Control System of the Self-Mobile Space Manipulator

Yangsheng Xu, *Member, IEEE*, H. Benjamin Brown, Jr., Mark Friedman, Takeo Kanade, *Fellow, IEEE*

Abstract— **The Self-Mobile Space Manipulator, $(SM)^2$, is a simple, 5-DOF, 1/3 scale, laboratory version of a robot designed to walk on the trusswork and other exterior surfaces of Space Station Freedom. It will be capable of routine tasks such as inspection, parts transportation and simple maintenance procedures. We have designed and built the robot and gravity compensation system to permit simulated zero-gravity experiments. We have developed the control system for the $(SM)^2$ including control hardware architecture and operating system, control station with various interfaces, hierarchical control structure, multi-phase control strategy for step motion, and various low-level controllers. The system provides operator friendly, real-time monitoring, robust control for 3-D locomotion movements of the flexible robot. A hierarchical structure allows the control to be executed in various levels autonomously or by teleoperation, and a multi-phase control strategy facilitates the control in different tasks. Based on the dynamic model developed, a linear-structured joint-level controller and model-based control scheme with acceleration feedback is being implemented to provide a stable and fast motion. The configuration-independent control scheme allows the control parameters to adapt to changes in system dynamics due to the robot configuration variation. With a variety of low-level controllers we developed, the system has demonstrated to be robust to the uncertainties in modeling and in payload.**

I. Introduction

ASTRONAUT extra-vehicular activity (EVA) at a space station is costly, potentially dangerous, and requires extensive preparation. Some EVA tasks, such as unplanned repairs, may require the versatility, skill, and on-site judgment of astronauts. Many other tasks, particularly routine inspection, maintenance and light assembly, can be done more safely and cost effectively by robots.

Robots designed for EVA work on Space Station Freedom are capable, but expensive and highly complex. NASA's Flight Telerobotic Servicer (FTS) and Canada's Special Purpose Dextrous Manipulator (SPDM) are examples of such EVA robots. These two human-size robots depend on other devices, such as the giant space station remote manipulator system and its spider-like mobile transport system, to transport them to space station work sites. This dependency could limit their usefulness.

We have developed a relatively simple, modular, low mass, low cost robot for space station EVA that is large enough to be independently mobile on the station exterior, yet versatile enough to accomplish many vital tasks. Because our design is for a robot that is independently mobile, yet capable of conventional manipulation tasks, we call it the Self-Mobile Space Manipulator, or $(SM)^2$.

The $(SM)^2$ walks on the nodes of a suitably scaled truss structure, and demonstrates capabilities of material transporting and simple manipulation. Optimized for the micro gravity conditions of space, and sized to step between nodes that are 1.67 m apart, the robot's links are constructed from thin-walled tubing to keep its mass to a minimum. This produces a robot that is highly flexible in structure, and controlling a 3-D locomotion movements of such a robot is a challenge. To simulate zero-gravity environment for realistic testing of the prototype robot's performance, we have developed a position servoed gravity compensation system utilizing video tracking and an innovative counterbalancing mechanism.

Salient features of the $(SM)^2$ control system include the hierarchical control structure, the multi-phase control strategy, and various robust low-level controllers such as acceleration feedback control scheme and configuration-independent control scheme. To execute the robot motion in different levels, from stepping sequences to joint motion control, the hierarchical control system is designed and can be executed autonomously or by various human interfaces in the control station. In the single step motion from one node to another, multi-phase control strategy has been implemented. For a large coarse motion a low-gain linear control scheme or a model-based control scheme with acceleration feedback provides a stable, fast motion. In the specified destination vicinity, a high-gain control scheme is executed incorporated with an automatic vision system measuring the destination with respect to the robot tip.

We have developed a dynamic model which has been verified by experiments and is being used for model-based control. A configuration-independent control scheme allows the control parameters to adapt to the changes in dynamic model, due to the variation of robot configuration. An adaptive control scheme for unknown mass/inertia of an object that is manipulated or transported by the manipulator has been successfully implemented. With these robust controls, the robot system performance is highly improved, subject to the unmodeled dynamics, due to unknown payload to be manipulated, modeling error, and disturbances from space station structure.

This paper will report on the hardware development of the gravity compensation system and the scaled-down version

Manuscript received July 9, 1993. This work was supported by the Space Projects Office, Shimizu Corporation, Japan, and the Robotics Institute at Carnegie Mellon University. Paper recommended by Associate Editor D. Repperger.

The authors are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA.

IEEE Log Number 9403415.

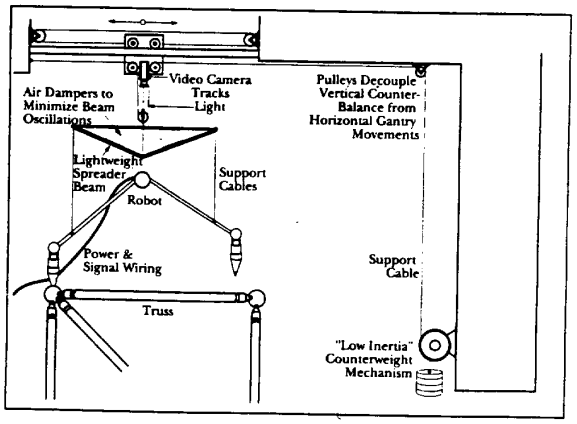


Fig. 1. Gravity compensation system simulates zero gravity for realistic laboratory experiments. A passive system of counterweights, cables and pulleys provide a constant, vertical balance force. A powered overhead carriage is servocontrolled to keep the support point directly above the robot.

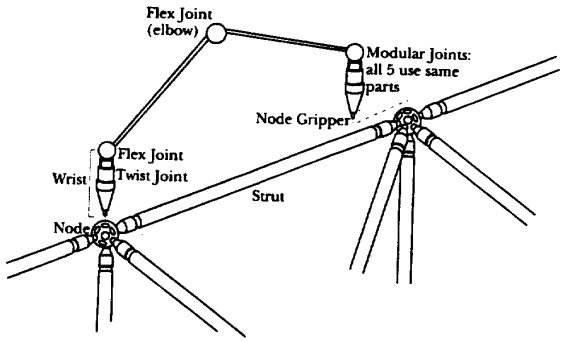


Fig. 2. The robot has five joints connected by two slender links. Grippers at each end attach to threaded holes in the truss nodes, enabling the robot to walk by stepping from node to node.

of the robot mechanism in Sections II and III. The control hardware architecture and control station are discussed in Sections IV and V. The high level structure and features of control software are addressed in Sections VI and VII. Dynamic modeling, and low-level controllers are presented in the rest of paper.

II. Gravity Compensation System

The zero-gravity environment at an orbiting space station has significant impact on the design and performance of a robot. The absence of gravitational forces permits a long, spindly robot to move relatively large masses with small force and power consumption. In order to perform realistic experiments on earth, we have developed two gravity compensation systems that balance gravitational effects on the robot so that it behaves as if it were weight-less.

The first gravity compensation system (Fig. 1) includes a passive, vertical counterweight system, and an actively controlled, horizontal system. The vertical system comprises a counterweight mechanism, a spreader beam, and a series of cables and pulleys. The counterweight mechanism provides a constant vertical force to the end of the support cable to balance the weight of the robot. Because of the 10:1 ratio of the mechanism, the counterweight moves very slowly, stores little energy, and consequently increases the inertia of the system by only 10% in the vertical direction. The support cable is routed through idler pulleys on the overhead structure and spreader beam in a manner such that horizontal motions of the carriage have no effect on the vertical balance force, aside from those due to pulley friction and inertia. The spreader beam transfers the force from a single support point to two balance points on the robot links. Its lightweight design and the use of kite-like air dampers minimize beam oscillations that can disturb the robot and gravity-compensation active control system.

The overhead carriage is actively controlled in the two horizontal axes to maintain the suspension point for the support cable directly above the robot so that the compensating force is purely vertical. A video camera mounted at the center of the carriage tracks an infrared LED at the center of the spreader beam. An automatic vision system locates the image of the LED in the camera's field of view, and generates error signals proportional to the horizontal displacement of the image relative to the camera. A motor/control system drives the carriage in two axes to minimize the error signal, thus keeping the carriage directly above the robot. Control gains are selected to give stable motion during large excursions of the robot; when the speed goes below a preselected threshold value, gains are automatically increased by the control system to minimize static error. The two-phase control strategy improves overall performance substantially.

The first gravity compensation system has allowed $(SM)^2$ to walk on one vertical face of the trusswork, and from one face to another with the truss at 45 degrees to horizontal. Discrepancies in the compensation forces due to friction and tracking errors amount to about 1% of the robot's weight in the vertical direction and 2-4% in the horizontal.

To improve dynamic response of the system, we have developed a second gravity compensation system. The configuration of the system is in polar coordinates, with tangential and radial motion actively controlled, and vertical motion compensated for by a passive counterweight system. Two DOF optical sensors are mounted to measure the tangential and radial motions. The new system provides two independent, actively controlled, suspension points needed for experiments in manipulation and payload transport where additional weights must be supported. The new system has greatly improved the system tracking capability and has been successfully used for experiments of fine manipulation when the robot base is fixed. For general walking on trusswork where both feet of the robot could detach for sequential stepping motion, the first system is still useful.

III. Robot Design

$(SM)^2$ was conceived and designed to have the minimum size and complexity needed for walking on the space-station trusswork. The basic walker includes five rotational joints and

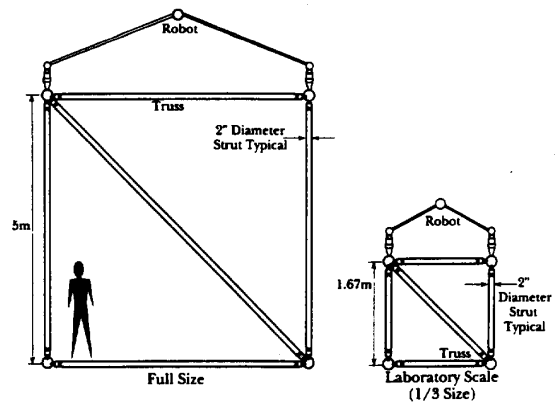


Fig. 3. Overall dimensions of the truss and robot are scaled to 1/3 to permit experiments in the laboratory, while local dimensions (sizes of nodes, joints, and grippers) are the same to keep local behavior similar, and mechanism size workable.

two slender links (Fig. 2). Grippers at each end of the robot enable it to attach itself to threaded holes in the truss nodes or other regular structure. Walking is accomplished by alternate grasping and releasing of the nodes by the grippers, and swinging of the feet from one node to the next. During each walking step, one end of the robot releases from a node, swings 90 or 180 degrees to a desired node location, and re-attaches to that node. $(SM)^2$ moves along the trusswork using such steps with alternate feet. To minimize size, mass and structural compliance, the robot has sufficient span to just reach between adjacent nodes, a distance of 1.67 m on the 1/3 scale trusswork. In theory, $(SM)^2$ can access any unobstructed point on the exterior of the trusswork. Because of its relatively small size (compared to other space station structures and robots), low mass, and low power consumption, $(SM)^2$ has the potential to be useful in light construction, inspection and maintenance tasks on the space station or other space structures.

As a starting point in the robot development, we designed a hypothetical, full-size, self-contained robot to be used on the space station. The design included estimated masses of the major components: motors, drives, links, manipulation devices, and on-board power supply (batteries) needed for a reasonable travel range. The design also considered link and joint compliances, and the resultant structural vibration frequencies. Then a 1/3-size laboratory robot was designed and built using scaling rules to keep the dynamic parameters (masses, stiffnesses, natural frequencies, linear speeds) of the scaled-down robot similar to that of the hypothetical one. Overall dimensions of the truss and robot were reduced to 1/3, while local dimensions (truss nodes, joints and grippers) were kept equal (Fig. 3). This allows the testbed to be used in an average size laboratory, while mechanisms are not unworkably small.

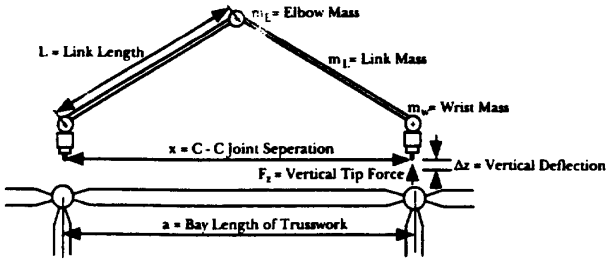
Fig. 4 gives some basic parameters for the full-size and scaled designs. Several parameters are noteworthy. Step time (the time for the robot to swing through a 180-degree arc) was set to 20 s for full scale, 6.7 s for 1/3 scale; this was judged to be fast enough for reasonable travel speed, yet slow enough for reasonable energy efficiency and safety. At both scales, the lowest structural frequency is 2.2 rad/s (0.35 Hz), and the tip deflects 148 mm (5.8 in) under maximum joint torque while producing 9.0 N (2.0 lb) of force. Thus, the robot is highly flexible and generates relatively small forces at the tip. These characteristics dictate the kind of tasks for which $(SM)^2$ is suited, and highlight the need for a gravity-compensation system to remove the disturbances of gravitational forces on the robot.

To simplify repairs to the robot and minimize the required inventory of parts, $(SM)^2$ was designed with five compact, modular, self-contained joints, all identical except for gear ratios. Each joint contains a rare-earth-magnet DC motor; harmonic-drive speed reducer (60:1 or 100:1 ratio) and joint angle sensors [3]. The motors and drive components were selected and arranged to give maximum power and torque in a small, lightweight package. A combination of conductive plastic potentiometer and incremental optical encoder provides both absolute position information and a low-noise signal for precise position and velocity measurement. Each joint has a mass of 0.55 kg (1.2 lb). For the 100:1 joints, peak torque is 13.7 N-m (125 lb-in) and maximum speed is about 5.8 rad/s (0.92 rev/s). For the 60:1 joints, peak torque is 8.2 N-m (75 lb-in) and maximum speed is 9.7 rad/s (1.5 rev/s).

We intended to produce a robot structure that could be simply modeled for control purposes. To this end, we kept the links as light as possible so that all mass could be assumed to reside at the joints and end-effectors. Further, we attempted to keep most of the mass at the two ends of the robot, maximizing the frequencies of vibration associated with the mass of the middle (*elbow*) joint. An ideal model would then include only point masses at the two ends, greatly simplifying the dynamics; a more realistic model includes a third point mass at the elbow. (Stiffening the links to make the robot behave as a rigid arm results in a drastic mass increase. For example, to reduce the full-load static deflection from 148 mm to 5 mm would require a 5.5-fold increase in the link mass, assuming we maintained the same wall-thickness ratio for the links.)

The node gripper, the device that attaches the robot to the nodes of the trusswork, is a critical part of the design. Unlike a typical robot end-effector, it must be able to anchor the robot firmly to the nodes since the robot's base of support shifts from one end-effector to the other during walking. The node gripper [3] includes a motor-driven screw that engages the threaded holes in the nodes, and a cam mechanism that draws the gripper against the node with more than 1800 N (400 lb) of force. This large anchoring force is needed to prevent twisting or rocking on the node, which would disturb the robot's frame of reference. Signals from a gap-sensing button and motor-current sensor are used in the automatic control of the gripping and ungripping cycles.

We are developing an end-effector to enable $(SM)^2$ to carry and manipulate objects. This end-effector will have two joints of the same, modular design and a simple, general-purpose gripper, and will attach to the side of the node gripper [3]. The gripper will act as a controllable carrying device, and the combined 7 degrees-of-freedom (DOF) of the walker and end-effector will provide a general manipulation capability (Fig. 5).



Parameter	Formula	Full Size	1/3 Size
Bay Length	a	5.00m	1.67m
Link Length	L	2.91m	0.97m
Link Tube Dimensions (OD x wall thickness)[1]		51mm x 1.0mm	19mm x 0.7mm
Link Mass[2]	m_L	1.25kg	0.11kg
Wrist Mass*	m_w	14kg	14kg
Tip Stiffness (@ x= 21)[2]	$k=3EI/(2L)^3$	55.6N/m	55.4N/m
Joint Torque, peak[3]	T	41.1 N-m	13.7N-m
Tip Force @ Max Joint Torque[3]	F_z	9.0N	9.0N
Tip Deflection @ Max Joint Torque[3]	$\Delta z=T(2L)^2/3EI$	148mm	148mm
Lowest Natural Frequency (@ x=21)[4]	$\omega_N=\sqrt{(k/m_w)}$	2.20 rad/sec	2.19 rad/sec
Joint Speed, max[3]	$\dot{\theta}$	1.93 rad/sec	5.8rad/sec
Step Time (nominal for 180° step)	t_{180}	20sec	6.7sec

* Estimated for self-contained robot with auxiliary manipulators.
1 Link cross-sectional dimensions were scaled to maintain proportional outside diameters and equal linear stiffness at the tip
2 Based on 6061-T6 aluminum, 2.8 g/cm3 density, 68,900N/mm2 elastic modulius
3 Based on 100:1 joint ratio
4 Assuming point mass at tip, all robot joints locked

Fig. 4. Scaled parameters for the full-size hypothetical and 1/3–scale laboratory robot.

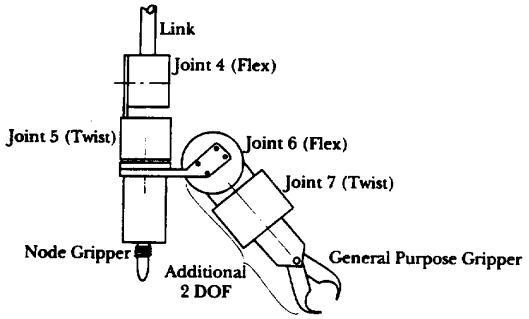


Fig. 5. End effector with two joints and general-purpose gripper will permit carrying and manipulation of parts.

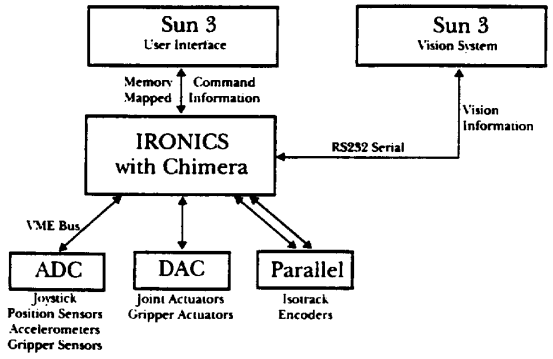


Fig. 6. The current control architecture is based on a single Ironics 68020 CPU communicating with the robot via ADC and DAC interfaces. Parallel and serial ports allow communication with vision systems and operator controls. A Sun3 permits operator inputs and graphically displays data.

IV. Control Architecture

The control architecture includes control computers, related hardware, real-time operating system and other software that is independent of the control strategies (such as I/O processing and communication with peripheral processes). Currently, real-time control is implemented on an Ironics 68020-based single board computer running the CHIMERA II real-time operating system [97]. Aside from supplying a high-performance real-time kernel, CHIMERA II provides multiprocessing features and a layer of transparency between the diverse hardware and the control software. A Sun 3/260 host workstation is used for code development and operator interfacing. The current hardware configuration is shown in Fig. 6.

The software architecture of the controller has evolved with the robot to include support for a wide range of devices and operations. The goal of this controller support software is to provide all the interfaces and procedures needed by the higher level strategy algorithms. Currently, this software supports several analog interfaces which carry motor and actuator commands to the robot, and sensor data from the robot to the control processor. Also supported are a number of parallel and serial devices. A full range of functions, including calibration and filtering of transmitted data are

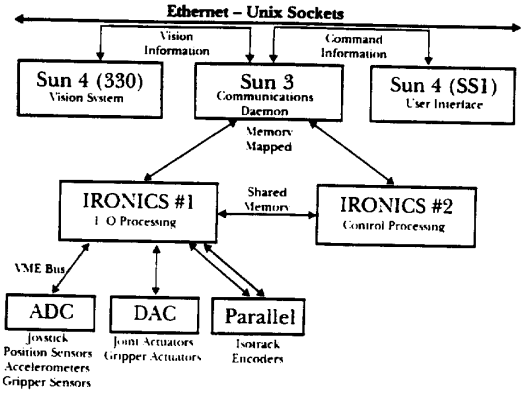


Fig. 7. The control architecture being developed will split I/O and control functions between two real-time control processes on separate Ironics CPL's. A Sun3 will mediate communication with other Sun4 processors which will provide enhanced interface graphics and vision processing.

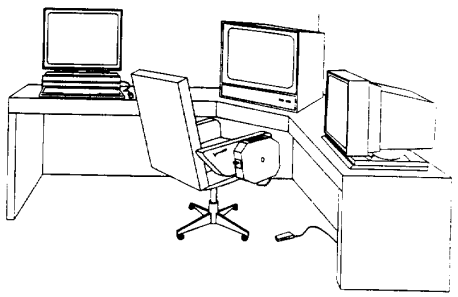


Fig. 8. Control station. To the operator's left is a sun workstation used to program the user interface. To the operator's right is the interface display for menu selections (see Figs. 15 and 16). The monitor in front of the operator shows video from the camera mounted on the robot's active gripper. Hand controls (described in the text) are mounted on the chair arm. The foot switch is to enable robot motion.

provided by the support software. The CHIMERA II system allows rigid control of the timing and sequencing of all operations.

We are currently upgrading the hardware and support software to increase processing power and facilitate software modifications. A second Ironics processor will double the power of the CHIMERA II system and allow separation of I/O functions and control algorithms. A new I/O interface, using a Sun 3 communications daemon, will allow better communication with the user interface, vision system and other Unix-based processes, and will allow each process to be developed independently. Several new Sun SPARC systems will greatly improve the performance of the vision systems and user interface. The second generation hardware is shown in Fig. 7.

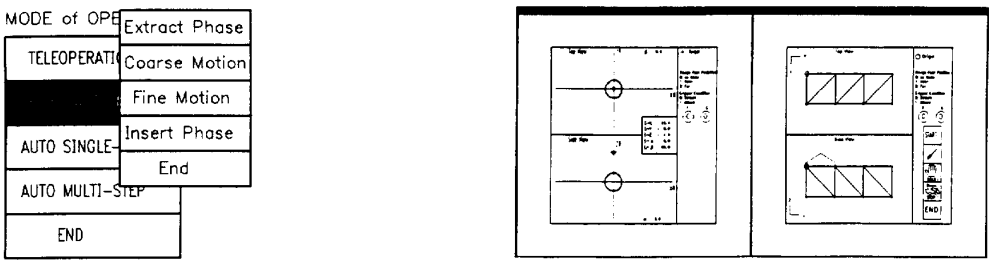
V. Control Station

We are developing a control station designed for remotely controlling the robot system, taking advantage of both human guidance and the computer-driven controls of an autonomous robot. We want to provide smooth transitions from low-level, joint-specific teleoperation by an operator at the robot's control station through various levels of telerobotic control, where the human operator provides increasingly complex and higher level instruction to the robot, all the way to goal-specified, semi-autonomous operation. A hierarchical, video-display-based control structure allows the human operator to choose an appropriate level of control for a given task or robotic motion component. The control system is model-driven, so predictive displays are available for automatically controlled operating modes.

Using trackball, joystick, or computer mouse, the user interface presents the operator with an interactive visual display to specify the desired level of robot control. If the least automated mode of robot control, teleoperation, is selected from the menu display, then the operator may position the robot by using the computer display and one of the position control devices. The user selects which positioner he wants by rotating the box by the side of the controller's chair (Fig. 8). Trackball or joystick would be used to control selected pairs of joints, while the articulated hand controller would be used to simultaneously specify all five joint positions. The Polhemus 6-DOF hand controller may be used to specify a target position for the robot's free end [4]. This latter controller is an electromagnetic sensor built into a stylus grip that can detect its position, relative to a small transmitter module, within a radius of about 1 m of the transmitter. By analogy with the computer *mouse*, we refer to the 6-DOF Polhemus stylus as a *bat* (i.e., a flying mouse). When the bat is selected, then computer mediation helps translate from the 6-DOF of the controller to the 5-DOF possible with the robot.

Selecting the *phased automatic motion* control mode from the display menu provides the operator with computer-mediated assistance in specifying the components of robot motion necessary for locomotion (Fig. 9(a)). This mode is useful when automatic control of locomotion is inadequate or fails. For example, unusual lighting may prevent the vision system from properly aligning the robot's node gripper for insertion. The operator would select the fine motion option and use one of the hand position controls while viewing computer graphics based on robot joint encoders (Fig. 9(b), left) and a view from the gripper-mounted camera to guide the gripper to the insertion point. A similar procedure is used for guiding larger robot motions on the trusswork (Fig. 9(b), right). The graphics used to specify this *coarse motion* is shown on the right side of Fig. 9(b). The operator can command the gripper to release a node using the *extract phase* selection (Fig. 9(a)) and initiate an attachment sequence using the *insert phase* selection. While guiding the robot with these computer-assisted controls, exceptional conditions can be dealt with by holding the robot's present position, by cutting power with an emergency stop command, or by reverting to joint-level teleoperation.

Displays for higher level control allow the operator to specify a single step to a new position, or a sequence of steps starting with the release of a specified gripper and concluding with the insertion of the grippers at the designated



(a) (b)

Fig. 9. Screen graphics phased autonomous motion. (a) Menu for mode of operation. (b) Graphical x-y-z view of robot position for fine motion control (left) and coarse motion control (right).

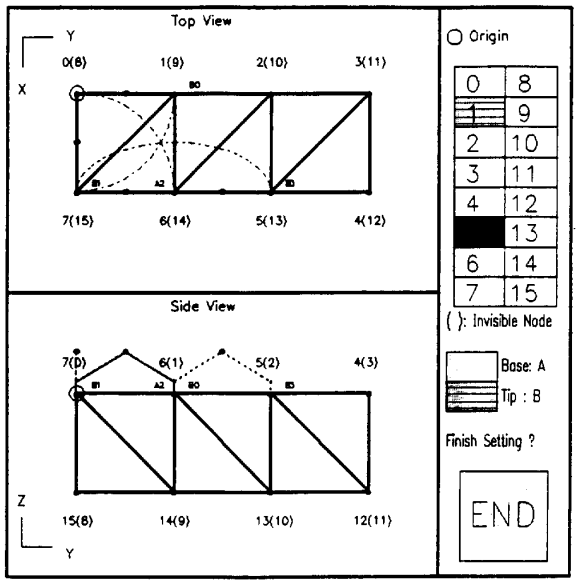


Fig. 10. Screen graphics for control of multi-step paths. Operator selects target nodes along path to destination, then can preview simulated motion before comanding execution.

target location (Fig. 10). For these semi-autonomous walking modes, the operator may preview model-based animation of the robot's computer generated route prior to giving the locomotion command. He can then accept or modify the automatically planned route and monitor the robot's progress on the display. During actual locomotion, the robot animation is driven by the robot's joint position sensors.

During both direct teleoperation and the higher levels of telerobotic locomotion control, cameras mounted on the end-effectors provide the operator at the control station with visual feedback for final docking (insertion) maneuvers. Also used by the robot's machine vision guidance system, these camera views are most useful to a human operator during the insertion phase of the stepping sequence to ensure correct alignment. To provide a broader view of the robot's activities, we plan to extend camera coverage so that one robot gripper can be seen from the perspective of the other gripper.

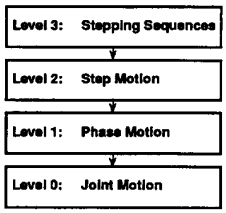


Fig. 11. The hierarchical control structure for general walking motion on space station trusswork.

Presently, the robot's control algorithms are most highly refined for locomotion, rather than manipulation tasks, and this is reflected in the operator interface. We specify position information for the robot's five basic degrees of freedom plus the functioning of its two end-effectors, the node-grippers. One current focus of development is to enhance the robot's manipulative capability by providing for hybrid position/force control and by increasing the number of degrees of freedom so that more complex manipulation tasks can be performed. A force reflecting, articulated hand controller, as well as improved visual and auditory feedback, will be provided for operator control of these extended robot manipulation capabilities.

VI. Hierarchical Control Structure

Control software of $(SM)^2$ has been developed primarily for 3-D reliable and accurate locomotion movements on space station trusswork. Recently we have been working on tasks of material transporting and manipulation. In this and the next sections, we intend to present high-level software structure; the rest of paper addresses the low-level control algorithms for various purposes.

The first feature of the control system is capable of executing hierarchically. The system can automatically generate motion specifications, optimize the controller, execute the system, and monitor the execution in different levels. For a typical 3-D walking on trusswork, four-level control flow is shown in Fig. 11.

The highest level is *executing stepping sequences* from the robot initial configuration to the final configuration with respect to the space station truss. Considering each node

on the truss as a sphere with threaded holes at 45 degree spacing, specifying the robot's two feet on two nodes of the truss as an initial configuration and as a final configuration, and then specifying the stepping motion traveling on each configuration, is too complicated. Instead, we developed a system using six parameters to describe the configurations and sweeping motion, so that optimal sequence of stepping motion on one face, or from one face to another, of the trusswork can be generated [13]. The generated stepping sequences can be displayed on screen so that the operator can preview the stepping sequences to assure they are satisfactory before commanding automatic execution.

The next level is *controlling motion in a single step* which is generated from the previous level or specified by operator. In each step, the robot detaches from a threaded hole on a node, and sweeps 90 degrees or 180 degrees about the first joint, and at the same time drives other joints to execute the specified trajectory to the destination hole on the other node. So in one step, one foot is fixed on a node, and the other foot travels from one node to another. For an efficient and accurate stepping motion, we employed a multi-phase control strategy in each step. Each step motion is decomposed into four phases: extraction from attached node, coarse motion from the node to the neighborhood of the destination node, fine motion to precisely locate the robot foot above the hole, and insertion in the destination hole.

Then, the following level is *controlling the motion in single phase*. We implemented different control laws to optimize the performance in different phases. For the previous two levels, control specifications and geometric parameters of the motion are fixed. In this level, however, we can specify all possible low-level control structures, control parameters, trajectories and arbitrary initial and final configurations of the robot. So this is the level normally used for performing control experiments. Various trajectories, such as parabolic and near-optimal trajectories [11], are available for a general motion. Since an arbitrary configuration can be specified, and various controllers and trajectories can be executed, manipulation and transporting experiments are also performed in this level.

The lowest level is *controlling individual joint motion*. The usage of this level is normally for testing the joint controller, executing certain joint(s) without involving forward and inverse kinematics for special purposes, or checking sensor reading in each joint.

VII. Multi-phase Control Strategy

The second feature of the $(SM)^2$ control system is multi-phase control strategy for the motion in each step. Controlling the $(SM)^2$, a 3-D flexible long-reach robot carrying possible large payload is a challenge due to potential vibration caused by high flexibility, unmodeled significant joint friction, positioning error amplified by a long-reach, and unavailability of global tip sensing [11]. For such a system, it is difficult to achieve a reasonable speed while at the same time maintain an accurate motion. Moreover, for locomotion or most tasks in space applications, requirements for the robot performance are different for different periods of time and various purposes.

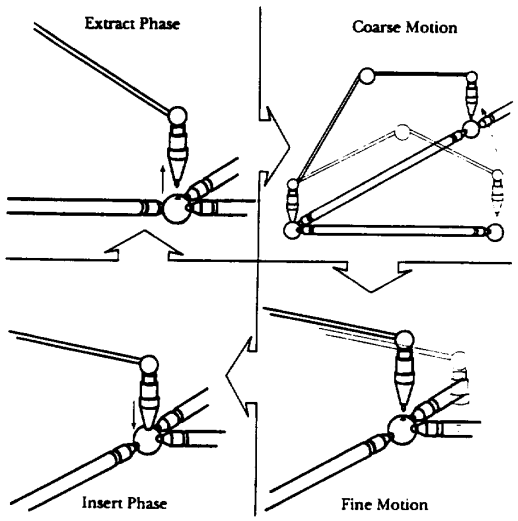


Fig. 12. Automatic control of walking is divided into four phases, each tuned to optimize performance in that phase.

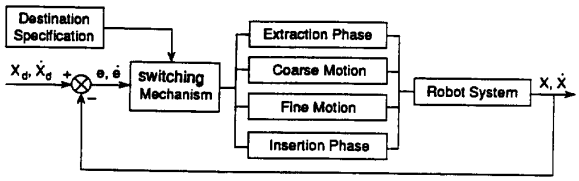


Fig. 13. Multi-phase control strategy for step motion.

Based on this fact, we partition the step motion into four phases, *extraction, coarse motion, fine motion, and insertion*, (Fig. 12).

In the *extraction* and *insertion phases*, the robot motion is combined with control of the node gripper, and gripper gap and motor current sensors are used. The goal in these two phases is to provide a reliable detaching and firm attaching, with a reasonable speed. In the *coarse motion*, a fast and stable motion is desired. The tracking error from the specified trajectory is not as important as motion efficiency. Therefore from a control point of view, a simple coarse motion can be considered as a regulation problem, rather than a tracking problem. When the position error is within a certain region near the destination node, the motion phase is switched to the *fine motion* in which the precise location is our main concern. The control block diagram is shown in Fig. 13 where the *switching mechanism* is a function to determine the motion phase depending on the position and velocity errors with respect to destinations in each phase that are specified automatically for step motion.

In *coarse motion*, we implemented a linear joint controller with low integral gains and low-pass filters. The control gains and the orders and cut-off frequencies of filters are determined by dynamics model and experiment performance. The sensor sampling rate is 50 Hz currently. The filter cut-off frequencies are normally set to 2–2.5 Hz to avoid the 3 Hz mode associated with the lateral vibration of elbow joint. The closed-loop

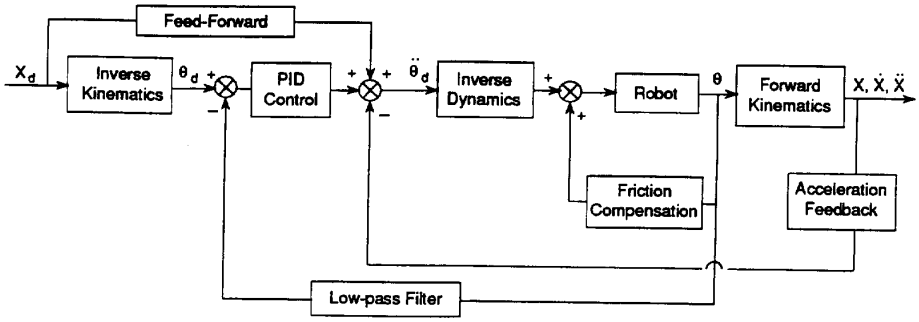


Fig. 14. Block diagram of typical low-level controller.

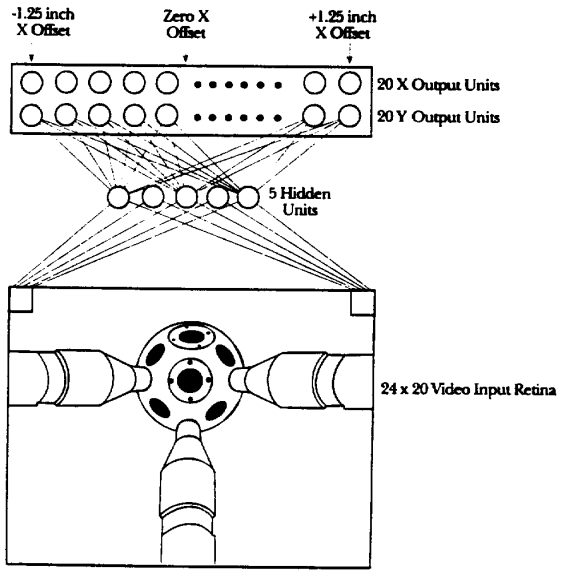


Fig. 15. Low resolution video images from a camera mounted at the robot's tip are processed by a three-layer, back-propagation neural network. The output units give the x-y displacements of the tip from the target position.

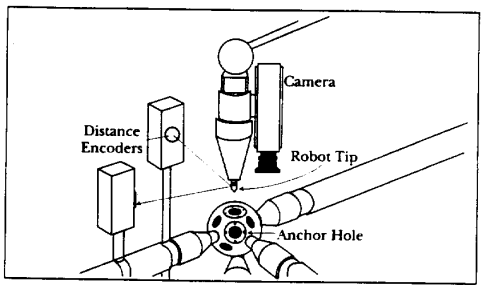


Fig. 16. The neural network is trained on a jig that provides x-y position data while video images are collected. Correlation of the position data and images enables the network to learn the proper mapping between input and output.

frequencies for the first three joints are set to about 1/6 of the filter frequencies, i.e., 1/3–1/2 Hz. Using acceleration feedback from the tip, a higher gain can be employed and filter delays can be compensated. A typical PID-based low-level controller is shown in Fig. 14. With this controller, the robot is able to execute a 90-degree sweeping step in 5 s, a 180-degree step in 7 s. We are developing a model-based control scheme for a fast coarse motion which is discussed in the section on model-based control.

In the *fine motion, extraction* and *insertion phases*, a high gain control is implemented to minimize steady-state error in order to achieve the precision needed for insertion of the node gripper. Because of the small range of motion, and thereby small deflections of structure in these phases, we believe a linear structured high gain control is sufficient, and model-based control is not necessary. During fine motion, we use an automatic vision system installed at the robot tip to measure the position error between the tip and destination node hole.

The vision system uses a video camera mounted at the end of the robot to provide images of the target node. A 3-layer, back-propagation neural network processes the low resolution, 24×20-pixel video images (Fig. 15). The output consists of two 20-element vectors representing the offset of the robot's tip relative to the node in the x and y dimensions [8]. The network learns to analyze the image through a training procedure in which the tip is moved by hand around the vicinity of the node hole. Several hundred video images are collected while a special jig (Fig. 16) measures the x-y position. Then a back-propagation algorithm is invoked to adjust the weights of the network's 2645 internal connections to produce the desired mappings between the input video images and output displacement vectors. After training, the vision system is able to recognize the target while processing images at 15 Hz, and provide error signals that enable the control system to bring the gripper screw into alignment with the hole. The neural network approach is particularly powerful because new or modified objects can be accommodated through a simple retraining operation.

The extraction and insertion controllers are identical, except these do not use the vision data. During extraction and insertion, additional signals from the gripper-gap sensor and motor current are used to automatically control the gripper cycle.

Friction in the motors and harmonic drives produces a significant disturbance to the control loop. We have found empirically that the torque needed to overcome friction is more than 10% of the maximum joint torque, and nearly 50% of normal operating torques. To compensate for frictional effects, the control system increments the joint torque by a constant value in the direction of motion. This scheme improves static positioning accuracy at the tip from about 25 to 5 mm.

VIII. Dynamic Modeling

Modeling the dynamics of $(SM)^2$ is important for both studying dynamic characteristics of the robot and designing controllers based on the model. To simplify the model to be realistic in implementation, the robot is modeled as a lumped-mass model. The dynamic model is derived based on Lagrange dynamics, and transformations from rigid and flexible coordinate frames are defined.

Considerable attention has been directed to flexible arm modeling and control [2]. However, most researches have been limited to the robots with 2 or fewer degrees-of-freedom. Moreover, most efforts have been directed to the cases of either very flexible link and no (or very little) mass on the tip [1], or to relatively rigid robots [6]. For the first case, a distributed mass model is acceptable, while for the second case, the finite element or small perturbation method based on a semi-rigid model will be more appropriate. The $(SM)^2$ system, however, presents significant flexibility on structure, and at the same time also possesses substantial mass on the joint and the tip due to the mass of the end-effector and the object being manipulated. Moreover, a general 3-D motion, instead of 2-D, is controlled during walking on the trusswork. In this case, a lumped-mass model provides a simple solution with reasonable accuracy.

To obtain the model, the following assumptions are made: (1) the middle joint (*elbow*) and the robot tip (i.e., a combination of the node-gripper and two joints) are lumped as point masses. High reduction gearing at joints is considered in the moment of inertia of joints; (2) the mass and the moment of inertia of the light-weight links are neglected; (3) the tensile and shear distortion are neglected, and only compliance in the bending and torsional directions of links is considered; (4) the deflections for links are small with respect to the length of the link; (5) the gravity effect is not taken into account for space applications in zero-gravity environment.

Based on these assumptions, a simple model is shown in Fig. 17. Robot kinematics are obtained in terms of angles of three joints and deflections of two links, using two types of the coordinates frames, rigid coordinate frame (RCF) and flexible coordinate frame (FCF). The RCF is a common representation of rigid robot motion using the D-H notations. The FCF is located at the end of the link with deflection to represent the deflection of a single link. The kinematics of the robot can be obtained by coordinate frames transformations using FCF and RCF. The detailed discussion can be found in the paper [10], [12]. Based on Lagrange dynamics and the kinematics representation, we may derive the dynamic equation as follows [12].

$$\boldsymbol{M}(\mathrm{q})\ddot{\boldsymbol{q}} + \boldsymbol{K}(\mathrm{q})\mathrm{q} + \boldsymbol{C}(\mathrm{q}, \dot{\boldsymbol{q}}) = \boldsymbol{f} \tag{1}$$

where

$$\begin{aligned}\boldsymbol{q} &= [q_1 \cdots q_{13}]^T \\ &= [\theta_1\ \theta_2\ v_{y_2}\ v_{z_2}\ \phi_{x_2}\ \phi_{y_2}\ \phi_{z_2}\ \theta_3\ v_{y_3}\ v_{z_3}\ \phi_{x_3}\ \phi_{y_3}\ \phi_{z_3}]^T \\ \boldsymbol{f} &= [f_1 \cdots f_{13}]^T \\ &= [\tau_1\ \tau_2\ 0 \cdots \cdots \cdots 0\ \tau_3\ 0 \cdots \cdots \cdots 0]^T\end{aligned}$$

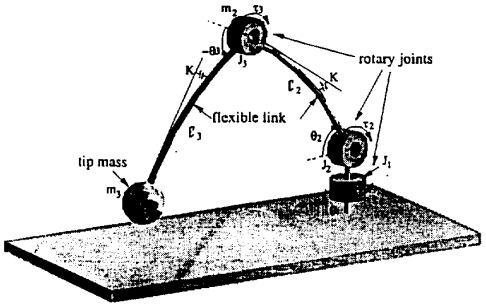


Fig. 17. Simplified dynamic model of the $(SM)^2$

where $\mathbf{q}$ is a vector of generalized position including joint angle and link deflections, and $\mathbf{f}$ is a vector of joint torques and zero elements. The dynamic equation above is nonlinear due to the translational kinetic energy term in Lagrange equation. We further simplify the dynamic equation by neglecting centrifugal and Coriolis effects due to low speed, and the effect of deflection on the inertia matrix. In this way, a linear dynamic model can be obtained.

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{K}\mathrm{q} = \boldsymbol{f} \tag{2}$$

For further frequency analysis, the modal frequencies of the model are needed. However, in the model obtained above, the generalized displacement variables are not independent, so the eigen-matrix can not be inverted. This problem arises from the fact that we do not consider the complete set of variables that is required to represent the model discussed previously. For example, only considering lumped mass at the tip and ignoring its resultant moment of inertia implies dependency between the angular deflection and bending deflection. Reconsidering these facts, we introduce a set of new variables in order to retain the dynamic model linear while making variables independent. The corresponding dynamic equation is given below, where the first and second equations describe the tangential and radial motion, respectively. The parameters and variables are normalized for a compact form. The detailed derivation of each term can be found in our paper [12].

$$\bar{\boldsymbol{M}}_t\bar{\boldsymbol{q}}_t'' + \bar{\boldsymbol{K}}_t\bar{\boldsymbol{q}}_t = \bar{\boldsymbol{\tau}}_t \tag{3}$$

$$\begin{aligned}\bar{\boldsymbol{q}}_t &= [\bar{\theta}_1 \quad \bar{v}_{z_2} \quad \bar{v}_{z_3}]^T \\ &= [\theta_1 \quad \tfrac{v_{z_2}}{\ell_3} \quad \tfrac{1}{\ell_3}(v_{z_3} + \ell_3 \sin\theta_3\phi_{x_2} + \ell_3\cos\theta_3\phi_{y_2})]^T \\ \bar{\boldsymbol{\tau}}_t &= [\bar{\tau}_1 \quad 0 \quad 0]^T \\ &= \left[\tfrac{\tau_1}{K_{b_3}\ell_3^2} \quad 0 \quad 0\right]^T\end{aligned}$$

$$\bar{\boldsymbol{M}}_r\bar{\boldsymbol{q}}_r'' + \bar{\boldsymbol{K}}_r\bar{\boldsymbol{q}}_r = \bar{\boldsymbol{\tau}}_r \tag{4}$$

$$\begin{aligned}\bar{\boldsymbol{q}}_r &= [\bar{\theta}_2 \quad \bar{v}_{y_2} \quad \bar{\theta}_3 \quad \bar{v}_{y_3}]^T \\ &= \left[\theta_2 \quad \tfrac{v_2}{\ell_3} \quad \theta_3 \quad \tfrac{v_3 + \ell_3\phi_{z_2}}{\ell_3}\right]^T \\ \bar{\boldsymbol{\tau}}_r &= [\bar{\tau}_2 \quad 0 \quad \bar{\tau}_3 \quad 0]^T \\ &= \left[\tfrac{\tau_2}{K_{b_3}\ell_3^2} \quad 0 \quad \tfrac{\tau_3}{K_{b_3}\ell_3^2} \quad 0\right]^T\end{aligned}$$

TABLE I
ROBOT MODEL PARAMETERS

j_1	:	normalized joint 1 inertia (gear ratio 60:1)	0.01342
j_2	:	normalized joint 2 inertia (gear ratio 100:1)	0.03728
j_3	:	normalized joint 3 inertia (gear ratio 60:1)	0.01342
m	:	normalized mass 2	0.3846
ℓ	:	normalized length of link 2	1.000
k_t	:	normalized torsional stiffness on link 2	0.750
k	:	normalized bending stiffness on link 2	1.000
ω	:	normalized time	7.631

TABLE II
MODAL FREQUENCIES OF TANGENTIAL MOTION FROM SIMULATION

θ_2 [deg]	0	30	45	60	75
θ_3 [deg]	0	60	90	120	150
mode 1 [rad/sec]	0.00	0.00	0.00	0.00	0.00
mode 2 [rad/sec]	3.98	2.79	2.36	2.14	2.02
mode 3 [rad/sec]	16.6	13.3	11.8	10.0	7.4

It is noted that, in the above model, no centrifugal and Coriolis force is considered. This is because the robot motion is not so fast and these nonlinear terms are not significant. We found that joint friction and damping effects are more significant, and thus a complete dynamic model should include friction and damping effects. However, it is difficult to model them, because they are dependent on velocity, robot payload, and configurations. Recently, we used neural-networks to identify the robot dynamic model and the results were greatly improved. The detailed discussion on the neural network approach can be found in our paper [7].

IX. MODAL FREQUENCIES

When the robot moves within its workspace, the dynamic effects vary significantly with robot different configurations. This can be shown from the modal characteristics of the system, based on the model developed. The modal parameters listed in Table I are computed from the laboratory robot and used in computing the modal frequencies. For simplicity, we normalize all parameters so that they are all dimensionless.

The modal frequencies have been computed for both tangential and radial motions. The configuration of the robot is selected as being the regular configuration with the tip on the plane $Z_B = 0$. The tangential motion has three modes of natural frequencies as shown in Table II. The first mode ($f_1 = 0$) corresponds to the rigid mode. The first non-zero modal frequency decreases with the increase of third joint angle. Table III shows four modes of the vibration for the radial motion. The lowest non-zero modal frequency decreases slightly as the angle of the third joint increases.

We may see that the modal frequencies varies significantly with the change of robot configuration. The detailed discussion about the model dependency on the configuration can be found in our report [10].

TABLE III
MODAL FREQUENCIES OF RADIAL MOTION FROM SIMULATION

θ_2 [deg]	0	30	45	60	75
θ_3 [deg]	0	60	90	120	150
mode 1 [rad/sec]	0.00	0.00	0.00	0.00	0.00
mode 2 [rad/sec]	0.00	0.00	0.00	0.00	0.00
mode 3 [rad/sec]	8.25	8.25	8.26	8.30	8.38
mode 4 [rad/sec]	13.7	12.7	12.6	12.5	12.5

TABLE IV
COMPARISON BETWEEN EXPERIMENTAL AND SIMULATIONAL RESULTS

θ_2	56.27		111.82		77.64		90.0	
θ_3	62.20		36.23		25.95		0.00	
	Exp.	Sim.	Exp.	Sim.	Exp.	Sim.	Exp.	Sim.
f_1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
f_2	2.22	2.81	1.98	2.52	2.96	3.59	3.29	3.98
f_3	10.3	12.8	9.06	15.4	8.31	15.5	14.4	16.6
f_4	13.6	——	15.4	——	14.3	——	——	——

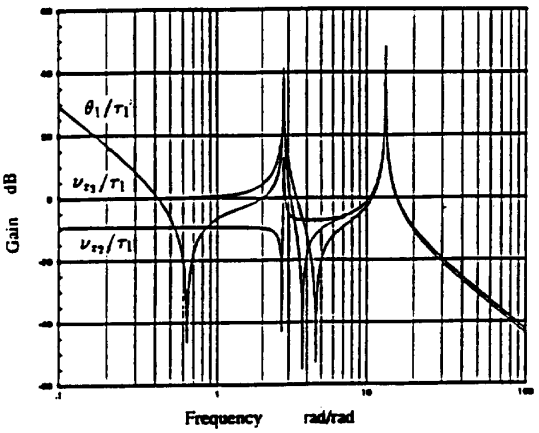


Fig. 18. Bode plot of the transfer functions in a tangential motion.

An experiment has been performed to verify the analytical model. By sweeping the sinusoid torque from low to high frequency, the resonant frequencies as the local peak of the vibration sensor readings are observed. In order to obtain the modal frequencies of the tangential motion experimentally, we apply the torque to the first joint and measure the elbow and tip accelerations in the tangential direction, while locking the rest of joints. The comparison between the simulation and the experimental results are listed in Table IV. For the tangential motion, the modal frequencies computed are slightly larger than that from experiments, and the ratio of the error is consistent for each case. Because the moment of inertia of the last three joints is not taken into account, the experiments have shown four modes in this frequency range, while only three modes appeared in the simulation.

We have numerically obtained the transfer function of the linearized dynamic system at each motion. Fig. 18 and Figs. 19 and 20 show the transfer functions of the joint torques to the tangential and radial motions in Bode-plot at certain configurations, respectively. Within the lower frequency ranges, the gain of the joint variables has the second order

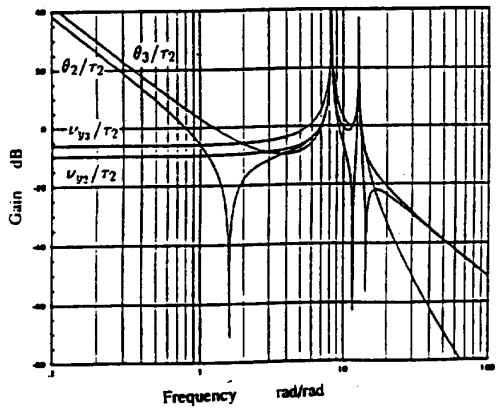


Fig. 19. Bode plot of the transfer functions in radial motion.

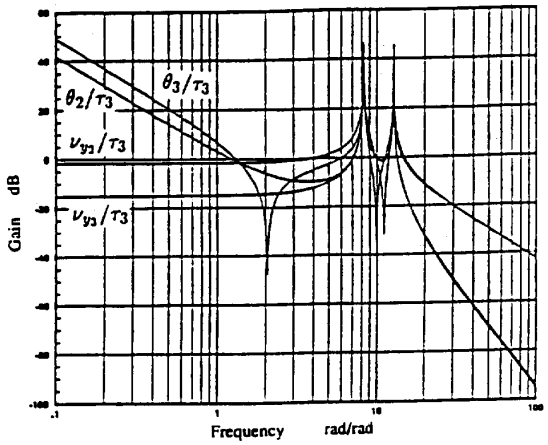


Fig. 20. Bode plot of the transfer functions in radial motion.

slope, while the gain of the deflection variables is nearly constant.

The experimental Bode-plot in the tangential motion has been obtained at a certain configuration. The sinusoidal torque on the first joint of the robot is applied as the input, while the elbow acceleration in the tangential direction is measured as the output. The elbow acceleration is selected so that the first mode of the vibration is significantly visible. By measuring the magnitudes of the input and output at each frequency, the Bode-plot of the elbow acceleration with respect to the first joint torque is obtained as shown in Fig. 21. A sufficient magnitude of the input torque is selected so that the stiction in the joint is negligible in comparison to the input. The experimental results well matched the simulation results.

X. Model-Based Control Using Acceleration Feedback

We are developing a model-based control scheme to improve the system performance in a fast motion for use as a coarse motion controller in the multi-phase step motion. The feedback controller is designed based on the dynamic model derived previously.

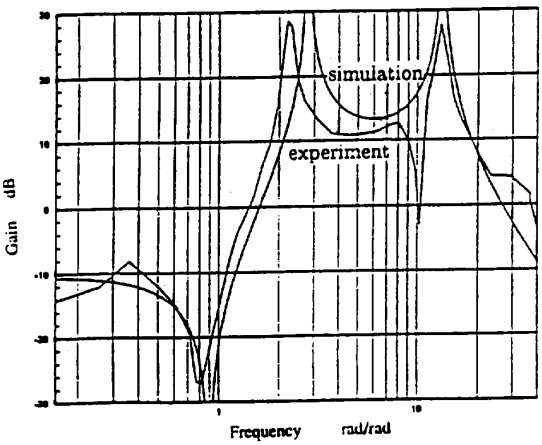


Fig. 21. Bode plot of the transfer functions from the elbow acceleration to the first joint angle.

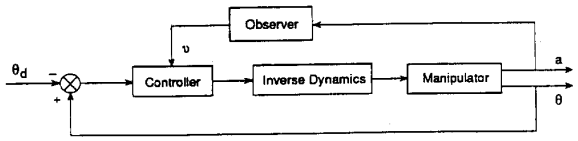


Fig. 22. Block diagram of model-based acceleration feedback control.

$$M\ddot{\mathbf{q}} + K\mathbf{q} = \mathbf{f}$$

where M and K are both configuration dependent inertia and stiffness matrices, $\mathbf{q}$ is a generalized coordinate vector including joint angle and normalized deflections, $\mathbf{f}$ is composed of the normalized joint torque and zero elements. The controller is designed such that both tracking error $\Delta\theta$ and the deflection ν are stabilized, i.e.,

$$U = K_p\Delta\theta + K_v\Delta\dot{\theta} + K_d\nu$$

where K_p, K_v, and K_d are gain matrices. By using the dynamics relation between the deflection and the acceleration at the end of each link, we can obtain

$$a = S_1\nu + S_2\dot{\nu}$$

where a is an acceleration vector measured at the end of links, and S_i is a normalized coefficient matrix related to damping or stiffness. Assuming the damping is negligible, i.e., $S_2 = 0$,

$$a = S_1\nu.$$

When the structural damping is not negligible, using the standard procedure suggested in [5], the observers of ν and $\dot{\nu}$ can also be obtained using acceleration measurement. A block diagram of the control scheme is shown in Fig. 22. The simulation has shown that the vibration is reduced effectively by 80-95% using the model-based control. The control scheme is being implemented.

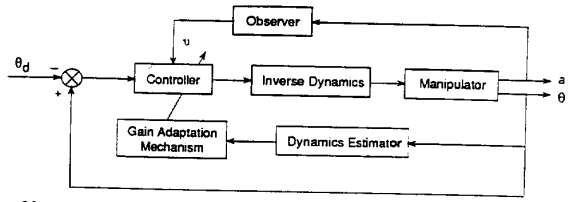


Fig. 23. Block diagram of configuration independent control.

XI. Configuration-Independent Control Scheme

It has been shown that the inertia and stiffness matrices are all configuration dependent. Although the inertia and stiffness matrices are updated by the joint measurement and the observer, the control parameters remain constant. As we have seen, when the robot moves in any configuration, the dynamic effects vary significantly. Therefore, it is desirable to allow the parameters in the controller to adapt to dynamics variation due to the robot configuration change.

The configuration-independent controller uses the estimated dynamics and measured configuration to determine the optimal gains of the controller based on the predefined criterion. For simplicity, we use joint measurement to approximately estimate the dynamics and then compute the modal frequencies of the system. Using the modal frequencies of the current system as an input, the gains of the controller as well as the orders and cut off frequencies of the filters are calculated by maintaining the desired performance identical at all times. The block diagram is shown in Fig. 23.

As the simplest case, we update linear control and filter parameters by evaluating the first modal frequency during the motion and keep a certain relationship between the modal frequency and updating parameters. For example, we use the following relationship to determine the filter frequency:

$$\omega_f = \omega_p \frac{1}{\sqrt{(M_p^{(2/N)} - 1)}} \tag{5}$$

and

$$\omega_p = \sqrt{(1 - 2\zeta_h^2)}\omega_h \tag{6}$$

$$M_p = \frac{1}{2\zeta_h\sqrt{1 - \zeta_h^2}} \tag{7}$$

where ω_f is the filter frequency, N is the order of filter, ω_p is the peak frequency from measurement, M_p is the peak magnitude, ω_h is the modal frequency, and ζ_h is the modal damping ratio. Then, the parameters of feedback controllers are determined by considering filter dynamics to maximize the system bandwidth, i.e., effective closed-loop frequency. Usually, the closed-loop frequencies are set to 1/6-1/8 of the filter frequency.

XII. Summary

We have developed a simple, 5-DOF, 1/3 scale, laboratory version of a robot designed to walk on the trusswork of Space Station Freedom or other space structures. It will be capable of routine tasks such as inspection, parts transportation and simple maintenance procedures. We have designed and built the robot and gravity compensation system to permit simulated zero-gravity experiments. We have developed servo-controls for locomotion movements of the 3-D highly flexible robot and are a developing prototype control station with various operator interfaces. A neural-network-based vision system has been implemented to aid in the precise positioning required for truss node attachment. The $(SM)^2$ has demonstrated its ability to walk reliably to any unobstructed external area of the truss.

The $(SM)^2$ control system includes: the hierarchical control structure which allows control to be executed in various levels autonomously or by teleoperation, the multi-phase control strategy which facilitates the control in different tasks, and various robust low-level controllers. We have developed a dynamic model which has been verified by experiments and is being used for model-based control. The configuration-independent control scheme allows the control parameters to adapt to the dynamics changes due to the robot configuration variation. With these robust control schemes, the robot performance is greatly improved when the system is subjected to the time-variant or unmodeled dynamics.

Acknowledgment

The authors thank Hiroshi Ueno, Miyuki Ueno, Todd Newton, Nina Zumel, Randy Casciola, Tetsuji Yoshida, John Dolan, Dean Pomerleau, Ju-Jang Lee, and David Stewart for their technical support and contributions. Some figures were originally prepared by Mary Jo Dowling of Carnegie Mellon University, for the 1990 Annual Research Review of the Robotics Institute.

References

[1] W. J. Book, "Analysis of massless elastic chains with servo controlled joints," *J. Dyn. Syst., Meas., and Contr.*, vol. 101, pp. 187–192, Sept. 1979.

[2] W. J. Book, "Modeling, design and control of flexible manipulator arms: status and trends," *Proc. NASA Conf. on Space Telerobotics*, 1989.

[3] B. Brown, M. Friedman, T. Kanade, and Y. Xu, "Self-mobile space manipulator project," *The Robotics Institute Annual Research Review*, 1990.

[4] J. Dolan, M. Friedman, N. Nagurga, and J. Gotow, "Gestural control of industrial robots," *Adv. Topics in Mfg. Tech.: Product Des., Bioengineering, and Space Commercialization*, 1987.

[5] D. G. Luenberger, "An introduction to observers," *IEEE Trans. Automat. Contr.*, vol. 16, pp. 596–602, 1971.

[6] F. Matsuno and Y. Sakawa, "A simple model of lfexible manipulators with axes and vibration control by using accelerometers," *J. Robotic syst.*, vol. 7, no. 4, pp. 575–597, 1990.

[7] R. T. Newton and Y. Xu, "Neural network control of a space manipulator," *IEEE Control Syst.* vol. 13, no. 6, 1993.

[8] D. Pomerleau, "Neural network based vision for precise control of a walking robot," *Machine Learning*, 1990.

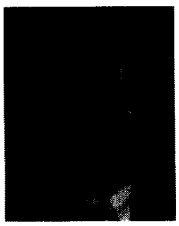
[9] D. Stewart, D. Schmitz, and P. Khosla, "Implementing real-time robotic systems using Chimera ii," *Proc. IEEE Conf. Syst. Engg.*, 1990.

[10] H. Ueno and Y. Xu, "Modeling of a 3–d lightweight space manipulator," Tech. Rep. CMU-RI-TR-91–08, The Robotics Institute, Carnegie Mellon University, 1991.

[11] H. Ueno, Y. Xu, B. Brown, M. Ueno, and T. Kanade, "On control and planning of a space station robot walker," *Proc. IEEE Conf. Syst. Engg.*, 1990

[12] H. Ueno, Y. Xu, and T. Yoshida, "Modeling and control of a 3–d flexible space robot," *Proc. Int. Conf. on Intelligent Robots and Syst.*, 1991

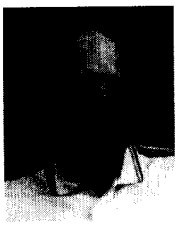
[13] M. Ueno, W. Ross, and M. Friedman, "Torcs: a teleoperated robot control system for the self mobile space manipulator," Tech. Rep. CMU-RI-TR-91–07, The Robotics Institute, Carnegie Mellon University, 1991.



Yangsheng Xu (S'89–M'90) received the Ph.D. degree from the University of Pennsylvania in 1989.

From 1987 to 1989, he was research assistant and post doctoral research associate in the GRASP laboratory at the University of Pennsylvania where he developed a robotic compliant wrist system and hybrid position/force control algorithms. Since November 1989, he has been working with the Robotics Institute of Carnegie Mellon University where he is currently a Research Scientist. At CMU, he has been leading and conducting research in several projects on space robotics and robotic manufacturing, including Self-Mobile Space Manipulator (SM^2) as a principal investigator. His current interests include dynamics and control problems in space and dynamically-stable robot systems, human and robot coordination, and learning control.

He has published more than 70 papers on robot control, space robotics, teleoperation, and learning control.



H. Benjamin Brown, Jr. received the M.S. degree in mechanical engineering at Carnegie Mellon University in 1976.

He is Project Scientist in the Robotics Institute at Carnegie Mellon University. He is interested in the analysis and design of mechanisms and electromechanical systems, and specializes in the development of high-performance structures and equipment. Current research projects relate to the development of novel robots for space and lunar applications, and to the development of various sensors. His previous research was in mechanical design and development of legged, running robots.

Mark Friedman received the B.S. degree in biology in 1966 from Massachusetts Institute of Technology, Cambridge, MA and the Ph.D. degree in neurobiology in 1972 from Rutgers University, New Brunswick, NJ.

From 1972 through 1974 he was a University Fellow doing post-doctoral research at the University of Edinburgh, Scotland. In 1974 he joined the psychology faculty of Carnegie Mellon University. After developing an interest in rehabilitative technology through volunteer work, he became a Research Engineer at CMU's Robotics Institute in 1980. With several students, he co-founded Sentient Systems Technology, Inc. to commercialize technology conceived in the Technical Volunteer Program at The Rehabilitation Institute of Pittsburgh, PA. This company has become the leading producer of eye-gaze controlled computers and one of the leaders for augmentative communication devices. In 1990, he formed a consulting engineering company, AugmenTech, Inc., which does work in cognitive prosthetics and he also teaches as an Adjunct Professor in CMU's Biomedical Engineering Program.

Takeo Kanade (M'80–SM'88–F'92) received the Ph.D. in electrical engineering from Kyoto University, Japan, in 1974.

He is the U. A. and Helen Whitaker Professor of Computer Science and Director of the Robotics Institute at Carnegie Mellon University, Pittsburgh, PA. Before joining Carnegie Mellon in 1980, he was Associated Professor at the Department of Information Science, Kyoto University, Japan. At Carnegie Mellon, he has led and has been successfully leading many major robotics projects as the principal or co-principal investigator funded by ARPA, NASA, NFS, DOE, and industries. For education in robotics, he was a funding Chairman (1989–93) of the Robotics Ph.D. Program at CMU. He has made technical contributions in multiple areas of robotics: vision, manipulators, autonomous mobile robots, and sensors. His contributions in vision include: shape recovery from line drawings (known as Origami World theory and skew symmetry), stereo, color, and face recognition. He is the co-developer of the concept of direct-drive manipulators and the world's first prototype (CMU DD Arm I)—now on exhibit at Boston's Computer Museum. In the area of autonomous mobile robots, he has been the leader and developer of vision systems for Carnegie Mellon's NavLab. Also, he holds patents for a few unique 3D sensors which he has developed.

He has written two books and more than eighty journal papers on these research topics. He is a Fellow of the IEEE, a Fellow of the American Association of Artificial Intelligence, the founding Editor of the *International Journal of Computer Vision*, and an Administrative Committee member of the IEEE Robotics and Automation Society. He has received several awards, including the Memorial Paper Award from the 25th Year of Information Processing Society of Japan, the Marr Prize Award in 1990, and selection in 1992 as one of the most influential papers that appeared in *Artificial Intelligence* journal in past ten years. He has served on many government, industry, and university advisory or consultant committee including Aeronautics and Space Engineering Board (ASEB) of National Research Council, NASA's Advanced Technology Advisory Committee (Congressionally mandate committee) and Advisory Board of Canadian Institute for Advanced Research.