Automatic planning of needle placement for robot-assisted percutaneous procedures

Esia Belbachir · Ehsan Golkar · Bernard Bayle · Caroline Essert

Abstract

Purpose Percutaneous procedures allow interventional radiologists to perform diagnoses or treatments guided by an imaging device, typically a Computed Tomography (CT) scanner with a high spatial resolution. To reduce exposure to radiations and improve accuracy, robotic assistance to needle insertion is considered in the case of X-ray guided procedures. We introduce a planning algorithm that computes a needle placement compatible with both the patient's anatomy and the accessibility of the robot within the scanner gantry.

Methods Our preoperative planning approach is based on inverse kinematics, fast collision detection, and bidirectional rapidly-exploring random trees coupled with an efficient strategy of node addition. The algorithm computes the allowed needle entry zones over the patient's skin (accessibility map) from 3D models of the patient's anatomy, the environment (CT, bed), and the robot. The result includes the admissible robot joint path to target the prescribed internal point, through the entry point. A retrospective study was performed on 16 patients datasets in different conditions: without robot (WR), and with the robot on the left or the right side of the bed (RL/RR).

Results We provide an accessibility map ensuring a collision-free path of the robot and allowing for a needle placement compatible with the patient's anatomy. The result is obtained in an average time of about 1 minute, even in difficult cases. The accessibility maps of RL and RR covered about a half of the surface of WR map in average, which offers a variety of options to insert the needle with the robot. We also measured the average distance between the needle and major obstacles such as the vessels, and found that RL and RR produced needle placements almost as safe as WR.

E. Belbachir, E. Golkar, B. Bayle, C. Essert
ICube, Université de Strasbourg / CNRS (UMR 7357)
300 bd Sébastien Brant - BP 10413 - 67412 Illkirch, France
E-mail: {golkar;bernard.bayle;essert}@unistra.fr

E. Belbachir
UTBM, Université Bourgogne Franche Comté, Belfort, France
E-mail: esia.belbachir@utbm.fr

Conclusions The introduced planning method helped us prove that it is possible to use such a 'general purpose' redundant manipulator equipped with a dedicated tool to perform percutaneous interventions in cluttered spaces like a CT gantry.

Keywords Preoperative planning · Decision-making process · Robot-assisted interventions · Interventional radiology

1 Introduction

Percutaneous procedures are now well established interventions. For instance, the minimally invasive resection of abdominal tumors consists in inserting a needle shaped tool through the skin into the tumor, in order to destroy it by applying extreme heat or cold. The chances of success and the limitation of the risks of damaging an organ, causing a hemorrhage, or avoiding a recurrence, are strongly related to a thorough preoperative planning. Nowadays, these interventions are mostly manual, often performed by interventional radiologists under CT guidance, since this imaging modality has the best spatial resolution. Unfortunately, manual interventions limit targeting accuracy, specially in some difficult cases like out-of-the-plane orientations. This, combined with the risks of exposure to radiations incurred by the physician, led to the development of several robotic solutions.

Several robotic systems have been developed in laboratories for needle insertion assistance (see [11, 13, 15, 16, 21, 23] for example). In most cases, these systems have been designed purposely, in order to be as compact as possible. They are sometimes directly mounted on the patient's body [11, 16, 23], or attached to the table [15], and can enter within the imaging device gantry with the patient. These systems are generally not redundant, which means that they have only one configuration allowed to reach the desired needle placement, once installed on the patient. Most grounded systems are generally bigger devices, which may have additional degrees of freedom dedicated to the raw placement of the needle manipulation arm [21]. Outside the laboratories, only the Perfint ROBIO [3], and the iSYS interventional system [2] are commercially available, to the best of our knowledge. None of them is equipped with needle insertion tools, the insertion itself being manual, which does not prevent from X-ray exposure. In parallel to the development of these systems, research groups like ours have studied the feasibility of robotized needle manipulation, including grasping and insertion functionalities that could be adapted to existing robots, such as in [17]. Such systems made of polymer material compatible with the constraints of X-ray imaging could be used within the CT gantry for controlled CT-guided needle interventions. In this paper, the solution consisting of such a tool mounted on a redundant manipulator is considered.

One of the goals of our study is to prove that collaborative robots, purposely designed for safe human robot interaction but with architectures similar to industrial robots, could be used for medical interventions in highly constrained environments. Such a robot offers interesting flexibility, being able to use different tools, depending on the task to be performed. However, little attention has been paid to robotic assistance planning in this context, while two important challenges remain: 1) to find an accessible path for the robot in a very cluttered environment; 2) to place the robot in order to cope with other intervention constraints. Some authors have compared manual and robotics insertions, such as for instance [22], but to

the best of our knowledge, none of the robots proposed so far in the literature is associated with an automatic needle placement planner. The first purpose of this work is to propose such a planner able to compute automatically the robot configurations allowing an accessibility to the target in a constrained environment, and display them as a map. Besides being a software tool necessary to operate safely, the proposed planner can also be very useful to evaluate the choice, design and size of the robotic structure and its needle insertion tool.

Decision support systems for needle placement have been proposed in the past by various teams [4,5,7,18–20] to assist the preoperative planning process. Some of them focused on optimizing the effect of the treatment [4,7] and estimate the therapy outcome. Others used various techniques to account for anatomical structures surrounding the target to propose safe and efficient needle placement [5, 18–20]. These systems have been designed for the anatomy-based planning of manual insertion only. They usually have two main goals: 1) determining the safe entry points and needle placements and 2) among them finding the most optimal ones. Note that in the present study, we only focus on the first objective. In the following, we introduce a novel approach that computes the feasible entry points on the skin. Feasible entry points are defined as follows: they allow for a safe insertion towards the target, and they are accessible by the robot using a collision-free motion. This way, not only we ensure safety regarding the internal anatomy but also accessibility of the insertion site by a robot's arm. It combines techniques from needle anatomy-based placement planning and robotics path planning.

The main contributions of this paper are: 1) a novel needle placement planning algorithm accounting for both the internal anatomy of the patient, the constraints of a robotic arm, and the confined space of a CT gantry, dedicated to abdominal percutaneous surgery, 2) the RRT connect algorithm was enhanced to converge fast enough to be included into a loop, and 3) we performed an evaluation of the performances of the algorithm which allowed us to conclude that the use of such a robot was not constraining too much the insertion possibilities, demonstrating the theoretical feasibility of using a manipulator for such interventions.

After detailing our new methods in Section 2, we present our results on 16 datasets of patients models in Section 3 and discuss them in Section 4.

2 Materials and Methods

2.1 Robot path planning

Without loss of generality, our study focuses on a redundant robotic arm LBR IIWA 14 from KUKA [6], with the objective of mounting on it a motorized metal-free end-effector tool. This robot has seven joints, resulting in a *configuration* defined by vector $\theta \in \mathbb{R}^7$ of joint angles. The manipulator end-effector *pose* (position and orientation of the tool in space) is denoted as $x \in \mathbb{R}^5$, if the needle self rotation is not considered[1].

---

[1] Planning the self rotation of the needle would be of particular interest for needle bending/steering control, but it is far beyond the scope of the present study.

2.1.1 Inverse kinematics

The computation of the robot configuration θ^* that allows to place the end-effector at a given pose x^*, starting from an initial configuration θ_0 and pose x_0, is a classical inverse kinematics problem, even for a redundant system. We chose to solve it by using the Levenberg-Marquardt damped least squares (LM-DLS) method for its numerical stability around singularities [24]. This approach computes incremental joint motions $\Delta\theta$ leading to θ^*:

$$\Delta\theta = J^T(JJ^T + \lambda^2 I)^{-1}\Delta x \qquad (1)$$

In this equation, $J(\theta) = \left(\frac{\partial x}{\partial \theta}\right)$ is the robot Jacobian, λ a damping constant, I the identity matrix, and Δx an elementary displacement resulting from the sampling of the end-effector path, from the current pose to x_t. The damping constant, which can be variable, must be chosen carefully for the best trade-off between stability and fast convergence (see Section 2.3).

2.1.2 Bi-directional RRT

To compute the path between θ_0 and θ^*, a simple interpolation could possibly be sufficient if there were no obstacle in the workspace. In our case, the workspace is constrained by the presence of obstacles along the robotic arm path, such as the CT gantry, the bed, and of course the patient. Therefore, the shortest path in the configuration space might not be feasible without collision. In order to compute a safe path from θ_0 to θ^*, we take advantage of the redundancy of the manipulator and explore the configuration space to find a collision-free path.

A variant of the Rapidly-exploring Random Tree (RRT) called RRT-connect [14], which is a bidirectional RRT, has been chosen as the most suitable for our purpose. An RRT builds a tree of random paths in the configuration space. Starting from an initial configuration, the algorithm iteratively adds extra configurations as nodes and links them to the closest node already in the tree when the connection satisfies feasibility constraints. The bidirectional variant builds one tree from the initial configuration (R_0) and one from the target configuration (R^*), and tries to link both trees as illustrated on Fig.1. In our case, the feasibility constraints are to avoid collisions and metallic component of the robot arm within the acquisition zone of the CT (to avoid artifacts in the image).

2.1.3 Collision detection

To compute the collisions between the complete body of the robot and the obstacles, the tridimensional (3D) surface meshes of the robot and the obstacles are used. The main issue is the high density of the meshes that introduce high computation times, as collisions need to be computed for each new link in the tree. To reduce this computation time, we used two approaches. First, multi-threading allows to browse different branches of the tree in parallel. Second, the 3D meshes are partitioned with a bounding volumes hierarchy, the Object-oriented minimum Bounding Box (OBB) trees [9], to improve the efficiency of the collision detection.

2.2 Producing the accessibility map

Our objective is to determine which insertion points on the skin model would allow for a needle placement towards the pathology to be feasible. In this paper, we will consider that a candidate *insertion point* p_i is always associated to the *insertion vector* v_i from this point towards the centroid of the targeted pathology. A candidate needle placement following a straight trajectory T_i is a couple (p_i, v_i). The 3D model of the skin is the initial search space containing n candidate insertion points ($i \in [1, n]$). To reduce the search space, some candidates are rapidly discarded using two types of constraints:

1. **Crop**: bottommost candidates not reachable by the robot. If p_i is below the highest point of the scanner table, T_i is discarded.
2. **Anatomy**: candidates not anatomically feasible. Thanks to a process inspired by the literature [5, 19, 20] T_i is eliminated if it does not satisfy three hard constraints: avoid anatomical **obstacles** (vessels, bones), **tangency** with the liver (min. angle 45°), trajectory must be shorter than the needle **length**.

After that, the reachability of each remaining candidate needle placement by the robot is checked. For each T_i, the algorithm first searches for a valid collision-free configuration θ_i that allows to place the end-effector at a given pose x_i corresponding to an insertion of the needle at T_i. The needle insertion device is constrained to be at a chosen distance h from the skin for a safe insertion (Fig.1). The corresponding pose x_i is easily computed from h and (p_i, v_i). Then θ^i is found thanks to the LM-DLS approach described in Section 2.1.

Once θ_i is computed, the algorithm looks for a collision-free path from the initial configuration θ_0 to θ_i. The process begins with a one-directional RRT (R_0) starting from θ_0, which is precomputed in an initialization step. For each candidate T_i, an attempt to connect θ_i to R_0 is made. If the connection does not succeed, a tree R_i is built from θ_i and the RRT-connect algorithm is used to connect it to R_0 without collision, as illustrated on Fig.1. Once θ_0 is connected to θ_i, we know that needle placement at T_i is accessible by the robot.

To ensure the generality of the approach, we wanted our method to work for any possible initial position of the robot. However, it may occur that for a candidate T_i the configuration θ_i found from the initial θ_0 by the inverse kinematic LM-DLS is not collision-free. But this does not mean that there is no collision-free solution. In
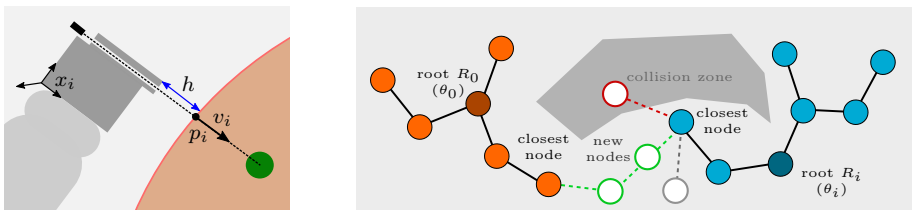

Fig. 1: Left: end effector (dark grey) at final position, mounted on the robot (light grey), pointing towards the target (green) in the body (orange). Right: illustration of the RRT-connect. Tree R_0 (red) is connected to tree R_f (blue) by iteratively adding new random nodes (white) until a connection without collision is found.

Algorithm 1: Efficient exploration of the candidate insertion points

---

Data: candidate_list: a set of candidate needle placements (unmarked).
θ_0: the initial configuration of the robot.
Result: sol_list: a set of solutions. A solution contains a needle placement, a final robot
config, and a path from the default config to the final config.
foreach $T \in$ candidate_list **do**
   | STORE_CANDIDATE_IN_TESTLIST(T, θ_0, test_list);
while IS_NOT_EMPTY(test_list) **do**
   | $\text{test}_i \leftarrow$ EXTRACT_FIRST_TEST(test_list);
   | **if** IS_NOT_MARKED(test_i) **then**
      | $\theta_i \leftarrow$ LM-DLS(test_i);
      | $\text{path}_i \leftarrow$ RRT_CONNECT(θ_0, θ_i);
      | **if** IS_COLLISION_FREE(path_i) **then**
         | MARK(test_i);
         | STORE_SOLUTION(test_i, θ_i, path_i, sol_list);
         | **foreach** T_j **in** CLOSEST_TRAJECTORIES(T_i, candidate_list) **do**
            | STORE_CANDIDATE_IN_TESTLIST(T_j, θ_i, test_list);

---

order to find a collision-free solution, the redundancy property of the manipulator is useful. A first possibility would be to modify randomly some of the joint values in θ_i without modifying pose x_i until a collision-free alternative is found. However, a few tests showed that this approach is very time consuming and does not guarantee that a solution can be found in a reasonable time. Alternatively, we propose to rather choose wisely an intermediate configuration to start the inverse kinematics with, before connecting to R_0.

Intermediate configurations are provided thanks to an intelligent browsing of the candidate needle placements. First, one T_i for which LM-DLS is able to provide a collision-free configuration θ_i directly from θ_0 is found. Then, the direct neighbours of p_i are browsed: for each neighbour p_j and its associated v_j, θ_i is used as a new initial configuration and LM-DLS produces θ_j. Candidates T_i and T_j being very close, we hypothesize that θ_i and θ_j are also close and the chances that θ_j has no collision is higher. Once θ_j is found, it is linked to the RRT and T_j is marked as done if it succeeded, or discarded otherwise. The process is reiterated until all candidates have been linked or discarded. Note that a candidate can be examined several times from different neighbours. The algorithm is summarized in Algorithm 1 where a *test* is a pair needle placement/initial configuration. At the end, only the candidate needle placements that allowed to find collision-free robot paths are kept in what we call the *accessibility map*.

2.3 Experimental setup and input data

To validate our approach and the feasibility of the device, the method was experimented on 16 datasets of patients models from the "3D-IRCADb" database [1]. This database consists of anonymized images of real patients with structures of interest already segmented by clinical experts and reconstructed as 3D models. For this study, we chose a subset of cases with liver tumors.

The 3D models consisted in triangular surface meshes of abdominal organs: skin, liver, bones, liver vascular structure. They also included tumors of various sizes, depths, and densities of environment, as described in Table 1. An illustration
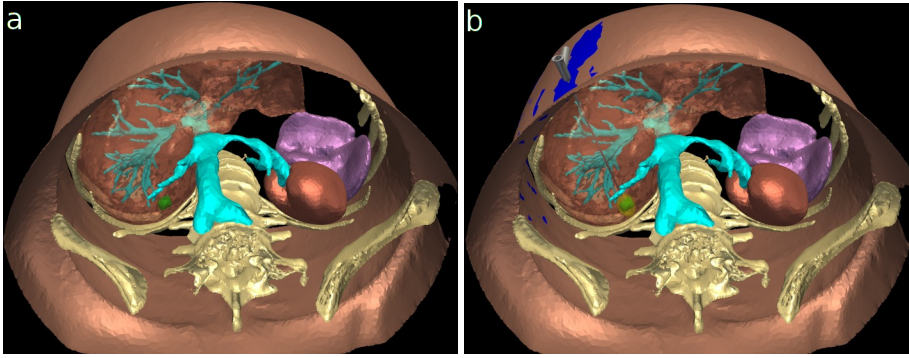
Fig. 2: Example of 3D scene, with: (a) initial 3D models, and (b) a needle placement (gray) through the accessibility map (blue) in deep liver's tumor (green) and close to vascular obstacles (cyan). For image clarity, some organs are omitted and the opacity of liver is manipulated.

of the 3D models is shown on Fig.2(a). An accessibility map for a patient with a deep tumor in the liver is depicted on Fig.2(b). As we can see, the accessibility is limited when the tumor is deeply located. Surface meshes of a CT gantry and bed were also used to compute the collisions, as shown on Fig.7. The models were positioned on the bed in supine position, with the abdomen within the gantry.

The surface meshes and specifications of the redundant manipulator LBR IIWA 14 from KUKA with seven degrees of freedom were used to compute and simulate the paths and the limits of the joint rotations, and compute the collisions. The end-effector was simulated by two dark grey boxes ($100 \times 153 \times 157$ mm) with a mobile needle within the last box, as can be seen on Figs.1 and 7. Distance h was set to 20 mm, λ to 1, and the maximum distance in the RRT-connect algorithm to 20 (euclidean norm in the 7-dimensional space).

As mentioned in subsection 2.2, four hard constraints were defined to discard unfeasible needle placements from the set of candidates:

– C1: bottommost candidates elimination,
– C2: anatomical obstacles avoidance,
– C3: ensure needle trajectory is shorter than needle length,
– C4: ensure minimal tangency with the liver

The set of these four constraints represents the condition without robot WR. Two more constraints were added to check the feasibility with the robot:

– CRL: feasibility with the robot located on the left of the patient
– CRR: feasibility with the robot located on the right of the patient

Table 1: Summary of the tumors sizes (L=Large, M=Medium, S=Small), depths (D=Deep, M=Medium, S=Shallow), and close obstacles (B=Bones, V=Vessels).

Case #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Size	L	M	L	S	S	S	L	M	S	S	S	L	S	L	M	L
Depth	D	S	M	S	S	D	S	D	D	S	S	D	M	D	D	S
Close to	B	B	-	B	-	V	B	-	V, B	-	-	V	B	-	V	B

The four WR constraints together with the CRL constraint constitute condition RL (robot left), while WR constraints together with the CRR constraint constitute condition RR. Each condition with the robot having a total of five constraints. As already explained, the process consists in successively eliminating candidate needle placements that do not respect these constraints. As a result, the accessibility map produced with five constraints in RL and RR conditions are subsets of the accessibility map produced with only four constraints in WR condition.

The algorithms that we presented have been programmed into the MITK [25] platform for the experiment. The planning was tested in three different conditions: without robot (**WR**), with the robot at the left of the CT bed (**RL**), and with the robot at its right (**RR**). In each case, we launched the algorithm described above until an accessibility map is found, and measured computation times, areas of the maps (sums of their triangles areas), and minimal distance to the main obstacles. Due to the randomness within the RRT algorithm, 50 runs were launched for each patient in order to have a more objective measure of the computation times. All the experiments were performed on a Core i7-6700 (3.40 GHz) with 16 Gb RAM.

3 Results

The computation times are summarized in Table 2. For the first four hard constraints times are homogeneously short for all cases (columns C1 to C4). The time to produce collision-free solutions using constraints CRL or CRR is notably higher due to the RRT approach that involves numerous collision checks. The gray columns present the total computation time with or without robot. At the bottom of the table, we present the average times, standard deviations, minimum and maximum values. We can note that the average time for the whole planning process is under 2 s. (1.55±1.01) for WR condition ("Total WR" column), and approximately 40 s. with any of the robot conditions (41.30±15.13 in "Total RL" column, and 38.49±12.48 in "Total RR" column) which is a reasonable computation time in our context and suitable for clinical routine.

The maximum overall computation times for WR, RL and RR were respectively 3.85 s., 70.65 s. and 55.33 s. In contrast, the minimum times were respectively 0.62 s., 17.79 s. and 16.69 s. In addition, the average ratio of robot computation time over the total time is 96.27% for CRL and 96% for CRR, meaning that ≈ 96% of total computation time is taken by the RRT algorithm when the robot is considered. A paired two-sample Student's T-test showed that there is no statistically significant difference between the processing times of both sides (p-value=0.57).

Chart on Fig.3 presents the results of columns CRL and CRR of Table 2, decomposing the computation times into two parts. The bottom (orange) part of each bar corresponds to the time spent in collision detections (IS_COLLISION_FREE in Algorithm 1), and the top (black) parts correspond to the time spent in the rest of Algorithm 1. It can be observed that even with the approaches used to accelerate the collisions detection, it still represents most of the computation time of the robot constraint (average of 92% for RL and 87% for RR).

The surface areas of the accessibility maps for all cases are reported on the bar charts of Fig.4. The average area of the initial surface of skin mesh is 197.9 mm^2, and 9.2 mm^2, 4.7 mm^2 and 4.5 mm^2 for WR, RL and RR, respectively. This

Table 2: Average computation times (s.) of 50 runs for each individual case. C1, C2, C3, C4 constraints are presented separately, and summed up in "Total WR" column. Times for CRL and CRR are also defined separately, and summed up with Total WR in "Total RL" and "Total RR" respectively. Average times and standard deviations are shown at the bottom of the table.

Case #	C1	C2	C3	C4	Total WR	CRL	Total RL	CRR	Total RR
#01	0.03	0.74	0.02	0.28	1.06	38.32	39.39	35.48	36.53
#02	0.03	0.77	0.02	0.38	1.20	33.96	35.16	25.53	26.72
#03	0.06	0.36	0.03	0.62	1.07	43.96	45.04	54.26	55.33
#04	0.06	0.91	0.03	0.27	1.28	17.53	**18.81**	15.43	**16.69**
#05	0.06	1.06	0.04	0.78	1.94	33.02	34.97	44.70	46.65
#06	0.06	2.49	0.04	1.26	3.85	50.92	54.81	35.29	39.15
#07	0.06	0.96	0.04	0.35	1.41	34.01	35.43	53.69	55.12
#08	0.06	0.29	0.03	0.23	0.62	30.35	30.97	24.91	25.53
#09	0.06	0.63	0.03	0.17	0.89	16.89	**17.79**	17.55	**18.45**
#10	0.06	0.22	0.03	0.35	0.67	69.98	<u>70.65</u>	50.40	51.08
#11	0.06	0.47	0.03	0.51	1.07	23.42	24.49	34.55	35.63
#12	0.06	2.56	0.03	1.04	3.68	61.51	<u>65.13</u>	47.72	51.34
#13	0.06	0.72	0.03	0.37	1.18	54.88	56.07	32.64	33.82
#14	0.06	0.27	0.03	0.33	0.68	41.96	42.65	31.10	31.79
#15	0.06	0.79	0.04	0.51	1.40	43.73	45.09	42.39	43.74
#16	0.05	1.30	0.03	1.34	2.73	41.71	44.42	45.59	48.30
AVG	0.06	0.91	0.03	0.55	1.55	39.76	41.30	36.95	38.49
STD	0.01	0.70	0.01	0.37	1.01	14.76	15.13	12.16	12.48

represents average coverage ratios of 4.68% of the initial surface for WR, 2.37% for RL, and 2.28% for RR. In average, RL and RR produce insertion surfaces that cover respectively 50.67% and 48.79% of the area produced in WR condition, which means that approximately half of the manually feasible surface is also accessible by the robot. Although the algorithm is able to find a suitable needle placement for all cases for WR, no needle placement accessible with a robot was found for case #9. In addition, in case #4 the accessibility map covers a very small area (15 mm^2) with RL and there is no accessibility by a robot on the right (RR). A two-sample Student's T-test showed that there is no statistically significant difference between the surface areas produced with a robot on the left or on the right (p-value= 0.92).

In terms of safety of the proposed solutions, the results are comparable in the three tested conditions. Fig.4 summarizes for each case the average distance between candidate needle placements of the resulting accessibility map and the bones mesh. In our dataset, the average distances to bones are 22.60 mm for WR, 18.44 mm for RL, and 21.18 mm for RR. Similarly, Fig.4 shows the average distance to the vascular structure, computed as the distance to the closest vessel. The average safety distance to vessels is 9.62 mm for WR, 8.68 mm for RL, and 9.25 mm for RR. The computed p-values show that there is no statistically significant difference between the average distances with or without robot (all p-values > 0.1). In case #9 the tumor was not accessible by the robot, and in case #4 the tumor was accessible in WR and LR only, so that values are missing for these cases. Fig. illustrates with case #3 the visualization of the 3D scene displayed in our software after the accessibility maps are computed.
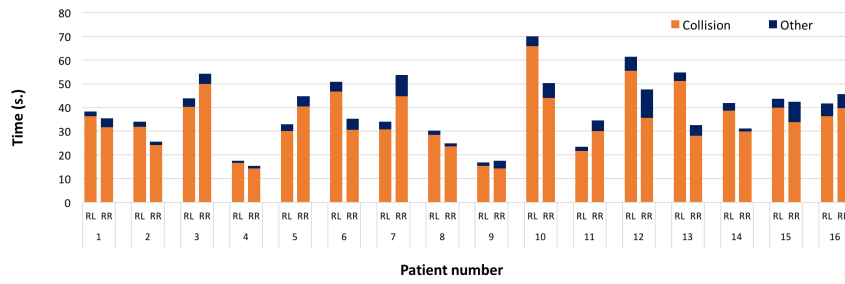
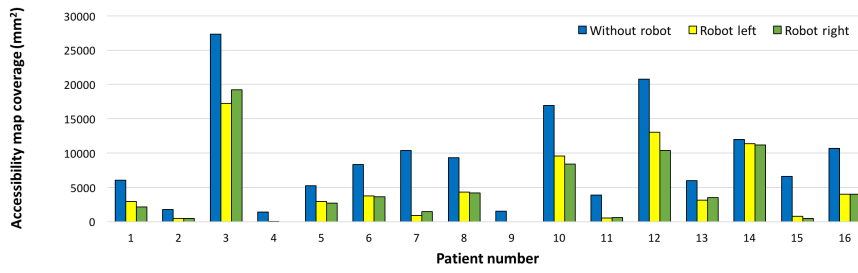Fig. 3: Computation times for CRL and CRR

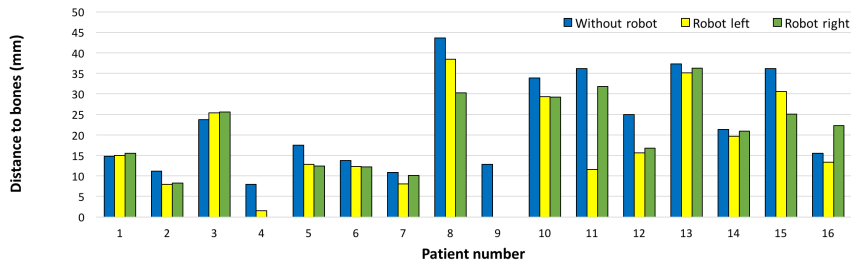Fig. 4: Coverage of the accessibility map for each case in mm^2.

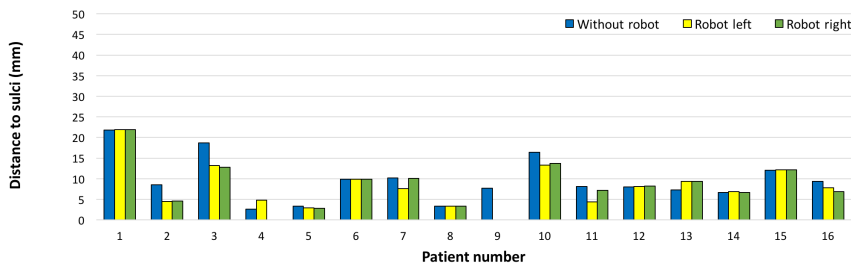Fig. 5: Average distances to bones for each case in mm.

Fig. 6: Average distances to vessels for each case in mm.

4 Discussion

In this experiment, we used a 7 DOF redundant manipulator, while some other groups used 6 DOF robots (for instance [15]). With such a large robot, the redundancy is an advantage to help avoid collisions with the environment. Many different motions are possible to reach a particular needle placement. However, it increases the complexity of the search for possible motions towards a needle placement. Our results show that it was possible to find solutions in reasonable time.

The approach we presented to compute accessibility maps for an interventional robot with 7 DOF was able to find solutions from both sides of the bed for 14 cases out of 16. In the two remaining cases, it was not possible to find solutions or only for one side. This is mainly due to the location of the tumor. For case #9, the tumor was located deeply in the organ and very close to obstacles, so that the tumor was difficult to reach even without the robot. The few remaining insertion points were impossible to reach with the robot. In case #4, the tumor was located closer to the surface of the skin and not so close to obstacles, however it was on the back of the patient, making it difficult to reach without colliding with the bed. We can hypothesize that with another position of the patient, for example in prone or lateral position, some tumors could be more easily accessible. Testing the algorithm with different parameters or position of the robot base might also allow to discover other access options. In future works, we could take advantage of such interesting information to propose automatically better patient positioning or robot placement. We also notice that in many other cases (#1, #6, #8, #12, #14, #15) the algorithm could find solutions for deep tumors of difficult access.

In the experiment, we measured for each case the average distance between candidate needle placements and bones or vessels. The distances have a high variability depending on the location and size of tumors. But here the idea was to
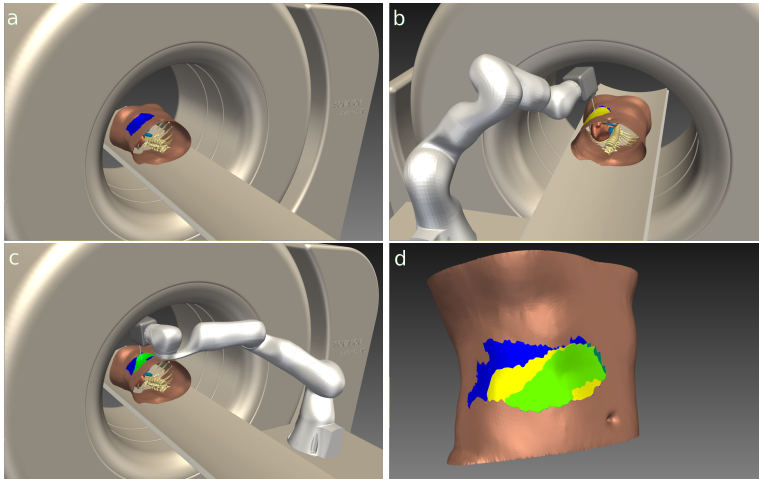

Fig. 7: Illustration of the feasibility areas on case #3 in the three conditions: (a) WR (blue area), (b) RL (yellow area), (c) RR (green area), and a closeup of the three maps superimposed. (b) and (c) show collision-free configurations of the robot. For the sake of visibility, h has been set to 7 cm for the screenshots.

determine whether the reduction of the surface area of the map due to the robot constraint had a trend to discard safe candidates, or if the average safety of the candidates was preserved. We observed that there was no significant difference between the average distances, demonstrating that the safety was comparable. The addition of the robot in the planning does not seem to be at the cost of safety. However, in some cases such as case #11, one of the conditions has a lower value. This result could be used to determine which side of the bed would be a good choice for the robot.

In some cases, the average distance of WR condition is shorter than with the robot (such as case #3 in Fig., or case #13 in Fig.). As accessibility maps produced with RL or RR are subsets of those produced with WR, their higher averages is not due to the presence of safer trajectories. The reason is that the eliminated part contained more unsafe trajectories compared to the remaining part.

When looking at Table 2, we observe that all of the 32 computation times with one of the robot constraints are under one minute except in two cases (underlined in the table). The maximal computation time is 70.65 seconds, which is acceptable in our context. The minimal computation times (in bold in the table) were obtained for cases #4 and #9. However these are also the cases that produced no or very small accessibility maps. The short computation times is due to the very reduced surface that already remained after applying constraints C1 to C4: 14 mm^2 for case #4 and 15 mm^2 for case #9, the average being 92 mm^2. Therefore, the search space for CRL or CRR was very small and the algorithm concluded very fast that there was no needle placement candidate accessible by a robot in this remaining area. Once again, this is due to the location of the tumor on the back of the patient, which produced already a high elimination of needle placement candidates by the first four constraints. This is the reason why the robot constraint (CRL or CRR) is applied after the other constraints. CRL and CRR are more time-consuming (almost 96% of the total time), so for a better efficacy most of the elimination should be done before by the other constraints.

Computation times also depend on the resolution of the meshes. First, the resolution of the skin mesh has an impact on the elimination process duration. In our dataset, the average number of triangles of this mesh is $\approx$ 39,000. As triangles are considered individually for elimination, a coarser mesh would allow a faster process and reduce the number of collision detections to be done, but reduce the accuracy on the border of the maps. It might even cause small accessible areas to disappear. The resolution of the obstacle meshes also influence the time spent in each collision detection. In this experiment, the average number of triangles in the vessels mesh was 54,327, and 118,967 in the bones mesh.

In this paper, we considered only the first part of an "offline" planning from preoperative CT images. The algorithm presented here allows to build an accessibility map, which is a binary information: candidate needle placements are feasible or not. It would be interesting in future developments to provide also information on the quality of each needle placements regarding both the anatomy (for instance give priority to the safest needle placements) as in [5] and the robot's specificities (safe and short robot paths, avoid joint-limits and singularities), to propose optimal solutions.

In this kind of planning, organ deformation is a hard problem and a limitation factor. Deformations can occur either from breathing or from the force applied during needle insertion. However, in clinical routine, the needle insertion is done

under shallow breathing, causing deformations which do not exceed 3mm [12]. Similar observations were reported when using gating techniques [8]. So, respiratory motion can be significantly reduced intraoperatively. Irregular motions like coughing can also be addressed intraoperatively. In [10], our team also proposed to anticipate deformations due to the force of the needle insertion by integrating simulations in the optimization loop. In future works, it would also be interesting to investigate the use of such simulations or monitoring of needle insertion with an imaging device to update the trajectory or even constrain more the accessibility map by forcing the needle placements to have a margin around it where the needle can be rotated freely without any risk to collide the robot with the CT gantry. Another interesting way to account for deformability in future works would be to perform a monitoring of the target, obstacles, and needle locations intraoperatively, and update the preoperative planning in real time.

5 Conclusion

We presented a decision support system for the planning of robotic assistance to needle insertion. It combines techniques from anatomy-based needle placement planning and robotics path planning. Our approach, based on inverse kinematics, RRT, and an intelligent exploration of the set of candidates, allows to provide an accessibility map. For the end-effector model used in this test, which was intentionally designed as quite rough and bulky, the algorithm was able to find an access area of at least 460 mm^2 in 14/16 cases, which means that an actual end-effector fitting into this bounding volume would be usable in the same cases. Our algorithm is useful to test 3D models of end-effectors before manufacturing in realistic conditions. In future works, in addition to ensuring the accessibility, it would be interesting to optimize the robot path, for instance maximizing the distance of the robot path to the obstacles, or minimizing the proximity to singularities of the manipulator.

References

1. 3D-IRCADb (3D Image Reconstruction for Comparison of Algorithms Database), accessed 2017. URL `http://www.ircad.fr/research/3dircadb/`

2. "iSYS 1", accessed 2017. URL http://interventionalsystems.eu/products/
3. "Robio", accessed 2017. URL http://www.perfinthealthcare.com
4. Altrogge, I., Preusser, T., Kröger, T., Büskens, C., Pereira, P., Schmidt, D., Peitgen, H.: Multiscale optimization of the probe placement for radiofrequency ablation. Academic Radiology **14**(11), 1310–1324 (2007)
5. Baegert, C., Villard, C., Schreck, P., Soler, L.: Multi-criteria trajectory planning for hepatic radiofrequency ablation. In: proceedings of MICCAI'07, *Springer LNCS*, vol. 4791, pp. 584–592 (2007)
6. Bischoff, R., Kurth, J., Schreiber, G., Koeppe, R., Albu-Schffer, A., Beyer, A., Eiberger, O., Haddadin, S., Stemmer, A., Grunwald, G., Hirzinger, G.: The KUKA-DLR lightweight robot arm - a new reference platform for robotics research and manufacturing. In: 41st International Symposium on Robotics, pp. 1–8 (2010)
7. Chen, C.C., Miga, M., Galloway, R.: Optimizing electrode placement using finite-element models in radiofrequency ablation treatment planning. IEEE Transactions on Biomedical Engineering **56**(2), 237–245 (2009)
8. Clifford, M.A., Banovac, F., Levy, E., Cleary, K.: Assessment of hepatic motion secondary to respiration for computer assisted interventions. Computer Aided Surgery **7**(5), 291–299 (2002)
9. Gottschalk, S., Lin, M.C., Manocha, D.: OBBTree: A hierarchical structure for rapid interference detection. In: 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '96, pp. 171–180. ACM (1996)
10. Hamzé, N., Peterlík, I., Cotin, S., Essert, C.: Preoperative trajectory planning for percutaneous procedures in deformable environments. Computerized Medical Imaging and Graphics **47**, 16–28 (2016)
11. Hungr, N., Baumann, M., Long, J.A., Troccaz, J.: A 3-d ultrasound robotic prostate brachytherapy system with prostate motion tracking. IEEE Transactions on Robotics **28**(6), 1382–1397 (2012)
12. Korin, H.W., Ehman, R.L., Riederer, S.J., Felmlee, J.P., Grimm, R.C.: Respiratory kinematics of the upper abdominal organs: a quantitative study. Magnetic resonance in medicine **23**(1), 172–178 (1992)
13. Krieger, A., Song, S.E., Cho, N.B., Iordachita, I.I., Guion, P., Fichtinger, G., Whitcomb, L.L.: Development and evaluation of an actuated mri-compatible robotic system for mri-guided prostate intervention. IEEE/ASME Trans. on Mechatronics **18**(1), 273–284 (2013)
14. Kuffner, J.J., LaValle, S.M.: RRT-connect: An efficient approach to single-query path planning. In: proceedings of IEEE International Conference on Robotics and Automation (ICRA'00), vol. 2, pp. 995–1001 (2000)
15. Masamune, K., Sonderegger, M., Iseki, H., Takakura, K., Suzuki, M., Dohi, T.: Robots for stereotactic neurosurgery. Advanced Robotics **10**(4), 391–401 (1995)
16. Maurin, B., Bayle, B., Piccin, O., Gangloff, J., de Mathelin, M., Doignon, C., Zanne, P., Gangi, A.: A patient-mounted robotic platform for ct-scan guided procedures. IEEE Transactions on Biomedical Engineering **55**(10), 2417–2425 (2008)
17. Piccin, O., Sieffert, J., Schmitt, F., Barbé, L., Meylheuc, L., Nageotte, F., Bayle, B.: Design and characterization of a novel needle insertion tool. In: IEEE International Conference on Biomedical Robotics and Biomechatronics, pp. 266–271 (2016)
18. Ren, H., Campos-Nanez, E., Yaniv, Z., Banovac, F., Abeledo, H., Hata, N., Cleary, K.: Treatment planning and image guidance for radiofrequency ablations of large tumors. IEEE Journal of Biomedical and Health Informatics **18**(3), 920–928 (2014)
19. Schumann, C., Rieder, C., Haase, S., Teichert, K., Süss, P., Isfort, P., Bruners, P., Preusser, T.: Interactive multi-criteria planning for radiofrequency ablation. Int J CARS **10**(6), 879–889 (2015)
20. Seitel, A., Engel, M., Sommer, C., Redeleff, B., Essert-Villard, C., Baegert, C., Fangerau, M., Fritzsche, K., Yung, K., Meinzer, H.P., Maier-Hein, L.: Computer-assisted trajectory planning for percutaneous needle insertions. Medical Physics **38**(6), 3246–3260 (2011)
21. Stoianovici, D., Cleary, K., Patriciu, A., Mazilu, D., Stanimir, A., Craciunoiu, N., Watson, V., Kavoussi, L.: AcuBot: a robot for radiological interventions. IEEE Transactions on Robotics and Automation **19**(5), 927–930 (2003)
22. Tilak, G., Tuncali, K., Song, S.E., Tokuda, J., Olubiyi, O., Fennessy, F., Fedorov, A., Penzkofer, T., Tempany, C., Hata, N.: 3T MR-guided in-bore transperineal prostate biopsy: A comparison of robotic and manual needle-guidance templates. Journal of Magnetic Resonance Imaging **42**(1), 63–71 (2015)

23. Walsh, C.J., Hanumara, N.C., Slocum, A.H., Shepard, J.A., Gupta, R.: A patient-mounted, telerobotic tool for ct-guided percutaneous interventions. ASME Journal of Medical Devices **2**(1) (2008)
24. Wampler, C.W.: Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. IEEE Transactions on Systems, Man, and Cybernetics **16**(1), 93–101 (1986)
25. Wolf, I., Vetter, M., Wegner, I., Bttger, T., Nolden, M., Schbinger, M., Hastenteufel, M., Kunert, T., Meinzer, H.P.: The medical imaging interaction toolkit. Medical Image Analysis **9**(6), 594–604 (2005)