Available online at www.sciencedirect.com
ScienceDirect
Computerized Medical Imaging & Graphics 00 (2015) 1–15
Journal Logo

Preoperative Trajectory Planning for Percutaneous Procedures in Deformable Environments

Noura Hamzé[a,d,*], Igor Peterlík[b,d], Stéphane Cotin[c,d], Caroline Essert[a,d]

[a] *Université de Strasbourg - ICube, 300 boulevard S. Brant, 67412 Illkirch, France*
[b] *Institute of Computer Science, Masaryk University, Czech Republic*
[c] *Inria, 1 Place de l'Hôpital, 67000 Strasbourg, France*
[d] *Institut Hospitalo-Universitaire, 1 Place de l'Hôpital, 67000 Strasbourg, France*

Abstract

In image-guided percutaneous interventions, a precise planning of the needle path is a key factor to a successful intervention. In this paper we propose a novel method for computing a patient-specific optimal path for such interventions, accounting for both the deformation of the needle and soft tissues due to the insertion of the needle in the body. To achieve this objective, we propose an optimization method for estimating preoperatively a curved trajectory allowing to reach a target even in the case of tissue motion and needle bending. Needle insertions are simulated and regarded as evaluations of the objective function by the iterative planning process. In order to test the planning algorithm, it is coupled with a fast needle insertion simulation involving a flexible needle model and soft tissue finite element modeling, and experimented on the use-case of thermal ablation of liver tumors. Our algorithm has been successfully tested on twelve datasets of patient-specific geometries. Fast convergence to the actual optimal solution has been shown. This method is designed to be adapted to a wide range of percutaneous interventions.

© 2015 Published by Elsevier Ltd.

Keywords:
Constraint solving, optimization, trajectory planning, flexible needles, biomechanics, deformable models, Finite Element Method (FEM), interventional radiology, percutaneous procedures, radiofrequency ablation, cryoablation.

1. Introduction

Today, percutaneous procedures have become a common alternative to open surgery, particularly for the treatment of abdominal tumors. However, the preoperative planning, that aims at determining a secure and efficient needle path before the intervention, remains one of the main challenges of this therapy.

Computer assistance for the preoperative planning of this kind of surgery remains limited. Even if nowadays the visualization can be enhanced by volume rendering or 3D reconstruction based on CT (Computerized Tomography) or MRI (Magnetic Resonance Imaging) data acquired preoperatively, the surgeon still has to choose a needle path by mentally assessing possible trajectories inside the anatomy.

Nonetheless, the planning based on the static preoperative data does not take into account the deformations of the tissue which can significantly impact the accuracy and relevance of the planned trajectory, since both the target lesions and the obstacles such as vessels can move (see Fig. 1). In the case of the abdominal organs, there are two sources of deformations: the interaction between the needle and the tissue and the respiratory motion.

In Section 2, we present the motivations for this work as well as modeling hypotheses. After presenting the related works in Section 3, we detail in Section 4 the new algorithm *Haystack* (HST) that we propose to include a simulation of the deformations due to needle insertion within the automatic trajectory planning process, and the deformable models we used. In Section 5, we report the evaluation of *Haystack* in the context of hepatic radiofrequency ablation, with an assessment using 12 datasets of patient-specific geometries. We present a comparison between our algorithm and two reference algorithms (exhaustive method and Nelder-Mead downhill simplex method)

*Corresponding author
Email addresses: n.hamze@unistra.fr (Noura Hamzé), peterlik@ics.muni.cz (Igor Peterlík), stephane.cotin@inria.fr (Stéphane Cotin), essert@unistra.fr (Caroline Essert)

to evaluate the efficiency and accuracy of *Haystack*. The paper ends with a discussion on the results and the benefit of considering the possible deformations preoperatively, and a conclusion.

2. Motivations and problem statement

In [1], the survey of the experiments aiming at quantifying phenomena related to the needle–tissue interaction reports that forces applied to the tissue due to the interaction with the needle can exceed 1 N. For example, Hing *et al.* [2] present force profiles observed during 45 insertions of a brachytherapy needle into a porcine liver: cutting and friction forces of 1.5 N have been measured. At the same time, the liver is a highly deformable object with low stiffness as shown by [3, 4]). Therefore, the forces applied due to the interaction with the needle result in important deformations, mainly in the areas close to the needle trajectory. While these deformations are ignored by previous planning algorithms [5, 6, 7], they are usually considered an important issue that must be addressed directly during the intraoperative needle control and navigation. Some works describe different techniques to mitigate the impact of the needle interaction: for example Mahvash *et al.* propose increasing the insertion velocity to minimize the tissue deformation mainly during the puncture of the organ surface [8], while Kobayashi introduces an intelligent robotic manipulator driven by an imaging modality capturing the actual deformation of the tissue [9].

As for the deformations due to the respiratory motion, Rohlfing *et al.* quantify the motion of liver due to breathing [10]: after removing the rigid-motion component, a mean residual deformation of 10 mm has been reported. Nevertheless, in clinical routine, the needle insertion is done under shallow (i. e. quiet) breathing; in this case, Korin reports deformations which do not exceed 3 mm [11]. Similar observations are reported in [12] where gating techniques are considered. Finally, it can also be supposed that the inserted needle plays a role of an additional mechanical constraint further reducing the respiration-induced displacement of the internal structures of the organ.

In this work, we do not consider any deformation due to the breathing for two reasons: first, as described in the literature dealing with experimental assessment, the "periodical" component of the respiratory motion can be significantly reduced. Second, the "irregular" motions for example due to coughing cannot be taken into account preoperatively and must be addressed intraoperatively. In the general case, optimal needle placement can be achieved by a combination of accurate preoperative planning and intraoperative control. While there is still need for advanced intraoperative navigation and control, we show that by modeling the needle–tissue interactions and deformation we can improve the preoperative planning, thus reducing requirements for intraoperative control.

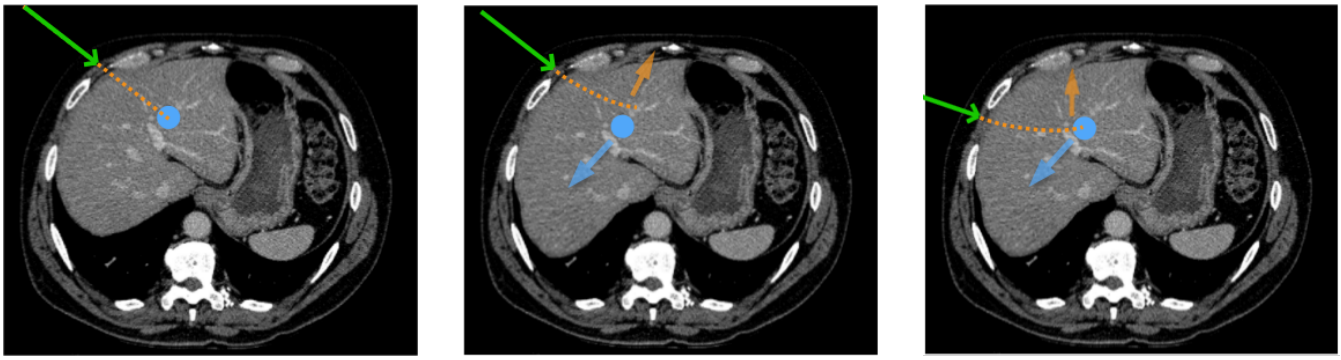
The objective is to demonstrate the feasibility of including simulations of deformations in an automatic preoperative planning algorithm in a reasonable time and with an acceptable accuracy.

We emphasize that an important aspect of this work is that it is generic. Firstly, the planning method we present here can be applied to an arbitrary percutaneous procedure involving deformable tissues such as liver, spleen, or kidney tissues. Secondly, we employ a simulation engine to demonstrate the robustness of the planning algorithm, but it could be replaced with any other similar simulator.

3. Related works and contributions

Much significant work has been performed in both fields: that of the preoperative planning of needle path (more particularly for percutaneous thermal ablations), and the flexible needle insertion in soft tissues.

As for the preoperative trajectory planning, most of the works in the literature relate to percutaneous thermal ablations and address the issue of optimizing the volume of the ablated tissue, i.e. destroying all the tumor while preserving most of the healthy tissue. Some authors approximate the ablation volume by ellipsoids: Butz [13] uses Powell's optimization algorithm to improve the placement of a manually positioned needle. Other works focus on the accurate estimation of the ablation volume: Altrogge [14], Chen [15], and Haase [16] simulate heat propagation in the tissues using finite element methods but without considering the feasibility and safety of the path itself. Wang [17] proposes a mathematical model based on geometric optimization to optimize the needle puncture placements and the ablation frequencies for treating large liver tumors with multiple needles. However, none of these methods accounts for any other surgical constraint in their path planning. For instance, the avoidance of certain surrounding anatomical structures that must not be damaged by the needle during the insertion, or the inclusion of a portion of healthy liver in the trajectory for a better cauterization are also important aspects of the planning to ensure the safety of the procedure. These aspects have been less studied in terms of computer-assisted path planning. The automatic method proposed by Seitel *et al.* [18] is capable of finding the best compromise between multiple clinical criteria. The optimization is computed on polygonal surfaces of patient models. In a similar way, Schumann [19] has investigated a fast automatic path proposal directly based on segmentation masks. Other works propose non-automatic (interactive) path planning: a user interface that helps the surgeon to define his intervention path manually has been introduced by März [20], and Seitel *et al.* [18] from the same group have also proposed a manual planning interface. Another work implemented by Stoll [5] and derived from radiotherapy planning methods combines automatic and interactive planning. Since 2003, we have proposed an approach [6, 7, 21] of preoperative planning based on a geometric solver using the Nelder Mead optimization algorithm to automatically find a trajectory (insertion point, angle and depth) optimizing an ellipsoid representing the ablation volume. This method also takes into account multiple other surgical rules to provide a safe and efficient plan. Like Seitel *et al.*, our approach has been developed under the form of a fully operational software with a GUI offering the possibility to browse a number of candidate optimal trajectories. Both works are quite close and differ mostly on the



(a) (b) (c)

Figure 1. (a) Trajectory (insertion point and angle, represented by the green arrow) planned within a static scene may actually lead to a deformed needle path (dotted line) that deviates from its ideal straight path because of the deformability of the tissues and needle itself. It can hit an obstacle or even (b) miss the target. (c) insertion point and angle able to anticipate the deformations of the needle and organs, to reach the target in optimal conditions.

optimization algorithm and the interaction and display philosophy. Another strength of our approach is also its generality, which allows to use our solver for other surgical tool placement problems [22]: for instance, it has been adapted and tested for a neurosurgery procedure [23].

Nevertheless, all the above mentioned work assume that both the tissues and needle are static, which is a significant oversimplification given the highly deformable environment and currently used needles.

Needle insertion planning in deformable environments has been studied by DiMaio [24] using potential fields attraction to the goal and repulsion from obstacles. Flexible needles have been considered by Park in [25] where a method of stochastic planning is introduced. In [26], Alterovitz has proposed a method of planning of flexible needle insertion in a two-dimensional environment with obstacles. Since uncertainty is added in each step of the needle insertion, the planning is regarded as a Markov decision process. Both needle insertion planning and control are studied in [27]: the tissue is represented by a highly-deformable 2D elastic model. However, to the best of our knowledge, an optimization-based planning method for flexible needles in soft tissue models, and in three-dimensions has yet to be proposed. In [28], Misra has presented a detailed study of flexible needles showing a complex model of the needle based on the Timoshenko formulation. However, possibly large deformation of the surrounding tissue is not considered. A physically-based model for both the surrounding tissue and flexible needle is employed by Duriez in [29] where both components are modelled with Finite Elements Method (FEM) and by Goksel in [30] where the needle is simulated with angular springs. Nevertheless, neither of these works employ the physical model of needle and tissue in the context of needle trajectory planning. Recently, preliminary results of needle insertion planning in prostate phantoms are available in [31]: the accuracy of needle insertion simulation is evaluated experimentally. Nevertheless, the model is not employed in the patient-specific scenario.

The main contribution of this paper is an optimization-based preoperative planning algorithm for percutaneous procedures: unlike other approaches, it accounts for non-straight trajectories occurring due to the flexibility of the needle and deformability of the punctured tissues, and computes an insertion point and angle that mark the origin of an optimal insertion trajectory inside the organ undergoing an elastic deformation. It extends an existing geometric solver dedicated to finding the optimal paths using static preoperative images [6, 7]. The planning algorithm is coupled with a biomechanical simulation engine [32] which, for a given location and orientation of the needle entry point, computes the deformation of the soft tissues and needle during the insertion process. The simulation engine we use employs constraint-based finite element (FE) elastic formulation which can be parametrized by patient-specific data. The biomechanical simulator is used to model possible deformations as a result of puncture, cutting and friction between the needle and the tissue, on a patient-specific geometry, with generic values of the mechanical parameters as reported in the literature. The simulated insertion can be regarded as an evaluation of the objective function optimized by the iterative planning process. The new optimization algorithm in deformable environment and the deformable models are detailed in the next section.

4. Methods

4.1. Trajectory planning: definitions and initial algorithm

First, let us recall briefly in this section the approach we have previously developed for computing preoperatively an optimal needle placement in static conditions. We start with a few definitions and explain the main principle of the planning algorithm we use, which is described in more detail in [7], and represented by the blue part of the pipeline in Fig.2.

In this paper, we give the term *planning* to the automatic definition of optimal *insertion settings* (IS) for the needle before the intervention which are a triplet composed of an insertion point P represented by a vector of coordinates $\mathbf{x}$, a direction $\mathbf{v}$ represented by a pair of angles (α,β), and a depth d. These parameters constitute the strategy the surgeon determines preoperatively during the planning phase, to use as starting indications to insert the needle towards the target. A *trajectory* T is

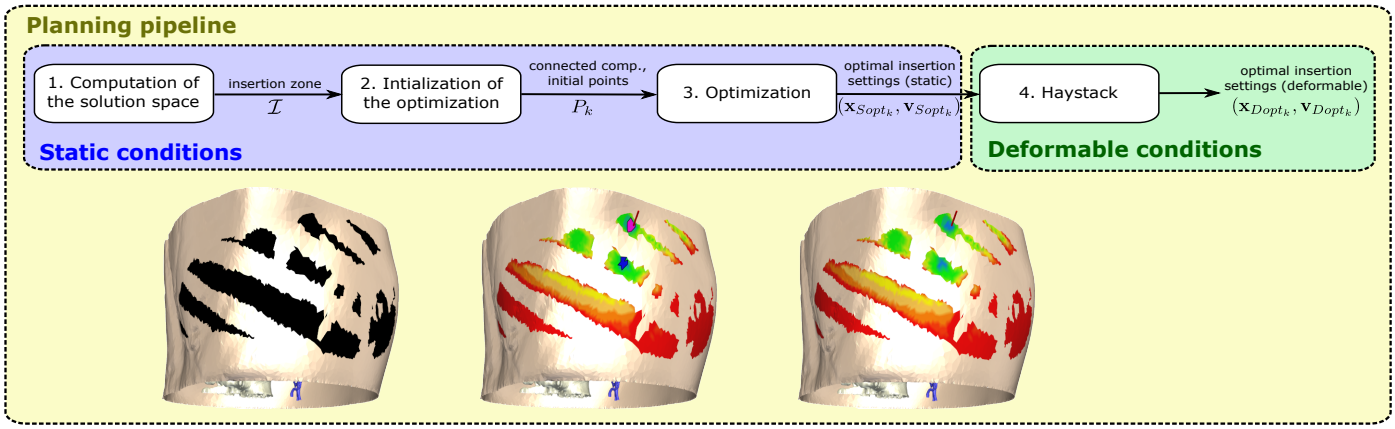


Figure 2. The different phases of planning in the static case: Phase 1 the insertion zone $\mathcal{I}$ (black) is delineated; Phase 2, the color map shows the best locations of $\mathcal{I}$ in blue, medium in a gradient from green to orange, and worst in red, and two connected components CC_1 and CC_2 (blue and pink) are delineated on $\mathcal{I}$ representing the most interesting areas; Phase 3, an optimal IS is computed for each CC_k and is represented as a red cylinder (here we can see the optimal IS for the pink connected component).

then defined by the geometry of the needle inside the body after its insertion when following a given set IS.

In static condition, where both the tissues and needle are assumed to be static, the trajectory is supposed to be straight and follow exactly the insertion direction $\mathbf{v}$ from the insertion point P. In this case, d can be easily deduced after P and $\mathbf{v}$ are selected. If the objective is simply to reach the center of the target, for instance for biopsy, d can be computed as the distance between P and the center of mass of the target. In more complex procedures, such as thermal ablations of tumors, d can be chosen as the depth minimizing the volume of damaged healthy tissue while still ensuring a full coverage of the tumor. This can be represented by a quadratic function with a single minimum as we showed in [7]. The objective of our method is then to automatically find an optimal pair $IS_{Sopt}(P_{Sopt}, \mathbf{v}_{Sopt})$ leading to an optimal trajectory T_{Sopt} to reach the target (here S stands for static). In this paper, we do not address the tracking or the control of the needle during the insertion.

An *obstacle* is defined as an anatomical structure which should not be crossed by the trajectory of the needle. We define *strict constraints* as rules that a trajectory must absolutely satisfy: obstacle avoidance, maximal length allowed for the path. When used for the planning of thermal ablation, ensuring a complete coverage of the tumor and safety margin by the resulting effect is also a strict constraint. A *feasible trajectory* is a safe trajectory which goes from an insertion point on the skin to the lesion volume and satisfies all the strict constraints. *Feasible insertion settings* are a pair $(P, \mathbf{v})$ that leads to a feasible trajectory.

We define *soft constraints* as rules to optimize at best: keeping the trajectory as far as possible from the risky anatomical structures, and minimizing the insertion depth. When used for the planning of thermal ablation, an extra soft constraint is added: minimizing the damage to healthy cells. Each soft constraint c is associated with a cost function $f_c : \mathbb{R}^5 \rightarrow [0, 1]$ which expresses the corresponding constraint as a numerical value to minimize. An *optimal trajectory* is the feasible trajectory which minimizes an objective function $f : \mathbb{R}^5 \rightarrow [0, 1]$ representing a compromise between all soft constraints. To each f_c is assigned a weighting factor w_c describing the importance given to constraint c with respect to the others. Then f is expressed by the weighted sum:

$$f = \frac{\sum_{c=1}^{n} w_c . f_c}{\sum_{c=1}^{n} w_c} \quad (1)$$

As an example, in this work, where we tested our algorithms on the planning of thermal ablation, f is composed of a weighted sum of three cost functions f_1 to f_3 corresponding respectively to the following soft constraints:

1. maximize the distance between the path and the vessels
2. maximize the distance between the path and the ribs
3. minimize the volume of ablation to preserve a maximum of healthy tissue while ensuring a complete coverage of the tumor by the resulting effect

When the constraint expresses a maximization problem, we formulate the function such that the problem is translated into a minimization problem. The method described in this paper could be applied to any other choice of soft constraints. For further details on the cost functions of the above soft constraints, we refer the reader to [7].

The weighting factors are set after discussion with medical experts, and it is possible to change them interactively in the software. For any trajectory T, the value of $f(T)$ expresses the quality of T: a lower $f(T)$ means a better quality of T. Note that in [18] Seitel *et al.* have used a different method based on the computation of the Pareto front which is an interesting alternative to the use of weighting factors which does not favor some constraints relatively to the others. Another recent Pareto-based optimization method for RFA has been presented by Schumann *et al.* [33]. Since their planning simulates heat distribution inside necrosis, they compute only a small number of well chosen

Pareto-optima paths in order to keep the computational costs in practical range. However, we chose to offer the surgeons the possibility to set the relative importance of the different rules based on their preferences and experience.

The purpose of our method is to automatically find $\mathbf{x}_{Sopt}$ and $\mathbf{v}_{Sopt}$ such that the resulting trajectory T_{Sopt} is feasible and $f(\mathbf{x}_{Sopt}, \mathbf{v}_{Sopt})$ is minimal. Our solver uses an optimization process in three phases. Basically, phase 1 ensures the feasibility of the trajectory and phase 3 minimizes $f(\mathbf{x}_{Sopt}, \mathbf{v}_{Sopt})$. The three phases are described below.

1. **Computation of the solution space**. In this phase, the strict constraints are solved to keep only the *solution space* $\mathcal{S}$ containing all the feasible insertion settings $IS(P, \mathbf{v})$ leading to feasible trajectories. The solution space is stored (and displayed) as an *insertion zone* $\mathcal{I}$, which is a subset of the surface of the skin representing the *feasible insertion points* (intersections between the feasible trajectories and the skin). An illustration of $\mathcal{I}$ is shown in Fig.2 (left).

2. **Initialization of the optimization**. In this phase, a rough estimation of objective function f is performed on a discretization $\mathcal{D}$ of $\mathcal{I}$ ($\mathcal{D} \subset \mathcal{I}$). The result is shown as a color map on $\mathcal{I}$, where the points are colored using a gradient from blue for the best point P_{min} (with a minimal evaluation of f) to red for the worst point P_{max} (with a maximal evaluation of f). Then, we extract m connected components CC_k from $\mathcal{I}$, containing only insertion points $P_{CC} \in \mathcal{I}$ such that $f(P_{CC}) < f(P_{min}) + \varepsilon_1$. Here ε_1 is a small threshold computed so that we obtain several (2-5) small-sized connected components. In other words, we choose the m most interesting areas of $\mathcal{I}$. An illustration of the color maps and the connected components is given in Fig.2 (middle).

3. **Optimization**. In this last phase, we start with m initial candidates IS_k in which points P_k are located at the barycenter of each CC_k. Then we apply a local optimization algorithm (Nelder–Mead [34]) to refine $\mathbf{x}$ and $\mathbf{v}$, in order to obtain one optimal pair $(\mathbf{x}_{Sopt_k}, \mathbf{v}_{Sopt_k})$ minimizing f per connected component k. Each pair represents a locally optimal set of insertion settings IS_{Sopt_k} leading to an optimal trajectory T_{Sopt_k}. An example of an optimal IS_{Sopt_k} is shown in Fig.2 (right). If only the best trajectory of all is requested, we keep the one minimizing f as T_{Sopt}. Let us recall that in this static case, a trajectory is straight and is assumed to follow exactly the initial direction.

   The optimization algorithm for deformable environments that we propose in the next section constitues a fourth phase in the process, represented by the green part of the pipeline in Fig.2. It is a variant of phase 3 described above, and is performed after the three phases in static conditions are achieved.

4.2. Trajectory planning in deformable environment

4.2.1. Planning requirements

Planning in static conditions has some limitations when using flexible needles. It is based on the analysis of preoperative images with the assumption that the rigid needle will follow a straight path to the target, and the soft tissues will not deform. However, the flexible needle bends during the insertion and follows a curved path, and the insertion causes the tissues to deform as well. Our approach in dynamic conditions anticipates the deformations by making biomechanical simulations of the insertion process, and injecting this valuable information into the optimization phase of the solver to perform a more realistic planning.

To estimate the deformations, an efficient biomechanical simulation module has been developed using the Sofa platform [35]. It is described in Sections 4.3 and 4.4.

Optimization algorithms usually require a large number of iterations to converge. A single simulation of the insertion process in a deformable environment requires a few seconds in our simulation module. To include estimations of deformations in the planning, a simulation has to be performed at each iteration of the optimization. Therefore, the choice of a classic optimization algorithm would not allow for an overall process to be performed in a reasonable time compatible with clinical routine. We have designed a dedicated optimization algorithm capable of converging in a relatively low number of iterations to reduce the optimization global time as much as possible.

Our search algorithm is called *Haystack* (inspired from the well-known "needle in a haystack" problem used to refer to something that is difficult to locate in a large space) and is described in the following section and Algorithm 1. The principle is to guide the new propositions of insertion settings at each iteration by tracking the movement of the target.

4.2.2. Haystack algorithm

Haystack algorithm starts after an initial optimization in a static environment E_S(as illustrated by the bounding lime rectangle in Fig. 4). Thanks to our previous approach, a first optimization is performed using the three phases 1, 2 and 3 described in Section 4.1, and optimal $\mathbf{x}_{Sopt}$ and $\mathbf{v}_{Sopt}$ leading to an optimal theoretical straight trajectory T_{Sopt} are calculated for the geometrical model G_S which is a surface mesh model reconstructed from the segmented images of the patient. However, in a deformable environment E_D, using $\mathbf{x}_{Sopt}$ and $\mathbf{v}_{Sopt}$ as insertion settings would lead to the deformation of G_S into a deformed geometry G_D and to a deformed trajectory T_D which could be curved and either miss the target or hit an obstacle.

The inputs of *Haystack* algorithm are $\mathbf{x}_0 = \mathbf{x}_{Sopt}$ and $\mathbf{v}_0 = \mathbf{v}_{Sopt}$, and the initial geometry G_S used as an initial model in the simulation module. At each iteration i, new candidate $\mathbf{x}_i$ and $\mathbf{v}_i$ are sent by the solver to the simulation module which computes the needle and tissue motions. The simulation module sends back the corresponding deformed anatomy G_{Di} and curved trajectory T_{Di}. The principle is to iteratively modify $\mathbf{x}_i$ and $\mathbf{v}_i$ by following the target movement until converging to $\mathbf{x}_{Dopt}$ and $\mathbf{v}_{Dopt}$ resulting in a deformed trajectory T_{Dopt} optimizing the soft constraints within E_D while being part of the

Algorithm 1: *Haystack* algorithm

Data: $IS_{Sopt}(\mathbf{x}_{Sopt}, \mathbf{v}_{Sopt})$: Optimal insertion settings in static environment E_S
G_S: Initial 3D geometry of patient's anatomy
Result: $IS_{Dopt}(\mathbf{x}_{Dopt}, \mathbf{v}_{Dopt})$: Optimal insertion settings in deformable environment E_D

$(T_D, G_D, C_D) \leftarrow simulation(\mathbf{x}_{Sopt}, \mathbf{v}_{Sopt})$;
$dist \leftarrow compute_distance(T_D, C_D)$;
$p \leftarrow 1$;
$i \leftarrow 0$;
while *((*$i < iter_{max}$*)* ***and*** *(*$tip_{T_D} \notin target_{G_D}$*))* **do**
  $(\mathbf{x}_{Tr}, \mathbf{v}_{Rot}) \leftarrow compute_displacements(\mathbf{x}_i, \mathbf{v}_i, p)$;
  $(T_{Di_1}, G_{Di_1}, C_{Di_1}) \leftarrow simulation(\mathbf{x}_{Tr}, \mathbf{v}_i)$;
  $(T_{Di_2}, G_{Di_2}, C_{Di_2}) \leftarrow simulation(\mathbf{x}_i, \mathbf{v}_{Rot})$;
  $dist_1 = distance_min(T_{Di_1}, C_{Di_1})$;
  $dist_2 = distance_min(T_{Di_2}, C_{Di_2})$;
  if *((*$dist_1 < dist$*)* ***or*** *(*$dist_2 < dist$*))* **then**
    $(\mathbf{x}_i, \mathbf{v}_i) \leftarrow keep_candidate_with_min_dist((\mathbf{x}_{Tr}, \mathbf{v}_i), (\mathbf{x}_i, \mathbf{v}_{Rot}))$;
    $dist \leftarrow min_distance(dist_1, dist_2)$;
  else
    $p \leftarrow modify_p(T_{Di_1}, T_{Di_2})$;
  $i \leftarrow i + 1$;
$p \leftarrow 0.1$;
$eval \leftarrow evaluate(T_S, G_S)$;
while *((*$i < iter_{max}$*)* ***and*** *(*$eval_{best} - eval < \varepsilon_2$*))* **do**
  $(\mathbf{x}_{Tr}, \mathbf{v}_{Rot}) \leftarrow compute_displacements(\mathbf{x}_i, \mathbf{v}_i, p)$;
  $(T_{Di_1}, G_{Di_1}, C_{Di_1}) \leftarrow simulation(\mathbf{x}_{Tr}, \mathbf{v}_i)$;
  $(T_{Di_2}, G_{Di_2}, C_{Di_2}) \leftarrow simulation(\mathbf{x}_i, \mathbf{v}_{Rot})$;
  $eval_1 = evaluate(T_{Di_1}, G_{Di_1})$;
  $eval_2 = evaluate(T_{Di_2}, G_{Di_2})$;
  if *((*$eval_1 < eval$*)* ***or*** *(*$eval_2 < eval$*))* **then**
    $(\mathbf{x}_i, \mathbf{v}_i) \leftarrow keep_candidate_with_min_eval((\mathbf{x}_{Tr}, \mathbf{v}_i), (\mathbf{x}_i, \mathbf{v}_{Rot}))$;
    $eval \leftarrow min_evaluation(eval_1, eval_2)$;
  else
    $p \leftarrow modify_p(T_{Di_1}, T_{Di_2})$;
  $i \leftarrow i + 1$;
$(\mathbf{x}_{Dopt}, \mathbf{v}_{Dopt}) \leftarrow (\mathbf{x}_i, \mathbf{v}_i)$

feasible deformed paths. So, at each iteration the objective function f needs to be evaluated for the curved trajectory T_{Di} on the deformed anatomy G_{Di} and the strict constraints need to be checked. Fig. 8 illustrates a deformed model G_{Di} superimposed on an initial static model G_S, and Fig. 3c shows a typical deformed trajectory.

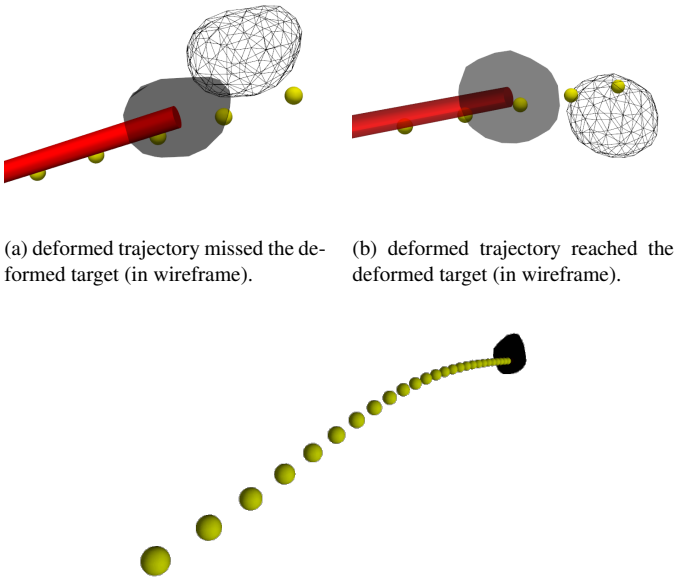
When receiving a deformed model G_{Di}, the *Haystack* algorithm proposes the next candidates depending on the situation:

1. **Target tracking phase**: if T_{Di} could not reach the target (Fig. 3a), then the objective is to reach it; in this case, we iterate over translations of the insertion point or rotations of the insertion direction according to the distance or angle between T_{Di} and the target in the deformed model to compensate the movement, until the target is reached.

2. **Refinement phase**: when T_{Di} could reach the target (Fig. 3b), then the objective is to optimize the soft constraints; in this case, we iterate over small translations of the insertion point or rotations of the insertion direction in order to stay within the target while optimizing the soft constraints in the deformed model. We stop when the improvement of the evaluation falls under a threshold ε_2, i.e. $f(T_{Di-1}) - f(T_{Di}) < \varepsilon_2$ or the number of simulations reached a maximum $iter_{max}$. Then T_{Di} is considered as the optimal trajectory T_{Dopt}.

It can happen that T_{Di} does not satisfy the strict constraints, *i.e.* is not a feasible deformed trajectory. For instance, T_{Di} may hit a vessel. In this case, T_{Di} is discarded by adding a penalty to $f(T_{Di})$ and the iterations continue. This is done within the *evaluate* function of the algorithm.

The interest of distinguishing these two phases is to reduce the number of iterations as much as possible by allowing



(a) deformed trajectory missed the deformed target (in wireframe).

(b) deformed trajectory reached the deformed target (in wireframe).

(c) a typical deformed trajectory.

Figure 3. Illustration of different scenarios after needle bending. The red cylinder represents the theoretical straight trajectory; the dotted path represents the deformed trajectory. The gray shape represents the initial position of the target, and the shape in wireframe represents the position of the target after the simulation.

to reach the target faster before refining precisely the trajectory. However, in both phases we consider two possible ways of moving — translations or rotations — because depending on the configuration of the surrounding anatomical structures, they can converge and/or be evaluated differently. At new attempt $i+1$, the *Haystack* algorithm proposes two new candidate pairs $(\mathbf{x}_{Tr}, \mathbf{v}_i)$ and $(\mathbf{x}_i, \mathbf{v}_{Rot})$ corresponding respectively to a translation and a rotation of the previously proposed solution $(\mathbf{x}_i, \mathbf{v}_i)$, and leading respectively to deformed geometries G_{Di_1} and G_{Di_2} and curved trajectories T_{Di_1} and T_{Di_2} (as illustrated in Fig.5). These propositions are issued thanks to function *compute_displacements*:

- Insertion point P_{Tr} represented by the vector of coordinates $\mathbf{x}_{Tr}$ is computed with the following formula:

$$\mathbf{x}_{Tr} = \mathbf{x}_i + p(\mathbf{x}_{C_{Di}} - \mathbf{x}_{E_{Di}})$$

  where C_{Di} of coordinates $\mathbf{x}_{C_{Di}}$ is the center of the target; E_{Di} of coordinates $\mathbf{x}_{E_{Di}}$ is the point of the deformed trajectory T_{Di} which is the closest to C_{Di}; p is a parameter allowing a progressive movement towards the target. Fig.5b illustrates this case and the proposition of a new candidate insertion point P_{Tr}.

- $\mathbf{v}_{Rot}$ is computed similarly. If we denote θ the oriented angle between points $\widehat{E_{Di}P_iC_{Di}}$, then we can obtain $\mathbf{v}_{Rot}$ by rotating $\mathbf{v}_i$ by angle θ weighted by p for a progressive movement. Fig.5c illustrates this case and the proposition of a new candidate insertion direction $\mathbf{v}_{Rot}$.

Factor p is chosen according to the phase of convergence. If we are at the target tracking phase (1), p is initialized to 1, whereas when in refinement phase (2) p is initialized to 0.1. Whatever the phase, due to the inhomogeneity of the tissues, we can not ensure that translating the insertion point by a vector $\vec{v}$ or rotating the insertion direction by an angle θ will lead to a placement of the tip of the deformed needle at the exact expected position. This is why, in the case where using the proposed translation or rotation does not lead to any improvement, we propose more attempts within a search range δ around the theoretical movement by modifying p until an improvement is found. If no improvement is found within δ, we extend δ to explore a larger solution space. This modification is represented by function *modify_p* in Algorithm 1.

The planning software and simulation module were designed to work separately and communicate using the *Boost* serialization library [36], as shown in Fig. 4. This figure illustrates the process of proposing new candidate insertion points and directions, simulating the insertions of the needle and the deformations of the anatomical model using the proposed candidates, and looping according to the quality of the resulting deformed trajectory. In the following section, we explain in detail the methods involved in the simulation unit.

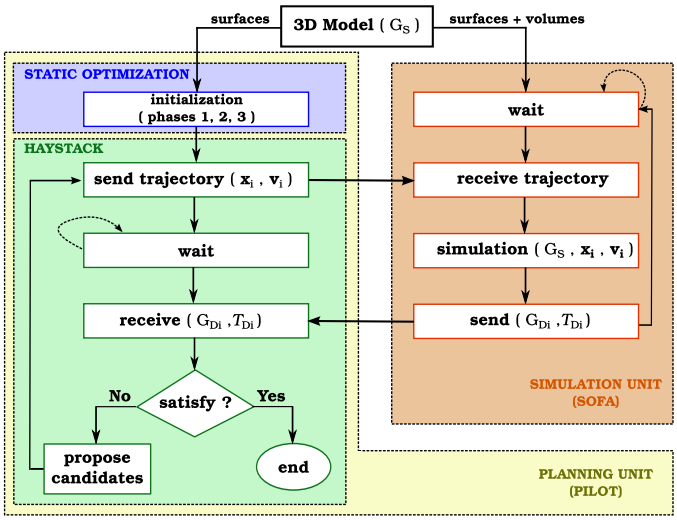


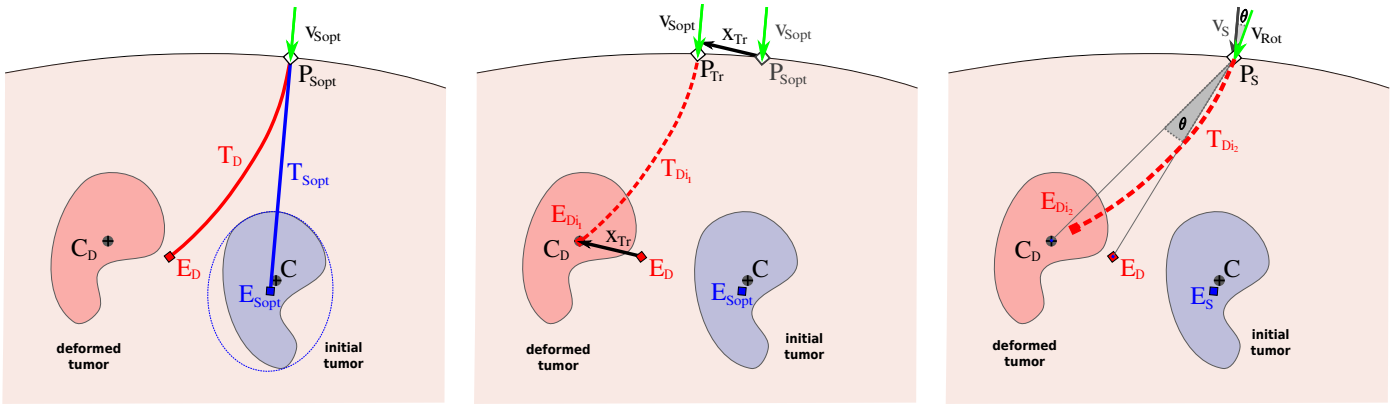
Figure 4. Communication pipeline between the planning and simulation units.

4.3. Soft tissue and flexible needle models

Both the tissues and the needle are regarded as dynamic deformable objects which are subjects to numerical modeling based on finite element method. Mathematically, a dynamic system is represented by

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{H}^\top \lambda \quad (2)$$

where $\ddot{\mathbf{q}}, \mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}$ are respectively the acceleration, position, velocity and force vectors and $\mathbf{M}$ stands for the mass matrix. Since a *constrained dynamics* is considered as explained below, vector λ stands for Lagrange multipliers (representing the



(a) using P_S and $\mathbf{v}_S$, T_D would not reach the deformed target.

(b) a new entry point P_{Tr} is proposed based on the movement of $\mathbf{C}$.

(c) a new direction $\mathbf{v}_{Rot}$ is proposed based on angle θ ($\mathbf{E}_D\widehat{P_S\,\mathbf{C}_D}$).

Figure 5. Illustration of the new propositions of: (b) insertion point P_{Tr}, and (c) direction $\mathbf{v}_{Rot}$ in the neighborhood of the previous candidate P_{Sopt} and $\mathbf{v}_{Sopt}$ leading to T_{Sopt} (blue line), in case the deformed trajectory T_D (red line) would not reach the target in the deformed model $\mathbf{G}_D$ (see (a)). On the three images, the blue shapes are the target in the initial model $\mathbf{G}_S$, and the red shapes are the target in the deformed model $\mathbf{G}_D$. The red dotted lines in (b) and (c) are the deformed trajectories T_{Di_1} and T_{Di_2} issued by the two propositions.

unknown response forces) and $\mathbf{H}^T$ provides the direction of the constraints. It should be emphasized that in this section (unlike the previous ones), by *constraint* we mean a physical coupling between two deformable objects (e. g. tissue and needle) which is a mathematical representation of mechanical interactions.

The temporal integration is based on the implicit backward Euler scheme [37]:

$$\dot{\mathbf{q}}^{t+h} = \dot{\mathbf{q}}^{t} + h\ddot{\mathbf{q}}^{t+h} \tag{3}$$

$$\mathbf{q}^{t+h} = \mathbf{q}^{t} + h\dot{\mathbf{q}}^{t+h} \tag{4}$$

where acceleration $\ddot{\mathbf{q}}^{t+h}$ is computed as the solution of a non-linear system

$$\mathbf{M}\ddot{\mathbf{q}}^{t+h} = \mathbf{f}(\mathbf{q}^{t+h}, \dot{\mathbf{q}}^{t+h}) + \mathbf{H}^T\lambda. \tag{5}$$

In Eq. 5, $\mathbf{f}$ is most generally a non-linear function that represents the sum of internal and external forces (nevertheless, the forces applied due to mechanical interactions are not included in $\mathbf{f}$). Therefore, the first order approximation (a single linearization per time step) is applied based on Taylor series expansion:

$$\mathbf{f}(\mathbf{q}^{t+h}, \dot{\mathbf{q}}^{t+h}) \approx \mathbf{f}(\mathbf{q}^{t}, \dot{\mathbf{q}}^{t}) + \mathbf{K}(\mathbf{q}^{t+h} - \mathbf{q}^{t}) + \mathbf{B}(\dot{\mathbf{q}}^{t+h} - \dot{\mathbf{q}}^{t}) \tag{6}$$

where $\mathbf{K}$ is the *stiffness* matrix and $\mathbf{B}$ the *damping* matrix.

In both tissue and needle, the computation of the stiffness matrix is based on a corotational FE formulation which allows for large displacements [38]. In the case of the tissue, the global stiffness matrix $\mathbf{K}_t$ is assembled from local element stiffness matrices $\mathbf{K}_{te}$:

$$\mathbf{K}_{te} = \mathbf{R}_{te}^{\top}\{\int_{V_{te}} \mathbf{C}_{te}^{\top}\mathbf{D}_{te}\mathbf{C}_{te}dV\}\mathbf{R}_{te} \tag{7}$$

where for each tissue element te: $\mathbf{C}_{te}$ is the strain-displacement matrix, $\mathbf{D}_{te}$ is the stress-strain matrix and $\mathbf{R}_{te}$ is a rotation matrix derived in the co-rotational formulation. While both $\mathbf{B}_{te}$ and $\mathbf{D}_{te}$ are constant during the simulation, $\mathbf{R}_{te}$ must be updated in each step. In our simulations, the domain of tissues is discretized with a mesh composed of P1 tetrahedral elements, each having four nodes with three degrees of freedom (DoF). Therefore, $\mathbf{K}_{te}$ is a 12×12 matrix organized in 3×3 blocks. Further details about the computation of each component can be found in [39].

Similarly, the needle stiffness matrix $\mathbf{K}_n$ is assembled from local matrices $\mathbf{K}_{ne}$:

$$\mathbf{K}_{ne} = \mathbf{R}_{ne}^{\top}\{\int_{V_{ne}} \mathbf{C}_{ne}^{\top}\mathbf{D}_{ne}\mathbf{C}_{ne}dV\}\mathbf{R}_{ne} \tag{8}$$

where $\mathbf{R}_{ne}$, $\mathbf{C}_{ne}$ and $\mathbf{D}_{ne}$ are respectively rotational, strain-displacement and stress-strain matrices computed for each needle element ne. The needle model is based on Timoshenko beam formulation which accounts for positional as well as rotational degrees of freedom. Therefore, each needle element is composed of two nodes, each having 6 DoFs. Thus, $\mathbf{K}_{ne}$ is a 12×12 matrix composed of two 6×6 blocks located on the main diagonal.

Finally, for both needle and tissue, the damping matrix $\mathbf{B}$ in Eq. 6 is approximated using *Rayleigh damping* computed as $\mathbf{B} = r_M\mathbf{M} + r_K\mathbf{K}$ where r_M and r_K are Rayleigh mass and Rayleigh stiffness coefficients.

Substituting (3) and (6) into (5) provides the final linearized system:

$$\underbrace{\left(\mathbf{M} - h\mathbf{B} - h^2\mathbf{K}\right)}_{\mathbf{A}}\mathbf{d}\dot{\mathbf{q}} = \underbrace{h\mathbf{f}(\mathbf{q}^{t}, \dot{\mathbf{q}}^{t}) + h^2\mathbf{K}\dot{\mathbf{q}}^{t}}_{\mathbf{b}} + \mathbf{H}^T\lambda \tag{9}$$

where $\mathbf{d}\dot{\mathbf{q}} = h\ddot{\mathbf{q}} = \dot{\mathbf{q}}^{t+h} - \dot{\mathbf{q}}^{t}$.

4.4. Needle–Tissue Interaction

Two different types of constraints must be considered for correct modeling of needle–tissue interaction [40] as the needle is

inserted from the outside of the tissue towards its interior. First, before entering the tissue, the needle tip is in contact with the surface of the tissue. Mathematically, this interaction is represented by a unilateral constraint $\Psi(\mathbf{q}_n, \mathbf{q}_t) \geq 0$. As soon as the needle enters the tissue, bilateral sliding constraints are created along the trajectory of the needle: $\Phi(\mathbf{q}_n, \mathbf{q}_t) = 0$. Both Ψ and Φ depend on actual positions of both needle and tissue nodes ($\mathbf{q}_n$ and $\mathbf{q}_t$, respectively) and can be non-linear, e. g. when friction is considered. Moreover, they depend on additional parameters such as coefficient of friction and force thresholds needed to puncture the tissue surface and to advance the needle inside the tissue.

To impose these constraints to the behavior of both needle and tissue, we use Lagrange multipliers and a single linearization per time step. A new value of the multipliers needs λ is to be computed during each simulation step. However, for interaction including deformations, there is often a temporal coherency on the multipliers values. Thus, we can provide an estimate $\tilde{\lambda}$ at the beginning of each time step and compute a correction $\Delta\lambda$ so that $\lambda = \tilde{\lambda} + \Delta\lambda$. For both interacting components, Eq. 9 is then replaced by:

$$\begin{aligned} \mathbf{A}_t \mathbf{d}\dot{\mathbf{q}}_t &= \mathbf{b}_t + \mathbf{H}_t^T(\tilde{\lambda} + \Delta\lambda) \\ \mathbf{A}_n \mathbf{d}\dot{\mathbf{q}}_n &= \mathbf{b}_n + \mathbf{H}_n^T(\tilde{\lambda} + \Delta\lambda) \end{aligned} \tag{10}$$

where

$$\mathbf{H}_t = [\frac{\delta\Phi}{\delta\mathbf{q}_t} \; ; \; \frac{\delta\Psi}{\delta\mathbf{q}_t}] \qquad \mathbf{H}_n = [\frac{\delta\Phi}{\delta\mathbf{q}_n} \; ; \; \frac{\delta\Psi}{\delta\mathbf{q}_n}]. \tag{11}$$

Given the equations above, the behavior of the constrained system composed of the tissue and needle is modelled in three steps:

Step 1, predictive motion: interacting objects are solved independently while setting $\Delta\lambda = 0$ (i.e. $\tilde{\lambda} = \lambda^t$). We obtain what we call a *predictive motion* $\mathbf{d}\dot{\mathbf{q}}_t^{\mathrm{p}}$ and $\mathbf{d}\dot{\mathbf{q}}_n^{\mathrm{p}}$ for each object. After the integration, we obtain $\mathbf{q}_t^{\mathrm{p}}$ and $\mathbf{q}_n^{\mathrm{p}}$. In this step, each object is solved independently using a dedicated solver which assembles the system matrices $\mathbf{A}_t$ and $\mathbf{A}_n$ and computes their factorizations.

Step 2, constraint solving: the constraint laws Φ and Ψ are linearized as follows:

$$\underbrace{\begin{bmatrix} \Phi(\mathbf{q}_t^{t+h}, \mathbf{q}_n^{t+h}) \\ \Psi(\mathbf{q}_t^{t+h}, \mathbf{q}_n^{t+h}) \end{bmatrix}}_{\delta^{t+h}} = \underbrace{\begin{bmatrix} \Phi(\mathbf{q}_t^{\mathrm{p}}, \mathbf{q}_n^{\mathrm{p}}) \\ \Psi(\mathbf{q}_t^{\mathrm{p}}, \mathbf{q}_n^{\mathrm{p}}) \end{bmatrix}}_{\delta^{\mathrm{p}}} + h\mathbf{H}_t \mathbf{d}\dot{\mathbf{q}}_t^{\mathrm{c}} + h\mathbf{H}_n \mathbf{d}\dot{\mathbf{q}}_n^{\mathrm{c}} \tag{12}$$

With $\mathbf{d}\dot{\mathbf{q}}_t^{\mathrm{c}}$ and $\mathbf{d}\dot{\mathbf{q}}_n^{\mathrm{c}}$ being the unknown corrective motions when solving Eq. 10 with $\mathbf{b}_t = \mathbf{b}_n = \tilde{\lambda} = 0$. By gathering equations 10 and 12, we have:

$$\delta^{t+h} = \delta^{\mathrm{p}} + \underbrace{h\left[\mathbf{H}_t \mathbf{A}_t^{-1} \mathbf{H}_t^T + \mathbf{H}_n \mathbf{A}_n^{-1} \mathbf{H}_n^T\right]}_{\mathbf{W}} \Delta\lambda \tag{13}$$

We obtain the value of $\Delta\lambda$ using a projected Gauss-Seidel algorithm that iteratively checks and projects the various constraint laws contained in Φ and Ψ (see [29]). During needle insertion, three main types of constraints are defined: first, a *surface puncture* constraint is used to penalize insertion through the surface (membrane) of the organ. The constraint allows for emulation of a membrane which is an anatomical feature of several abdominal organs, and is parametrized by a *puncture force* which is needed to penetrate the surface. Second, a *needle tip* constraint is defined in the tip of the needle as soon as it penetrates into the tissue. The constraint is parametrized by a *cutting force* which must be applied as long as the needle tip is creating a new path inside the tissue. Finally, *needle shaft* constraints are defined along the axis of the shaft. These constraints guarantee that the shaft is following the trajectory created by the advancing needle tip. They are parametrized by *friction force* which is needed in order to overcome a static friction between the needle and the surrounding tissue.

Step 3, corrective motion: when the value of $\Delta\lambda$ is available, the corrective motion is computed as follows:

$$\begin{aligned} \mathbf{q}_t^{t+h} &= \mathbf{q}_t^{\mathrm{p}} + h\mathbf{d}\dot{\mathbf{q}}_t^{\mathrm{c}} \;\text{ with }\; \mathbf{d}\dot{\mathbf{q}}_t^{\mathrm{c}} = \mathbf{A}_t^{-1}\mathbf{H}_t^T\Delta\lambda \\ \mathbf{q}_n^{t+h} &= \mathbf{q}_n^{\mathrm{p}} + h\mathbf{d}\dot{\mathbf{q}}_n^{\mathrm{c}} \;\text{ with }\; \mathbf{d}\dot{\mathbf{q}}_n^{\mathrm{c}} = \mathbf{A}_n^{-1}\mathbf{H}_n^T\Delta\lambda \end{aligned} \tag{14}$$

Although the Eq. 13 and 14 involve the inverses $\mathbf{A}_t^{-1}$ and $\mathbf{A}_n^{-1}$, which change at every time step because of the corotational formulation, we do not calculate the inverses explicitly, but we perform a back-substitution for each constraint using the factors of $\mathbf{A}_t$ and $\mathbf{A}_n$ computed in the predictive motion. Since this process is independent for each constraint, it can be parallelized or accelerated using CUDA as presented in [41].

5. Tests and Results

The tests described in this section have two main objectives. The first one is to underline the importance of accounting for deformations in a preoperative planning. The second one is to assess the efficiency of the *Haystack* algorithm by comparing it to two classical optimization methods.

5.1. Description of the Evaluation Scenario

Since a hepatic intervention is a good representative of a percutaneous procedure, and the human liver is a complex highly-deformable organ, the method has been evaluated on a set of patient-specific geometry for patients undergoing hepatic tumor radiofrequency ablation (RFA), which is a thermal ablation consisting in inserting a needle into the tumor to destroy cancerous cells by applying extreme heat from the needle tip. For this kind of procedure, we used the strict and soft constraints defined in Section 4.1. Let us note that the volume of the RFA, which is delimited by the 60°C isotherm surface defining the lethal temperature for the cells, can be either computed by methods such as Pennes equation to simulate bioheat propagation like in [42], or approximated by an ellipsoid. The simulation of bioheat propagation has the advantage of allowing to account for

the presence of vessels in the vicinity of the tumor, that may reduce the efficiency of the treatment, however, it is computationally expensive. In order to keep the time needed for the optimization in reasonable terms, we opted for the method based on the approximated ellipsoid which defines the safety margin as presented in [7].

To mimic the intervention scenario by the mechanical simulation, we have created a simulation scene in SOFA composed of four objects, as illustrated in Fig. 6:

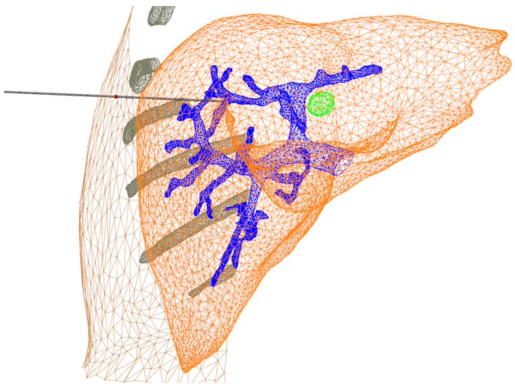


Figure 6. Example of a simulation scene consisting of a deformable needle, the abdominal wall, the rib cage, the liver, the vessels and the target (in green).

- *Deformable needle* modeled with Timoshenko beams (see section 4.3). In order to match the properties of a real needle, we set the needle diameter to 1.27 mm (approx. 18 gauge) and the Young's modulus to 75 GPa and its length to 17 cm.
- *Abdominal wall* modeled with linear FE method: only local deformations in the vicinity of the needle entry were modeled and the FE mesh was fixed by homogeneous Dirichlet boundary conditions in a distance of approximately 10 cm from the entry point. In order to achieve a reasonable amount of deformation, we set the Young's modulus of the wall to 5 kPa.
- *Rib cage* modeled only as a geometrical obstacles without any mechanical function.
- *Liver* modeled with co-rotational FE method which accounts for large deformations as described in section 4.3. In this scenario, we supposed that the behavior of the organ is simulated by a homogeneous isotropic model: the Young's modulus was set to 1 kPa. The value approximates the average moduli reported for healthy liver in the literature [3]. As for the boundary conditions, the liver was anchored in the space using homogeneous Dirichlet conditions in the areas corresponding to the entry of important vessels (hepatic and portal vein and the aorta) and location of ligaments. In this evaluation, the surrounding organs were not simulated.
- *Vascularization and tumor* modeled as geometrical objects. Nevertheless, unlike the rib cage, they were mapped to the FE model of the liver parenchyma. Therefore, as the liver model gets deformed, both vessels and the target follow this deformation and therefore move.

In each step of the optimization, the needle is inserted from the outside of the abdominal wall at a constant velocity of 4 cm/s. First, it penetrates the abdominal wall and then, the needle punctures the surface of the liver and continues penetrating the parenchyma. The insertion is stopped when the maximal insertion depth is achieved or the needle has hit and now exits the target lesion.

5.2. Datasets and deformations amplitude

The simulation scene was instantiated with 12 different datasets (referenced as DS1 to DS12) which were used for the evaluation. The datasets were prepared based on four real patients models from the *3D-IRCADb* database [43]. Each of them consists of 3D triangular surface meshes of the abdominal organs and the body contour generated from preoperative CT-scans. The surface triangulations were used to generate volume meshes composed of linear tetrahedral elements; the average size of a mesh used for the abdominal wall and the liver was 25,000 and 15,000 elements, respectively.

For each patient model, we created three datasets by making three versions of the tumor each having different size, form, and position, in order to quantify the impact of the tumor attributes on the planning. The 12 datasets we obtained are summarized in Table 1.

Table 1. Description of the tumors properties in the 12 experimental datasets

Dataset	**Diameter** (mm)	**Volume** (mm^3)	**Close to vessels**	**Close to skin**
DS1	14	180	no	yes
DS2	16	397	no	no
DS3	16	397	yes	no
DS4	14	198	no	no
DS5	14	147	yes	no
DS6	25	885	yes	yes
DS7	18	335	no	no
DS8	13	128	yes	yes
DS9	17	525	yes	yes
DS10	17	275	yes	no
DS11	16	372	no	no
DS12	13	218	yes	yes

The average size of a tumor was 12×10×7 mm (typical tumor diameters for RFA are inferior to 3.0 cm [44]). The details for each variation are given in Table 1. Fig.8b shows how deep in the vascular system the target can be located.

The planning on all datasets was performed on a PC with Intel Core i7 running at 2.67 GHz and 8 GB RAM.

The Euclidean distance between the initial position C of the tumor's center of mass and its position C_D after a simulation reached up to 25 mm, while the Euclidean distance between the needle tip's planned position in static condition $N_{S_{opt}}$ and its simulated position N_D reached up to 9 mm. This means that in

the worst case the target would be missed by 16 mm according to our simulation. Fig.7 shows the comparison between the displacements of the needle's tip and the tumor's center in each dataset. Fig.8a shows an example of amplitude of liver deformation and target displacement.

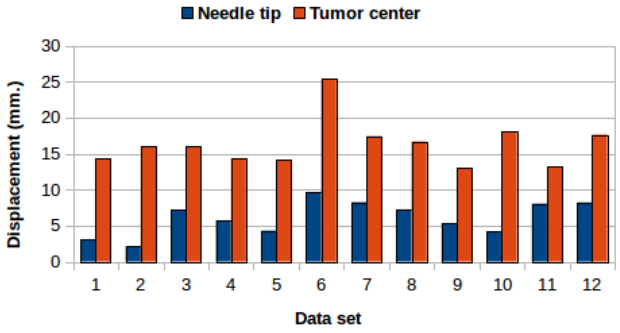
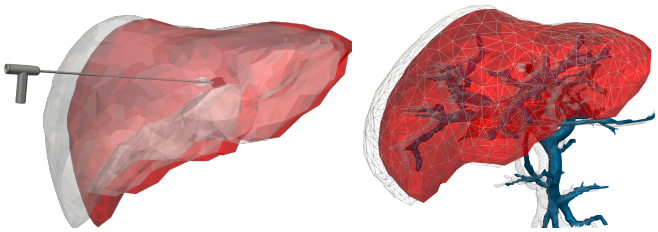


Figure 7. Comparison between needle tip and tumor center displacements.



(a) liver, tumor, and needle. (b) liver, portal vein, and tumor.

Figure 8. Static (in gray) and deformed (colored) models superimposed on each other. The needle causing the deformation is shown in the left part.

5.3. *Overview of optimization methods and evaluation of efficiency*

To evaluate the performances of our algorithm, we experimented (it) with two optimization techniques: an Exhaustive search (EX), the Nelder-Mead Downhill method (DH), in addition to our proposed *Haystack* optimization technique (HST). These three techniques were applied after an identical initialization phase in static conditions using Nelder-Mead method where T_{Sopt} is computed, so that their performances are compared only for the deformable phase.

We have used the EX method in the following way. A first simulation is performed with T_{Sopt} as an input. Then a search range around the initial proposed trajectory T_{Sopt} is defined based on the simulation. For the translations, $\Delta\mathbf{x} = \lambda|\mathbf{x}_{C_D} - \mathbf{x}_C|$ is defined. Similarly, an angle of $\lambda.\theta$ is used as maximum rotations. Parameter λ has been experimentally set to 2, in order to be sure to include the moved target within the range. At each iteration of the EX algorithm, $\mathbf{x}$ and $\mathbf{v}$ are modified by randomly choosing values within the defined range, with a precision of 10^{-5}mm for the coordinates of $\mathbf{x}$ and 10^{-5} degree for angles α and β of direction $\mathbf{v}$.

DH [34] is a derivative-free numerical method suitable to our non-derivable optimization function, which has proven in previous works to be able to converge to a minimum of f [7].

HST algorithm is in some way inspired by DH, it can be considered as an adaptation of the latter by replacing the "blind search" on the simplex vertices in DH by a "context-aware" search by integrating information from the simulations feedback. The interest is, as we ensured all over this work, to reduce as much as possible the number of simulations while preserving an acceptable precision.

The results are summarized in Table 2, where we first present the results of the initial static phase. For each dataset (column 1), we report the result of the initial deformed trajectory T_D computed from the simulation of insertion following IS_{Sopt} in static condition, as one of: miss target / hit target / or hit obstacle (column 2), and the value of cost function $f(T_D)$ in static condition (column 3). It can be noticed that in case of missing the target or hitting an obstacle, the evaluation is assigned its worst value (= 1), while a value < 1 is computed in case when the target has been already hit but the trajectory is not optimal. We also report in the table the number of simulations (column 5) for each dataset using the three techniques with the corresponding computation time (column 6). For EX method, the number of simulations was limited to 1,000 for each dataset. For the other two methods, the simulations were stopped when the improvement of f fell under a threshold of 10^{-3}. Finally, we report the value of $f(T_{Dopt})$ in the deformable environment (column 7). It can be also noticed that the values in this column are all < 1, which means that all methods converged to a curved path hitting the tumor for all datasets.

Fig. 9 shows the comparison of the performances of the three methods in terms of required number of simulations needed to converge, and Fig. 10 shows the comparison in terms of the quality of the optimal trajectory reflected by the value of $f(T_{Dopt})$ which had to be minimized in order to optimize the soft constraints.

From Fig. 9, it can be noticed that HST method requires a smaller number of simulations to converge than EX and DH, while Fig. 10 illustrates that the obtained precision in HST is close to the one in EX and 2 times better than DH in average. The results are discussed in more detail in the next section.

6. Discussion

The results presented in the previous section highlight two main points: accounting for deformations in the planning process is essential, and the algorithm *Haystack* is efficient to achieve this task. A dedicated iterative optimization algorithm has been proposed for preoperative path planning in percutanous procedures. The planning algorithm has been coupled with a simulation engine for flexible needle insertion in soft tissues. Tests were performed on 12 geometric patient-specific models, and a comparison with two well-known optimization methods has been done. The amplitude of the deformations measured after the simulations shows that by performing a mere static planning, the target could be missed by up to 16 mm. This

Table 2. Comparison between the three optimization techniques in a deformable environement in terms of required number of simulations, computation time, and cost function value.

Dataset	**Result of T_D**	$f(T_D)$ **[0, 1]**	**Algorithm**	**#Simulations**	**Time s.**	$f(T_{Dopt})$ **[0, 1]**
DS1	Miss target	1	EX	1000	16908	0,12
			DH	52	768	0,27
			HST	21	381	0.09
DS2	Hit obstacle	1	EX	1000	15823	0,0860
			DH	66	1345	0,33
			HST	13	241	0,123
DS3	Hit target	0,62	EX	1000	18980	0,26
			DH	77	1512	0,36
			HST	25	241	0,18
DS4	Hit target	0,79	EX	1000	21473	0,094
			DH	47	806	0,38
			HST	21	384	0,221
DS5	Miss target	1	EX	1000	18262	0,13
			DH	281	3018	0,22
			HST	27	433	0,232
DS6	Miss target	1	EX	1000	22411	0,162
			DH	347	4163	0,42
			HST	31	493	0,191
DS7	Miss target	1	EX	1000	15331	0,15
			DH	94	1618	0,57
			HST	19	330	0.19
DS8	Hit target	0,42	EX	1000	12776	0,23
			DH	156	2349	0,19
			HST	21	346	0,11
DS9	Miss target	1	EX	1000	16864	0,18
			DH	173	2672	0,23
			HST	29	437	0,18
DS10	Miss target	1	EX	1000	23026	0,17
			DH	41	387	0,23
			HST	37	455	0,17
DS11	Hit obstacle	1	EX	1000	27885	0,14
			DH	67	693	0,31
			HST	15	284	0,172
DS12	Hit target	0,89	EX	1000	20751	0,12
			DH	78	632	0,48
			HST	17	310	0,164
Average			EX	1000	19208	0.15
			DH	123	1664	0.33
			HST	23	373	0.169

observation emphasizes the interest of a planning taking the deformable environment into account.

The proposed optimization algorithm *Haystack* has been compared to two well known optimization methods. The results show that the three optimization techniques are capable of converging to some insertion settings leading to a feasible and optimal trajectory in a deformable environment, but the *Haystack* algorithm is always the fastest, as it requires the smallest number of simulations to converge. Since the simulation of insertion performed at each step of the optimization process is time consuming, this is an important feature of the proposed algorithm.

Although the proposed deformable approach is more time consuming than a simple rigid planning, it remains computationally efficient with an average computation time of 373 s. and an average of 23 simulations. This is significantly better when compared to the DH method, which completed in an average of about 27 minutes: HST is more than 4 times faster than DH in average.

Furthermore, HST algorithm converges to satisfying minimum values of the cost function, *i.e.* close to EX values that serve as a reference. The average value of $f(T_{Dopt})$ obtained with HST (0.169) is quite comparable to the one obtained with

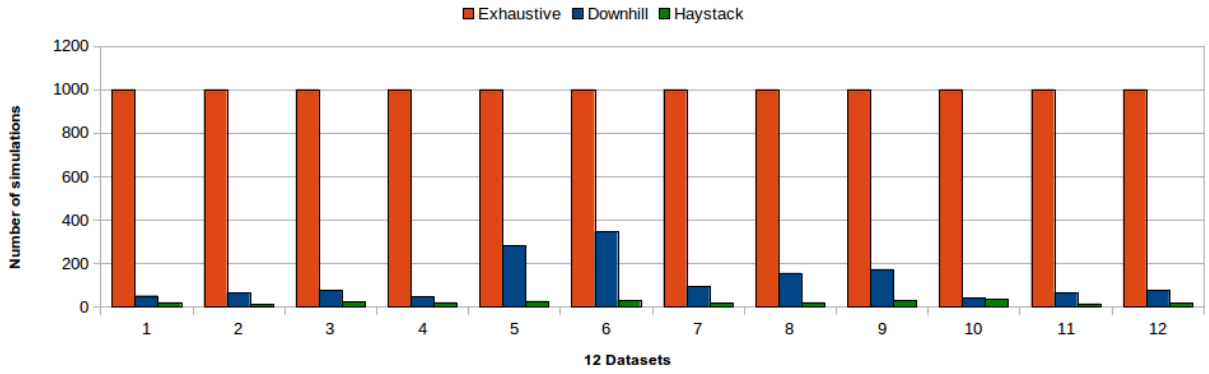


Figure 9. Comparison of the number of simulations for the three methods: EX (orange), DH(blue), and HST (green).

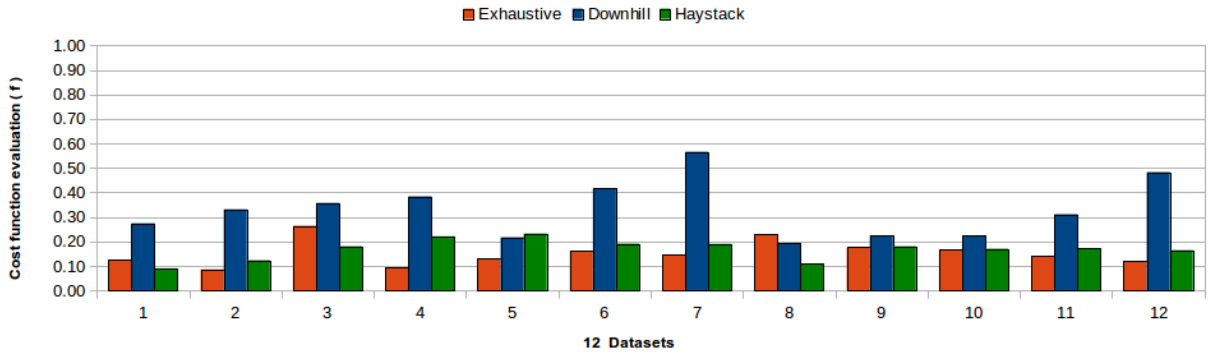


Figure 10. Comparison of the values of $f(T_{Dopt})$ for the three methods: EX (orange), DH(blue), and HST (green).

EX (0.15), with only 1.9% of quality loss within the value range $[0, 1]$. Comparatively, the average value of $f(T_{Dopt})$ obtained with DH (0.33), is twice higher than HST with 18% of quality loss within the same value range. This means that the quality of the trajectory obtained with HST is much better than with DH.

In addition, the results show that even when T_{Sopt} can already hit the target (as in DS3, DS4, DS8, and DS12 datasets), the optimizations in deformable environment can enhance the quality of T_D (compare column 3 and column 7), which is also an interesting outcome.

In the current implementation, the simulation code used by the planning algorithm relies on a geometrically non-linear constitutive soft tissue model accounting for large deformations. The interaction between the flexible needle and different tissue layers is modeled using advanced constraints solved at each step of the simulation. Although the accuracy of the simulation has not been clinically validated yet, the method has been verified by a comparison with a commercial FE package [32]. In this current work, the personalization of the model was limited to the geometry, while material parameters were based on generic values taken from the literature. Nevertheless, it has been shown previously that patient-specific parametrization of such models is straightforward and additional advanced properties of tissues such as heterogeneity and anisotropy can be added while keeping its performance [44, 34]. As a consequence, updating the parameter choice or constitutive model of the planning algorithm could improve the accuracy of the planning without requiring any particular change in the method. Obviously, the choice of model and parameters is problem-dependent, and need to be carefully chosen, as in our recent work for neurosurgery preoperative planning [45].

It is also important to keep in mind that any simulation remains an approximation of the real world. The promising results presented in this article illustrate the feasibility of biomechanically-inspired advanced planning, and could be further applied to real-world problems when completed by an intraoperative guidance. Nonetheless, we believe that such an advanced planning offers the advantage to propose better insertion settings than a simple planning in static conditions. By anticipating the deformations, it can reduce the need for manual adjustments usually required during the intraoperative guidance.

7. Conclusion

To conclude, this work shows that it is possible to include information from soft tissue deformations and needle bending in the planning process for percutaneous interventions. A

dedicated optimization algorithm named *Haystack* able to converge quickly towards precise optimal insertion settings has been introduced. Tests on 12 different datasets have shown that *Haystack* algorithm converges in a reasonable time (less than 7 minutes with our experimental setup) to a safe and efficient curved trajectory that is close in precision to an exhaustive method.

Further work should include experiments or analysis of data acquired intraoperatively, and taking simulations of breathing into consideration.

Acknowledgements

The authors would like to thank the French Research Agency (ANR) for funding this work through project ACouStiC (ANR 2010 BLAN 020901).

References

[1] D. J. van Gerwen, J. Dankelman, J. J. van den Dobbelsteen, Needle–tissue interaction forces–a survey of experimental data, Medical engineering & physics 34 (6) (2012) 665–680.

[2] J. T. Hing, A. D. Brooks, J. P. Desai, Reality-based needle insertion simulation for haptic feedback in prostate brachytherapy, in: Proceedings of IEEE International Conference on Robotics and Automation, IEEE, 2006, pp. 619–624.

[3] A. R. Kemper, A. C. Santago, J. D. Stitzel, J. L. Sparks, S. M. Duma, Biomechanical response of human liver in tensile loading, in: Annals of Advances in Automotive Medicine/Annual Scientific Conference, Vol. 54, Association for the Advancement of Automotive Medicine, 2010, p. 15.

[4] A. R. Kemper, A. C. Santago, J. D. Stitzel, J. L. Sparks, S. M. Duma, Effect of strain rate on the material properties of human liver parenchyma in unconfined compression, Journal of biomechanical engineering 135 (10) (2013) 104503–104503–8.

[5] M. Stoll, M. Boettger, C. Schulze, M. Hastenteufel, Transfer of methods from radiotherapy planning to ablation planning with focus on uncertainties and robustness., Biomedical Engineering / Biomedizinische Technik. 57 (SI-1 Track-C) (2012) 901–904.

[6] C. Baegert, C. Villard, P. Schreck, L. Soler, Multi-criteria trajectory planning for hepatic radiofrequency ablation, in: proceedings of MICCAI'07, Vol. 4791 of Springer LNCS, 2007, pp. 584–592.

[7] C. Baegert, C. Villard, P. Schreck, L. Soler, A. Gangi, Trajectory optimization for the planning of percutaneous radiofrequency ablation of hepatic tumors, Computer Aided Surgery 12 (2) (2007) 82–90.

[8] M. Mahvash, P. Dupont, Fast needle insertion to minimize tissue deformation and damage, in: In IEEE International Conference on Robotics and Automation, 2009, pp. 3097–3102. doi:10.1109/ROBOT.2009.5152617.

[9] Y. Kobayashi, J. Okamoto, M. Fujie, Physical properties of the liver and the development of an intelligent manipulator for needle insertion, in: In IEEE International Conference on Robotics and Automation, 2005, pp. 1632–1639. doi:10.1109/ROBOT.2005.1570347.

[10] T. Rohlfing, C. R. Maurer Jr, W. G. O'Dell, J. Zhong, Modeling liver motion and deformation during the respiratory cycle using intensity-based nonrigid registration of gated mr images, Medical physics 31 (3) (2004) 427–432.

[11] H. W. Korin, R. L. Ehman, S. J. Riederer, J. P. Felmlee, R. C. Grimm, Respiratory kinematics of the upper abdominal organs: a quantitative study, Magnetic resonance in medicine 23 (1) (1992) 172–178.

[12] M. A. Clifford, F. Banovac, E. Levy, K. Cleary, Assessment of hepatic motion secondary to respiration for computer assisted interventions, Computer Aided Surgery 7 (5) (2002) 291–299.

[13] T. Butz, S. Warfield, K. Tuncali, S. Silverman, E. Van Sonnenberg, R. Jolesz, F.A.and Kikinis, Pre- and intra-operative planning and simulation of percutaneous tumor ablation, in: proceedings of MICCAI'00, Vol. 1935 of Springer LNCS, 2000, pp. 317–326.

[14] I. Altrogge, T. Preusser, T. Kröger, C. Büskens, P. Pereira, D. Schmidt, H. Peitgen, Multiscale optimization of the probe placement for radiofrequency ablation, Academic Radiology 14 (11) (2007) 1310–1324.

[15] C.-C. Chen, M. Miga, R. Galloway, Optimizing electrode placement using finite-element models in radiofrequency ablation treatment planning, IEEE Transactions on Biomedical Engineering 56 (2) (2009) 237–245.

[16] S. Haase, T. Patz, H. Tiesler, I. Altrogge, T. Preusser, Radiofrequency ablation planning beyond simulation, in: Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE, 2012, pp. 191–194.

[17] K.-F. Wang, W. Pan, F. Wang, G.-F. Wang, P. Madhava, H.-M. Pan, D.-X. Kong, X.-G. Liu, Geometric optimization of a mathematical model of radiofrequency ablation in hepatic carcinoma, Asian Pacific Journal of Cancer Prevention 14 (10) (2013) 6151–6158. doi:10.7314/apjcp.2013.14.10.6151.

[18] A. Seitel, M. Engel, C. Sommer, B. Redeleff, C. Essert-Villard, C. Baegert, M. Fangerau, K. Fritzsche, K. Yung, H.-P. Meinzer, L. Maier-Hein, Computer-assisted trajectory planning for percutaneous needle insertions, Medical Physics 38 (6) (2011) 3246–3260.

[19] C. Schumann, J. Bieberstein, C. Trumm, D. Schmidt, P. Bruners, M. Niethammer, R. Hoffmann, A. Mahnken, P. Pereira, H. Peitgen, Fast automatic path proposal computation for hepatic needle placement, in: SPIE Medical Imaging, Vol. 7625, 2010, p. 76251J.

[20] K. März, A. M. Franz, B. Stieltjes, J. Iszatt, A. Seitel, B. Radeleff, H.-P. Meinzer, L. Maier-Hein, Mobile EM field generator for ultrasound guided navigated needle insertions, in: Proceedings of the 4th International Conference on Information Processing in Computer-Assisted Interventions, IPCAI'13, 2013, pp. 71–80.

[21] C. Villard, L. Soler, N. Papier, V. Agnus, A. Gangi, D. Mutter, J. Marescaux, RF-Sim: a treatment planning tool for radiofrequency ablation of hepatic tumors, in: proceedings of Information Visualization, IEEE Computer Society Press, 2003, pp. 561–566.

[22] C. Essert-Villard, C. Baegert, P. Schreck, Multi-semantic approach towards a generic formal solver of tool placement for percutaneous surgery, in: Proceedings of Knowledge Engineering & Ontology Development, 2009, pp. 443–446.

[23] C. Essert, C. Haegelen, F. Lalys, A. Abadie, P. Jannin, Automatic computation of electrode trajectories for deep brain stimulation: a hybrid symbolic and numerical approach, International journal of computer assisted radiology and surgery 7 (4) (2012) 517–532.

[24] S. DiMaio, S. Salcudean, Needle steering and motion planning in soft tissues, IEEE Transactions on Biomedical Engineering 52 (6) (2005) 965–974.

[25] W. Park, Y. Wang, G. S. Chirikjian, Path Planning for Flexible Needles Using Second Order Error Propagation, in: Algorithmic Foundations of Robotics, 2010, pp. 583–595.

[26] R. Alterovitz, M. Branicky, K. Goldberg, Motion Planning Under Uncertainty for Image-guided Medical Needle Steering., The International journal of robotics research 27 (11-12) (2008) 1361–1374.

[27] S. Patil, J. van den Berg, R. Alterovitz, Motion planning under uncertainty in highly deformable environments, Proceedings of Robotics: Science and Systems VII (2011) 241–248.

[28] S. Misra, K. B. Reed, B. W. Schafer, K. T. Ramesh, a. M. Okamura, Mechanics of Flexible Needles Robotically Steered through Soft Tissue., The International journal of robotics research 29 (13) (2010) 1640–1660. doi:10.1177/0278364910369714.

[29] C. Duriez, C. Guébert, M. Marchal, S. Cotin, L. Grisoni, Interactive simulation of flexible needle insertions based on constraint models, in: G.-Z. Yang, D. Hawkes, D. Rueckert, A. Noble, C. Taylor (Eds.), Proceedings of MICCAI 2009, Vol. 5762, Springer, 2009, pp. 291–299.

[30] O. Goksel, K. Sapchuk, S. Salcudean, Haptic simulator for prostate brachytherapy with simulated needle and probe interaction, IEEE Transactions on Haptics 4 (3) (2011) 188–198.

[31] P. Moreira, I. Peterlík, M. Herink, C. Duriez, S. Cotin, S. Misra, Modelling prostate deformation: Sofa versus experiments, Mechanical Engineering Research 3 (2) (2013) 64–72.

[32] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, S. Cotin, SOFA: A Multi-Model Framework for Interactive Physical Simulation, in: Y. Payan (Ed.), Soft Tissue Biomechanical Modeling for Computer Assisted Surgery, Vol. 11 of Studies in Mechanobiology, Tissue Engineering and

Biomaterials, Springer, 2012, pp. 283–321. doi:10.1007/8415_2012_125.
[33] C. Schumann, C. Rieder, S. Haase, K. Teichert, P. Süss, P. Isfort, P. Bruners, T. Preusser, Interactive multi-criteria planning for radiofrequency ablation, Int J CARS 10 (6) (2015) 879–889. doi:10.1007/s11548-015-1201-6.
[34] J. Nelder, R. Mead, A simplex method for function minimization, Computer Journal 7 (4) (1965) 308–313. doi:10.1093/comjnl/7.4.308.
[35] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, S. Cotin, SOFA: A Multi-Model Framework for Interactive Physical Simulation, in: Y. Payan (Ed.), Soft Tissue Biomechanical Modeling for Computer Assisted Surgery, Vol. 11 of Studies in Mechanobiology, Tissue Engineering and Biomaterials, Springer, 2012, pp. 283–321.
[36] Boost library, http://www.boost.org.
[37] D. Baraff, A. Witkin, Large steps in cloth simulation, in: SIGGRAPH '98, ACM Press, 1998, pp. 43–54.
[38] C. Felippa, B. Haugen, A unified formulation of small-strain corotational finite elements: I. theory, Computer Methods in Applied Mechanics and Engineering 194 (21) (2005) 2285–2335.
[39] M. Nesme, Y. Payan, F. Faure, Efficient, physically plausible finite elements, in: J. Dingliana, F. Ganovelli (Eds.), Eurographics (short papers), 2005.
[40] I. Peterlik, M. Nouicer, C. Duriez, S. Cotin, A. Kheddar, Constraint-based haptic rendering of multirate compliant mechanisms, IEEE Trans. Haptics, Special Issue on Haptics in Medicine and Clinical Skill Acquisition 4 (2011) 175–187. doi:http://dx.doi.org/10.1109/TOH.2011.41.
[41] H. Courtecuisse, J. Allard, K. Pierre, S. P.-A. Bordas, S. Cotin, C. Duriez, Real-time simulation of contact and cutting of heterogeneous soft-tissues, Medical Image Analysis (2014) 20.
[42] A. Jaberzadeh, C. Essert, Multi-probe three-dimensional placement planning for liver cryosurgery: comparison of different optimization methods, in: 14th International Conference Computational and Mathematical Methods in Science and Engineering (CMMSE'14), Vol. 3, 2014, pp. 743–754.
[43] Ircadb01, http://www.ircad.fr/research/3dircadb/.
[44] T. Livraghi, S. N. Goldberg, S. Lazzaroni, F. Meloni, L. Solbiati, G. S. Gazelle, Small hepatocellular carcinoma: Treatment with radiofrequency ablation versus ethanol injection, Radiology 210 (3) (1999) 655–661.
[45] N. Hamzé, A. Bilger, C. Duriez, S. Cotin, C. Essert, Anticipation of brain shift in deep brain stimulation automatic planning, in: IEEE Engineering in Medicine and Biology Society (EMBC), 2015.