JOURNAL OF SPACECRAFT AND ROCKETS
Vol. 32, No. 4, July–August 1995

Verification Tests of Automated Robotic Assembly of Space Truss Structures

Marvin D. Rhodes* and Ralph W. Will†
NASA Langley Research Center, Hampton, Virginia 23681
and
Cuong C. Quach‡
Lockheed Engineering and Sciences Company, Inc., Hampton, Virginia 23666

A multidisciplinary program has been conducted at the Langley Research Center to develop operational procedures for supervised autonomous assembly of truss structures suitable for large-aperture antennas. The hardware and operations required to assemble a 102-member tetrahedral truss and attach 12 hexagonal panels were developed and evaluated. A brute-force automation approach was used to develop baseline assembly hardware and software techniques. However, as the system matured and operations were proven, upgrades were incorporated and assessed against the baseline test results. These upgrades included the use of distributed microprocessors to control dedicated end-effector operations, machine vision guidance for strut installation, and the use of an expert-system-based executive-control program. This paper summarizes the developmental phases of the program, the results of several assembly tests, and a series of proposed enhancements. No problems that would preclude automated in-space assembly of truss structures have been encountered. The test system was developed at a breadboard level and continued development at an enhanced level is warranted.

Introduction

FUTURE space missions are anticipated to involve telescopes with apertures larger than that of the Hubble Telescope, which cannot be launched in one piece using the Space Shuttle. These and other structures are likely to require assembly on orbit. To accommodate this need a number of research studies have been conducted to develop in-space assembly techniques. For example, the ACCESS flight experiment, conducted on the 23rd Space Shuttle mission, demonstrated that astronauts could assemble and maneuver large structures in space.[1] Since that experiment, many additional tests performed in underwater extravehicular activity (EVA) simulations have substantiated the feasibility of erectable assembly operations.[2,3] Space is, however, a hazardous environment for manned operations, and alternative methods for erecting structures on orbit are being explored.

Several years ago a program was initiated at the Langley Research Center to develop automation techniques using a robotic manipulator for assembly of space systems. These techniques are based on supervised autonomy in order to minimize crew resources. The operator interactions have avoided time-critical, in-the-loop functions such as teleoperation so that they may be performed remotely from Earth. The operator is required to intervene only when the automated system encounters a problem which it is not prepared (programmed) to resolve. This mode of operation minimizes the need for critical astronaut crew time either in an EVA mode or an intravehicular activity (IVA) mode.

The purpose of the current paper is to present an overview of the program, describe the current capabilities of the system, identify some of the lessons learned from the assembly tests, and discuss the potential for additional enhancements that are necessary to expand the capability and thereby develop a reliable in-space assembly system.

Received Oct. 29, 1993; revision received Jan. 5, 1995; accepted for publication Jan. 13, 1995. Copyright © 1995 by the American Institute of Aeronautics and Astronautics, Inc. No copyright is asserted in the United States under Title 17, U.S. Code. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Governmental purposes. All other rights are reserved by the copyright owner.

*Aerospace Engineer, Structures Division, MS 190. Member AIAA.

†Aerospace Engineer, Information and Electromagnetic Technology Division, MS 152D.

‡Aerospace Engineer, Langley Programs Office MS 152D, NASA Langley Research Center, Hampton, VA 23681.

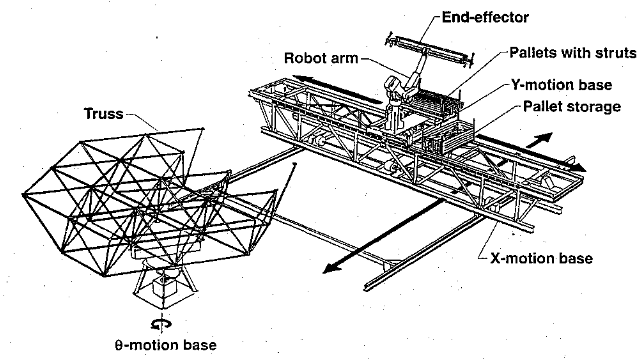
Objective

The objective of this research program is not fundamental robotics or structures research per se, but the development and evaluation of hardware concepts, software requirements, and operator interface and control techniques that are necessary for the assembly of truss structures using robotics. An experimental test facility was developed so that realistic hardware problems and constraints that are likely to be encountered will be identified and resolved through research-level testing prior to the development of a flight system.

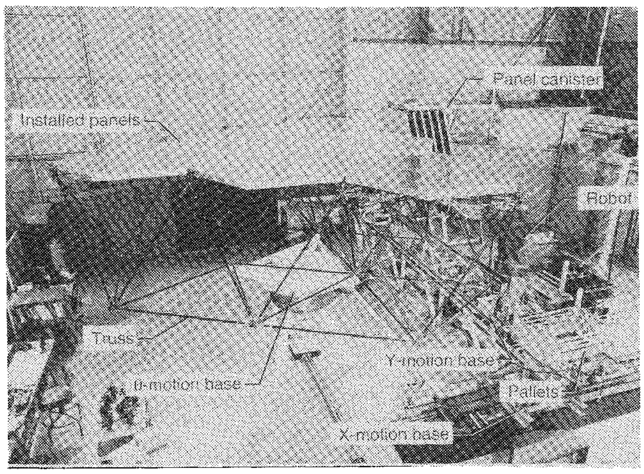
Approach

The assembly task was addressed from the total system viewpoint as opposed to performing a series of component bench tests. A test facility was developed, and the assembly activities performed in the facility have been based on an evolutionary approach. Individuals with backgrounds and experience in structures and robotics were brought together in a focused program that involved complete assembly of a generic structural unit using current state-of-the-art techniques. The schedule established at the inception of the program did not permit the incorporation of new and unproven technology. Therefore, the first tests involved brute-force automation of the various assembly tasks using many off-the-shelf, commercially available components. The first tests established a baseline from which progressively higher levels of system automation capability were incorporated and assessed. New features were added in a sequential manner to individually assess their effect against the baseline test results. The assembly procedures were developed around the requirements of the hardware, and the software control program was developed using standard coding and software engineering methods. The automated system performs all operations required, from the removal of components from supply canisters to their installation into the structure. No handoff or manual assistance is required during the performance of a task. Also, the hardware is designed so that each operation is totally reversible and disassembly is as easily achieved as assembly. Reversibility is critical to the recovery from any condition or commanded operation that is not totally successful.

The hardware components in the automated assembly system are illustrated by the schematic in Fig. 1a and the photograph in Fig. 1b. The hardware consists of the following: a robot arm mounted on a Cartesian motion-base system to permit positioning of the base of the robot and transport of components, the truss structure with attached



a) Sketch of test facility



b) Photograph of test facility

Fig. 1 Hardware test components for the automated assembly system.

panels mounted on a rotating motion base, specialized end effectors for fetching struts and panels from storage pallets and installing them in the structure, and a storage location for the strut pallets. The robot arm is an electrically driven six-degree-of-freedom industrial manipulator that was selected for its reach envelope, payload capacity, and positioning repeatability. No modifications have been made to the arm other than those that are available from commercial vendors. The Cartesian and rotational motion bases serve to extend the reach envelope of the arm; however, controls of the motion bases and the robot are not integrated. The end effectors are specialized tools that are mounted on the wrist of the robot and perform all functions required to complete both installation and removal of the struts and panels. A commercially available force–torque load cell is mounted between the end effector and the robot arm. The system is a terrestrially based research tool for the development of techniques, specialized end-effector components, computer software design, control algorithms, and operator interface requirements.

The procedures being developed are focused on the assembly of a tetrahedral truss with 2-m-long strut members. This truss has the same geometric configuration as that proposed for many space systems, and the length of the strut members is representative of that proposed for several telescopes and precision antennas. The test truss has 102 struts connected by 31 nodes. It has enough variety in assembly operations to be a challenging development task and enough repetition to gain insight into system operational reliability.

The system operator–monitor is located in a control room remote from the hardware test operations. Status information available to the operator is via: 1) a direct window view of the system, though the operator is approximately 12 m from the center of the truss; 2) limited video surveillance from two facility cameras with pan/tilt and zoom control; 3) two cameras (fixed position and fixed focus) mounted in the vicinity of the end-effector mechanisms; and 4) a keyboard interface with the control system.

Considerable effort was initially devoted to the development of an operator interface that is convenient and simple to use, and the

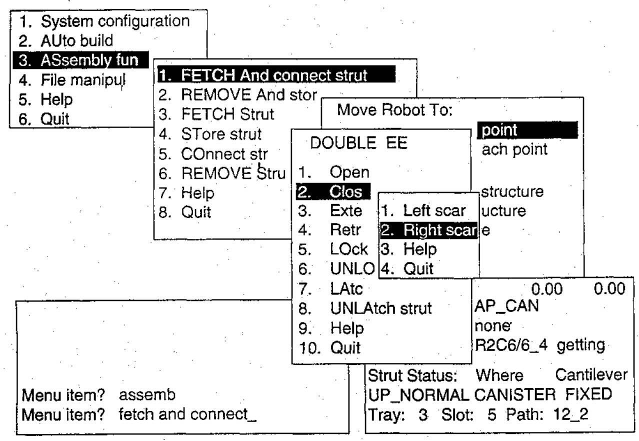


Fig. 2 Operator interface menu display.

interface has remained stable throughout the program. A menu command system was developed whereby the operator may select from various menu options that are displayed on a computer screen. The basic menu layout is illustrated in Fig. 2. Commands range in complexity from the highest-level functions such as "Assembly fun[ction]" shown in the far left box at the top of the figure, to the lowest-component-level commands that activate individual mechanisms on an end effector. The high-level commands are automatically decomposed in a highlighted, traceable fashion to the component level in order to keep the operator abreast of the path through which the current command was derived. The system status is displayed in other boxes on the operator's monitor. In addition to the basic commands required to initiate the assembly functions, a capability to pause and reverse any operation at virtually any time has proven to be necessary. To implement the pause and reverse required considerable decomposition logic, because the order and functions involved in the reverse sequence for most operations are not the same as the reverse order of the forward sequence. Therefore, special reverse procedures had to be developed. Details of the truss hardware and facility components are available in Refs. 4 and 5.

Developmental Phases

Many of the developments in this program have proceeded concurrently; however, for reporting purposes it is convenient to review the progress of the program as if it occurred in distinct phases. As indicated previously, brute-force automation was used to accomplish each hardware and software task for truss assembly in the initial phase. Four complete assembly and disassembly tests were conducted, followed by an evaluation of the test procedures and results. The evaluation was successful in identifying problems of a general nature and highlighting the need for system upgrades that could be incorporated in an orderly manner. The test results established a quantitative baseline, which has been used subsequently to assess the impact of new operational features. The second phase included the addition of several significant advancements to the control system. This phase also involved a more complex assembly task, the installation of reflector-type panels. Two complete assembly and disassembly tests were performed to evaluate these upgrades. The third phase, currently underway, includes truss assembly operations similar to the first two; however, the structural configuration is slightly different, and automated robot arm path planning and strut sequence planning have been added.

Phase I—Summary of Planar Truss Assembly

A photograph of a strut being installed during assembly of the truss is shown in Fig. 3. The robot, end-effector, truss, and motion-base support systems for the truss and the robot are all shown in the photograph. Truss assembly is accomplished by removing struts from a canister located immediately behind the robot and installing them into the truss structure. To install the strut, the end effector must grapple and hold the joining receptacle on the truss node during insertion to provide sufficient support and stability to in-

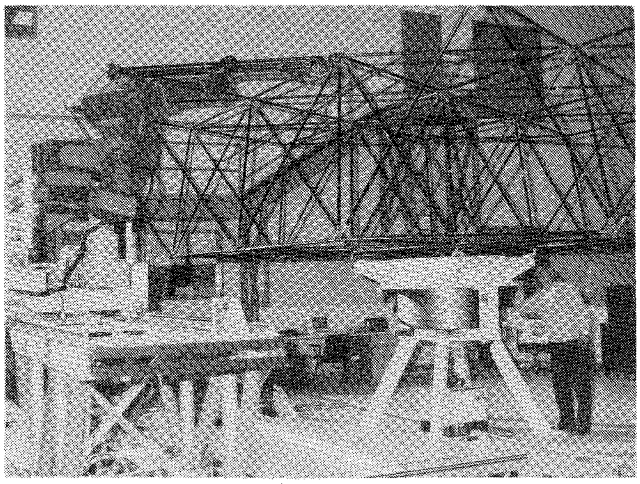


Fig. 3 Photograph of a strut being installed by the end effector during truss assembly.

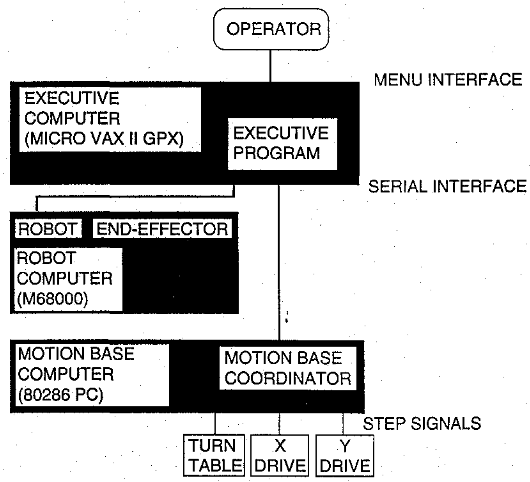


Fig. 4 Schematic of computer control system implemented for phase I tests.

sert and lock the joining mechanism. After grappling the receptacle, the end effector mechanism pushes the strut forward, driving joint components together. The end effector then locks the joints to secure the strut in the truss. The locking operation eliminates free play in the joint to enhance the predictability of the truss's structural response.

To reach the installation position in the truss, the robot arm is moved from the strut canister along a path that is composed primarily of predefined and stored ("taught") coordinates, or points. The points for each path include both position and orientation components $(x, y, z, \phi, \theta, \psi)$ and were developed by stationing an observer near the robot to direct the operator in positioning the robot arm. The paths require that the robot base be at specified locations with respect to a unit cell with in the truss. Nineteen independent paths are required for the installation of all 102 struts. The initial plan was to perform all operations using only "taught" robot points and accurately determined motion-base positions. However, during checkout tests it was determined that when the end effector grappled the truss, alignments to within several thousandths of a centimeter were required of the end-effector mechanisms to reliably perform the insertion operation. This accuracy was beyond the capability of the robot arm and motion-base positioning systems, particularly in the presence of thermal and mass variations; hence, realignment using active force–torque feedback was required. A control algorithm was developed to reposition the robot arm, and the algorithm operations are performed outside the robot control loop. Passive guidance features designed into the end effector and joints are responsible for guiding the robot to the proper installation position. Both the robot arm and the truss structure are quite stiff, and strut insertion is easily performed when forces are reduced below ± 2.2 N and moments below ± 0.56 N m, which correspond to approximately ± 0.005 cm and 2.0 deg of positioning accuracy. This exemplifies some of the modifications that were necessary to perform the initial tests using brute-force automation. The process is time-consuming, and the reliance on points taught in a $1g$ environment to position an end effector for installation makes this procedure unacceptable for space assembly.

All installation functions are managed and controlled by several digital computers that are serially connected as shown schematically in Fig. 4. The system executive program transfers commands and status information across various processors and reports current information to the system operator. The executive program resides on a workstation and is run from the operator's console. Also, the robot arm path-control logic and a motion-base collision-control algorithm are located on the workstation processor. The robot carriages are controlled by a personal-computer indexer board, and commands to this board are generated by a driver program. The robot arm motions are controlled by a special processor developed by the robot manufacturer. The robot processor includes the "taught" points that define the paths for strut installation, thereby minimizing the information that has to be transferred between processors. The robot processor controls the operations of the end effector through built-in analog-to-digital converters. The use of these converters helped to bring the system to an operational status in fairly short time.

Data from two of the four complete assembly and disassembly tests were reduced and analyzed, and the results indicate that the average time required to remove a strut from its pallet and install it is the truss is about 9.2 min. The reverse operation for disassembly requires about 8.7 min. These times include approximately 2 min that could be eliminated by enhancements such as modifying the robot control system to incorporate in-loop force and torque compliance. After allowing for effects of this type, and others to be discussed in a subsequent section, the test results were examined to estimate the installation time for in-space operation. The analysis indicate that an installation time of 4–5 min could be expected. One factor that limits faster operations for an in-space assembly system is the reliance on the operator to monitor the system continuously and to intervene if it appears that a collision may occur. Since the end effector is required to move in close proximity to previously installed struts, the operator must be able to react in time to avert a collision; therefore higher arm speeds are not desirable.

Significantly more important than the operational time required is the overall reliability and robustness of the system to permit the operator to deal effectively with errors. An error was defined to occur whenever an anomaly required control to be relinquished to the operator, and to resolve the anomaly the operator was required to modify the standard command sequence. The test results indicate that during the two assembly–disassembly tests 193 error conditions were successfully resolved by the operator from the console during the installation and removal of 408 struts. Most of the errors were the result of end-effector misalignments that required the operator to command adjustments to the position of the end effector. All of these positioning errors were outside the range of misalignments that could be accommodated by the force–torque algorithm, and they typically required only about 1.5 to 2 min to analyze and correct. Manual intervention to correct a problem (equivalent to an in-space EVA) was necessary only three times during these tests. Although the operator was capable of handling nearly all of the error conditions encountered, the system at this stage of development does not have adequate robustness for space operation. Enough operations have been performed, however, to identify many conditions that are associated with errors, and improvements to correct some of them were incorporated in subsequent phases.

Phase II—Integrated Truss and Panel Assembly

In the second phase of the program the computer architecture was upgraded to make it more closely approximate an anticipated in-space operational system by installing microprocessors on the end effectors, incorporating machine vision guidance for strut assembly, and implementing an expert-system-based executive-control program. In addition to the computer upgrades, a more complex assembly task, the installing of reflective-type panels, was added. This required an additional end effector, its task sequences, and automated end-effector exchange procedures to be implemented. Two additional assembly and disassembly tests were conducted to evaluate these enhancements. These capabilities and the assembly test results are discussed in the following subsections.

End-Effector Control

The first step in updating the control system was to remove the end-effector commands from the robot processor and rehost them on single-board computers. A microprocessor was serially connected to the executive computer, to perform this task. Associated electronics interface boards were also designed and fabricated. Both the single-board computers and the interface boards are mounted directly on the end effectors. With this capability each end effector can accept assembly-level commands such as "INSTALL" and "REMOVE" directly from the executive, and the sequence of component-level commands is stored on the microprocessor and issued specifically for the object that the particular end effector is designed to accommodate. Sensor verification conditions and operator-error recovery information for each end effector are also stored on the processor, which significantly reduces the size and complexity of the executive program. The value of this approach became readily apparent when the end effector for the installation of panels was incorporated. Several other benefits were also realized: 1) the speed at which end-effector operations are performed increased significantly because the processing of sensor checks is not delayed by communication between processors; 2) the number of power conductors and signal lines to the end effectors was significantly reduced; 3) the device-level programs and algorithms for each end effector could be developed and checked out independent of the operating system; and 4) architectural upgrades can be more easily developed and implemented in the executive system because sublevel procedures and sequences are stable and proven at a well-defined hierarchical level. A detailed description of the microprocessor implementation and control software can be found in Ref. 6.

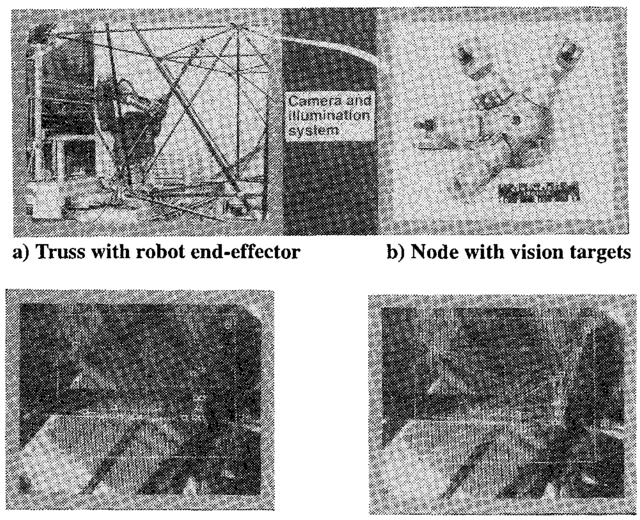
Machine Vision

The initial truss assembly tests, as indicated previously, relied totally on the use of "taught" robot positions supplemented by force–torque realignment for strut installation. The reliability required in locating the actual installation position makes this procedure unacceptable for in-space operations. Therefore, a special machine vision system was developed and implemented to guide the robot for strut installation and removal from a location about 38 cm from the grappling point. The operation of the machine vision system is illustrated in Fig. 5 and is described fully in Ref. 7. The system's main hardware components consist of a miniature charge-coupled device (CCD) video camera and target illumination sources mounted on each end of the strut end effector as shown in Fig. 5a. These had to be small and of low mass to fit on the already crowded end effector. Since the camera and lights were retrofitted to the end effector, they were packaged in a module and had to be located about 5.8 cm off the end effector centerline. Illumination of the target was necessary because sunlight through the laboratory windows creates numerous sources of shadowing, glare, and backlighting. The other vision components include a video image processor, a target identification algorithm, positioning and ranging software, a software guidance algorithm, and a passive target mounted adjacent to the receptacle of each connecting truss joint. The target consists of five retroreflective dots that are arranged in a distinctive pattern to aid in the identification process (Fig. 5b). The five retroreflective dots are made from tape that has high reflectivity when the incident illumination is aligned with the viewing sensor. Therefore, the illumination system includes a beamsplitter and light source to provide incident rays aligned with the camera optical axis. The size of the target envelope had to be small enough to fit on the strut connector receptacle without interfering with the previously developed and tested end effector receptacle grippers.

The system operates by taking a video image frame from the camera and storing it in a frame buffer in gray-level digital format. A technique based on histogram information is used to establish a gray-level threshold, and the video image is converted into a binary pixel array. This binary array is analyzed for contiguous pixel units (blobs), and the blobs are first evaluated for their potential as the target dots according to their size and shape. For those that pass this test, the centroid of each blob is determined (Fig. 5c). The centroids are then triangulated in every possible combination (Fig. 5d), and the geometric characteristics of the triangles are compared with the known configuration of the four triangles formed by the centroids of the target dots. When a single set of five blobs is identified that match the target in size and shape, and the centroids also match all of the geometric aspects, it is highlighted on the operator's monitor, and the five centroids are sent to a pose-estimation routine. The pose-estimation routine computes the target position in Cartesian space relative to the video camera and thus the robot end effector.

Three conditions may occur that can cause the system to fail to match the target blobs: 1) the target blobs may not be within the gray-level threshold, 2) the light intensity may not properly illuminate the target, and 3) the target may not be within the camera's field of view. To accommodate these cases the threshold level is first reset and the image is reinterrogated. If a match is still unsuccessful, the light intensity level is adjusted and gray-level thresholding is again performed. If both of these fail, an algorithm is initiated to incrementally change the position of the arm. If a match still fails, control is turned over to the operator, and he can incrementally adjust the arm and/or select the target. The machine vision system permits the operator to view the target and the processing operations simultaneously. The operator can easily verify that the selected target is on the desired installation receptacle or the location of the target if the system has relinquished control.

As indicated, the machine vision operations are initiated at a position about 38 cm from the receptacle grappling position. After the target has been identified and the location with respect to the camera calculated, the robot arm is commanded to move the end effector to a new position that both is nearer to the target and aligns the optical axis of the camera with the target center. At this location another target identification and pose estimation is performed. Three to four successive steps are generally commanded to bring the end effector to within 10 cm of the target. At this location the robot is commanded to move to the side to offset the displacement of the camera from the centerline of the end effector and then to move directly to the strut receptacle. During the phase II assembly tests the vision system was used for the installation of every strut.



a) Truss with robot end-effector **b) Node with vision targets**

c) Video image with centroids of candidate blobs **d) Video image with centroids triangulated**

Fig. 5 Photographs illustrating the operation of the machine vision system.

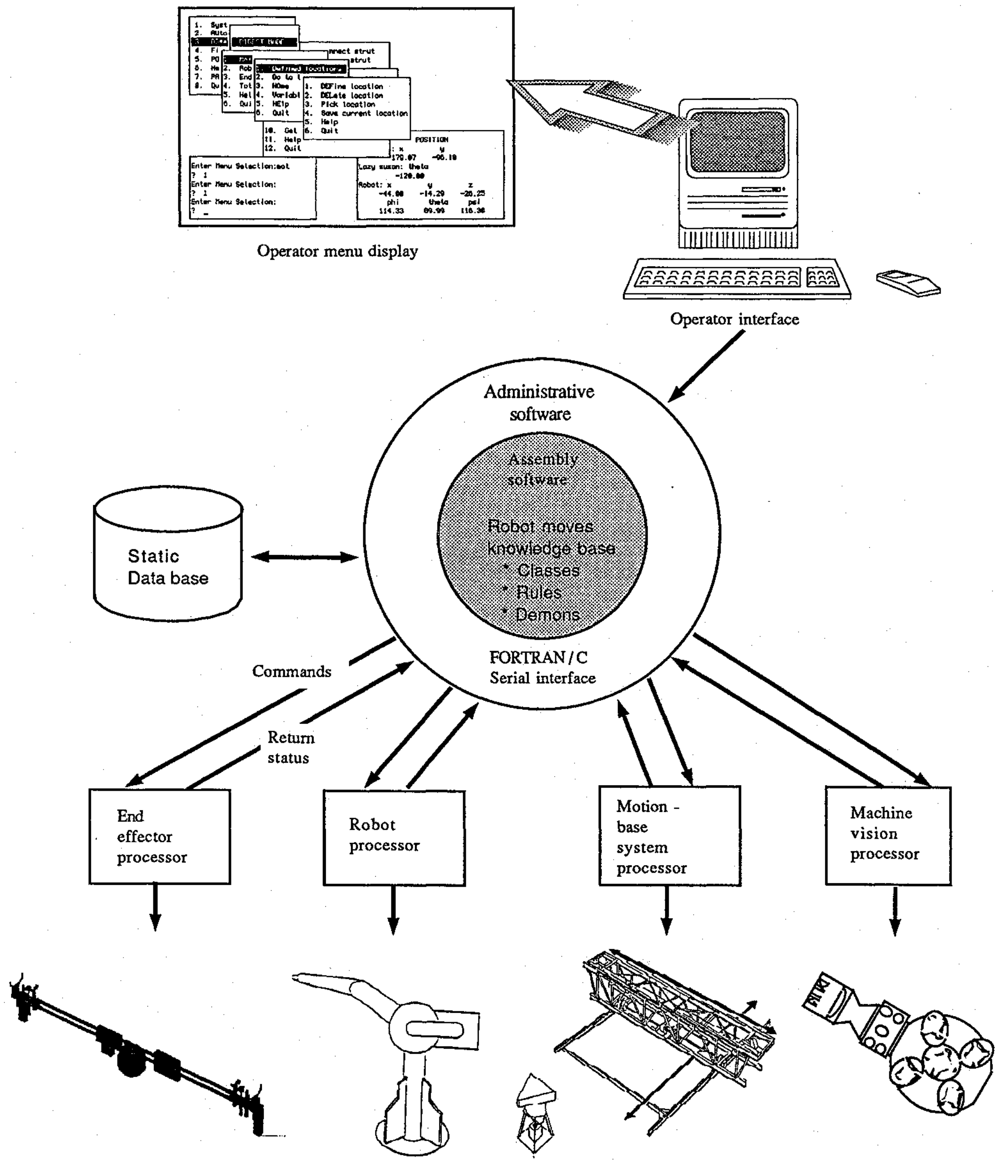


Fig. 6 Schematic of computer control system implemented for phase II tests.

Executive Control System

The volume of the knowledge to be managed by the software increased significantly with the addition of panel installation, the development of error recovery procedures, the addition of machine vision, and microprocessor end effector control. Keeping the control program current during the development of system upgrades became quite cumbersome. Therefore, the decision-intensive assembly executive was replaced with an expert-system-based program. An expert-system program is well suited for the current application because the operations are driven by the pursuit of a goal and fall naturally into if–then-type branching-logic decisions. By determining implicit subgoals and using backward-chaining inferencing, the system is driven toward the goal of total truss assembly. In addition, event-driven (forward chaining) inferencing is also initiated on the occurrence of an event rather than the achievement of a goal or subgoal. An expert-system shell that provided a convenient embedding technique for integrating the existing operator interface was selected for this task. It also allowed procedural code to send, receive, and modify database information through the use of special data types and run-time functions.

The upgraded control-system architecture is shown in Fig. 6. As indicated in the figure, the administrative-level and individual processor control software surrounds the expert-system knowledge base and was left intact to provide an interface function. When information regarding the status of a component is needed, a sensor associated with that component is polled through the device interface and the information is sent to the knowledge base. When a device-specific processor, such as the end effector processor (Fig. 6), has completed a series of commands, a return status is forwarded to the knowledge base, signaling that a subgoal has been achieved. If an error occurs, a recovery instruction to return the system to the last successful state may be sent. Information about all system functions is constantly updated and reported to the operator via status windows. Details of the expert-system control program can be found in Ref. 8. The software design and information on the operator-interface program can be found in Ref. 9.

The concise representation afforded by the rule-based expert system significantly reduced the amount and complexity of traditional procedural code required to command operations. For example, the phase I assembly tests required approximately 850 lines of code to move the robot through the cycle from the pickup point at the strut canister to the installation in the structure and return. When the control program was updated and this same sequence was implemented using the expert-system control code, only 22 rules were required. The total knowledge base for the integrated truss and panel assembly currently contains 59 rules: 22 for strut installation, 22 for panel installation, and 15 for transfer of strut pallets. The executive code for panel installation and removal was written and verified directly using the expert system. The use of the rule-based system has led to an increase in control-system maintainability, and modifications can be performed rapidly.

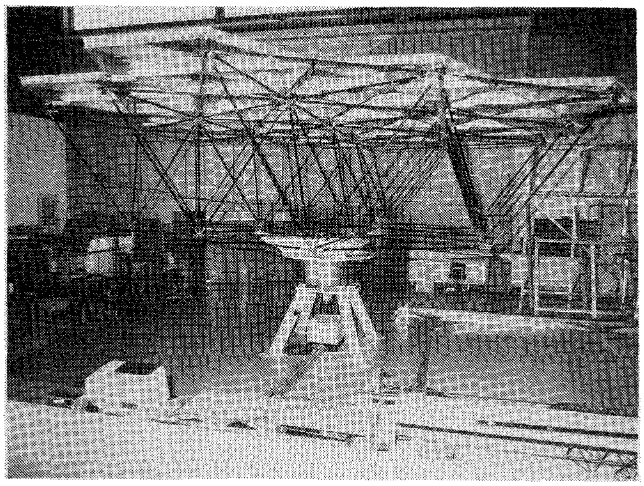


Fig. 7 Photograph of the truss with panels attached to the top surface nodes.

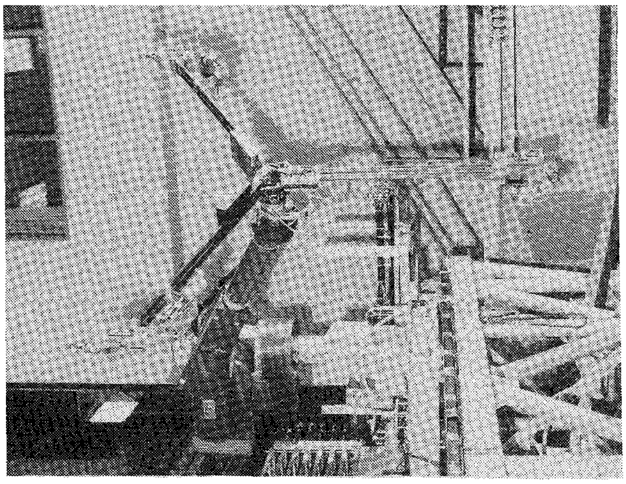


Fig. 8 Photograph of the end-effector for panel installation.

Panel Assembly Operations

A system of reflector-type panels was designed to expand the hardware and functional assembly capabilities of the system. A photograph of the truss and attached panels is shown in Fig. 7. Twelve hexagonal panels, each 2 m wide, are attached to the top nodes of the truss. The panels are fabricated as frames from six radial aluminum ribs connected at the perimeter by six straight members that form a hexagonal perimeter band. An aluminized plastic film is bonded to the top surface of the frame. The total mass of a panel is approximately 6.3 Kg. The panels are attached to the truss at three nodes, and a node must serve as an attachment location for as many as three panels. The size of the panels and the requirement for positioning and alignment at three points, as opposed to the two points required for the struts, represent a more stringent robot positioning task.

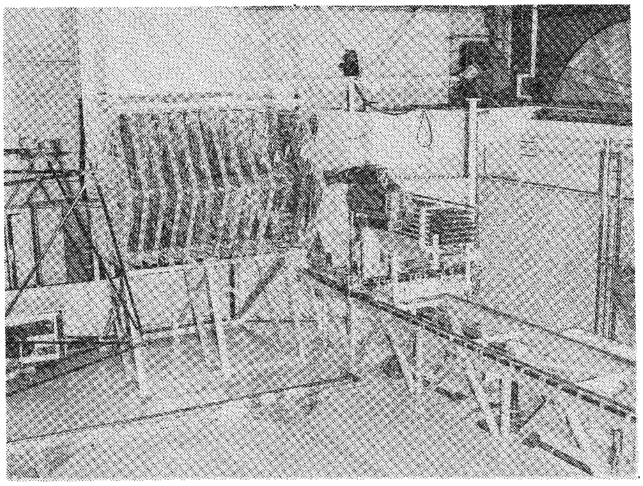
The end effector developed for panel installation is shown attached to the wrist of the robot arm in the photograph of Fig. 8. The end effector is a Y-shaped configuration with all of the actuators and sensors located at the ends of the arms. The two tubular elements that form each arm of the Y are graphite–epoxy to provide high stiffness and minimize the total mass of the end effector. The electronic components, including the microprocessor, power distribution system, and sensor interface boards, are all attached to the graphite–epoxy tubes near the robot wrist.

To preclude having to modify the strut end effector or the installation procedures for strut assembly that were well developed and proven by the phase I tests, a requirement was imposed that panel installation must be compatible within established strut installation and removal operations. It was also desirable to minimize the modifications to the existing truss nodes to add the panel attachment guides and latches. Therefore, the design of the panel attachments, the panel end effector design, and the development of panel installation operations was a challenging assignment. The panels are totally passive, i.e., all actuators required for latching and attachment are located on the end effector. The panels are stored in a canister adjacent to the Cartesian motion-base system, where they are fully accessible to the robot. All installation and removal operations are fully automated, and there is no manual assistance. Therefore, the same latching mechanisms used to attach the panels to the truss nodes are also used to attach them to the storage canister. However, the procedures for installing and removing a panel from the canister are significantly different from those for installing and removing it from the truss. This difference is associated with the way the panels are stored in the canister and the overall canister packing efficiency. The panels are stored close together in the canister to obtain a high packing efficiency and consequently must be removed in sequential order.

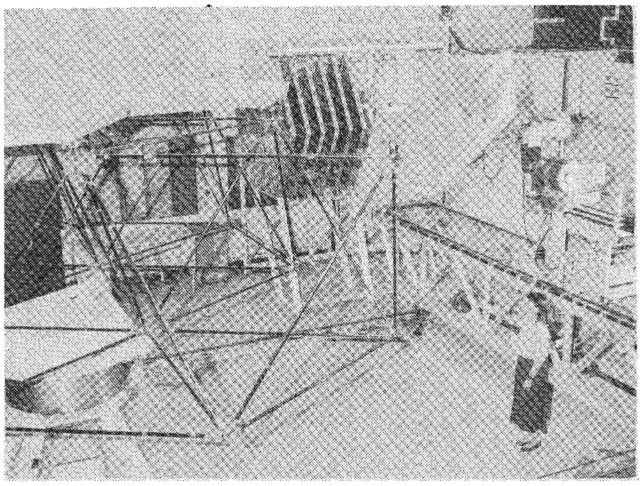
The removal of a panel from the canister and installation on the truss is illustrated by the photographs in Fig. 9. To fetch a panel, the motion base and robot arm position the end effector at the panel pickup point within the canister. As the panel is unlatched from the canister, it is automatically latched to the end effector (the reverse process occurs during installation on the truss). Following attachment to the end effector, the robot moves the panel toward its neighbor to provide clearance between the panel latches and the canister attachment fittings. (A more efficient packaging might have been achieved if this maneuver could have been eliminated.) Then the panel is removed by moving the robot's motion base away from the storage canister (Fig. 9a). After removal the robot turns about the waist axis (Fig. 9b) and raises the panel end effector to a position above the top of the truss, rotating the panel from a vertical to a near-horizontal orientation. The panel is then elevated 10 cm above the end effector by actuated platforms located on the end effector. This provides clearance between the current panel and other panels that may have been previously installed when the current panel is moved to the installation position. To provide access for the panel, the strut immediately in front of the truss cell where the panel is to be located is not installed until after the panel. Therefore, the panel must not block the strut or interfere with the strut installation procedure. The Cartesian motion bases move the robot base to the installation location (Fig. 9c) with the end effector approximately 10 cm in front of the panel installation position on the truss. The end effector is moved forward by the robot to the approximate panel installation position, and a set of fingers on the end-effector grapple the two joint receptacles for the strut immediately in front of the panel cell. The fingers and a truss joint receptacle are illustrated in Fig. 10, which shows the truss hardware and the components of the panel end-effector in the vicinity of the truss node. In the photograph on the left the fingers are in the open position, and in the one on the right they are in the closed position over the V-groove notch on the strut receptacle. This is basically the same grappling arrangement used for strut installation described previously.

Final positioning of the end effector is accomplished by the force–torque control algorithm with the assistance of the two V grooves to provide passive guidance for the correction of small misalignments. After realignment, the position of the end effector, and thus the panel, is precisely established with respect to the truss except for the pitch angle, which can be changed because the end effector fingers can be rotated around the receptacle V grooves. The panel is then lowered into position, the pitch angle is adjusted by force–torque-controlled repositioning, and the panel is latched to the truss. Using the truss as a positioning tool simplifies the installation process and eliminates the need for additional tools or sensors. The same path and installation sequence are used all panels; only the motion-base positions are different. Consequently this aspect of panel installation is easier than truss assembly, which involved 19 independent paths. After the panel is installed, the end effector is pitched down by rotating around the strut receptacles. This permits the end effector to clear the panel latches as it is backed away from the truss.

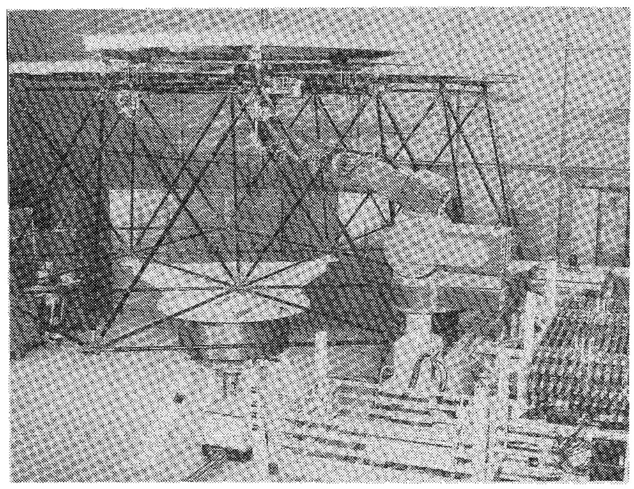
The panels designed for this experiment have clearances of 0.3–0.6 cm between them when they are latched to the truss. The posi-



a) Removal of panel from the canister



b) Panel being rotated and elevated to its installation orientation



c) End effector in position to install the panel on the truss

Fig. 9 Photographs illustrating removal of a panel from the supply canister and installation on the truss.

tioning technique worked well, and it is anticipated that even closer spacing between panels could be achieved by incorporating some refinements into the hardware design. The latch mechanisms were configured so that either the panel or the truss could expand and contract uniformly without introducing load into the other. The latches that hold the panels in position are spring-loaded sliding pins, and the latches are actuated by pneumatic cylinders on the end effector. Neither the spring-loaded pins nor the pneumatic actuators would be suitable for a space system; however, the concept would be suitable if the spring-loaded pins were replaced by threaded pins driven by gearhead drive motors mounted on the end effector. The drive motors would be required to have a spur gear that interfaces with a mating gear on the panel latch.

Combining the operations of strut and panel assembly made sequence development a challenging task. Since the installation of the panels involved a different end effector, switching between strut installation and panel installation requires end effectors to be changed. Also, since the truss had to be in place to position and support the panels, as much of the truss as possible was assembled without blocking the installation path to the panel. The assembly sequence, detailed in Ref. 10, involves installation of approximately 60% of the struts before any panels are attached. After the struts are in place, the end effectors are exchanged, and six of the twelve panels are installed. The end effectors are again exchanged and all remaining struts except those directly in front of the final six panels are installed. The final six panels are installed, followed by the remaining six struts. All operations, including end-effector exchange, are fully automated.

Summary of Phase II Assembly Results

Two complete assembly and disassembly tests were conducted to evaluate both the system upgrades for strut assembly operations and the newly developed panel assembly procedures. Recall that the system modifications for strut assembly included vision guidance, microprocessor-controlled end-effector operation, and an expert-system-based executive control program. The test results indicate that the average time to install struts increased 3.7 min. The installation time for the baseline tests of phase I averaged 9.2 min per strut, and the same operation in the phase II tests required about 12.9 min. Examination of these times indicates that the majority of the increase resulted from searches and incremental positioning required for the machine vision system. The machine vision also increased the strut removal time during disassembly, but only by about 1.4 min per strut, because disassembly requires fewer maneuver and search operations. The microprocessor controller on the end effector reduced the installation time; however, the reduction was too small to have a significant effect on the reported average.

The most significant difference in the results of the two test phases is the number of errors that required operator intervention. In the phase I tests the operator had to resolve a total of 193 errors, most of which were associated with end effector misalignments. In the phase II tests the operator was required to modify the sequence at the console only 80 times for the same number of struts installed and removed. Nearly one-fourth of these interventions were to assist the vision system in acquiring the target. These can be reduced by changing the focal length of the vision camera lens. The remaining errors resulted primarily from misalignments that occurred at the canister and are associated with "taught" robot positions. It should be noted that the vision system was totally effective in correctly discriminating the targets in a very cluttered environment without special background lighting. The expert-system control program and the end effector microprocessors were totally effective in directing the assembly and disassembly operations for both the struts and panels. All end effector exchange operations were performed as planned. The operator gained confidence with all of the control-system upgrades, never encountering an error to which the expert system failed to respond properly, and the control system never initiated a catastrophic condition. Also, the expert system was not an impediment to the operator during the resolution of errors.

The mechanical operations and robot paths associated with panel installation were developed around brute-force automation techniques similar to those used for the phase I tests. The installation of each panel required about 16.5 min. The installation and removal procedures involved a significant number of force–torque-controlled repositioning cycles that are very time-consuming. The time to implement each repositioning cycle might be reduced appreciably if it were performed inside the robot control loop. The number of force–torque cycles required might also be reduced and the installation reliability improved by incorporating vision

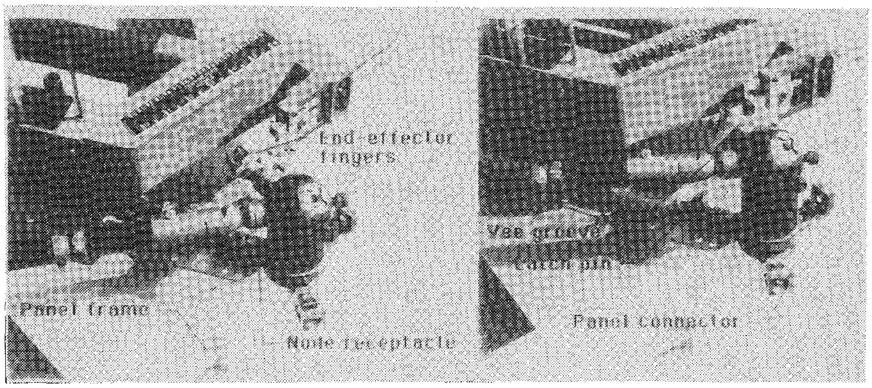


Fig. 10 Photographs that illustrate grappling of a strut receptacle by the fingers on the panel end effector.

guidance. The results of these tests are being evaluated; however, the operator encountered no serious difficulties with panel installation.

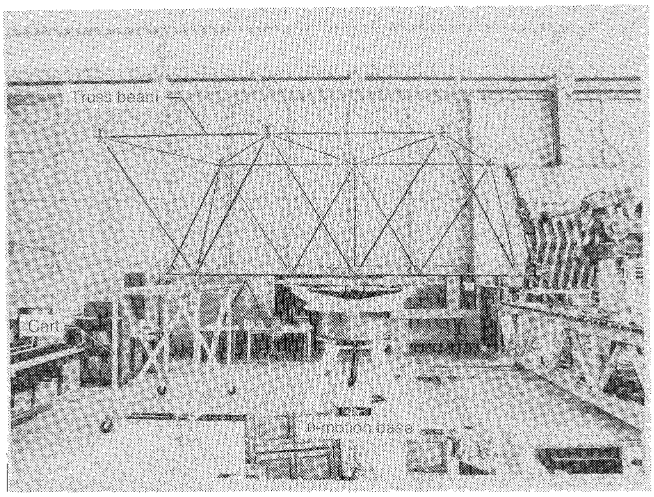
Phase III—Automated Planning and Truss Beam Assembly

The success of the tests performed in phases I and II has been very encouraging. The results have highlighted the need for both an expanded operational capability and automated planning features. The expanded operation was developed around the assembly of a linear truss beam, and the planning features involve both an assembly sequence planner and a robot path planner. These activities encompass the third program phase and are outlined in the following subsections.

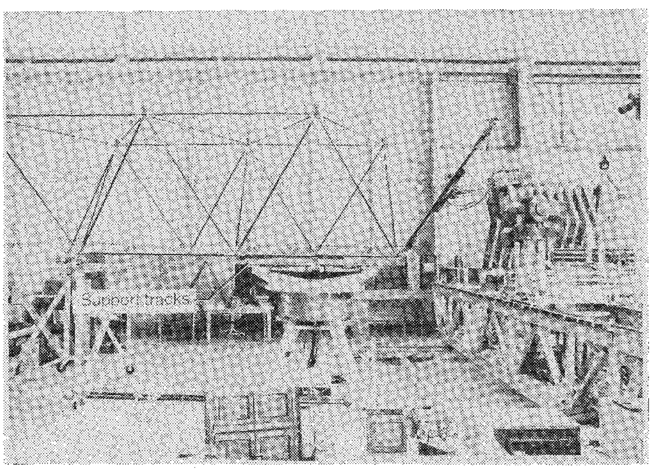
Truss Beam Assembly

Studies of concepts for deployable and erectable truss beams have been conducted for many years, and these studies are still of current interest. A telerobotically erected beam, however, has many potential advantages over most deployable beams, because each deployable beam concept typically requires a complex and unique deployer design. Also, each individual beam deployed in space requires a dedicated deployer. With the current concept, many beams with different structural configurations could be assembled. These beams would not require the use of revolute joints in the longeron load path that degrade structural performance[11] and are common to all deployable beams. The assembly study being performed in this phase presents an opportunity to demonstrate the versatility of the techniques developed in phases I and II. In addition, the current phase will permit additional upgrades of the control system, including additional automated error recovery techniques to be implemented and evaluated against the baseline results while performing a new assembly task, as opposed to an evaluation scheme that relies on repeating the same assembly test time after time.

A photograph of a portion of the beam and the assembly support system is shown in Fig. 11. The strut members used for the tetrahedral truss are used in the beam assembly, and many of the operations are similar. In fact, the beam is simply a structural subsection of the planar truss. However, for beam assembly the rotating motion base remains stationary and is modified with tracks for sliding the beam. The beam is assembled in 1-m-long sections in the region between the rotating motion base and the robot motion base. After a half-bay section has been assembled as shown in Fig. 11a, one member is grasped and the beam is pushed forward through a combination of robot arm and carriage moves. Pins through the side of the tracks into feet on the base nodes prevent the beam from being pushed forward inadvertently during assembly. The width of the track (the spacing between the tracks) can be adjusted using the robot to accommodate similar beams of different dimensions or beams of a different truss configuration. During assembly the beam is supported by three nodes on the track. While the beam is being advanced, additional support is provided by the end effector and robot. The cart at the left of the photo provides support for the 1 *g* gravity load as the length is extended.



a) Assembly of beam prior to being pushed forward in tracks attached to the rotating motion base



b) Continuation of assembly after the beam has been pushed forward to the next assembly station

Fig. 11 Assembly of truss beam.

Automated Sequence Planning

The generation of a strut sequences for assembly of the 102-member planar truss described in phase I and II is time-consuming and error-prone. Both sequences were developed manually, and the development was difficult because there is no unique order that the strut installation must follow. There are frequently three to five strut candidates at each step in the sequence, any one of which could be selected. However, the final configuration is fixed, and until the process is complete the developer cannot be assured that the sequence being pursued is viable, i.e., one that will lead to the final configuration. Many times a plan would be nearly complete when it was found that a path was blocked or that no strut with the proper assembly attributes was available in the supply canister. The developer would then have to go back to some intermediate point in the sequence, select an alternate strut, and resequence forward from that point. Also, it is anticipated that during an in-space assembly the operator may encounter a strut that cannot be installed on account of a mechanical malfunction. The truss is structurally redundant, therefore, an operationally stable system could still be achieved; however, the sequence would likely have to be replanned, and manual replanning at this point could not be performed in an expeditious manner. To rectify the problems associated with manual assembly planning, the development of a computer tool was pursued to automatically synthesize viable assembly sequences on line.

The sequence planner was developed using a knowledge-based expert-system shell program. An expert-system program is well suited for this application because the operations, like those of the executive control program, are driven by the pursuit of a goal and fall into branching-logic decisions. A commercially available microcomputer-based expert-system shell program[12] was selected.

The sequence planner initially stores all the struts and nodes in database files called "struts to assemble" and "nodes to assemble." There is no particular order to their arrangement in the database, and the attributes of each component (e.g., the strut name and installation path) are included in the data file. A series of candidate struts for installation are selected according to the geometric truss constraints and strut connectivity. All candidate struts are then examined relative to a set of rules that apply restrictive local conditions as selection criteria, and a strut is selected. The selected strut is then moved to another database called "struts assembled," the rules are reset, and the process is repeated. The selection criteria are governed by strut-by-strut connectivity considerations as opposed to using geometric subelements such as tetrahedrons or pentahedrons. However, an examination of several developed sequences indicate that the rules naturally produce this condition. To handle the situation where it is discovered late in the sequence that a strut is blocked (the condition that is so tedious for manually developed sequences), a backtracking process is implemented. The planner is reset to where the blocking strut was installed, the blocking strut is replaced by an alternate strut selection, and the sequence is replanned from that point. This may occur several times until a suitable sequence is developed. The sequence planner is fully automated and not interactive, i.e., an operator does not have the responsibility for choosing a strut from a supplied list of potential candidates.

The sequence planner is operational and a number of strut assembly cases have been developed and analyzed. The planner has been successful in demonstrating the following: 1) the development of sequences to evaluate the results of different rules, 2) examining sequences for disassembling the truss to replace a strut that was arbitrarily selected to simulate a failure condition, and 3) to generate alternative sequences for the initial test configuration with a goal of reducing the total assembly time. The planner has also been used to generate sequences for different structural configurations including the truss beam. The evolution of the operational version of the planner required several iterations, because the "rules" applied to generate the manual sequences were difficult to formulate and express to the developer of the sequence planner. The task was made possible by having the developer of the automated sequence planner and the developer of the manual sequences get together from day to day to review automatically developed sequences and formalize the assembly "rules."

Automated Path Planning

The paths for the installation of truss struts for the phase II tests were divided into two segments. The first segment uses "taught" points to guide the robot from the strut canister to a location that is aligned with the strut installation position but is approximately 38 cm from it. The second segment involved on-line guidance using machine vision to compensate for uncertainties in truss location, providing the reliability required for space operations. The first segment of the path was developed manually for both the phase I and the phase II tests. The structural configuration and restricted camera views make the development of this portion of the path by teleoperation virtually impossible. Therefore, we began development of an automated path planner using a potential-field algorithm with a number of search strategies to generate a collision-free path.

The potential-field algorithm being developed is based on the work of Barraquand et al.[13] Each assembly component and the end effector are surrounded by an electromagnetic potential field, and the end effector is attracted to the goal position. Similar approaches have been under development for some time, and they frequently fail because the search algorithm becomes trapped at a local minimum. This condition is avoided in the current path-planning algorithm by the use of both a random motion routine to escape local minima and a strategy that can plan the path in reverse, i.e., from the most congested area of the path toward a relatively uncongested area. The planning process either terminates successfully when the potential is close to zero or with a failure after searching for a preset time. After a successful termination, a path-smoothing algorithm is employed to connect successive points along the path.

A number of factors associated specifically with structural assembly make path planning difficult. Among these are: 1) the limited reach of the robot; 2) restricted robot dexterity associated with the six degrees of freedom, the limitations of the robot's joint axes, and the long forearm of the test robot; 3) the length of the end effector, which is required to be placed in restricted locations within the structure; 4) the clutter associated with the concentration of mechanisms and sensors on the end effector; and 5) the limited clearances between the end effector and the structure when the end effector is in the strut installation position. Even with those difficulties, the automated path planner has been very successful and has greatly simplified the path development process.

The path-planning algorithm is being evaluated on a computer-generated solid-model simulation of the automated assembly system that is hosted on a graphics workstation. A detailed graphics model of each component in the system has been developed and is used to permit monitoring the path-planning process. A typical computer-graphics simulation screen is shown in Fig. 12. The detailed model of the end effector is replaced by an oversized box for efficiency in operation and to create a buffer zone when performing path-planning studies. The path being planned may be observed in real time by the simulation operator from virtually any location in the work cell. This visualization capability is not available to the test-facility operator; consequently, a high level of confidence in the planning and simulation capability must be established before the hardware test operator will be comfortable with the autonomously planned paths. Following the development of a path, a simulation is typically performed with the detailed model of the end effector to verify that the planned path is collision-free. A customized collision detection mode that highlights a component during a collision may be invoked to give the simulation operator a visual indica-

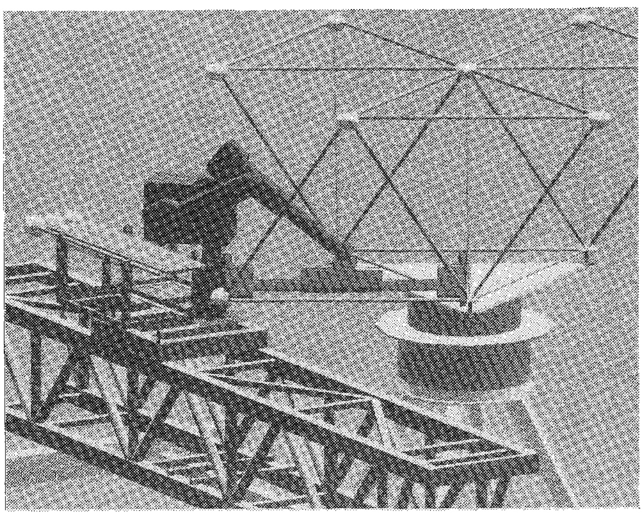


Fig. 12 Solid graphics model of the end-effector box, used for automated path planning.

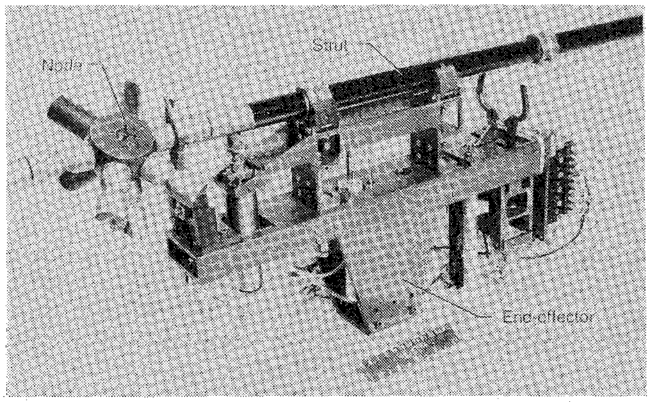


Fig. 13 End effector that picks up a strut and installs it independently at each end.

tion of interference. All of the paths for the beam assembly test were generated by the path planner and verified using the graphics solid model. The time typically required to generate a moderately complex path is about 10 min. This is significantly faster than even the simplest paths developed manually; therefore, a significant amount of time was saved in developing the paths for the beam assembly. The path planner has also been used successfully to verify assembly sequences generated by the automated sequence planner, confirming that collision-free paths are possible for each strut in the sequence. This is the most practical method of proving that the output of the sequence planner is valid. Currently the automated path planner utilizes only the six degrees of freedom of the robot manipulator, and the simulation operator selects the robot base and rotating motion-base positions. However, work is underway to expand the capabilities of the planner to nine degrees of freedom, which will eliminate the need for operator prepositioning. The capability to enable the operator to interactively define special points along the path is also being developed to reduce the search time for particularly complex paths.

Proposed Developments

The basic principle followed in the technology developments pursued throughout this program has been to develop hardware components and operations around the requirements of the task to be performed. Automation was implemented first by brute force, followed by the introduction of advanced procedures that include sophisticated guidance and planning techniques. This section covers several efforts that are anticipated or are currently underway to provide additional capability and robustness to the overall assembly system.

Accommodation of different assembly tasks, configurations, and end effectors has resulted in a very modular system in terms of computational architecture and software. The addition of distributed microprocessors to control end effectors significantly reduced the level of communication between processors and allowed the standardization of assembly commands by the executive. However, additional control-system improvements are necessary for space applications. To further minimize communications and increase the operational speed and performance, a true network architecture is being developed with a commercially available real-time distributed system based on ethernet communication. The executive scheduler for this system will also utilize a commercially available network-based expert-system shell. This should result in a fully distributed system that supports concurrent operations as required to achieve the projected strut assembly time of 4–5 min.

Another system advancement involves the capability to assemble truss structures that have members of different lengths such as those that support curved panels for antennas and aerobrakes. This will require either an end effector with a prismatic joint that permits a variation in the distance between the ends, or an end effector that picks up a strut on one end and, after installing that end, captures the other end, which is cantilevered, and installs that end also. This type of end effector could also be used to perform other functions, such as installing system payloads and control devices. An end effector with this capability has been designed and fabricated, and a photograph is shown in Fig. 13. Many of the functions and supporting software have been developed; however, the functionality remains to be demonstrated in hardware assembly studies. Until recently testing has been hindered by the lack of a path planner, because an additional path segment from one end of the strut to the other, as well as a new retreat path, is required. Another problem that must be addressed with this end effector is the implementation of the machine vision in locating the free end of a cantilevered strut, as well as the free end of a second cantilevered strut to which the first strut must be attached. These procedures will be developed and tested in a static test cell before they are attempted in the larger test facility.

Summary

During the past several years the Langley Research Center has conducted a development program to evaluate hardware concepts and software requirements, as well as operator interface and control techniques, for on-orbit automated assembly of truss structures using robotic manipulators. An experimental test facility was developed so that realistic hardware problems and constraints that are likely to be encountered could be identified and resolved through research-level testing. One key to the progress and success of the development in this relatively complex technology was the use of a total system approach by an interdisciplinary team. The structure and mechanism designs were developed around a fairly well-defined set of mission requirements, and the robotic operations, sensing requirements, and operator interface were coordinated with the operations of the various hardware devices. A second key to program success was the development of the basic functionality in the initial stage, followed by the introduction of system refinements and upgrades in an orderly manner and assessment of the benefits against the initial baseline test results. This approach minimized the problems associated with the simultaneous introduction of several complex features, which can frequently confuse and obscure operational issues. Accommodation of different assembly tasks, control-system configurations, and end effectors has resulted in a very modular system in terms of computational architecture and software.

Several techniques have emerged as being critical to automated assembly. Successful joining operations were quickly found to require grappling of the workpiece to provide support and stability for mechanical operations. Final positioning using force–torque control with the aid of passive guidance features also proved to be essential. Teleoperation does not appear suited for truss assembly tasks, because of the broad video coverage required for the operator and the complex angular configuration and positioning requirements that occur within the confined work space. Also, the lack of depth perception is a significant impediment to teleoperation. Automated path planning coupled with a machine vision system to find and guide the robot to a target on the receptacle has been shown to enable reliable robot positioning.

The success achieved with the current laboratory hardware and industrial robot arm in the 1 g environment indicates that automated assembly operations with a precision in-space system are quite feasible within the current state of technology. Particularly encouraging was the operator's success in resolving problems. This demonstrates that 1) the problems are simple and easily corrected; 2) the limited error information provided to the operator was adequate for successful error resolution; 3) the expert-system control program contributed to the operator's confidence and success, because he never encountered an error that became more difficult or produced a catastrophic condition due to the error-recovery decisions of the expert system. Also, the operator was able to resolve nearly every error condition using the commands available at the console. In short, the level of implementation of the expert system in the control hierarchy prevented the embedded error recovery from becoming an impediment to the operator and was a key factor in the success of recent test studies.

Much has been accomplished in this program in a short time and with very limited resources. No problems have been encountered in tests conducted to date that would preclude automated in-space assembly of truss structures. However, some additional enabling technology much be developed and evaluated by tests similar to those conducted in the current investigation. The current hardware and parts of the control system are at the breadboard level, and upgrades of selected components to advanced automation technology are warranted.

References

[1]Heard, W. L., Jr., Watson, J. J., Ross, J. L., Spring, S. C., and Cleave, M. L., "Results of the ACCESS Space Construction Shuttle Flight Experiment," AIAA Paper 86-1186, June 1986.

[2]Heard, W. L., Jr., Bush, H. G., Watson, J. J., Spring, S. C., and Ross, J. L., "Astronaut/EVA Construction of Space Station," AIAA Paper 88-2454, April 1988.

[3]Heard, W. L., Jr., Lake, M. S., Bush, H. G., Jensen, J. K., Phelps, J. E., and Wallsom, R. E., "Extravehicular Activity Compatibility Evaluation of Developmental Hardware for Assembly and Repair of Precision Reflectors," NASA TP 3246, Sept. 1992.

[4]Rhodes, M. D., Will, R. W., and Wise, M. A., "A Telerobotic System for Automated Assembly of Large Space Structures," *The 21st Century in Space*, Advances in the Astronautical Sciences, Vol. 70, edited by G.V. Butler, 1990, pp. 111–129.

[5]Rhodes, M. D., and Will, R. W., "Automated Assembly of Large Space Structures," IAF Paper 90-272, Oct. 1990.

[6]Doggett, W. R., Rhodes, M. D., Wise, M. A., and Armistead, M. F., "A Smart end effector for Assembly of Space Truss Structures," *Proceedings of the Fifth Annual Space Operations, Applications, and Research Symposium*, July 1991.

[7]Sydow, D. P., and Cooper, E. G., "Development of a Machine Vision System for Automated Structural Assembly," NASA TM 4366, May 1992.

[8]Allen, C. L., "Automated Assembly of Large Space Structures Using an Expert System Executive," *Proceedings of the Sixth Annual Space Operations, Applications, and Research Symposium*, May 1992.

[9]Herstrom, C. L., Grantham, C., Allen, C. L., Doggett, W. R., and Will, R. W., "Software Design for Automated Assembly of Truss Structures," NASA TP 3198, June 1992.

[10]Kenner, W. S., Rhodes, M. D., and Fichter, W. B., "Component Count and Preliminary Assembly Considerations for Large Space Truss Structures," NASA TM 102604, Feb. 1990.

[11]Rodes, M. D., "Design Considerations for Joints in Deployable Space Truss Structures," *NASA/DOD Control/Structures Interaction Technology*, NASA CP 2447, Part 1, pp. 383–398, Nov. 1986.

[12]Anon., "NEXPERT OBJECT" *Reference Manual*, Version 2.0, Neuron Data, Palo Alto, CA, Jan. 1991.

[13]Barraquand, J., Langlois, B., and Latombe, J.-C., "Robot Motion Planner with Many Degrees of Freedom and Dynamic Constraints," *Proceedings of the Fifth International Symposium of Robotics*, 1989.

E. A. Thorton
Associate Editor