2013 Third International Conference on Advanced Computing & Communication Technologies

Formal Modeling of a Tele-Surgery Domain as a Multi-Agent Planning Problem

Amod Kumar Lal
Electronics and Computer Engineering Department
Indian Institute of Technology Roorkee
Roorkee, India 247667
aklalpec@iitr.ernet.in

Rajdeep Niyogi
Electronics and Computer Engineering Department
Indian Institute of Technology Roorkee
Roorkee, India 247667
rajdpfec@iitr.ernet.in

***Abstract*— Technological advancements have led to the development of few commercially available telesurgery systems till date. However such systems are very expensive. In telesurgery, the task of a surgeon (the activities related to a surgery) is partially executed by a robot. Typically, the robot is under the control of a surgeon; it executes the instructions of the controlling surgeon. In this paper we make an attempt to formally model a telesurgery domain (heart surgery) as a multiagent planning problem. The actions related to the surgery are represented as planning operators. The model consists of two interactive agents. The state space of each agent is modeled as a transition system. We have also developed a simple prototype implementation incorporating the above features.**

Keywords- Telesurgery, agent communication, multiagent planning.

I. INTRODUCTION

A simple example of a multiagent system given in [1] would be a doubles tennis match. In this game each team has two players (agents). The two players (in a team) are necessary for the game to be played. Some important features here include: joint action, coordination, and communication among the agents. A joint action in a tennis match would be ‘waiting near the net’ for one agent while the other agent ‘returns the ball’. Here the two individual actions of the agents occurring concurrently would constitute one primitive action for the team, which is often referred to as a joint action. In order that the joint action is successful the agents need to coordinate and communicate, which results in synchronization of the agents.

In telesurgery [2], the objective is to perform a surgery remotely. That is, the surgeon and the patient are geographically apart. At the patient’s end a robot (instead of a human) performs the surgery. The robot is controlled by the surgeon, who is remotely located. The robot is expected to [faithfully] execute the instructions of the controlling surgeon. We may abstract this situation as a two-agent system. An important similarity of this system and the tennis domain is the feature of joint action. The robot waits till it has received an instruction from the surgeon. Here a joint action would include sending an instruction by the surgeon, performing the instruction by the robot upon receipt, and sending the resulting state back to the surgeon. (Modern systems may even avoid sending the resulting state, as the surgeon may observe the state remotely.) Thus ‘sending an instruction’ and ‘performing the instruction’ would constitute a joint action. A close look at this joint action and the one for the tennis match would reveal an important aspect on the roles of the agents. In the tennis domain the roles of the agents are interchangeable. This is not so in the telesurgery domain; instruction would always be send by the surgeon and the robot would always perform an instruction. Another crucial difference in these two domains is with respect to communication among agents. It is implicit in the tennis domain. It is explicit in the telesurgery domain and it would be appropriate to reflect the communication in the joint action. Given this informal notion of a joint action, a description of the initial state, a description of the actions, and a goal state, a multiagent planning problem then would be to find a sequence of such joint actions that transform the initial state to the goal state. We model this idea on the heart surgery domain that we have chosen for this study.

Telesurgery domains are quite complex. The complexity arises from several angles. Some among these include communication among the agents, controlling a robot, and medical imagery and transmission. Thus understanding such domains would call for expertise from several research areas like computer science, artificial intelligence, robotics, computer vision, and electronic communication technologies. Most works in telesurgery domains are primarily concerned on issues involving robotics [5,10] and on system development [3,4,7,9]. In this paper we make an attempt to give a formal modeling of such a domain using techniques from artificial intelligence and computer science.

The rest of the paper is organized as follows. In section II we define the planning operators corresponding to the steps of a heart surgery. In section III we suggest a formal model of the heart surgery domain as a two agent system. In section IV we discuss the implementation details. The conclusions are given in section V.

II. HEART SURGERY DOMAIN

In order to perform an open heart lung bypass surgery the following steps are performed [6][8][11].

978-0-7695-4941-5/13 $26.00 © 2013 IEEE
DOI 10.1109/ACCT.2013.22

Step1.	Harvesting the saphenous vein.
Step2.	Chest incision.
Step3a.	Open membranous sac.
Step3b.	Potassium-rich solution is injected.
Step3c.	Attach Heart-lung bypass machine.
Step4.	Graft incision.
Step5.	Sewing graft onto artery.
Step6a.	Attach pacemaker machine.
Step6b.	Injection of Protamine.
Step7.	Draining of excess fluid.
Step8.	Closing the breastbone: Stitching using stainless steel wire.
Step9.	Stitching up the chest.

We have identified the primary planning operators corresponding to the above steps.

Step1a:
Incision-Leg(s: scissor-type) scissor-type take on values scissor1, scissor2, …
PRECOND: s = scissor2
EFFECT: visible (sv)
Saphenous vein is denoted by sv
Step1b:
Harvesting (v: vein, o: object)
PRECOND: v = sv and o = tray and visible (v)
EFFECT: veinIn (v, o)
Step2:
Incision-Chest(s: scissor-type)
PRECOND: s = scissor1
EFFECT: visible (membranous sac)
Membranous sac is also called pericardium.
Step3a:
Incision-Sac(s: scissor-type)
PRECOND: s = scissor3 and visible (membranous sac)
EFFECT: visible (heart)
Step3b:
Stop-heart
PRECOND: heart(running) and visible(heart) and clamp-on-aorta and potassium-soln-injected
EFFECT: ¬ heart(running)

Placing a clamp on the aorta is an action. We assume here that the action has been performed. The result of the action is denoted by the truth of the proposition "clamp-on-aorta". Injecting a potassium rich solution is another necessary action that is to be performed. We also assume here that the action has been performed. The result of the action is denoted by the truth of the proposition "potassium-soln-injected".

Step3c:
Heart-lung-bypass
PRECOND: ¬ heart(running) and tubes-attached-right-atrium and bypassmachine-connected-aorta
EFFECT: aorta-receiving-blood

When the heart is functioning properly, blood flows from the atrium to the aorta. When the heart is stopped, the flow is maintained by a bypass machine that takes input from the right atrium and pumps blood to the aorta. The machine does both purification and pumping of blood.

Step4:
Graft-incision(affected-coronary-artery-wall: ca)
PRECOND: ¬ heart(running) and slit-at(ca)
EFFECT: graft-inserted(ca)

Step5-6:
Attach-pacemaker-wires
PRECOND: ¬ heart(running) and graft-inserted(CA) and blood-flowing(CA)
EFFECT: pacemaker-wire-attached

The steps 7 to 9 are trivial. The planning operators for these steps are not shown.

III. MODELING THE HEART SURGERY DOMAIN AS A TWO AGENT SYSTEM

In telesurgery, the task of a surgeon (the activities related to a surgery) is partially executed by a robot. Typically, the robot is under the control of a surgeon; it executes the instructions of the controlling surgeon. We model a telesurgery domain as a two agent system. We call the two agents INTERN and EXPERT. INTERN represents the robot. EXPERT represents the surgeon.

A. *Transition System*

The state space of each agent is a transition system. The states of the transition system are explained below.

1) States: Measurements like Body Temperature, Heart Rate, Blood Pressure, Pulse Oxymetry jointly constitute a state.

2) Actions and Transitions: An action is specified in terms of the preconditions that must hold before it can be executed and the effects that results when it is executed [1]. Let *a* be an action with precondition P and effect E. If the state *s* satisfies P then there is a transition from *s* to *s'* due to the action *a* and *s'* satisfies E (Fig. 1).

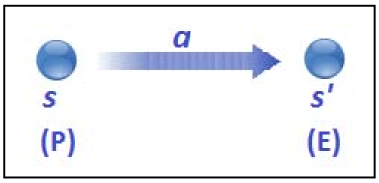

Figure 1. State Transition

B. *Agent Communication*

INTERN observes and sends the initial state s_0 to the EXPERT. In all subsequent steps, the INTERN on receiving a message (having the information of the next action a) from the EXPERT does the following:

1. Executes the action *a*
2. Observes the state *s'* resulting from executing *a*
3. Send *s'* to the EXPERT

EXPERT on receiving state *s'* from the INTERN, it first compares *s'* with the goal state *g;* if both are same then it sends a message to INTERN saying that the surgery is completed. Otherwise, it makes a decision about next action to be performed on *s'* (Fig. 2).

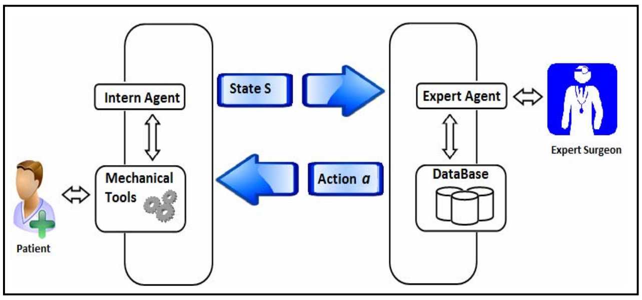

Figure 2. Communication Process between the EXPERT and INTERN

Decision Making: when EXPERT receives the message it will fetch the "State number" and "state_value" from the message and match the state_value with NORMAL and ABNORMAL set of values (Table I). If the value lies in the range of NORMAL values, it will consult database of surgical procedure in order to get next action, otherwise, it will consult database of recovery procedure in order to get alternate action to recover previous state.

TABLE I. Possible values of a State parameter

Possible Values for ***Body Temp.(^{o}F)***	
Normal Values	**Abnormal Values**
{97 to 99}	{100 to 104}

In communication the message formats for both EXPERT and INTERN will be different. The first field of the message format (sent by the INTERN) will specify the State and the second field is used to store the data belongs to the current state parameters; it may have body temperature, blood pressure, heart rate and many other surgery related data as parameters. In case of message format sent by the EXPERT the first field will consist the message identifier having one of the values among A, R, G and X. Where 'A' represents next action to be perform and 'R' represent a recovery action, G represents the goal state and X indicates the surgery abortion. The second field tells the details about the action. (Shown in Fig. 3 and 4).

State Number	Surgical data

Figure 3. Message Format for INTERN

Action Identifier	Action details

Figure 4. Message Format for EXPERT

C. *Multiagent Planning*

In the heart surgery domain a joint activity consist of sending an instruction by the surgeon, performing the instruction by the robot upon receipt, and sending the resulting state back to the surgeon. Thus 'sending an instruction' and 'performing the instruction' would constitute a joint action. We shall from now look at actions from the perspective of the INTERN.

We denote a joint action as <receive(*a)*;perform(*a*)>. The meaning of this is that first the INTERN receives an instruction (an action *a*) from the EXPERT; then it performs the action *a*. A sequence of such joint actions that transform an initial state to a goal state is defined as a multiagent plan. Formally, let s_o be an initial state and s_g be a goal state. Let σ_k= receive(a_k);perform(a_k) where $k \geq 0$. Let $\pi = \sigma_0; \sigma_1; \ldots ;\sigma_{n-1}$. π is a multiagent plan if it transforms s_0 to s_g. That is $\delta(s_0,\pi) = \delta(\delta(s_0,\sigma_0),\pi') = \delta(\delta(s_0,a_0),\pi') = s_g$, where δ is the transition function of the state transition system of the INTERN and $\pi' = \sigma_1; \ldots\ldots ;\sigma_{n-1}$. It can be easily seen that applying the transition function *n* times would result in the final state.

IV. IMPLEMENTATION DETAILS

In this section we discuss how we have simulated the heart surgery domain based on the model suggested in section III. We have used two laptops for the two agents. Each laptop has the following configuration:

- 2 GB RAM
- 320 GB Hard Disk
- 32 bit OS
- 2.20 GHz Processor (Intel i3)

We have used Java programming language. The communication of the agents is synchronous and it was implemented using Java Socket programming. We used AWT and SWING APIs to design graphical interface.

A. *Examination of States*

When the INTERN sends a state (as depicted in Fig. 3), the EXPERT upon receipt tries to find out which action is applicable at this state. For this the EXPERT examines the surgical data field in the message. For instance if temperature is a parameter whose value is chosen as say 99, by a random generation process. The EXPERT checks weather the value is within the desirable limits. If yes, it sends the suitable action. Otherwise it communicates a message saying that the state is not compatible.

B. *Recovery in case of surgical mistakes*

Performing an action in a medical surgical domain may not always result in the appropriate or desirable state. This may be due to several factors, like mistakes committed inadvertently and improper functioning of an instrument. We have implemented one such simple scenario for the heart surgery domain.

We refer to Fig 6. In the Data Table given in the left hand side, performing A1 at S0 results in S1. However due to some mistakes the resulting state is not S1; we denote this new state by ~S1. There are two recovery actions at ~S1 namely $R1_i$ and $R1_j$ as shown in Fig. 7; the resulting states are S1 and S2 respectively. In the first case we get back to the original desirable state S1. In the 2nd case we go to S2. Sometimes recovery is not possible as in ~S2.

Data Table - Blue Print

Current State	Next Action	Next State
S0	A1	S1
S1	A2	S2
S2	A4	S4
S4	A5	S5
S5	A3	S3
S3	--	--

S0: Initial State, S3: Goal State

Recovery - Matrix

	S0	S1	S2	S3	S4	S5
~S0	0	0	0	0	0	0
~S1	0	1	1	0	0	0
~S2	0	0	0	0	0	0
~S3	0	0	0	0	0	0
~S4	0	0	0	1	1	1
~S5	0	0	0	0	0	0

Figure 6. Data Structure for Surgical Data storage

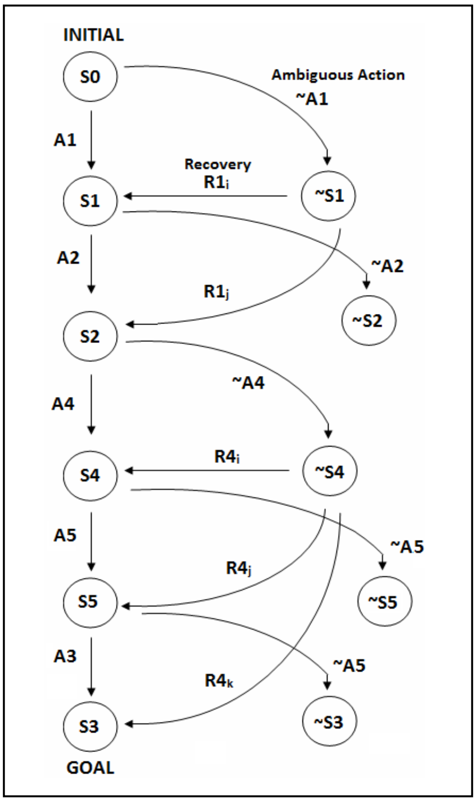

Figure 7. Transition System in Surgery

V. CONCLUSIONS

In this paper we tried to develop a formal modeling of a telesurgery domain. For this study we have chosen the heart surgery domain. We defined the steps of the surgery as planning operators. We have shown that performing a telesurgery amounts to finding a multiagent (here two-agent) plan that involves joint actions. We feel that the modeling task has enhanced our knowledge of the domain. Further improvements on the model would lead to a better understanding and analysis of the domain. We have also implemented the model. The preliminary computer simulations reflect the complexity of the task at hand. As part of our ongoing work we would like to include more real-life features and implement it on different machines (platforms). We expect to develop simulations for other surgery domains as well. We would also like to enrich our model that accounts for trials that may be needed at some point of a surgery.

ACKNOWLEDGMENT

The second author acknowledges the support of the DST grant SR/S3/EECE/0075/2011 (G).

REFERENCES

[1] S. Russell and P. Norvig, “Artificial Intelligence: A modern Approach”, Prentice Hall, 1995

[2] “Remote Surgery”, http://en.wikipedia.org/wiki/Remote_surgery, 2012.

[3] K. Brady and T. J. Tarn, “Internet-Based Remote Teleoperation”, Proceedings of the 1998 IEEE Int. Conf. on Robotics and Automation, 1998.

[4] A. Ferworn, R. Roque, I. Vecchia and MAX, “Wireless teleoperation via the World Wide Web”, Proc. IASTED Conf. on Robotics & Applications, 1999.

[5] H. Hu, L. Yu, P. W. Tsui, Q. Zhou , “Internet-based Robotic Systems for Teleoperation”, International Journal of Assembly Automation, Vol. 21, No. 2, 2001.

[6] “The Circulatory System” http://lsa.colorado.edu/essence/texts/heart.html

[7] A. Moreno, “Medical Applications of Multi-Agent Systems”, Computer Science & Mathematics Department, Universitat Rovira i Virgili, Spain, 2003.

[8] “Coronary Artery Disease”, Surgical Associates of Texas, P.A., Texas heart Institute, April 2005.

[9] M. de Weerdt, A. ter Mors, and C. Witteveen. “Multi-agent planning - an introduction to planning and coordination”. Technical report, Delft University of Technology, 2005.

[10] J. J. Regan, MD and E. C. Benzel, MD, “Robotics and Computers in Minimally Invasive Spine Surgery, January 2nd 2010.

[11] “Virtual Open Heart Surgery”, http://www.abc.net.au/science/lcs/swf/heart.swf