High Rate Mechanical Coupling of Interacting Objects in the Context of Needle Insertion Simulation With Haptic Feedback*

Claire MARTIN[1,3], Christian DURIEZ[2] and Hadrien COURTECUISSE[4,3,1¶]

Abstract

Needle-based procedures such as biopsies or radio-frequency ablation (RFA) of tumors are often considered to diagnose and treat liver cancer for their low invasiveness but raise difficulties for practitioners related to needle placement and visibility of internal anatomical structures. Efforts are being conducted to build real-time needle insertion simulators with both visual and haptic rendering, facing challenges related to model accuracy and real-time computational performance. This work focuses on the contact model involved in needle-tissue interactions in order to improve the realism of the resulting haptic rendering. We present a novel method to update the compliant coupling at high rates of a complete contact system involving the mechanics of a large object and the complete model of a flexible needle. These updates allow to adapt the contact directions to the needle deformations in the haptic thread, with the aim of improving the resulting haptic feedback. Updates of contact directions and the related mechanical system according to high-rate deformations decrease force feedback artifacts associated with low-rate mechanics while maintaining high-rate performances for the haptic loop.

1 INTRODUCTION

Liver cancer is responsible for a high number of deaths worldwide. Among the possible diagnosis and treatment methods, needle-based procedures, such as biopsies or radio-frequency ablations (RFA) of tumors, are the least invasive for patients. However, such methods are challenging due to the absence of direct visibility of internal anatomical structures. Imaging guidance can help the practitioner through the procedure, but difficulties remain in the choice of placement and orientation of the needle at the entry point to properly reach the target. The need to withdraw and insert the needle again for a better placement of the needle increases the risk of tumor cell propagation within healthy tissues and organs. Consequently, training is key to successful target reaching during a needle insertion procedure, and benefiting the patient while limiting risks and complications.

Among training methods, numerical simulators in real time offer a large variety of scenarios, with the possibility to model patient-specific anatomy, and to quantify users' performances. They are very promising, especially when coupled with a visual immersion system within the operating environment, and haptic rendering from the virtual needle insertion [1]. However, a complete training simulator faces challenges related to the realism of the numerical models, and the real-time computations requirements to allow user interaction through the haptic interface.

*This work was supported by French National Research Agency (ANR) within the project SPECULAR ANR-21-CE45

†1INRIA, MIMESIS team, Strasbourg, France

‡2INRIA, DEFROST team, Lille, France

§3ICube Laboratory, MLMS team, Strasbourg, France

¶4ICube Laboratory, RDH team, Strasbourg, France

This work focuses on the needle insertion mechanics of a training simulator with haptic feedback. Haptics is one of the main resources to gauge tissue properties in the context of needle-based procedures. The needle base is virtually attached to the haptic interface, that is manipulated by the user. We aim at providing the trainee with force of different nature (puncture, lateral forces) that influence each other, in a scenario modeling a punctured organ. Therefore, the collision model must allow for contact coupling. The need of complex deformable needle models to capture fine haptic effects for training simulators is a real challenge in the context of mechanical simulation. Modeling these interactions with high fidelity remains an active domain in haptics research.

Needle-tissue interactions involve computations of the mechanical models (volume and needle) and the forces involved in the simulation. A survey described by Corrêa et al. in [1] reports the use of both rigid and flexible needle models. Some studies consider the shape of the needle tip in the needle-tissue interactions, but this can make the simulation more complex, which impacts the simulation performances. Computational issues are more often raised by the punctured volume model for its usually higher number of degrees of freedom (DOF) and complex mechanical behavior. While the survey reported the use, in some cases, of linear models that become inaccurate with large deformations, Wu et al. in [2] considered a mass-spring system to model tissue non-linearity. For a higher accuracy, Peterlik et al. considered for the volume a co-rotational formulation of a linear tetrahedral mesh at low-rates in [3]. Their haptic thread, parallel to the simulation loop, only involved a rod modeled as a set of beams, and contact information was shared with the simulation to compute the corresponding forces.

Several force models exist in the literature to compute the haptic rendering provided to the user. While Pepley et al. considered different exponential models depending on the insertion depth in [4], Wu et al. computed a non-linear force based on the needle insertion velocity and the needle diameter [2]. Okamura et al. split needle-tissue interactions into stiffness, friction, and cutting forces in [5], each force having its own model depending on parameters such as insertion depth or velocity. This approach was followed by Alamilla et al. who consider cutting, friction, and normal forces after puncture in [6], using the LuGre model for friction, or a deviation-based strategy for the forces orthogonal to the needle shaft. These force models have proved their realism, but they do not allow for mechanical coupling in case contacts influence each other, as required in this work. To reach this goal, Peterlik et al. used compliant mechanical system in the context of haptics [3], which was also considered in [7], [8] or [9] for needle insertion simulation or robotics coupled with the FEM.

One major challenge in haptics is the need of a high update rate (above 600Hz) for a smooth rendering [10], which is not always compatible with the complexity of realistic simulation models. This requirement is highlighted in [1], although all the studies could not reach this performance, as in [2] where forces are updated at 140Hz. Pepley et al. and Alamilla et al. underlined how computing force models as theirs could help reach update rate requirements, as opposed to heavier computations related the Finite Elements (FE) method [4, 6]. However, the emergence of high-performance solvers allowed Peterlik et al. to reach 1000Hz update rates in the haptic loop with their FE models and constraint-based coupled contacts [3], although compliant contact information related to the volume model was updated at 50Hz in the simulation loop. They could therefore simulate objects at different frequencies while using haptics, but the problem was extended to several mechanical systems to build and solve, and contact linearization remained constant within the haptic loop between two updates of the low-rate simulation loop. However, updating contact directions for the needle-tissue interactions would be important for a better haptic rendering, especially if several objects are colliding, as such complex simulations tend to have a lower update rate: at the end of the simulation step, the initial contact directions provided by a contact linearization are not always relevant regarding the high-rate motions of the haptic device.

In this work, we develop a method allowing to update needle-tissue contact directions at high haptic rates, while considering the complete model of a flexible needle and a compliant mechanical coupling of the entire contact system involving a larger volume model. For this purpose, we exploit the *IsoDOFs* method [9], exposed in section 2.2, to extract contact information of the numerically heavy volume model

interacting with the needle, and to update the compliant contact system so as to better consider the high-rate motions of the haptic interface in the contact problem. As opposed to the approaches presented above, we develop a method gathering both 1) the accuracy and contact coupling from the FE method together with a constraint-based contact model exposed in section 2, and 2) the possibility to update mechanical contact coupling at high rates in the haptic loop while considering the mechanics of a large volume model. By doing so, we aim at improving haptic feedback realism in the needle insertion simulation, in a way that is fast enough to reach the high haptic update frequencies.

2 BACKGROUND

2.1 Mechanical system

Considering FE models, the needle will be simulated with a set of beam elements following the Timoshenko beam theory. Volumes will be generated from linear tetrahedral meshes coupled with a co-rotational model, formulated in [11], to describe large displacements as rotations coupled with small deformations, thus allowing to model non-linearity without using more expensive hyper-elastic models.

Each simulated object is subjected to Newton's second law, expressed by (1), where $\mathbf{p}$ and $\mathbf{v}$ respectively denote the nodes positions and velocities. $\mathbf{g}$ and $\mathcal{F}$ represent external and internal force vectors respectively, while $\mathbf{c}$ defines contact forces. We consider the quasi-static formulation of the law similarly to [3].

$$\mathbf{g} - \mathcal{F}(\mathbf{p}, \mathbf{v}) + \mathbf{c} = 0 \tag{1}$$

The non-linear internal forces are linearized at each time step with a first order Taylor development. Moreover, a backward Euler integration is performed to recover node positions at the end of the time step $\mathbf{p}_{t+h}$ from the velocities $\mathbf{v}_{t+h}$ at the same instant, with h being the time interval between two simulation updates. Readers can refer to [9] and [3] for more details. Considering these points, simulated objects therefore individually follow the mechanical behavior described by (2), where the system matrix $\mathbf{A}$ gathers stiffness and damping information, while vector $\mathbf{b}$ represents known forces from the beginning of the time step, and $\mathbf{x} = h\mathbf{v}_{t+h}$ is the unknown vector to find:

$$\mathbf{A}\mathbf{x} = \mathbf{b} + \mathbf{c} \tag{2}$$

In this paper, we model needle-tissue interactions in the context of a needle insertion simulation into a volume, involving both puncture and internal forces of different types. A compliant mechanical coupling is therefore used in order to model these global interactions. Contacts are described with a proximity formulation, where pairs of contact points are defined from the collision detection method, each point of the pair belonging to one of the two colliding objects. The collision detection method also defines constraint directions, along which constraint violations are computed. In this work, internal needle-tissue interactions are composed of bilateral constraints orthogonally to the needle shaft to prescribe lateral relative motions of the needle within the tissues.

Equations of individual objects are coupled with the contact term defining the constraint violation at the end of the time step $\boldsymbol{\delta}^{t+h}$ (eq. 3).

$$\boldsymbol{\delta}^{t+h} = \boldsymbol{\delta}^{\text{free}} + \underbrace{\sum_{object_i} \mathbf{H}_i \mathbf{A}_i^{-1} \mathbf{H}_i{}^{\mathrm{T}} \boldsymbol{\lambda}}_{\mathbf{W}} \tag{3}$$

$\boldsymbol{\delta}^{\text{free}}$ is the constraint violation after the free motion (the motion of individual objects without considering their interactions i.e. $\boldsymbol{\lambda} = 0$). Matrices $\mathbf{H}_i$ store the contact directions of each object *i*, defined after

the collision detection process, allowing to describe constraints in the DOF space; $\mathbf{W}$ is the Delassus operator that couples the contacts by describing how constraints will influence each-others. The objective of the resolution process is to find the Lagrange Multiplier $\boldsymbol{\lambda}$, modeling forces in the constraint space, that will lead to a null constraint violation vector $\boldsymbol{\delta}^{t+h}$ for bilateral constraints. Complete object motions $\mathbf{x}_i$ will then be computed with (2) by expressing forces in the DOF space $\mathbf{c}_i$ from the Lagrange Multiplier in the constraint space $\boldsymbol{\lambda}$: $\mathbf{c}_i = {\mathbf{H}_i}^{\mathrm{T}}\boldsymbol{\lambda}$. This step is called "motion correction":

$$\mathbf{x}_i = \underbrace{\mathbf{A}_i^{-1}\mathbf{b}_i}_{\mathbf{x}_i^{\text{free}}} + \underbrace{\mathbf{A}_i^{-1}{\mathbf{H}_i}^{\mathrm{T}}\boldsymbol{\lambda}}_{\mathbf{x}_i^{\text{corr}}} \tag{4}$$

2.2 The *IsoDOFs method*

The Delassus operator is expensive to compute mainly due to the inversion of matrices $\mathbf{A}_i$ which sizes are the number of DOFs of the considered models. In [9], Zeng et al. developed a method to accelerate the assembly of the Delassus operator, reformulating it according to (5):

$$\mathbf{W} = \sum_{object_i} \hat{\mathbf{H}}_i \underbrace{\bar{\mathbf{I}}_i \mathbf{A}_i^{-1} {\bar{\mathbf{I}}_i}^{\mathrm{T}}}_{\bar{\mathbf{W}}_i} {\hat{\mathbf{H}}_i}^{\mathrm{T}} \tag{5}$$

where $\bar{\mathbf{I}}_i$ is a pseudo-identity matrix that selects the DOFs impacted by at least one constraint (referred to as *IsoDOFs*), among all the DOFs of the system. $\hat{\mathbf{H}}_i$ stores the constraint directions in the *IsoDOFs* space, and is therefore a contracted version of the sparse $\mathbf{H}_i$ matrix where all the empty columns were deleted, as the associated DOFs were not impacted by any constraint (see [9] for more details).

Part of contacts generally remain during a few time steps, especially during a needle insertion where the entire needle follows a path within the tissues based on the trajectory of the needle tip. Therefore, $\bar{\mathbf{I}}_i$ slightly changes between consecutive time steps. Moreover, the *IsoDOFs* method is based on an asynchronous computation of $\mathbf{A}_i^{-1}$ developed in [12], where this inversion is considered constant in the simulation during a few time steps while a new one is being computed in a parallel thread from updated mechanics. As a result, $\bar{\mathbf{W}}_i$ is subjected to few changes between two updates of $\mathbf{A}_i^{-1}$. In addition, Zeng et al. could benefit from the particular structure of $\bar{\mathbf{I}}_i$ containing only one value 1 per row to update $\bar{\mathbf{W}}_i$ with small computations, reusing shared information between time steps. Consequently, the Delassus operator becomes a sum of products between a small matrix $\hat{\mathbf{H}}_i$ and $\bar{\mathbf{W}}_i$ that slightly changes between consecutive time steps, thus considerably increasing computational performances related to the assembly of the Delassus operator.

3 METHODOLOGY

Using flexible needles is of major importance for a high-quality haptic rendering, but raises challenges. Peterlik et al. proposed in [3] an asynchronous method where a rod is modeled at high rates. However, its high-rate deformations can generate misalignment of the low-rate constraints with respect to the rod, thus deteriorating the haptic rendering.

In this section, we describe a method aiming at updating needle-tissue contact coupling and resolution at high rates, so as to recover a finer haptic rendering based on the high-rate motions of the complete flexible needle. To do so, a low-rate simulation loop and a high-rate haptic thread are running in parallel and exchanging data so as to compute mechanics of interacting models, while computing force feedback at high rates to satisfy haptic fluidity.

The general loops organization will be presented in section 3.1, followed by technical details in 3.2, 3.3 and 3.4.

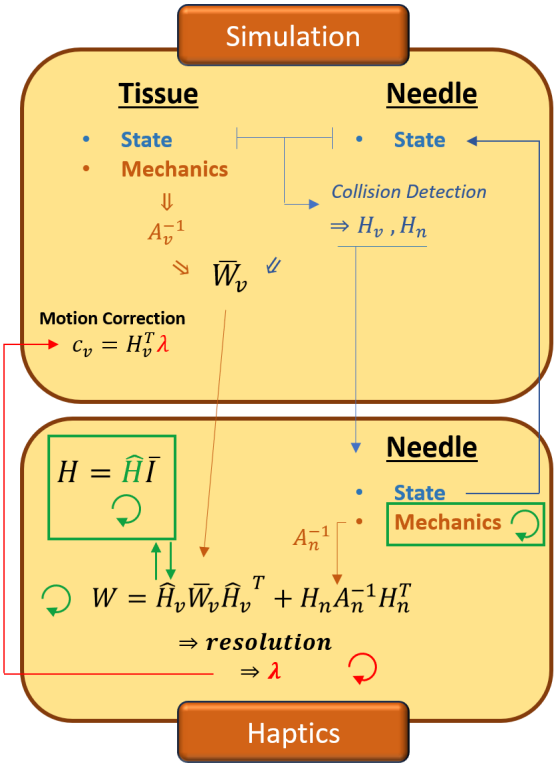

Figure 1: Organization of the simulation and haptic loops, and their communication.

3.1 Organization of the simulation and haptic loops

Following Peterlik et al.[3], our simulation and haptic threads are asynchronous. However, the volume IsoDOFs positions and the related contact coupling will also be updated at high rates. Fig. 1 shows the loops organization. The gathering of node positions and free positions will be referred to as a "state", and "mechanics" will define the behavior law of an object. Indices n and v will respectively refer to the needle and the volume.

The low-rate simulation loop gathers both the state and the mechanics of the volume, but only contains the positions of the needle retrieved from the high-rate loop at low frequency. A complete simulation cycle is divided into several steps:

Volume free motion: It is performed as described in [9].

Collision detection: Computed from the low-rate volume mesh and the latest available high-rate needle positions, it allows to retrieve the needle-tissue contact points to which two bilateral constraints will be associated.

Constraint matrices construction: Contact directions are stored in $\mathbf{H}_n$ and $\mathbf{H}_v$. From the latter matrix, the mechanics $\mathbf{A}_v$ and the *IsoDOFs* method, $\bar{\mathbf{W}}_v$ is assembled.

System resolution and motion correction: These steps will be further detailed in section 3.4.

Data shared with the haptic loop: Data sent to the haptic loop includes information required to solve the mechanical problem: matrices $\mathbf{H}_n$, $\mathbf{H}_v$, and $\bar{\mathbf{W}}_v$.

The high-rate haptic loop only contains the needle state and mechanics. Similar steps are performed:

Needle free motion: The needle mechanics are first computed, leading to $\mathbf{A}_n$ and its free motion. As opposed to tissue mechanics in the low-rate loop, $\mathbf{A}_n$ is updated and inverted at each step of the haptic loop.

The inversion exploits the particular Block-Tridiagonal structure of the needle system matrix recovered from the assembly of beam elements, connected to each other with one single node. Added to the small size of $\mathbf{A}_n$, the Thomas algorithm is used to compute $\mathbf{A}_n^{-1}$ at low costs in the haptic loop, making the use of the *IsoDOFs* method non-relevant for the needle.
Constraint matrices construction: Contact data is recovered from the low-rate loop, including $\mathbf{H}_n$, $\mathbf{H}_v$ and $\bar{\mathbf{W}}_v$.
Constraint solving: The system is then solved with (3), resulting in contact forces $\boldsymbol{\lambda}$.
Motion correction: A motion correction is applied to the needle to recover its new positions for the following haptic step, as presented in (4). The volume *IsoDOFs* positions are also updated as detailed in section 3.2.

Between two simulation steps, needle-tissue contact directions can be updated at high rates with the updated needle state (also considering the haptic interface motions by the user) and volume *IsoDOFs* positions, as exposed in sections 3.2 and 3.3, to provide a more relevant haptic rendering.

3.2 Partial updates of the heavy tissue model at high rates

Unlike the needle case, the large number of volume DOFs leads to a computationally costly motion correction step to be performed in the high-rate haptic thread. Therefore, only a subpart of the volume around the contact area is updated at high rates, by exploiting the *IsoDOFs* method.

The motion correction step applied to the complete volumetric tissue would be expressed according to (6).

$$\mathbf{x}_v^{\text{corr}} = {\mathbf{A}_v}^{-1}{\bar{\mathbf{I}}_v}^{\mathrm{T}}{\hat{\mathbf{H}}_v}^{\mathrm{T}}\lambda \tag{6}$$

To extract information related to the *IsoDOFs* only, the volume motion correction is projected onto the *IsoDOFs* space with the pseudo-identity matrix $\bar{\mathbf{I}}_v$, as shown by (7).

$$\mathbf{x}_{v,Iso}^{\text{corr}} = \underbrace{\bar{\mathbf{I}}_v{\mathbf{A}_v}^{-1}{\bar{\mathbf{I}}_v}^{\mathrm{T}}}_{\bar{\mathbf{W}}_v}{\hat{\mathbf{H}}_v}^{\mathrm{T}}\lambda \tag{7}$$

$\hat{\mathbf{H}}_v$ is a small matrix, while $\bar{\mathbf{W}}_v$ was already computed in the simulation loop and remains constant in the haptic thread during a simulation step, making $\mathbf{x}_{v,Iso}^{\text{corr}}$ fast to compute. The new positions of the contact points are recovered from the updated *IsoDOFs* positions and the barycentric coordinates of the contact points with respect to these *IsoDOFs*. This method allows to simulate at high rates the volume deformations on the *IsoDOFs* only, i.e. around the needle shaft.

3.3 Updating contact directions in the haptic thread

For each contact point, bilateral constraints are generated orthogonally to the needle trajectory (defined as the alignment of the contact points on the volume). The barycentric coordinates of the contact points within the tissue remain unchanged within a simulation step: a projection of these points onto the needle is performed in the haptic loop as an intermediate collision detection, updating their location with respect to the needle. Constraint directions must then be updated according to the volume *IsoDOFs* motion, which requires important changes in the contact problem, especially in $\mathbf{H}_i$ matrices and the resulting $\mathbf{W}$. The small size of the needle state allows for a fast computation of $\mathbf{W}_n$ from $\mathbf{A}_n^{-1}$ and $\mathbf{H}_n$. Regarding the larger volume, preserving the contact points implies keeping $\bar{\mathbf{I}}_v$, and therefore $\bar{\mathbf{W}}_v$, constant within a simulation step: only the small matrix $\hat{\mathbf{H}}_v$ is changed with new contact directions. Exploiting the *IsoDOFs* formulation of $\mathbf{W}_v$ allows to update contact directions at low cost for a large mesh. The main strength of our method is the ability to efficiently update at high rates the coupling of the simulated contacts, while considering the volume mechanics that is usually computationally costly. The constraint

violations $\boldsymbol{\delta}^{\text{free}}$ are also updated at high rates, defined as the distance between the needle and tissue contact points of a same proximity pair projected onto the contact directions.

3.4 Using the latest haptics computations in the simulation

The last steps of a standard simulation loop (constraint resolution and motion correction) are already performed in the haptic thread. Therefore, after having built $\mathbf{H}_n$, $\mathbf{H}_v$ and $\bar{\mathbf{W}}_v$ in the simulation loop, the threads are synchronized to retrieve the latest high-rate computations. A new simulation step starts with a copy of the haptic needle positions as a motion correction. The volume motion correction is computed according to (6), by using $\bar{\mathbf{I}}_v{}^{\mathrm{T}}\hat{\mathbf{H}}_v{}^{\mathrm{T}}\lambda$ from the latest high-rate computations. Therefore, the complete $\mathbf{W}$ matrix does not need to be assembled in the simulation as no resolution is performed in this loop.

4 RESULTS AND DISCUSSIONS

4.1 Profiling

Our method was evaluated using a 343-node volume and a 9-node needle. We focused on computation time related to the haptic loop only, but the efficiency of the *IsoDOFs* method to build $\mathbf{W}$ has been proved on very large meshes of about 20.000 nodes [9]. The computation time related to the main steps of the haptic loop was recorded and averaged over 1 second. The case without any constraint was first considered, and compared to a needle insertion involving 12 to 16 constraints. Table 1 shows the relatively high contribution of the computation of $\mathbf{A}_n$ when the needle and the tissue are not interacting. However, it does not prevent the loop from reaching high haptic rates. The contact problem resolution becomes the major step as needle-tissue interactions occur, but still allows to reach haptic update rates. Computations required to update constraint directions and the related mechanical problem (matrices $\mathbf{H}_v$, $\mathbf{H}_n$ and $\mathbf{W}$, and the vector $\boldsymbol{\delta}^{\text{free}}$) do not decrease the computational performances of the haptic loop.

4.2 Needle-tissue directions within the haptic loop

To highlight the necessity to adapt the trajectory constraint directions to the deformation of the needle, both a straight and a curved needle insertions were performed into a cube. Within a simulation step involving 8 contact points (with 2 orthogonal bilateral constraints each), we measured the norm of the orientation update of the first constraint normal of each contact point in the haptic loop, between the beginning and the end of the considered simulation step. Results exposed in Fig. 2 show that the order of magnitude of the changes in the constraint directions is about one hundredth of the length of the edges composing the needle. If constraint directions remain constant in the haptic loop according to the simulation collision detection, bilateral constraint forces can have a component aligned with the needle shaft, which deteriorates the resulting haptic feedback. Here, updating the contact normals allows to adapt the force directions to the local deformation of the flexible needle at high rates so as to provide the user with a more relevant haptic rendering. Fig. 2 indeed shows greater direction updates for a curved needle insertion (compared to a straight one), which may be a necessary situation in case a target must be reached while avoiding anatomical structures such as blood vessels.

4.3 Force Measurement

Improvements of the force feedback resulting from our method is exposed in this section. We performed several needle insertions into a cube with partial withdrawals, while curving the needle with a recorded (thus reproducible) motion of the haptic interface. We measured the force rendering provided to the interface by the haptic loop, and expose in Fig. 3 the Y component of the measured force. The same

Table 1: Computation time of the main steps of the haptic loop averaged over 1 second, with and without constraints.

		Average / Max time per call (ms)	
	Level	No constraints	12 to 16 constraints
Total step	0	0.16 / 0.33	0.39 / 0.97
Free Motion	2	0.09 / 0.15	0.06 / 0.15
$\mathbf{A}_n^{-1}$	2	0.02 / 0.03	0.01 / 0.03
Update constraint pos	3	0 / 0	0 / 0
Update constraint dir	3	0 / 0	0 / 0.03
Update $\mathbf{H}_n$, $\mathbf{H}_v$	3	0 / 0.01	0 / 0.01
Update $\boldsymbol{\delta}^{\text{free}}$	3	0 / 0	0 / 0
Compute $\mathbf{W}_n$, $\mathbf{W}_v$	3	0 / 0.03	0.04 / 0.12
Solve system	4	0.02 / 0.12	0.22 / 0.58
Motion correction	5	0.01 / 0.04	0.01 / 0.03
Force feedback	5	0.01 / 0.01	0.01/ 0.02

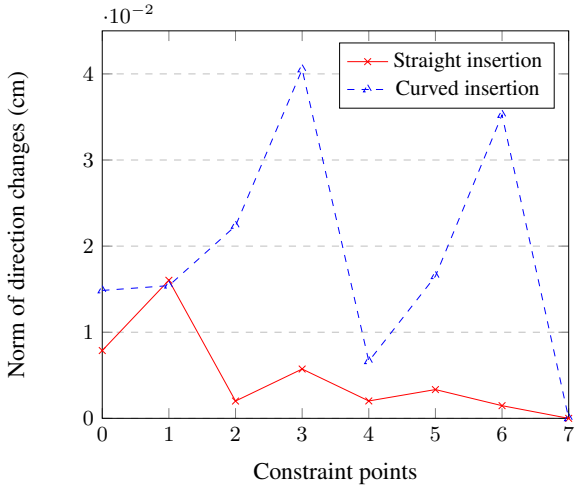

Figure 2: Norm of constraint direction changes in the haptic loop between two simulation steps, for both a straight and a curved needle insertion with 8 contact points.

motion and measurements were performed with and without updating the constraint directions between two simulation steps. When constraint directions are preserved from the collision detection, Fig. 3 shows peaks in the resulting force. These peaks correspond to sudden changes in the force direction when constraint points jump from a needle edge to the following one. Indeed, constraint points are fixed on the needle trajectory, and only move according to the deformations of the volume. Consequently, constraints

slide along the needle shaft as the needle is progressing within the tissue. However, if constraints are not updated in the haptic loop, contact points may be located on a different edge between two simulation steps. As the edges of the needle are straight, a bent needle results in different directions between consecutive edges, and lead to force peaks as seen in Fig. 3. Updating at haptic rates the location of the contact points on the needle and the resulting directions according to intermediate states of the needle allows to smoothen the contact direction changes between two simulation steps. The resulting force profile is considerably improved as the force peaks, resulting from the needle geometry, are eliminated, which generates a higher quality haptic rendering.

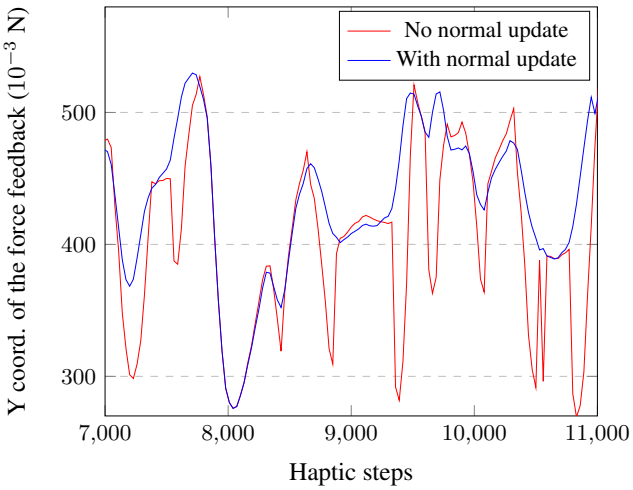

Figure 3: Y dir. of the force feedback during a needle insertion, with and without contact direction updates in the haptic loop.

4.4 Impact of the volume complexity on the haptic loop

Eventually, we illustrate the impact of the volume complexity on the haptic loop. The latter mainly depends on the number of constraints and *IsoDOFs*. For needle-tissue contact, constraints should not be too numerous so as to avoid ill-conditioned problems regarding the needle geometry. On the other hand, increasing the size of the volume mesh could generate more *IsoDOFs*, also impacting the simulation and synchronization steps.

The previous recorded needle motion was used several times with a volume of different mesh sizes. We evaluated the number of *IsoDOFs*, the computation time of the main haptic steps involving the number of *IsoDOFs*, and that of the synchronization step, shown in table 2. We plotted in Fig. 4 the same Y-direction force rendering as in section 4.3 for two of the tested cases. The *IsoDOFs*-related computational cost does not significantly increase in the haptic loop, allowing to maintain high update rates. The synchronization step is more impacted by the volume complexity, while being performed at low simulation rates only. With a fine volume mesh, the haptic rendering is still smoothed by contact direction updates, thus deleting the needle geometry sensation.

Table 2: Average computation time (over 1 sec.) of the synchronization and the main volume-related high-rate steps, depending on the number of volume nodes.

# nodes	**1000**	**1728**	**2744**	**4096**	**5832**
# *IsoDOFs*	36	42	51	54	63
$\mathbf{W}_v$ **(ms)**	0.02	0.04	0.04	0.03	0.04
$\mathbf{x}_{v,Iso}^{\text{corr}}$ **(10^{-3} ms)**	1	3	4	5	7
Synchronization (ms)	0.09	0.07	0.09	0.10	0.13

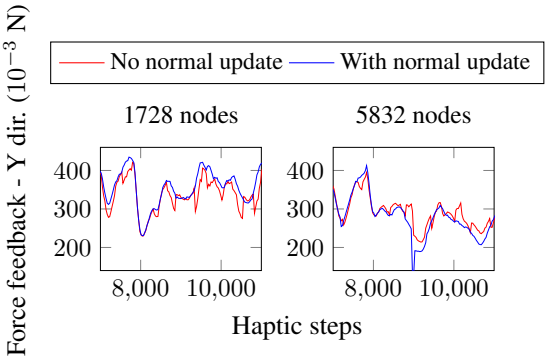

Figure 4: Y dir. of the force feedback during a needle insertion, with and without contact direction updates in the haptic loop, for a volume of 1728 (left) and 5832 (right) nodes.

5 CONCLUSION

In this paper, we present a novel method to improve haptic feedback of a needle insertion simulation for teaching purposes. The accuracy required for a training tool imposes the use of a flexible needle, entirely simulated at high haptic rates. With the help of the *IsoDOFs* method, we simulate part of the punctured volume involved in the needle-tissue contact at high frequency. An intermediate collision detection allows to update the location of contact points on the needle based on volume updates. Constraint directions are then updated, and quickly considered in the contact system thanks to the *IsoDOFs* method. The resulting force feedback is therefore more finely computed while considering the coupling of a contact system within the high update rates required for a smooth haptic rendering. Although this work focused on the theory behind the structure of the loops, future work consists in handling more complex contacts with friction, and including this method into a large scale simulation involving multiple colliding organs at low rates (while still being considered in the high-rate contact coupling system). New resolution methods could be interesting to handle a greater contact system, with organ interactions.

References

[1] C. G. Corrêa, F. L. Nunes, E. Ranzini, R. Nakamura, and R. Tori, "Haptic interaction for needle insertion training in medical applications: The state-of-the-art," *Medical Engineering and Physics*, vol. 63, pp. 6–25, 1 2019.

[2] H. Wu, C. Chen, Y. Zhou, J. Wang, and Y. Xie, "VR-based haptic simulation for dynamic needle insertion," in *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2019, pp. 924–929.

[3] I. Peterlik, C. Duriez, and S. Cotin, "Asynchronous haptic simulation of contacting deformable objects with variable stiffness," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2608–2613, 2013. [Online]. Available: http://hal.inria.fr/hal-00823762

[4] D. F. Pepley, M. A. Yovanoff, K. A. Mirkin, S. R. Miller, D. C. Han, and J. Z. Moore, "Integrating cadaver needle forces into a haptic robotic simulator," *Journal of Medical Devices, Transactions of the ASME*, vol. 12, 3 2018.

[5] A. M. Okamura, C. Simone, and M. D. O'Leary, "Force modeling for needle insertion into soft tissue," *IEEE Transactions on Biomedical Engineering*, vol. 51, pp. 1707–1716, 10 2004.

[6] M. A. Alamilla, C. Barnouin, R. Moreau, F. Zara, F. Jaillet, H. T. Redarce, and F. Coury, "A virtual reality and haptic simulator for ultrasound-guided needle insertion," *IEEE Transactions on Medical Robotics and Bionics*, vol. 4, pp. 634–645, 8 2022.

[7] C. Duriez, C. Guébert, M. Marchal, S. S. S. Cotin, and L. Grisoni, "Interactive simulation of flexible needle insertions based on constraint models," *Lecture Notes in Computer Science*, vol. 5762 LNCS, pp. 291–299, 2009. [Online]. Available: http://link.springer.com/10.1007/978-3-642-04271-3_36

[8] P. Baksic, H. Courtecuisse, C. Duriez, and B. Bayle, "Robotic needle insertion in moving soft tissues using constraint-based inverse finite element simulation," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2407–2413, 2020.

[9] Z. Zeng, S. Cotin, and H. Courtecuisse, "Real-time FE simulation for large-scale problems using precondition-based contact resolution and isolated DOFs constraints," *Computer Graphics Forum*, vol. 41, pp. 418–434, 9 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1111/cgf.14563https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.14563https://onlinelibrary.wiley.com/doi/10.1111/cgf.14563

[10] F. G. Hamza-Lup, K. Bergeron, and D. Newton, "Haptic systems in user interfaces – state of the art survey." Association for Computing Machinery, Inc, 4 2019, pp. 141–148.

[11] C. Felippa, "A systematic approach to the element-independent corotational dynamics of finite elements," *Center for Aerospace Structures*, 2000. [Online]. Available: http://www.colorado.edu/engineering/cas/Felippa.d/FelippaHome.d/Publications.d/Report.CU-CAS-00-03.pdf

[12] H. Courtecuisse, J. Allard, C. Duriez, and S. Cotin, "Preconditioner-based contact response and application to cataract surgery," *Medical Image Computing and Computer Assisted Intervention - MICCAI*, vol. 6891 LNCS, pp. 315–322, 2011.