(Optimal) State Estimation of (Non) Linear, (Un) Stable Systems

Richard W. Longman
Columbia University
New York, NY

NSF Workshop on
Mathematical Modeling and Control of Plasmas in Magnetic Fusion
General Atomics, San Diego
May 12,2006

WORDS

- **STATE** $x(t)$
  - **A column vector of the variables you would use as initial conditions (given at time *t*)**
  - **Or any column vector related by an invertible transformation**
  - **There are an infinite number of choices for state variables**

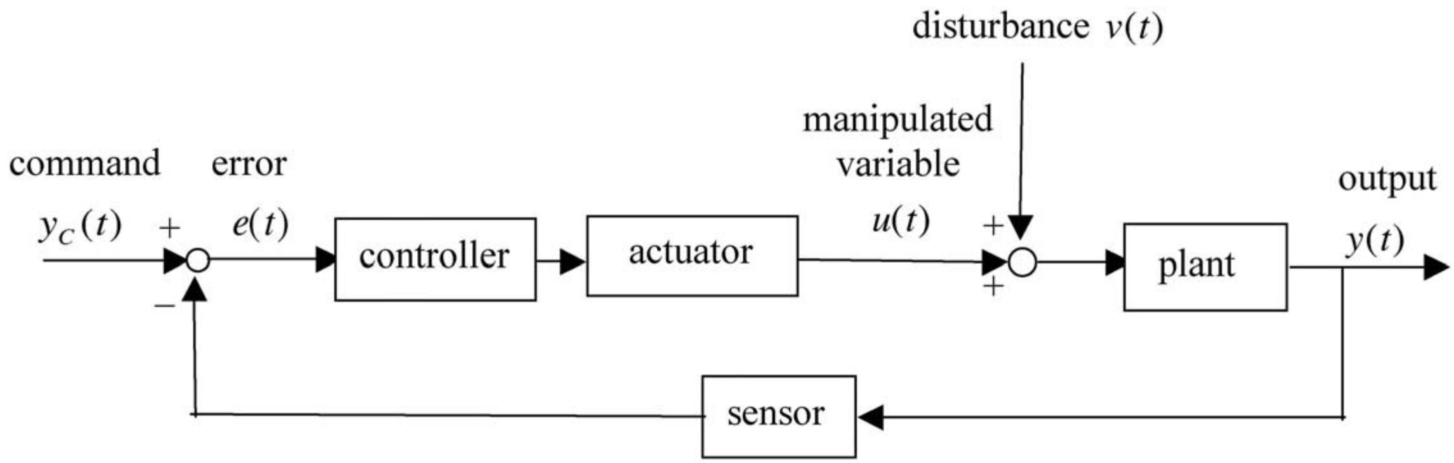
- **THE FUNCTION OF A STATE ESTIMATOR**
  - **Given the time history of all sensor outputs, construct the current state vector**

- **WORDS FOR STATE ESTIMATOR**
  - **State estimator = observer, general term**
  - **Luenberger observer: an observer designed based on deterministic model**
  - **Kalman filter: an observer designed based on a stochastic model**

SEQUENCE OF TOPICS

- **Why might we want an observer when we build a control system?**
  Review of concepts from classical control and their limitations
- **The structure of state feedback and of state estimators**
- **Design of feedback and Luenberger observers by pole placement**
- **The separation principle (deterministic)**
- **Other methods to design the state feedback**
- **Stochastic models**
- **Kalman filter design**
- **Issues of ODE vs. PDE – control and observation spillover**
- **Some nonlinear controller and observer concepts**
- **Control of nonlinear systems with adaptive local linear model updates**

CLASSICAL FEEDBACK CONTROL BACKGROUND



- **Example plant:** $\dddot{y}(t) + a_3\ddot{y}(t) + a_2\dot{y}(t) + a_1 y(t) = u(t) + v(t)$

- **Error:** $e(t) = y_c(t) - y(t)$

- **PID controllers:**

  $u(t) = K_1 e(t)$ (proportional, P)

  $u(t) = K_1 e(t) + K_2\dot{e}(t)$ (P plus derivative, PD)

  $u(t) = K_1 e(t) + K_2\dot{e}(t) + K_3\int_0^t e(\tau)d\tau$

  (PD plus integral, PID)

DESIGN PROCESS

- **The $K_1, \; K_2, \; K_3$ are called control gains. They are to be adjusted to obtain acceptable performance.**
- **Combine the equations to make a differential equation that predicts the behavior of the closed loop system. Consider proportional control:**

$$\dddot{y}(t) + a_3\ddot{y}(t) + a_2\dot{y}(t) + a_1 y(t) = K_1[y_C(t) - y(t)] + v(t)$$

$$\dddot{y}(t) + a_3\ddot{y}(t) + a_2\dot{y}(t) + (a_1 + K_1)y(t) = K_1 y_C(t) + v(t)$$

- **A nonhomogeneous ODE with two forcing functions. Solution has three parts:**

$$y(t) = y_H(t) + y_{PC}(t) + y_{PV}(t)$$

$y_H(t) = C_1 e^{s_1 t} + C_2 e^{s_2 t} + C_3 e^{s_3 t}$**, solution of homogeneous equation**
$y_{PC}(t)$**, particular solution associated with command** $y_C(t)$
$y_{PV}(t)$**, particular solution associated with disturbance** $v(t)$

THREE COMPETING DESIGN OBJECTIVES

- $y_{PC}(t)$ **is the only part of the solution related to the command. Want it to equal the command, if possible.**

- $y_{PV}(t)$ **reacts only to disturbances. We want disturbances to have no effect on the behavior of the output, if possible.**

- $y_H(t)$ **is only related to initial conditions, and is unrelated to the command. We need stability and a reasonable settling time.**

<u>Stability</u>:

- So we want $y_H(t) \to 0$**, and do so for all possible initial conditions, i.e. for all possible** $C_1,\ C_2,\ C_3$ **(asymptotic stability)**

- If $s_j = -\alpha_j + i\beta_j$**, the associated solution is** $C_j e^{-\alpha_j t}[\cos(\beta_j t) + i\sin(\beta_j t)]$**. We need all** $\alpha_j > 0$ **(all roots in open left half plane)**

Settling Time:

- Settling time, i.e. how long we have to wait for $y_H(t)$ to become negligible, is often computed as 4 times the longest time constant, which is maximum over all roots of $4/\alpha_j$. Then $y_H(t)$ is less than 2% of its original value [exp(-4)=0.016].
- Too long a settling time means you wait a long time for the system to respond.
- Too short a settling time means you use a lot of energy, or you saturate your actuator.

WITH CLASSICAL CONTROL YOUR HANDS ARE TIED

- **Consider the proportional control system above. The characteristic polynomial is**

$$s^3 + a_3 s^2 + a_2 s + (a_1 + K_1) = 0$$

- **Written in factored form with roots** $s_1, \; s_2, \; s_3$

$$(s - s_1)(s - s_2)(s - s_3) = 0$$
$$s^3 - (s_1 + s_2 + s_2)s^2 + (s_1 s_2 + s_2 s_3 + s_3 s_1)s - (s_1 s_2 s_3) = 0$$

- **Note that the coefficient of** s^2 **cannot be influenced by the one controller gain** K_1**. And this means that no matter what gain is picked:**

the average location of the roots $(s_1 + s_2 + s_3)/3$ ***is the same for all gains***
(possibly a serious limitation when stabilizing unstable systems)

- **A PD controller influences** a_2 **but not** a_3 **(if you can isolate a second order mode in your data, this would be enough). A PID does not work because it increases the order to 4th order and there is one more coefficient.**

WHAT WOULD BE REQUIRED TO HAVE COMPLETE FREEDOM?

- **Assume that we can measure the whole state, have three sensors, and abandon the structure of the classical feedback control loop to write**

$$u(t) = K_1 y_C(t) - k_1 y(t) - k_2 \dot{y}(t) - k_3 \ddot{y}(t)$$

The closed loop differential equation becomes

$$\dddot{y}(t) + (a_3 + k_3)\ddot{y}(t) + (a_2 + k_2)\dot{y}(t) + (a_1 + k_1)y(t) = K_1 y_C(t) + v(t)$$

Clearly, the feedback gains $k_1,\ k_2,\ k_3$ **allow one to pick all coefficients independently.**

- **As an aside – what would be required so that the particular solution associated with the command were identical to the command for all commands? $y_{PC}(t)$ will equal $y_C(t)$ if one replaces $K_1 y_C(t)$ by input**

$$\dddot{y}_C(t) + (a_3 + k_3)\ddot{y}_C(t) + (a_2 + k_2)\dot{y}_C(t) + (a_1 + k_1)y_C(t)$$

Picking $K_1 = a_1 + k_1$ is enough if one is only interested in constant commands

- **When considering controlling multiple outputs simultaneously, one wants the system to be controllable, or at least all unstable modes controllable.**

POLE PLACEMENT CONTROLLER DESIGN

- **Roots** $s_1,\ s_2,\ s_3$ **are the poles of the closed loop system. To design the system for stability and for a desired settling time, use state feedback (set command to zero to address stability)**

$$u(t) = -Kx(t) = -\begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix} \begin{bmatrix} y(t) \\ \dot{y}(t) \\ \ddot{y}(t) \end{bmatrix}$$

- **Pick any root locations that you want for** $s_1,\ s_2,\ s_3$ **and then solve for** $k_1,\ k_2,\ k_3$ **to match coefficients**

$$\begin{aligned} a_1 + k_1 &= -s_1 s_2 s_3 \\ a_2 + k_2 &= s_1 s_2 + s_2 s_3 + s_3 s_1 \\ a_3 + k_3 &= -(s_1 + s_2 + s_3) \end{aligned}$$

COMMENTS

- **Mathematically you have complete freedom to place the roots wherever you choose, and sometimes this is an effective design method.**
    - **Trouble: this asks for too much detail – you don't really know where you want all the roots, so you have to make trial simulations**
    - **But you do know a reasonable settling time**
    - **And you know you don't want to saturate your actuators**

- **Other design methods such as quadratic cost (LQR) produce the same state feedback structure – LQR picks root locations based on some gains you choose in the cost function.**

- **Implementing state feedback, obtained by whatever design approach, assumes you know the state. It is very rare that you can measure the whole state.**

<u>Conclusion</u>: You need an observer to estimate the state from measurements

- **In some situations one can simplify this:**
  - **Suppose that you have enough sensors located in appropriate spatial positions so that you can take a linear combination of the measurement to isolate the response of one mode**
  - **Suppose that mode is a second order mode, then you need** $y(t)$ **and** $\dot{y}(t)$
  - **It is not uncommon to make only a measurement of** $y(t)$ **and approximate** $\dot{y}(t)$ **by a finite difference** $[y((k+1)T) - y(kT)]/T$
  - **The observers here are more sophisticated, and use knowledge of the dynamics of the system. Result is likely to have less contamination by noise.**

WRITING DIFFERENTIAL EQUATIONS IN CONTROLLABLE CANONICAL FORM

- **Classical control uses transfer functions as its standard representation. Modern control methods usually use a state space representation**

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

The first equation represents the dynamics of the system, the second says what state variables are being measures, or are the output for the system.

- **The most natural definition for the state vector components**

$$x_{1c} = y, \quad x_{2c} = \dot{y} = \dot{x}_{1c}, \quad x_{3c} = \ddot{y} = \dot{x}_{2c}$$

Subscript c **denotes controllable canonical form**

Then the state equations become

$$\dot{x}_c(t) = A_c x_c(t) + B_c u(t)$$
$$y(t) = C_c x_c(t)$$

where

$$A_c = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_1 & -a_2 & -a_3 \end{bmatrix} \quad B_c = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad C_c = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

Note the coefficients of characteristic equation on bottom of matrix.

- **Apply state feedback, obtained by pole placement of any other method**

$$u(t) = -Kx_c(t)$$
$$\dot{x}_c(t) = (A_c - B_c K)x_c(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -(a_1 + k_1) & -(a_2 + k_2) & -(a_3 + k_3) \end{bmatrix} x_c(t)$$

From this form it is obvious how to pick K **to place the poles if desired**

WRITING DIFFERENTIAL EQUATIONS IN OBSERVABLE CANONICAL FORM

- **Another choice for state variables is the observable form**

$$\frac{d}{dt}\left(\frac{d}{dt}\left(\frac{dy}{dt}+a_3y\right)+a_2y\right)+a_1y=u$$

Define

$$x_{3o}=y$$

$$x_{2o}=\dot{y}+a_3y=\dot{x}_{3o}+a_3x_{3o}$$

$$x_{1o}=\dot{x}_{2o}+a_1y=\dot{x}_{2o}+a_1x_{3o}$$

Then $\dot{x}_o(t)=A_ox_o(t)+B_ou(t) \qquad y(t)=C_ox_o(t)$

$$A_o=\begin{bmatrix}0 & 0 & -a_1\\ 1 & 0 & -a_2\\ 0 & 1 & -a_3\end{bmatrix} \quad B_o=\begin{bmatrix}1\\0\\0\end{bmatrix} \quad C_o=\begin{bmatrix}0 & 0 & 1\end{bmatrix}$$

The characteristic polynomial coefficients are in the last column

- **Note the relationship between the state variable definitions**

$$x_{3o} = x_{1c}$$

$$x_{2o} = x_{2c} + a_3 x_{1c}$$

$$x_{1o} = \ddot{y} + a_3 \dot{y} + a_2 y = x_{3c} + a_3 x_{2c} + a_2 x_{1c}$$

or

$$x_o = M x_c \qquad M = \begin{bmatrix} a_2 & a_3 & 1 \\ a_3 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

DESIGNING A LUENBERGER OBSERVER BY POLE PLACEMENT

- **The form of a Luenberger observer**

$$\dot{\hat{x}}_o(t) = A_o\hat{x}_o(t) + B_o u(t) + F[y(t) - C_o\hat{x}(t)]$$

  - **The $\hat{x}_o(t)$ is the observer estimate of the state $x_o(t)$.**
  - **The $y(t) - C_o\hat{x}(t)$ is the difference between the measurement predicted by the state estimate, and the actual measurement**
  - **This difference drives the equation through gain matrix**

$$F = -\begin{bmatrix} f_1 & f_2 & f_3 \end{bmatrix}^T$$

- **Define the error in the state estimate as $e_o(t) = \hat{x}_o(t) - x_o(t)$. The above difference term is $-C_o e_o(t)$. Subtract state equation from observer equation to obtain**

$$\dot{e}_o(t) = (A_o + FC_o)e_o = \begin{bmatrix} 0 & 0 & -(a_1 + f_1) \\ 1 & 0 & -(a_2 + f_2) \\ 0 & 1 & -(a_2 + f_2) \end{bmatrix} e_o(t)$$

- **The characteristic polynomial is therefore**

$$s^3 + (a_3 + f_3)s^2 + (a_2 + f_2)s + (a_1 + f_1) = 0$$

As before, pick any desired root locations and adjust F **to create those roots.**

- **What are the limitations:**
  - **Pole placement for control was limited by saturation limits of the actuator**
  - **For the observer, the limitation is noise in the data**
  - **Settling time of observer must be long enough that the change in the measurements is dominated by changes in the system output, not by changes in the noise in the measurements**
  - **Of course, to be useful in control, the settling time of the observer must be faster than that of the controller**

- **Note that one should require that the system is observable**

THE REAL TIME COMPUTATION LOAD

- **The computation load to obtain the control action $u(t)$ to apply to the system is to integrate the observer equation in real time, and multiply the result by controller gain matrix –K.**

- **In practice, one uses a parallel development from digital control to obtain a state space difference equation, updating the control every T seconds using**

$$\hat{x}((k+1)T) = A\hat{x}(kT) + Bu(kT) + F[y(kT) - C\hat{x}(kT)]$$
$$u(kT) = -K\hat{x}(t)$$

And these computations must be completed in the allotted time T sec.

MINIMAL ORDER LUENBERGER OBSERVER

- **Suppose that one measures both $y(t)$ and $\dot{y}(t)$.**
  - **Then we know $x_{3o}(t) = y(t)$, and $x_{2o}(t) = \dot{y}(t) + a_3 y(t)$.**
  - **We only really need to use an observer to find $x_{1o}(t)$**
  - **The formula for this state is just**

$$\dot{x}_{1o}(t) = -a_1 x_{3o}(t) + u(t)$$

  - **Using this equation reduces the real time computation burden**
  - **Note, however, the observer will likely give less noisy values for the measured states, and would still be preferred unless computation time is a dominant issue**

A DETERMINISTIC SEPARATION PRINCIPLE

- **The following separation principle holds for designing feedback control systems with observers:**

  - **Design a state feedback controller as if you know the state. Use any design method to determine the matrix of gains K, and note the roots of the characteristic polynomial.**
  - **Design an observer by any chosen method, and note the roots of the characteristic polynomial.**
  - **Then, if one uses the estimated state from the observer in the control law, the closed loop characteristic equation has twice the order of the system, and the roots are the combination of the roots above.**
  - **In other words, using the observer does not destroy the characteristics of the controller design.**

- **One can establish this result as follows:**

  - **Use $x_o = Mx_c$ to rewrite the controllable canonical form equation as**

$$\dot{x}_o = (MA_cM^{-1})x_o + MB_cu = A_ox_o + B_ou$$

  and combine with the error propagation equation for the filter

$$\frac{d}{dt}\begin{bmatrix} e_o \\ x_o \end{bmatrix} = \begin{bmatrix} A_o + FC_o & 0 \\ -B_oKM^{-1} & A_o - B_oKM^{-1} \end{bmatrix} \begin{bmatrix} e_o \\ x_o \end{bmatrix}$$

  - **This is lower block triangular so the eigenvalues are those of the diagonal blocks**
  - **The first block has the eigenvalues of the observer**
  - **The second diagonal block is $M(A_c - B_cK)M^{-1}$ and has the eigenvalues of the controller design**

SOME METHODS TO DESIGN A CONTROLLER

- **Pole placement, as above.**

- **Linear-Quadratic Regulator (LQR). Pick $u(t)$ to minimize the quadratic cost function**

$$J = \int_0^\infty [x^T(t)Qx(t) + u^T(t)Ru(t)]dt$$

subject to the state equation $\dot{x} = Ax + Bu$, with R positive definite, Q at least positive semidefinite. Solution:

$$u(t) = -Kx(t)$$

$$K = R^{-1}B^T S$$

$$SA + A^T S + Q - SBR^{-1}B^T S = 0$$

- **Comments:**

- **Solution requires state feedback**
- **You normally do not know in advance what** Q **and what** R **to use, they are adjusted in simulation**
- **Thus, they serve as parameters to adjust for good performance like the** K_1, K_2, K_3 **in a PID controller**
- **LQR has an advantage that no matter what values you try, the system is always stable**
- **And it has the advantage that the state feedback nature gives it enough freedom to put roots anywhere it wants**
- **Tuning – consider a simple SISO case**

$$J = \int_0^\infty (qy^2 + ru^2)dt$$

If q/r **is very large,** J **asks to get** y **to be small very quickly, i.e. the factor is adjusting the settling time.**
If a choice of q/r **saturates the actuators in typical applications, increase** r **to put more emphasis on how much control effort is being used.**

- **Note that the stochastic version of this problem is a dual problem to the Kalman filter problem. Both require solution of a matrix Riccati equation.**
- **One can solve for K a priori. But if one wants to use on-line system identification, then one has to solve the Riccati equation on line.**

- **Model Predictive Control (MPC). Consider discrete time. At time step k, compute the control history to minimize a quadratic cost over the next q time steps**

$$J_k = \sum_{i=k}^{k+q} [x^T((i+1)T)Qx((i+1)T) + u^T(kT)Ru(kT)]$$

Then apply the first time step of this solution. Repeat the next time step. If q is small enough this can take less real time computation. Also has advantages when applied with adaptive updates of the model.

STOCHASTIC MODELS

- **One can include measurement noise and plant noise in one's modeling of the world dynamics**

$$\dot{x} = Ax + Bu + \gamma_p$$
$$y = Cx + \gamma_m$$

Consider each noise to be zero mean, Gaussian, and to be white (uncorrelated from one time to another, no matter how close the times are – otherwise there is some ability to predict the noise and this should be included in the model dynamics)

$$E[\gamma_p(t)] = 0, \quad E[\gamma_m(t)] = 0$$
$$E[\gamma_p(t)\gamma_p^T(\tau)] = \Gamma_p \delta(t-\tau), \quad E[\gamma_m(t)\gamma_m^T(\tau)] = \Gamma_m \delta(t-\tau)$$

THE KALMAN FILTER

- **You pick Γ_p, Γ_m and the Kalman filter picks the matrix F.**
- **There are various optimization criteria that produce the same filter.**
  - **One is to find the conditional expected value of $x(t)$ based on all available data**

$$\hat{x}(t) = E\{x(t) \big|\ y(\tau), u(\tau);\ \ 0 \le \tau \le t\}$$

  - **Or, one can ask for $\hat{x}(t)$ to be generated by a linear transformation of past data such that the estimation error $\tilde{x}(t) = x(t) - \hat{x}(t)$ has minimum trace of the covariance $E[\tilde{x}^T(t)\tilde{x}(t)] = trace[\tilde{x}(t)\tilde{x}^T(t)] = trace P(t)$**
- **The solution is**

$$F = PC^T\Gamma_m^{-1}$$
$$\dot{P} = AP + PA^T + \Gamma_p - PC^T\Gamma_m^{-1}CP$$

where the initial condition is the estimate of the covariance of a specified initial state. In practice, everyone uses the steady state solution to this equation.

- **Comments:**
  - **One often has a reasonable basis to pick Γ_m**
  - **The plant noise model is a pure fake**
    - **the actual plant noise can be due to missing modes in the model,**
    - **due to model errors,**
    - **due to disturbances such as gravity torque on a robot arm following a path**
    - **it is very unlikely that plant noise would be white**
  - **Note that if you pick $\Gamma_p = 0$, then the Kalman filter turns off, $F = 0$. Watch noisy measurements long enough on a completely deterministic system, and you know the state perfectly – no need to keep filtering.**

THE BOTTOM LINE

- **Γ_p and to a lesser extent Γ_m are just design parameters to adjust to get good performance.**
- **Γ_p basically determines how far back in time one still uses data to create the current state estimate – controls "memory length"**
- **Whether it is noise or the dynamics of the system evolving in time, adjust Γ_p so that the data used is still relevant**

LQG CONTROLLER DESIGN FOR STOCHASTIC MODELS

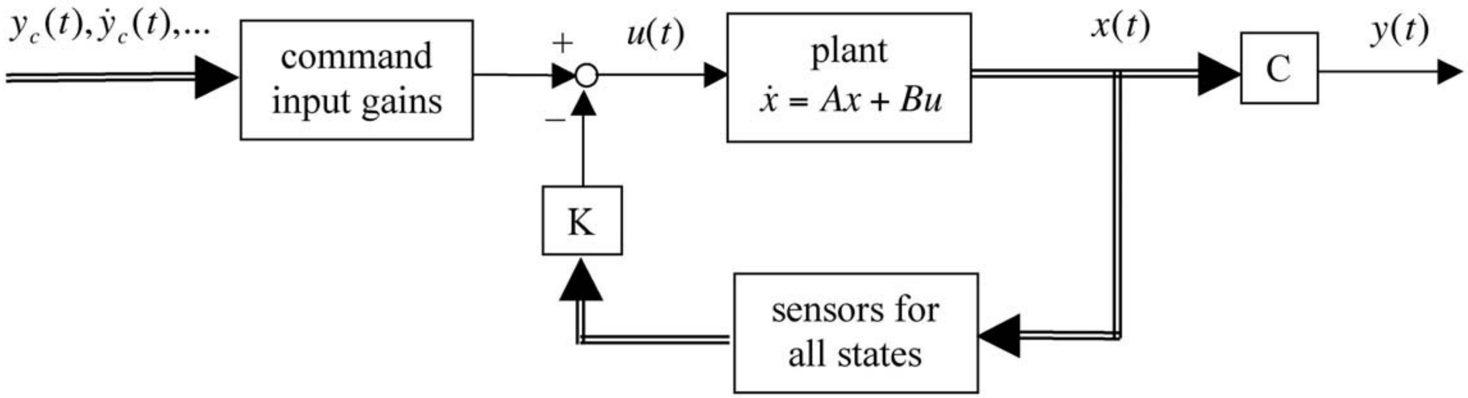
- **The LQR quadratic cost must be modified – with plant noise the integral of the square of the state goes to infinity. Instead, one can minimize**

$$J = \lim_{t \to \infty} E\{x^T(t)Qx(t) + u^T(t)Ru(t)\}$$

- **Stochastic separation principles apply to various problems, minimize the cost as if it is deterministic, use the Kalman result for the state in the result, and one minimizes the expected value of the cost.**

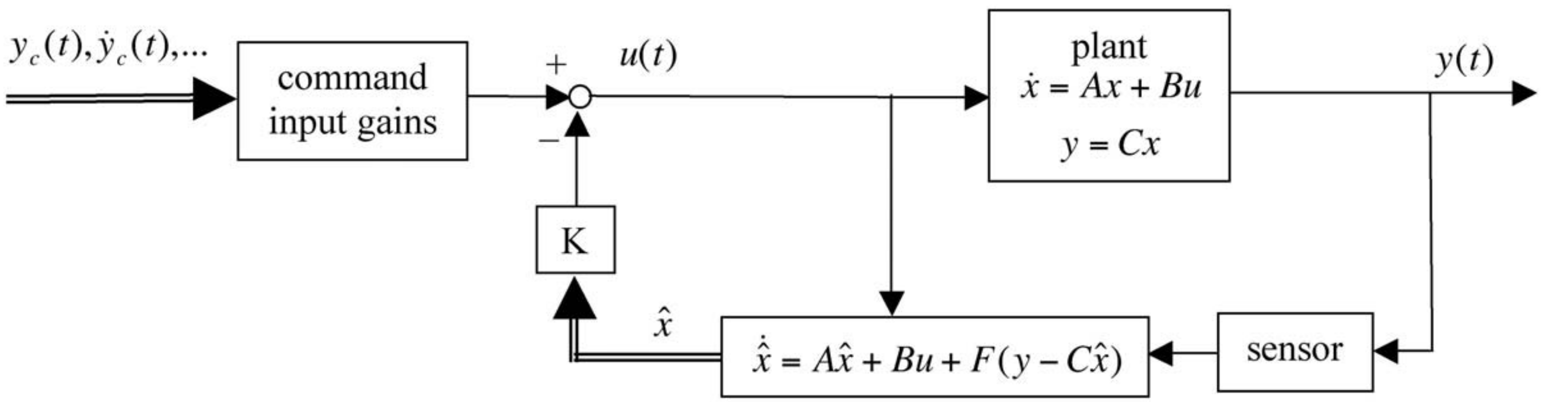
BLOCK DIAGRAM STRUCTURE OF STATE FEEDBACK CONTROL

- **The block diagram structure for state feedback control systems is slightly different than for classical control systems. If the <u>whole state is measured</u> then it looks like the following**



- The double line arrows refer to multivariable quantities.
- The diagram considers *y* and *u* a scalars as before.
- Control using state space models allows *y* and *u* to be vector valued if desired.
- As discussed, one can pick many forms for the command input, and it could be vector valued (perhaps more naturally done in digital control where one can use a sequence of outputs instead of derivatives).

- **If one needs to use an observer the structure becomes**



CONTROL AND OBSERVATION SPILLOVER

- **Whatever model one uses, in the real world there is always some missing dynamics -- called parasitic poles, residual modes, etc.**

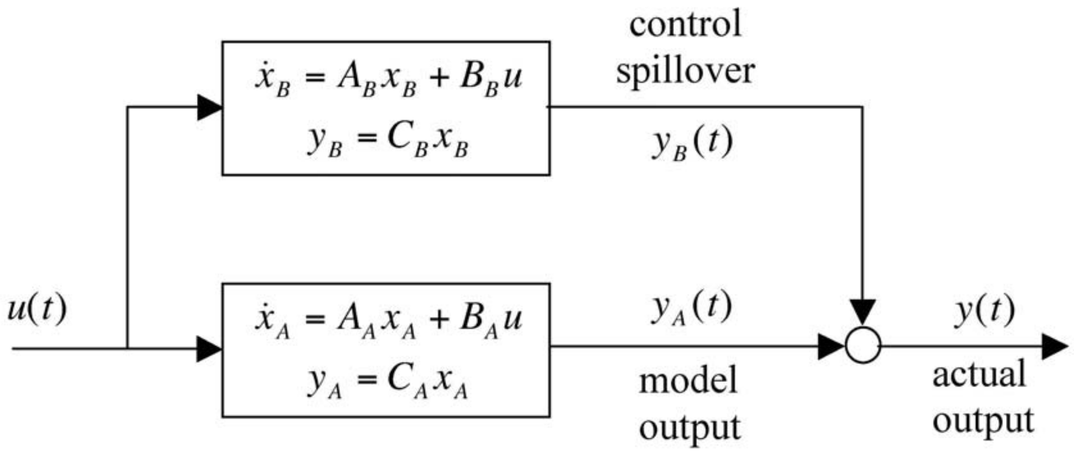
- **Suppose we make a transformation of state variable to diagonalize the state equation including all parasitic poles. Let $x_A(t)$ contain all states for our model, and $x_B(t)$ contain the states for all parasitic poles**

$$\begin{bmatrix} \dot{x}_A \\ \dot{x}_B \end{bmatrix} = \begin{bmatrix} A_A & 0 \\ 0 & A_B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix} + \begin{bmatrix} B_A \\ B_B \end{bmatrix} u$$

$$y = y_A + y_B = \begin{bmatrix} C_A & C_B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix}$$

Ref. [1]

- **Control spillover (a term from the control of large flexible spacecraft) – the control excites modes that are not in the model but are in the measurements**



- **Observation spillover – the observer sees data corrupted by modes not in the model**

$$\dot{\hat{x}}_A = A_A \hat{x}_A + B_A u + F(y_A - C_A \dot{\hat{x}}) + F y_B$$

- **Control spillover is zero if $B_B = 0$. Observation spillover is zero if $C_B = 0$.**

  - **Control spillover alone cannot destabilize a system. It excites a part of the system that is not seen by the observer.**
  - **Observation spillover alone cannot destabilize a system. It creates an additive disturbance to the observation equation, but the disturbance does not propagate around the loop.**
  - ***But when both are present, it is easy to have a control system be unstable.***

PDE vs. ODE ISSUES

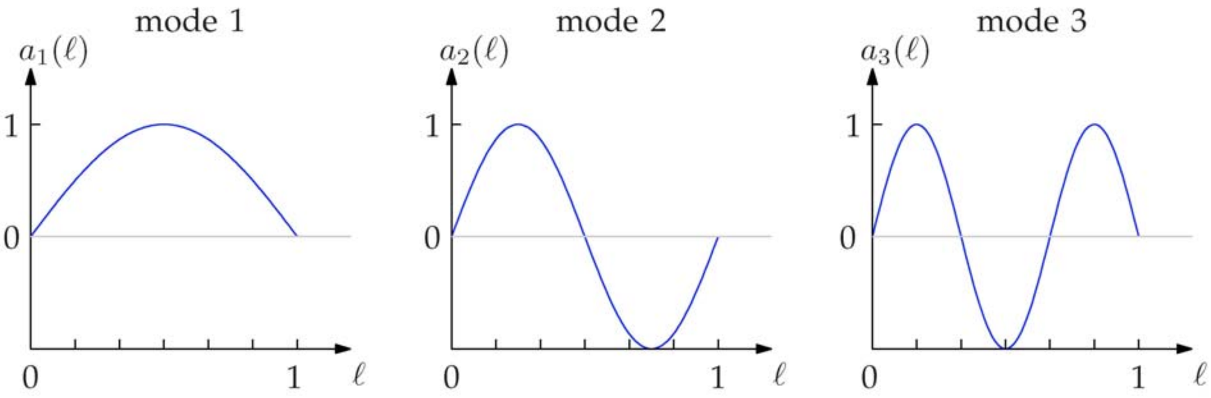
- **Following are some comments from the control of large flexible spacecraft field. There are some similarities to the plasma control problem.**
  - **Flexible structures are governed by PDE's. In practice, must be replaced by ODE's to design controllers.**
  - **The ODE dimension is infinite, so the number of modes retained in the control system design must be truncated at some point.**
  - **Therefore there are always missing modes that would cause control and observation spillover.**
  - **Structures are lightly damped, so roots are near imaginary axis. As you move roots in the model toward the left to get more damping, it is common that the first mode missing in the model goes the other direction, toward instability.**

- **In order to improve the situation:**
  - **One can use a feedback control that takes energy out of the system, independent of the model, so that all the missing modes get some extra damping, and are further from the stability boundary.**

- **The simplest version is to measure velocity at a point on the structure, and co-located with the sensor is an actuator that pushes in the opposite direction, proportional to velocity. Use this many places in the structure.**
- **Sometimes called a low authority controller, it does not produce good performance.**
- **One then puts a high authority controller, designed with a model, around the system. And residual modes are less likely to destabilize.**

- **Because of the distributed nature of the problem, there are more tricks that one can use:**
  - **An observer filters the time histories of the measurements to determine states.**
  - **If you have sensors distributed spatially, one can also do spatial filtering to help determine states. A combination of both can be beneficial.**

- **Consider putting sensors on a vibrating beam with pin joints at each end. And suppose there are three modes of vibration, with mode shapes** $a_1(\ell), a_2(\ell), a_3(\ell)$ **and amplitudes at time** t**,** $x_1(t), x_2(t), x_3(t)$



- **If we have three measurements taken at positions** ℓ_1, ℓ_2, ℓ_3**, the sensor outputs would be**

$$y_1(t) = a_1(\ell_1)x_1(t) + a_2(\ell_1)x_2(t) + a_3(\ell_1)x_3(t)$$
$$y_2(t) = a_1(\ell_2)x_1(t) + a_2(\ell_2)x_2(t) + a_3(\ell_2)x_3(t)$$
$$y_3(t) = a_1(\ell_3)x_1(t) + a_2(\ell_3)x_2(t) + a_3(\ell_3)x_3(t)$$

$$y(t) = C_D \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} \quad ; \quad \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = C_D^{-1} y(t)$$

This gives the 3 position state variables. One only needs to get velocities for each.

- **Suppose that we are using 2 modes in the model and we want no spillover from mode 3. Then place actuators at** $\ell = 1/3$ **and** $\ell = 2/3$**, the nodes of the third mode.**
- **Suppose we have two measurements** $y_1(t), y_2(t)$ **that are not at nodes of mode 3. We can combine the measurements to eliminate contribution from mode 3 producing no observation spillover**

$$y_1(t) = a_1(\ell_1)x_1(t) + a_2(\ell_1)x_2(t) + a_3(\ell_1)x_3(t)$$

$$y_2(t) = a_1(\ell_2)x_1(t) + a_2(\ell_2)x_2(t) + a_3(\ell_2)x_3(t)$$

$$y_4(t) = y_1(t) - \frac{a_3(\ell_1)}{a_3(\ell_2)} y_2(t)$$

Ref. [1]

NONLINEAR CONTROLLERS

- **Linear control methods address a large class of systems. Control methods for nonlinear systems are normally limited to some rather specialized class of nonlinearities.**
- **Neural networks is the only nonlinear modeling method that applies to a large class of nonlinear systems. Most likely takes too long to tune, and too slow to adapt to changing system behaviors for plasma problems.**
- **If you know the form of the nonlinear terms, one can use feedback linearization, i.e. make the control have a term that cancels the nonlinear term in the dynamics. Then use linear control design methods. A main difficulty is the lack of control over actuator saturation.**
- **George Leitman has methods that apply when one knows that the nonlinear terms are bounded, e.g.** $\sin(x(t))$**.**
- **Liapunov based methods -- one needs to know that a Liapunov function is satisfied. Sliding mode control.**

NONLINEAR OBSERVERS

- **Just as in control design, nonlinear observers have significant limitations, and often apply only to certain classes of systems. A survey article by Misawa and Hedrick lists the following approaches (Ref. [2]):**

- Extended Kalman filter – heavy real time computation burden
- Constant gain extended Kalman filter – fixes real time burden at the expense of performance
- Statistically linearized filter – like describing functions in control, can appy to a specific nonlinear function to linearize e.g. in frequency response
- Global linearization methods – see if you can find a nonlinear transformation that makes the equations linear
- Extended linearization methods – make a set of filters for different operating points in state space, and use the closest one
- A robust adaptive observer – for systems with nonlinear coefficients known to be bounded
- Sliding observers – satisfy positive real assumption or other conditions

• Comments in the summary section of the survey

- All methods have drawbacks
- Some require perfect knowledge of the system
- Some require heavy real time computation
- Some have no stability guarantee
- Robustness can be narrowly restricted, or hard to know if conditions satisfied
- All statements are relative to use of observer only
- Future research into nonlinear observers used in closed loop control is essential

• Field is not mature enough to be of much use in plasma control problems

• One should settle for making system identification updates and adjusting linear observers periodically using the new model

COMMENTS AND RECOMMENDATIONS

- **Attached is a list of 32 papers, both theory and experiments, doing feedback control of plasma instabilities, written by Amiya Sen of Columbia University, published over the last 33 years. Three of these papers are attached, including a 1975 paper dealing with controllability and observability, properties required by the quadratic cost/Kalman filter approach.**

- **The OKID (observer/Kalman filter identification) algorithm identifies a dynamic model of the system, and at the same time finds the Kalman filter gain *F*, directly from data (Ref. [3]). Software available from NASA. This avoids some of the modeling guesswork.**

- **Optimal control using a stochastic model to represent the very noisy plasma environment, was studied for the D-IIID resistive wall mode instability in Ref. [3], attached.**

- **There are issues of nonlinearities and robustness. I suggest that the best method to deal with both issues is to use a linear model, but use model updating from input-output data, at an update rate related to the time scale for evolution of the system dynamics.**

- **Ref. [4] does adaptive stochastic optimal control, using the same control as above, but including model updates using extended least squares identification of an ARMAX model. The ID result updates the control gain and the Kalman filter gain.**

- **Real time computation limitations are a serious issue for feedback control in plasma instabilities. The adaptive optimal control approach above is fast enough to do real time, but there is not a substantial margin.**

- **One might not need to explicitly use an observer. Ref. [5] studies adaptive stochastic output (instead of state) feedback control using a quadratic cost, applied to the D-IIID wall mode instability. The approach bypasses the Kalman filter, and requires solution of a Diaphantine equation for the updates of the control law. Simple tests indicate the method saves about 1/3 in real time computation.**

- **There are also methods by Prof. M. Q. Phan that use data to directly identify from data the control gains to minimize a quadratic cost, without identifying the system or using an observer.**

- **Generalized model predictive control using model updates is a candidate for handling the complex and changing nature of the plasma environment, see Refs. [7,8,9]. These methods have been very effective in handling very complex noise cancellation problems, and might be helpful in the complex plasma environment as well. The methods again bypass the state reconstruction, in fact do not use a state space model. A sequence of inputs and outputs can produce the feedback gains, which multiply a set of previous inputs and outputs. The main limitation is that the controllers are normally high dimensional, aggravating the real time computation difficulty. Work is currently in progress to study the approach.**

REFERENCES

[1] R. W. Longman, "Annihilation or Suppression of Control and Observation Spillover in the Optimal Shape Control of Flexible Spacecraft," *The Journal of the Astronautical Sciences*, Vol. 27, No. 4, October-December 1979, pp. 381-399.

[2] E. A. Misawa and J. K. Hedrick, “Nonlinear Observers – A State-of-the-Art Survey,” ASME Transactions, Journal of Dynamic Systems, Measurement, and Control, Vol. 111, 1989, pp. 344-351.

[3] J.-N. Juang, M. Phan, L. G. Horta, and R. W. Longman, "Identification of Observer/Kalman Filter Markov Parameters: Theory and Experiments," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 2, 1993, pp. 320-329.

[4] A. K. Sen, M. Nagashima, and R. W. Longman, “Optimal Control of Tokamak Resistive Wall Modes in the Presence of Noise”, *Phys. Plasmas*, **11**, 4350-4357, 2003.

[5] Z. Sun, A. K. Sen, and R. W. Longman, “Adaptive Optimal State Feedback Control of Resistive Wall Modes in Tokamaks,” *Phys. Plasmas*, **13**, 012512, 2006.

[6] Z. Sun, A. K. Sen, and R. W. Longman, “Adaptive Stochastic Output Feedback Control of Resistive Wall Modes in Tokamaks,” to appear.

[7] J. L. Bialasiewicz, L. G. Horta, M. Q. Phan, “Identified Predictive Control,” *Proceedings of the 1994 American Control Conference*, Baltimore, MD, 1994.

[8] J.-N. Juang and K. W. Eure, *Predictive Feedback and Feedforward Control for Systems with Unknown Disturbances*, NASA/TM-1998-208744, December 1998.

[9] R. S. Darling and M. Q. Phan, “Predictive Controllers for Simultaneous Tracking and Disturbance Rejection,” *Proceedings of the 2004 AIAA Guidance, Navigation, and Control Conference*, Providence, RI, 2004.

Feedback Control of Plasma Instabilities
(Papers published by Sen with his students and colleagues)

"Feedback Stabilization of a Multimode Two-Stream Instability", R. M. Chervin and A. K. Sen, J. Plasma Phys., **15**, 387-397(1973).

"Feedback Control of Ionization Instability", A. K. Sen, Energy Conversion, **13**, 13-17 (1973).

"Variable Gain-Phase Method for Feedback Stabilization of Multimode Plasma Instability", A. K. Sen, J. Plasma Phys., **16**, 509-515 (1974).

"Plasma Transfer Functions", A. K. Sen, IEEE Trans. Plasma Sci., **PS--3**, 163-167(1975).

"Observability and Controllability of Plasma Instabilities", A. K. Sen, Phys. Fluids, **18**, 1187-1191 (1975).

"Transfer Functions of Hyperbolic Distributed Parameter Systems", A. K. Sen, Trans. ASME, J. Dynamic Systems, Measurements and Control, **98**, G, 318 (1976).

"Dynamic Feedback for Control of Multimode Plasma Instabilities", A. K. Sen, J. Plasma Phys., **20**, 1-16(1978).

"State Feedback Control of Multimode Plasma Instabilities", A. K. Sen, IEEE Trans. Plasma Sci., **PS-7**, 116-119(1979).

"Feedback Control of a Class of Plasma Instabilities", A. K. Sen, IEEE Trans. Automa. Contr., **AC--24**, 272-276(1979).

"Control of a Trapped Electron Instability by a Modulated Neutral Beam", M. Ghassemi and A. K. Sen, J. Plasma Phys., **23**, 273-282(1981).

"Characteristics of a Modulated Neutral Beam as r Remote Suppressor for Plasma Instabilities", M. Ghassemi and A. K. Sen, J. Plasma Phys., **23**, 219-225(1981).

"Optimal Control of Trapped Electron Instability by a Modulated Neutral Beam", M. Ghassemi and A. K. Sen, IEEE Trans. Plasma Sci., **PS-9**, 41-45(1981).

"Production, Study and Feedback Control of a Collisionless, Curvature and Rotationally Driven, Trapped Particle Mode", A. K. Sen, R. Scarmozzino, A. Sekiguchi and G. A, Navratail, Proceedings of the Twelfth Inst. Conf. on Plasma Physics and Controlled Nuclear Fusion Res, IAEA, Vol. 3, 495-505 (1989).

"Axial Feedback Stabilization of Flute Modes for Mirror Machines", B. K. Kang, M. A. Lieberman and A. K. Sen, IEEE Trans. Plasma Sci., **PS-16**, 27-38 (1988).

"Halo Feedback Control of Trapped Particle Instabilities", A. Sekiguchi, A. K. Sen and G. A. Narvatail, IEEE Trans. Plasma Sci., **PS-18**, 685-686 (1990).

"Feedback Control of the Thermonuclear Thermal Instability with a Poloidal Divertor Suppressor", A. K. Sen, M. Lisak and D. Anderson, Nuclear Fusion, **31**, 2392-2396 (1991).

"Feedback-Modulated Ion Beam Stabilization of a Plasma Instability", P. Tham, A. K. Sen and R. Greaves, Phys. Rev. Lett., **67**, 204-207 (1991).

"Compact Ion Beam Source for Feedback Control of Plasma Instability", A. Sekiguchi, A. K. Sen, P. Tham, G. A. Narvatil and R. G. Greaves, Rev. Sci. Instrum., **63**, 4427 (1992).

"Feedback Stabilization of Plasma Instabilities Using a Modulated Ion Beam", P. Tham and A. K. Sen, Phys. Fluids B, **4**, 3997-4001(1993).

"Robustness of the Feedback-Modulated Control Scheme of Plasma Instabilities to Parameter Variations", P. Tham and A. K. Sen, IEEE Trans. Plasma Sci., **PS-21**, 588-592 (1993).

"Feedback Suppression of the Radiative Condensation Instability", A. K. Sen, Phys. Fluids B, **5** 3058-3065 (1992).

"Remote Feedback Stabilization of Tokamak Instabilities", A. K. Sen, Phys. Plasmas, **1**, 1479-1488 (1994).

"Feedback Reduction of Anomalous Transport", J. S. Chiu, M. Tinkle and A. K. Sen, Phys. Rev. E, **54**, 2158-2161 (1996).

"Feedback Control of Major Disruptions in Tokamaks", A. K. Sen, Phys. Rev. Lett., 76, 1252 (1996).

"Feedback Control of Tearing Modes Via Modulated Beams", A. K. Sen, R. Singh, A. Sen and P. K. Kaw, Phys. Plasmas, **4**, 3217 (1997).

"Multimode Control of MHD Instabilities in Tokamaks", A. K. Sen, Phys. Plasmas, **5**, 2956 (1998).

"Control and Diagnostic Uses of Feedback", A. K. Sen, Phys. Plasmas, **7**, 1759 (2000).

"Feedback Reduction of Drift Wave Fluctuations", C. F. Figarella, A. K. Sen, et al., Plasma Phys. Contr. Fusions, **45**, 1294 (2003).

"Optimal Control of Tokamak Resistive Wall Modes in the Presence of Noise", A. K. Sen, M. Nagashima and R. W. Longman, Phys. Plasmas **11**, 4350-4357 (2003)

"Feedback Control of Kink and Tearing Modes via Novel ECH Modulation", A. K. Sen, Plasma Phys. Contr. Fusions, **46**, L41 (2004).

"Adaptive Optimal State Feedback Control of Resistive Wall Modes in Tokamaks", Z. Sun, A. K. Sen, and R. W. Longman, Phys. Plasmas **13**, 012512 (2006)

"Adaptive Stochastic Output Feedback Control of Resistive Wall Modes in Tokamaks", Z. Sun, A. K. Sen, and R. W. Longman, to appear.