

# On-line motion planning for medical applications

C. Burghart, C. Wurll, D. Henrich, J. Raczkowski, U. Rembold and H. Wörn

Institute of Process Control and Robotics

University of Karlsruhe

76128 Karlsruhe, Germany

[burghart, wurll, dHenrich, rkowsky, rembold, woern@ira.uka.de]

**Abstract** – Enhancing the quality of surgical interventions is one of the main goals of surgical robotics. Thus we have devised a surgical robotic system for maxillofacial surgery which can be used as an intelligent intraoperative surgical tool. Up to now a surgeon preoperatively plans an intervention by studying twodimensional X-rays, thus neglecting the third dimension. In course of the special research programme "Computer and Sensor Aided Surgery" a planning system has been developed at our institute, which allows the surgeon to plan an operation on a three-dimensional computer model of the patient. Transposing the preoperatively planned bone cuts, bore holes, cavities, and milled surfaces during surgery still proves to be a problem, as no adequate means are at hand: the actual performance of the surgical intervention and the surgical outcome solely depend on the experience and the skill of the operating surgeon. In this paper we present our approach of a surgical robotic system to be used in maxillofacial surgery. Special stress is being laid upon the modelling of the environment in the operating theatre and the motion planning of our surgical robot.

## I. INTRODUCTION

Using a robot in surgery implies an adequate modelling of the patient and the intraoperative environment, the operating field. As soon as a surgical robot autonomously performs some steps of the surgical intervention like the cutting of bone, an adequate and safe motion planning has to be employed. The issue of robot motion planning in static environments has been studied for a couple of decades and many important contributions to the problem have been made. However, motion planning in medical surgery can often only be solved in dynamic environments, which enhances the complexity of the problem enormously. Therefore, powerful on-line motion planners for surgical robots with six degrees of freedom (DOF) are needed. The motion planner requires *on-line* capability in order to immediately react to dynamic changes in the environment (i.e. movements of the patient or the surgical instruments) instead of performing any time consuming off-line computations.

### A. Overall Concept

Before applying a robot in surgery a specific three-dimensional patient model is needed. The required patient data are gained by CT-scans which form the basis of

both a volume and a surface model (Fig. 1). Then a surgical planning tool [1, 2] is used in order to preoperatively plan every step of a surgical operation and compute bone cuts, bore holes, cavities, and milled free formings.

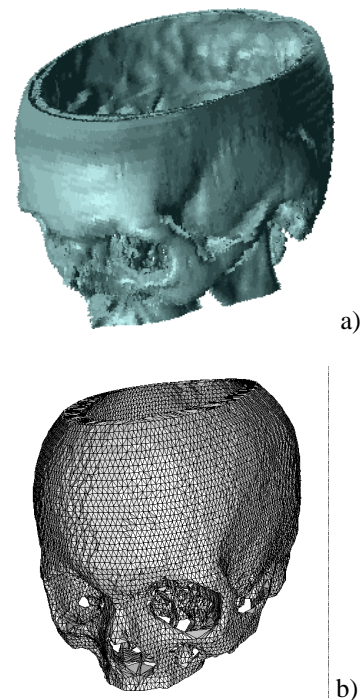


Fig. 1. The volume model a) and the triangulated surface model b) of a patient's skull gained from the patient's CT data.

Generating the robot's collisionfree trajectory requires an appropriate up to date environment model. As modelling the patient's soft tissue would be too time and memory consuming we have chosen a different approach by modelling the soft tissue indirectly [3]. During each intervention the wound is held open by surgical hooks, in the wound itself the prepared bone can be seen. If the positions of the patient's bone, the surgical hooks, the intended bore holes or bone cuts, the surgical robot, and the highest points of the patient are known, an environment model of the operating field can be computed.

In our set up an infrared navigation system intraoperatively detects and monitors the positions of the patient's skull, the robot's tool and the surgeon's instruments (Fig. 2), as all devices and objects are equipped with infrared diodes. Preoperatively small titanium screws have been implanted into the patient's skull, which can be segmented in the CT and MRI scans. During surgery, infrared diodes are mounted onto the screws, and thus the patient's head and all preplanned cutting paths and bore holes

can be described with respect to the coordinate frame of the navigation system. As the robot's endeffector is fitted with infrared diodes as well, the top of the robot's tool can be described with respect to both the robot's base and the navigation system [4].

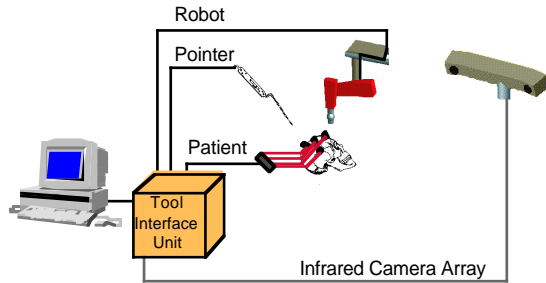


Fig. 2. Intraoperative monitoring concept.

All types of surgical hooks applied in maxillofacial surgery have been modelled and their data have been stored in a data base. At present the surgical hooks are not fitted with infrared diodes. Thus cone shaped holes have been milled into their handles in order to detect a hook by using the infrared pointer. A specific surgical hook is now identified by the relative position of its three artificial landmarks.

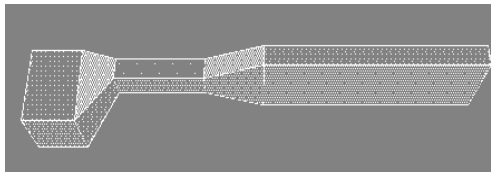


Fig. 3. Model of a surgical hook.

The infrared pointer is also used to indicate the highest points in the operating field, like the patient's nose. As soon as all relevant data are known the environment model can be generated, and the motion planner can compute the trajectory of the surgical robot.

## II. ENVIRONMENT MODELLING

Modelling an adequate intraoperative environment is a prerequisite for using a surgical robot. As the modelling of soft tissue structures is too complex and time consuming, a different approach has been chosen, which is based on an indirect modelling of soft tissue. Various hooks are applied in the operating field in order to hold back the soft tissue. All hooks applicable in maxillofacial surgery have been modelled and have been stored in a database.

During surgery, the positions of both the patient's skull and the hooks can be detected by using the infrared navigation system. The highest points within the operating field are registered by using the infrared pointer, too. Then a convex hull of the patient and the surgical hooks is generated modelling the surgical wound as well.

The established convex hull can then be imported into our operating theatre modelled in ROBCAD, a robot simulation system. The set up created in ROBCAD and the

preoperatively planned cutting paths then are transferred to the on-line motion planner. Knowing all obstacles, the current position of the robot and the desired goal position (start point of the cutting path or position of the intended bore hole) the motion planner computes a collision-free path for our surgical manipulator [5].

### A. Algorithm

As soon as the appropriate models of the surgical hooks have been identified the neighbours of each hook have to be computed. Each hook possesses a front plane facing other hooks in the wound and a back plane at the end of the handle. For this purpose a specific coordinate frame is generated with the goal position of the robot as origin and the required orientation of the robot's tool as z-axis. The x-axis is perpendicular to the z-axis and includes a vertice of the front plane of the first surgical hook. The y-axis is the vector product of z-axis and x-axis.

In a next step all faces of two neighbouring surgical hooks facing each other are computed and divided into groups. Actually only the edges of the found groups of faces are considered. Then new faces connecting the neighbouring hooks are generated.

The edges of the afore mentioned groups of faces are searched in opposite directions. Then triangles connecting two neighbouring hooks are generated alternately. Sometimes an edge has to be smoothed by constructing an additional surface triangle or the alternating order has to be upset in order to fusion the hooks. Finally the boundary connecting the back sides of the hooks is generated.

The convex hull of the patient and the connected hooks are fused in the following manner: the edge of the afore generated fusion of the surgical hooks is projected onto the convex hull of the patient. Edges on the same side as the required orientation of the robot's surgical instrument are projected in direction of that direction, otherwise the edges are projected in direction of the z-axis of the convex hull which is congruent to the z-axis of the world coordinate frame. The generated polygon on the surface of the convex hull is cut out. Then the edge of the connected hooks and the projected polygon are connected by triangles. Finally the edges of the top surface of the convex hull which has been deleted are connected to the projected polygon.

### B. Simulation in Geomview

Before experimenting with the real set up in our laboratory a simulation platform based on Geomview was developed. On the one hand an environment can be created by loading the triangulated patient data set and choosing the desired surgical hooks. The positions and orientations of the hooks can be varied by using the mouse. As soon as a new configuration of the patient and the hooks is detected or confirmed by the user the environment model is computed.

On the other hand the developed platform offers the option to detect and identify the surgical hooks and to monitor the position of the patient by the infrared

navigation system. The generated environment model can then be integrated in the simulated operating theatre on a ROBCAD server or it can be directly transferred to the motion planner. In the following the modelling of the operating field by our platform is illustrated.

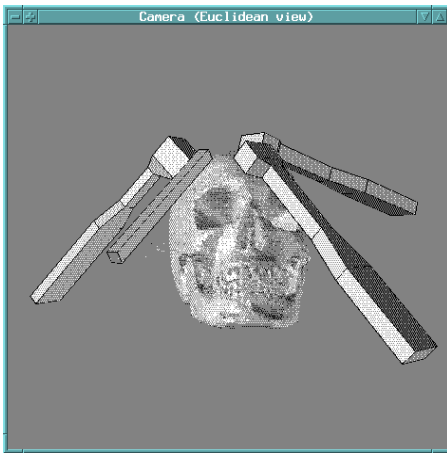


Fig. 4. Surface model of the patient's skull and the detected surgical hooks needed to hold back the soft tissue.

At the beginning the surgeon chooses the required patient data set and the surgical hooks to be applied (Fig. 4).

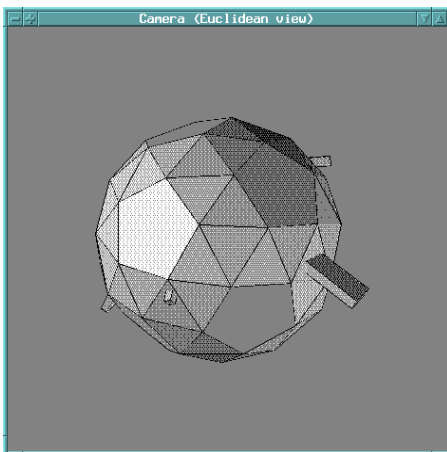


Fig. 5. Simple convex hull around the patient's skull.

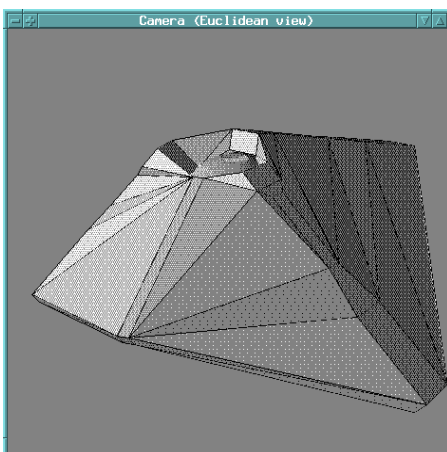


Fig. 6. Fusion of the surgical hooks and modelling of the surgical wound.

The hooks are registered either by mouse or by the infrared pointer.

Then the convex hull of the patient is computed (Fig. 5) and displayed on the monitor. The surgical hooks are fused (Fig. 6) and the surgical wound is modelled. For better understanding the connected hooks and the triangulated skull are depicted in Fig. 7.

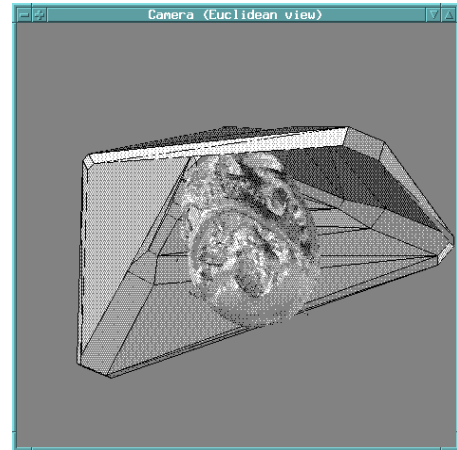


Fig. 7. Fused surgical hooks and surface model of the patient's skull.

Then the edge of the fused hooks is projected onto the convex hull of the patient. Faces directly lying in the shade of the fused hooks are deleted as they are not needed for further computation of the environment (Fig. 8).

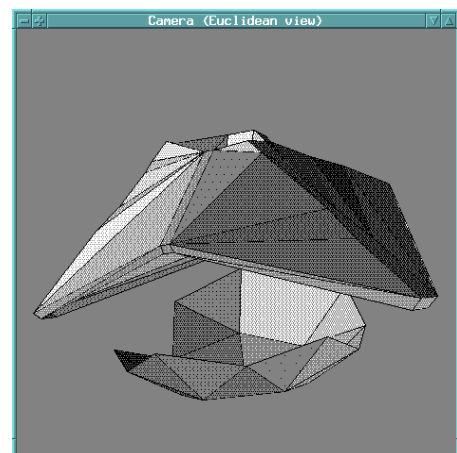


Fig. 8. Fusion of the hooks and the convex hull of the patient's skull. Faces that are covered by the fused hooks are not contemplated by our algorithm.

In Fig. 9 a different view of the fused hooks and the convex hull of the patient is displayed. It can be clearly seen that now the edges of the faces composing the lower part of the convex hull and the edge of the fused hooks will be connected by additional triangles in order to create an environment model.

A close approximation of the patient is not intended, as a fusion of hundreds of thousands triangles would be too time consuming. A surface model of the patients skull generated by our approach consists of about 40.000 surface triangles [6]. Other mesh generation methods need more than 200.000 triangles to model a skull.

Thus we have simplified our method for modelling the operating field even more. Instead of using a convex hull of the patient's skull we generate a convex hull of the whole patient (Fig. 10).

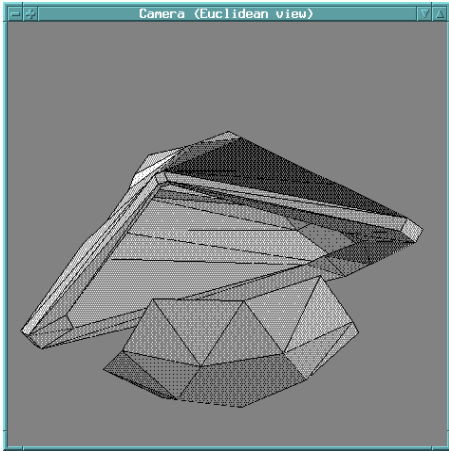


Fig. 9. View of the fused hooks and the convex hull of the skull from underneath.

Before the surgical intervention starts the convex hull of the patient is initialized by default values (width and length of the operating table and a height of 30 cm). During the operation the surgeon has to use the infrared pointer in order to indicate the boundaries and the highest points of the patient.

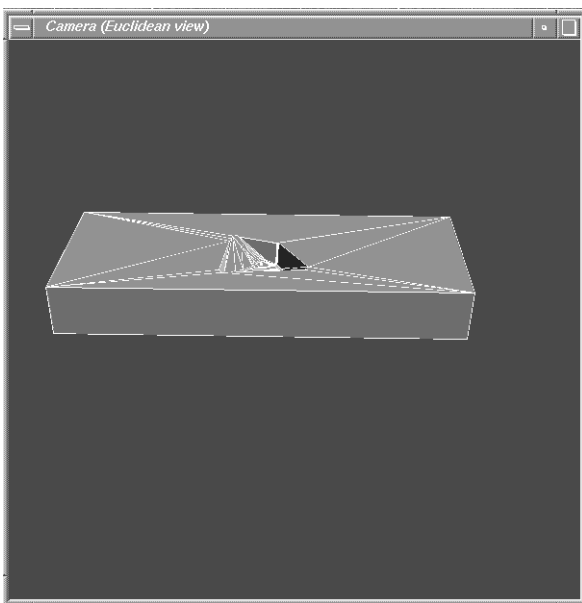


Fig. 10. Simplest environment model of the operating field.

Then the surgical hooks are detected and the fusion of the hooks and the convex hull is computed as depicted above.

### C. Integration into ROBCAD

The generated environment model is integrated into the set up of the operating theatre which is simulated by ROBCAD on an SGI workstation (Fig. 11). Obstacles modelled in ROBCAD can then be easily transformed into the format required by the motion planner.

Actually the motion planner only needs to be currently informed about the state of the operating field. Other objects which remain static during the whole intervention are combined to a so called benchmark which is transformed into the format of the motion planner only at the beginning of the surgical intervention.

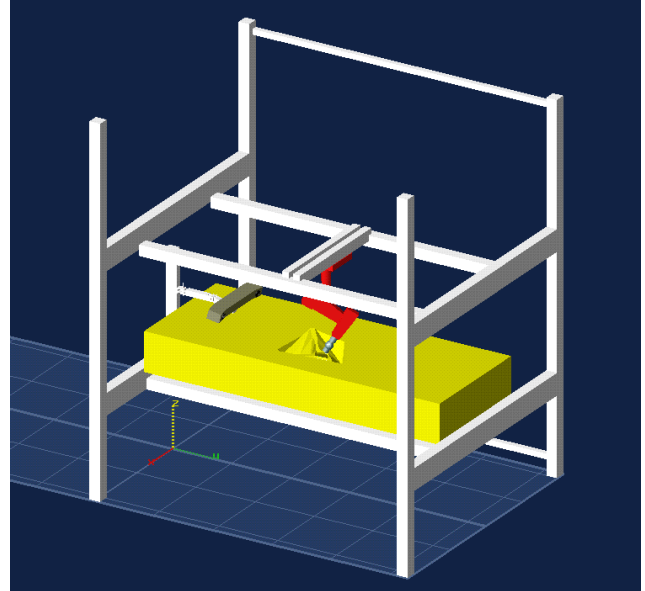


Fig. 11. ROBCAD-screenshot of the surgical workcell with the patient's convex hull and the surgical robot.

As soon as the positions of the patient or a surgical hook changes, a new environment model is generated, displayed in ROBCAD and sent to the motion planner.

ROBCAD is also used as a visualization tool for the planned trajectories of the motion planner which can be run parallelly to the robot's actual performance of the trajectory.

In order to take a short cut the generated environment model can be transformed directly into the format required by the motion planner without being modelled in ROBCAD.

## III. MOTION PLANNER

Avoiding time consuming obstacle transformations is managed by our motion planner which searches a solution in the configuration space ( $C$ -space) and checks for collisions in the workspace. Thus our motion planner is able to cope with dynamic environments like a medical scenario.

In order to search the implicit  $C$ -space which is set up by the robot's joint values we apply the well known  $A^*$ -search algorithm [7]. The main task of the  $A^*$ -algorithm consists of the expansion and the processing of configurations which are saved in the priority list OPEN. In every iteration, the best configuration of OPEN is expanded. According to a heuristic evaluation function, these successors will be considered in the succeeding iterations. After the expansion, the parent configuration is saved in the hashing table CLOSED. The search continues until the goal is found or the OPEN list is empty. In the

latter case the algorithm stops without a solution. In Fig. 12 an example for a 2D search is given. The dots indicate investigated configurations and the arrows refer to the corresponding successors.

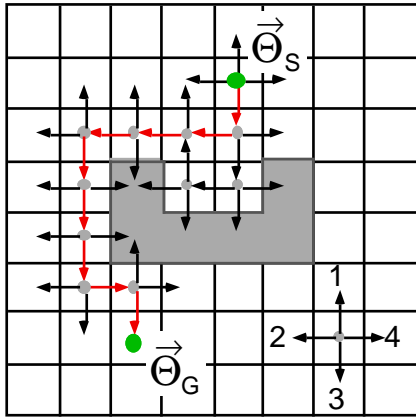
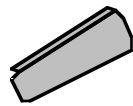


Fig. 12. A\*-search in the implicit C-space from the start configuration  $\vec{\Theta}_S$  to the goal configuration  $\vec{\Theta}_G$ .



- [6] Grabowski H., Burghart C.: "Generating Finite Element Meshes from Volumetric Medical Images", In Proceedings of the 2nd IARP International Workshop on Medical Robotics, Heidelberg, 1997.
- [7] Hart P.E., Nilsson N.J., Raphael B.: „A formal basis for the heuristic determination of minimum cost paths“, IEEE Trans. Syst. Sci. Cybern, pp. 100-107, 1968 -> A\*, Ref. aus [Korf90].
- [8] Henrich D., Cheng X., "Fast Distance Computation for On-line Collision Detection with Multi-Arm Robots", IEEE International Conference on Robotics and Automation, Nice, France, May 10.-15., pp. 2514-2519, 1992.
- [9] Qin C., Henrich D.: „Path planning for industrial robot arms - A parallel randomized approach“, In Proc. of the International Symposium on Intelligent Robotic Systems (SIRS '96), Lissabon, Portugal, pp. 65-72, July 22-26, 1996.
- [10] Henrich D., Wurll Ch., Woern H.: "Multi-directional search goal switching for robot path planning ". In: The 11th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems (IEA/AIE'98), Castellon, Spain, July 1-4, 1998.
- [11] Henrich D., Wurll Ch., Woern H.: "Heuristic Hierarchical Motion Planning in Dynamic Environments". IECON'98: The 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, Germany, August 31 - Sept. 4, 1998.
- [12] Henrich D., "Parallel processing approaches to motion planning - An overview", IEEE Int. Conf. on Robotics and Automation, Minnesota, USA, pp. 3289-3294, 1996.
- [13] Henrich D., „Fast motion planning by parallel processing - A review“, Accepted for: Journal of Intelligent and Robotic Systems, 1997.