A supervisory controller for semi-autonomous surgical interventions

Fabio Falezza , Nicola Piccinelli , Andrea Roberti , Francesco Setti , Riccardo Muradore , and Paolo Fiorini

Department of Computer Science, University of Verona, Verona, Italy
{name.surname}@univr.it

INTRODUCTION

Nowadays the main research interests in the field of Robotic Minimally Invasive Surgery (R-MIS) are related to robots' autonomy [1]. Techniques like trajectory planning, collision avoidance [2, 3], decision making and scene understanding [4] require technical advances in order to be applied to such an environment. In this paper, we propose a deterministic supervisory controller for a surgical semi-autonomous robotic platform.

The proposed method uses a three-level Hierarchical Finite State Machine (HFSM) to define all the possible behaviours of the autonomous system. The transitions of the HFSM are triggered by the *Observers*, a set of functions fed with the state of the system (robot kinematics, anatomical structures, etc.) that output a logical description of the surgery state. We tested the supervisory controller performing the "bladder neck incision" phase of a Radical Prostatectomy (RARP) procedure.

SUPERVISORY CONTROLLER

We propose a hierarchical HFSM with three layers defined as: (1) **Procedure**, which encompasses a complete surgical procedure and is composed of a set of surgical phases and transitions between them; (2) **Phase**, which defines a complex of surgical actions with a defined intention or objective having a clear beginning and end; (3) **Action**, which defines a simple tool task with a specific objective and defined as a set of surgemes and their interactions, i.e. the atomic actions within a surgery that cannot be decomposed further. The medical knowledge used to define the HFSM is provided by surgeon interviews and literature review. The control and supervision of the procedure, phases and actions is conducted by means of finite state machines (FSM). This methodology enables a strict control of the status of each phase, action and surgeme executed by each robotic tool during a procedure. The procedure is modelled as a FSM, where each state represents a phase. Following with the decomposition, each phase FSM contains an action FSM whose states are surgemes representing atomic actions performed by a robot or a surgical tool.

The supervisory controller is in charge of providing commands to the robot controller and to trigger the transitions of the HFSM. Three sub-modules compose the controller: *Supervisor* uses the knowledge of the surgical procedure to choose the atomic movement (i.e. surgeme); *Observer* converts the information generated by the perception of the environment to the trigger events encoded into the surgical procedure; and *Dispatcher* is in charge of dispatching the surgeme execution to the lower level robot controller (i.e. the trajectory reconfiguration and obstacle avoidance modules).

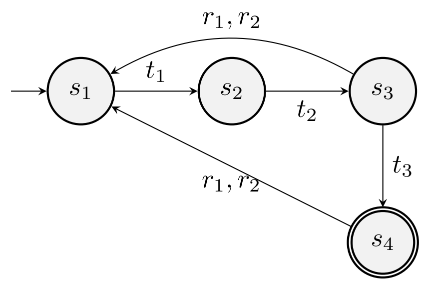


Figure 1: The finite state machine of the first action (approach catheter) of the bladder neck incision. Labels s_i, r_i and t_i are defined in Table 1.

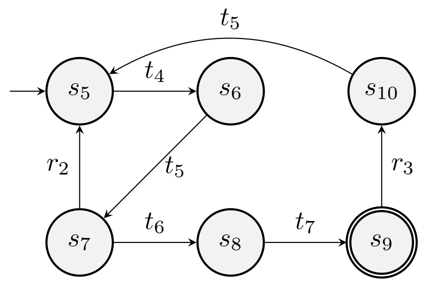


Figure 2: The finite state machine of the second action (grasp catheter) of the bladder neck incision. Labels s_i, r_i and t_i are defined in Table 1.

EXPERIMENTAL SETUP

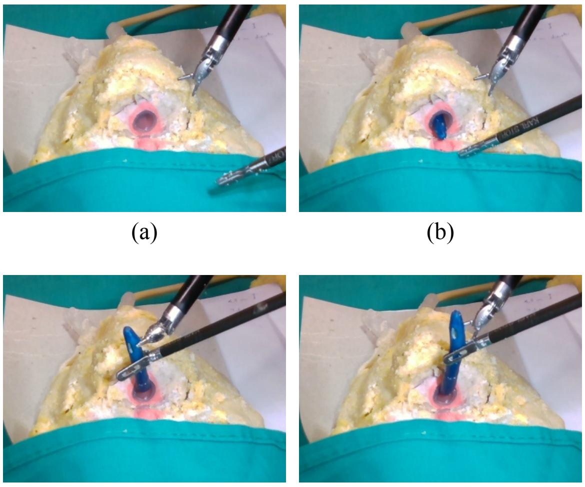
The experimental setup consists of a *da Vinci* surgical robot controlled through the *da Vinci* Research Kit (dVRK), a SARAS robotic arm [1] acting as the assistant surgeon and an Intel RealSense d435 RGB-D camera mounted on the *da Vinci* camera arm (ECM).

The HFSM used in the experiment is composed of one phase FSM and two action FSMs shown in Fig. 1 (*wait until the catheter is detected before moving SARAS arm towards it*) and Fig. 2 (*grasping and pulling movements*). A detailed description of the surgemes and the transition triggers are presented in Table 1.

Label	State description
s_1	Move to safe position
s_2	Wait until catheter is recognised
s_3	Follow the catheter position
s_4	Approach the grasping position
s_5	Idle, wait in the grasping position
s_6	Open the grasper
s_7	Reach the grasping position
s_8	Close the grasper
s_9	Pull up the grasper
s_{10}	Open the grasper

Label	Trigger description
t_1	Robot tool is in safe position
t_2	Catheter has been recognised
t_3	Catheter has been pulled-up
t_4	Robot tool is ready to start grasping
t_5	Robot tool is open
t_6	Catheter is on the grasping position
t_7	Robot tool is close
r_1	Catheter tracking is lost
r_2	Reset command by surgeon
r_3	Target position not reachable

Table 1: Description of the surgemes and triggers generated by the Observer.



(a) (b) (c) (d)

Figure 3: The automatic catheter grasping experimental validation. a) the initial position of the autonomous system, b) the arm starts moving to the catheter, c) the arm approaches the grasping point, d) once the catheter is grasped the main surgeon releases it.

The detection and tracking of the catheter is done using the OpenCV library. To detect the catheter we use a template matching technique, the resulting features are then tracked frame by frame using Lucas-Kanade optical flow. The position and the velocity of the catheter are projected in 3D (using the depth map of the RGBD camera) providing to the supervisory controller the odometry of the grasping point. The proposed tracking method is further improved by restricting the search area around the *da Vinci* arm during the catheter extraction.

RESULTS

The proposed method, as shown in Fig. 3, is able to accomplish the catheter detection and grasping autonomously. Fig. 3a shows the robot in the initial position, corresponding to the state s_1 of the FSM of Fig. 1. When the vision module recognises the catheter, the SARAS tool reaches the approach grasping position (s_4) provided by the features in the 3D system space, as shown in Fig. 3b. The surgeon extracts the catheter with the *da Vinci* arm (PSM) and pulls it up to the desired position, triggering the transition from state s_5 and s_6. Therefore, the SARAS tool starts moving towards the grasping point and can proceed with the grasping action shown in Fig. 3c. After that, the SARAS arm pulls the catheter autonomously as shown in Fig. 3d. Pictures in Fig. 3 refer to the full video available at `https://youtu.be/mTAWF56iEek`.

CONCLUSION

In this work, we proposed a supervisory controller for semi-autonomous surgical robots. The controller has been tested with realistic phantom on a specific phase of a radical prostatectomy. Further improvements of the proposed method are: the integration of machine learning approaches to provide phases and actions segmentation as input to the observer, and the extensions to the full RARP.

ACKNOWLEDGEMENTS

This work has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme under grant agreement No. 742671 (ARS) and from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 779813 (SARAS).

REFERENCES

[1] Elettra Oleari et al. "Enhancing surgical process modeling for artificial intelligence development in robotics: The SARAS case study for minimally invasive procedures". In: *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)*. IEEE. 2019, pp. 1–6.

[2] Alessio Sozzi et al. "Dynamic Motion Planning for Autonomous Assistive Surgical Robots". In: *Electronics* 8.9 (2019), p. 957.

[3] Michele Ginesi et al. "Autonomous task planning and situation awareness in robotic surgery". In: *arXiv preprint arXiv:2004.08911* (2020).

[4] Giacomo De Rossi et al. "Cognitive Robotic Architecture for Semi-Autonomous Execution of Manipulation Tasks in a Surgical Environment". In: *2019 IEEE/RSJ IROS*. IEEE. 2019, pp. 7827–7833.