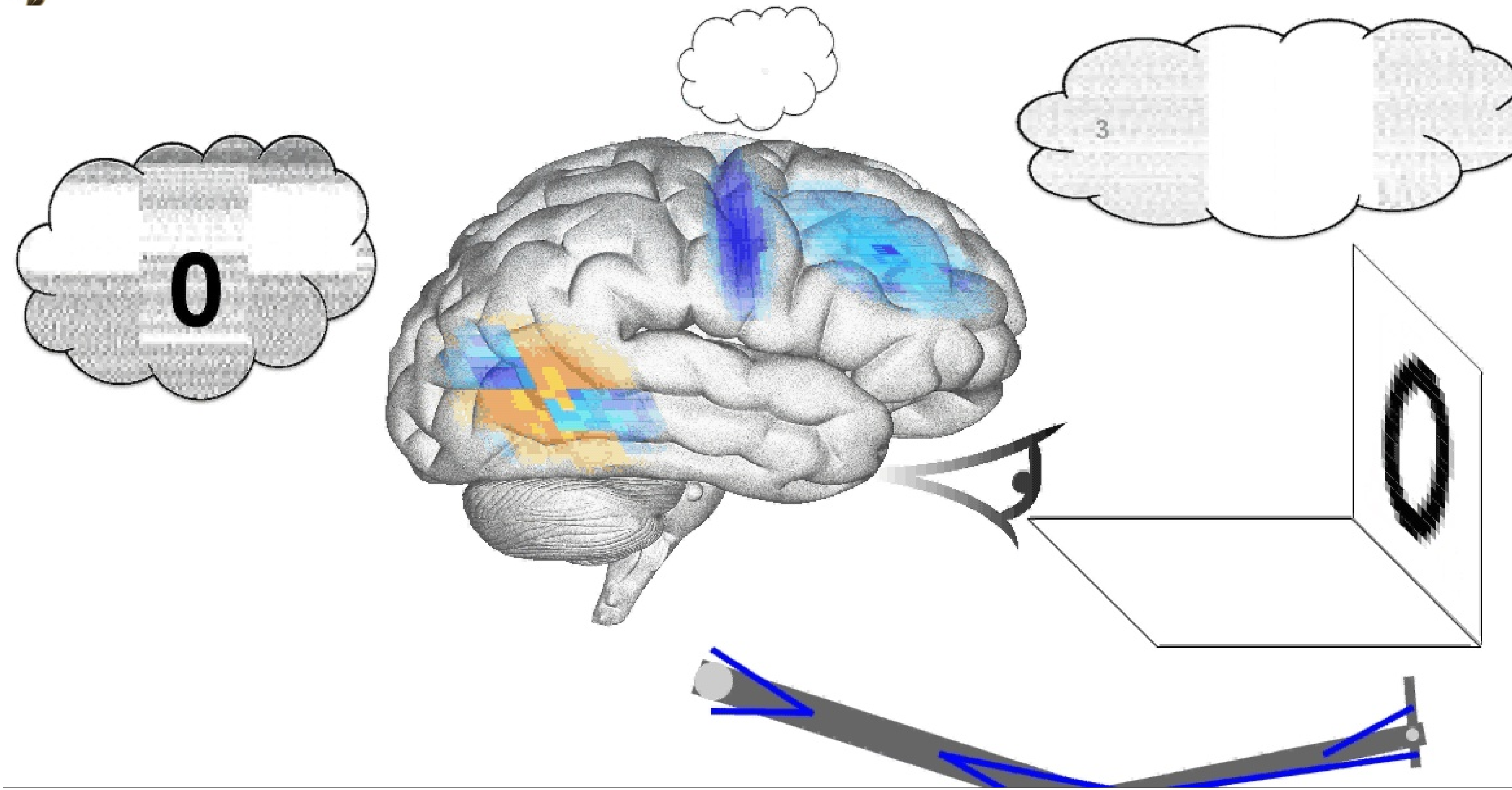
Computing with neurons

Session 2: Dynamics and Making Models



Terrence C. Stewart, Centre for Theoretical Neuroscience, University of Waterloo

Biological Algorithms

- What do neural algorithms look like?
  - Each node (group of neurons) stores a vector
  - Each connection computes a function
    - and applies a filter
    - (set of functions and filter depends on neuron model)
- Different from standard connectionism
  - There, connections can only do linear weights
  - Some functions are easier than others
    - max(a,b) takes a very large number of neurons
    - sin(a+b)*cos(b - a) is pretty easy

Recurrent connections

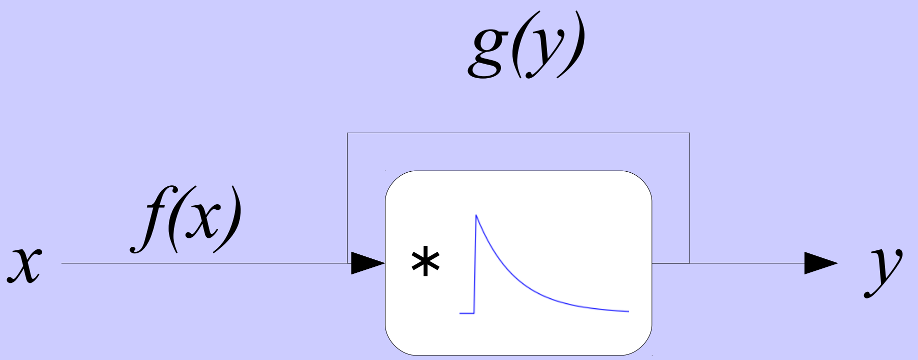
- What happens if a group of neurons connects back to itself?
  - Depends on what function is being computed on the connection

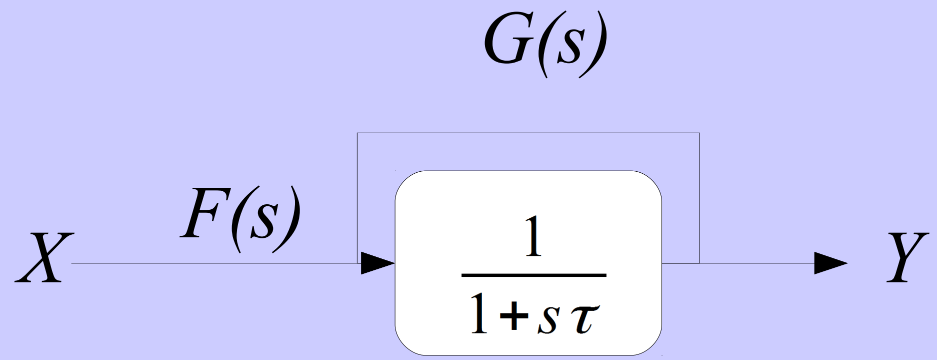
$$f(x) = x + 1$$

$$f(x) = -x$$

$$f(x) = x^2$$

Recurrent connections





$$y(t) = h(t) * (f(x(t)) + g(y(t)))$$

$$Y = \frac{1}{1+s\tau}\left[G(s)+F(s)\right]$$

$$\frac{dy}{dt} = \frac{g(y) - y}{\tau} + \frac{f(x)}{\tau}$$

$$sY = \frac{G(s) - Y}{\tau} + \frac{F(s)}{\tau}$$

$$\frac{dy}{dt} = a(y) + b(x)$$

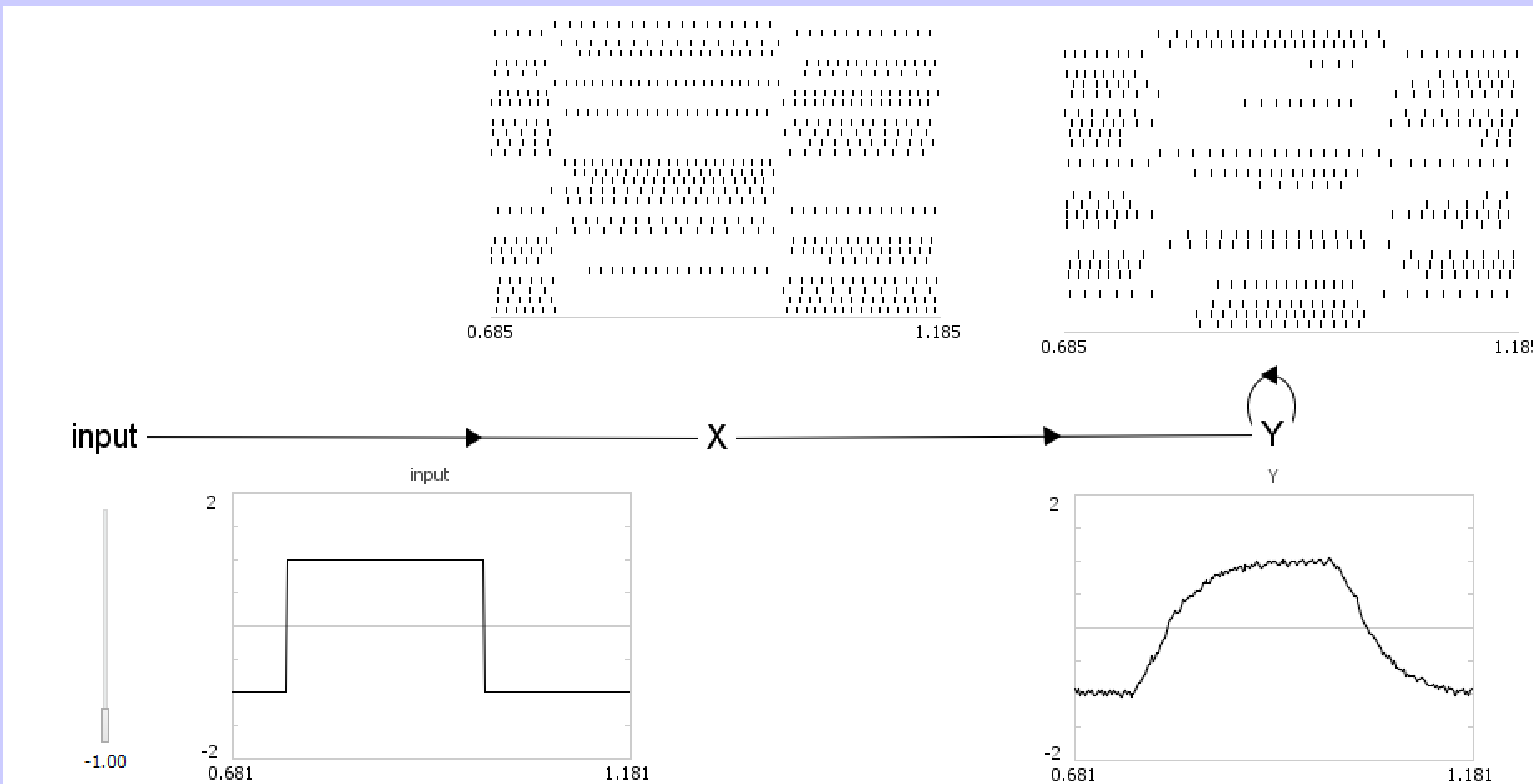
$$g(y) = \tau a(y) + y$$
$$f(x) = \tau b(x)$$

Longer Time Constants

- Desired filter: 50milliseconds

$$\frac{dy}{dt} = \frac{x - y}{0.05}$$

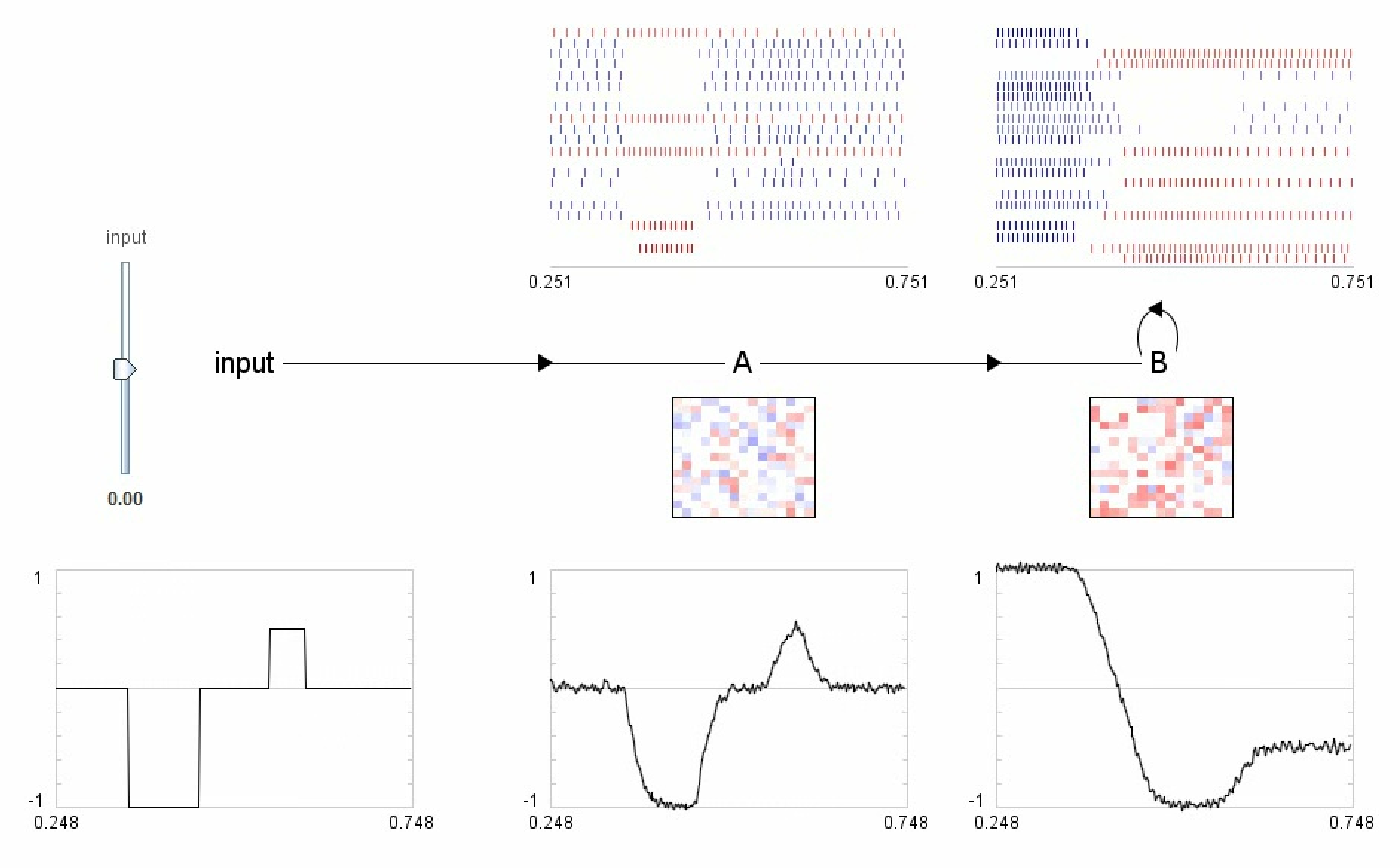
$$g(y) = (1 - \tau/0.05)\,y$$
$$f(x) = (\tau/0.05)\,x$$



Memory

$$\frac{dy}{dt} = x$$

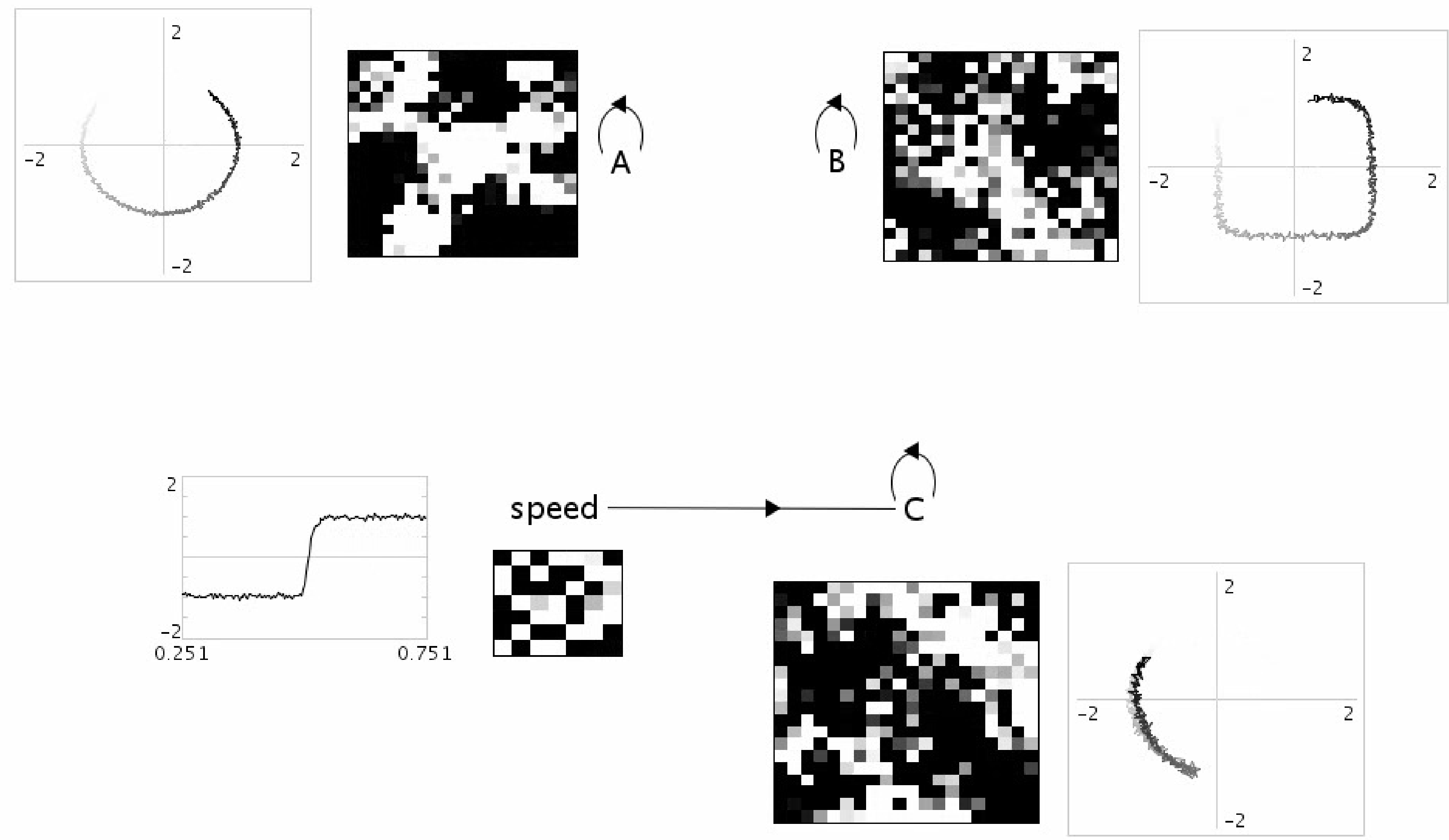
$$g(y) = y$$
$$f(x) = \tau x$$



Oscillators

$$\frac{dy_0}{dt} = -\ y_1$$

$$\frac{dy_1}{dt} = -\ y_0$$



$$\frac{dy_0}{dt} = -\ y_1\, y_2$$

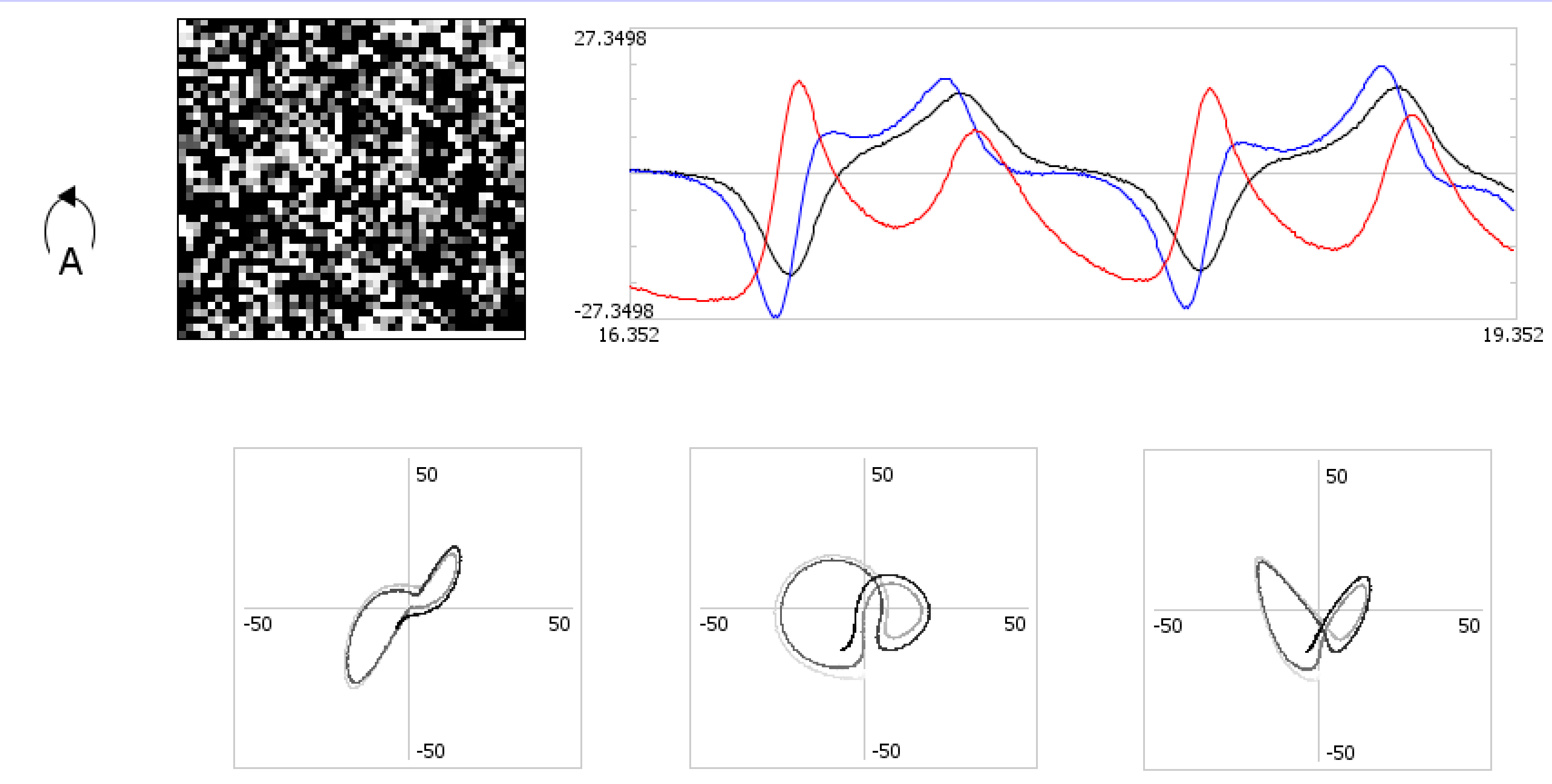
$$\frac{dy_1}{dt} = -\ y_0\, y_2$$

Chaotic Attractors

$$\frac{dy_0}{dt} = 10\,y_1 - 10\,y_0$$

$$\frac{dy_1}{dt} = -\,y_0 y_2 - y_1$$

$$\frac{dy_2}{dt} = y_0 y_1 - \frac{8}{3}(y_2 + 28) - 28$$



Biological Algorithms

- With recurrence
  - Instead of just $y = f(x)$
  - Can also include $\frac{dx}{dt} = f(x) + g(u)$
  - And, with a change of variables, $\frac{dx}{dt} = f(x, u)$
- Same restriction on the complexity of the function (low-degree polynomials)

- [nengo example - memory]

Model creation: An example

- Let's make a critter
  - Two inputs:
    - A desired velocity
    - A “fear” indicator
  - Behaviour:
    - If “fear” is low, move with the desired velocity
    - Otherwise, move back to the starting location

Is that enough?

- What about cognition?
  - How can we represent symbols?
  - How can we manipulate symbol structures?
  - Need to use vectors to represent symbols
    - Manipulate symbols by computing on vectors
- What about cognitive control?
  - How can we change tasks?
  - How can we route information between brain areas?

Increasing Complexity

- What about learning?
  - Learning new behaviours?
  - Learning when to do old behaviours?
  - Adjusting the functions being decoded

Cognitive Control