Autonomous Multilateral Debridement with the Raven Surgical Robot

Ben Kehoe[1], Gregory Kahn[2], Jeffrey Mahler[2], Jonathan Kim[2], Alex Lee[2], Anna Lee[2], Keisuke Nakagawa[4],
Sachin Patil[2], W. Douglas Boyd[4], Pieter Abbeel[2], Ken Goldberg[3]

***Abstract*— Autonomous robot execution of surgical sub-tasks has the potential to reduce surgeon fatigue and facilitate supervised tele-surgery. This paper considers the sub-task of *surgical debridement*: removing dead or damaged tissue fragments to allow the remaining healthy tissue to heal. We present an autonomous multilateral surgical debridement system using the Raven, an open-architecture surgical robot with two cable-driven 7 DOF arms. Our system combines stereo vision for 3D perception with trajopt, an optimization-based motion planner, and model predictive control (MPC). Laboratory experiments involving sensing, grasping, and removal of 120 fragments suggest that an autonomous surgical robot can achieve robustness comparable to human performance. Our robot system demonstrated the advantage of multilateral systems, as the autonomous execution was 1.5× faster with two arms than with one; however, it was two to three times slower than a human. Execution speed could be improved with better state estimation that would allow more travel between MPC steps and fewer MPC replanning cycles. The three primary contributions of this paper are: (1) introducing debridement as a sub-task of interest for surgical robotics, (2) demonstrating the first reliable autonomous robot performance of a surgical sub-task using the Raven, and (3) reporting experiments that highlight the importance of accurate state estimation for future research. Further information including code, photos, and video is available at: http://rll.berkeley.edu/raven.**

I. Introduction

Robotic surgical assistants (RSAs), such as Intuitive Surgical's da Vinci® system, have proven highly effective in facilitating precise minimally invasive surgery [9]. Currently these devices are primarily controlled by surgeons in a local tele-operation mode (master-slave with negligible time delays). Introducing autonomy of surgical sub-tasks has potential to assist surgeons, reduce fatigue, and facilitate supervised autonomy for remote tele-surgery.

Multilateral manipulation (with two or more arms) has potential to reduce the time required for surgical procedures, reducing the time patients are under anaesthesia and associated costs and contention for O.R. resources. Multilateral manipulation is also necessary for sub-tasks such as suturing; hand-off of tissue or tools between arms is common as each arm has limited dexterity and a workspace that may not cover the entire body cavity. Autonomous multilateral manipulation is of particular interest as surgical robot systems can be configured with 3, 4, or more arms (one might imagine surgical octobots). Even when surgical robot arms operate in parallel, it is important to avoid collisions as their workspaces are rarely disjoint.

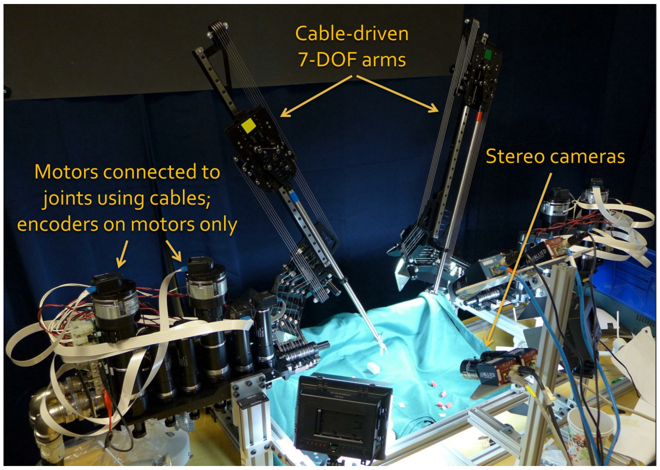


Fig. 1. The autonomous multilateral debridement system using the Raven surgical robot.

In this paper, we introduce *surgical debridement* (pronounced de-BREED-ment) as a relevant sub-task for autonomous surgical robotics. Surgical debridement is a tedious surgical sub-task in which dead or damaged tissue is removed from the body to allow the remaining healthy tissue to heal [2], [10]. Autonomous surgical debridement requires perception to locate fragments, grasp and motion planning to determine collision free trajectories for one or more arms and grippers to grasp them, and control to deposit them into a receptacle (see Fig. 2(b)).

To the best of our knowledge, this project is the first to demonstrate a reliable autonomous surgical sub-task with the Raven robot. We use the Raven surgical robot [12] and a custom stereo vision system to study autonomous multilateral surgical debridement. Our experimental system is shown in Fig. 5.

Sterilization demands that robot actuators and encoders remain outside the body so actuation inside the body is achieved using long cables and flexible elements that compound uncertainty and control of end-effector position and orientation. Most surgical robots, such as the da Vinci and the Raven, have 6 DOF per arm (plus a grasp DOF), so there is no joint redundancy. Also, each arm must enter the body through a fixed portal that constrains the motion at that point akin to a spherical joint. Thus each arm has very limited

[1]Department of Mechanical Engineering; benk@berkeley.edu

[2]Department of Electrical Engineering and Computer Sciences; {gkahn, jmahler, jonkim93, alexlee_gk, leeanna, pabbeel}@berkeley.edu

[3]Department of Industrial Engineering and Operations Research and Department of Electrical Engineering and Computer Sciences; goldberg@berkeley.edu

[1–3] University of California, Berkeley; Berkeley, CA 94720, USA

[4]Division of Cardiothoracic Surgery; University of California Davis Medical Center; Sacramento, CA 95817, USA; {keisuke.nagakawa, walter.boyd}@ucdmc.ucdavis.edu

dexterity and a workspace that intersects the boundaries of the body cavity.

Human surgeons provide a compelling existence proof that complex and precise manipulation is achievable using such robot hardware; the challenge for robotics is to reproduce the extraordinary perception and control skills of humans.

Kinect-like RGBD cameras can significantly improve robot perception, but they are extremely difficult to use in surgical environments due to the highly reflective surfaces of organic tissues, fluids, and sterilizable actuator surfaces. Therefore, we rely on stereo vision, which is commonly used in minimally-invasive surgery. The challenges of computer vision are well known; they include noise, calibration, correspondence, segmentation, and occlusions.

Because of the uncertainty in robot state estimation and control, replanning is required to prevent robot collisions with obstacles (the other arm, the worksurface, and other objects). Collisions are a familiar problem in robotics, but are exacerbated in surgical robots because a collision with the worksurface can cause human injury or snap delicate cables requiring extensive repair time.

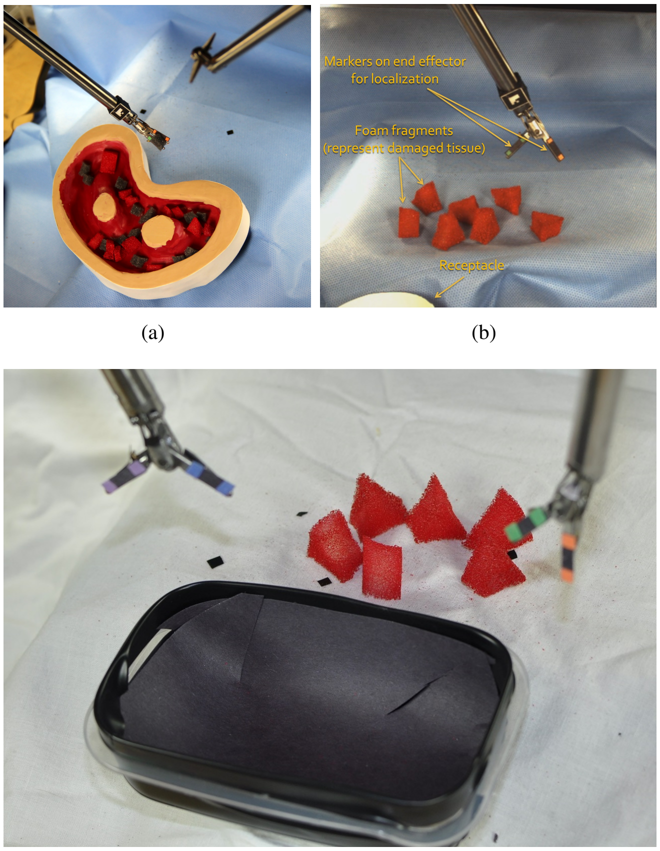
For 30 fragments, we recorded the timing and reliability of the debridement sub-task when performed by a medical student who has experience on a laparoscopic surgical simulator. The medical student viewed a 3D display from the stereo camera pair and used a game controller device to perform local teleoperation. We then recorded the timing and reliability of the same sub-task for 120 fragments using the autonomous system: 60 using one arm only, and 60 using both arms.

II. Related Work

Existing robotic surgical systems can be categorized into a spectrum based on the modality of interaction with the surgeon [27], [34]. These systems range from pure teleoperated or master/slave systems that directly replicate the motions performed by the surgeon [11], [27], to supervisory or shared-control systems where the surgeon holds and remains in control of the medical instrument and the robot provides assistance [33], to purely autonomous systems where medical motions are planned off-line when detailed quantitative pre-operative plans of the surgical procedure can be laid out and executed autonomously without intraoperative modification [35]. In addition, intelligent robotic assistants have also been proposed for rendering assistance in minimally invasive surgery [17], [19].

In this work, we focus on autonomous execution of a tedious surgical sub-task known as surgical debridement [2], [10], which involves removing damaged tissue from an affected area to allow the surrounding tissue to heal. We note that prior work has addressed the problem of designing planning and control algorithms for autonomous execution of other surgical sub-tasks such as knot tying or suturing [25], [36] and tissue retraction during surgery [14], [20].

Recent advances in motion planning, control, and perception have enabled robotic systems to perform complex manipulation tasks in real world domains [3], [8], [7], [28], [30]. These systems perform integrated task and motion planning (see e.g., [1], [6], [16], [37]) by using state machines or task graphs [4], [31] for high-level task specification and motion planning algorithms for realization of low-level sub-tasks. Extensions have been proposed to consider uncertainty in task execution [15], [32]. Our work uses a similar architecture for autonomy that integrates a high-level task specification in terms of a state machine [4] with low-level planning. However, instead of open-loop execution of motion plans for accomplishing low-level sub-tasks, we re-plan after every time-step in the spirit of model predictive control [24] to mitigate uncertainty.



(a) (b)

(c)

Fig. 2. Varying levels of realism in the surgical debridement task. (a) Two arm surgical debridement with simulated anatomical structures containing multiple foam colors. (b) Single arm surgical debridement. (c) Two arm surgical debridement with plain white background. This is the setup used in the experiment.

There is extensive prior work on calibration of kinematic parameters of robotic manipulators [13]. Extensions have been proposed to simultaneously calibrate robot and sensor (e.g., camera) parameters [22], [38]. These methods do not account for errors resulting from material non-linearities such as cable stretch, prevalent in cost-effective cable-driven actuation mechanisms.

III. Surgical Debridement

Minimally-invasive surgery requires the execution of many sub-tasks, including incisions, suturing, clamping, retraction, etc. Not all of these are suited to autonomous operation; for example, cutting tissue requires very high precision and has

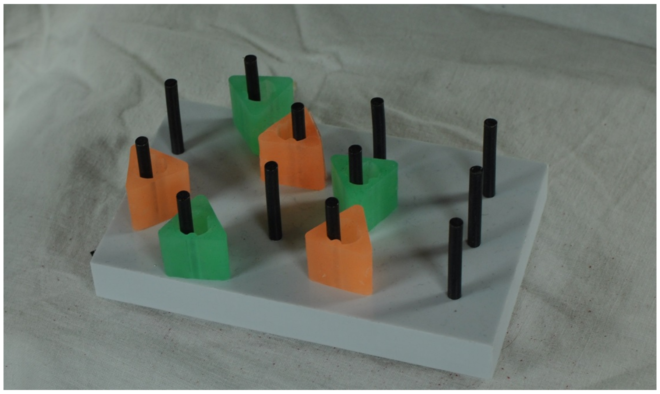


Fig. 3. Fundamentals of Laparoscopic Surgery pick-and-place task, used in a skills training program. The blocks must be grasped and transferred between pegs.

very low error tolerance, so was not a sub-task we considered for automation.

The Fundamentals of Laparoscopic Surgery [26], a training kit and set of sub-tasks for students of laparoscopic surgery includes a pick-and-place sub-task requiring the transfer of triangular plastic blocks between between vertical pegs (see Fig. 3), which also requires hand-off between arms. The clearance between the block and peg is extremely limited and prone to collisions that can cause snapping of cables during development, especially with the current kinematic uncertainty in the Raven. This motivated us to consider related sub-tasks but we hope to revisit FLS in the future.

A. Task Definition

We propose *surgical debridement* as a sub-task of interest for experimental autonomous surgical robots. Surgical debridement is a tedious surgical sub-task in which dead or damaged tissue is removed from the body to allow the remaining healthy tissue to heal faster [2], [10]. It is tedious, so automating it has potential to reduce surgeon fatigue and there are contexts where increasing speed of debridement could speed healing. Surgical debridement involves detection, grasping, and motion planning components. Importantly, debridement can be considered at different levels of difficulty (see Fig. 2), allowing us to start with a less complex environment as a first step toward more realistic environments.

Thus far, we have considered an idealized environment in which fragments designated as damaged tissue are placed randomly on a planar worksurface. The robot must find the damaged tissue fragments, grasp them, and move them to a receptacle. Future versions of the sub-task can include different types of fragments of varying sizes, more complex cavities with obstacles, and attaching the fragments to the work surface and requiring a cutting action for removal.

B. Failure Modes

We identify the following nine failure modes for the robot system:

Identification:

1) Fragment false negative: no detection of a fragment in the workspace.
2) Fragment false positive: detection of a fragment where none exists.
3) Pickup false negative: after successful grasping, no detection of a fragment in the gripper, causing an unnecessary regrasp.
4) Pickup false positive: after a pickup failure (see below), detection of a fragment in the gripper.

Grasping:

5) Grasp failure: the gripper is closed, but no part of the fragment is within the gripper.
6) Multiple grasp: the gripper unintentionally grasps multiple fragments. When targeting a single fragment for pickup, any other fragments grasped could possibly be healthy tissue, even if they happen not to be.
7) Pickup failure: the gripper has closed on some part of the fragment, but the fragment falls out of the gripper on lifting.

Movement:

8) Drop en route: after lifting, the fragment falls out during the move to the receptacle.
9) Dropoff failure: the fragment is dropped from the gripper upon arrival to the receptacle, but the fragment lands outside the receptacle.

IV. Hardware

A. Raven Surgical Robot

We use a Raven surgical robot system (Fig. 5). The Raven is an open-architecture surgical robot for laparoscopic surgery research with two cable-driven 7 DOF arms, intended to facilitate collaborative research on advances in surgical robotics [12].

The primary difficulty in using the Raven for autonomous operation is state estimation. For surgical robots where space is limited and sterilization is essential, cable-driven actuators are often used and it is not feasible to install joint sensors at the distal ends of the devices. Such indirect control and sensing is inherently imprecise. As a result, even a small amount of slack or stretch in the cables can greatly increase the uncertainty in gripper pose. State estimation has previously been explored in simulation [23], but not in physical experiments.

B. Vision Hardware

Since the kinematics introduce considerable uncertainty in the calculation of the gripper pose, we use a vision system to obtain direct measurements of the pose. The Raven presents challenges on this front as well. The size of the grippers is too small to use complex fiducial markers like those based on 2D bar codes. We were able to place a fiducial marker on the wrist link of the robot, but the small size meant the cameras had trouble detecting the marker, and the measurement was highly noisy even when it was detected.

We use a stereo vision system to estimate the pose using colored dots mounted on the gripper (Fig. 2(b)). The stereo

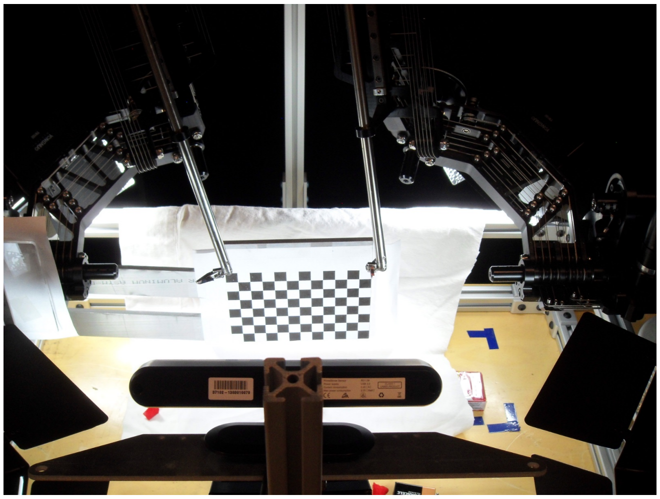


Fig. 4. Removable bracket for rigidly mounting a checkerboard in the workspace for registering the stereo cameras. We use routines from OpenCV for this purpose [5].

vision system is also used to construct a static 3D point cloud from the disparity image, which is used to localize the fragments. Off-the-shelf stereo cameras are usually built for larger workspaces, and thus the camera pair would be too widely separated for our environment. We constructed a custom stereo camera using a pair of Prosilica GigE GC1290C cameras with 6 mm focal length lenses at a separation of 4.68 cm for this purpose.

We also experimented with a Primesense Carmine sensor for obtaining point clouds of the environment. However, the Carmine relies on a projected texture, which does not work on specular reflective surfaces like the stainless steel the Raven tool is constructed from. Therefore, the Carmine cannot be used for detecting the gripper.

The cameras must be registered to the robot frame to allow their detections to be used to direct the robot. However, the small size of the workspace prevents the camera field of view from including the robot base. To register the cameras, we fabricated a removable bracket for a checkerboard that could be mounted to the robot base (see Fig. 4), putting the checkerboard in the camera field of view with a known pose relative to the base. This also allows calculation of the transform between bases of the individual arms, which are not precision mounted relative to each other, by using the camera as an intermediate frame.

V. Problem Definition and Method

The surgical debridement task environment considered for this work focuses primarily on the motion planning component of the task. The vision component is simplified through the use of a uniform white background. For the grasping components, the tissue fragments were modeled with small, irregular pieces of foam rubber.

This section covers the vision system for fragment detection and gripper pose estimation in Section V-A, the optimization-based MPC approach in Section V-B, and the multilateral coordination required by the task in Section V-C.

A. Vision System

We use the vision setup outlined in Section IV-B to detect and segment the fragments and for detecting the gripper pose.

1) Fragment Segmentation: In order to reliably retrieve the fragments, we must localize the fragments with respect to the robot using the vision system. To simplify the localization, we restricted all fragments to be a specific red hue with a known upper and lower bound of HSV (Hue, Saturation, Value) given the lighting conditions of the workspace. Furthermore, this HSV range was not present elsewhere in the workspace. Given this constraint, localizing the fragments was a three-step process. First, we threshold the image based on HSV values to identify the groups of pixels representing the fragments. Then, we find a reference point for the fragments by tracing the contours and computing the region centroid. Finally, we use the disparity of the fragment centroid between the left and right images to calculate the position of the fragment centroid in 3D space. To help deal with partially occluded foam pieces, we use an alternative fragment reference point with a constant offset from the lower bound of the fragment in the image.

2) Inverse Control: Reliable, autonomous execution requires precise positioning of the gripper pose during execution. The Raven control software takes as input a desired pose, but since the forward kinematics used by this software produces an unreliable estimate of the true gripper pose, we cannot directly input the desired pose.

The purpose of the inverse control process in this section is not primarily to estimate the pose (in which case standard estimation methods like the Kalman Filter would be appropriate), but to calculate, given a desired true pose, the input pose to send to the control software to reach the desired true pose.

We use the vision system to detect the gripper pose, and we estimate the gripper pose using color-based fiducial marks. For each gripper finger, we designate a specific color with a known range of HSV values, given the constrained lighting conditions of the workspace. Each gripper finger has exactly two such marks of the same color, one on the end closest to the joint, and one on the end closest to the tip (see Fig. 2(b)). Using a process similar to the fragment segmentation, we threshold incoming images from the stereo pair for each of the four known HSV values and use the centroids of the regions along with the disparity to find the points in 3D space.

To determine the position of the gripper, we take the average of the position of upper left and upper right fiducial marks on each of the grippers. We calculate the orientation of the gripper by finding the vectors along each gripper finger using the fiducial marks. These vectors are coplanar, and the orientation is determined from the component-wise average of the vectors (the axis along the center of the gripper) and the normal (parallel to the gripper joint axis).

The detected pose is assumed to be the true pose. However, we receive these updates from the camera at 10 Hz under ideal conditions but often slower, and updates may not happen for some time, for example while carrying a fragment, the markers on the gripper may be occluded. To allow for estimated poses in between these updates, we use a pose estimation algorithm using updates from the forward kinematics pose. We use the following notation:

$T_{c,t}$	Detected pose at time t
$T_{\mathrm{K},t}$	Calculated pose at time t
$\widehat{T}_{c,t}$	Estimated pose at time t
$\Delta T_{c,t_k \to t_{k+1}}$	Incremental change in detected pose from time t_k to time t_{k+1}
$\Delta T_{\mathrm{K},t_k \to t_{k+1}}$	Incremental change in calculated pose from time t_k to time t_{k+1}

We need a system for approximating, at time t_n, $\widehat{T}_{c,t_n}$ based on the most recent calculated pose T_{c,t_k} for some $k < n$, the calculated poses $T_{\mathrm{K},t_k}, \ldots, T_{\mathrm{K},t_n}$ and Our approach uses two adjustment matrices, using the following notation:

$T_{A,L,k}$	Left-multiplied adjustment matrix after the kth detected pose
$T_{A,R,k}$	Right-multiplied adjustment matrix after the kth detected pose

We assume that the change in actual pose can be estimated from the change in actual pose. We assume that locally, the change in actual pose is related to the change in calculated pose by a rigid transformation ($T_{A,L,k}$) and an offset ($T_{A,R,k}$). However, these matrices may vary both over the workspace of the robot and between different runs of the robot. Therefore, we take an iterative approach to calculate these matrices based on the difference between the detected and calculated poses.

Given the detected and calculated poses at two times t_0 and t_1, we first calculate the delta-pose for each, $\Delta T_{c,t_0 \to t_1} = T_{c,t_0}^{-1} T_{c,t_1}$ and $\Delta T_{\mathrm{K},t_0 \to t_1} = T_{\mathrm{K},t_0}^{-1} T_{\mathrm{K},t_1}$ As described below, we use two adjustment transforms $T_{A,L,0}$ and $T_{A,R,0}$ such that

$$\Delta T_{c,\,t_0 \to t_1} = T_{A,L,0} \Delta T_{\mathrm{K},t_0 \to t_1} T_{A,R,0} \tag{1}$$

Then, given a new calculated pose at time t_2, without having received a new detected pose, we estimate the true pose by finding $\Delta T_{\mathrm{K},t_1 \to t_2} = T_{\mathrm{K},t_1}^{-1} T_{\mathrm{K},t_2}$, and applying the adjustment transforms:

$$\begin{aligned} \Delta T_{c,t_1 \to t_2} &= T_{A,L,0} \Delta T_{\mathrm{K},t_1 \to t_2} T_{A,R,0} && (2) \\ \widehat{T}_{c,t_2} &= T_{c,t_1} \Delta T_{c,t_1 \to t_2} && (3) \\ &= T_{c,t_1} T_{A,L,0} \Delta T_{\mathrm{K},t_1 \to t_2} T_{A,R,0} && (4) \end{aligned}$$

The adjustment matrices in Eq. 1 are iteratively updated with each received detected pose, alternating between updating $T_{A,L,k}$ and $T_{A,R,k}$. They are both initialized to identity, $T_{A,L,0} = T_{A,R,0} = I_4$. Given the above update at t_1, we keep $T_{A,R,1} = T_{A,R,0}$ and update the left adjustment matrix as follows:

$$T_{A,L,1} = \mathit{interp}(T_{A,L,0}, \Delta T_{c,t_0 \to t_1} (\Delta T_{\mathrm{K},t_0 \to t_1} T_{A,R,0})^{-1})$$

Then, given a second update of the detected pose at t_3, we keep $T_{A,L,2} = T_{A,L,1}$ and update the right adjustment matrix:

$$T_{A,R,2} = \mathit{interp}(T_{A,R,1}, (T_{A,L,1} \Delta T_{c,t_1 \to t_3})^{-1} \Delta T_{\mathrm{K},t_1 \to t_3})$$

where the interp function is linear interpolation of the position and spherical linear interpolation for the orientation.

B. Optimization-based Motion Planning with trajopt

Due of the large kinematic uncertainty, an arm may not closely follow the path it is given, which increases the chance of collisions. There are two options for dealing with this situation: (i) allow for this error with a safety margin around the path that the other arm must keep out of, or (ii) use a Model Predictive Control approach and replan frequently using updated pose estimates. Because of the small size of the workspace for the Raven, the first option is not feasible; the size of the safety margin would preclude the other arm from operating anywhere near it. Additionally, this means that both arms must plan together; a path planned for a single arm would have to include this safety margin if the other arm was independently planning its own path. For more details, see Section V-C.

Frequent replanning is also required to maneuver the arm onto the fragment for grasping. In the current system, each arm is permitted to move a maximum of 2.5 cm before replanning. With this maximum distance, the safety margin can be set very small. During experiments, the safety margin was set to 1 mm with no collisions occurring.

We use trajopt [29], a low-level motion planning algorithm based on sequential convex optimization to plan locally-optimal, collision-free trajectories simultaneously for both arms. An important feature of trajopt is the ability to check continuous collisions: the arm shafts are very narrow, which could allow them to pass through each other between points on the path.

Additionally, trajopt provides flexible facilities for integrating many different constraints, including collision constraints, pose constraints, and feasibility (e.g., joint limit) constraints. We use all three kinds of constraints. The pose constraint is used to ensure the orientation of the gripper keeps the colored markers towards the cameras so that pose estimation will continue receiving updates.

C. Multilateral Coordination

While the debridement task can be performed by a single arm, using multiple arms can reduce the overall execution time. This type of multilateral is more flexible than inherently multi-arm tasks like hand-offs.

Performing multilateral debridement requires coordination between the arms in two important ways: planning and resource contention.

1) Two-arm Planning: Planning for two arms can be performed in a centralized manner, where both 6 DOF arms are planned using a single 12 DOF planner, or in a decentralized manner by two 6 DOF planners. While the centralized planner can produce higher-quality plans, it requires more coordination between the arms than decentralized planners.

	Human	Autonomous	
	Local teleoperation	Single arm	Two arm
Total number of fragments	30	60	60
Average time per fragment (s)	29.0	91.8	60.3
Average time for perception (%)	–	12.1	10.0
Average time for planning (%)	–	32.8	36.6
Average time for arm movement (%)	–	55.1	45.7
Average time waiting on other arm (%)	–	–	7.7
Average number of replanning steps	–	11.06	10.58
Fragment false negative (%)	0.0	1.9	0.0
Fragment false positive (%)	0.0	0.0	0.0
Pickup false negative (%)	0.0	0.0	0.0
Pickup false positive (%)	0.0	0.0	3.6
Grasp failure (%)	5.0	3.5	3.6
Grasp multiple fragments (%)	0.0	5.2	7.1
Pickup failure (%)	0.0	0.0	0.0
Drop en route (%)	0.0	0.0	1.8
Dropoff failure (%)	0.0	0.0	0.0

TABLE I

Average execution time and occurrences for failure modes defined in Section III-B. The number of replanning steps is the number of times during execution that the system accepts a new input and output state and generates a new plan. The two-arm autonomous system performed approximately half as fast as teleoperation, but 1.5× faster than the one-arm autonomous system. The failure rates were similar between autonomous and teleoperation, with all but one failure mode occurring less than 5% of the time. The multiple-fragment grasp failure mode occurred due to segmentation lumping close fragments together.

When using decentralized planners, each arm plans using the other arm's existing plan as an obstacle to be avoided. With high kinematic uncertainty, this can cause problems because the actual location of the other arm may not follow that plan. The Model Predictive Control approach outlined in Section V-B would not necessarily update the plans of the arms at the same time. Therefore, we chose to use centralized planning.

Each arm must replan when changing its target pose (for example, after picking up a fragment, the target pose changes from the fragment location to the receptacle) and as part of the MPC approach described above. In one-arm operation, the arm control code submits its target pose to the planner at these replanning points, and planning can happen immediately. In two-arm operation, the centralized planner must plan for both arms simultaneously. The arms use the same planning interface: when an arm reaches a replanning point, it submits its target pose to the planner; in contrast to the one-arm case, the planner will wait until it has received requests from both arms to replan, and then it returns the plans to the respective arms. If one arm completes its plan before the other arm completes its plan, it must wait to receive a new plan. The size of this effect is shown in Table I.

2) Resource Contention: There are two aspects of the debridement task where two arms contend for the same resource: fragment allocation and dropoff.

Fragment allocation is performed with a single shared fragment allocator that receives the fragment detections and processes requests from both arms. The fragments are allocated in a greedy manner, with a request from the left arm receiving the left-most unallocated fragment, and similarly for the right arm. Only one fragment is allocated at a given time for each arm.

The single fragment receptacle is not large enough for both arms to drop fragments into at the same time given the kinematic uncertainty of the system. Therefore, the arms synchronize their access to the receptacle using a shared token.

VI. Results and Experiments

The experiment was performed with six foam rubber fragments in a random configuration, as shown in Fig. 2(c). The receptacle, located at the front of the workspace, measured approximately 11×7 cm.

For teleoperation, the human operator viewed the workspace through the stereo pair using a 3D monitor, and controlled the Raven using the Razer Hydra controller.

We use the failure modes defined in Section III-B. We experienced occasional static cling in which the fragment would not fall out of an opened gripper; in these cases, failure was indicated if the fragment would have fallen outside the receptacle.

A. Autonomous Performance and Comparison with Human Local Teleoperation

As a baseline comparison, we had the task performed in teleoperation by a third-year medical student with experience on a laparoscopic telesurgery simulator. The purpose is to provide the reader with a rough idea of the execution time for a human, rather than to perform a rigorous human-robot comparison experiment, especially since autonomous results

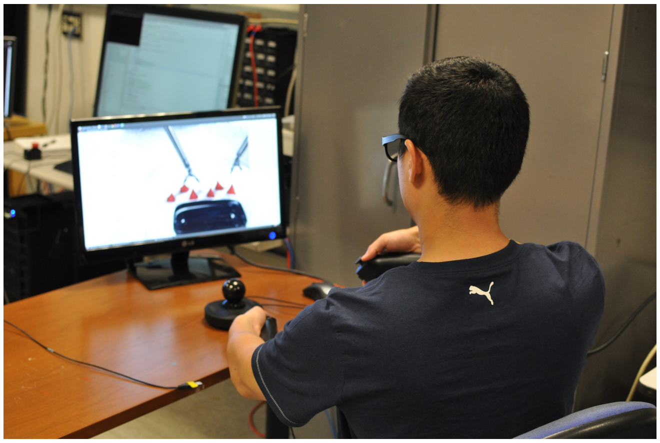


Fig. 5. Setup for human local teleoperation, using a 3D monitor and the Razer Hydra controller.

shown here are far slower than the human. To simulate surgical conditions, the teleoperation was performed by viewing the workspace on an LG D2342 3D monitor (which uses polarized glasses-based technology) using the same cameras used by the autonomous system.

Table I shows the comparison between single-arm autonomous, two-arm autonomous, and teleoperated execution. The autonomous system was executed ten times for each test, and the human operator executed the task five times.

The autonomous system in two-arm operation took on average 2.1× longer than in teleoperation. However, the amount of time spent in motion for the two arm system was actually slightly less than for the overall teleoperation execution time. This was despite the fact that, in the autonomous operation, the robot moved slowly due to the need to obtain recent updates from the vision system. Although the teleoperator was permitted to use both arms simultaneously, we did not observe him using them in this manner. Each fragment was picked up sequentially. The autonomous system, however, was able to parallelize its arm movements. If the kinematics errors were reduced and the pose estimation improved, the camera updates could be less frequent and the speed of the robot higher.

The planning and perception together took nearly 50% of the time. The perception code was coded in Python and was not optimized to take advantage of available GPU hardware, which indicates that significant speedups can be made.

The planning time was due in large part to the number of times the system must generate a new plan. Currently, the system must plan an average of 10.81 times during the move to, grasping, and dropoff of a single fragment. This is due to the 2.5 cm maximum distance that an arm is permitted to move before replanning. We found that increasing this distance caused the actual path to deviate too far from the planned path. Improved state estimation would reduce this deviation, allowing for longer distances between replanning.

As noted in Section V-B, the short replanning distance allowed for a very small safety margin to be used, 1 mm. This allowed the two arms to pick up closely-packed fragments more quickly, as the arms could pick up adjacent fragments without penetrating the safety margin.

The two-arm autonomous system was on average 1.5× faster than the one-arm system. This is less than a 2× speedup due in part to waiting time and to increased planning and perception times under the added complexity of two arms.

Both autonomous and teleoperated systems were able to successfully complete all trials, recovering from grasp and motion failure modes. No false negatives were observed, though the vision system would occasionally lump two close fragments together as a single detection; once one of the fragments was picked up, the other would be correctly detected. The grasp failure rate was slightly higher for human teleoperation than for the autonomous system; we believe this is due to the 3D camera not being spaced optimally for human viewing, which led to the human operator reporting a lack of sufficient depth perception.

Further information on this research, including code, photos, and video, is available at: http://rll.berkeley.edu/raven.

VII. Conclusion and Future Work

We have introduced surgical debridement as a sub-task of interest for autonomous robotics and developed a working system. Laboratory experiments suggest that an autonomous multilateral surgical robot can perform debridement with robustness comparable to human performance, with two arm operation 1.5× faster than with one arm. However, our robot system operates two to three times slower than a human. Most of the delay is produced when the robot stops to sense and replan. We assume conservatively that due to accumulated state uncertainty the robot must do this after every movement of 2.5 cm, averaging over 10 replan cycles per fragment. Execution speed could be improved by 1) reducing the time per replan cycle with faster processors (Moore's Law) and multicore computing, and 2) reducing the number of replan cycles with better state estimation that would remain valid for longer movements, allowing more travel between MPC cycles.

In future work we will focus on state estimation using empirical models of systematic and residual error and probabilistic models based on the Belief Space framework [18], [21]. We will also study robot performance in more complex debridement scenarios, including a mix of fragment types (eg, healthy vs. diseased) and more complex body cavity models with obstacles. We are also interested in exploring hybrid systems with both autonomous and human supervisory modes, as in a remote tele-surgery scenario, where a human supervisor is in the loop to periodically confirm a set of fragment detections and motion plans prior to execution. Finally, we will improve and explore the multilateral aspect of this task, including closer cooperation between arms (such as transferring fragments between arms as in the FLS training tasks), adding additional robots for more arms, and human-robot collaboration in which one arm is autonomous and one arm is controlled by a human.

VIII. Acknowledgments

We thank our many collaborators on this project, in particular PI Allison Okamura and co-PIs Greg Hager, Blake Hannaford, and Jacob Rosen, as well as Ji Ma and Hawkeye King. This work is supported in part by a seed grant from the UC Berkeley Center for Information Technology in the Interest of Science (CITRIS), by the U.S. National Science Foundation under Award IIS-1227536: Multilateral Manipulation by Human-Robot Collaborative Systems, by AFOSR-YIP Award #FA9550-12-1-0345, and by Darpa Young Faculty Award #D13AP00046.

References

[1] R. Alami, R. Chatila, S. Fleury, M. Ghallab, and F. Ingrand, "An architecture for autonomy," *Int. Journal of Robotics Research*, vol. 17, no. 4, pp. 315–337, 1998.

[2] C. E. Attinger, E. Bulan, and P. A. Blume, "Surgical debridement: The key to successful wound healing and reconstruction," *Clinics in podiatric medicine and surgery*, vol. 17, no. 4, p. 599, 2000.

[3] J. A. Bagnell, F. Cavalcanti, L. Cui, T. Galluzzo, M. Hebert, M. Kazemi, M. Klingensmith, J. Libby, T. Y. Liu, N. Pollard, *et al.*, "An integrated system for autonomous robotics manipulation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2012, pp. 2955–2962.

[4] J. Bohren and S. Cousins, "The SMACH high-level executive [ROS news]," *IEEE Robotics & Automation Magazine*, vol. 17, no. 4, pp. 18–20, 2010.

[5] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly, 2008.

[6] S. Cambon, R. Alami, and F. Gravot, "A hybrid approach to intricate motion, manipulation and task planning," *Int. Journal of Robotics Research*, vol. 28, no. 1, pp. 104–126, 2009.

[7] S. Chitta, E. G. Jones, M. Ciocarlie, and K. Hsiao, "Perception, planning, and execution for mobile manipulation in unstructured environments," *IEEE Robotics and Automation Magazine*, vol. 19, no. 2, pp. 58–71, 2012.

[8] A. Cowley, B. Cohen, W. Marshall, C. Taylor, and M. Likhachev, "Perception and motion planning for pick-and-place of dynamic objects," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS) (to appear)*, 2013.

[9] S. A. Darzi and Y. Munz, "The impact of minimally invasive surgical techniques," in *Annu Rev Med.*, vol. 55, 2004, pp. 223–237.

[10] M. Granick, J. Boykin, R. Gamelli, G. Schultz, and M. Tenenhaus, "Toward a common language: Surgical wound bed preparation and debridement," *Wound repair and regeneration*, vol. 14, no. s1, pp. 1–10, 2006.

[11] G. Guthart and J. Salisbury Jr, "The Intuitive telesurgery system: Overview and application," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, vol. 1, 2000, pp. 618–621.

[12] B. Hannaford, J. Rosen, D. C. Friedman, H. King, P. Roan, L. Cheng, D. Glozman, J. Ma, S. Kosari, and L. White, "Raven-II: AN open platform for surgical robotics research," *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 954–959, Apr. 2013.

[13] J. Hollerbach, W. Khalil, and M. Gautier, "Model identification," in *Springer Handbook of Robotics*. Springer, 2008, ch. 14, pp. 321–344.

[14] R. Jansen, K. Hauser, N. Chentanez, F. van der Stappen, and K. Goldberg, "Surgical retraction of non-uniform deformable layers of tissue: 2d robot grasping and path planning," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2009, pp. 4092–4097.

[15] L. P. Kaelbling and T. Lozano-Pérez, "Integrated task and motion planning in belief space," *Int. Journal of Robotics Research*, 2013.

[16] ——, "Hierarchical task and motion planning in the now," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011, pp. 1470–1477.

[17] H. Kang and J. T. Wen, "Robotic assistants aid surgeons during minimally invasive procedures," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 1, pp. 94–104, 2001.

[18] A. Lee, Y. Duan, S. Patil, J. Schulman, Z. McCarthy, J. van den Berg, K. Goldberg, and P. Abbeel, "Sigma hulls for gaussian belief space planning for imprecise articulated robots amid obstacles," in *Proceedings of the 26th IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[19] V. Munoz, C. Vara-Thorbeck, J. DeGabriel, J. Lozano, E. Sanchez-Badajoz, A. Garcia-Cerezo, R. Toscano, and A. Jimenez-Garrido, "A medical robotic assistant for minimally invasive surgery," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, vol. 3, 2000, pp. 2901–2906.

[20] S. Patil and R. Alterovitz, "Toward automated tissue retraction in robot-assisted surgery," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, 2010, pp. 2088–2094.

[21] S. Patil, Y. Duan, J. Schulman, K. Goldberg, and P. Abbeel, "Gaussian belief space planning with discontinuities in sensing domains," in *Int. Symp. on Robotics Research (ISRR) (in review)*, 2013.

[22] V. Pradeep, K. Konolige, and E. Berger, "Calibrating a multi-arm multi-sensor robot: A bundle adjustment approach," in *Int. Symp. on Experimental Robotics (ISER)*, 2010.

[23] S. Ramadurai, S. Kosari, H. H. King, H. Chizeck, and B. Hannaford, "Application of unscented kalman filter to a cable driven surgical robot: A simulation study,," in *2012 IEEE International Conference on Robotics and Automation, St. Paul-Minneapolis,*, May 2012.

[24] J. Rawlings, "Tutorial overview of Model Predictive Control," *IEEE Control Systems Magazine*, vol. 20, no. 3, pp. 38–52, 2000.

[25] C. E. Reiley, E. Plaku, and G. D. Hager, "Motion generation of robotic surgical tasks: Learning from expert demonstrations," in *Int. Conf. on Engg. in Medicine and Biology Society (EMBC)*, 2010, pp. 967–970.

[26] E. M. Ritter and D. J. Scott, "Design of a proficiency-based skills training curriculum for the fundamentals of laparoscopic surgery," *Surgical Innovation*, vol. 14, no. 2, pp. 107–112, 2007. [Online]. Available: http://sri.sagepub.com/content/14/2/107.abstract

[27] J. Rosen, B. Hannaford, and R. M. Satava, *Surgical robotics: Systems, applications, and visions*. Springer, 2011.

[28] R. B. Rusu, I. A. Sucan, B. Gerkey, S. Chitta, M. Beetz, and L. E. Kavraki, "Real-time perception-guided motion planning for a personal robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2009, pp. 4245–4252.

[29] J. Schulman, J. Ho, A. Lee, H. Bradlow, I. Awwal, and P. Abbeel, "Finding locally optimal, collision-free trajectories with sequential convex optimization," in *Robotics: Science and Systems (RSS)*, 2013.

[30] S. S. Srinivasa, D. Ferguson, C. J. Helfrich, D. Berenson, A. Collet, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and M. V. Weghe, "HERB: A home exploring robotic butler," *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, 2010.

[31] I. A. Sucan and L. E. Kavraki, "Mobile manipulation: Encoding motion planning options using task motion multigraphs," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011, pp. 5492–5498.

[32] ——, "Accounting for uncertainty in simultaneous task and motion planning using task motion multigraphs," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, 2012, pp. 4822–4828.

[33] R. Taylor, P. Jensen, L. Whitcomb, A. Barnes, R. Kumar, D. Stoianovici, P. Gupta, Z. Wang, E. Dejuan, and L. Kavoussi, "A steady-hand robotic system for microsurgical augmentation," *Int. Journal of Robotics Research*, vol. 18, no. 12, pp. 1201–1210, 1999.

[34] R. Taylor, A. Menciassi, G. Fichtinger, and P. Dario, "Medical robotics and computer-integrated surgery," in *Springer Handbook of Robotics*. Springer, 2008, pp. 1199–1222.

[35] R. H. Taylor, B. D. Mittelstadt, H. A. Paul, W. Hanson, P. Kazanzides, J. F. Zuhars, B. Williamson, B. L. Musits, E. Glassman, and W. L. Bargar, "An image-directed robotic system for precise orthopaedic surgery," *IEEE Trans. on Robotics and Automation*, vol. 10, no. 3, pp. 261–275, 1994.

[36] 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 *IEEE Int. Conf. Robotics and Automation (ICRA)*, 2010, pp. 2074–2081.

[37] J. Wolfe, B. Marthi, and S. J. Russell, "Combined task and motion planning for mobile manipulation." in *Int. Conf. on Automated Planning and Scheduling (ICAPS)*, 2010, pp. 254–258.

[38] H. Zhuang, K. Wang, and Z. S. Roth, "Simultaneous calibration of a robot and a hand-mounted camera," *IEEE Trans. on Robotics and Automation*, vol. 11, no. 5, pp. 649–660, 1995.