# Overcoming Sensor Noise for Low-Tolerance Autonomous Assembly

Brennan Sellner, Frederik W. Heger, Laura M. Hiatt, Nik A. Melchior
Stephen Roderick, Dave Akin, Reid Simmons, Sanjiv Singh

*Abstract*— **The capability to assemble structures is fundamental to the use of robotics in precursor missions in orbit and on planetary surfaces. We have performed autonomous assembly in neutral buoyancy of elements of a space truss whose mating components require positioning tolerances of the same order of magnitude as the noise in the sensor systems used for the docking. Numerous trade-offs, design decisions, and innovations were made during the development of the assembly system in order to both reduce and compensate for the sensor noise. By using relative positioning, decoupling sensing and manipulation, caching high-quality position estimates, and developing a new waypoint-completion metric, we were able to reduce sensor noise to the sub-millimeter level and autonomously assemble components with millimeter tolerances. In this paper, we discuss our approaches to the problem and report the results of a series of autonomous assembly operations.**

## I. INTRODUCTION

Autonomous assembly is a compelling use of robotic technology in orbit and on planetary surfaces. By sending a robotic assembly team ahead of human explorers, infrastructure can be constructed autonomously, allowing the humans to use their time on-planet more efficiently.

Robotic assembly of structures is replete with challenges. We have begun to address the most fundamental of these: assembly of rigid structures with slim tolerances and limited sensing. We have demonstrated the assembly of a portion of the EASE structure (Fig. 1(a)) [1] [2], a large underwater truss used in neutral-buoyancy astronaut assembly training. The assembly was performed using the Ranger robot (Fig. 1(c)) [2] [3] for manipulation and a stereo camera pair as the sole extrinsic sensor. Fiducials placed on all elements allowed them to be localized to within 1.27-1.52 mm in translation and 5 degrees in rotation, the same magnitude as the assembly tolerances.

In this paper, we discuss the challenges inherent in assembly operations with noise of similar magnitude. Myriad approaches were used to reduce or compensate for sensor noise. We focus on four: (a) the use of relative positions to eliminate error in the localization of the sensors; (b) caching high-quality estimates of the relative locations of objects that are difficult to sense; (c) a new more robust metric to determine when a waypoint has been reached; and (d) the decoupling of sensing and manipulation. In addition, we report the results of a series of 24 autonomous assembly operations.

## II. RELATED WORK

### A. Multi-Agent Assembly Systems

The most common multi-agent assembly systems are those in factory settings, where multiple industrial robots work together to assemble products. A system of four industrial robots arranged around a conveyor network for material handling is described in [4]. The stationary robots' interaction is limited to the sharing of common resources such as conveyor systems or storage areas.

Coordinated assembly performed by teams of mobile robots is of particular interest to the space community. We have previously used the Syndicate architecture to support closely coupled coordination between heterogeneous agents to complete various spatial construction assembly tasks [5]. The distributed nature of the architecture allows each agent to be controlled separately. The agents then flexibly form and dissolve distributed control loops as necessary to complete the assembly task at hand. For example, we have used separate agents to sense and manipulate [6]. This servoing methodology allows sensors to be placed without regard to the manipulator's workspace constraints and movements.

Stroupe et al. [7] use the CAMPOUT architecture to coordinate robots with purely behavior-based strategies to perform coupled tasks similar to ours. Their team is homogeneous and performs a single, albeit complex, task. Rather than decoupling sensing and manipulation, their task explicitly links them: two robots transport a rigid beam, maintaining their relative position by detecting the forces applied by the other robot.

### B. EASE and Ranger

Structural assembly has been a prime focus of the Space Systems Laboratory (SSL). Due to the large-scale motions of elements with high mass and moments of inertia, it provides a physical task that requires both strength and fine motor skills. Extensive tests in simulated extravehicular activity (EVA) led to the Experimental Assembly of Structures in EVA (EASE), an EVA flight experiment which was performed successfully during STS 61-B in November, 1985.

The SSL's first dexterous robot, the Beam Assembly Teleoperator (BAT), was designed around those same assembly tasks and correlated to flight data from STS 61-B. BAT first

This work was supported by NASA award NNX06AD23G under subcontract Z627402

B. Sellner, F. W. Heger, L. M. Hiatt, N. A. Melchior, R. Simmons, and S. Singh are with the Robotics Institute of Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, USA brennan@ri.cmu.edu, fwh@cs.cmu.edu, lahiatt@cs.cmu.edu, nmelchio@cs.cmu.edu, reids@cs.cmu.edu, ssingh@ri.cmu.edu

S. Roderick and D. Akin are with the Space Systems Laboratory of the University of Maryland, 382 Technology Drive, College Park, MD 20742 kiwi.2008@mac.com, dakin@ssl.umd.edu

successfully assembled the EASE structure via teleoperation in 1984. Ranger, BAT's successor, has been used to investigate cooperative EVA/robotic assembly of large space structures. The experiments reported here are the first time any portion of EASE has been autonomously assembled.

## III. TASK AND APPROACH

### A. Task

The EASE assembly [1] is a large inverted pyramid structure of six beams and four nodes (Fig. 1(a)). Each edge of the pyramid is approximately 4 meters long. For the collaborative project described here, the objective was to perform a small sub-task of the EASE assembly: autonomously docking one beam to a node.

Establishing a node-beam connection is a two-step process. First, the beam end is mated with one stem of the node (Fig. 1(b)). Then, a sleeve attached to the beam is slid over the connection point to lock the beam to the node. The node-beam connection mechanism must be assembled precisely for the sleeve to slide over the joint without sticking.

### B. Hardware

We utilized the Ranger robot for manipulation, with visual feedback from tracking fiducials provided by a stereo camera pair. Ranger consists of up to three redundant robot arms. For this project, we used the 8-DOF right arm and parts of the 6-DOF positioning leg, as shown in Fig. 1(c). All grasping, docking and manipulation motions were carried out by the arm, while the leg was used for gross motion. Ranger was autonomously controlled by a visual servo process running within the Syndicate task control architecture [5].

Sensing is provided by a stereo camera pair that is positioned independently of Ranger. The stereo pair is used to observe fiducials attached to all relevant parts of hardware, providing the raw images necessary for visual servoing. As camera stability significantly affects sensing accuracy, the cameras were affixed to a tripod.

### C. Visual Tracking and Servoing

The results reported in this paper make use of an improved version of the visual servoing system described in [5] and [8], which consists of two components: visual tracking and visual servoing. Visual tracking uses square ARTag fiducials [9] attached to the relevant portions of the hardware to estimate the positions of objects relative to the cameras. The fiducials are detected in the camera frames using the ARTag library [9]. Our tracking process triangulates the image coordinates of the fiducial corners and fits a 6-DOF transform from the camera to the fiducial, using a singular value decomposition-based method [10]. The location of the object to which the fiducial is attached is then inferred, by applying a premeasured transform from the fiducial to the object's (arbitrary) origin. The final output of the tracking process is a list of transforms from the stereo pair to the detected objects. Since all locations are reported relative to the cameras, the camera location is irrelevant, and no global frame or registration is required. The tracking system used in these results provided data at approximately 5 Hz.

Visual servoing consists of commanding a manipulator to move along a trajectory such that a specified point on the manipulated object or manipulator passes through a series of waypoints defined relative to a target object.

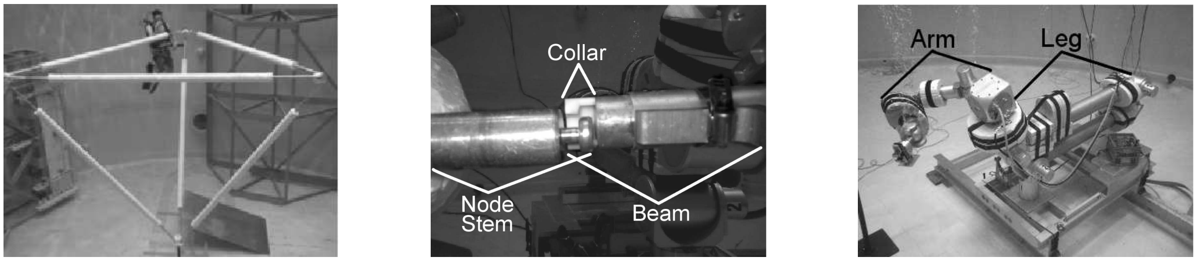
The visual servo algorithm uses the output of visual tracking to compute the position of the moving object relative to the target object. This removes the stereo pair's location from the data, and ensures that the cameras may be positioned solely with concern for capturing the best possible images. The relative position is then smoothed by averaging a number of transforms in SE3 (Alg. 1, [11]). The results reported here use the five latest observations (i.e. $t = 5$ in Alg. 1): this keeps latency low, while smoothing some of the sensor noise.

**Algorithm 1**: SE3 Pose Averaging Algorithm [11]

```
input $T_1, T_2, \ldots T_t$
output $\mu$ = average of transforms
$\mu = I_4$
while $(true)$ do
    for $i$=1:$t$ do
        $dt_i = \ln(\mathrm{inv}(\mu) * T_i)$
    end
    $\Delta\mu = e^{\frac{1}{t}\cdot\Sigma_{j=1}^{t} dt_j}$
    $\mu = \mu \cdot \Delta\mu$
    $est = \mathrm{norm}(\Delta\mu)$
    if $est <= \varepsilon$ then
        return $\mu$
    end
end
```

The visual servo process uses the smoothed data to compute the motion of the end effector necessary to bring the moving object to the next waypoint. In computing this 6-DOF motion, it makes use of the transform between the moving object (if any) and the manipulator. This transform may be prespecified (in the case of a permanent, rigid connection), part of the output of visual tracking, or cached (if the manipulator, moving, and target objects cannot be simultaneously viewed). The computed end effector motion is used as the input to a PID controller, the output of which is sent to Ranger's control software. The PID parameters may be set on a per-waypoint basis; we used a purely proportional controller, with a $p$ value ranging from 0.04 to 0.3.

This approach to servoing has three key advantages: (a) there is no need for an absolute (world) frame, eliminating the need to localize the cameras, along with any associated error; (b) the cameras may be freely repositioned, subject only to the needs of the tracking system; and (c) the system is robust to noise and errors of various types. Because all measurements are relative, the initial positions of the objects are largely irrelevant. In addition, due to the small size and high frequency of the commands sent to the manipulator, slight command errors will not accumulate, since the next relatively-computed command will compensate.



(a) The EASE structure in neutral buoyancy. Ranger (c) was used to autonomously dock a 4 meter beam to a node.

(b) The beam approaches the node stem from behind for docking. The collar on the tip of the beam accepts the node's T-shaped connector.

(c) The Ranger robot in this scenario's configuration. The arm was used for manipulation while the leg provided gross motion.

Fig. 1. Experimental hardware.

## IV. NOISE REDUCTION

The ability to perform manipulation tasks is limited by the system's accuracy in sensing, modeling, and actuation. Each of these stages in the visual servoing process introduces additional error. Accurately determining the 6-DOF pose of fiducials relative to the camera is the most prolific source of challenges, but strategies to reduce error in all areas are described below. Even after applying all of the below methods, the sensing error was of the same order of magnitude as the mechanical tolerances, requiring trade-offs between different sources of error, as will be discussed in Section V.

Most errors in our sensing system are inherent to the use of visual imaging devices. For instance, wider borders were added to the fiducials to counteract blooming effects. When we discovered increased sensing error near the borders of images due to lens distortion, we ensured that our camera calibration procedure methodically covered the entire field of view; the method used is described in [12]. In addition, we selected camera locations that kept the fiducials away from the edge of the image. By using a tripod for the stereo cameras instead of placing them in the hands of a diver, motion blur was reduced, and sensing precision increased by an order of magnitude. After experimentation, we found that the robot's motion was slow enough to not affect the tracking system: velocities of less than 6 cm/s at a range of one meter did not cause noticeable loss of precision.

Another factor to consider when using stereo vision is the nonuniformity of sensor noise with respect to dimension. Estimation of the fiducial's distance from the camera and out-of-image-plane rotations were noticeably worse than in-plane rotation and position within the plane (Figs. 2 and 3). In addition, noise increases as the angle between the fiducial and image plane normals increases. These variations in noise dictate the placement of fiducials and cameras: fiducials should be parallel to the image plane, and dimensions requiring high accuracy should correspond to in-plane translations and rotations. In this scenario, the beam must be precisely placed relative to the node (aligned in Y and Z, with a +X offset in Fig. 2), before being docked (by moving along -X). We situated the cameras such that the initial alignment would be as accurate as possible: this minimized the chance that the beam and node would become jammed during docking, but made the determination of when docking is complete more difficult, as the final docking motion is along the cameras' normal (see Section V-B). Similarly, aligning the beam such that its axis is parallel with the node's stem requires a high degree of rotational accuracy in two dimensions (about X and Z in Fig. 2). Our camera placement allowed the accurate estimation of rotation about X in Fig. 2, while rotation about Z was less accurate. The mechanism is slightly more tolerant to Z rotations, and we did not encounter any significant problems with rotational alignment.

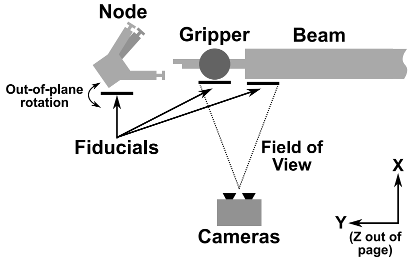


Fig. 2. The camera image plane (YZ) is perpendicular to the page and parallel with the fiducials. Accuracy is high when detecting in-plane translation (along Y or Z) and in-plane rotation (about X). Noise is larger for out-of-plane rotations (about Y or Z) and translations towards or away from the camera (along X).

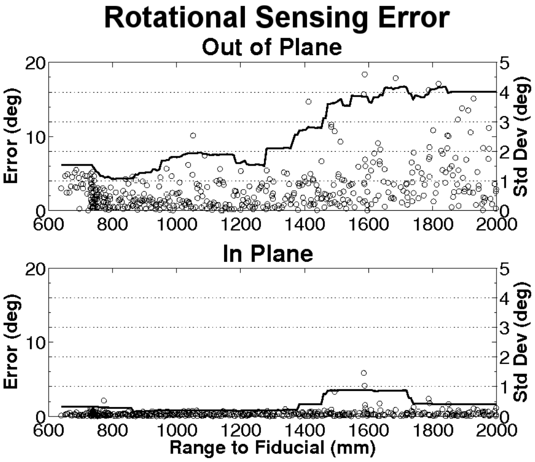


Fig. 3. Rotational sensing error is a function of the range from the camera to the fiducial, and varies with the axis being estimated: rotations parallel to the image plane are detected more accurately than those out-of-plane. These are plots of out-of-plane (upper; about Z in Fig. 2) and in-plane (lower; about X in Fig. 2) rotation as a function of range to the fiducial. 'o's are individual error measurements; the solid line is a windowed standard deviation of the error, and is an estimate of the measurement noise.

As discussed in Section III-C, visual tracking consists of the determination of the position of an object, and specific points on it, based on knowledge of the fiducial attached to it. Clearly, reducing error in this case requires a precise model of the points of interest and fiducials relative to one another. We used a measurement arm[1] to determine their relative positions with sub-millimeter precision. Modeling accuracy can also be improved by carefully choosing the location of the fiducial on the object. Although the choice of fiducial position is limited by available mounting hardware and camera placement, there is an advantage to placing the fiducials as close as possible to the points on the objects that will be brought into contact. Since some amount of error is inevitable in sensing the location of the fiducials, we benefit from a shorter moment arm in the transformation from fiducial to point of interest. We placed fiducials on the node, beam, and gripper as close as possible to the docking faces of the node and beam and the jaws of the gripper without impeding docking operations (Fig. 2).

The final source of error is actuation. We must consider both the hardware's ability to faithfully produce commanded movements in the end-effector's coordinate frame and the accuracy of the commands themselves as generated by the visual servoing process. Ranger's relative positioning accuracy is 0.495 mm, with a standard deviation of 0.408 mm. This is a measurement of how accurately the robot can place its end effector relative to its previous position. While quite low, this still represents a full third of the mechanical tolerance between the EASE beam and node.

Since Ranger's startup calibration procedure does not allow the gripper's command coordinate frame to be known precisely, the frame must be estimated by the operator, leading to slight errors between the frame in which the visual servoing procedure is producing commands and the frame in which they are being executed. These sources of error are effectively mitigated by the relative nature of our visual servo approach, as described in Section III-C. Finally, the latency between tracking fiducial positions and producing movement commands while the arm is moving may introduce small errors. In practice, no appreciable error is introduced because the robot's movements are slow and the latency is small.

## V. Noise Compensation

While we were able to significantly reduce sensor noise, it remained too large to perform an EASE docking, forcing a series of trade-offs between sources of noise and error. By caching high quality estimates of transforms that were unlikely to change, we were able to achieve more accurate tracking during visual servoing operations, in exchange for an inability to correct some unlikely errors. We developed a new method for determining waypoint achievement that factors out elements of the sensor noise, at the cost of generality. Finally, by decoupling sensing and manipulation we were able to avoid the need to trade off between manipulation and sensing motion constraints. We accomplish this decoupling by ensuring that sensor placement does not affect manipulation, as they are controlled by different agents.

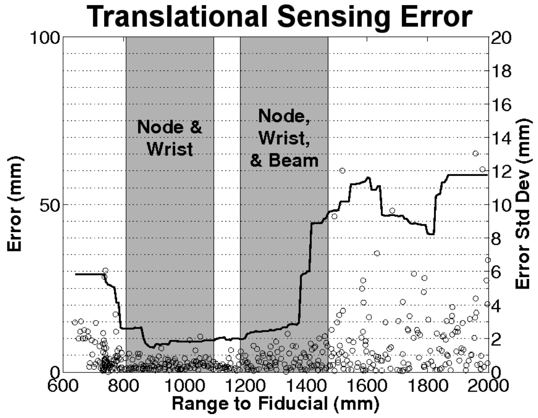


Fig. 4. Translational sensing error and noise as a function of the range to a target consisting of two fiducials attached to a rigid plate. The reported error is the Euclidean distance between the actual and estimated position of one fiducial relative to the other. The errors are plotted as grey 'o's, and the standard deviation as a solid line. The standard deviation is calculated by moving a window 75 data points wide across the data set. The grey regions represent the ranges at which the camera could be placed if simultaneously observing either the node and wrist or the node, wrist, and beam.

### A. Transform Caching

Excessive distance between the camera and fiducial is a major source of sensor noise. In order to characterize this effect in our system, we collected observations of two rigidly-joined fiducials, with a known fiducial-fiducial transform, as they were moved along the normal of the cameras' image plane. We compared the sensing system's estimates of the inter-fiducial transform with ground truth to determine the error of each reading, measured as the Euclidean distance between the estimated and true transforms (Fig. 4). The standard deviation of the estimates is also very relevant, as that is a measure of how noisy the sensing system's output will be (solid line in Fig. 4).

As can be seen in Fig. 4, pose estimation noise increases rapidly as the distance to the fiducial rises. As the fiducials become smaller in the image, triangulation becomes less accurate. Noise also increases if the fiducials become too close to the cameras, as our tracking system extrapolates missing fiducial corners as they move off-image, yielding much less accurate estimates. Camera positioning is straightforward for simple two-body servoing operations, such as grasping the beam or sliding the locking sleeve. However, when operating with chains of objects, there is an opportunity to trade off between different sources of error and noise.

For instance, during the docking operation, the arm has grasped the beam, and is moving it to dock with the node. The servoing routine is attempting to make the node and beam stems coincident by issuing commands to the arm. As a result, there are two transforms that are relevant in order to calculate the desired arm motion: the node to the beam and the beam to the arm's gripper. The obvious approach is to simultaneously observe all three bodies (Fig. 5(a)) and update both transforms throughout the docking operation. This method has the advantage of automatically detecting any slippage of the beam in the gripper that may occur, but forces the camera to be at least 1.2 meters from the observed

[1] In this instance, we used a Faro Platinum arm [13]: a passive 7-DOF arm that can be used to dimension objects to a high degree of accuracy.

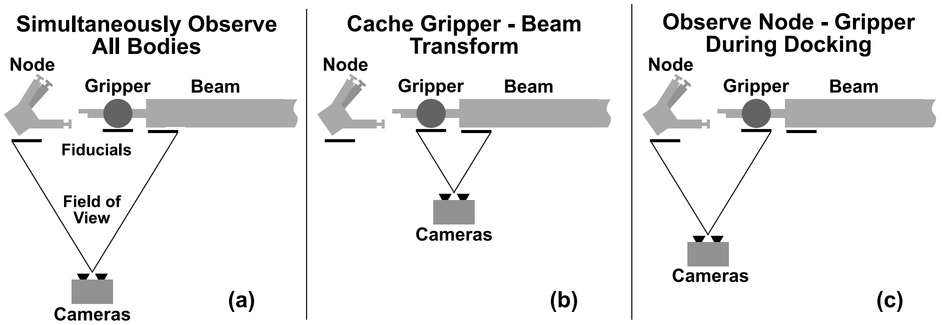

Simultaneously Observe All Bodies
Node
Gripper
Beam
Fiducials
Field of View
Cameras
(a)
Cache Gripper - Beam Transform
Node
Gripper
Beam
Cameras
(b)
Observe Node - Gripper During Docking
Node
Gripper
Beam
Cameras
(c)

Fig. 5. A top-down schematic of camera positions while observing different combinations of fiducials. (a) and (c) correspond to the grey regions on the right and left of Fig. 4, respectively.

fiducials. Moving the camera out to this range increases sensor noise by 1-8 mm, as the right shaded region in Fig. 4 shows; this is too much noise to successfully dock.

Our solution is to add an initial caching step, where 90 observations are collected of the gripper-beam transform from close range (Fig. 5(b)), then averaged using Alg. 1 to obtain an accurate estimate. We determined the number of observations to collect by plotting the standard deviation of the set as a function of the number of readings. We estimated where the plot plateaued, and collected twice as many observations.

During docking, the cameras are moved to observe the node and gripper fiducials only, allowing the cameras to be placed much closer (Fig. 5(c), left region in Fig. 4). The cached transform between the gripper and beam is then used to calculate the necessary gripper movement. The disadvantages of this approach are that: (a) any slippage of the beam will go undetected; (b) an additional camera repositioning is required; and (c) additional time is required for the initial gripper-beam estimation step. If the undetected slippage is large enough, the calculated manipulator commands will be incorrect enough to prevent docking. However, small errors will be ameliorated, as the manipulator motion is recalculated at every iteration. In practice, we observed no measurable slippage of the beam once the gripper was closed securely. This approach allows cameras to be placed closer to the observed fiducials, reducing measurement noise. The results reported in Section VI were achieved using this caching approach.

### B. Waypoint Completion

One of our more persistent problems was determining when docking was complete: that is, when the final visual servo waypoint had been achieved. This is particularly challenging because the movement from the penultimate waypoint to the final waypoint is along the cameras' normal (X in Fig. 2), where sensor noise is high. Initially, we relied on a system of per-axis tolerances around each waypoint to detect when it was achieved (Fig. 6). Note that the waypoint completion is evaluated in six dimensions, but here is represented in 2-D for clarity. There were two intended methods by which a waypoint could be completed. The primary way was to approach the waypoint and register a specified number of consecutive readings inside the inner tolerance bound (trace 1 in Fig. 6). Alternatively, in cases where the inner bound was very small or to appropriately handle offset approaches (trace 2 in Fig. 6), a waypoint also was considered to be completed if a specified number of consecutive measurements were registered inside the outer tolerance bound and the error was growing in at least one dimension. This completion condition was intended to detect cases where the arm passed by the waypoint just outside the inner tolerance bound. Under the assumption that, in general, the measurements will get progressively closer to the waypoint until it is reached (or just barely missed), this approach worked satisfactorily in previous scenarios [5] [6].

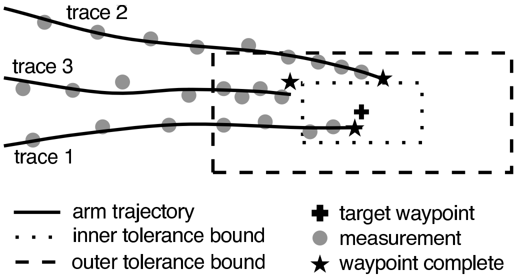


Fig. 6. Our initial implementation to determine waypoint completion. A waypoint was considered achieved if either three consecutive measurements were recorded inside the inner tolerance bound (trace 1) or five consecutive measurements fell into the outer tolerance bound and the error along at least one dimension increased (trace 2). Sensor noise in the tracking system caused the second condition to trigger too early (trace 3).

During operations for the EASE assembly task, many waypoint completions were triggered too early. The high noise in the system resulted in many situations similar to trace 3 in Fig. 6. While the arm itself was moving as desired, the measurements varied significantly between tracking iterations. As the arm approached the waypoint and entered the outer tolerance bound, noisy measurements would show error increasing in directions orthogonal to the arm's motion, which triggered waypoint completion. We modified the size and shape of the inner and outer tolerance bounds in an effort to eliminate the problem, but the noise was too high to find a balance that would both minimize early completions and still detect a legitimate achievement of the waypoint in an acceptable amount of time.

Because the arm motion tracked the desired trajectory quite well, we developed an approach that would ignore error orthogonal to the primary axis of motion. To do so, we compared each measurement to the previous and the target waypoints. Consider two vectors: one from the previous to the current waypoint (or the manipulator's initial position if

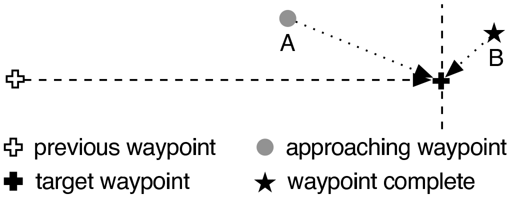


Fig. 7. The new implementation to detect waypoint completion considers a plane normal to the vector from the previous waypoint to the current target waypoint. A waypoint is complete if a fixed number of consecutive measurements fall beyond that plane (e.g. measurement B).

there is no previous waypoint), and another from the current measurement to the target waypoint. As long as the dot product of the two vectors is positive, the measurement is on the approach side of the target waypoint (A in Fig. 7). When the sign of this inner product flips, we conclude that the waypoint has been achieved translationally (B in Fig. 7). Rotational errors are evaluated as in the previous approach: a waypoint is achieved when the rotational error is within bounds and the inner product has remained negative for a number of consecutive readings.

This approach eliminates false positives caused by sensor noise orthogonal to the primary axis of motion, at the risk of determining that a waypoint has been achieved despite large orthogonal error. Because our visual servoing system approaches the target along a series of waypoints, this is not generally an issue, as for very fine-tuned maneuvers the waypoints can be made to be close enough to eliminate or correct this error. This approach was developed after two-thirds of the experiments reported in Section VI; insufficient data is available to unequivocally state its effectiveness, but the trends are promising (see Table I).

### C. Separation of Sensing and Manipulation

As discussed in Section V-A, the proximity of the cameras to the fiducials has a significant impact on pose estimation error. In tightly constrained workspaces, placing cameras close to the assembly can interfere with both the manipulator's movements and the cameras' unblocked view of the workspace. This proved to be the case when we attached the cameras to Ranger at the base of the arm: the arm often blocked the cameras, and the intersection of the cameras' field of view with the arm's workspace was too restrictive.

In our previous work, we have used independent sensing and manipulation agents to decouple the two tasks. This is possible only because our visual tracking and servoing system operates entirely using relative positions: the location of the sensor does not need to be known, allowing its placement to be constrained solely by its characteristics. The second Ranger arm (similar to the manipulation arm we currently use) is an obvious choice for such a sensing agent, and may be utilized in the future for camera positioning, but was unavailable during our experiments. Instead, we placed the cameras under the control of a different 'agent': we mounted them on a tripod that divers manually positioned at a series of predetermined locations. This was the only human involvement in the assembly procedure, and occurred only at two distinct points in the sequence: when moving the cameras from the beam grasping location to where the beam-gripper transform would be cached, and from there to a point where the node and gripper could be observed. In addition, we placed the cameras on the opposite side of the assembly from the manipulator (Fig. 8). This avoided constraining the manipulator's movements while providing the cameras with a clear, unobstructed view of the assembly. The results reported below were achieved using this camera positioning strategy.

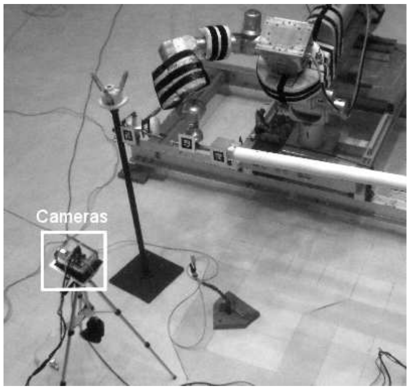
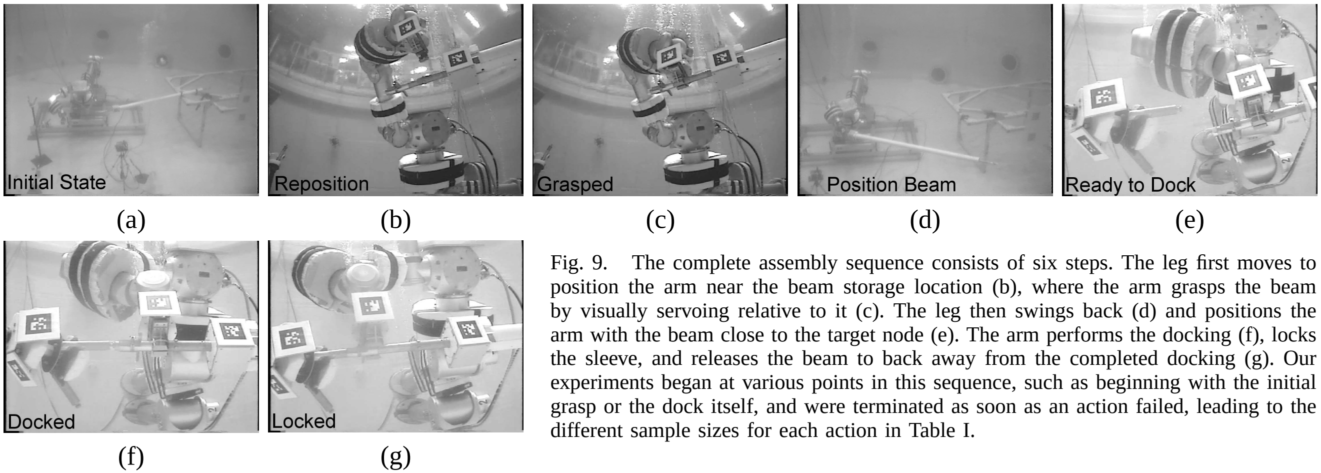


Fig. 8. The placement of the cameras, shown here on a tripod, in relationship to the beam, node, and manipulator during a docking operation.

## VI. RESULTS

The assembly sequence consisted of a blind joint move to the general location of the stored beam, a visual servo to grasp the beam, another blind move back to the approximate location of the node, visually servoed docking and locking maneuvers, and a blind release action (Fig. 9). Over the course of several days at the Space Systems Lab's Neutral Buoyancy Research Facility, we performed 24 runs of various portions of this sequence. Due to the limited availability of Ranger and a lack of safety divers, we had insufficient time for large numbers of identical experimental runs. Instead, we performed various segments of the scenario (including several end-to-end runs), and have aggregated the resulting data here. Table I details the number of successes, near successes, and failures for each stage in the process. "Near successes" are operations that came very close to succeeding: for instance, a docking attempt whose final waypoint was deemed to have been finished 1-2 mm early would be considered a near success.

The most persistent problem during these experiments was that of waypoint completion detection, as discussed in Section V-B. The new detection method was implemented starting with run 13, and was tuned throughout the remaining runs. As can be seen from the bottom row of Table I, the system became more reliable as development proceeded. The figures in the bottom row plot the result of each run, with the run number increasing from left to right. If nothing is plotted for a particular run, that segment of the sequence was not attempted during the given run. The entire sequence consumes roughly thirty minutes, necessitating exercising portions of it in isolation. In combination with the very limited robot and diver time available, this constrained the number of runs we were able to perform overall, as well



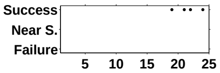
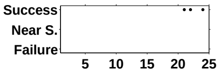
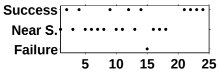
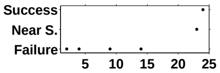
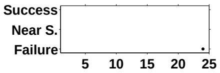
(a) (b) (c) (d) (e)

(f) (g)

Fig. 9. The complete assembly sequence consists of six steps. The leg first moves to position the arm near the beam storage location (b), where the arm grasps the beam by visually servoing relative to it (c). The leg then swings back (d) and positions the arm with the beam close to the target node (e). The arm performs the docking (f), locks the sleeve, and releases the beam to back away from the completed docking (g). Our experiments began at various points in this sequence, such as beginning with the initial grasp or the dock itself, and were terminated as soon as an action failed, leading to the different sample sizes for each action in Table I.

TABLE I

EXPERIMENTAL RESULTS

| Task | Joint Moves | Grasp | Dock | Lock | Release |
|---|---|---|---|---|---|
| **Success** | 4 (100%) | 3 (100%) | 9 (41%) | 1 (17%) | – |
| **Near Success** | – | – | 12 (55%) | 1 (17%) | – |
| **Failure** | – | – | 1 (5%) | 4 (67%) | 1 (100%) |
| **Over Time** |  |  |  |  |  |

as preventing us from testing the release maneuver before attempting it for the first time during experimentation.

By run 24, we were able to complete the entire sequence autonomously, with the exception of the untested release maneuver. The only human involvement was the repositioning of the external camera pair.

## VII. FUTURE WORK

Future work on this scenario will focus on further reducing sensor noise and performing the entire EASE assembly. Moving to higher resolution cameras is one possible approach to reducing noise. Since tracking accuracy varies with the orientation of the image plane, adding an additional camera pair oriented orthogonally to the existing pair should greatly reduce noise. Assembling the entire EASE pyramid will require management of the arm's workspace around a developing structure, as well as the creation of heuristics to guide the automated placement of cameras, since most of the assembly will be out of reach of a tripod on the floor.

## VIII. CONCLUSION

This paper has examined some of the difficulties inherent in autonomous assembly, especially when sensor noise is of the same magnitude as the mechanical tolerance. We have improved camera placement by caching some transforms and separating sensing and manipulation, allowing a reduction in camera-fiducial range and sensor noise. In addition, we have developed a method for the evaluation of waypoint completion that is more tolerant to the sensor noise that remains. We were able to sufficiently reduce and compensate for sensor noise to allow the autonomous assembly of one element of the EASE structure under neutral buoyancy conditions, and reported the results of a series of autonomous assembly operations. Autonomous structure assembly is a rich area of research with many challenges and opportunities that have only begun to be addressed.

## REFERENCES

[1] D. L. Akin and M. L. Bowden, "EVA, robotic, and cooperative assembly of large space structures," in *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, March 2002.

[2] D. L. Akin, J. C. Lane, B. J. Roberts, and S. R. Weisman, "Robotic capabilities for complex space operations."

[3] J. L. Parrish, D. L. Akin, and G. G. Gefke, "The ranger telerobotic shuttle experiment: Implications for operational eva/robotic cooperation," in *Proceedings of the 30th International Conference on Environmental Systems*, Toulouse, France, July 10-13 2000.

[4] P. Valckenaers, H. V. Brussel, L. Bongaerts, and F. Bonneville, "Programming, Scheduling, and Control of Flexible Assembly Systems," in *Computers in Industry*, vol. 26, no. 3, Aug. 1995, pp. 209–218.

[5] B. Sellner, F. Heger, L. M. Hiatt, R. Simmons, and S. Singh, "Coordinated multi-agent teams and sliding autonomy for large-scale assembly," *Proceedings of the IEEE*, vol. 94, no. 7, July 2006.

[6] F. W. Heger, L. M. Hiatt, B. Sellner, R. Simmons, and S. Singh, "Results in Sliding Autonomy for Multi-Robot Spatial Assembly."

[7] A. Stroupe, T. Huntsberger, A. Okon, and H. Aghazarian, "Precision Manipulation with Cooperative Robots," in *Multi-Robot Systems: From Swarms to Intelligent Automata*, L. Parker, F. Schneider, and A. Schultz, Eds. Springer, 2005.

[8] D. Hershberger, R. Burridge, D. Kortenkamp, and R. Simmons, "Distributed Visual Servoing with a Roving Eye," in *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, Takamatsu Japan, October 2000.

[9] M. Fiala, "Artag, a fiducial marker system using digital techniques," *Computer Vision and Pattern Recognition, 2005. IEEE Computer Society Conference on*, June 2005.

[10] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 698–700, September 1987.

[11] V. M. Govindu, "Lie-algebraic averaging for globally consistent motion estimation," in *Proceedings of the IEEE Converence on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[12] J.-Y. Bouguet, "Camera calibration toolbox for matlab," June 2008, http://www.vision.caltech.edu/bouguetj/calib_doc/.

[13] "Faro platinum arm, company website: http://www.faro.com/content.aspx?ct=di&content=pro&item=2&subitem=2."