N88-14876

52-37
116665
TQ 124807 20P

DYNAMICS, CONTROL AND SENSOR ISSUES PERTINENT TO ROBOTIC HANDS FOR THE EVA RETRIEVER SYSTEM

Final Report

NASA/ASEE Summer Faculty Fellowship Program--1987

Johnson Space Center

Prepared by: Robert A. McLauchlan, Ph.D.,P.E.

Academic Rank: Associate Professor

University & Department: Texas A & I University
Civil & Mechanical Engineering
Department
Kingsville, Texas 78363

NASA/JSC

Directorate: Engineering

Division: Crew & Thermal Systems

Branch: Engineering Crew (EC2, EC5)

JSC Colleagues: Cliff Hess, EC5
Paul Marshall, EC2

Date: September 3, 1987

Contract: NGT 44-001-800

ABSTRACT

Basic dynamics, sensor, control and related artificial intelligence issues pertinent to smart robotic hands for the Extra Vehicular Activity (EVA) Retriever system are summarized and discussed. These smart hands are to be used as end effectors on arms attached to manned maneuvering units (MMU). The Retriever robotic systems comprised of MMU, arm and smart hands, are being developed to aid crewmen in the performance of routine EVA tasks including tool, object, etc. retrieval. The ultimate goal of this work is to enhance the effectiveness of EVA crewmen.

INTRODUCTION

Overview

Manned Extra Vehicular Activity (EVA) is an important part of planned orbital Space Station development and operations as, e.g.,

* Initial assembly, on-going maintenance of the Space Station.
* Maintenance and servicing - as per refueling and repair of satellites, platforms, and free-flyers.

Manned EVA is also available to be used as backup to telerobotic systems and for unplanned or unique operations.

Here EVA safety needs require a man-in-a-suit retrieval capability. More routine needs for such things as tool and other item retrieval, and assistance with workpiece/object - handling and maneuverability during manned-maintenance and construction tasks, also require at least a low-level, dextrous and intelligent robotic assistant capability.

Retriever Rationale and Background

More specifically, during EVA operations the potential exists for an EVA crewman or piece of equipment to separate from or inadvertently be launched away from the Space Station. Equipment failure or procedural error may cause such separation. Proposed Space Station EVA procedures require that there be at least two crewmen present for any EVA. However, it may not be practical because of safety considerations or time constraints for the second crewman to (a) Properly equip himself, (b) Leave the Space Station, and (c) Retrieve the lost equipment or rescue the first crewman who may be helpless.

Figure 1 from [Ref. 1] plots range versus range rate of separation for an object which is drifting free from the Space Station. The initial velocity for the drifting object is 2 ft/sec in the case depicted. Here estimates of the position and drift rate of the object are indicated at the discrete data points shown.

Table 1 summarizes these estimates of position and drift rate as functions of time. It indicates that at ten minutes from separation the object will already be 1400 feet away and have a rate of separation of 3.2 ft/sec. These values are 2350 ft and 5.3 ft/sec ,respectively at fifteen minutes from separation. The results summarized in

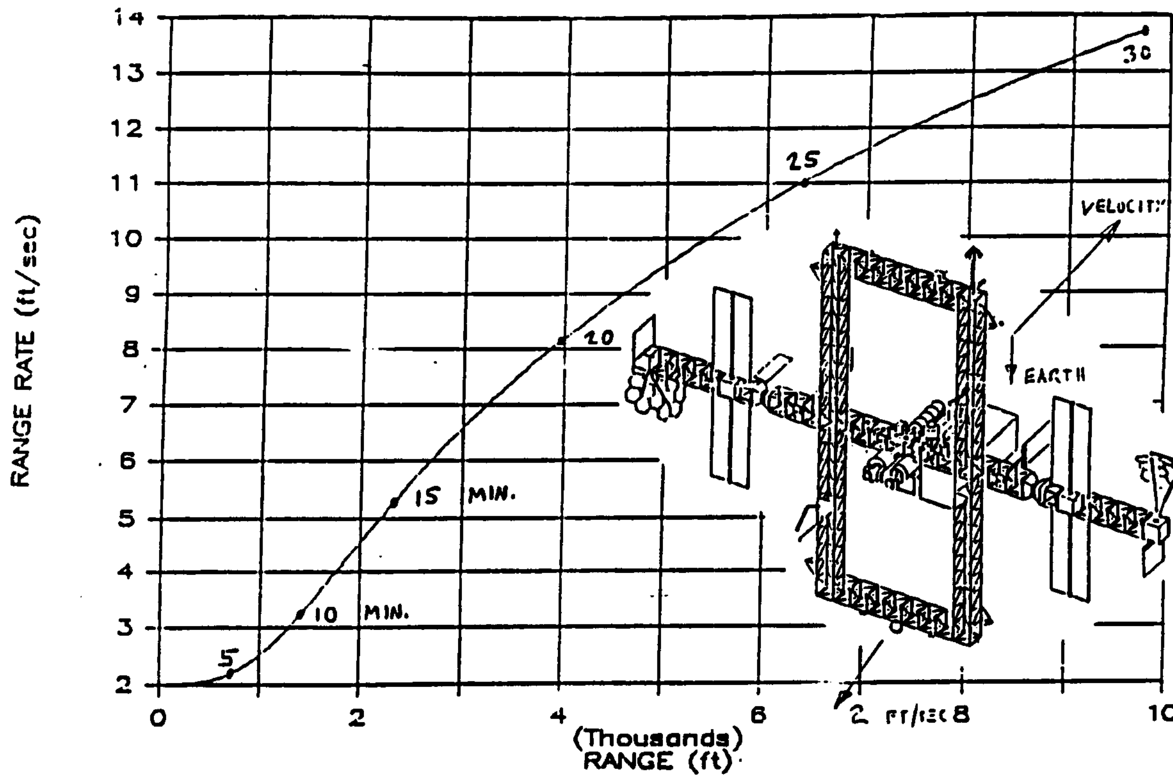
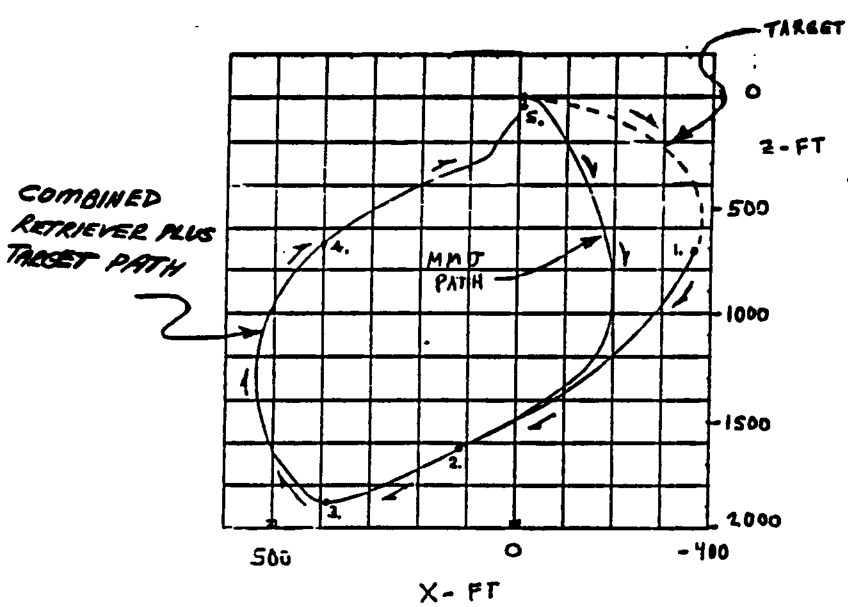


FIGURE 1 RANGE VERSUS RANGE RATE OF SEPARATION FOR AN OBJECT DRIFTING FREE FROM THE SPACE STATION



POINT NUMBER I	DRIFT RATE R (FPS)	TIME T (MIN)
1.	1.42	10.0
2.	2.65	17.2
3.	-.17	19.1
4.	-2.98	25.2
5.	-.26	31.0

SUMMARY OF RETRIEVAL STAGES

* APPROACH T_1 to T_2
* CAPTURE T_2 to T_{3-}
* GRASP T_{3-} to T_{3+}
* ADJUSTMENT AND RETURN T_{3+} to T_5

FIGURE 2 REPRESENTATIVE TRAJECTORY FOR RETRIEVAL SCENARIO

Table 1. Estimated Range and Range Rate as Functions of Time

Time T, Min	Range R, Kft	Range Rate R, ft/sec
5	0.07	2.2
10	1.40	3.2
15	2.35	5.3
20	3.96	8.2
25	6.39	11.0
30	9.67	13.7

Table 1 thus emphasize that time is extremely critical in initiating any retrieval operation. This is especially true if elapsed time and fuel costs are to be routinely minimized and indeed successful retrieval be assured.

Figure 2 depicts a representative trajectory for a retrieval scenario with an astronaut piloted Manned Maneuvering Unit (MMU) chasing a target. Here drift rate and time at five discrete times in the range from 10 to 31 minutes. Figure 2 also illustrates the urgency of initiating rescue as quickly as possible after separation.

JSC has developed the concept of the EVA Retriever, which is a highly autonomous, free-flying robot or robotic system. In use it will be on standby during EVA operations to provide the immediate chase, capture, and return capability required for adrift crewmen or station equipment. The Retriever will be used to fill the Space Station requirement which has been authorized by Space Station Control Board Directive BB000169A [Ref. 1]. The conceptual design of the EVA Retriever is depicted in Figure 3. By way of summary, Ref.1 indicates that the Retriever is:

* Intended to be highly autonomous
* Capable of locating and tracking the target
* Able to plan a path to the target and to avoid obstacles encountered during the chase, etc. segment
* Able to grapple - i.e., capture and grasp, etc. the target through the use of dextrous arms and hands
* Intended to interface with the Space Station MMU - thereby providing commonality with the crew interface
* Activated and supervised by voice command.

Table 2 gives a descriptive summary of the attributes of the smart hand plus arm plus MMU - EVA Retriever robotic system. Here 14 attributes ranging from principal function, environment, control architecture, etc. through sensors, computer processing, construction, and power are considered. These autonomous EVA Retriever attributes are somewhat different from those of a space telerobotic system. It should be noted that this system is directed toward the true leveraging of an astronaut crewman during EVA. Here an analogy can be drawn to a shepherd with his dog(s) working a flock. A telerobotic system on the other hand will occupy at least one astronaut crewman per system at a maximum work efficiency of 50 % of one EVA crewman.

Research Work Description

The work reported herein is part of an overall effort to investigate the intelligent control of robotic hands on the EVA Retriever. It has involved the exploration of basic dynamics, sensor,

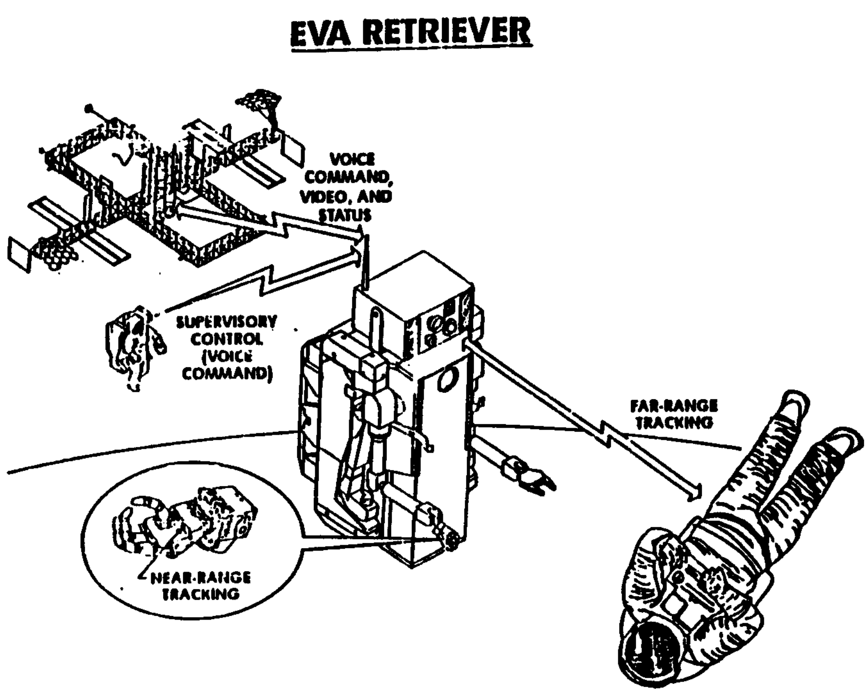


FIGURE 3 CONCEPTUAL DESIGN OF THE EVA RETRIEVER

TABLE 2, SMART HAND, ARM, MMU RETRIEVER SYSTEM ATTRIBUTES

Attribute	Description
1. Principal Function	Repetitive and/or Hazardous EVA Operations; Tool, Crewmen Retrieval.
2. Environment	Complex, Time Limited, Often Hazardous.
3. Primary Control Architecture, Parameters	Parallel Time, Hierarchical Task Accomplishment; Deep Knowledge; Position, Rate, and Force.
4. Operating Speed	Task Dependent, Subject to Human, Robot, and Equipment Safety -- As Fast as Possible.
5. Load Range	5 to 8 Lbm * Hand-Fingers 8 to 20 Lbm * Finger Wrap + Wrist 20 to 250 Lbm * Arm + Finger-lock. * At 1g Acceleration.
6. Kinematics	Arm/Hand Rotary, Prismatic (and Spherical) Joints; Various DOF.
7. End-Effectors	General Purpose Grippers/Fingers
8. Compliance	<u>Hand</u> - Relatively Flexible, Backdriveable, Some Backlash. <u>Arm/Wrist</u> - Relatively Stiff, Ideally Backdriveable, Minimal Backlash.
9. Actuators	<u>Hand</u> - Usually Local or Close to Joint. <u>Arm/Wrist</u> - Local or Close to Joint.
10. Transmission	Arm - Gear Box Drive; Hand - Torque Tubes, Kevlar Cable and Pulley Tendons, FourBar and Related Linkages.
11. Sensors	Vision - Nearfield/Farfield; Tactile - Proximity and Contact/Touch. Joint Torque, Position Sensors as Close to Joint as Possible.
12. Computer Processing Power	Parallel - Transputer/Multi-processor; Significant Pre-Processing of Signals at the Sensor Array(s); Potential Use of Multilayer Neural Net Arrays.
13. Construction	Possibly Composites; Space Grade Aluminum, Steel, and Titanium as Appropriate.
14. Power	Electric; Electromechanical, Piezoelectric.

control, and related Artificial Intelligence issues pertinent to robotic hands. These hands are to be used as end effectors on arms attached to Manned Maneuvering Units (MMU). The Retriever robotic systems comprised of MMU, arm and smart hands, are being developed to aid crewmen in the performance of routine EVA tasks including tool, object, etc. retrieval. The ultimate goal of this work is to enhance the effectiveness of EVA crewmen.

BASIC ISSUES

Smart, Dextrous Robotic Hands

At this point it is important to consider the requirements (or desirable characteristics of typical smart robotic hands for the NASA/JSC EVA Retriever. Figure 4 shows a later generation version (4 articulated fingers - 3 fingers plus a thumb) which the Crew and Thermal Systems Division (CTSD) at NASA/JSC is using. As far as dextrous hands are concerned, this is perhaps the best hand available in the world today. However, its pneumatic servo actuation system is impractical as far as space operations of the EVA Retriever are concerned. CTSD at NASA/JSC has also or will shortly have the Salisbury/JPL hand, the Jameson/Stanford hand, and the Grubbs/CTSD hand. These latter hands each have three articulated fingers. Along with the four-fingered Utah/MIT hand, they are being used by CTSD to explore basic issues relevant the dynamics, sensing, and control of (a) the smart hands per se, and (b) the integrated smart hand plus articulated arm plus MMU - comprising the EVA Retriever robotic system. In addition, related Artificial Intelligence (AI) and computer issues are being investigated.

Table 2 as discussed previously summarizes the overall characteristics of the EVA Retriever. here it is seen that the dextrous hand/wrist/arm system is of great importance to EVA Retriever success. It is important to note that the hand-wrist-arm combinations cover the load range typically encountered by a human arm in a 1 g acceleration field.

Figure 5 shows the six different types of prehension which comprise human hand grasping. These prehensions characterize the basic object grasp or gripping functions which humans use everyday. Here the six grasps can be interpreted as basic operations for smart robotic hands as well.

Table 3. Summary of Manipulation Functions Performed by the Human Hand

1. Trigger Grip
2. Flipping a Switch
3. Transferring Pipe to Grip
4. Using Cutters/Shears
5. Screwing a Pen
6. Rolling a Cylindrical Object
7. Transferring a Pen
8. Typewriting/Using a Keyboard
9. Writing With a Pen

Nine basic manipulation functions performed by the human hand

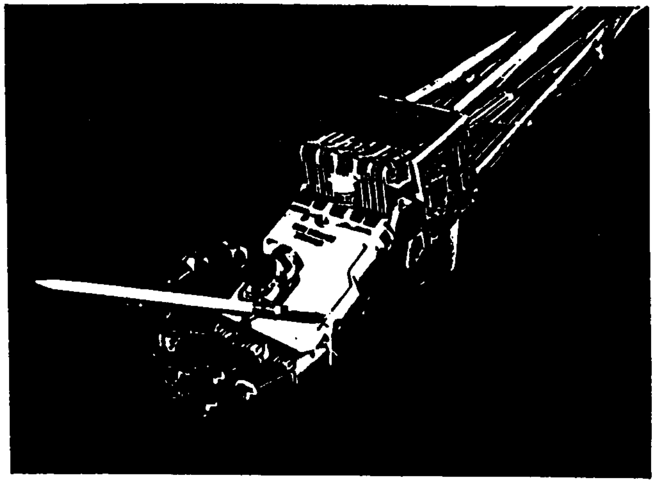


FIGURE 4, Later Generation Version of the UTAH/MIT Dextrous Hand Used by CTSD at NASA/JSC.

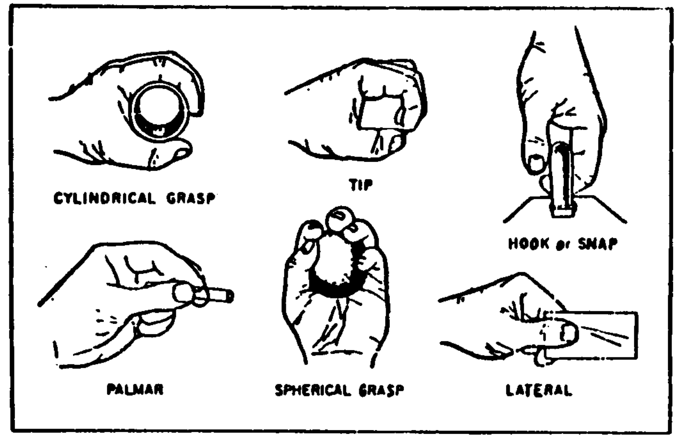


FIGURE 5, SUMMARY OF THE SIX DIFFERENT TYPES OF PREHENSION COMPRISING HUMAN-HAND GRASPING

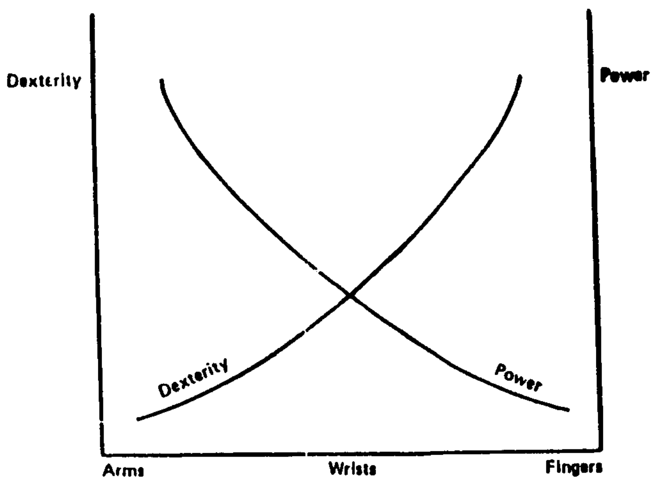


FIGURE 6, REPRESENTATIVE TRADE-OFF BETWEEN ARM, WRIST AND FINGER MANIPULATION. NOTE THAT THE ABSCISSA AXIS HAS UNITS OF APPLIED FORCE (Ffingers < Fwrist < Farms)

Table 4. Summary of the Approximate Relations Between the Number of Fingers and Joints, and the Functions of the Hand/End Effector.

Hand Finger Configurations Number	Type	Functions: Grasping	Shape Accommodation	Manipulation
2	Rigid	Yes	None	None
2	Articulate	Yes	Yes-Some	None
3	Rigid	Yes	Yes-Some	None
3	Articulate	Yes	Yes-Some	Yes-Some
4	Rigid	Yes	Yes-Some	Yes-Some
4	Articulate	Yes	Yes	Yes
5	Rigid	Yes	Yes	Yes-Some
5	Articulate	Yes	Yes	Yes

are summarized in Table 3. Of these manipulation functions, the 'lower level' ones: (1) Trigger grip, (2) Flipping a switch, (3) Rolling a cylindrical object, (4) Transferring pipe to grip, (5) Screwing a pen (or a bolt-nut combination), (6) Using cutters /shears -- are considered by the author as necessary for the EVA Retriever. This is especially true if the Retriever is to be used for assistance with a workpiece/in object handling and thereby, truly enhance the effectiveness of EVA crewmen.

Humans perform grasp and manipulate functions on a routine, everyday basis with their five finger (four fingers plus an opposable thumb) hands. Table 4 indicates that the six grasp and at least six of the manipulate functions can be performed with three or more fingered hands. Here each finger in multiple (three to four revoute) jointed. Figure 6 gives representative plots of dexterity and power as functions of the type of element (finger/hand, wrist, arm) combination or force level applied. These dexterity and power curves are composites of the maximum value portions of the curves for (a) Hand-fingers, (b) Finger wrap plus wrist, and (c) Arm plus finger-lock combinations. The crossover points occur at nominal values of 8, 20, and 250 Lbf (8, 20, and 250 Lbm in 1 g acceleration field). As expected, hand-finger combinations exhibit maximum dexterity at corresponding minimum force and power levels. Arm plus finger-lock combinations on the other hand, give maximum power and force with minimum dexterity.

Table 5. Summary of the Primitive Operations for the Robotic Hand.

1. Movement of the Fingers with Defined Speed
2. Movement of the Fingers With Defined Speed to a Defined Position
3. Grasping With Defined/Specified Force and Torque Limits
4. Movement of the Fingers With Defined Speed to a Defined Position, Combined With the Exertion of a Defined Force
5. Stopping the Fingers at Any Closing Distance. This Could Include Locating the Hand Fingers Concentric With/In Accommodation to - an Object - Using Touch Sensors.
6. Activation of the Sensor System (Tactile and Nontactile)
7. Deactivation of the Sensor System
8. Transfer of the Sensor Data to the Robot Hand Controller

Table 5 summarizes the eight primitive operations required for the robotic hand, which it should be remembered is part of the MMU, arm, hand - EVA Retriever system. Note that

* The first five primitive operations ultimately make closed loop control of (1) Position, (2) Speed, and (3) Grasp force necessary.
* Speed control can be realized via hardware.
* Force and position are controlled by software - which implies the use of dynamics/dynamic models of the robotic hand, arm, MMU system.
* For control of the (grasped/manipulated) object with the robotic hand it is necessary to use both tactile and nontactile sensors.

Dynamics

Dynamics can be considered to be made up of the two subareas:

kinematics and kinetics. As applied to robotic hands and manipulators, kinematics is concerned with the (a) Direct mapping of joint coordinate motion to link Cartesian coordinate motion, and (b) Inverse mapping of link Cartesian coordinate motion to joint coordinate motion. Here the emphasis is on the hand/manipulator motion without regard to the forces and torques causing the motion. Hand/manipulator kinematics involves the position, velocity, and acceleration aspects of the motion in link Cartesian coordinate and joint coordinate spaces.

Kinetics as applied to robotic hands and manipulators is concerned with the time rate of change of the Cartesian configuration of these robotic systems as a function of the applied joint torques and forces. Here the Newton-Euler or the Lagrangian forms of the equations of motion can be used to define the dynamic response of the hand/manipulator linkage system to input joint forces and torques. Once the equations of motion have been developed, the main problem is to determine, i.e., identify those design parameters and force/torque terms which are important in controlling the Cartesian space motion of the hand/manipulator system. Thus, dynamics is important to the design and the control of the dextrous hand plus arm system. It can lead to better nonlinear and linearized models for simulation tools in task planning and control strategy implementation. This is especially true for the multi-fingered hand plus multi-armed EVA Retriever systems. This section of the report will concentrate on the kinetics aspects of dynamics.

Table 6 summarizes a nonlinear - recurrence formulation of the Newton-Euler equations for a hand manipulator system. This is the Luh, Walker, Paul formulation. Table 7 defines the nomenclature used in Table 6. Details can be found in Luh et al 1983, Lathrop 1985, and Murray and Newman 1986. The corresponding linearized - recursive formulation of the Newton-Euler equations is given in Table 8. The nonlinear Newton-Euler dynamic equations can be used to generate feedforward control torques/forces for the hand plus arm system. Depending upon the computer processing power available, the nonlinear dynamics calculations can be done either offline or online. Multiprocessor implementation of the latter is required for robust control in real-time situations.

The linearized - recursive dynamic equation models can be used for (a) Developing optimal/more-optimal robot hand, etc. designs, (b) Online/real-time identification of unknown object inertia and robot drive train, etc. characteristics, and (c) Implementing observer synthesis giving robust adaptive control.

Implementing the dynamic formalisms given above can lead to direct concern with the following bottlenecks - i.e., operation count and memory impediments to real-time dynamic models. The computational load problem of many advanced control schemes lies in developing an efficient inverse dynamics algorithm. Such algorithms can compute the actuator torques/forces required to produce the desired joint accelerations (a la inverse kinematics) for a given set of link Cartesian displacements. Such algorithms must be evaluated by the control computer at least 100 times per second for the hand/manipulator control scheme to be effective.

Here it is important to exploit the reduction of operation

TABLE 6. SUMMARY OF THE RECURSIVE NEWTON-EULER FORMULATION

Forward Iterations for $i = 1, 2, \ldots, N$.
Initialize: $\omega_0 = \dot{\omega}_0 = 0$ and $\dot{\mathbf{v}}_0 = -\mathbf{g}$.

$$\omega_i = \mathbf{R}_i^T[\omega_{i-1} + \sigma_i \mathbf{z}_0 \dot{\theta}_i]$$

$$\dot{\omega}_i = \mathbf{R}_i^T\{\dot{\omega}_{i-1} + \sigma_i[\mathbf{z}_0\ddot{\theta}_i + \omega_{i-1} \times (\mathbf{z}_0\dot{\theta}_i)]\}$$

$$\dot{\mathbf{v}}_i = \mathbf{R}_i^T\{\dot{\mathbf{v}}_{i-1} + (1-\sigma_i)[\mathbf{z}_0\ddot{d}_i + 2\omega_{i-1} \times (\mathbf{z}_0\dot{d}_i)]\} + \dot{\omega}_i \times \mathbf{p}_i + \omega_i \times (\omega_i \times \mathbf{p}_i)$$

Backward Iterations for $i = N, N-1, \ldots, 1$.
Initialize: $\mathbf{f}_{N+1} = \mathbf{n}_{N+1} = 0$.

$$\dot{\mathbf{v}}_{ci} = \dot{\mathbf{v}}_i + \dot{\omega}_i \times \mathbf{r}_i + \omega_i \times (\omega_i \times \mathbf{r}_i)$$

$$\mathbf{F}_i = m_i \dot{\mathbf{v}}_{ci}$$

$$\mathbf{N}_i = \mathbf{I}_{ci}\dot{\omega}_i + \omega_i \times (\mathbf{I}_{ci}\omega_i)$$

$$\mathbf{f}_i = \mathbf{R}_{i+1}[\mathbf{f}_{i+1}] + \mathbf{F}_i$$

$$\mathbf{n}_i = \mathbf{R}_{i+1}[\mathbf{n}_{i+1}] + \mathbf{p}_i \times \mathbf{f}_i + \mathbf{N}_i + \mathbf{r}_i \times \mathbf{F}_i$$

$$\tau_i(t) = \sigma_i \mathbf{n}_i^T(\mathbf{R}_i^T\mathbf{z}_0) + (1-\sigma_i)\mathbf{f}_i^T(\mathbf{R}_i^T\mathbf{z}_0)$$

Table 7 Nomenclature

Scalars

σ_i = Boolean joint type variable: $\sigma_i = 1$ if joint i is revolute; $\sigma_i = 0$ if joint i is prismatic.
m_i = total mass of link i
$\tau_i(t)$ = joint torque/force at joint i

3-Vectors (Referenced to the ith Coordinate Frame)

$\omega_i, \dot{\omega}_i$, and $\dot{\mathbf{v}}_i$ = angular velocity, acceleration, and linear acceleration of the ith coordinate frame
$\dot{\mathbf{v}}_{ci}$ = linear acceleration of the center-of-mass of link i
$\mathbf{F}_i$ and $\mathbf{N}_i$ = net force and moment exerted on link i
$\mathbf{f}_i$ and $\mathbf{n}_i$ = force and moment exerted on link i by link $(i-1)$
$\mathbf{p}_i$ = position of the ith coordinate frame with respect to the $(i-1)$st coordinate frame: $\mathbf{p}_i = [a_i, d_i \sin(\alpha_i), d_i \cos(\alpha_i)]^T$
$\mathbf{r}_i$ = position of the center-of-mass of link i: $\mathbf{r}_i = [r_{ix}, r_{iy}, r_{iz}]^T$
$\mathbf{z}_0$ = $[0\ 0\ 1]^T$

(3 × 3) Matrices

$\mathbf{R}_i$ = orthogonal rotation matrix which transforms a vector in the $(i-1)$st coordinate frame to a coordinate frame which is parallel to the ith coordinate frame:

$$\mathbf{R}_i = \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\sin(\theta_i) & \sin(\alpha_i)\sin(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cos(\theta_i) & -\sin(\alpha_i)\cos(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix}$$

for $i = 1, 2, \ldots, N$, where $\mathbf{R}_{N+1} = \mathbf{I}$.
$\mathbf{I}_{ci}$ = classical inertia tensor [13] of link i about a coordinate frame parallel to the ith coordinate frame and translated to the center-of-mass of link i; with principal inertias I_{cixx}, I_{ciyy}, and I_{cizz}; and cross-inertias I_{cixy}, I_{cixz}, and I_{ciyz}.

TABLE 8 SUMMARY OF THE RECURSIVE LINEARIZED NEWTON-EULER FORMULATION

Forward Iterations for $i = 1, 2, \ldots, N$.
Initialize: $\delta\omega_0 = \delta\dot{\omega}_0 = \delta\dot{\mathbf{v}}_0 = 0$.

$$\delta\omega_i = \mathbf{R}_i^T\{\delta\omega_{i-1} + \sigma_i[\mathbf{z}_0\delta\dot{\theta}_i - \mathbf{Q}(\omega_{i-1} + \mathbf{z}_0\dot{\theta}_i)\delta\theta_i]\}$$

$$\delta\dot{\omega}_i = \mathbf{R}_i^T\{\delta\dot{\omega}_{i-1} - \sigma_i[\mathbf{z}_0\delta\ddot{\theta}_i + \delta\omega_{i-1} \times (\mathbf{z}_0\dot{\theta}_i) + \omega_{i-1} \times (\mathbf{z}_0\delta\dot{\theta}_i)] - \sigma_i\mathbf{Q}[\dot{\omega}_{i-1} + \mathbf{z}_0\ddot{\theta}_i + \omega_{i-1} \times (\mathbf{z}_0\dot{\theta}_i)]\delta\theta_i\}$$

$$\delta\dot{\mathbf{v}}_i = \mathbf{R}_i^T\{\delta\dot{\mathbf{v}}_{i-1} - \sigma_i\mathbf{Q}\dot{\mathbf{v}}_{i-1}\delta\theta_i + (1-\sigma_i)[\mathbf{z}_0\delta\ddot{d}_i + 2\delta\omega_{i-1} \times (\mathbf{z}_0\dot{d}_i) + 2\omega_{i-1} \times (\mathbf{z}_0\delta\dot{d}_i)]\} + \delta\dot{\omega}_i \times \mathbf{p}_i + \delta\omega_i \times (\omega_i \times \mathbf{p}_i) + \omega_i \times (\delta\omega_i \times \mathbf{p}_i) + (1-\sigma_i)(\dot{\omega}_i \times \mathbf{p}_{di} + \omega_i \times (\omega_i \times \mathbf{p}_{di}))\delta d_i$$

Backward Iterations for $i = N, N-1, \ldots, 1$.
Initialize: $\delta\mathbf{f}_{N+1} = \delta\mathbf{n}_{N+1} = 0$.

$$\delta\dot{\mathbf{v}}_{ci} = \delta\dot{\mathbf{v}}_i + \delta\dot{\omega}_i \times \mathbf{r}_i + \delta\omega_i \times (\omega_i \times \mathbf{r}_i) + \omega_i \times (\delta\omega_i \times \mathbf{r}_i)$$

$$\delta\mathbf{F}_i = m_i\delta\dot{\mathbf{v}}_{ci}$$

$$\delta\mathbf{N}_i = \mathbf{I}_{ci}\delta\dot{\omega}_i + \delta\omega_i \times (\mathbf{I}_{ci}\omega_i) + \omega_i \times (\mathbf{I}_{ci}\delta\omega_i)$$

$$\delta\mathbf{f}_i = \mathbf{R}_{i+1}[\delta\mathbf{f}_{i+1}] + \delta\mathbf{F}_i + \mathbf{Q}\mathbf{R}_{i+1}[\mathbf{f}_{i+1}]\delta\theta_{i+1}$$

$$\delta\mathbf{n}_i = \mathbf{R}_{i+1}[\delta\mathbf{n}_{i+1}] + \delta\mathbf{N}_i + \mathbf{p}_i \times \delta\mathbf{f}_i + \mathbf{r}_i \times \delta\mathbf{F}_i + (1-\sigma_i)(\mathbf{p}_{di} \times \mathbf{f}_i)\delta d_i + \sigma_{i+1}\mathbf{Q}\mathbf{R}_{i+1}[\mathbf{n}_{i+1}]\delta\theta_{i+1}$$

$$\delta\tau_i(t) = \sigma_i[\delta\mathbf{n}_i^T(\mathbf{R}_i^T\mathbf{z}_0) - \mathbf{n}_i^T(\mathbf{R}_i^T\mathbf{Q}\mathbf{z}_0)\delta\theta_i] + (1-\sigma_i)[\delta\mathbf{f}_i^T(\mathbf{R}_i^T\mathbf{z}_0)]$$

NOTE THAT THIS LINEARIZED DYNAMIC ROBOT MODEL CAN BE USED FOR REAL TIME DIGITAL SIMULATION - OBSERVER SYNTHESIS TO GENERATE A FEEDBACK SIGNAL THAT REGULATES THE PERTURBATION POSITIONS AND VELOCITIES TO ZERO. IT CAN ALSO BE USED FOR IDENTIFICATION OF THE MANIPULATOR, ETC. PARAMETERS VIA THE LINEARIZED DYNAMIC ROBOT MODEL

TABLE 9 SUMMARY COMPARISON OF DYNAMICS FORMULATION (ADAPTED FROM HOLLERBACH 1980)

	Comparison of Time Dependencies		Comparison for $n = 6$	
Method	Mults.	Addns.	Mults.	Addns.
Uicker/Kahn (original Lagrangian)	$32\frac{1}{2}n^4 + 86\frac{5}{12}n^3 + 171\frac{1}{4}n^2 + 53\frac{1}{3}n - 128$	$25n^4 + 66\frac{1}{3}n^3 + 129\frac{1}{2}n^2 + 42\frac{1}{3}n - 96$	66,271	51,548
Waters (partially recursive)	$106\frac{1}{2}n^2 + 620\frac{1}{2}n - 512$	$82n^2 + 514n - 384$	7,051	5,652
Hollerbach (4 × 4 Lagrangian)	$830n - 592$	$675n - 464$	4,388	3,586
Hollerbach (3 × 3 Lagrangian)	$412n - 277$	$320n - 201$	2,195	1,719
Luh, Walker, Paul (Newton-Euler)	$150n - 48$	$131n - 48$	852	738
Horn, Raibert (table lookup)	$2n^3 + n^2$	$n^3 + n^2 + 2n$	468	264
Luh, Lin (scheduled parallel Newton-Euler)	$57n - 18$ (estimated; see text)	$50n - 18$ (estimated)	323 (estimated)	280 (estimated)
Lathrop (linear parallel Newton-Euler)	$2n + 3$	$6n + 7$	15	43
Lathrop (logarithmic parallel Newton-Euler)	$2\lceil\log_2(n+1)\rceil + 5$	$6\lceil\log_2(n+1)\rceil + 10$	11	28
Lathrop (systolic pipeline)	1 (successive; see text)	3 (successive)	1 (successive)	3 (successive)

Note: This table reflects the algorithmically indicated cost for the fully general six-link rotary manipulator, as in Hollerbach (1980). By consideration of special cases, introduction of configuration or workspace assumptions, or tailoring of the computation, additional reductions are possible.
Tables I & II from J. M. Hollerbach, "A Recursive Formulation of Manipulator Dynamics," *IEEE Trans Systems, Man & Cybernetics*, SMC-10, No. 11, pp. 730–736, have been adapted to include new algorithms since 1980.

count/time of calculation as a function of the number of degrees of the hand/manipulator robotic system. Such reductions arise from the geometry and hierarchical design structure (as, e.g., open tree and constrained loop - connectedness partitioning) of the smart hand plus arm robotic system.

Horak 1984 has considered the exploitation via the arm, wrist/hand partitioning of the relatively simple geometry of arm - manipulator systems having 4 to 7 links. Lathrop 1985 has treated the exploitation of parallelism in manipulator dynamics. Table 9 summarizes his results. It also shows the evolution toward reduced operation (addition, multiplication) and machine cycle time. Here as with the forward/backward recurrence schemes in Tables 6, 8, linear operation counts are possible. However by exploiting the space-time parallelism (a) Order log(NDOF) operations and (b) Systolic 3 vector adds plus 1 vector multiply implementations are also possible. Lathrop's work indicates that such efficiencies obtain for both the nonlinear and the linearized dynamic equation forms.

As far as the control of object motion for manipulators is concerned, Kerr and Roth 1986 have analyzed the inverse dynamics problem for multi-fingered hands. They give a general formulation to evaluate the finger, etc. torques/forces required for the redundant object plus multi-finger linkage system. The author has considered the degree of freedom reductions possible if position and velocity/slip sensors are used with each finger. Details will be available in an extended version of this report.

Sensors and Sensing Systems

Sensors and sensing systems are the beginning elements or componnents in perception. They are of prime importance in performing the observation function - for the intelligent, controlled interaction of a robotic system with its environment. The primary sensors/sensing systems considered for use in general space robotics can be summarized as (1) Vision, (2) Tactile, (3) Proximity, (4) Force, (5) Position, and (6) Velocity.

The smart, dextrous hand plus arm plus MMU - EVA Retriever system needs a variety of sensors. Multi-sensing is necessary in order for the Retriever robotic system to adapt to disturbances and unpredictable changes in (a) Its EVA - space environment, and (b) Itself in interacting with that environment. Here the Retriever robotic system will use (1) External sensors as, e.g., visual, infrared - light, solid-state CCD cameras and touch sensors; and (2) Internal sensors such as optical, magnetic-joint encoders, and Hall-effect based, joint torque/force sensors. In this vein, Table 10 summarizes the sensor environment of the robotic hand, etc system. Figure 7 gives a corresponding representation of a robotic manipulator/hand system which is provided with the senses of feeling (internal sensing) and sight (external sensing). Figure 7a also gives an operational diagram for the (1) Scene recognition, (2) Signal/image processing - programs, and (3) Grasp program for the manipulator hand. Figure 7b depicts the teaching-phase use of the sensed information.

In the smart, dextrous robotic hand plus arm system, sensors can measure triaxial force at the finger/hand contact points with an ob-

TABLE 10. SUMMARY OF THE SENSOR ENVIRONMENT OF THE ROBOTIC HAND.

1. Tactile Sensors As Per:
   - Force-Sensing Wrist
   - Grasp Force Sensor
   - Touch Sensors
2. Nontactile Sensors As Per:
   - Object Approach Sensors (Farfield, Midfield Vision)
   - Object 'Capture' Sensors (Nearfield Position Recognition Sensors, As for Example, Infrared for Determining Object Position Between the Fingers)

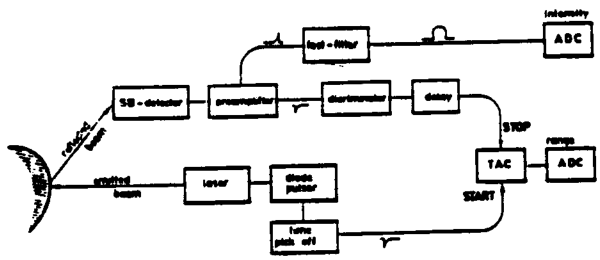


FIGURE 8. REPRESENTATION BLOCK DIAGRAM FOR A TIME-OF-LIGHT LASER SCANNER

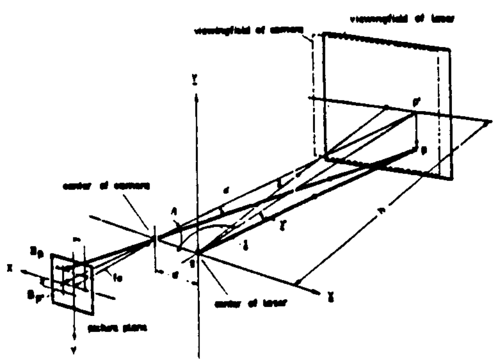


FIGURE 9. BASIC GEOMETRY OF A TRIANGULATION TYPE LASER SCANNER

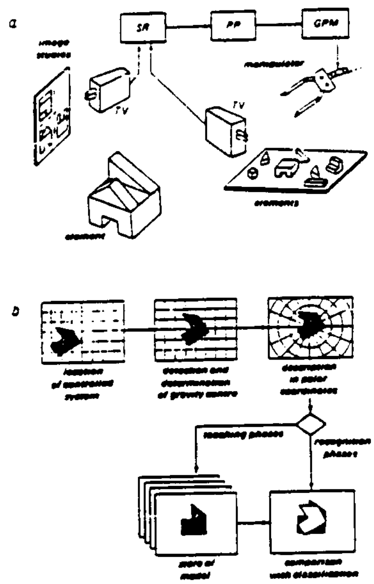


FIGURE 7, REPRESENTATION OF ROBOTIC MANIPULATOR/HAND PROVIDED WITH SENSES OF FEELING (INTERNAL SENSING) AND SIGHT (EXTERNAL SENCING). FIGURE a GIVES THE OPERATIONAL DIAGRAM WITH (1) SR= SCENSE RECOGNITION, (2) PP= PROCESSING PROGRAMS, AND (3) GPM= GROUP PROGRAM FOR THE MANIPULATOR HAND; FIGURE b DEPICTS THE TEACHING PROCESS.

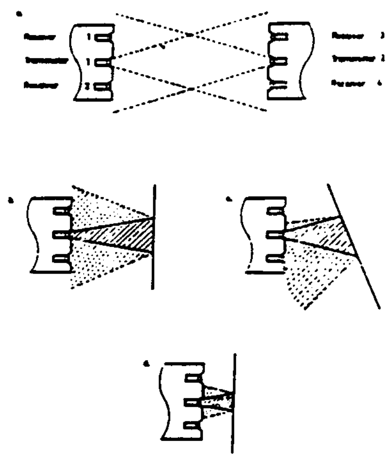


FIGURE 10, DIPICTION OF THE PRINCIPLE OF INFRARED DETECTION OF AN OBJECT BETWEEN FINGERS
(A) CONFIGURATION FOR INFRARED TRANSMITTER AND RECEIVER
(B and C) REFLECTION OF SIGNALS BY PARALLEL AND NON PARALLEL OBJECT SURFACES, RESPECTIVELY.
(D) SENSOR CONFIGURATION FOR OBJECT APPROACH SITUATION

ject [Stocic et al 1986]. The object will be grasped/manipulated - as, e.g., while either in motion during a carry/transport operation or in contact with the working environment, as in an assembly task. In this context, sensing can allow the robotic system to identify and then compensate for - uncertainties in the object inertia related - load and torque characteristics [Neuman, Khosla 1986, Seraji 1987, Stephenko 1987].

For sensor-based, effective task planning and control operations, the following dynamic world data are used for trajectory calculations: (1) Position and orientation of the retrieved object, (2) Distances, (3) Diameters, (4) Geometry of the retrieved object, (5) Path of the object (position, velocity), and (6) Contours. Vision (visual light and infrared) sensors provide such data for the EVA Retriever robotic system. Figure 8 gives a block diagram for a time-of-light laser scanner. Figure 9 depicts the basic geometry for a triangulation type laser scanner. These sensor systems can be used in mid-field to farfield vision systems to obtain rest of the dynamic world data discussed above. Up close - that is, in the near field of the robotic hand between the finger - arrays of infrared sensors can be used to sense object approach/proximity and orientation. Figure 10 depicts this use of infrared receiver, transmitter arrays.

Smart hand tactile sensing is required when the intelligent EVA robotic retriever system must: (a) Verify capture and stable grasping of an object, and (b) Perform delicate manipulate/assembly operations. It should be noted that the tactile sensors can make many manipulate/assembly etc. tasks simple which are very difficult to accomplish visually.

Tactile-sensing reduces to the following three fundamental sensing operations:

(1) Joint forces sensing - ie, sensing the forces applied to the joint of the robotic hand, wrist, and arm manipulator).
(2) Touch-sensing of the pressures applied at important contact points on the finger/hand surface.
(3) Slip-sensing of movement/incipient movement of the object while it is being grasped.

Joint forces are typically sensed using arrays of strain gages (strain-gage wrist force sensor) or Hall effect sensors,(fingers, etc. of the UTAH/MIT dextrous hand). Vibration or relative displacement sensors are used to detect object slip [Staugard 1987].

The sense of touch is used to generate the finer detail data required to describe the interaction between the robotic hand and the object or workpiece. This is because most of the manipulative/assembly action occurs at the contact interface between the hand and the object. A robot touch sensor should duplicate the object recognition, and position and orientation determination functions of the human skin/nerve ending system. Usually temperature sensing per se is not required. Table 11 summarizes the design specifications/characteristics of a representative tactile sensor. Figures 11 and 12 give schematic representations of the Hillis conductive rubber resistance change, and the Raibert and Tanner VLSI tactile/touch sensor array designs, respectively. Acoustic pulse through tuned elastomer array designs from Bonne-

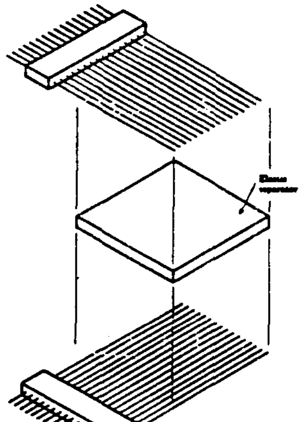


FIGURE 11. HILLIS DESIGN TACTILE SENSOR
THIS SENSOR CONSISTS OF A MONOLITHIC ARRAY OF 256 INDIVIDUAL SENSORY ELEMENTS. TYPICAL DESIGNS HAVE TWO SHEETS OF WIRES RUNNING PERPENDICULAR TO EACH OTHER SEPARATED BY A THIN ELASTIC MEDIUM. HERE THE INTERSECTION POINTS OF THE WIRES FORM THE INDIVIDUAL SENSORY ELEMENTS FOR PRESSURE MEASUREMENT.

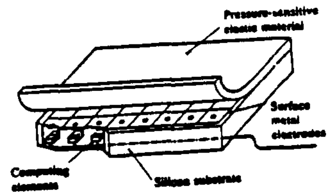


FIGURE 12. PHYSICAL STRUCTURE OF THE RAIBERT AND TANNER VLSI TACTILE ARRAY SENSOR DESIGN.
HERE A 1-mm THICK SHEET OF PRESSURE-SENSITIVE CONDUCTIVE PLASTIC (DYNACION B) IS PLACED IN CONTACT WITH A VLSI WAFER (AN NMOS INTEGRATED CIRCUIT. THE WAFER IS COMPRISED OF A TWO-DIMENSIONAL ARRAY OF CELLS, EACH OF DIMENSION 1.6 x 0.9 mm.

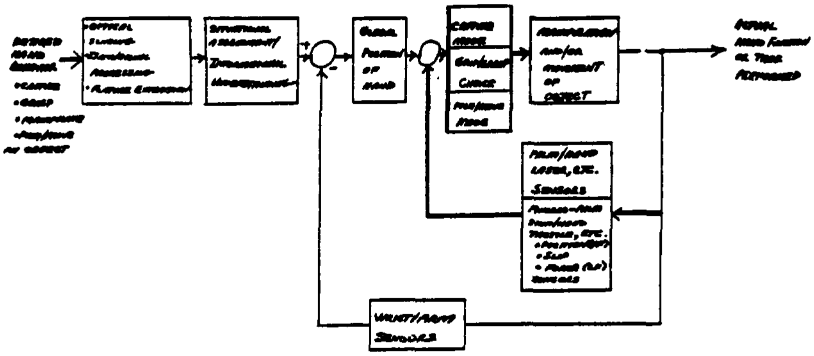


FIGURE 13. SCHEMATIC REPRESENTATION OF SMART HAND PREHENSION PROCESS AS A FEEDBACK OPERATION

TABLE 11. SUMMARY OF THE CHARACTERISTICS/DESIGN SPECIFICATIONS OF A REPRESENTATIVE TACTILE TRANSDUCER.

1. An Array Consisting of at Least 10 X 10 Force-Sensing Elements on a 1 sq-in Surface. This Corresponds to a Human Fingertip.

2. Each Element Should Have a Response Time of 1 to 10 msec. This should Be Closer to 1 msec. (Corresponds to 500 Hz Via Nyquist Sampling Criteria).

3. Threshold Sensitivity for the Elements Should Be Alpha X 1 g , With the Upper Limit of the Force Range at Alpha X 1000 g . Here Alpha Is a Factor Appropriate to the Space Environment (0 << Alpha < 1).

4. The Elements Do Not Need to Be Linear. However, They Must Have Low Hysteresis.

5. The Skinlike Material Must Be Robust. That Is, It Must Stand Up Well to the Harsh Space Environment and Be 'Modularly' Replaceable.

TABLE 12. FACTORS TO BE CONSIDERED BY A LANGUAGE FOR A VISION AND MULTISENSOR SYSTEM.

1. Image/Signal Preprocessing (Including Restoration, Enhancement, Etc.)
   - Instructions for Component 'Visual' and Multi-sensor Systems
   - Processing of All 'Pixel' or Array Element Level Information
   - Parallel Local Processing Operation
   - Efficient Storing of Multidimensional (2D,3D,Etc.) Structured Data

2. Image/Signal Extraction (Feature Extraction, Object Recognition Including MultiSensor Fusion)
   - Arithmetic, Delay Operations
   - Statistical Operations - Fusion of MultiSensor Information
   - Instructions for the 'Teach-In'/Learning Mode
   - Instructions for Feature Definition
   - Quantitative Feature Extraction

3. Image Analysis (Description)
   - Definition of Data Structures (Lists, Graphs, Etc.)
   - Definition of Object Value Relations ('Above', 'Below', 'Left', 'Right', Etc.)
   - Composition (Decomposition) of Relations
   - Knowledge Acquirement (Knowledge Accretion)
   - Model Generation (Via Geometric, Symbolic, CAD, Etc. Representations)

4. Image/Signal Context ('Scene' Analysis, Including Situation Assessment/Informational Understanding)
   - Analysis Based on Stored Knowledge
   - Inference Rules
   - Reasoning and Queries
   - Knowledge Management

Note that Points 3 and 4 Motivate the Construction of Symbolic, Rule-Based Languages Derived from the Results of Artificial Intelligence Research.

vile Scientific [Astle 1987] also look promising.

In considering the use of sensors in effective, task planning and control strategies for an intelligent hand, etc. Retriever robotic system, the computer language used must be given great consideration. Table 12 gives the factors to be considered by a computer programming language for a vision and multisensor system. Here the four main areas of (1) Image/Signal Processing, (2) Image/Signal Extraction, (3) Image/Analysis, and (4) Image/Signal Context (Scene Analysis) are considered. Requirements in these areas drive the design specification features which must be offered in a multi sensor-based, intelligent control system.

Retriever Robotic Control

Intelligent, sensor-based control of the smart dexterous hand plus arm plus MMU components of the Retriever robotic system is of primary importance to its successful EVA function. Here the arm plus hand/manipulator tasks of the Retriever can be specified in Cartesian coordinates. As discussed in the previous report section, sensors perform the observation function. The measured internal variable for feedback control purposes are joint displacements and velocities. Thus, the goal or command state variables and these measured output variable quantities are in different coordinates. This means that control of the position and orientation of the arm and hand/manipulator system by actuators at the joints requires knowledge of the transformation between the Cartesian hand and joint space respresentations.

Figure 13 gives a schematic representation of the smart hand prehension process as a feedback operation. In the system depicted desired hand behavior task commands are used in combination with global sensing to position the arm/hand system. At this level the hand mode is chosen and used to bring about grasping/low-level and /or movement of the object. Sensor information is continually fed back to control finger and hand object position, velocity and contact force.

Deep Knowledge Control

In developing grasp and manipulate - "trajectory" control of the path of a retrieved object, or an object/robotic hand system in contact with the environment, the dynamic modeling and the sensing functions discussed previously can be exploited. The hand/manipulator dynamics can be defined in joint space coordinates as

$$H \ddot{q} + h = T \qquad (1)$$

in which, q = Vector of joint displacements (θ, d) as discussed previously.

T = Vector of joint torques.

$H(q)$ = Manipulator inertia matrix.

$h(q, \dot{q}, t)$ = Nonlinear term containing the centrifugal, Coriolis, friction, and gravitational forces/torques.

As discussed previously, sophisticated on-line computational strategies are required to compute the arm plus hand/manipulator dynamics for the EVA Retriever robotic system. In today's arm plus

TABLE 13. SUMMARY OF THE NECESSARY FEATURES OF A MULTIPROCESSOR-BASED ROBOT CONTROLLER FOR THE EVA RETRIEVER.

1. Different Control Levels (Including Task Accomplishment, Opportunistic Scheduling)
2. Distribution of the Control Hierarchy for the Robot Among High-Speed, 32 Bit Microprocessors
3. Distributed System Operation (Necessary for Parallel Task Processing on Different Control Levels)
4. Modular Symmetric Hardware Configuration
5. Multiple Bus System
6. Expandability With No Need for Hardware or Software Reconfiguration
7. Special-Purpose Modules (Sensor Input/Output, Arithmetic, FFT, Interpolators, Kalman Filters for Observation and Control, Etc.)
8. Parallel Task Decompositions (A La Brooks, Lozano-Perez) for Real-Time/Real-World Problem Situations
9. Handling of a World Model in Global Memory

TABLE 14. BASIC COMPUTER ARCHITECTURE REQUIREMENTS - UNDER WHICH DEVELOPMENT AND IMPLEMENTATION OF THE ROBOT CONTROL SYSTEM SHOULD BE PERFORMED.

1. Clear, Well-Defined Separation of All Control Levels
2. Modular Architecture Built-Up on the Basis of a Symmetrically Distributed System
3. Standardized Interfaces Between the Control Modules (Hardware and Software)
4. Transparency to the User at All Control Levels
5. Defined Format and Standardized Protocols
6. Multibus System to Ensure Parallel Communication
7. Hierarchically-Based, 'Decentralized' System Control
8. Dynamic Reconfiguration for a (High Signal Communication to Processing Power - A La Hypercube Type) Polyprocessor Environment

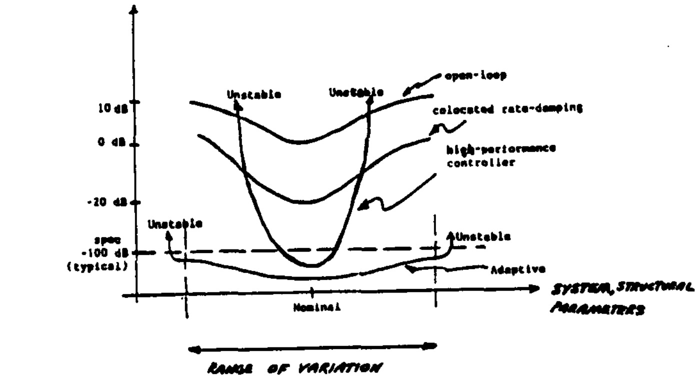


FIGURE 14. CLOSED-LOOP PERFORMANCE VERSUS VARIATION IN SYSTEM, STRUCTURAL PARAMETERS

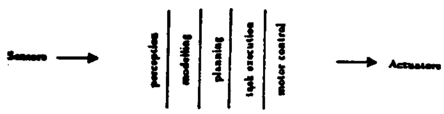


FIGURE 15. (A) REPRESENTATION OF THE TRADITIONAL DECOMPOSITION OF A MOBILE ROBOT CONTROL SYSTEM INTO FUNCTIONAL MODULES ('STAFF' ORGANIZATION)

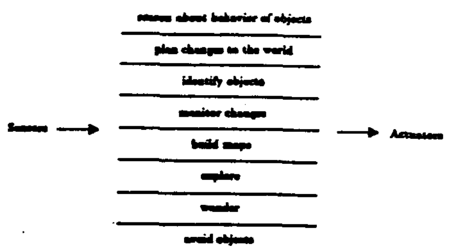


FIGURE 15. (B) REPRESENTATION OF A ROBOT CONTROL SYSTEM BASED ON TASK-ACHIEVING BEHAVIORS ('PROJECT' ORGANIZATION)

hand, etc. systems for the EVA Retriever the trend is toward mechanically "cleaner", more positive designs with fewer gear, etc. drive train components. This means that actuators, as, e.g., high torque motors are coupled directly or as closely as possible - to each joint in direct drive robotic system designs. Such designs have high mechanical stiffness, little or no backlash, and low friction. However, they also can have higher sensitivity to external disturbances, full link-to-link coupling, and nonlinear dynamics effects which are made worse by the high speeds which such systems can attain. These factors can increase the need for on-line computational dynamics and control.

Controlling robotic system dynamics of the form of Eq. (1) usually involves one of the three approaches:

(1) <u>Individual joint proportional, integral, derivative</u> (PID) control - which (a) Ignores link dynamics, and (b) Uses local decoupled PID's at each joint (ignores link coupling).

(2) <u>Computed torque control</u> - in which control torque T is defined using a structure identical to Eq. (1). That is,

$$T = H\,u + h \qquad (2)$$

Here the problem reduces to that of controlling the system

$$\ddot{q} = u \qquad (3)$$

Such control can be done using decoupled PID as per:

$$u = \ddot{q}_{Des_j} - KD_j\,\dot{\tilde{q}}_j - KP_j\,\tilde{q}_j - KI_j\int_0^t \tilde{q}_j\,dt \qquad (4)$$

in which, $\tilde{q}_j = q_j - q_{Des_j}$.
KP_j, KI_j, KD_j = PID coefficients which are positive and sufficiently large.

(3) <u>Robust Controller Design</u> - as, e.g., using pole-placement modal control and gain scheduling in combined feedforward plus feedback control strategies [Norcross et al 1986, Seraji 1987, Stephenko 1987]. Here the basic issue is how to minimize performance sensitivity to model model uncertainty, as, e.g., parameter uncertainty or inaccuracies in manipulator and object mass properties, torque constants of the actuators, friction of the drive train, unknown loads, etc. Model uncertainty can also encompass high-frequency unmodeled dynamics - such as structural resonant modes, sampling rates, and neglected time delays.

Figure 14 illustrates the idea of controlled system robustness. It plots closed-loop performance versus the variation in system parameters. This is done for different types of controlled dynamic system - ranging from open-loop (no feedback control) to adaptive (model reference ,parameter estimation, etc. types), closed-loop control.

Relevant control of the EVA Retriever robotic system it should be noted that there is an important tradeoff between (1) Speed, (2) Cost, and (3) Generality of application. This motivates the selecton of a two-level control structure as per: (1) Linear low level (inner loop) feedback control law, and (2) Generally nonlinear upper level (outer loop) feedforward control law. The outer loop optimizes the linear feedback control law parameters. This is done based upon the nonlinear equations of motion for the robotic system.

Hierarchical Strategy and Intelligent Control

In implementing control of the hand plus arm plus MMU comprising the EVA Retriever robotic system, it is important to consider the strategy/control architecture employed. Figure 15a gives a representation of the traditional decomposition of a mobile robot control system into functional modules. This "staff" organization of the control system is inadequate for the real-time implementation on a multiprocessor of control for the EVA Retriever. Figure 15b is a representation of a multilayered robot control system based on task-achieving behaviors. This "project" organization of the control system is made up of hierarchically parallel task layers. It lends itself to implementation on a multiprocessor based controller and hence shows great promise for real-time control of the complete EVA Retriever system [Brooks 1986].

Basic intelligent control algorithms can be implemented using a parallel microprocessor (i.e., multiprocessor based) structure. Here such rule-based algorithms for MIMO (multi-input, multi-output) systems have the property of decomposing or decoupling the multivariable control input into a set of single-input, multi-output systems. Thus, this allows for a parallel, pipelined (i.e., systolic) structure using a common rule base (as, e.g., with cloned CLIPS, etc.) that is easily implementable on an array of processors.

Any discussion of smart robotic hand control would be incomplete without mentioning the possibility of using neural nets [Lippmann 1987] or generalized learning algorithms [Miller et al 1987]. These can be used for multi-sensor pattern recognition functions and for the control of robotic hand/manipulator systems. Here the pattern recognition and control-learning schemes can be based solely upon observations of the input-output relationship for the system being controlled. Research work with Hopfield type neural nets [Freeman 1987] used in solving optimization problems, indicates that control objectives can be included. The author suggests the formulation of a generalized Hopfield neural net structure. This would include optimal robotic controls imbedding - as a Hamiltonian/Lagrangian form, augmented with state vector and control vector constraints.

Multiprocessor-Based Robot Control

The multiprocessor implementation of real-time intelligent control of the hand plus arm plus MMU comprising the EVA Retriever is a function of such issues as:

* Computational load balance
* Tradeoffs between (a) Signal communication time, and (b) Processing time
* Programming techniques - as a function of (a) Hardware/software architecture, and (b) Language environment
* Effective raw processor speed, quantization, and memory utilization
* "Impedance Match" with the problem formulation and algorithmic structure and complexity.

Table 13 summarizes the necessary features of a multiprocessor-based robotic controller for the EVA retriever. The basic computer architecture requirements - under which development and implementation of the robotic control system should be performed are given in Table 14. Both tables emphasize features/requirements relating to doing parallel task processing on different control levels.

Conclusions

An investigation has been made of the basic issues and concepts relating to dynamics, sensing, and intelligent control for the dextrous hand plus arm plus MMU system. This system defines the space-going components of the EVA Retriever robotic system. Specific attention has also been given to artificial intelligence, rule-based Expert Systems and the new area of Neural Nets. Based upon the research conducted, the author considers the following of prime importance: (1) Integrated dextrous hand plus articulated arm, mechanical and electrical drive system design, (2) Multisensor capabilities for position, velocity, force/torque, and slip/grasp stability, (3) "Standard" control and neural net observer synthesis for control purposes, and (4) Multiprocessor-based implementation of real-time, on-line dynamics, robust/adaptive control and hierarchally-layered (Brooks architecture) - task planning. Specific recommendations are summarized in the next report section in terms of a research and development program plan.

Recommendations

This section of the report presents recommendations for the smart robotic hands plus arms plus MMU - EVA Retriever system. These recommendations are presented in the form of a research and development program plan. The R & D program plan gives activities that can be pursued on a one, two, and three year basis. A more complete version, broken down into specific research and development plans will be available in an extended version of this report.

(1) Implementation of basic primitive operations for the robotic hands: (a) open-loop with force limit, and (b) Sensor-hand -- low level (basic movement) and high level (position, force contact, and slip/grasp stability) control.
(2) Implementation of task planner for six grasp and six basic manipulate robotic hand functions.
(3) Neural Nets developmental work: (a) Testbed w.r.t. robotic hands applications (pattern recognition for rule-based systems, control primitives, basis for "common sense" reasoning, use in fuzzy/noisy situations), (b) Use with A.D. Little/Marcus hand - master system, (c) Developmental operating system,(d) Knowledge representation significance versus sentence-like representations of knowledge, (e) Explanation of Neural Net reasoning and knowledge representation utility.
(4) Development of multiprocessor, systolic - type transputer versions of CLIPS, Common LISP: (a) "Cloned" CLIPS, etc. architecture with "Guarded Horn Clause" resolution of

contention for system resources, (b) Interfaced with multiprocessor versions of C and FORTRAN.

(5) Development of integrated AI/ES & NN plus digital signal processing system for using multisensor information.

(6) Development of distributed operating system which is necessary for/mandated by - the need to do parallel task processing on different hierarchical control/task levels.

(7) Development of ongoing testing and performance evaluation: (a) Prototyping tools as per (6) above, and (b) hardware, software AI/ES & NN controller testbed (hands etc.) for EVA Retriever (and Space Station intelligent control applications.

REFERENCES

1. "Extravehicular Activity (EVA) Retriever Program Plan," NASA/Johnson Space Center, Houston, Texas, Report No. JSC-22144, May 1987.
2. L. Astle, Presentation to CTSD - NASA/JSC on Bonneville Scientific - Acoustic Tactile Sensor Array and Related Micromecnanical and VLSI Capabilities, July 1987.
3. J. Freeman, "Neural Nets Seminar," NASA/JSC, July-August 1987.
4. D. Hillis, "Active Touch Sensing," International Journal of Robotics Research, Vol. 1, No. 2, 1982, pp 33-44.
5. D.T. Horak, "A Simplified Modeling and Computational Scheme for Manipulator Dynamics," Journal of Dynamic Systems, Measurement, and Control, Vol. 106, No. 4, December 1984, pp 350-353.
6. J. Kerr, B. Roth, "Analysis of Multifingered Hands," International Journal of Robotics Research, Vol. 4, No. 4, Winter 1986, pp 3-17.
7. R.H. Lathrop, "Parallelism in Manipulator Dynamics," International Journal of Robotics Research, Vol. 4, No. 2, Summer 1985, pp 80-102.
8. R.P. Lippmann, "An Introduction to Computing With Neural Nets," IEEE Acoustics, Speech, and Signal Processing (ASSP) Magazine, April 1987, pp 4-22.
9. J.Y.S. Luh, M.W. Walker, R.P. Paul, "On-Line Computational Scheme for Mechanical Manipulators," Journal of Dynamic Systems, Measurement, and Control, Vol. 102, 1980.
10. W.T. Miller, et al, "Application of a General Learning Algorithm to the Control of Robotic Manipulators," International Journal of Robotics Research, Vol. 6, No. 2, Summer 1987, pp 84-98.
11. J.J. Murray, C.P. Neuman, "Linearization and Sensitivity Models of the Newton-Euler Dynamic Robot Model," Journal of Dynamic Systems, Measurement, and Control, Vol. 108, No. 3, September 1986, pp 272-276.
12. C.P. Neuman, K.P. Khoshla, "Identification of Robot Dynamics: An Adaptation of Recursive Estimation," In K.S. Narendra (Editor), Adaptive and Learning Systems: Theory and Applications, Plenum, New York, 1986, pp 175-194.
13. R.J. Norcross, et al, "Pole-Placement Methods for Multivariable Control of Robotic Manipulators," Journal of Dynamic Systems, Measurement, and Control, Vol. 108, No. 4, December 1986, pp 340-345.
14. M.H. Raibert, J.E. Tanner, "Design and Implementation of a VLSI Tactile Sensing Computer," International Journal of Robotics Research, Vol. 1, No. 3, 1982, pp 3-18.
15. H. Seraji, "An Approach to Multivariable Control of Manipulators," Journal of Dynamic Systems, Measurement, and Control, Vol. 109, No. 2, June 1987, pp 146-154.
16. A.C. Staugaard, Robotics and AI: An Introduction to Applied Machine Intelligence, Prentice-Hall, Englewood Cliffs, New Jersey, 1987.
17. Y. Stephenko, "Modal Control of Fast Large-Scale Robot Motions," Journal of Dynamic Systems, Measurement, and Control, Vol. 109, No. 2, June 1987, pp 80-87.
18. D. Stocic, M. Vukubratovic, D. Hristic, "Implementation of Force Feedback in Manipulation Robots," International Journal of Robotics Research, Vol. 5, No. 1, Spring 1986, pp 66-76.