JOURNAL OF SPACECRAFT AND ROCKETS
Vol. 39, No. 2, March–April 2002

Hardware Experiments of a Truss Assembly by an Autonomous Space Learning Robot

Kei Senda,* Yoshisada Murotsu,† Akira Mitsuya,‡ Hirokazu Adachi,§
Shin'ichi Ito,§ Junya Shitakubo,§ and Tsutomu Matsumoto§
Osaka Prefecture University, Sakai, Osaka 599-8531, Japan

An experimental system simulating a free-flying space robot, which has been constructed to study autonomous space robots, is addressed. The experimental system consists of a space robot model, a frictionless table system, a computer system, and a vision sensor system. The robot model is composed of two manipulators and a satellite vehicle and can move freely on a two-dimensional planar table, without friction, using air bearings. The robot model has successfully performed the automatic truss structure assembly, including many tasks, for example, manipulator berthing, component manipulation, arm trajectory control collision avoidance, assembly using force control, etc. Moreover, even if the robot fails in a task planned in advance, the robot replans the task by using reinforcement learning and obtains the task goal. The experiment demonstrates the possibility of the autonomous construction and the usefulness of space robots.

Nomenclature

a	=	action for Q learning
$\boldsymbol{f}$	=	external force vector applied to the hand
$\boldsymbol{h}$	=	centrifugal and Coriolis forces
i_i	=	mass moment of inertia of link i about its mass center
$\boldsymbol{J}$	=	Jacobian matrix
$\boldsymbol{K}_D$	=	positive-definite derivative gain matrix
$\boldsymbol{K}_P$	=	positive-definite proportional gain matrix
ℓ_{gi}	=	distance of mass center of link i from joint i
$\boldsymbol{M}$	=	inertia matrix
Q	=	action-value function
$\boldsymbol{q}$	=	$[\theta_1, \ldots, \theta_n]^T$ joint angle vector
r	=	reward for Q learning
s	=	state for Q learning
t	=	time
U	=	conic artificial potential
$\boldsymbol{u}$	=	feedback force derived from artificial potential U
$\boldsymbol{y}$	=	manipulation variable vector at hand
α	=	learning rate for Q learning
γ	=	discount rate for Q learning
θ_i	=	vehicle attitude angle ($i = 0$) or joint angle ($i \neq 0$)
$\boldsymbol{\tau}$	=	joint input torque vector
$\phi(\boldsymbol{q})$	=	$\boldsymbol{0}$ or $\phi(\boldsymbol{y}) = \boldsymbol{0}$ constraint equations at hand
∇_q	=	$\partial/\partial \boldsymbol{q}^T$

Subscripts

d	=	desired value
e	=	$(\,) - (\,)_d$ for any variable

Superscript

$\cdot$	=	differentiation with respect to time

Presented as Paper 2000-4377 at the Guidance, Navigation, and Control Conference, Denver, CO, 14–17 August 2000; received 2 February 2001; revision received 5 July 2001; accepted for publication 17 July 2001.

*Associate Professor, Department of Aerospace Engineering. Senior Member AIAA.

†Professor Emeritus, Department of Aerospace Engineering; currently President, Osaka Prefectural College of Technology, Saiwai-cho, Neyagawa, Osaka 572-8572, Japan. Senior Member AIAA.

‡Research Associate, Department of Aerospace Engineering.

§Graduate Student, Department of Aerospace Engineering.

Introduction

SPACE robots are necessary for future space projects for construction, repair, and maintenance of satellites and space structures in orbits. Hence, it is important to develop a free-flying space robot consisting of manipulators and a satellite vehicle that can fly freely in an orbit (referred to here as a space robot). Many new complicated dynamic problems have been raised, for example, interaction between the manipulators and satellite, structural flexibility caused by lightweight requirements, etc. There exist many papers focused on the dynamic problems,[1–5] whereas the references cited here are not extensive. Some studies using hardware equipment on the ground to examine control and identification methods have been reported.[5–12]

Moreover, studies of autonomous systems (e.g., recognition using force and vision information, planning and reasoning, etc.) are necessary to realize the autonomous space robots that can achieve their mission as commanded by human operators.[13] The following projects emphasize this point: the Telerobotics Research Program,[11] the space robot technology experiment,[14] the Ranger Telerobotic Flight Experiment,[15] and the Engineering Test Satellite-VII.[16] As of this year, those projects are almost complete, but there remain many applications for autonomous space robots. There are many tasks autonomous space robots can accomplish, thus replacing human astronauts. For such autonomous robots, adaptation and learning in a real work environment are key issues. Therefore, testbeds are necessary for research and development.

To that end, this study has developed a ground experimental system and started researching robot autonomy, by simulating a free-flying space robot under microgravity conditions in orbit (Fig. 1). With this system, many control techniques are used to make the space robot model automatically assemble a truss structure. In the assembly demonstration, the robot model performs several tasks, for example, manipulator berthing, component manipulation, arm trajectory control collision avoidance, assembly using force control, etc. Repeating the sequence would enable construction of large structures.

However, the space robot may fail in a task planned in advance because of uncertainties and variations of the worksite. To obtain the task goal, the robot must suitably modify the task for the real work environment. For this purpose, the robot replans by using reinforcement learning with trial-and-error processes. The robot experimentally achieves the goal by carrying out the replanned task.

The rest of this paper is organized as follows. The experimental system is introduced in the next section. The third section explains fundamental control techniques realizing the assembly. In the fourth section, the autonomous truss structure assembly is experimentally demonstrated by synthesizing the techniques. The fifth section illustrates the method using reinforcement learning to plan

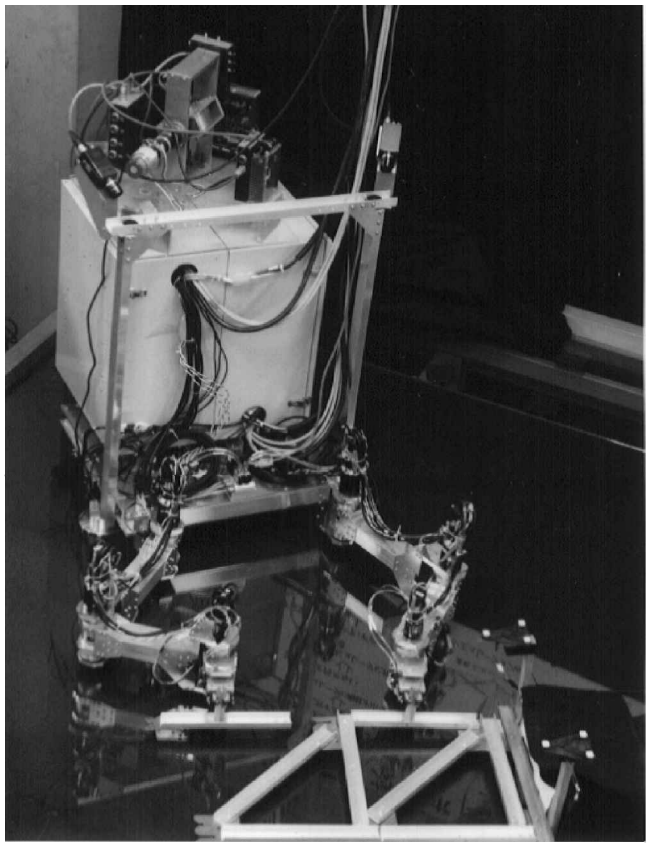

Fig. 1 Photograph of space robot model and truss.

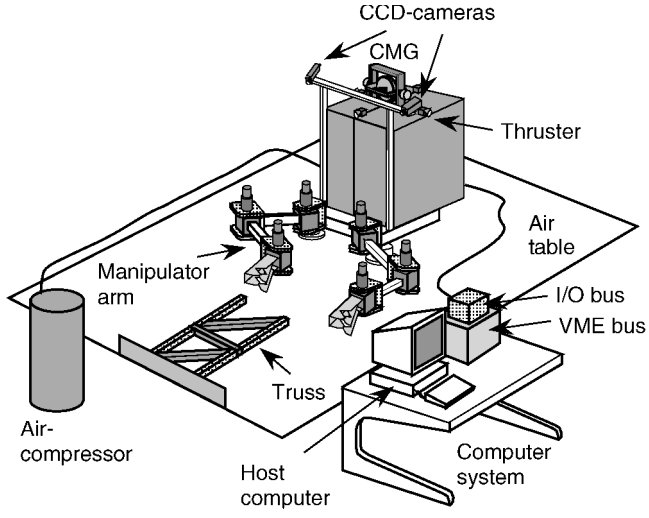

Fig. 2 Schematic of experimental system.

the task sequence appropriately for the real work environment when the robot fails in the task planned in advance. Some concluding remarks are given in the final section.

Experimental System

Outline of Experimental System

Figure 1 is a photograph of the space robot model and a truss structure under assembly. Figure 2 shows a schematic diagram of an experimental system constructed in this study. The experimental system simulates a free-flying space robot in orbit while motion of the robot model is restricted in a two-dimensional plane. The experimental system consists of four subsystems: 1) the space robot model, 2) the frictionless table system, 3) the computer and I/O system, and 4) the vision sensor system. The robot model is supported on the horizontal table without friction by the use of air pads.

Table 1 Specification of space robot model

Link	Mass m_i, kg	Length ℓ_i, m	Mass center ℓ_{gi}, m	Inertia i_i, kg · m^2
0	66.36	0.464	0.000	5.917
1	2.96	0.320	0.182	0.045
2	2.25	0.260	0.132	0.028
3	1.70	0.118	0.031	0.002
Payload	1.95	——	0.000	0.020

Specifications of Experimental System

Space Robot Model

The robot model consists of a satellite vehicle and dual three-degree-of-freedom (DOF) selective compliance assembly robot arm-type manipulators. A pair of charge-coupled device (CCD) cameras for stereo vision and a position/attitude control system are installed on the satellite vehicle. The position/attitude control system consists of four thrusters and a control momentum gyro.[17]

The parameters of the robot model are listed in Table 1, where ℓ_{gi} and i_i indicate the position of mass center and the mass moment of inertia of link i about its mass center, respectively. The links and joints are numbered from the base to the tip of the manipulators. Link 0 and 3 are the satellite vehicle and a manipulator hand, respectively. Both manipulators have the same specifications.

Each manipulator is composed of rigid links and a hand. The hand has one DOF of open/close states and can grasp a payload. All articular joints are driven by dc servomotors with harmonic drive reduction gears. The joint angle can be measured by the use of an optical rotary encoder mounted on each dc motor. The motor drive amplifiers directly control the amplitude of the current applied to the geared dc motors. The system uses a torque servoactuator[18] with a torque sensor and a servocontroller installed in every joint. The measured torque error is fed back to the servocontroller and a fine torque control of the output axis is achieved. The applied forces and torque at the end effector can be calculated from the measurements of the joint torque sensors.

Frictionless Table System

The frictionless table system realizes a two-dimensional weightless condition in the horizontal plane. The table system consists of three parts: the horizontal planar table, air pads, and a pressure air source. The top of the table is leveled carefully to realize a horizontal flat plane on the tabletop. Air pads are used to support the space robot model on the planar table without friction. The friction coefficients of the air pads are less than 1.0×10^{-4}. The maximum traverse speed of the pads is over 1 m/s.

Computer and I/O System

Figure 3 is a schematic diagram of the computer and I/O system. The system is composed of three subsystems: a host computer, VME bus, and I/O bus. The host computer is used for complicated offline processes and operator interface. The VME bus has a CPU board, an image processing board, and a digital signal processor (DSP) board. The real-time operating system working in the CPU board controls the image processing board and the DSP board. The CPU board processes many online operations, for example, information transmissions to the host computer and the DSP board, communications among the host and the DSPs, etc. The I/O bus has several boards for the real-time control and interface to the robot model. The DSP board in the I/O bus is connected with DSPs in the VME bus by the DSP links, and parallel processing capability is available.

Vision Sensor System

The two CCD cameras are used to capture a stereo image to measure the position and orientation of targets. The image processing board deals with primitive processes of the stereo image and forwards the results to the CPU board and a DSP board. The CPU and DSP boards compute the spatial positions.

The spatial positions of reference points in the work environment are calculated from the stereo image by using triangulation. Three light-emitting diodes on a target marker enable measurement of their positions and orientation simultaneously. The vision sensor system takes a stereo image every 33.3 ms. The total time delay

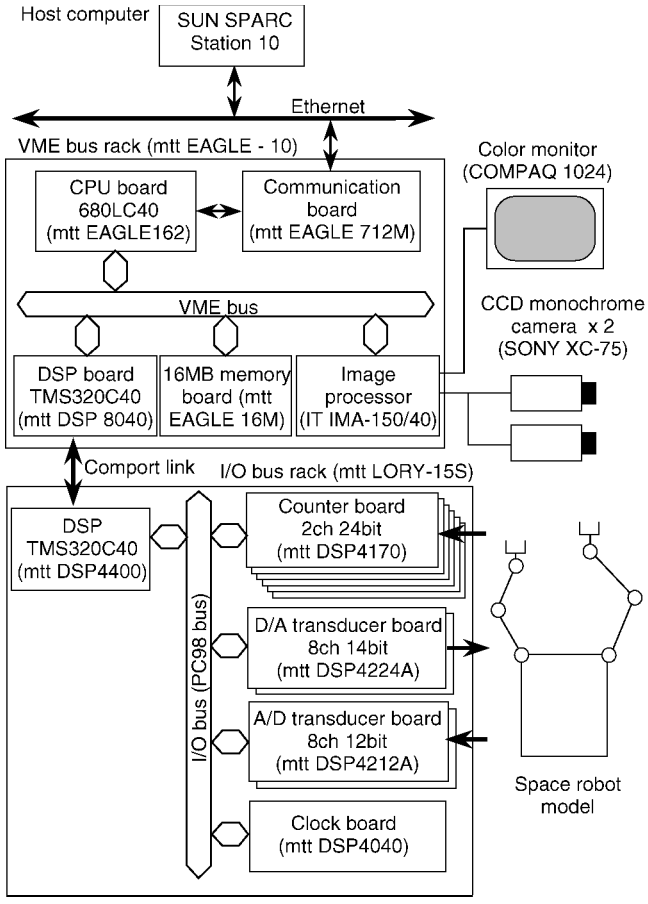

Fig. 3 Schematic of computer and I/O system.

of its measurement is 66.7 ms, including the computation of the position and orientation. After an appropriate hand–eye calibration, the stereo vision yields the spatial position measurement with a mean error of 2 mm. For the position and orientation of the robot, the vision system measures the target marker at the worksite, which is regarded as the inertial reference (see Fig. 1).

Assessment of Microgravity Condition

The robot model is subjected to some external forces caused by the electrical I/O cables and the air-supply tubes, inclination of the horizontal table, and other disturbances. The microgravity condition was assessed by translation acceleration of the mass center of the robot model during a motion of the manipulator.[10] The maximum translation acceleration of the mass center was approximately 2.0×10^{-4} g, and the average was below 1.0×10^{-4} g during the motion, where g is the unit of the gravitational acceleration. For reference, the average acceleration of an orbiting space shuttle is approximately 1.0×10^{-4} g.

Control of Space Robot

This section explains the fundamental control techniques for the space robot, for example, the visual servoing, the position and attitude control of the satellite vehicle, the positioning control of the free-floating space robot, path planning of arms for avoiding collision with the local work environment, force controls considering contact with the work environment, etc.

System Dynamics

The robot performs several tasks in the truss assembly. The robot moves a hand and captures a part of the worksite under a floating condition, that is, manipulator berthing. The robot holds on to the worksite and positions the main body by one arm. Using the other arm, the robot manipulates a component and assembles it, contacting with the work environment. During the assembly, the robot experiences three situations: 1) a floating situation without any constraints with the work environment, 2) a situation wherein the stabilizing arm is fixed to the environment and the other is free from the environment, and 3) a situation wherein the stabilizing arm is fixed to the environment and the other constrained by the environment.

For these situations, the following equations are obtained by the Lagrangian method. The equations of motion of the situation 1 can be derived as

$$\tilde{\boldsymbol{M}}(\tilde{\boldsymbol{q}})\ddot{\tilde{\boldsymbol{q}}} + \tilde{\boldsymbol{h}}(\tilde{\boldsymbol{q}}, \dot{\tilde{\boldsymbol{q}}}) = \tilde{\boldsymbol{\tau}} \tag{1}$$

where $\tilde{\boldsymbol{q}} = [\boldsymbol{r}_0^T, \theta_0^T, \boldsymbol{q}^T]^T$ is the generalized coordinate vector, $\tilde{\boldsymbol{\tau}} = [\boldsymbol{f}_0^T, \tau_0^T, \boldsymbol{\tau}^T]^T$ the generalized force vector corresponding to $\tilde{\boldsymbol{q}}$, $\tilde{\boldsymbol{M}}$ an inertia matrix, and $\tilde{\boldsymbol{h}}$ the centrifugal and Coriolis forces. Here, $\boldsymbol{r}_0$ and θ_0 are the position and the attitude angle of the satellite vehicle, respectively, and $\boldsymbol{q}$ is the joint angle vector of the manipulators. Their generalized forces $\boldsymbol{f}_0$ and τ_0 represent the translation forces and the rotation force applied to the satellite, respectively.

In situation 2, if the inertial coordinate frame is placed on the worksite, the equations of motion can be obtained as well as those for manipulators on the ground:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\tau} \tag{2}$$

In situation 3, constraint conditions due to contact with the work environment are inserted into Eq. (2):

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = [\nabla_q \phi(\boldsymbol{q})]^T [\boldsymbol{f}/\|\nabla_y \phi(\boldsymbol{y})\|] + \boldsymbol{\tau} \tag{3}$$

where the first term on the right-hand side is the reaction force transformed into the joint space.

Visual Servoing

The space robot must measure the relative position and orientation to the work environment or the manipulated objects by using a visual sensor. It must then control the manipulators because the position and attitude of the satellite vehicle changes during tasks. To achieve such control, a feedback control called visual servoing is often used by integrating the information from a vision sensor into the control loop. There exist two control systems, that is, the image-based and the position-based visual servoing systems.[19] This study uses the position-based visual servoing (PBVS). For the PBVS, the reference input to the servosystem is given by the relative position/orientation to the target. Various control methods for robotic arms can be used in the visual servoing systems. In this study, some control systems have been examined by replacing the arm controller. In each control system, the sampling time of the visual sensor is 33.3 ms and the control loop is 1 ms.

Position/Attitude Control of Satellite Vehicle

The space robot flies and positions the satellite vehicle near the worksite before beginning the assembling task. This maneuver is regarded as position/attitude control of a rigid satellite because the manipulators are fixed not to create a disturbance. The dynamics of the system are modeled in Eq. (1) constrained by $\ddot{\boldsymbol{q}} = \dot{\boldsymbol{q}} = \boldsymbol{0}$. For the system, a good position control result has been obtained by a thruster on/off controller that has generated a sliding manifold and that has been robustly stable.[20] Some good attitude controllers have been obtained where the installed control momentum gyro has changed the vehicle's attitude quickly and maintained stability.[17,20]

Positioning Control of Floating Robot

The vehicle position/attitude control system is off at the manipulator berthing to avoid affecting the manipulator control. In this situation, control methods for free-floating space robots should be used because manipulator motions vary the position/attitude of the satellite vehicle. The sensory feedback control for space robots[2] employed in this study is induced by the following conic artificial potential:

$$\boldsymbol{\tau} = -\boldsymbol{J}^T(\theta_0, \boldsymbol{q})\boldsymbol{u} - \boldsymbol{K}_D\dot{\boldsymbol{q}} \tag{4}$$

$$\boldsymbol{u} = -\frac{\partial U(\boldsymbol{y}_e)}{\partial \boldsymbol{y}_e} = \begin{cases} -(\alpha^2/\sigma)\boldsymbol{y}_e & \text{for} \quad \|\boldsymbol{y}_e\| \le \sigma/\alpha \\ -\alpha(\boldsymbol{y}_e/\|\boldsymbol{y}_e\|) & \text{for} \quad \|\boldsymbol{y}_e\| > \sigma/\alpha \end{cases} \tag{5}$$

The dynamics of the free-floating space robot is obtained by Eq. (1) when $\boldsymbol{f}_0 = \mathbf{0}$ and $\tau_0 = 0$ are taken into account, which is equivalent to the linear and angular momentum being conserved. For the system, Eq. (4) makes the joint angular velocities and the position/orientation error of the manipulator hands converge to zero, guaranteeing the asymptotic stability of a static target. Furthermore, the control scheme is compatible with the visual servoing because it enables the manipulation variable error, measured in the satellite frame, to be fed back.

A resolved acceleration control (RAC)[3] and a resolved motion rate control[1] have also been available for the free-floating space robot. Those controls and the sensory feedback were evaluated by hardware experiments.[10] In terms of control performance, the RAC and the sensory feedback were the best. In terms of computational effort, Eq. (4) was the minimum. Moreover, the sensory feedback is the most robustly stable. Therefore, this study uses the sensory feedback.

Path Planning and Trajectory Control of Arms

After the robot is berthed to the worksite, it manipulates the components to be assembled, plans the path of its arms to avoid collision with the work environment, and controls its arms to track the obtained path. This study plans a path to a goal of avoiding obstacles by an artificial potential method, where objects in the environment, for example, the truss under assembly, are regarded as obstacles. The method introduces an artificial potential field as a solution of the Laplacian differential equation, which has only one global minimum at the goal and does not have any other local minima in the state space.[21] It then plans the desired path to the goal by using a gradient method. The obtained path is transformed into a trajectory with respect to time.[22]

The manipulator is then controlled to track a planned joint trajectory by the servoloop of each joint. For a planned trajectory in the manipulation variables, this study usually uses the sensory feedback control equivalent to Eq. (4) for a base-fixed manipulator.[23]

Force Control Contacting with Work Environment

For the assembly, the space robot applies some force to the truss, such as pushing a truss component into a connector, and receives its reaction. Consequently, the space robot works on stabilizing the satellite vehicle with respect to the worksite. In this situation, the space robot is similar to the base-fixed robot on the ground. On the other hand, force control is needed for Eq. (3), considering constraints due to contact with the work environment. This study uses the following position/force hybrid control called saturated proportional and differential feedback (SP-DF) control.[24]

The control input $\boldsymbol{\tau}$ is given as

$$\boldsymbol{\tau} = -\boldsymbol{K}_D\dot{\boldsymbol{q}} - \boldsymbol{K}_P\boldsymbol{S}(\boldsymbol{q}_e) - (\nabla_q \phi(\boldsymbol{q}))^T [\, \boldsymbol{f}_d / \| \nabla_y \phi(\boldsymbol{y}) \|] \qquad (6)$$

where $\boldsymbol{f}_d$ is the desired external force vector applied to the hand, and

$$\boldsymbol{S}(\boldsymbol{q}_e) = \mathrm{diag}\,[\mathrm{Sin}(\theta_{1e}), \ldots, \mathrm{Sin}(\theta_{ne})] \qquad (7)$$

where $\mathrm{Sin}(\theta)$ denotes

$$\mathrm{Sin}(\theta) = \begin{cases} -1 & (\theta \le -\pi/2) \\ \sin(\theta) & (-\pi/2 < \theta < \pi/2) \\ 1 & (\pi/2 \le \theta) \end{cases} \qquad (8)$$

saturating at $\pi/2 \le \|\theta\|$. The derivative gain $\boldsymbol{K}_D$ and the proportional gain $\boldsymbol{K}_P$ are positive-definite and diagonal matrices. The control scheme guarantees that $\dot{\boldsymbol{q}}(t) \to \mathbf{0}$ and $\boldsymbol{q}(t) \to \boldsymbol{q}_d$ as $t \to \infty$ if $\boldsymbol{q}_d \equiv \mathrm{const}$, $\phi(\boldsymbol{q}_d) = \mathbf{0}$ holds, and the tip is constrained to move on $\phi(\boldsymbol{q}) = \mathbf{0}$.

Truss Assembly

Assembled Truss

Figure 4 illustrates a sequence of truss structure assembly. Manipulating a truss component and connecting it to a node precede the assembly process. The component is installed in the planned position and direction because the connector at the node has a notch to insert the component. Corners at the notch are planed off to ease insertion of the component. The installed component will not detach because the connector has a latch mechanism. The truss is designed to be robot friendly and can be assembled with one arm.

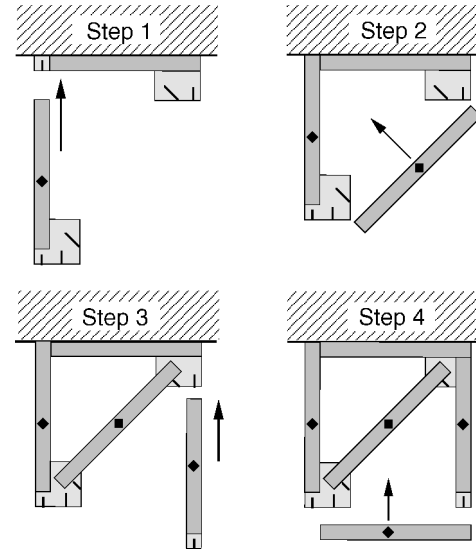

Fig. 4 Sequence of truss structure assembly.

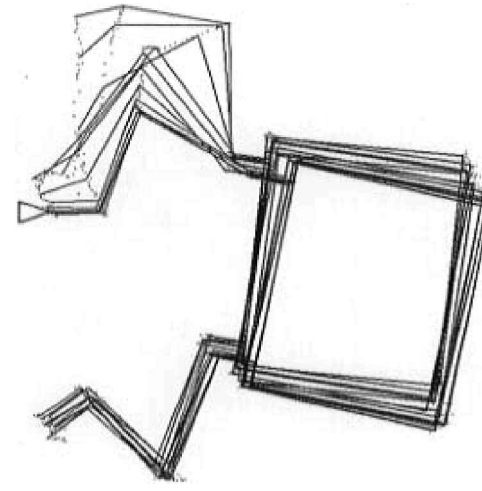

Fig. 5 Experimental result of manipulator berthing.

Assembly Experiments

An experimental result of the manipulator berthing is shown in Fig. 5, where the visual servoing with the sensory feedback control for space robots is used. The right manipulator hand is controlled well, and the manipulator berthing is successful, whereas the satellite vehicle is moved by the reaction of the arm motion and the disturbance of cables suspended from above.

Figure 6 graphically illustrates experimental results of the assembly sequence. The robot holds on to the worksite by its right arm to compensate for reaction force through the assembly. The arm path is planned and the manipulator is controlled to track the obtained path. The component installation is performed well by the SP-DF control.

Figure 7 is a series of photographs of the experimental assembly. Panel i is of the manipulator berthing corresponding to Fig. 5. The robot installs the first component, member 1, during panels ii and iii for step 1 in Figs. 4. The robot installs other members successively and assembles one truss unit from panels iv–vi, steps 2–4 in Fig. 4.

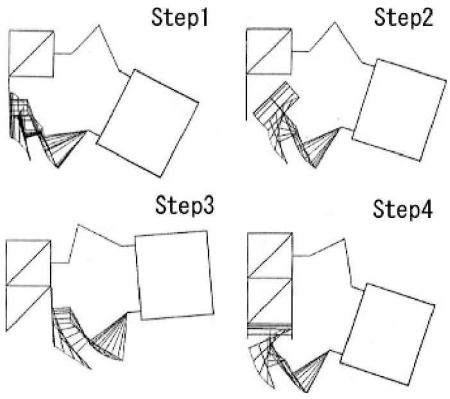

Fig. 6 Experimental result of truss structure assembly.

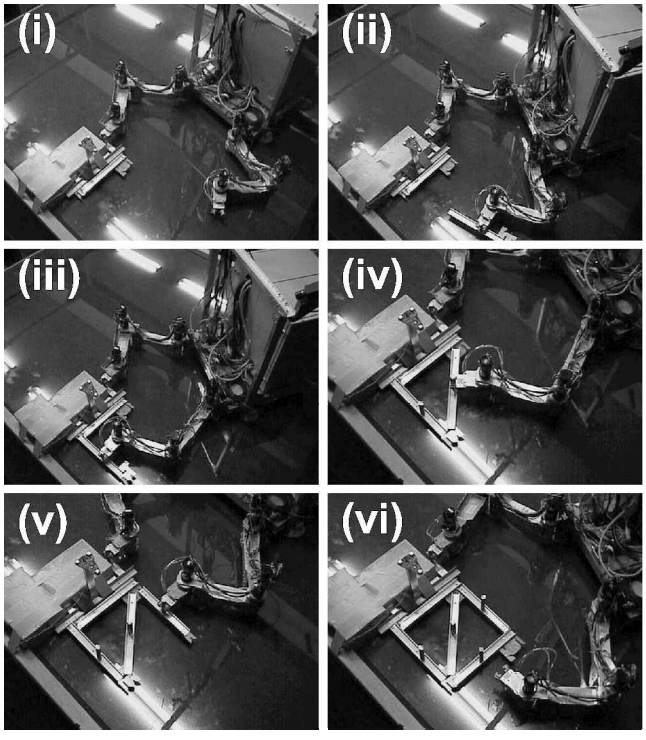

Fig. 7 Photograph of truss structure assembly.

Evaluation

The robot friendly truss is one of the main reasons why the robot has succeeded assembling whereas the vision system has a 2-mm mean measurement error after a hand–eye calibration. However, success is not ensured because of the measurement error.

Autonomy with Learning

In the preceding section, the robot has successfully achieved the truss structure assembly of the task sequence planned in advance. However, the space robot may sometimes fail in the task because of uncertainties and variations in the worksite. To recover from the error and obtain the task goal, the robot must replan the task suitable for the real work environment.

Application of Reinforcement Learning

For the replanning, one of typical reinforcement learning algorithms, Q learning,[25] is used. The reinforcement learning is used because the robot learns how to perform suitably for the real environment to maximize a numerical reward that is given by the designer where the environment cannot be modeled exactly.

Time t, state s, and action a are discretized following a general Q-learning formulation. The Q-learning algorithm estimates the optimal action-value function $Q(s, a)$ through interactions between the robot and the environment with trial-and-error processes. The $Q(s, a)$ evaluates action a at state s. During learning, the robot chooses a from s using a policy derived from Q. The robot takes action a, observes new state s' and reward r, and updates Q as

$$Q(s, a) \leftarrow (1 - \alpha) Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') \right] \quad (9)$$

where $\alpha (0 < \alpha \leq 1)$ and $\gamma (0 < \gamma \leq 1)$ are a learning rate and a discount rate, respectively. It has been shown that the estimated Q converges to the optimal if the system is modeled as a finite Markovian decision process and all actions are chosen enough times. To choose the action appropriately through learning, this study uses the ϵ-greedy policy,[25] where any action is selected randomly with probability ϵ, otherwise the optimal action is chosen by using the current estimated $Q(s, a)$.

Autonomous Actions with Learning

Case 1: Compensation for Measurement Error

The bad lighting condition in the space environment often yields a measurement error in visual sensing. Consider a situation in which the space robot fails in step 1 of Fig. 6 because of the measurement error. Let the robot acquire suitable actions for the situation by using the reinforcement learning with trial-and-error process. Task achievement is examined by sensor information regarding joint angles and applied forces because the component is not moved when it is installed in the right node and latched correctly.

A discrete state space with $9^3 = 729$ states is made for the reinforcement learning, where each coordinate of hand position (x, y) and orientation θ is discretized in nine states. The x and y coordinates are quantized every 0.02 m and θ every 2.0 deg. Each state has $2 \times 3 = 6$ actions that are one-step movements of discrete coordinates to the neighbor. Parameters in Eq. (9) for updating Q are $\alpha = 0.1$, $\gamma = 0.6$, and $r = 10$. For the ϵ-greedy policy, $\epsilon = 0.1$ is initially used and reduced gradually to be the policy deterministic. The manipulator is controlled by the force control of Eq. (6) for a contact situation.

Figure 8 is the graphic of the experimental robot motion. After an adequate trial-and-error process, the robot obtains adaptive action suitable for the measurement error. The learning method enables it to accomplish the task of compensating for the measurement error. This action with learning can be an approach to the sensor-fusion problem.

Case 2: Adaptive Action to New Environment

Consider a situation in which the diagonal element of step 2 is lost after step 3 during the truss structure assembly sequence of Fig. 6. In this situation, the robot cannot assemble the diagonal element into the truss structure by the sequence planned in advance because the element attached in step 3 becomes an obstacle. In the preceding section, the manipulator path is planned by the artificial potential method. However, the artificial potential method is not suitable for the online planning because it requires greater computational cost as the work environment gets more complicated.

A discrete state space with $15^3 = 3375$ states is made for the reinforcement learning, where each coordinate of hand position (x, y) and orientation θ is discretized in 15 states. The x and y coordinates

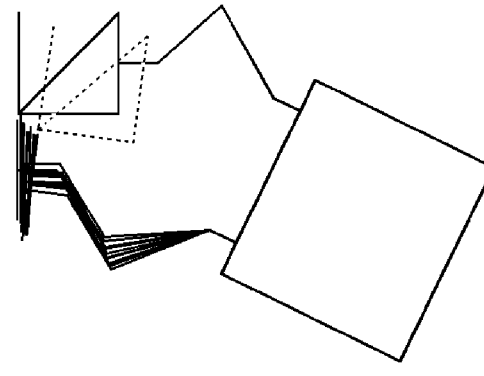

Fig. 8 Experimental result of case 1. (Dashed line represents the measured position of truss by the vision sensor with measurement error.)

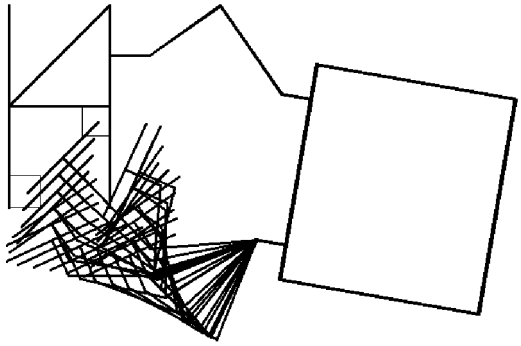

Fig. 9 Experimental result of case 2.

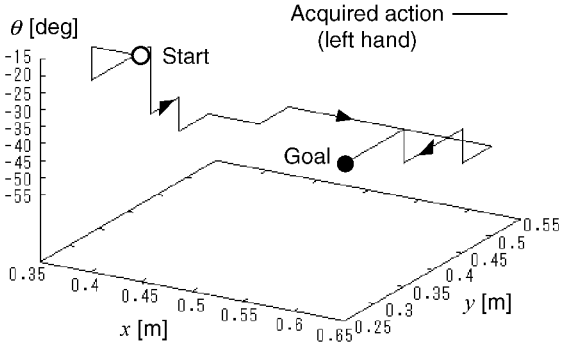

Fig. 10 Acquired action of position and orientation of left hand.

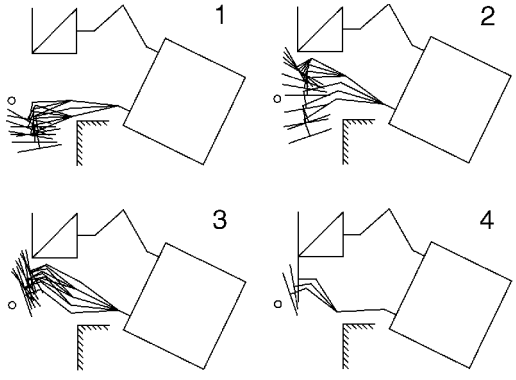

Fig. 11 Experimental action acquired to avoid collision.

are quantized every 0.05 m and θ every 10 deg. Other conditions are the same as case 1.

Figure 9 is the experimental robot motion planned by the reinforcement learning. The generated task sequence in the state space is illustrated in Fig. 10. The learning method enables accomplishment of the task while avoiding collision against the environment.

Most path planning methods generate a path from the initial state to the desired state. On the other hand, the reinforcement learning estimates the optimal action-value function Q for all states, which derives a policy to choose the best action. Therefore, the robot can take the best action at any state in the environment after the optimal Q is estimated.

Case 3: Complicated Obstacle Avoidance

Consider a situation in which the robot cannot assemble a component of the truss structure by the sequence planned in advance due to unexpected obstacles that interfere with the manipulator motion. Figure 11 is the graph of this experimental robot motion, where the suitable action for the environment is obtained by the reinforcement learning using the same conditions as case 2. The learning method enables the robot to perform such a complicated action to avoid collision with the obstacles in the environment.

Table 2 Number and period of computations for convergence of learning

Parameter	Case 1	Case 2	Case 3
State no.	729	3,375	3,375
Trial and error no.	33,666	238,092	154,043
Episode no.	1,041	854	886
Period, s	3	15	25

Evaluations and Discussion

For the three preceding cases, Table 2 shows the step numbers of the trial-and-error process, the episode numbers, and the periods for learning convergence, where an episode is a process from start to goal. The computation periods are measured by a Pentium II 266-MHz CPU for numerical simulations using modeled environments. From case 1 to case 3, the learning method requires a longer period as the environment becomes more complicated. The computation periods are within a few tens of seconds, and the learning method is feasible for the class of problems.

Here, the learning method acquires actions for the basically kinematic problems. For a dynamic problem, it requires a larger number of states and actions to treat the state space with a higher dimension and to model the interaction between the robot and the environment. For this class of dynamic problems, the computational periods and efforts are infeasible for online learning. An approach to this class of problems is still an open problem.

Conclusions

This study has demonstrated autonomous truss structure assembly by an experimental autonomous space robot system. The fundamental techniques have been developed and synthesized for the assembly task, that is, the stereo image measurement, the visual servoing, the positioning control of the free-floating space robot, the arm path planning, and the force control taking into account contact with the work environment. The robot successfully achieved the autonomous truss structure assembly. Furthermore, the robot replanned the task sequence by using reinforcement learning and obtained the goal even when the robot failed in the task sequence planned in advance. As a result, this study has shown a possible method of constructing the truss structure, as well as the usefulness of space robots.

There remain some subjects of study where autonomous space robots are concerned. The approach to autonomy and/or intelligence is the primary subject to be studied toward the realization of useful space robots. This study has approached this issue by use of the reinforcement learning algorithm, the application of which is still limited.

Acknowledgments

A part of this work is financially supported by a Grant-in-Aid for Scientific Research from the Ministry of Education, Science, Culture, and Sports of Japan and a Grant-in-Aid for Advanced Scientific Research from Osaka Prefecture University.

References

[1]Umetani, Y., and Yoshida, K., "Continuous Path Control of Space Manipulators Mounted on OMV," *Acta Astronautica*, Vol. 15, No. 12, 1987, pp. 981–986.

[2]Masutani, Y., Miyazaki, F., and Arimoto, S., "Sensory Feedback Control for Space Manipulators," *Proceedings of International Conference on Robotics and Automation*, Inst. of Electrical and Electronics Engineers, New York, 1989, pp. 1346–1351.

[3]Yamada, K., and Tsuchiya, K., "Efficient Computation Algorithms for Manipulator Control of a Space Robot," *Transactions of the Society of Instrument and Control Engineers*, Vol. 26, No. 7, 1990, pp. 765–772 (in Japanese).

[4]Murotsu, Y., Tsujio, S., Senda, K., and Hayashi, M., "Trajectory Control of Flexible Manipulators on a Free-Flying Space Robot," *IEEE Control Systems*, Vol. 12, No. 3, 1992, pp. 51–57.

[5]Murotsu, Y., Senda, K., Ozaki, M., and Tsujio, S., "Parameter Identification of Unknown Object Handled by Free-Flying Space Robot," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 3, 1994, pp. 488–494.

[6]Alexander, H. L., "Experiments in Control of Satellite Manipulators," Ph.D. Dissertation, Dept. of Electrical Engineering, Stanford Univ., Stanford, CA, Dec. 1987.

[7]Umetani, Y., and Yoshida, K., "Theoretical and Experimental Study on In-Orbit Capture Operation with Satellite Mounted Manipulator," *Preprints of XIth IFAC Symposium on Automatic Control in Aerospace*, International Federation of Automatic Control, Luxembourg, 1989, pp. 137–142.

[8]Komatsu, T., Ueyama, M., and Iikura, S., "Autonomous Satellite Robot Testbed," *Proceedings of International Symposium on Artificial Intelligence, Robotics and Automation in Space*, REN Associates, Inc., Tokyo, 1990, pp. 113–116.

[9]Toda, Y., Iwata, T., Machida, K., Otuka, A., Toriu, H., Shinomiya, Y., Fukuda, Y., Asakura, M., and Matuhira, N., "Development of Free-Flying Space Telerobot: Ground Experiments on Two-Dimensional Flat Testbed," *Proceedings of Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1992, pp. 33–39.

[10]Murotsu, Y., Senda, K., Mitsuya, A., Fujii, K., and Nunohara, T., "Experimental Studies for Control of Manipulators Mounted on a Free-Flying Space Robot," *Proceedings of International Conference on Intelligent Robots and Systems*, Inst. of Electrical and Electronics Engineers, New York, 1993, pp. 2148–2154.

[11]Bejczy, A. K., "Toward Advanced Teleoperation in Space," *Teleoperation and Robotics in Space*, AIAA, Washington, DC, 1995, pp. 107–138.

[12]Shimoji, H., Inoue, M., Tsuchiya, K., Ninomiya, K., Nakatani, I., and Kawaguchi, J., "Autonomous Capture Experiment of Free-Flying Target on the Zero Gravity Simulator," *Proceedings of International Symposium on Artificial Intelligence, Robotics and Automation in Space*, REN Associates, Inc., Tokyo, 1990, pp. 77–80.

[13]Skaar, S. B., and Ruoff, C. F. (eds.), *Teleoperation and Robotics in Space*, AIAA, Washington, DC, 1995, Chap. 10.

[14]Brunner, B., Hirzinger, G., Landzettle, K., and Heindl, J., "Multisensory Shared Autonomy and Tele-Sensor-Programming,"*Proceedings of International Conference on Intelligent Robots and Systems*, Inst. of Electrical and Electronics Engineers, New York, 1993, pp. 2123–2139.

[15]David, L., "Robots for all Reasons," *Aerospace America*, Sept. 1995, pp. 30–35.

[16]Yoshida, K. (ed.), "Robot Experiments on ETS-VII," *Journal of Robotics Society of Japan*, Special Issue, Vol. 17, No. 8, 1999, pp. 1055–1104 (in Japanese).

[17]Senda, K., Murotsu, Y., Nagaoka, H., and Mitsuya, A., "Attitude Control for Free-Flying Space Robot with CMG," *Proceedings of Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1995, pp. 1494–1502.

[18]Murotsu, Y., Senda, K., Mitsuya, A., Yamane, K., Hayashi, M., and Nunohara, T., "Theoretical and Experimental Studies for Continuous Path Control of Flexible Manipulators Mounted on a Free-Flying Space Robot," *Proceedings of Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1993, pp. 1458–1471.

[19]Corke, P. I., "Visual Control of Robot Manipulators—A Review," *Visual Servoing*, edited by K. Hashimoto, World Scientific, Republic of Singapore, 1993, pp. 1–31.

[20]Senda, K., Murotsu, Y., Nagaoka, H., and Mitsuya, A., "Position/Attitude Control for a Ground Testbed Simulating a Space Robot," *Journal of Robotics Society of Japan*, Vol. 16, No. 6, 1998, pp. 824–831 (in Japanese).

[21]Connolly, C. I., Burns, J. B., and Weiss, R., "Path Planning Using Laplace's Equation," *Proceedings of International Conference on Robotics and Automation*, Inst. of Electrical and Electronics Engineers, New York, 1990, pp. 2102–2106.

[22]Tsuji, T., Morasso, P. G., Shigehashi, K., and Kaneko, M., "Motion Planning for Manipulators Using Artificial Potential Field Approach that Can Adjust Convergence Time of Generated Arm Trajectory," *Journal of Robotics Society of Japan*, Vol. 13, No. 2, 1995, pp. 285–290 (in Japanese).

[23]Takegaki, M., and Arimoto, S., "A New Feedback Method for Dynamic Control of Manipulators," *Journal of Dynamic Systems, Measurement and Control*, Vol. 102, No. 2, 1989, pp. 119–125.

[24]Arimoto, S., "Quasi-Natural Potential and Saturated-Position Feedback," *Control Theory of Non-linear Mechanical Systems*, Oxford Univ. Press, Oxford, 1996, pp. 93–121.

[25]Sutton, R. S., and Barto, A. G., *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA, 1998, Chap. 6.

D. B. Spencer
Associate Editor