Towards Automated Surgical Robotics: a Requirements Engineering Approach

Marcello Bonfè, Fabrizio Boriero, Riccardo Dodi, Paolo Fiorini, Angelica Morandi, Riccardo Muradore,
Liliana Pasquale, Alberto Sanna and Cristian Secchi

Abstract— **The paper describes a design specification process for the development of novel and intelligent surgical robots. Nowadays, surgical robots are usually controlled by the surgeons manually by using teleoperation. The possibility to carry out simple surgical actions automatically has been the subject of academical research, but very few real-world applications exist. The main objective of this research is to address realistic case studies and develop systems and methods to provide surgeons with autonomous robotic assistants, performing basic surgical actions by combining sensing, dexterity and cognitive capabilities. This goal can only be achieved by means of a formal and rigorous assesment of surgical requirements, so that they can be analysed and translated into behavioral specifications for an autonomous robotic system. Therefore, the paper describes the application of Requirements Engineering to surgical knowledge formalization and propose a methodology for the transformation of requirements into formal models of robotic tasks.**

I. INTRODUCTION

Robot assistants, like surgeon extenders and auxiliary surgical supports [1], provide a fundamental aid to the surgeons in the execution of critical surgical tasks. Surgeon extenders are teleoperated by surgeons and allow them to achieve superhuman performance in terms of scale, accessibility and teamwork distance, as in the case of Da Vinci®by Intuitive Surgical™[2]. Surgical supports perform auxiliary functions, such as endoscope holding [3] or suture thread cutting [4].

Fully automated robotic assistants have been developed and tested in different medical scenarios [5], even if operations on real patients are quite rare in most surgical specialties. A notable exception is represented by orthopedics (i.e. bone drilling and hip/knee replacement, see also the commercial system ROBODOC®[6]), which is compatible with robotics because rigid tissues are treated. Interaction with other biological tissues, in fact, is much more challenging for a computer-controlled robot. On the other hand, thanks to recent advances in cognitive sciences, sensing technologies and control techniques, modern robotic systems can behave smoothly and gently in many contexts, especially those requiring human interaction [7]. Therefore, times are ripe enough to start thinking of intelligent robotic assistants that can automatically perform selected surgical tasks on the patient, under the supervision of human surgeons. Needless to say, to let the robot take charge of even the simplest surgical gesture it is necessary to acquire human knowledge and experience and tranform it into machine-readable behavioral models.

Several works [8], [9] have already proposed strategies for modeling the behavior of the surgeon for skill assessment purposes. However, these strategies do not explicitly address the problem of building a model for the surgical task so that it can be executed by a robot and verified by means of formal methods. On the other hand, other authors studied robotic automation of tasks requiring high dexterity, particularly knot tying, by applying iterative learning techniques [10] or models mimicking fluid dynamics [11], but without considering interaction with soft tissues and considering only manual gestures, not complete surgical procedures.

This paper contributes to the automation of robotic surgery on soft tissues presenting the preliminar, but essential, task of surgeons knowledge modeling, aiming to describe requirements on the execution of a surgical task, the parameters that must be monitored during the operation and the way in which main complications must be handled. In particular, this research addresses the following surgical operations:

- percutaneous cryoablation of small tumoral masses (i.e. *puncturing / needle insertion*), with specific interest in kidney cancer treatment;
- execution of a laparotomy (i.e. *cutting*), to enable open abdominal surgery;
- *suturing* of a planar wound.

To decouple the formalization of surgical procedures and technical development, a two-stage procedure is proposed. In a first stage, requirements for the surgical process are captured using a goal-oriented methodology, whose advantage is that it is focused on the real objectives of the operation and the complications that may arise during its execution. Moreover, the proposed modeling methodology is based on a formal language, allowing rigorous translation into a complete behavioral specification in the form of a state machine. In a second stage, the knowledge acquired and formalized in the previous step is translated into a

---

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement n. 270396 (I-SUR)

M. Bonfè is with the Engineering Department (ENDIF), University of Ferrara, Italy. E-mail: `marcello.bonfe@unife.it`

F. Boriero, P. Fiorini and R. Muradore are with the Department of Computer Science, University of Verona, Italy. E-mail: `name.surname@univr.it`

A. Morandi, R. Dodi and A. Sanna are with Fondazione Centro San Raffaele del Monte Tabor, Milano, Italy. E-mail: `name.surname@hsr.it`

C. Secchi is with the Deparment of Science and Methods for Engineering (DISMI), University of Modena and Reggio Emilia, 42122 Reggio Emilia, Italy. E-mail: `cristian.secchi@unimore.it`

L. Pasquale is with Lero - The Irish Software Engineering Research Center, Ireland: `liliana.pasquale@lero.ie`

specification that can be used for control design, namely by partitioning the complete state model into sub-tasks that can be assigned to robotic assistants.

The goal-oriented methodology used to represent the requirements for the surgical actions is called FLAGS (Fuzzy Live Adaptive Goals for Self-adaptive systems [12]). The main feature of FLAGS, facilitating the engineering of requirements, is that it supports hierarchical decomposition of high level objectives until the basic tasks, such as those that can be executed by a robot, are identified. Other authors [13], [14] applied requirements engineering to the medical field. In particular, these approaches use model checking techniques to find and correct flaws in surgical protocols. Conversely, our objective is to create a knowledge base and reasoning framework to support control design for autonomous surgical robots. In particular, actions, sequences and non-functional requirements, expressed by the goal model, are transformed respectively into trajectories that can be tracked by a robot, state machines for control logic, safety bounds and guard conditions. The expected result of this task is a model that can be easily mapped into a software architecture, based on state-of-the-art robotic frameworks (e.g. Orocos [15]). The modeling language used in this phase is UML (Unified Modeling Language) [16], which is currently the most known formalism for Model-Based Software Engineering (MBSE). Since detailed control software design is out the scope of the paper, UML models will be used mainly to describe how goal requirements relate to high-level structural and behavioral decomposition of the intelligent robotic system.

The rest of the paper is organized as follows: Section II briefly introduces the case studies selected as overall objectives of the research project; Section III describes the requirements engineering methodology; Section IV shows the UML design that models the intelligent robotic system being developed; Section V contains some preliminar experimental results on the puncturing case study, that will be used to instantiate the behavioral model with engineering details, such as safety bounds on interaction forces or expected stiffness of the punctured tissues.

II. SURGICAL CASE STUDIES

It is commonly acknowledged by surgeons that humans are not always able to guarantee either high precision and fast execution of surgical operations. For this reason, the final objective of this research is to demonstrate the feasibility of robotic automation for a given set of surgical procedures, which have been selected among those that would benefit most from the accuracy and repeatability levels that could be achieved by intelligent robots. As said before, the three chosen actions are: needle insertion for ablating procedures, laparotomy cut and suture of a wound. For simplicity reasons, in this paper we will refer our analysis to the first case, even if the same methodological approach has been applied to the others and three full task specifications for an intelligent surgical robot have been obtained.

Percutaneous cryoablation requires the use of pre- and intra-operative images (CT, MRI/US) to precisely place one or more cryoprobes directly through the skin into the tumoral mass to be destroyed - in this case, a Renal Cell Carcynoma - and to check the real-time position of the tools inside the patient. Trajectory misalignments are usually due to the deformation of soft tissues and kidney displacement because of the respiration (breathing will be not considered in the early experimental stage of the project as simplification). By using real-time image registration and a precisely calibrated mechanical arm, needle insertion would be executed much more efficiently by the robotic system than by humans.

The complete cryoablation operation involves, once that needles are inserted, cycles of freezing and thaw to create an *iceball* covering and killing the tumoral cells, while preserving the healthy tissue and the surrounding abdominal structures [17], [18]. The most common complications are pain or paresthesia related to the probe insertion site (64%) [19]. Major complications refer to bleeding and renal fractures caused by the extraction of the probes when the ice ball is not melted enough to release the needles. The evaluation of the force applied to extract a needle from the patient is, for human surgeons, the only way to detect the melting status of the iceball. However, only personal experience allows surgeons to correctly analyse this situation. Even in this case, robotic assistants and advanced intraoperative processing of US images would increase safety margins during completion of the surgical procedure.

III. REQUIREMENTS ENGINEERING METHODOLOGY

The proposed methodology, to collect the requirements for a software-intensive and intelligent surgical robot and translate them into control-oriented specifications, is composed of several steps. The main phases of the process and their outcomes can be schematized as shown in Figure 1. The process is applied to each one of the surgical tasks to be supported by the robotic system, so that at the end a set of *operating plans* is obtained.

The methodology starts by interviewing a group of expert surgeons to collect the objectives of the operation, the main procedures ("best practice") to be performed, the elements of the domain and the critical events related to the surgical action. The information collected through the interviews is used to represent the *goal model*. Such a model is converted into a set of formal properties using the *Alloy language* [20], a specification language for expressing complex structural constraints and behavior in a software system, based on first-order logic. This formalization enables automated analysis, by means of *model-finding* algorithms [21], whose result is a state machine compatible with the goal model specification and representing the whole operation from a surgical point of view. Finally, this state model is refined, by using UML as a modeling language (see Section IV), to obtain a software-oriented design specification. The process from interviews to software modeling is called *High-level knowledge representation* and, as outcome, provides two kind of UML diagrams: a Statechart, describing the required robot operating sequence, and the Sequence diagram, describing the interactions among the components of the intelligent system

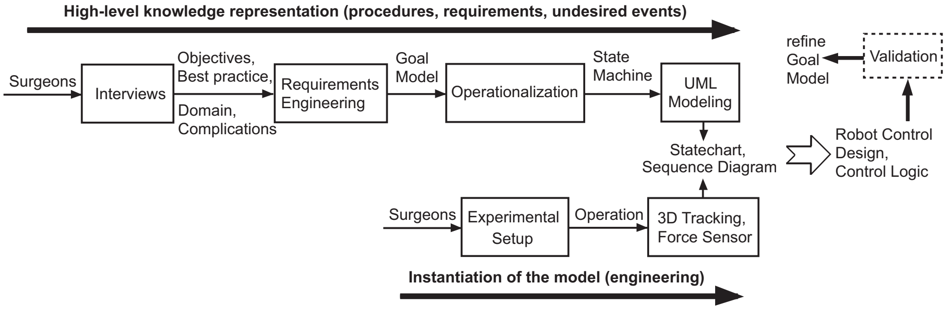

Fig. 1. The procedural flow of the proposed methodology.

and between the system and human users. Such diagrams still require fundamental links with the real operations that must be carried out by the robots. More precisely, it is necessary to *instantiate* these models by taking into account elements like spatial trajectories, bounds, force profiles. This second process is called *Instantiation of the model* in Figure 1. In this way the abstract model is endowed with all the technical and engineering details needed by the robot to performing autonomously the surgical procedure. Of course, the task model must be validated, on the basis of experiments on phantoms or virtual environments and subsequent evaluation of the task execution, by expert surgeons. If necessary, the validation procedure may trigger a refinement of the goal model, to obtain a more precise specification of the surgical requirements.

Expressing the requirements for surgical actions is not an easy task. In this context, using a goal model [22] seems to be the right reasoning abstraction to link the objectives of the doctors, expressed in the medical language, and the requirements for the underlying system. As a matter of facts, doctors naturally think in terms of goals, operations to perform during the surgery, and patients vital parameters that have to be controlled. Goals also facilitate the design of sensing and monitoring activities, as the domain model identifies the entities of interest, the events that it is necessary to monitor and the exceptions that might happen during the execution of the surgical procedure. Requirements have been represented through the FLAGS [12] model, which extends the KAOS (Knowledge Acquisition in autOmated Specification [22]) model with a new kind of goals, the adaptation goals, specifying how the model can adapt to changes (i.e. adaptation goals represent both exceptions and their countermeasures). The FLAGS model includes four different sub-models: the domain model, the goal model, the operation model and the adaptation model. The domain model represents the entities and the events of the environment that it is possible to control or observe. Entities are objects providing an informative content and they can evolve through a set of states, depending on the value assumed by their attributes, while events correspond to something that may happen during the execution of the system. An example of entity is the *Needle*, which can be in different states depending on whether it has being selected, inserted or extracted, and related events may be the completion of a desired movement (e.g. the needle is fully inserted and touches the tumor) or an undesired event indicates that a needle touched a forbidden region (e.g. a bone, a nerve or another organ not affected by the operation).

The goal model specifies the main objectives of the doctors and refines them as a hierarchical set. High level goals can be related to several conjoined subgoals (AND-refinements) or to alternative combinations of subgoals (OR-refinements). Leaf goals representing a functional requirement can be further decomposed into a set of operations, which are described in the operation model. Each goal is formally defined using LTL (Linear Temporal Logic [23]). As an example, the main goal of a cryoablation procedure, defined as *Execute puncturing* and marked as G1.2, has an AND-refinement into 3 subgoals. The first sub-goal (G1.2.1) requires all needles to reach a specific position in the tumoral mass. The second (G1.2.2), that when all needles have been inserted, the ablation cycle must be performed. The third one (G1.2.3) states that if an iceball has been created, the doctor must verify whether the tumor has been completely destroyed. In its turn, goal G1.2.1 has also an AND-decomposed into three sub-goals, namely requiring that the target position for each needle is within the tumoral mass (G1.2.1.1), that every time a movement is performed the needle must not touch forbidden regions (G1.2.1.2), and that only one needle can be active at a time (G1.2.1.3). Each leaf goal is associated with a formal definition in LTL. For example, goal G1.2.1.2 is specified as follows: $mp => !(fr \ \wedge \ (fr.needle = mp.needle))$, a formula asserting that every time a movement is performed (event *mp*), no forbidden region must be touched (event *fr*) by the *needle* entity associated to both elements of the model.

The operation model represents, in an abstract way, the actions that the robotic tool can perform. Operations are performed on the entities of the domain, can be triggered by an event and can fire other events. Each operation is defined through name, input and output values, and a set of pre- and post-condition, expressed in FOL (First Order Logic). The latter formal specification enables subsequent analysis with the Alloy tool [20] and the automatic generation of a sequential plan achieving the goals.

Finally, the adaptation model represents the countermeasures to cope with undesired situations. Countermeasures are represented through adaptation goals, which are associated to a trigger event (i.e. a one that prevents one or more of the main goals to be satisfied) and a sequence of corrective actions, in terms of elements of the operation model.

The transformation of a goal model into a sequence of operations and adaptations, that satisfy the goals of the surgical procedure, is called *operationalization*. Sequences of operations are constrained according to pre/post-conditions of operations themselves and by the way leaf goals are composed (i.e. goal refinements) for the achievement of their parent goals. Thus, operationalization is a constraint satisfaction problem, whose solution is obtained by using the SAT-solver Kodkod [24] embedded in the Alloy tool. In particular, the information of the goal model are encoded into an Alloy model, then the set of properties of the model is checked, using the model-finding features of Kodkod. As a result, the tool provides a sequential model equivalent to the traces of a state machine, representing the whole system behavior that guarantees the achievement of the root goal.

IV. ROBOTIC SYSTEM DESIGN MODEL

Mapping goal requirements and their operationalization into the design of a real surgical robotic assistant requires to define first of all the overall architecture of the system. The autonomous system designed in this project is composed of the following main entities:

Surgical Interface (SI): a software-intensive system allowing humans (i.e. surgeons and technicians) to play an active role both during the preoperative and the intraoperative phase. In the first one, the focus is on identification of anatomical features of the patients, by means of segmentation of medical images (CT, MRI or Ultrasound), and detailed planning of the surgical intervention. Planning (e.g. enumeration and placement of cryoablation needles for maximal tumor coverage) may be performed by automated reasoning algorithms, but must always be validated by a surgeon. During operations, the interface will provide real-time visual navigation of the surgical scenario, using sensory/image data and augmented reality virtual environments.

Robot Control (RC): the unit responsible for the control of surgical actions and tasks sequencing during the intraoperative phase. The event-driven behavior extracted from the goal model is mapped into the control logic of the robotic system, specified by a UML State Diagram, while continuous control loops must include hybrid force/motion control, based on force sensors mounted on the tip of the robot, and real-time US-based guidance. Safety-critical requirements put a strong demand for strict interaction of this component with both surgeons and technicians, by means of the SI, and the Sensing/Reasoning module during the execution of surgical operations.

Sensing/Reasoning (SR) and Situation Awareness (SA): the role of this composite sub-system module is to acquire and elaborate data from sensing devices, using real-time reasoning algorithms based on probabilistic networks. The outcomes of this sub-system are required to either support the planning task, during the preoperative phase, or provide prompt identification of anatomical changes or discrepancy between the tasks being executed and the *nominal* surgical plan and, consequently, signal the occurrence of undesired events and critical situations, during the intraoperative phase. Such signals must trigger corrective actions, if any exists, to be applied autonomously by the system or, otherwise, force switching to a safe mode and let human surgeons take over. For example, during cryoprobe insertion a small misalignment of the needle can be compensated, thanks to the elasticity of human tissues. However, this requires to adapt accordingly the parameters of the robot controller (i.e. a hybrid force/position loop).

The definition of behavioral elements of the UML model requires a refinement of goal model requirements and, particularly, of the state machine inferred from the goal model as described in Sec. III. In fact, the goal model captures knowledge of the whole surgical procedure, from the *domain* point of view. However, system-level design of the SI, SR/SA and RC units requires to define for each one its role in the surgical tasks and its contribution to the fulfillment of a given goal. In particular, the structural decomposition implies also a behavioral decomposition of the state machine obtained as a result of Goal Model analysis of Sec. III, to allow the specification of each component behavior by means of a UML State Diagram associated to each related classifier.

Focusing on the Robot Control unit, its partition of the *global* state machine must include states and operations strictly related to the execution of surgical gestures, while the analysis of medical imaging, the definition of operative plans or the activation of intraoperative monitoring tasks should be associated with the other system components SI and SR/SA. The interaction among such system components can be described using UML Sequence Diagrams, which represents scenarios compatible with a given collaborative behavioral specification, in the form of a sequence of events, exchanged among a number of *object lifelines*, implicitly ordered in time from top to bottom. As an example, Figure 2 shows an admissible scenario for the cryoablation execution, up to the completion of needles insertion. The scenario includes the adaptation required if *Forbidden Regions* (FRs) are touched by the needle: this event is detected by the Sensing/Reasoning unit, that performs real-time continuous monitoring of the needle motion within an anatomical atlas of the patient and eventually raises an alarm signal (`FRTouched`).

The state machine for the Robot Control system has been designed, compatibly with the prototype state machine obtained from the Goal Model, as the UML State Diagram shown in Figure 3. As can be seen, during the *iceball configuration* and during the execution of the actual cryoablation cycles, the robot motion is not required and, therefore, the Robot Control unit is in a *sleeping state*, waiting for specific external events. Moreover, the hierchical features of UML State Diagrams allow to embed exception handling mechanisms, by means of transitions exiting composite states, like for example those handling the *force limit* exception in the `InsertingNeedle` and `ExtractingNeedle` states.

The proposed UML State Diagram can be translated into executable code using, for example, the Orocos framework

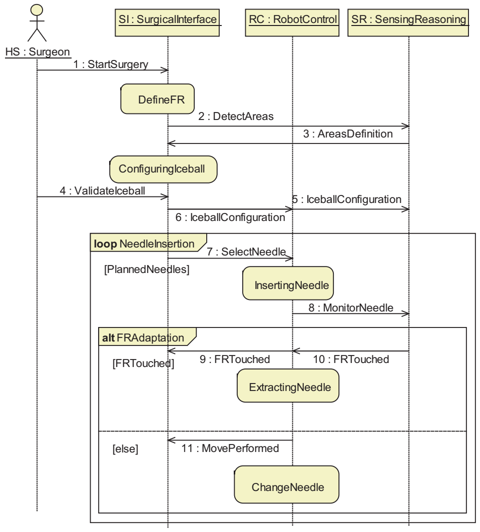

Fig. 2. UML Sequence Diagram of the interaction among system components during a puncturing task for cryoablation: scenario including the *Forbidden Region touched* (event *FRTouched*) adaptation

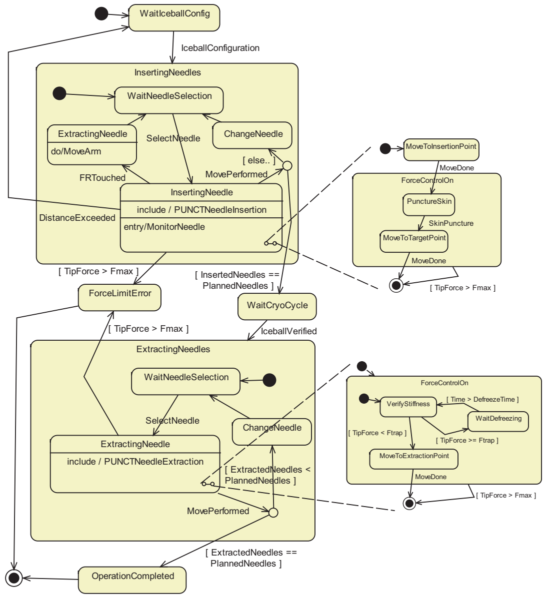

Fig. 3. UML State Diagram of the behavioral specification for the Robot Control sub-system

[15] and dedicated tools for the management of hierarchical state machines [25]. However, the knowledge representation acquired so far lacks of all the details related to cartesian motion trajectories, control primitives and safety bounds expressed in engineering units. The plan of this research project is to collect and specify such kind of quantitative requirements by means of experiments with phantoms and augmented reality simulators. Next section presents a preliminar setup for the exeuction of such experiments.

V. EXPERIMENTS

At the end of the requirements capture and logic control design activities described in previous sections, surgeons knowledge about surgical operations is transformed into a machine-readable sequential specification. However, this description is actually not executable by a robotic assistant, because it lacks crucial quantitative details that cannot be captured during surgeons interviews. As an example, human operators are not able to specify the *nominal* force that is applied during the insertion of a needle or, which is even worst, a safe limit on that force.

For this reason, it is necessary to derive an accurate model of forces exerted on soft tissues during a surgical operation, by means of a data-driven identification procedure. To collect the data, a specific experimental setup is needed for each one of the surgical tasks under study. Considering the cyroablation procedure, if the surgeon performs the operation on a synthetic organ by using a needle equipped with sensors, it would be possible to reconstruct the needle trajectory (position and orientation) and the force (i.e. forces and torques) that the human is applying during the surgical action. This section describes a preliminar set of experiments on this case study, following an approach similar to the one applied in [26].

To measure forces and positions, a modified standard cryoablation needle equipped with a force sensor and some tracking markers is used. The sensor is an Ati Nano-17 6 DOF force-torque sensor, whereas the tracking markers rigidly attached to the needle allows at the NaturalPoint optiTrack tracker to give at run-time position and orientation of the needle. The data have been acquired by means of specifically developed software application based on Orocos, a well-known framework for robotic applications [15], and running on a standard PC with Xenomai real-time operating system. The Ati Nano-17 and the tracker boards acquire data and send the information through standard Ethernet UDP packets. Two Orocos components have been implemented to log forces and positions at 1 kHz sampling frequency.

At the moment, the phantom has not the shape of a human organ, but it is simply composed by three different layers of gel wax, having different percentage of paraffin inclusion and, therefore, different stiffness. At this stage of the research we mainly need to understand if the experimental setup is accurate enough to distinguish different layers (by using the force sensor) and, at the same time, to provide all the information needed to automate the surgical procedure.

Figure 4 shows an example of the kind of information we can retrieve by using the experimental setup. In particular, the experiment refers to a movement of the needle along the z axis. The blue trajectories show the force along the z axis versus the displacement still along z, when the needle is entering into the phantom, whereas the red lines refers to the motion in opposite direction. The slope of the ramps depends on the stiffness of the material and of the velocity of insertion. This means that different trials have different slopes and also different target points. Discontinuities happen when the needle is entering a given layer. However, it is possible to distinguish only the discontinuity between the first and the second layer, whereas the discontinuity between the second and the third layer is not clearly visible. This is due to the different stiffness between the layers. As could be expected the in- and out-trajectory are not symmetric.

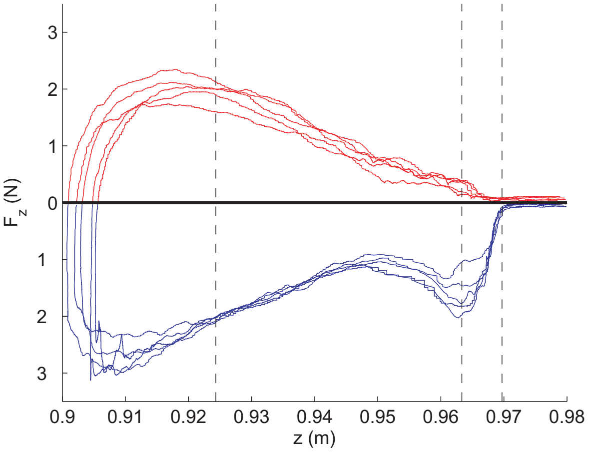

Fig. 4. Plot of the force F_z versus displacement z. The vertical lines correspond to the different layers along the z axis.

By processing the experimental data shown on the plot and the full set of measurements related to spatial coordinates and orientation, it is possible to estimate the nominal trajectory that has to be tracked by the robot and the required bounds on exerted forces. In particular, the upper value of $|F_z|$ plus a conservative safety margin allows to fix the maximal force F_{max} for the insertion and extraction of the needle, which is used in the guard conditions of sub-machines `InsertingNeedle` and `ExtractingNeedle`, embedded in the UML State Diagram of Figure 3. Similar experiments with a realistic setup of a post-cryoablation situation would allow to define the value of the force F_{trap} which can be expected during the extraction of the needle, if the latter is trapped because of incomplete defreezing of the configured iceball.

VI. CONCLUSIONS AND FUTURE WORK

The paper has shown the application of Goal Modeling techniques to the formalizaton of surgical requirements, as a first step towards the automation of intelligent surgical robots. Moreover, the Goal Model obtained with the FLAGS tool has been trasformed as a behavioral specification in the form of a UML model, which can be *prepared* for automatic translation into executable code for robotic control systems. This *preparation* phase pass through experiments with real surgical instruments, during which forces and tools positions are captured and analysed to provide technical details that human surgeons may not be able to state as formally as required. In future works, we aim to replicate such experiments with robots fully equipped and programmed as specified by the proposed formal behavior model.

REFERENCES

[1] R. Taylor, "A perspective on medical robotics," *Proceedings of the IEEE*, vol. 94, no. 9, pp. 1652–1664, September 2006.

[2] http://www.intuitivesurgical.com.

[3] A. Casals, J. Amat, D. Prats, and E. Laporte, "Vision guided robotic system for laparoscopic surgery," in *Proceedings of the IFAC Internationa Congress on Advanced Robotics*, Barcelona, Spain, 1995.

[4] N. Padoy and G. Hager, "3D thread tracking for robotic assistance in tele-surgery," in *Proc. IEEE/RSJ Conference on Intelligent Robots and Systems*, 2011, pp. 2102–2107.

[5] J. Rosen, B. Hannaford, and R. Satava, Eds., *Surgical Robotics: Systems Applications and Visions*. Springer, 2011.

[6] http://www.robodoc.com.

[7] A. Albu-Shäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger, "The DLR lightweight robot: lightweight design and soft robotics control concepts for robots in human environments," *Industrial Robot Journal*, vol. 34, no. 5, pp. 376–385, 2007.

[8] J. Rosen, J. Brown, L. Chang, M.-N. Sinanan, and B. Hannaford, "Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model," *IEEE Transactions on Biomedical engineering*, vol. 53, no. 3, pp. 399–413, March 2006.

[9] T. Neumuth, P. Jannin, G. Strauß, J. Meixensberger, and O. Burgert, "Validation of Knowledge Acquisition for Surgical Process Models," *Journal of the American Medical Informatics Association*, vol. 16, pp. 72–80, 2009.

[10] J. van den Berg, S. Miller, D. Duckworth, H. Hu, A. Wan, X.-Y. Fu, K. Goldberg, and P. Abbeel, "Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA, May 2010, pp. 2074–2081.

[11] H. Mayer, I. Nagy, D. Burschka, A. Knoll, E. U. Braun, R. Lange, and R. Bauernschmitt, "Automation of manual tasks for minimally invasive surgery," in *Proceedings of the Fourth International Conference on Autonomic and Autonomous Systems*, 2008, pp. 260–265.

[12] L. Baresi, L. Pasquale, and P. Spoletini, "Fuzzy Goals for Requirements-Driven Adaptation," in *Proc. of the 18th Int. Requirements Eng. Conf.* IEEE Computer Society, 2010, pp. 125–134.

[13] C. Damas, B. Lambeau, F. Roucoux, and A. van Lamsweerde, "Analyzing Critical Process Models through Behavior Model Synthesis," in *Proc. of the 31st Int. Conf. on Software Engineering*. IEEE Computer Society, 2009, pp. 441–451.

[14] L. A. Clarke, G. S. Avrunin, and L. J. Osterweil, "Using software engineering technology to improve the quality of medical processes," in *Proc. of the 30th Int. Conf. on Software Engineering*. ACM, 2008, pp. 889–898.

[15] The Orocos Project, "Smarter control in robotics and automation," http://www.orocos.org.

[16] Object Management Group, "UML v. 2.2 Superstructure specification," Document N. formal/2009-02-02, 2009, http://www.omg.org/spec/UML/2.2/.

[17] S. Permpongkosol, M. Nielsen, and S. Solomon, "Percutaneous Renal Cryoablation," *Urology*, vol. 68, no. 1 Suppl., pp. 19–25, 2006.

[18] D. Clarke, A. Robilotto, E. Rhee, R. VanBuskirk, J. Baust, A. Gage, and J. Baust, "Cryoablation of renal cancer: Variables involved in freezing-induced cell death," *Technol Cancer Res. Treat.*, vol. 6, pp. 69–79, 2004.

[19] D. Johnson, S. Solomon, L. Su, E. Matsumoto, L. Kavoussi, S. Nakada, T. Moon, W. Shingleton, and J. Cadeddu, "Defining the Complications of Cryoablation and Radio Frequency Ablation of Small Renal Tumors," *Urology*, vol. 172, no. 3, pp. 874–877, 2004.

[20] D. Jackson *et al.*, "Alloy Analyzer," http://alloy.mit.edu/community/.

[21] D. Jackson, "Automating first-order relational logic," in *Proc. of 8th ACM SIG-SOFT Symposium on Foundations of Software Engineering*, November 2000.

[22] A. van Lamsweerde, *Requirements Engineering: From System Goals to UML Models to Software Specifications*. John Wiley, 2009.

[23] A. Pnueli, "The Temporal Logic of Programs," in *Proceedings of the 18th Annual Symposium on Foundations of Computer ScienceA*, 1977, pp. 46–57.

[24] E. Torlak and G. Dennis, "Kodkod for Alloy users," in *Proc. of 1st ACM Alloy Workshop*, April 2008.

[25] M. Klotzbücher, P. Soetens, and H. Bruyninckx, "OROCOS RTT-Lua: an execution environment for building real-time robotic domain specific languages," in *Proc. of the Int. Workshop on Dynamic Languages (DYROS'10)*, Darmstadt, Germany, November 2010.

[26] A. Okamura, C. Simone, and M. O'Leary, "Force modeling for needle insertion into soft tissue," *IEEE Trans. on Biomedical Engineering*, vol. 51, no. 10, pp. 1707–1716, October 2004.