Lecture 14. Systems Concepts

Reading Assignment:

TMB2 Section 3.1

Note: To prepare for Lecture 15 make sure that you have mastered the basic ideas on eigenvectors and eigenvalues that are briefly reviewed in TMB2 Section 3.1.

A System is Defined by Five Elements

The set of *inputs*

The set of *outputs*

(These involve a choice by the modeler)

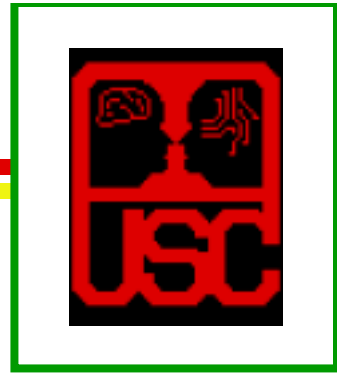
The set of *states:* those internal variables of the system — which may or may not also be output variables — which determine the relationship between input and output.

state = the system's "internal residue of the past"

The *state-transition function* : how the state will change when the system is provided with inputs.

The *output function* : what output the system will yield with a given input when in a given state.

Finite Automata and Identification Theory



Formally, we describe an automaton by the sets
X, Y and Q of inputs, outputs and states, respectively,
together with the

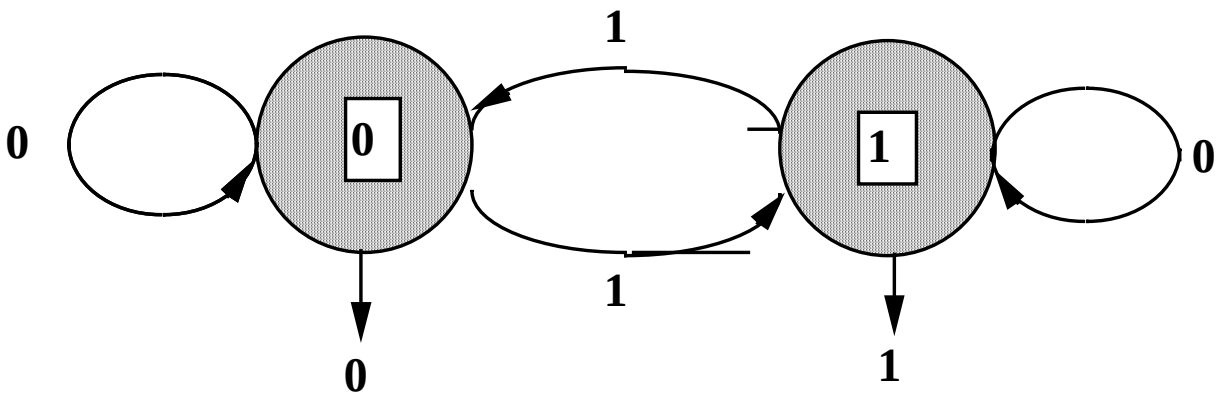
next-state function δ: Q x X $\rightarrow$ Q and the

output function β: Q $\rightarrow$ Y.

If the automaton in state q receives input x,

next state will be $\delta(q, x)$

next output will be $\beta(q)$.



External versus Internal Descriptions

We distinguish between

a system characterized by an internal structure or process, &

a system characterized by an external pattern of behavior

X*: the set of strings of inputs (input sequences)

wx: following string w by the input x

Extend δ to a map $\delta^*: Q \times X^* \rightarrow Q$ such that

$\delta^*(q, w)$ is the state obtained on starting in state q and reading in input string w.

$q \xrightarrow{\quad wx \quad} \delta^*(q, wx)$

$q \xrightarrow{\quad w \quad} \delta^*(q, w) \xrightarrow{\quad x \quad} \delta(\delta^*(q, w), x)$

If M is started in state q and fed input sequence w,

it will emit a sequence of outputs whose last element

$M_q(w)$ is just $\beta(\delta^*(q, w))$

The Identification Problem

It is thus straightforward to go from an internal description to the corresponding external description.

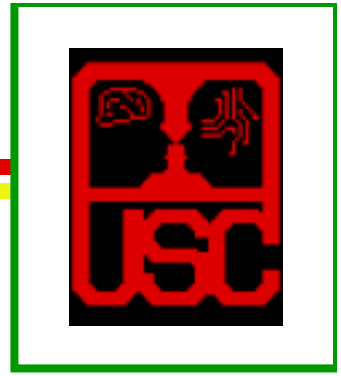
The reverse process is a special case of the identification problem. (See TMB Sec. 3.4.)

For an introduction to the formal theory, see Section 3.2 of *Brains, Machines and Mathematics*, Second Edition.

Exact computation of the minimal automaton consistent with noise-free input-output data.

The theory of adaptive neural nets provides one approach to approximate identification.

From Newton to Dynamic Systems



Newton's mechanics describes the behavior of a system on a continuous time scale.

Rather than use the present state and input to predict the *next state*, the present state and input determine *the rate at which the state changes*.

Newton's third law says that the force F applied to the system equals the mass m times the acceleration a

$$F = ma$$

Position x(t)

Velocity v(t) = $\dot{x}(t)$

Acceleration a(t) = $\ddot{x}(t)$

Newtonian Systems

According to Newton's laws, the state of the system is given by the position and velocity of the particles of the system.

We now use

u(t) for the input = force

y(t) (equals x(t)) for the output = position.

Note: In general, input, output, and state are more general than in the following, simple example.

State Dynamics

With only one particle, the state is the 2-dimensional vector

$$q(t) = \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}$$

Then $\frac{d}{dt} x(t) = \dot{x}(t)$ *and* $\frac{d}{dt} \dot{x}(t) = \frac{1}{m} u(t)$ *since* $u(t) = m\ddot{x}$

yielding the single vector equation

$$\dot{q}(t) = \begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} u(t)$$

The output is given by

$$y(t) = x(t) = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}$$

The point of the exercise: Think of the state vector as a single point in a multi-dimensional space.

Linear Systems

This is an example of a **Linear System:**

$$\dot{q} = A\,q + B\,u$$

$$y = C\,q$$

where the state q, input u, and output y are vectors (not necessarily 2-dimensional) and A, B, and C are linear operators (i.e., can be represented as matrices).

Generally a physical system can be expressed by

State Change: $q(t) = f(q(t), u(t))$

Output: $y(t) = g(q(t))$

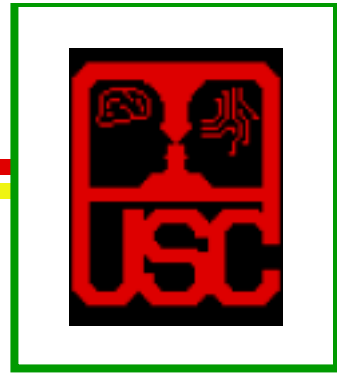
where f and g are general (i.e., possibly nonlinear) functions

For a network of leaky integrator neurons:

the state $m_i(t)$ = arrays of membrane potentials of neurons,

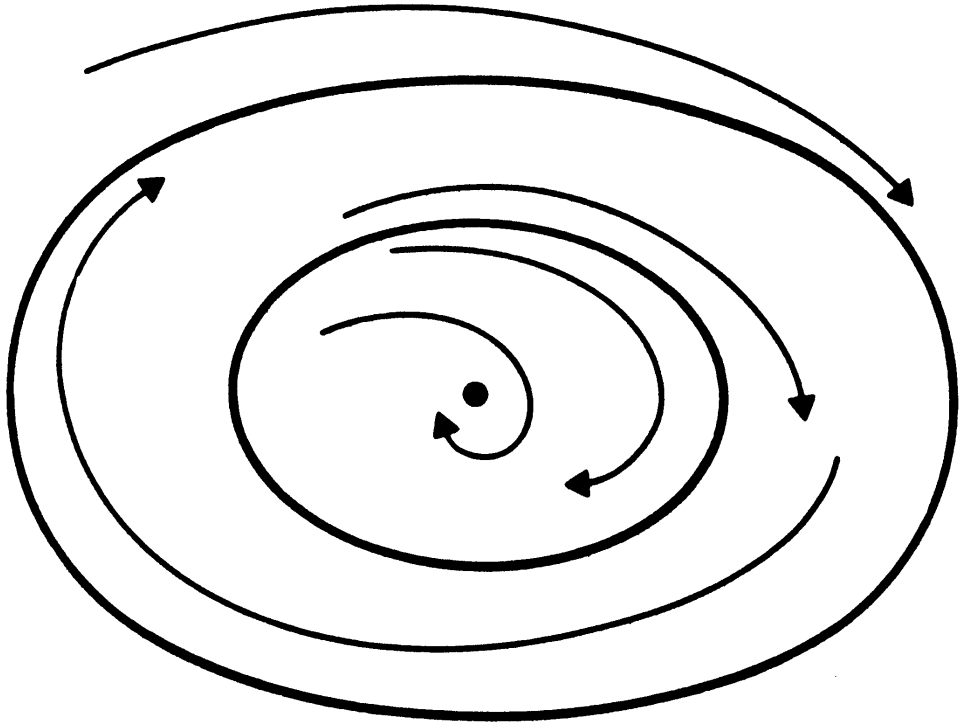
the output $M(t) = \sigma(m_i(t))$ = the firing rates of output neurons, obtained by selecting the corresponding membrane potentials and passing them through the appropriate sigmoid functions.

Attractors



For all recurrent networks of interest (i.e., neural networks comprised of leaky integrator neurons, and containing loops), given initial state and fixed input, there are just three possibilities for the asymptotic state:

1. The state vector comes to rest, i.e. the unit activations stop changing. This is called a **fixed point**. For given input data, the region of initial states which settles into a fixed point is called its **basin of attraction**
2. The state vector settles into a periodic motion, called a **limit cycle**.

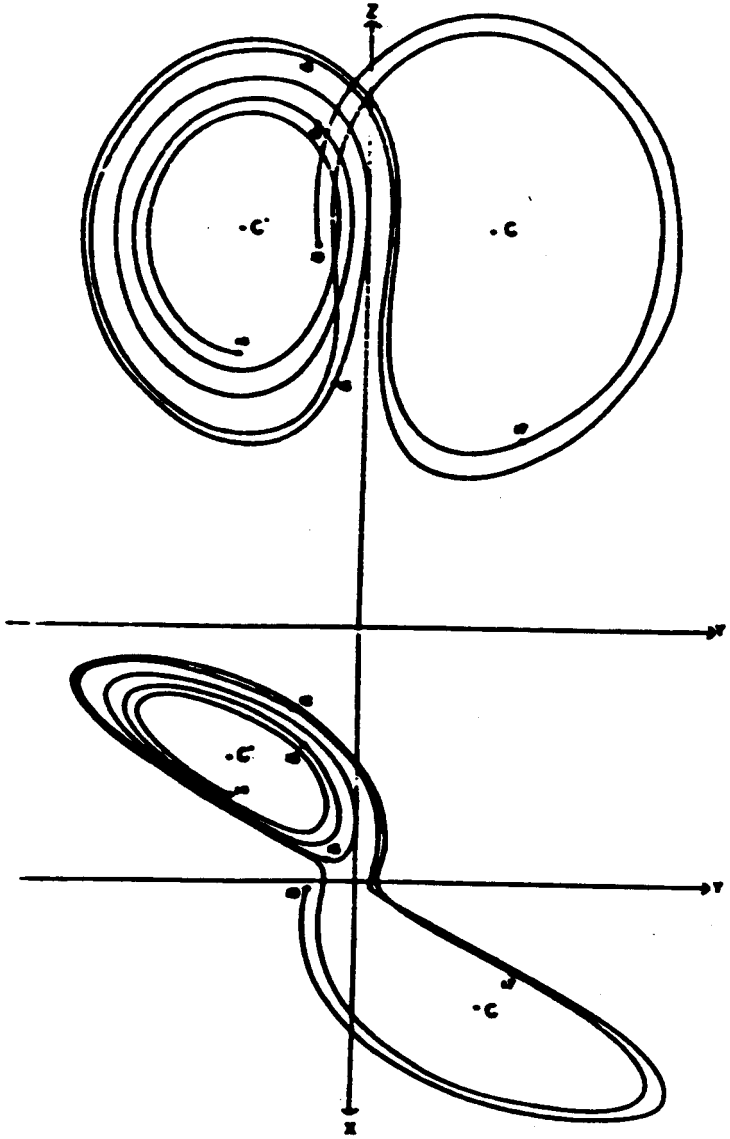


Strange attractors

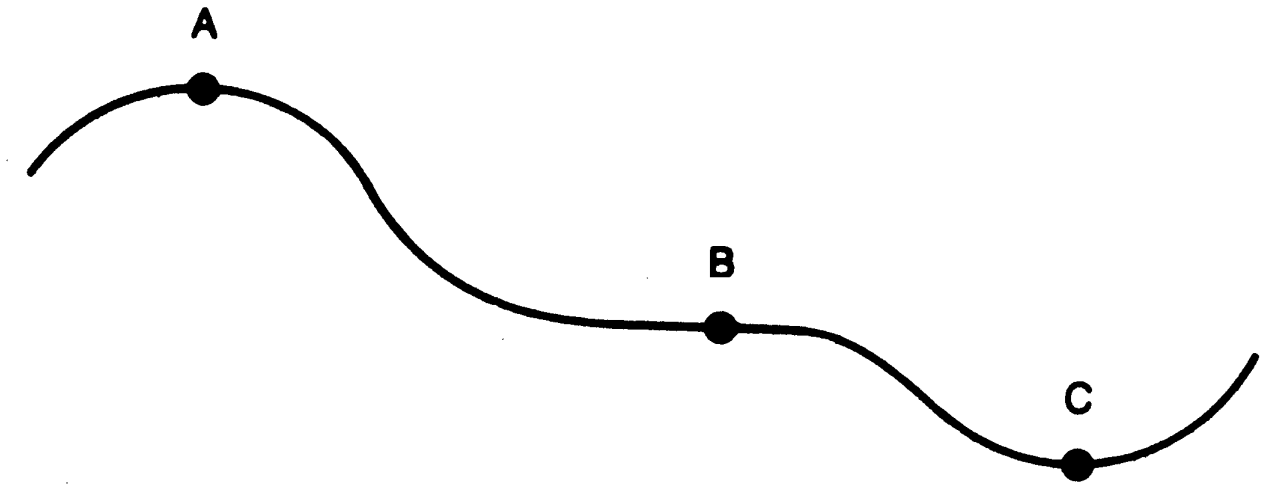
3. **Strange attractors** describe such complex paths through the state space that, although the system is deterministic, a path which approaches the strange attractor gives every appearance of being random.

Two copies of the system which initially have nearly identical states will grow more and more dissimilar as time passes.

Such a trajectory has become the accepted mathematical model of **chaos** and is used to describe a number of physical phenomena such as the onset of turbulence in weather.



Stability



The study of stability of an equilibrium is concerned with the issue of whether or not a system will return to the equilibrium in the face of slight disturbances:

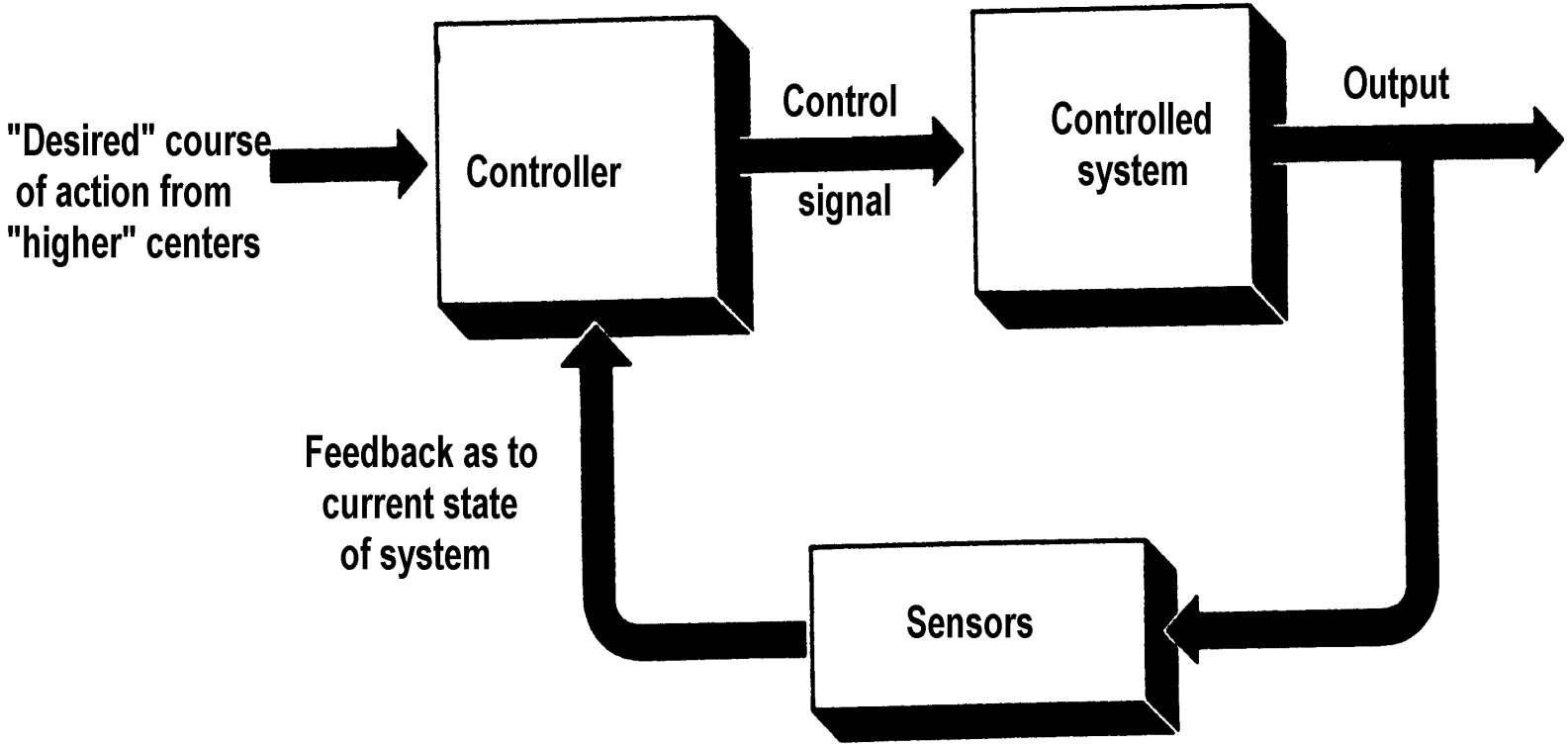
A is an unstable equilibrium

B is a neutral equilibrium

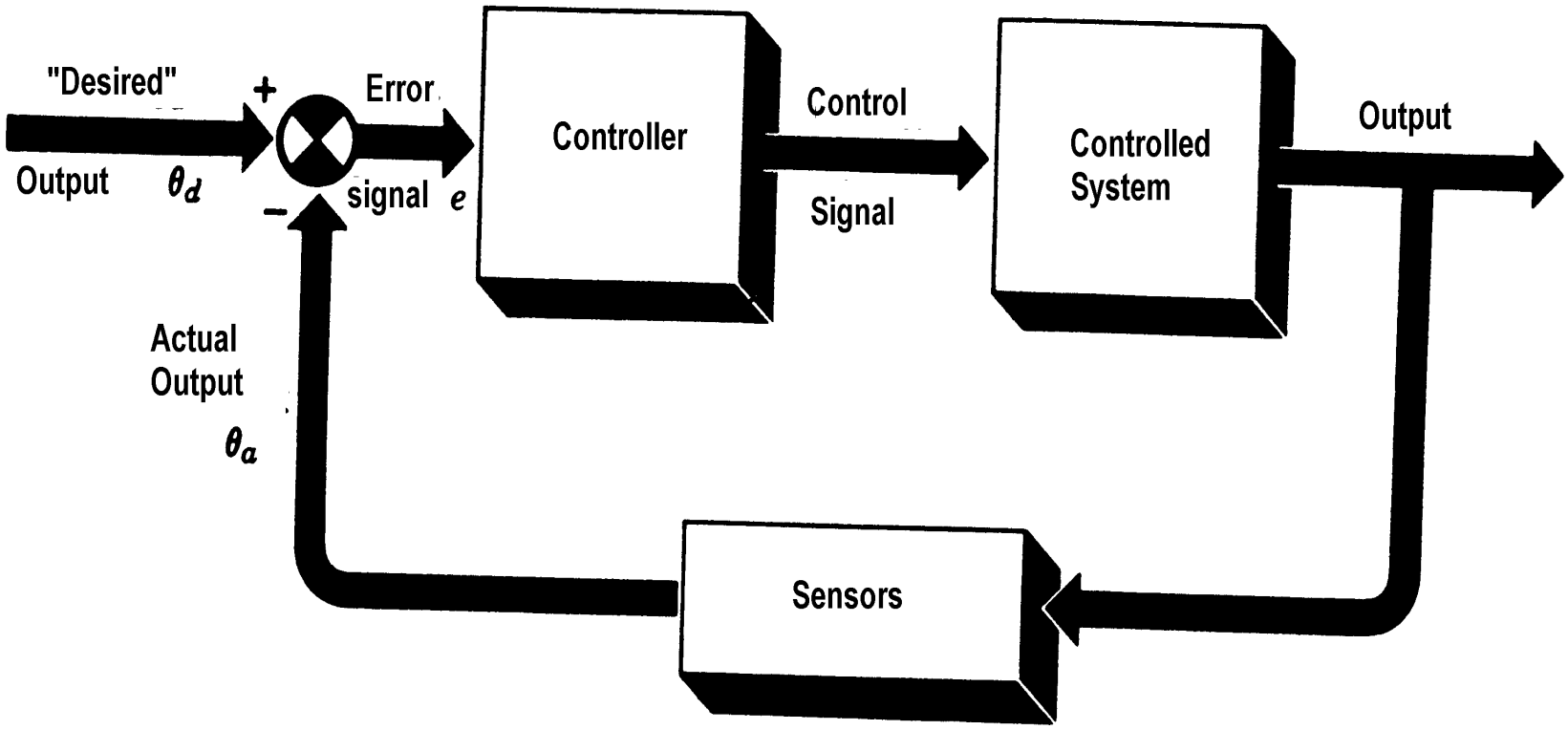
C is a stable equilibrium, since small displacements will tend to disappear over time.

Note: in a nonlinear system, a large displacement can move the ball from the basin of attraction of one equilibrium to another.

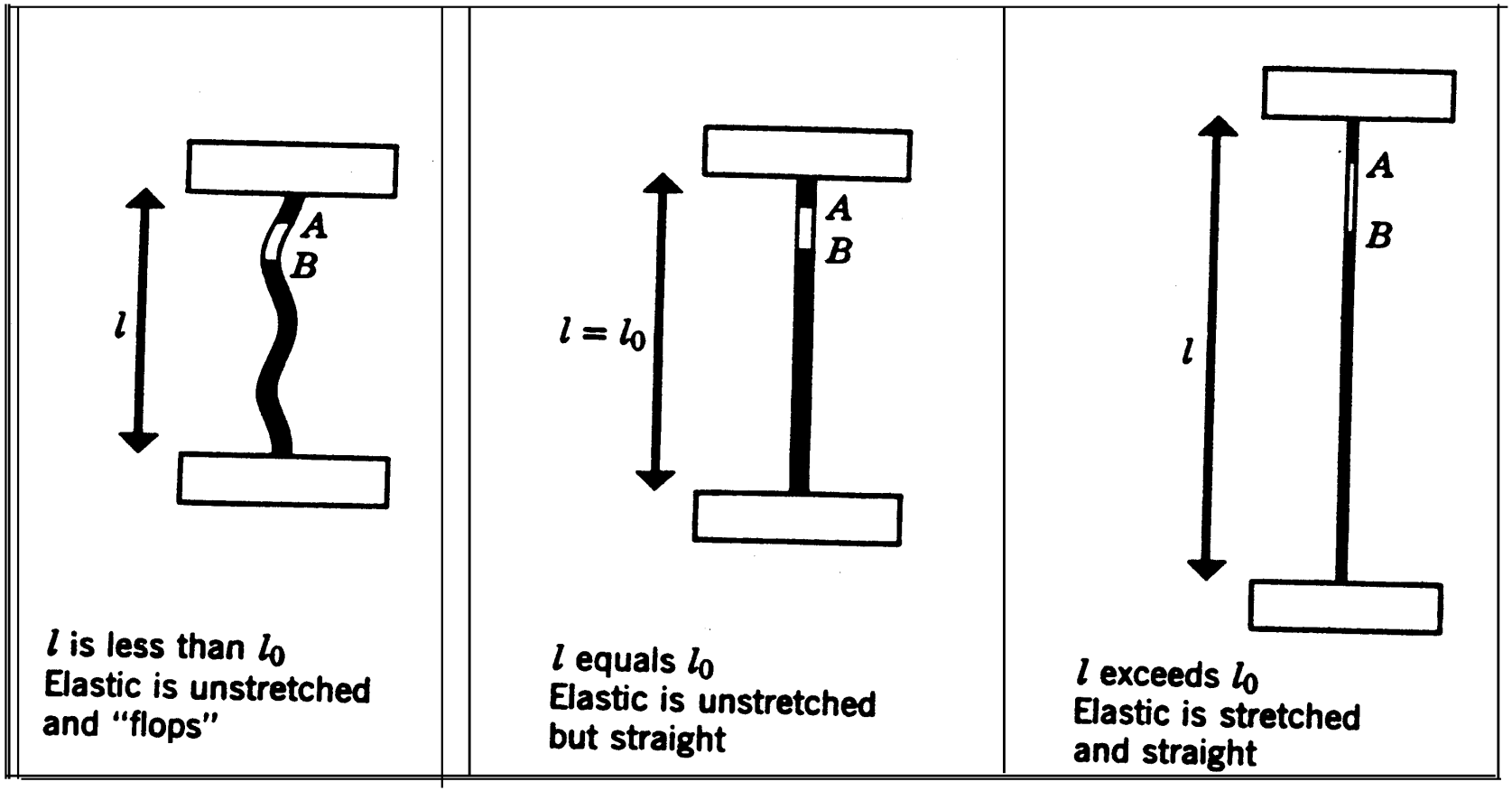
General Feedback Setup



Negative Feedback Controller (Servomechanism)

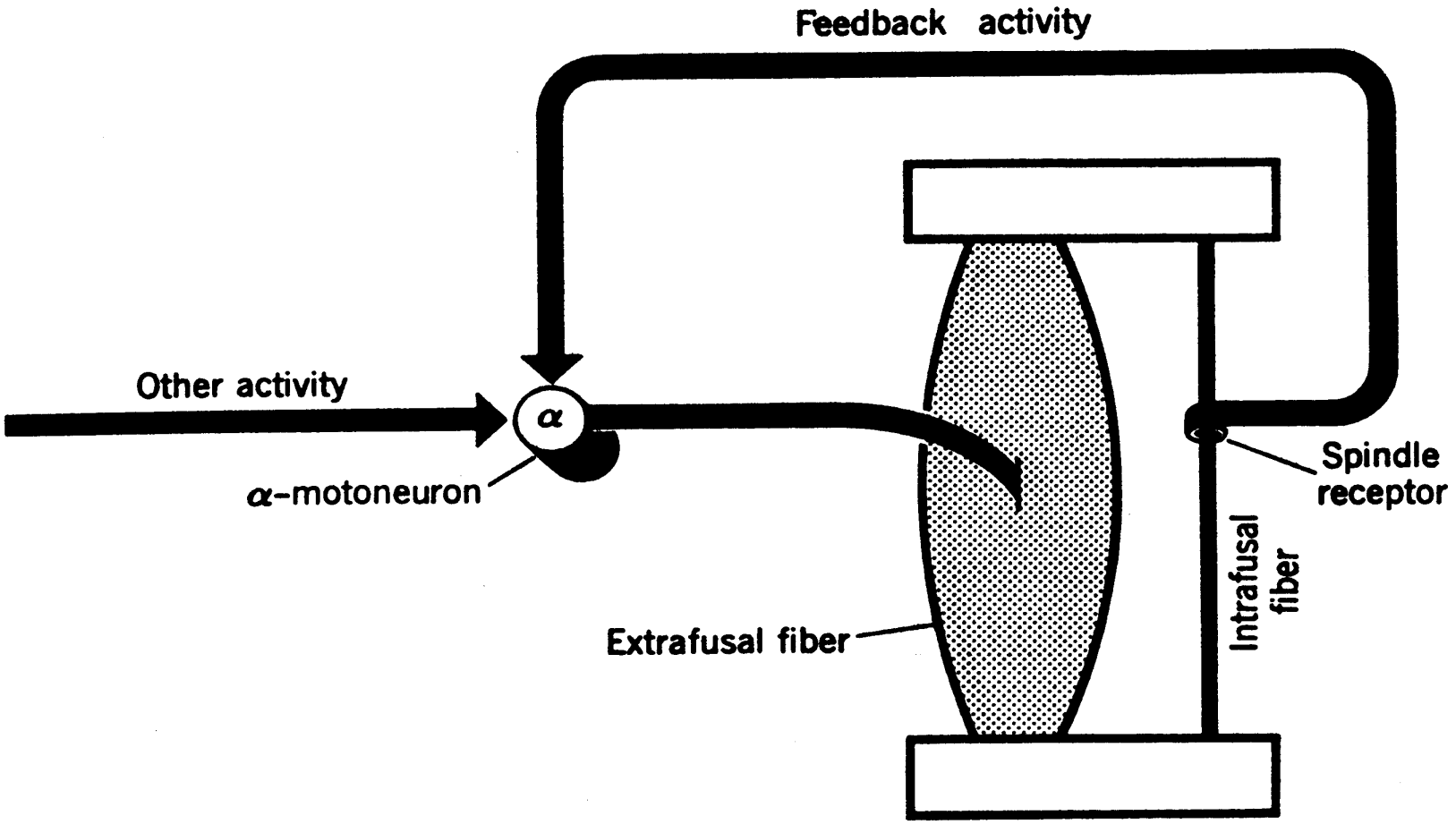


Judging the Stretching of an Elastic Band

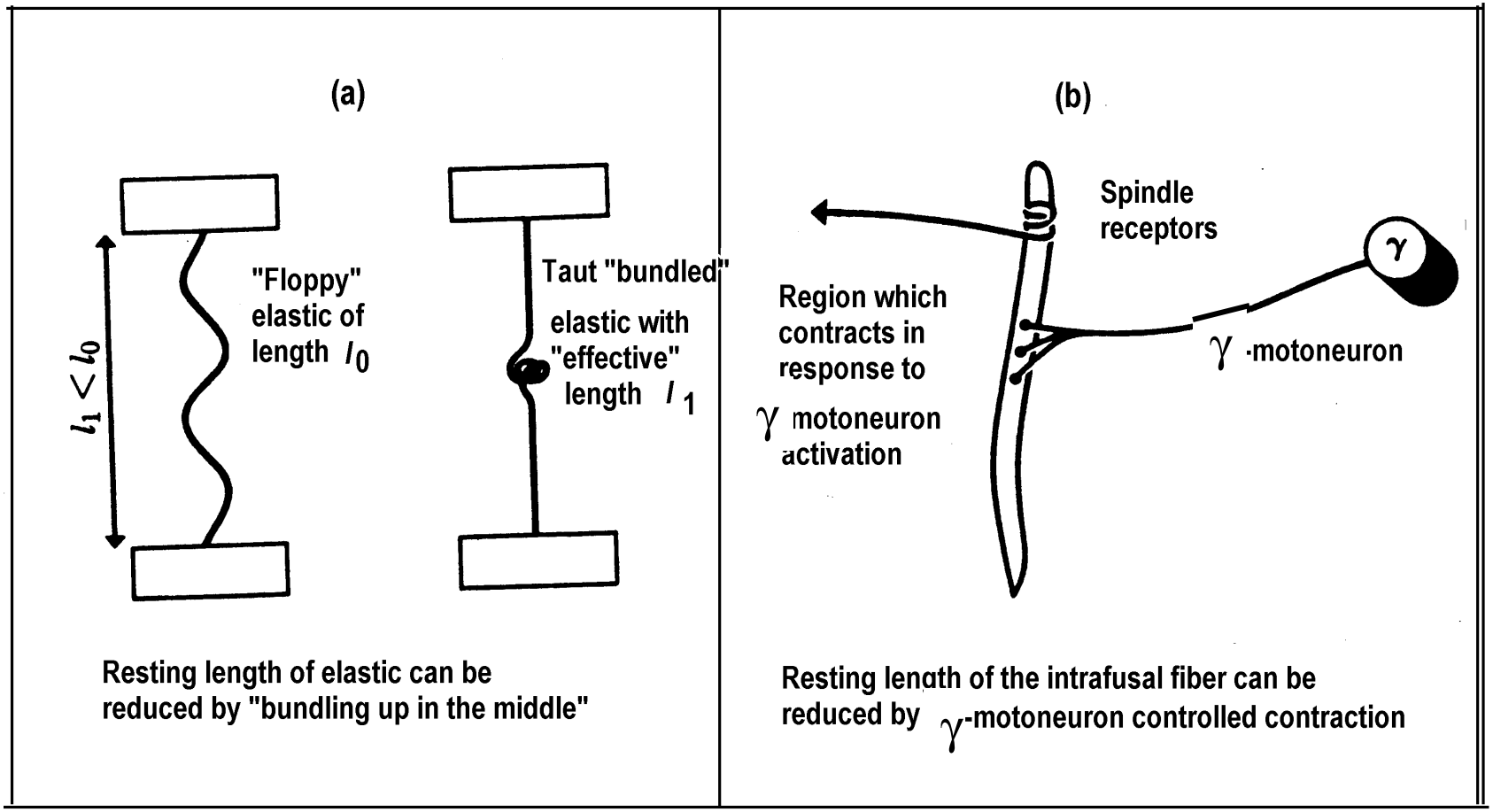


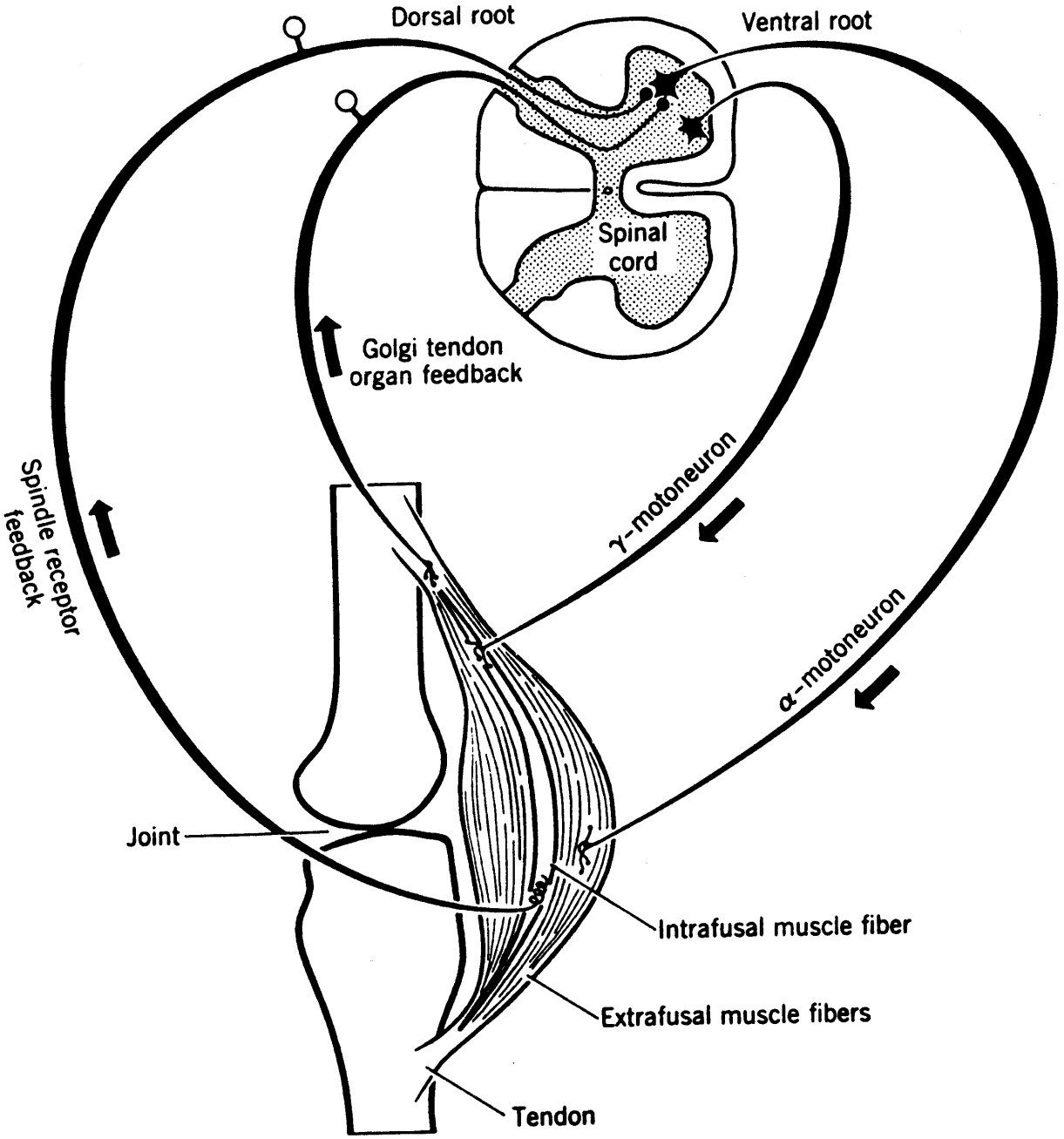
Using Spindles to Tell α-Neurons if a Muscle Needs to Contract

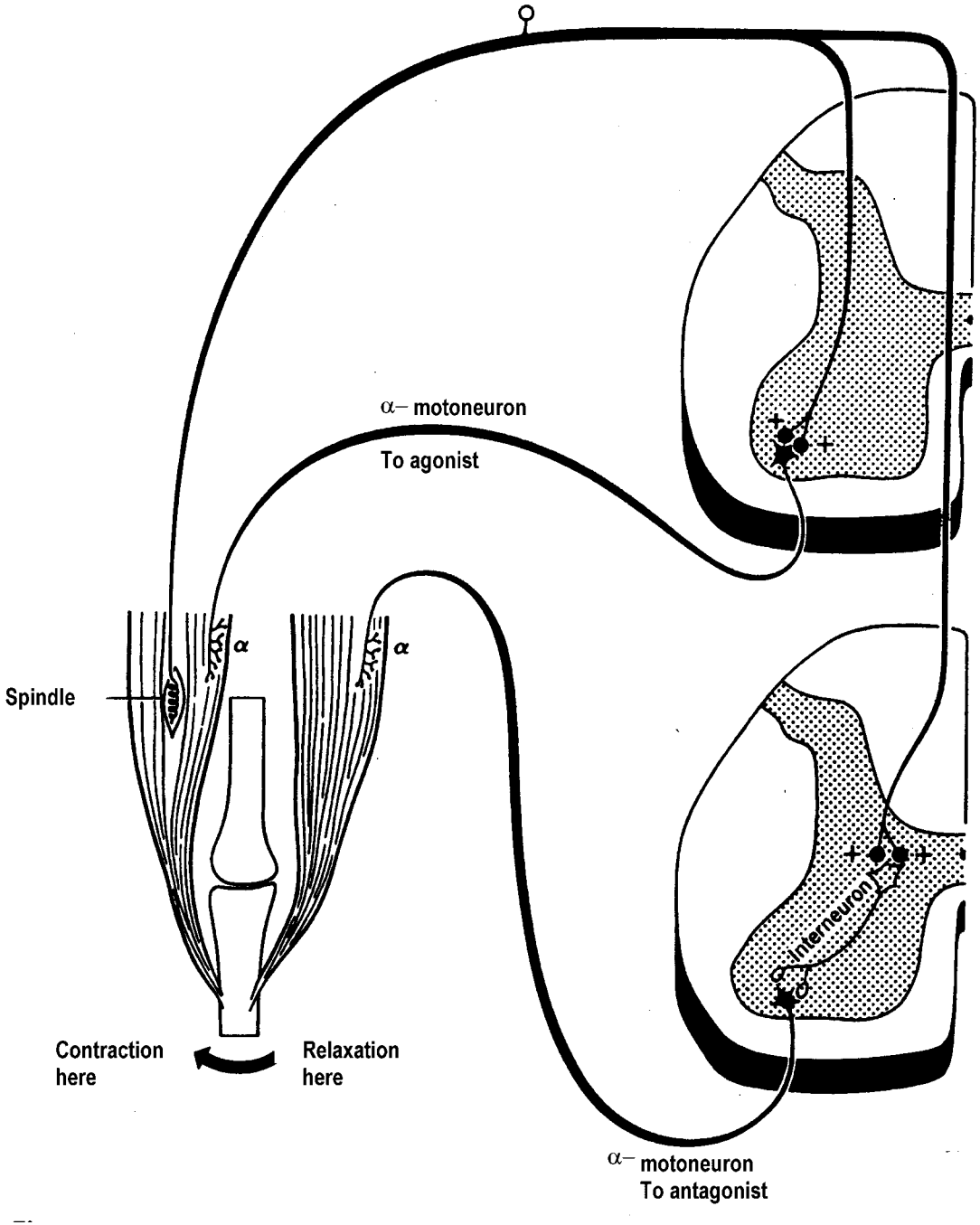
What's missing in this Scheme?



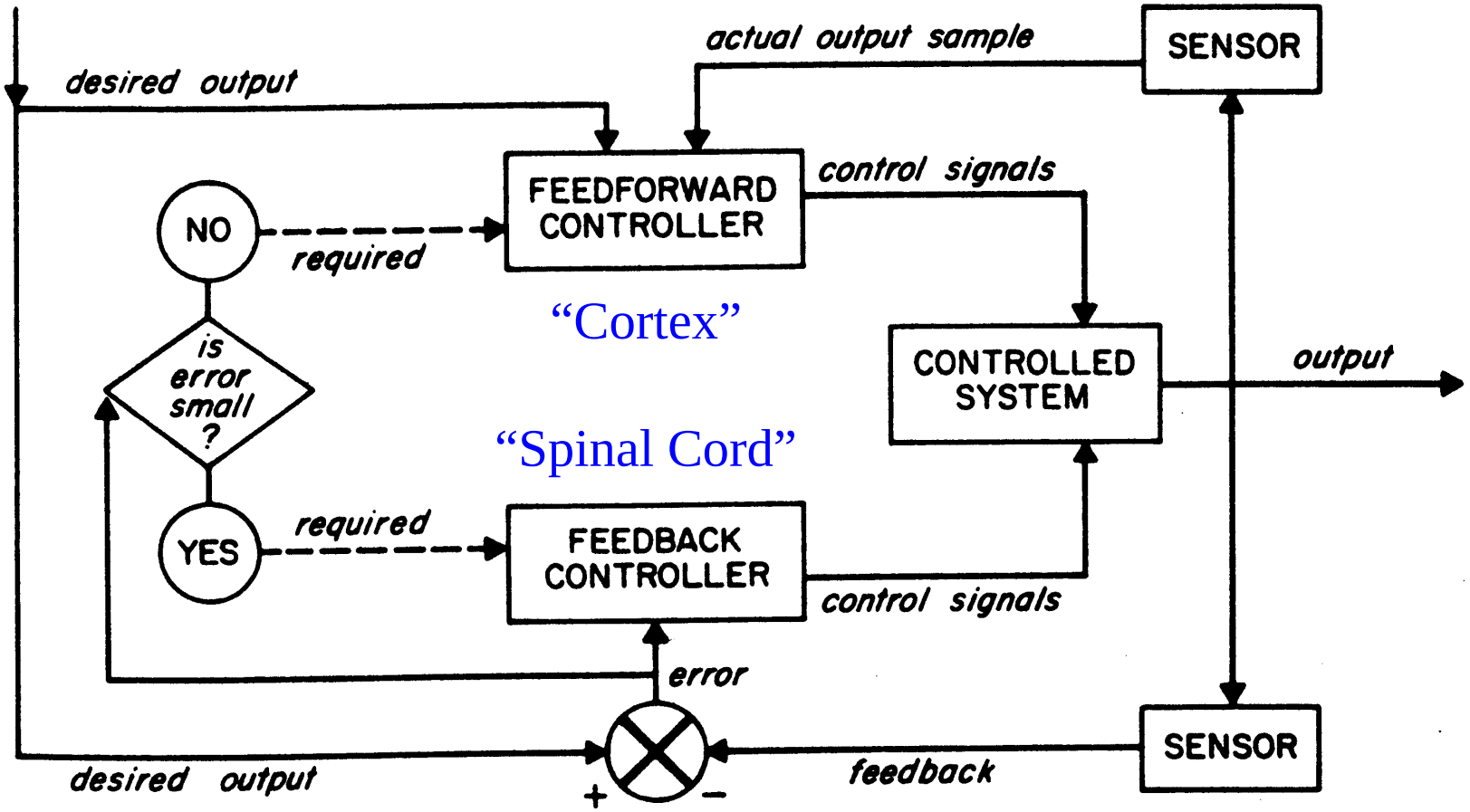
Using γ-Neurons to Set the Resting Length of the Muscle



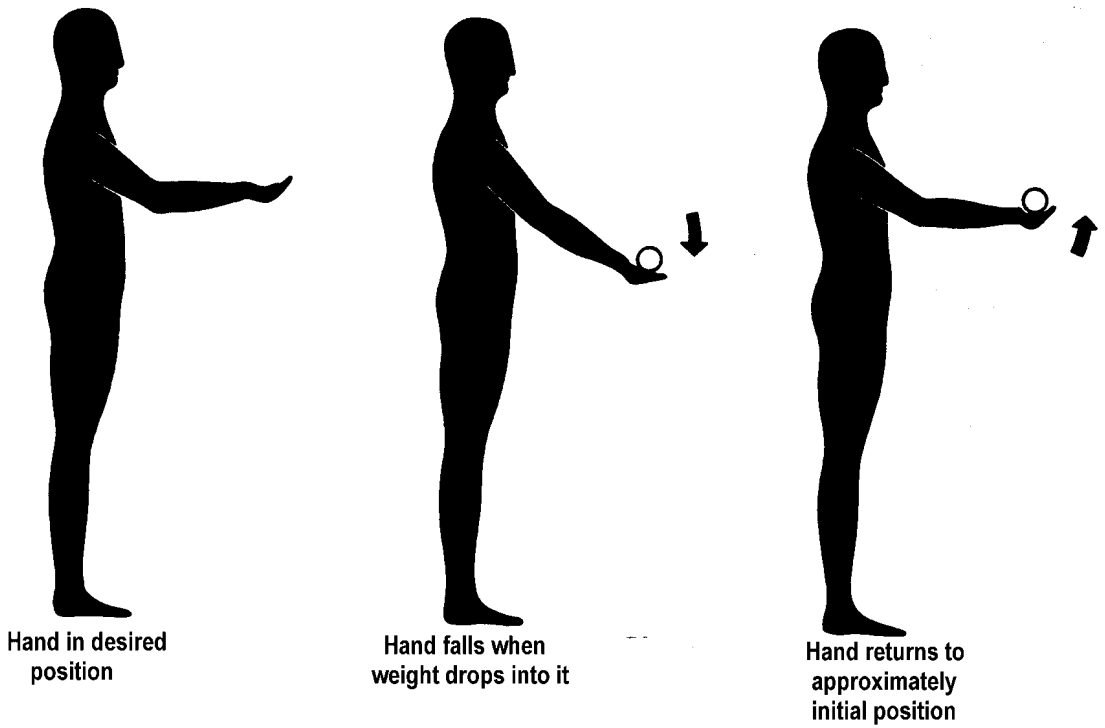

Dorsal root
Ventral root
Spinal cord
Golgi tendon organ feedback
Spindle receptor feedback
γ-motoneuron
α-motoneuron
Joint
Intrafusal muscle fiber
Extrafusal muscle fibers
Tendon


α– motoneuron
To agonist
Spindle
α
α
Contraction
here
Relaxation
here
Interneuron
α– motoneuron
To antagonist

Discrete-Activation Feedforward



"Ballistic" Correction then Feedback



This long latency reflex was noted by Navas and Stark.

Reminder: To prepare for next lecture's treatment of a mathematical model of the mass-spring muscle model, review the basic theory of eigenvectors and eigenvalues.