. J. Chizeck, "A proxy method for real-time 3-DOF haptic rendering of streaming point cloud data," in *IEEE Transactions on Haptics*, vol. 6, pp. 257–267, 2013.
- [4] C. Basdogan and M. a. Srinivasan, "Haptic Rendering in Virtual Environments," in *Handbook of virtual environments: Design, implementation, and applications*, no. 1, pp. 117–134, 2002.
- [5] C. Zilles and J. Salisbury, "A constraint-based god-object method for haptic display," in *Proceedings IROS 1995*, vol. 3, pp. 146–151, 1995.
- [6] P. N. Brett, T. J. Parker, a. J. Harrison, T. a. Thomas, and a. Carr, "Simulation of resistance forces acting on surgical needles," *Proceedings of the Institution of Mechanical Engineers. Part H, Journal of engineering in medicine*, vol. 211, no. 4, pp. 335–347, 1997.
- [7] T. Yang, H. Yin, X. Zhao, J. Han, and W. Xu, "Interaction modeling and simulation of a flexible needle insertion into soft tissues," *Proceedings for the Joint Conference of ISR 2014 - 45th International Symposium on Robotics and Robotik 2014 - 8th German Conference on Robotics, ISR/ROBOTIK 2014*, pp. 611–616, 2014.
- [8] T. K. Podder, J. Sherman, E. M. Messing, D. J. Rubens, D. Fuller, J. G. Strang, R. A. Brasacchio, and Y. Yu, "Needle insertion force estimation model using procedure-specific and patient-specific criteria," *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, pp. 555–558, 2006.
- [9] H. Yang, P. X. Liu, and J. Zhang, "Modelling of needle insertion forces for surgical simulation," *IEEE International Conference on Mechatronics and Automation, ICMA 2005*, no. July, pp. 592–595, 2005.
- [10] H. Kataoka, T. Washio, K. Chinzei, K. Mizuhara, C. Simone, and A. M. Okamura, "Measurement of the Tip and Friction Force Acting on a Needle during Penetration," in *Lecture Notes and Computer Science*, pp. 216–223, 2002.
- [11] L. Barbé, B. Bayle, M. De Mathelin, and A. Gangi, "Needle insertions modelling: Identifiability and limitations," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 6, no. PART 1, pp. 129–134, 2006.
- [12] C. H. Lehmann, *Analytic Geometry*.
- [13] F. Conti, F. Barbagli, R. Balaniuk, M. Halg, C. Lu, D. Morris, L. Sentis, J. Warren, O. Khatib, and K. Salisbury, "The CHAI libraries," in *Proceedings of Eurohaptics 2003*, pp. 496–500, 2003.
- [14] A. Gordon, A. C. Barnett, I. Kim, and J. Z. Moore, "Needle insertion force model for haptic simulation," *ASME 2015 International Manufacturing Science and Engineering Conference, MSEC 2015*, vol. 2, pp. 18–20, 2015.