SAND96-0571C

Building a Modular Robot Control System using Passivity and Scattering Theory

Robert J. Anderson
Sandia National Labs
M.S. 1176, P.O. Box 5800
Albuquerque, NM 87185

Abstract

This paper analyses the problems and presents solutions for building a modular robot control system. The approach requires modeling the entire robot system using multi-dimensional passive networks, breaking the system into subnetwork "modules," and then discretizing the subnetworks, or n-ports, in a passivity preserving fashion. The main difficulty is the existence of "algebraic loops" in the discretized system. This problem is overcome by the use of scattering theory, whereby the inputs and outputs of the n-ports are mapped into wave variables before being discretized. By first segmenting the n-ports into nonlinear memoryless subnetworks and linear dynamic subnetworks and then discretizing using passivity preserving techniques such as Tustin's method, a complete modular robot control solution is obtained.

1.0 Introduction.

Sandia National Laboratories has been developing sensor-based robot control and advanced telerobotic control for remediating hazardous waste and decommissioning nuclear facilities for the last eight years. One of the key technologies developed as part of this effort is SMART (Sequential Modular Architecture for Robotics and Teleoperation). [2, 4]. SMART allows the system developer to build up advanced telerobotic behavior by combining SMART modules in different sequences. It has been applied to tasks as diverse as multi-arm control[5], waste-tank clean-up[6], and kinesthetic virtual reality[3].

With SMART it is assumed that all telerobot system components, such as input devices, sensors, control laws, robots, constraints, and robot environmental interaction can be modelled by multi-dimensional networks consisting of lumped parameter positive-definite matrix-valued one-port elements connected by possibly nonlinear two-port Jacobians, and driven by independent effort and flow sources.

The system is modularized by breaking up the complete closed network into individual ports [7,8], which can then be separately discretized and distributed among processors in a multiple-processor environment. In this paper, we will present the discretrization techniques by which the modularity is achieved in the discrete domain, and the steps needed to develop a SMART module.

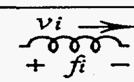
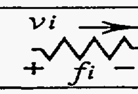
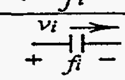
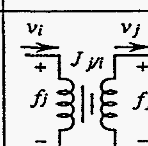
1.1 Terminology

In this paper networks will follow the effort flow analogy[9] which relates force (effort) to voltage and velocity (flow) to current. Subscripts will denote the space in which these signals are based. The operators derived from this analogy, and the new scattering variables are summarized in the two tables below.

TABLE 1.1 *Network variables*

symbol	description
v_i	velocity, angular velocity
f_i, fd_i	force or torque, desired force or torque
x_i, xd_i	position, desired position
a_i	input wave: $a_i = f_i + Z_0 v_i$
b_i	output wave: $b_i = f_i - Z_0 v_i$
Z_0, $Z0_p$	characteristic impedance: $Z0_p = Z_0 I_p$

TABLE 1.2 *Network Operators*

	name/ analogy	equation	network
M_i	inertia/ inductance	$f_i = \frac{d}{dt}(M_i v_i)$	
B_i	damping/ resistance	$f_i = B_i v_i$	
K_i	stiffness/ capacitance	$f_i = \int K_i dx_i$	
$J_{j/i}$	Jacobian/ modulated transformer	$v_j = J_{j/i} v_i$ $f_i = J_{j/i}^T f_j$	
S_i, S_{ixj}	scattering operator	$b_i = S_i a_i$ $\begin{bmatrix} b_i \\ b_j \end{bmatrix} = S_{ixj} \begin{bmatrix} a_i \\ a_j \end{bmatrix}$	

1.2 A Sample Problem

To demonstrate how SMART modules can be developed for a run time system we will design a sample modular telerobot control system. The sample system, albeit extremely

This work was supported by the United States Department of Energy under Contract DE-AC04-94AL85000.

DISCLAIMER

simplified, will illustrate the key issues and design techniques for building and discretizing a multitude of robot control modules.

The sample system consists of a filtering module (KB1), an input device module, (SPACE_BALL), a kinematics module (PUMA_KIN), and a dynamics emulation module (MB1). A SMART module icon representation, which shows both a pictorial representation of each module's functionality, and a network depiction of the underlying system is shown in Fig 1 below

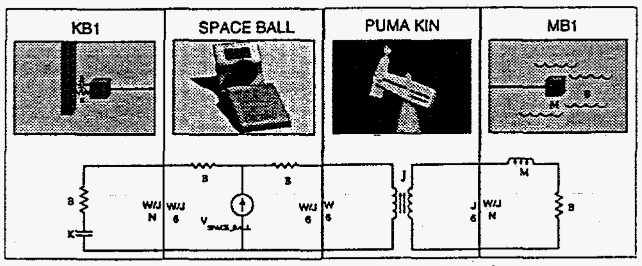


FIGURE 1: SMART Modules for sample system

During operation with these modules, six degree-of-freedom (DOF) commands are input by the human operator by pressing the space-ball, which decouples strain-gauge readings into a force/torque vector at the ball center. These measurements are converted to velocity commands by the SPACE_BALL module, and injected into the system as world position commands. The KB1 module serves to remove any spikes from the operator's input command stream, and directs the flow towards the robot. The PUMA_KIN module maps the world space position and velocity commands into joint space using a PUMA robot's kinematic equations. Finally, velocity commands are received by the MB1 module and mapped into joint torques using simulated joint inertia and damping. An actual robot would be controlled either by copying the stream of input velocities received by the MB1 module to a set-point controller on the robot, or by passing the torque commands directly to the robot's motor amplifiers.

In SMART each of these modules represents a separate computational element which can be mixed and matched with other modules to create behaviors. For instance, the PUMA_KIN module can be switched with the kinematics module for another robot. The SPACE_BALL module can be replaced or cascaded with another input device module.

In this paper we will investigate how each of these sample modules are independently designed and discretized, and how these modules can be combined to achieve the desired telerobot behavior in a multi-processor computing environment. In so doing, we hope to illustrate the basic concepts and techniques needed for building new modules. Before this is done, however, the problem of algebraic loops will be addressed.

2.0 The Problem of Algrebraic Loops

In a network representation of a system we are accustomed to thinking in terms of standard state variables, e.g., effort, flow, force, velocity, voltage, current, etc. Unfortunately, when the system is discretized, either for simulation or for control, these variables are generally the wrong variables, since their usage can introduce non-passivity, and instability into a system whose ideal continuous representation would be passive and robustly stable. Overcoming this problem is the key to designing SMART modules. Before developing these techniques for our prototype telerobot system, we shall start with a much simpler example.

2.1 Modularizing a Velocity Divider Network

Consider the figure shown below (Fig 2).

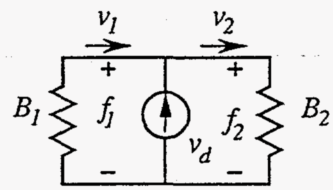


FIGURE 2: Velocity divider network

It consists of an ideal velocity source, $v_d(t)$, and two damping terms, B_1 and B_2. The constituent equations for the network elements are:

$$f_1 = -B_1 v_1, \tag{1}$$
$$f_2 = B_2 v_2, \tag{2}$$

and the connection equations are given by

$$v_2 = v_1 + v_d \tag{3}$$
$$f_1 = f_2 \tag{4}$$

which when combined, result in the current (velocity) divider equations:

$$\begin{aligned} v_1 &= -(B_1 + B_2)^{-1} B_2 v_d \\ v_2 &= (B_1 + B_2)^{-1} B_1 v_d \end{aligned} \tag{5}$$

Suppose however, that these equations are to be implemented in a modular fashion, and the constituent equations and the connection equations for the network are computed separately.

In this case a sample delay is incurred, since some computations depend on results of other computations. If the connection equations can be computed without sample delay, i.e., for each sample instant k,

$$v_2(k) = v_1(k) + v_d(k), \tag{6}$$
$$f_1(k) = f_2(k), \tag{7}$$

then the constituent equations must contain some sampling delay, as shown below

$$v_1(k) = -B_1^{-1} f_1(k) \tag{8}$$

$$f_2(k) = B_2 v_2(k-1) \tag{9}$$

Combining these equations yields

$$\begin{aligned} v_2(k) &= v_1(k) + v_d(k) \\ &= -B_1^{-1} f_1(k) + v_d(k) \\ &= -B_1^{-1} B_2 v_2(k-1) + v_d(k) \end{aligned} \tag{10}$$

Which has the discrete characteristic equation

$$0 = z + B_1^{-1} B_2 \tag{11}$$

For a scalar system this will be stable if and only if $|B_2| < |B_1|$. For a matrix valued system with diagonal damping operators B_1 and B_2 each element of B_2 must be less than the corresponding diagonal element of B_1, and for matrix-valued damping the induced *two-norm*[1] condition

$$\| B_1^{-1} B_2 \|_2 < 1 \tag{12}$$

must be satisfied.

Sampling in a different order may reverse the stability requirements, requiring that B_1 be less than B_2, but it will not eliminate the problem. As long as effort and flow variables are sampled, instability may arise due to sampling. This problem is called *algebraic instability*, where the ideal continuous system contains elements of the same order, which when separated into modular components result in additional discrete-time states. Thus, in our example, a zero-order damping system, becomes a first order system in the discrete domain. Likewise, two first-order coupled systems might result in a third-order discretized system. It is important to note that this problem exists *independent of the sampling rate*. Thus the common belief that "sampling fast enough" will cause our discrete system to exhibit the same behavior of the ideal continuous system is in general invalid for modular systems.

Now if the only goal was to design linear, time-invariant robotic control systems for single, rigid-body robots, with known a priori interactions then present methods would be adequate. The entire system could be combined into one large equation before sampling (as in Eq.5), or the stringent coupled stability requirements of Equation 12 could be met. However, if the goal is to build flexible, expandable, control systems such as SMART then discretization is mandatory, and modularity is required, not because it is conceptually and computationally pleasing, but because it is the only way to handle complex systems.

Luckily, there is a body of theory that can be applied, where stability is always maintained after discretization, where components can be designed in a completely modular fashion, and where the "sampling fast enough" adage will be true, even in the presence of additional discrete time states. The theory is called *scattering theory*.

1. The two-norm for a matrix, A, is equal to the maximum singular value, i.e., the maximum eigenvalue of the matrix A^TA.

3.0 Scattering Theory

Scattering theory was developed to handle problems in transmission line analysis. Ideal lossless transmission lines transmit effort and flow (voltage and current in this case) across distances without losing energy and without changing the steady-state behavior of the signal. However, depending on the line impedance, Z_0, and the impedances terminating either end of the line, the transient behavior of the signal will be effected.

3.1 Wave variables

Scattering theory involves a change of basis. Instead of using effort-flow variables to describe dynamic time-delay phenomena at port connections, *wave variables* are used. Wave variables represent a bilinear map of the standard effort-flow variables. The input wave, a, is defined as the effort plus the scaled flow entering the port, $a = f + Z_0 v$, where Z_0^2 is a constant representing the characteristic impedance. The output wave, b, is defined as the effort minus the scaled flow, $b = f - Z_0 v$

For a two-port (Fig. 3)

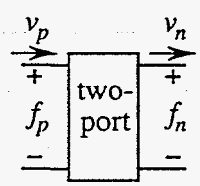


FIGURE 3: Two-port network

with flow, v_p, entering the port from the previous port, with flow, v_n, exiting the port to enter the next port, and with efforts, f_p, and f_n, at the port terminals, the corresponding wave variables are defined as

$$\begin{bmatrix} a_p \\ a_n \end{bmatrix} = \begin{bmatrix} f_p + Z0_p v_p \\ f_n - Z0_n v_n \end{bmatrix} \tag{13}$$

$$\begin{bmatrix} b_p \\ b_n \end{bmatrix} = \begin{bmatrix} f_p - Z0_p v_p \\ f_n + Z0_n v_n \end{bmatrix}. \tag{14}$$

If the underlying reference frame is matrix-valued, then the same equations still apply, except in this case Z_0, represents the scaled identity matrix.

2. The characteristic impedance, Z_0, would normally be derived from transmission line impedance parameters, and would in general be a complex number. In SMART, however, this parameter is arbitrary, and should be set to a real constant that causes the units of force and scaled velocity to have the same order of magnitude.

3.2 Scattering Operators.

Using this mapping of effort and flow variables into wave variables, we can determine how impedance, admittance and hybrid operators can be mapped into **scattering operators**. A scattering operator, S, is the mapping between input waves variables, a, and output wave variables, b.

$$b = S\,a, \tag{15}$$

and can be readily derived from the constituent equations for the given element.

For instance, suppose a linear-time-invariant (LTI) one-port is represented by its equivalent impedance, $Z(s)$,

$$f(s) = Z(s)v(s). \tag{16}$$

Then its input wave is defined by,

$$a(s) = f(s) + Z_0 v(s) = (Z(s) + Z_0)\, v(s), \tag{17}$$

and its output wave is defined by

$$\begin{aligned} b(s) &= f(s) - Z_0 v(s) = (Z(s) - Z_0)v(s) \\ &= (Z(s) - Z_0)(Z(s) + Z_0)^{-1} a(s) \end{aligned} \tag{18}$$

and thus the scattering operator is defined by,

$$S(s) = (Z(s) - Z_0)\,(Z(s) + Z_0)^{-1}.$$

The inverse, $(Z(s) + Z_0)^{-1}$, whether scalar or matrix valued, is always well defined since $Z(s)$ is positive real and Z_0 is strictly positive.

The scattering approach transforms passive impedances, which have Nyquist frequency plots in the right-half plane, to scattering operators, which have Nyquist frequency plots in the unit-circle.

Suppose we have a LTI one-port consisting solely of passive elements, e.g., masses, springs, dampers, and Jacobians, with driving point impedance $Z(s)$. It is a well known result of network theory[1] that the Nyquist plot of the impedance will lie entirely in the right-half plane. Likewise, the singular values for a passive two-port impedance operator or admittance operator will also lie entirely in the right half plane.

Thus for a LTI system where the characteristic impedance is chosen to be a real constant all of the following are equivalent:

- An n-port is passive.
- An n-port can be represented by a network of passive elements.
- An n-port has an impedance operator with a Nyquist plot contained entirely in the right half plane.
- An n-port has a scattering operator with frequency response in the unit circle.
- An n-port has a scattering operator with norm less than or equal to one.

The beauty of the scattering operator approach for robotic systems is that the scattering operator representation is equally valid for nonlinear systems as it is for linear systems. Thus although we can not talk about Nyquist plots for nonlinear systems we can still maintain the following equivalence:

- An n-port is passive
- An n-port can be represented by a network of passive elements.
- An n-port has a scattering operator with norm less than or equal to one.

3.3 Revisiting the Velocity Divider Problem

Let us now apply the scattering approach to the velocity divider problem. In terms of wave variables the connection equations are given by

$$\begin{aligned} a_1(k) &= f_1(k) - Z_0 v_1(k) = f_2(k) - Z_0 v_2(k) + Z_0 v_d(k) \\ &= b_2(k) + Z_0 v_d(k) \end{aligned} \tag{19}$$

$$\begin{aligned} a_2(k) &= f_2(k) + Z_0 v_2(k) = f_1(k) + Z_0 v_1(k) + Z_0 v_d(k) \\ &= b_1(k) + Z_0 v_d(k) \end{aligned} \tag{20}$$

and the constituent equations are given by

$$b_1(k) = S_1 a_1(k) \tag{21}$$

$$b_2(k) = S_2 a_2(k-1) \tag{22}$$

where $S_1 = (B_1 - Z_0)(B_1 + Z_0)^{-1}$ and $S_2 = (B_2 - Z_0)(B_2 + Z_0)^{-1}$.

Combining these equations then yields

$$\begin{aligned} a_2(k) &= b_1(k) + Z_0 v_d(k) = S_1 a_1(k) + Z_0 v_d(k) \\ &= S_1(b_2(k) + Z_0 v_d(k)) + Z_0 v_d(k) \\ &= S_1 S_2 a_2(k-1) + (S_1 Z_0 + Z_0) v_d(k) \end{aligned} \tag{23}$$

which has the characteristic equation

$$0 = z - S_1 S_2 \tag{24}$$

This equation has all of its poles in the unit circle since $\|S_1\| < 1$ and $\|S_2\| < 1$ for any positive damping B_1 and B_2. Furthermore this will be true no matter what sampling strategy is chosen. This will also be true even if B is a nonlinear function of position and/or velocity, as long as it remains positive definite at all times.

Thus when a system represented by effort-flow variables is modularized and sampled directly, (as in Section 2.1) it is likely to go unstable. If, on the other hand, its impedance is represented in terms of scattering operators, wave-variables rather than effort-flow variables are sampled, and the system is then modularized into individual ports, a discrete time modular system is derived which is as stable and robust as the original compound continuous system. The reason for this is summarized by the theorem below:

Theorem 3.1. Any delayed passive scattering operator, $e^{-sT}S(s)$, remains passive.

Proof. $\left\|e^{-sT}S(s)\right\| \le \left\|e^{-sT}\right\|\left\|S(s)\right\| \le \left\|S(s)\right\| \le 1$. ■

Because a pure phase shift cannot take the frequency response of the scattering operator outside of the unit circle, it cannot affect the passivity of the scattering operator.

4.0 Scattering Theory for Robot Ports

To illustrate how scattering operator techniques can be applied to robot control modules we will develop scattering operator maps for each of the SMART modules shown in Fig 1. We will first analyze the KB1 and MB1 one-ports at either end of the network, and then tackle the SPACE_BALL and PUMA_KIN two-ports.

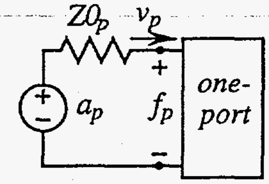


FIGURE 4: One-port network with driving force, a_p.

4.1 One-Port Scattering Operators

The simplest way to compute the scattering operator for a network is to apply a wave effort source in series with the characteristic impedance at each port, and then apply superposition of the inputs. For example, for a one-port, as shown in Fig 4, we can apply a driving signal, a_p, and a series impedance, Z_0, which results in the standard wave equation,

$$a_p = f_p + Z_0\, v_p \tag{25}$$

then using standard network analysis techniques we can derive

$$b_p = f_p - Z_0\, v_p \tag{26}$$

as a function of the input wave, a_p.

Example 4.1. KB1-Port Scattering Operator

Consider the stiffness-damping one-port (KB1-port) shown in Fig5 with driving wave attached.

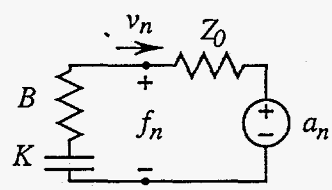


FIGURE 5: KB1-port with driving wave.

Applying Kirchoff's laws we get

$$a_n = -(K/s + B + Z_0)\, v_n. \tag{27}$$

Therefore we get for the wave equation

$$\begin{aligned} b_n &= a_n + 2Z_0 v_n \\ &= \left(I - 2Z_0\left(K\left(\frac{1}{s}\right) + B_1 + Z_0\right)^{-1}\right)a_n \\ &= (K + Bs - Z_0 s)(K + Bs + Z_0 s)^{-1} a_n \\ &= S_n a_n \end{aligned} \tag{28}$$

where the scattering operator, S_n, is given by

$$S_n = (K + Bs - Z_0 s)(K + Bs + Z_0 s)^{-1}. \tag{29}$$

Likewise the MB1 port scattering operator is given by

$$S_p = (Ms + B - Z_0)(Ms + B + Z_0)^{-1} \tag{30}$$

4.2 Two-Port Scattering Operators

We can apply the same technique to two-ports, by connecting a input driving wave at both port connections.

Example 4.2. Spaceball-Port

Consider the Spaceball two-port in Fig 6 with both driving waves attached. The spaceball adds a desired velocity command signal, v_d, to the input from the previous port to get the velocity input for the next port. It also adds a small amount of symmetrically placed damping, B, to help reduce wave reflections in the module.

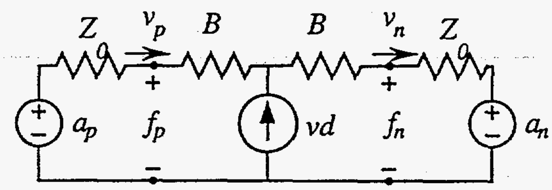


FIGURE 6: Spaceball -port with driving waves.

Applying Kirchoff's laws we get

$$v_p + \frac{1}{2}vd = v_n - \frac{1}{2}vd = \frac{1}{2}(Z_0 + B)^{-1}(a_p - a_n) \tag{31}$$

and thus for the wave equations:

$$\begin{aligned} \begin{bmatrix} b_p \\ b_n \end{bmatrix} &= \begin{bmatrix} a_p \\ a_n \end{bmatrix} - 2Z_0\begin{bmatrix} v_p \\ -v_n \end{bmatrix} \\ &= (Z_0 + B)^{-1}\begin{bmatrix} B & Z_0 \\ Z_0 & B \end{bmatrix}\begin{bmatrix} a_p \\ a_n \end{bmatrix} - Z_0\begin{bmatrix} vd \\ vd \end{bmatrix} \\ &= S\begin{bmatrix} a_p \\ a_n \end{bmatrix} - Z_0\begin{bmatrix} vd \\ vd \end{bmatrix} \end{aligned} \tag{32}$$

5.0 The Jacobian Scattering Operator.

A Jacobian operator, J, can be used to define any mapping between reference frames which satisfies the principal of virtual work. In robotics it typically refers to the velocity mapping between a robot's joint space and its world space motions. In SMART the Jacobian is used for a wide variety of variable mappings. It is used to map between different frames on a manipulator, it is used to implement nonlinear boundary functions, and it is used to map between spaces with different degrees-of-freedom. More than any other element it enables the versatility and generality of the vector network approach. In this section the scattering operator representation for the Jacobian element will be derived.

5.1 Transformer Representation of Jacobians

Consider the Jacobian element shown in Fig 7 below and represented by a modulated transformer.

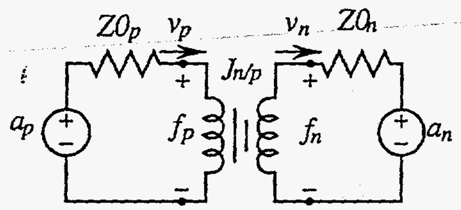


FIGURE 7: Jacobian element driven by wave inputs

The Jacobian, $J_{n/p}(x_p)$, maps the velocities from space p to space n as a function of the position x_p, according to the equation, $v_n = J_{n/p}(x_p)v_p$, and thus has the force mapping, $f_p = J_{n/p}^T f_n$. The goal is to determine the mapping between the input waves (eq. 13-14),

Applying the loop equation to Fig 7 gives,

$$a_p - Z0_p v_p = J_{n/p}^T a_n + J_{n/p}^T Z0_n v_n . \quad (33)$$

since the impedance operators are just scaled identity operators, $Z0_p = Z_0 I_p$ and $Z0_n = Z_0 I_n$, it follows that

$$a_p - J_{n/p}^T a_n = (I_p + J_{n/p}^T J_{n/p}) Z0_p v_p \quad (34)$$

By defining the matrix,

$$D_p = 2(I_p + J_{n/p}^T J_{n/p})^{-1} \quad (35)$$

the output waves can be written as

$$\begin{bmatrix} b_p \\ b_n \end{bmatrix} = \begin{bmatrix} f_p - Z0_p v_p \\ f_n + Z0_n v_n \end{bmatrix} = \begin{bmatrix} a_p - 2Z0_p v_p \\ a_n + 2Z0_n J_{n/p} v_p \end{bmatrix}$$
$$= \begin{bmatrix} I_p - D_p & D_p J_{n/p}^T \\ J_{n/p} D_p & I_n - J_{n/p} D_p J_{n/p}^T \end{bmatrix} \begin{bmatrix} a_p \\ a_n \end{bmatrix} \quad (36)$$

which implies that the scattering operator for the two-port Jacobian, $J_{n/p}$, is given by,

$$S_{p \times n} = \begin{bmatrix} I_p - D_p & D_p J_{n/p}^T \\ J_{n/p} D_p & I_n - J_{n/p} D_p J_{n/p}^T \end{bmatrix} . \quad (37)$$

[1]

This matrix inversion is always easily computable since the desired inverses are positive definite symmetric. This is an important point, even if the Jacobian is singular or non-square, the scattering operator derived from the Jacobian is always well defined. Thus the functional dependency of the Jacobian operator on the x_p can be changed without being concerned by the effects of singularities.

5.2 The Jacobian Scattering Operator Norm.

The imposed two-norm on a matrix valued system is defined as the maximum singular value of the matrix. To compute the singular values for S_{pxn}, we need to compute the eigenvalues of,

$$S_{pxn}^T S_{pxn} = \begin{bmatrix} I_p - D_p & D_p J_{n/p}^T \\ J_{n/p} D_p & D_n - I_n \end{bmatrix} \begin{bmatrix} I_p - D_p & D_p J_{n/p}^T \\ J_{n/p} D_p & D_n - I_n \end{bmatrix}$$
$$= \begin{bmatrix} I - 2D_p + D_p^2 + D_p J^T J D_p & D_p J^T D_n - D_p^2 J^T \\ D_n J D_p - J D_p^2 & J D_p^2 J^T + D_n^2 - 2D_n + I \end{bmatrix} \quad (38)$$

but $D_p J_{n/p}^T = J_{n/p}^T D_n$, and $D_p J^T J D_p + D_p^2 = 2D_p$.

Therefore, $S_{pxn}^T S_{pxn} = \begin{bmatrix} I_p & 0_{p/n} \\ 0_{n/p} & I_n \end{bmatrix}$ and consequently all of the eigenvalues are identity, and thus $\|S_{pxn}\|_2 = 1$ for any choice of $J_{n/p}$.

So despite the fact that the Jacobian can vary widely as a function of position, the resulting scattering operator is always orthonormal, with a norm of one. This somewhat surprising mathematical result, is not surprising at all when the physics of the system are analyzed. Because the Jacobian represents a lossless transformation, and the norm of

---

1. This scattering operator will only work directly for the three linear DCFs. The angular velocity DOFs can be computed in a similar fashion only by premultiplying and post-multiplying the wave variables by the appropriate rotation matrix. This technique is beyond the scope of this paper.

1. A similar form can be computed using the matrix, $D_n = 2(I_n + J_{n/p} J_{n/p}^T)^{-1}$ which may be more efficient than (Eq. 35) depending on the number of respective DOFs.

and calling the update routine for each of the modules at a fixed rate. For the SMART system each module is placed arbitrarily across the processor boards in a VME backplane. Dual ported memory is used to link up the outputs of one module to the inputs of the next module.

If all of the modules in the system have been designed using the techniques described in this paper, then they will have the same stability and performance independent of the order that they are sampled. If sampling is increased then the performance will more closely approximate the continuous system upon which they are based. The system will remain stable even if modules are sampled at different rates, miss sample updates, or stop updating all together. In SMART the modules are all clocked at the same basic period by setting up a clocked interrupt process on the first CPU, and using mailbox interrupts to trigger the other CPUs.

Stable force/velocity feedback paths are maintained in SMART by using the wave variable approach presented here. Exact position tracking, however, requires that position information be passed as well. This can be done in a stable fashion by passing position only in a feedforward loop, based on the feedback wave variables. As long as the scattering operators are passive for all positions, the force and velocity loops will remain stable, and as long as feedforward position is based only on the stable wave variables, the position will remain stable as well.

8.0 Conclusions.

Our approach to building a modular robot control system consists of the following steps. First represent the system and the controller as a closed network, such as the teleoperator control system in Figure 1. Second, break up the network system into a series of n-port modules. Third, for each n-port module, break the system into linear time invariant sections and nonlinear Jacobian sections. Fourth, apply Tustin's passivity preserving discretization method to all linear-time-invariant components of the system. Fifth, apply scattering theory to all network components to get input-output mappings between wave variables for every module, precompute all filter constants. Sixth, distribute the modules across the processors in the system and connect the output waves of each module to the input waves of the next module using pointers in dual-ported memory.

By basing the SMART architecture on these intuitive network modular blocks, rather than reducing the system to a state-space format, even novice users can build up complex nonlinear, behavior based telerobotic control systems, incorporating multiple robots, nonlinear constraint forces, human operator input, and sensor feedback. Furthermore, since the modules can be independently updated and connected, these systems can be automatically generated and flexibly adapted to fit any distributed computer architecture.

9.0 References.

[1] B. D. O. Anderson, Network Analysis, Prentice Hall, Englewood Cliffs, NJ, (1973).

[2] R. J. Anderson, "SMART: A Modular Control Architecture for Telerobotics", *IEEE Robotics and Automation Society Magazine*, Vol. 2, No., 3, Sept. '95, pp. 10-18.

[3] R. J. Anderson, "Teleoperation with Virtual Force Feedback", *Proceedings of the'93 SPIE Int. Symp. on Optical Tools for Manufacturing and Advanced Automation*, Sept. 1993, Boston, MA.

[4] R. J. Anderson, "Autonomous, Teleoperated, and Shared Control of Robot Systems", '96 IEEE International Conference on Robotics and Automation, St. Paul, Minn. April 22-28.

[5] R. J. Anderson and K. Lilly, "A Modular Approach to Multi-Robot Control", *IEEE Conference on Decision & Control"*, New Orleans, Dec. 1995.

[6] R. J. Anderson and B. Davies, "Using Virtual Objects to Aid Underground Storage Tank Teleoperation", *Proceedings of '94 IEEE International Conference on Robotics and Automation"*, pp. 1421-1426, San Diego, May. 1994.

[7] R. J. Anderson and M. Spong, "Asymptotic Stability for Force Reflecting Teleoperators with Time Delay", *The International Journal of Robotics Research*, vol. 11, pp. 135-149, 1992.

[8] G. Raju, G. C. Verghese and T. B. Sheridan, "Design Issues in 2-port Network Models of Bilateral Remote Manipulation", *IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, pp. 1316-1321, 1989.

[9] R. Rosenberg and D. Karnop, *Introduction to Physical System Dynamics*. New York: McGraw-Hill, 1983.