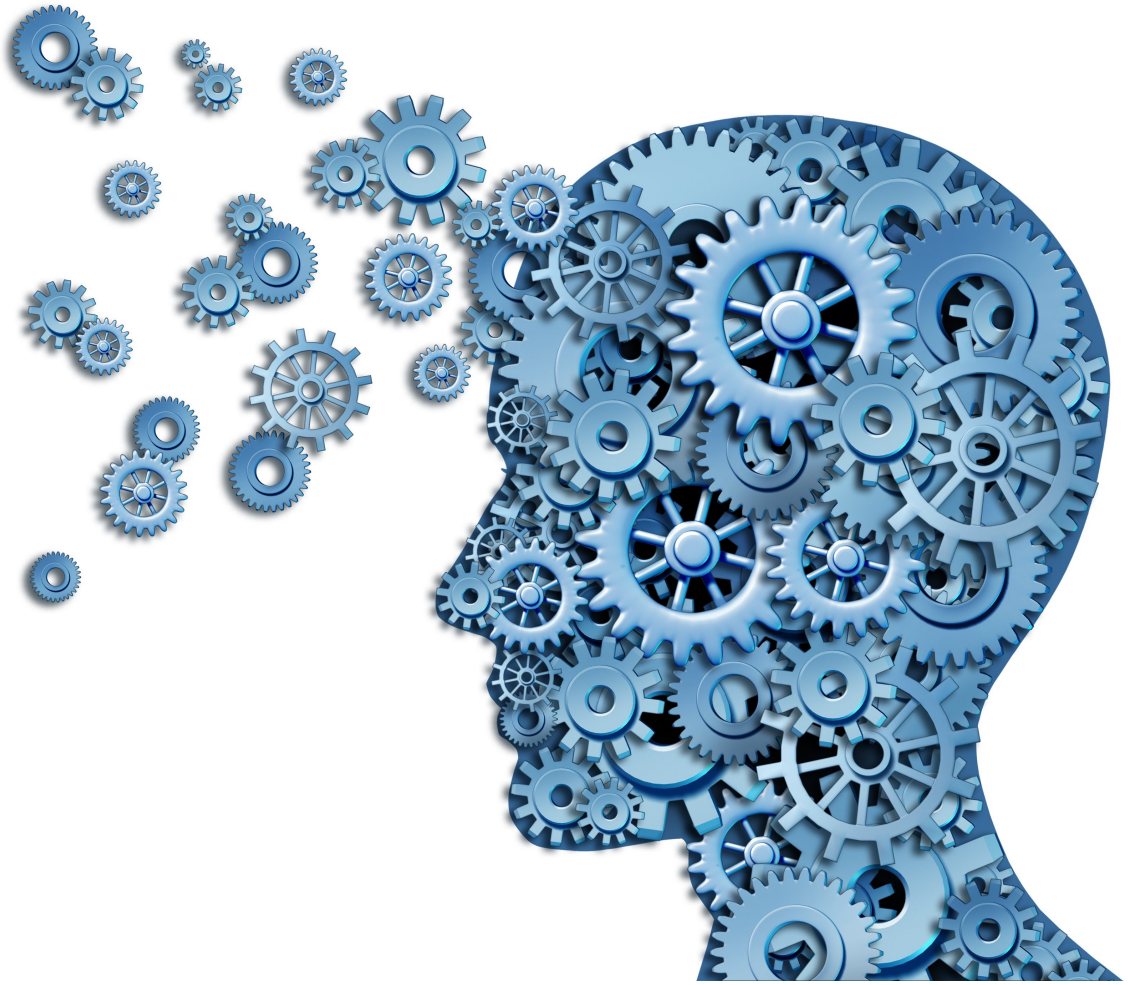
Neural & Behavioral Modeling

Week 2

System Dynamics



by Tsung-Ren (Tren) Huang 黃從仁

Goals for today

Learning dynamic equations as WHAT models

E.g., psychological states change over time

Learning system dynamics as HOW models

Structure generates behavior

More familiarity with & confidence in modeling

As you see so many good/bad models today

Goals for today

Learning dynamic equations as WHAT models

E.g., psychological states change over time

Learning system dynamics as HOW models

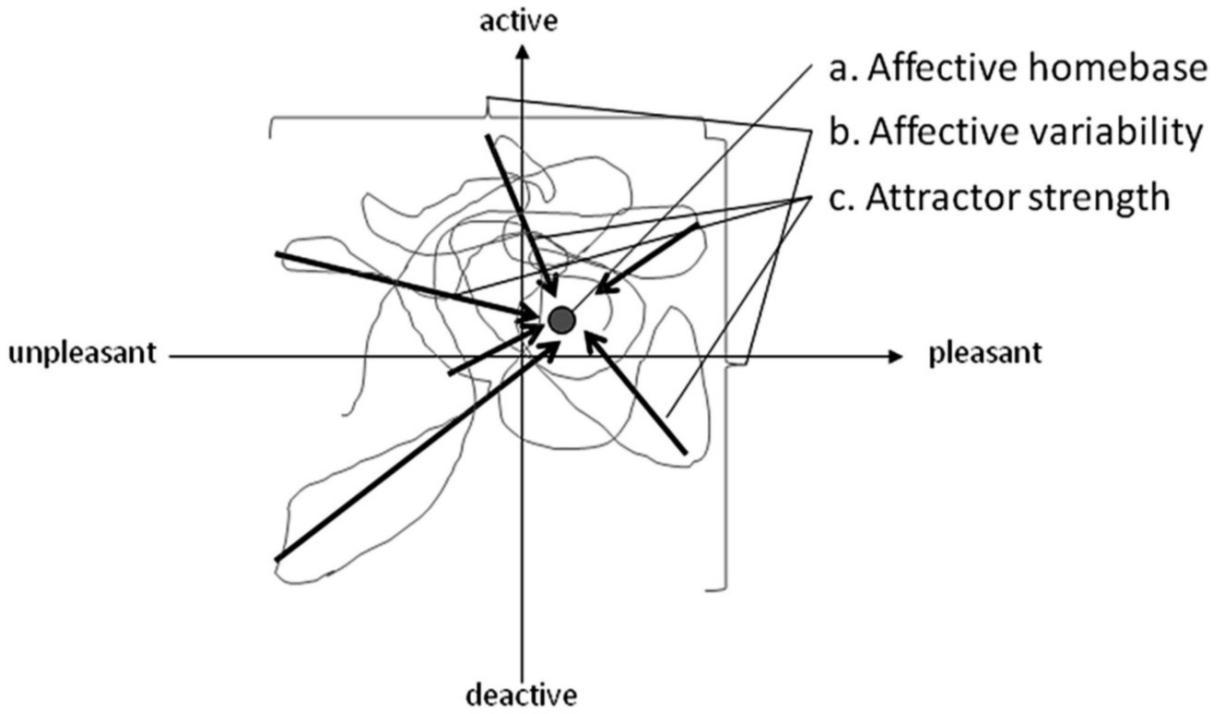
Structure generates behavior

More familiarity with & confidence in modeling

As you see so many good/bad models today

Emotion dynamics as a time series (1/2)

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \epsilon_t$$



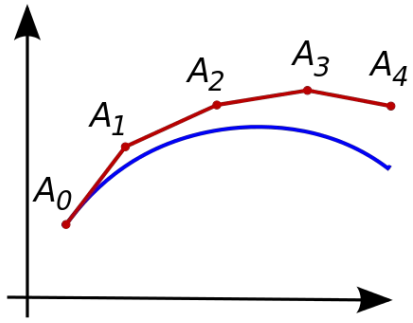
$$d\theta_p(t)/dt = \beta_p \times (\mu_p - \theta_p[t]) + \xi_p(t)$$

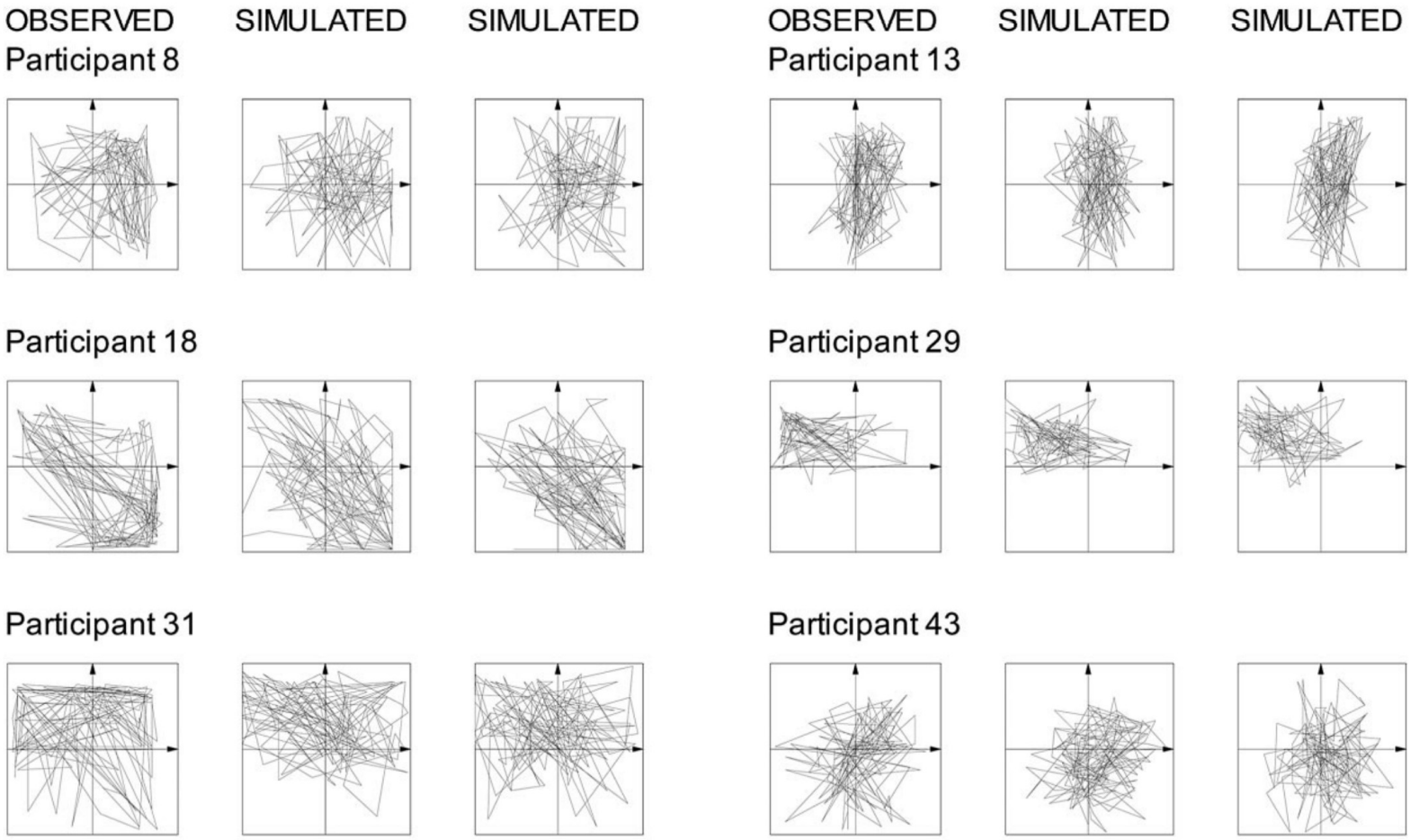
Emotion dynamics as a time series (2/2)

Fitting the dynamic model to extract (β, μ, ξ) for each person

Then you obtain the emotion characteristics of each individual/group

$$d\theta_p(t)/dt = \beta_p \times (\mu_p - \theta_p[t]) + \xi_p(t)$$





Goals for today

Learning dynamic equations as WHAT models

E.g., psychological states change over time

Learning system dynamics as HOW models

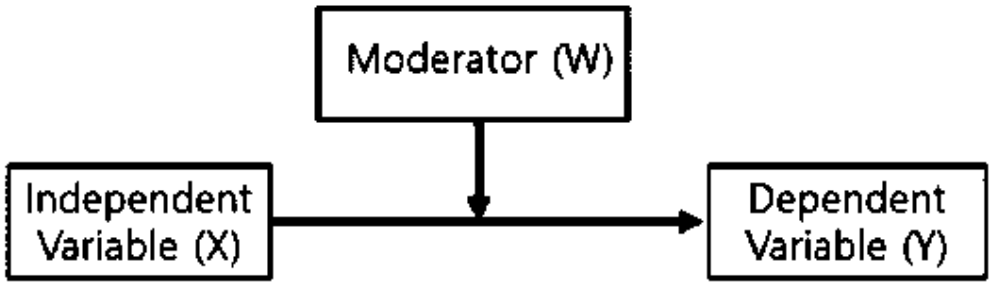
Structure generates behavior

More familiarity with & confidence in modeling

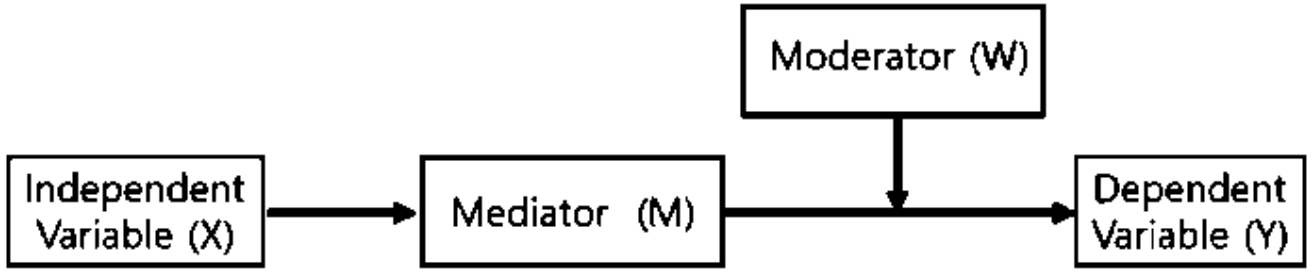
As you see so many good/bad models today

Open-loop Statistical/Psychological Models

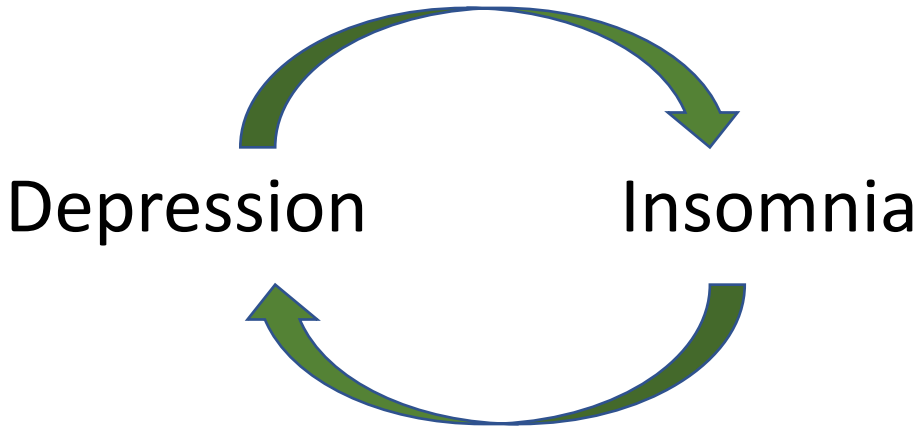
(a) Mediation Analysis



(b) Moderation Analysis

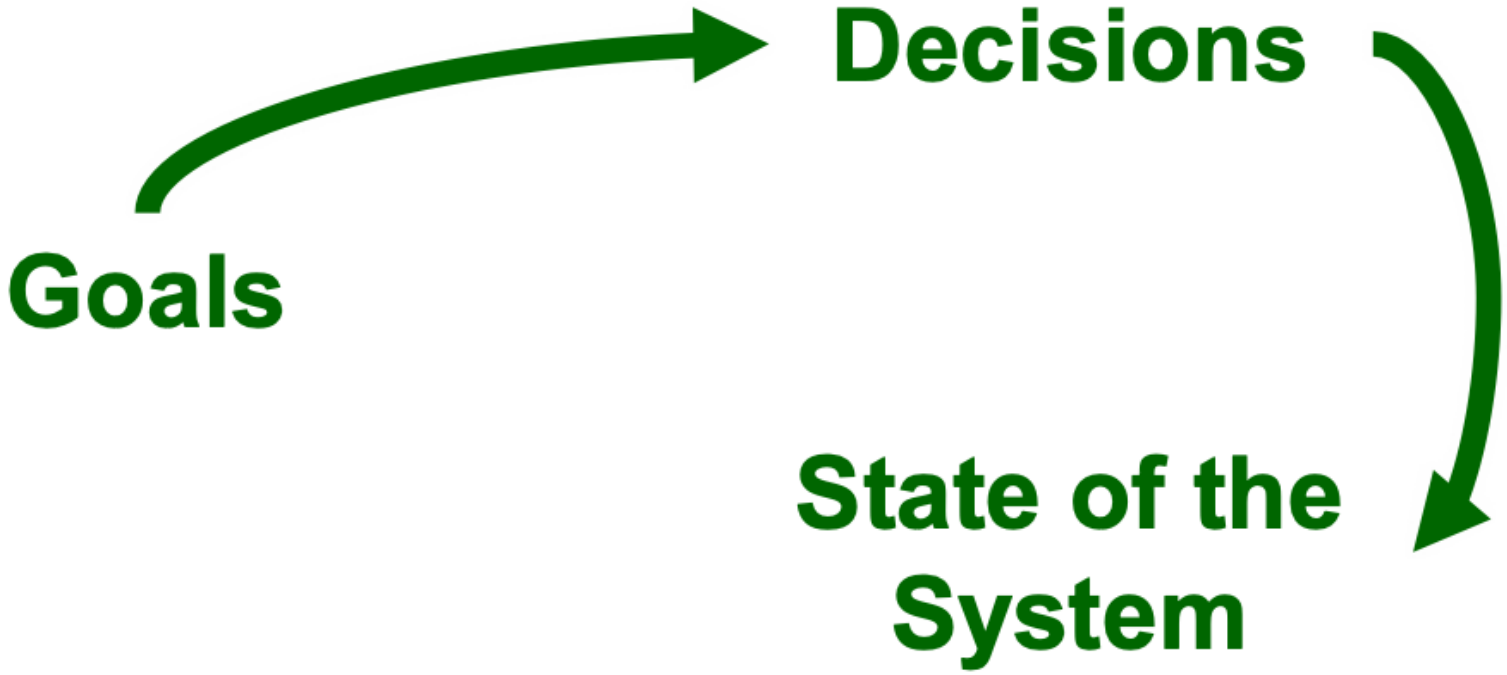


(c) Moderated-Mediation Analysis



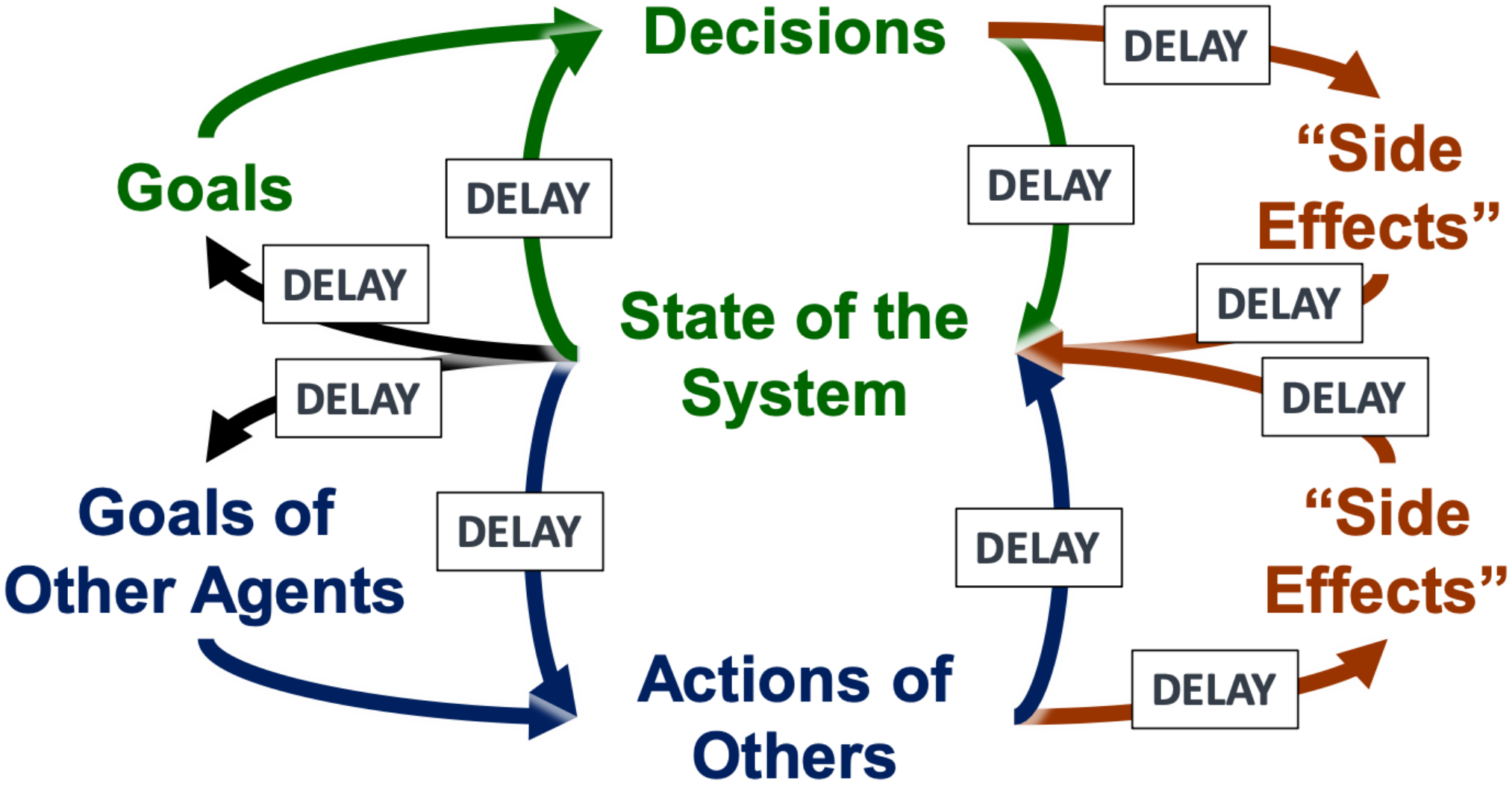
System Dynamics (1/2)

The world appears to work in a linear manner



System Dynamics (2/2)

But it does not…



E.g., Love Dynamics

A simple model for their ill-fated romance is

$$dr/dt = -aj, \qquad dj/dt = br,$$

where

$r(t)$ = Romeo's love/hate for Juliet at time t

$j(t)$ = Juliet's love/hate for Romeo at time t.

Positive values of r, j signify love, negative values signify hate. The parameters a, b are positive, to be consistent with the story.

The sad outcome of their affair is, of course, a neverending cycle of love and hate; their governing equations are those of a simple harmonic oscillator. At least they manage to achieve simultaneous love one-quarter of the time.

As one possible variation, the instructor may wish to discuss the more general second-order linear system

$$dr/dt = a_{11}r + a_{12}j$$

$$dj/dt = a_{21}r + a_{22}j,$$

Why bother system dynamics?

Considering feedback from other parts of a system for prediction

To account for unexpected “side effects” and identify high-leverage policies

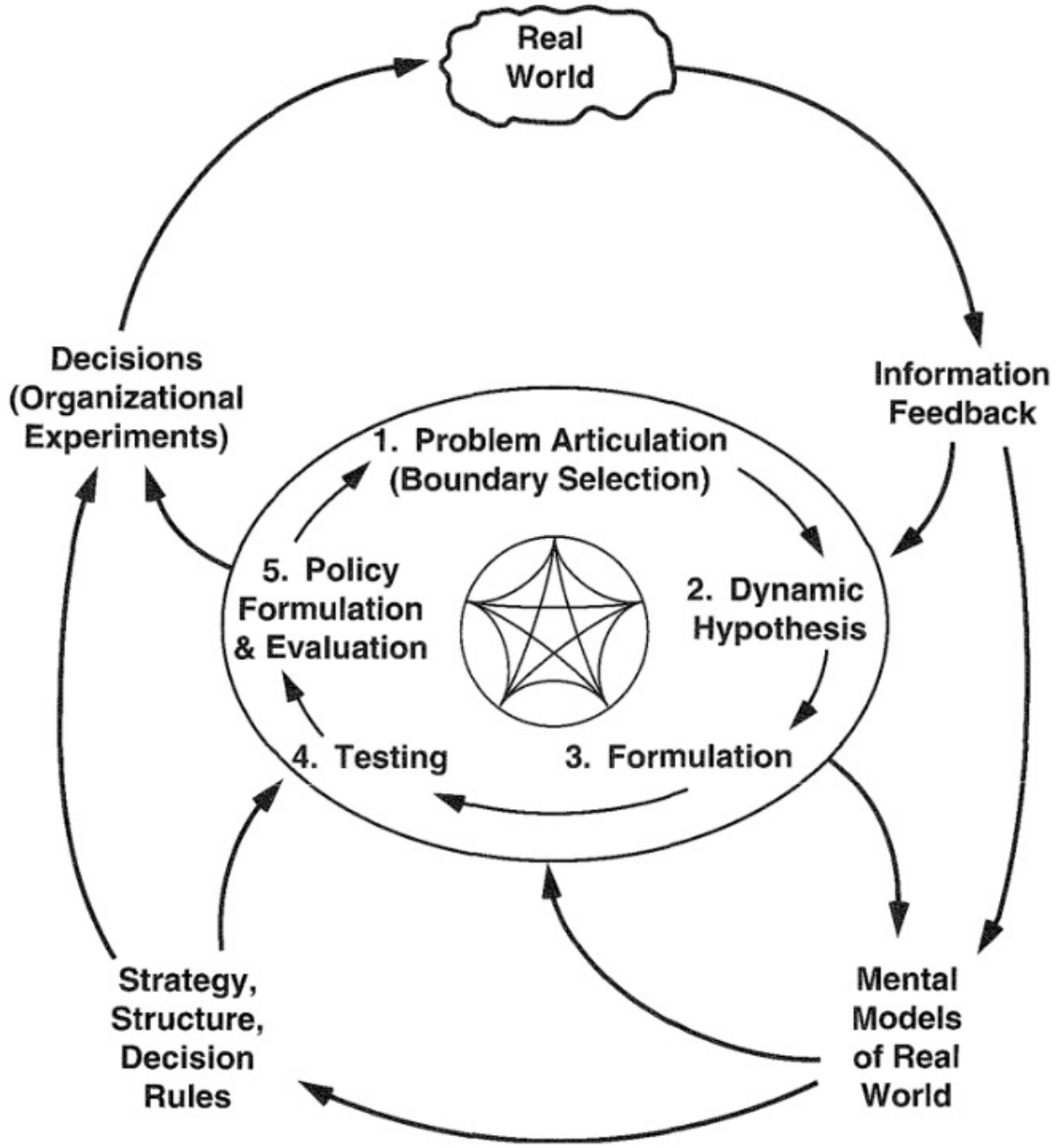
Explicitly expanding our mental model

To overcome our fallacy of “Fundamental Attribution Error”

Developing a habit of system thinking

Thinking about the underlying structure from a wider & more long-term perspective

The modeling process

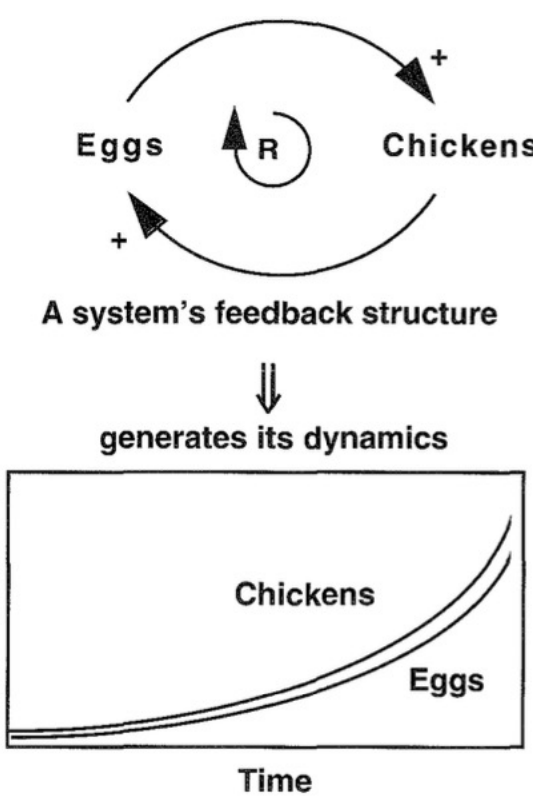


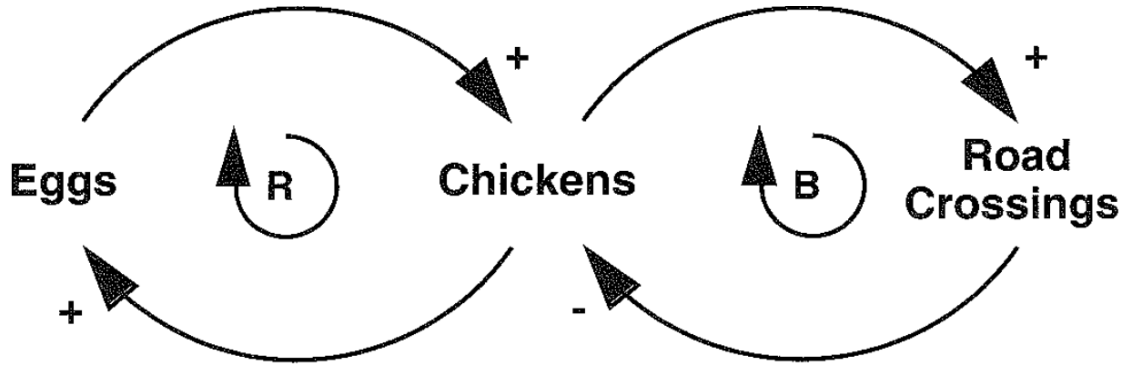
- Reference Modes
- Causal Loop Diagrams
- Stock & Flows
- Equation Formulation
- Dimensional Analysis
- Simulation
- Sensitivity Analysis
- Policy Testing

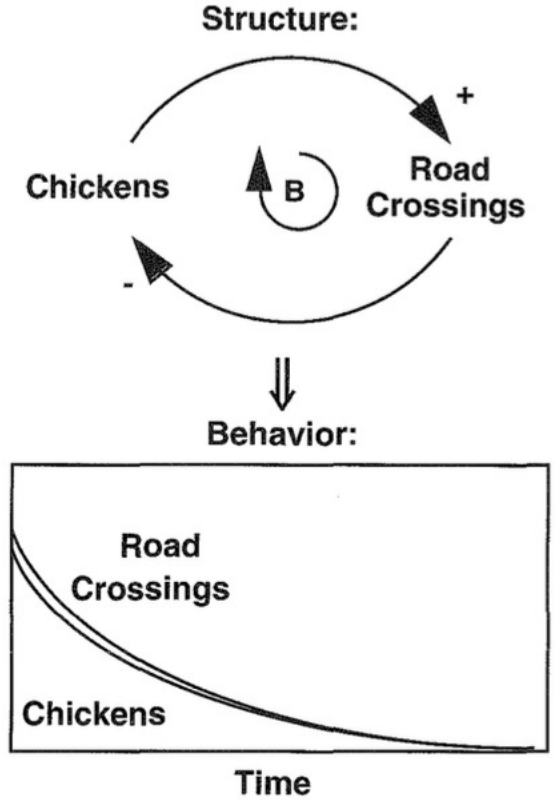
Diagrams in System Dynamics (1/2)

Causal loop diagrams (CLD) provide conceptual understanding

Each component in a CLD does not necessarily translate to a variable/stock







Diagrams in System Dynamics (2/2)

Stock-flow diagrams correspond to dynamic equations

They are often unbounded without range hacks

Integral Representation

$$Stock(t) = \int_{t_0}^{t} [Inflow(s) - Outflow(s)]ds + Stock(t_0)$$

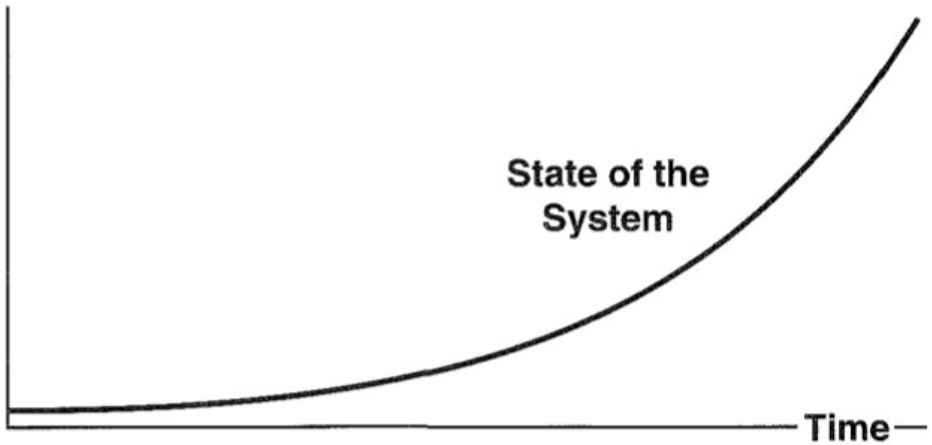
Differential Representation

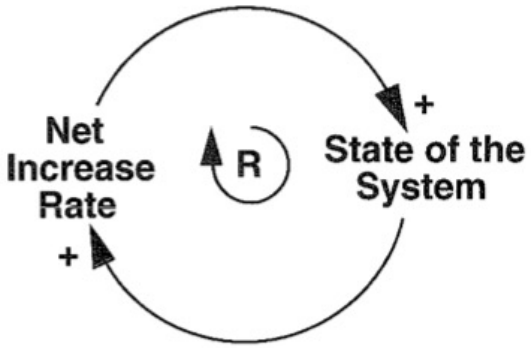
$$\frac{d}{dt}Stock = Inflow(t) - Outflow(t) = Net\ Change\ in\ Stock(t)$$

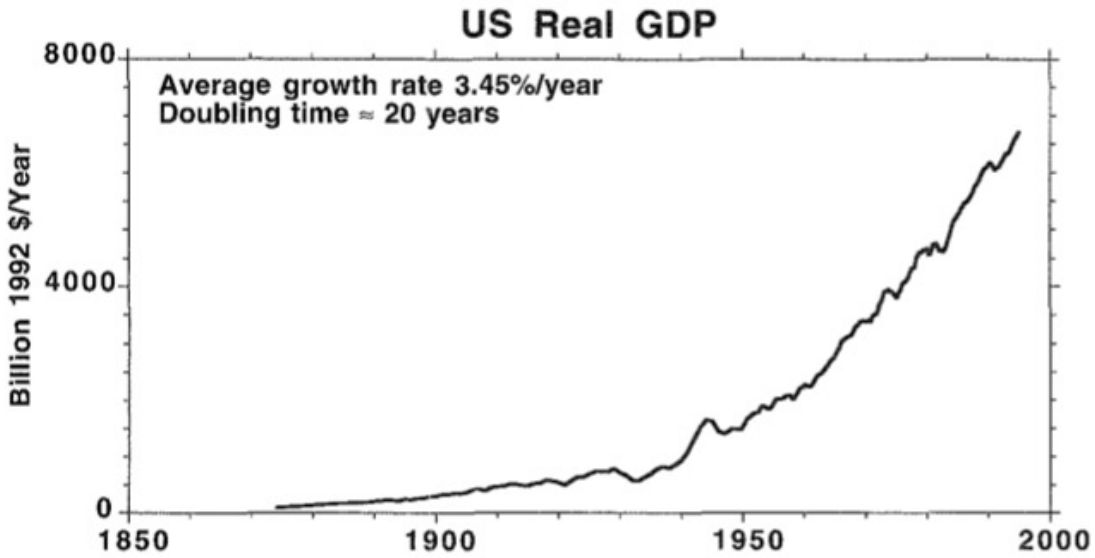
Basic Loops in System Dynamics (1/2)

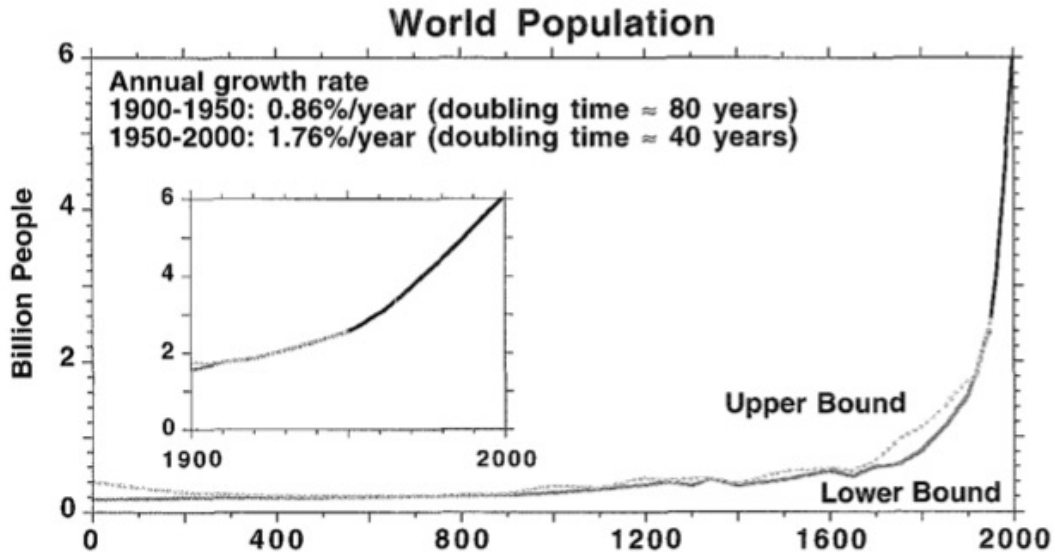
Positive/Reinforcing loop: dy/dt=y

It amplifies the current trend





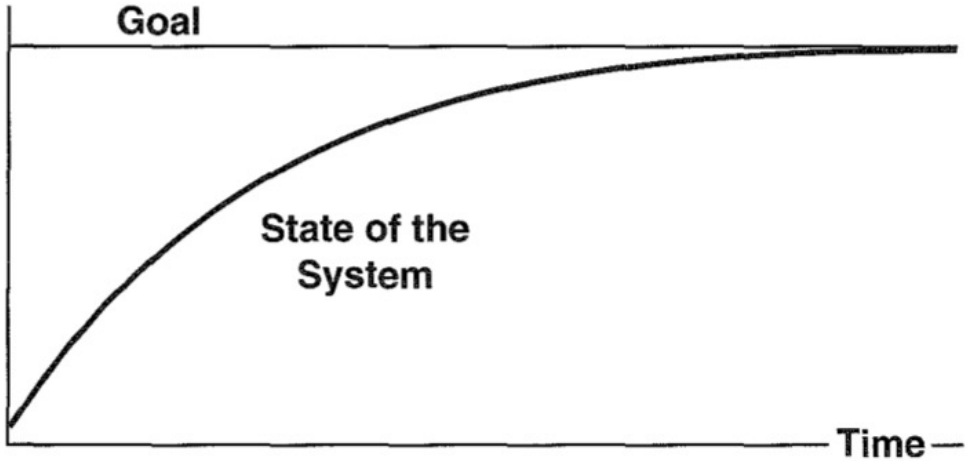


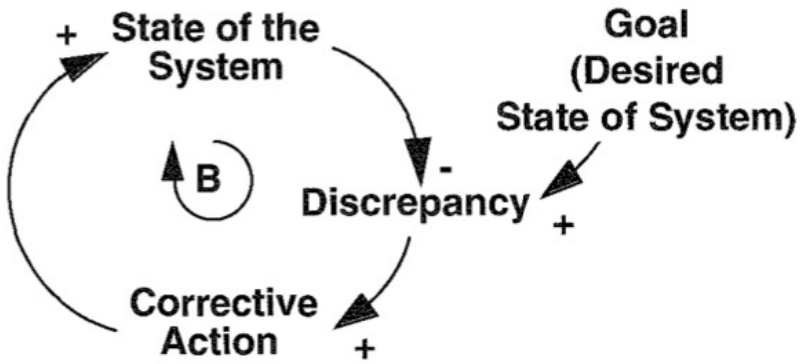


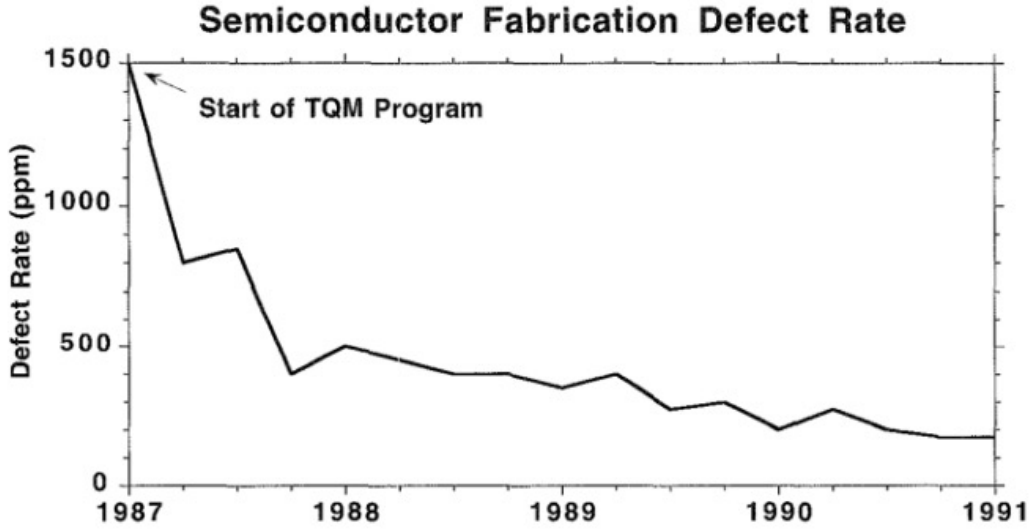
Basic Loops in System Dynamics (2/2)

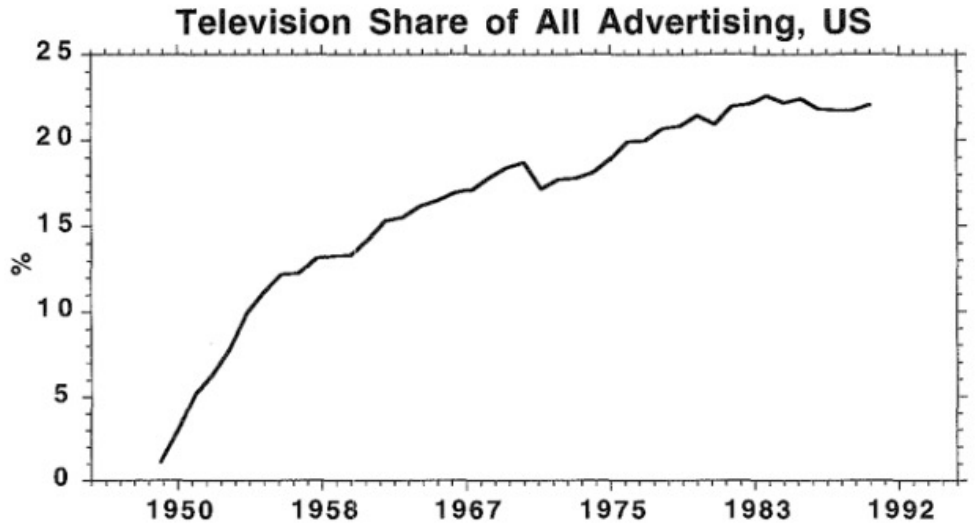
Negative/Balancing loop: dy/dt=(G-y)

It approaches the goal/equilibrium

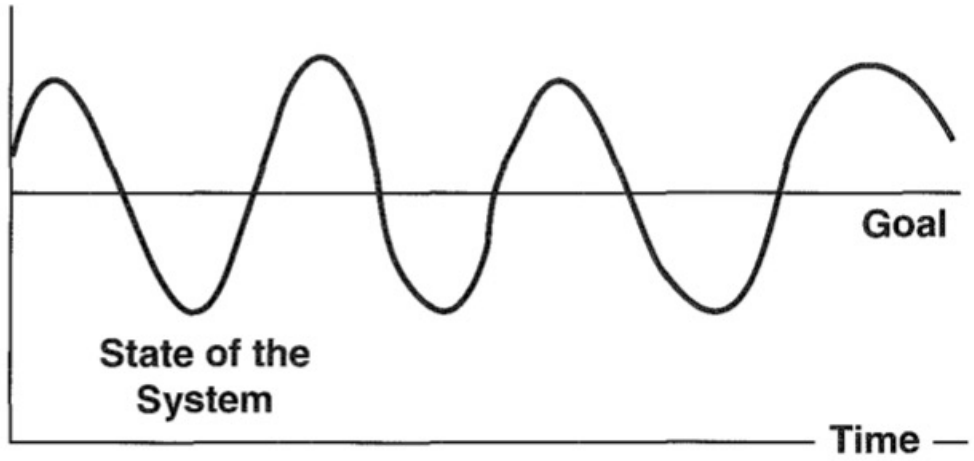


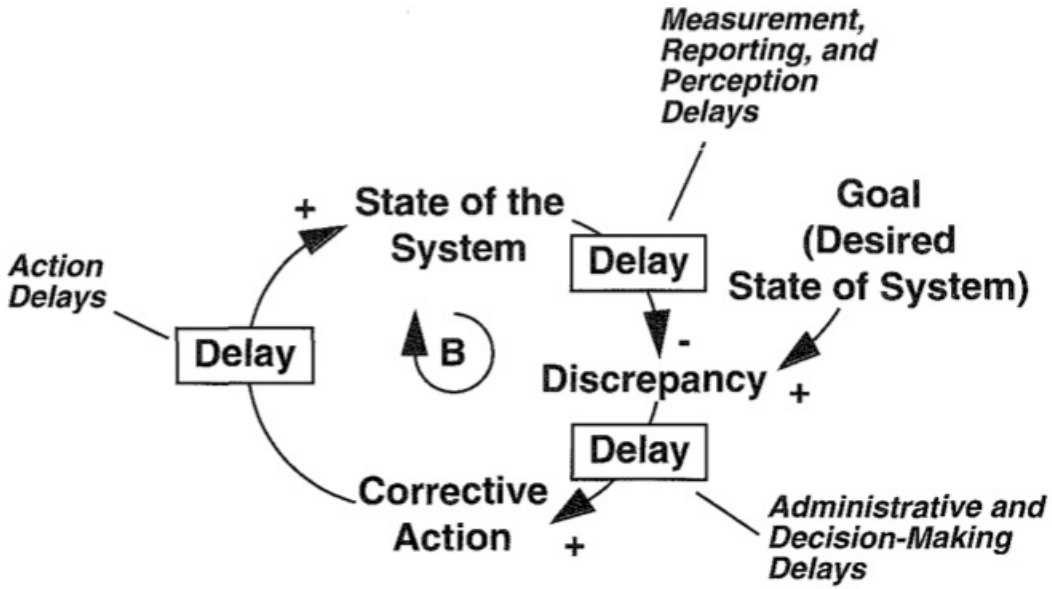


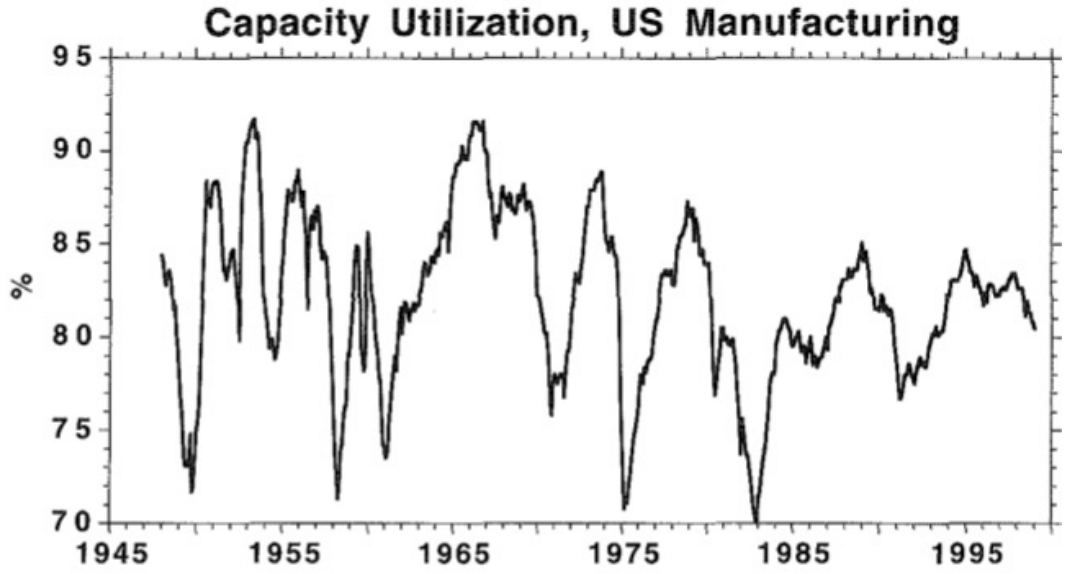




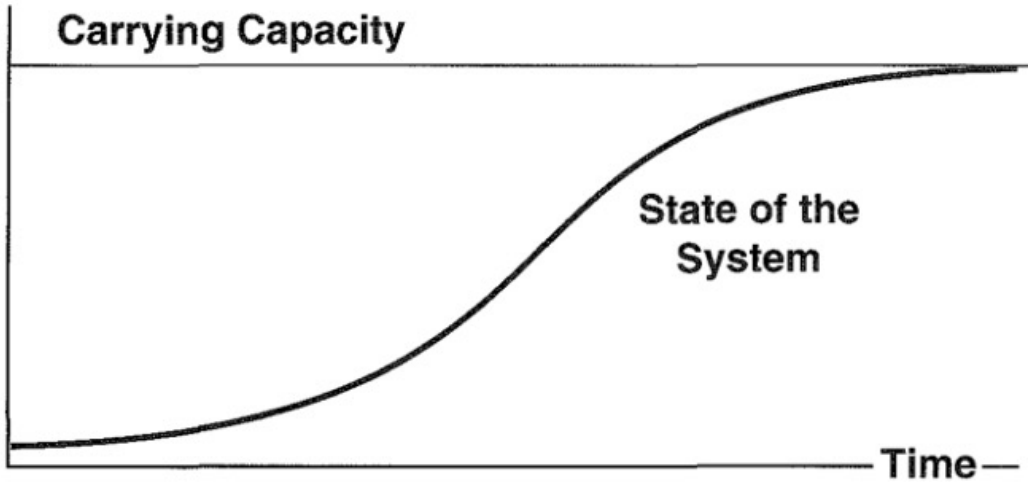
Other common modes: Oscillation

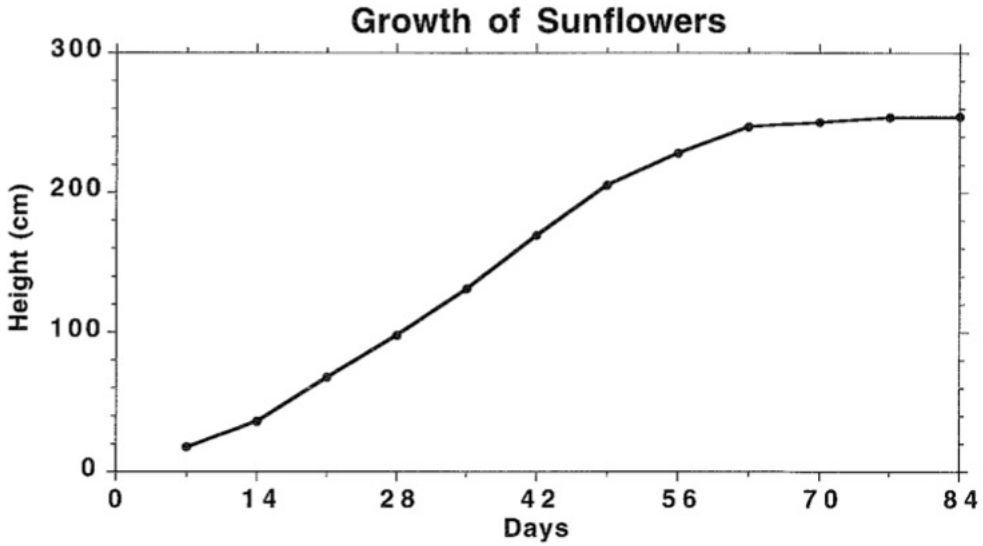


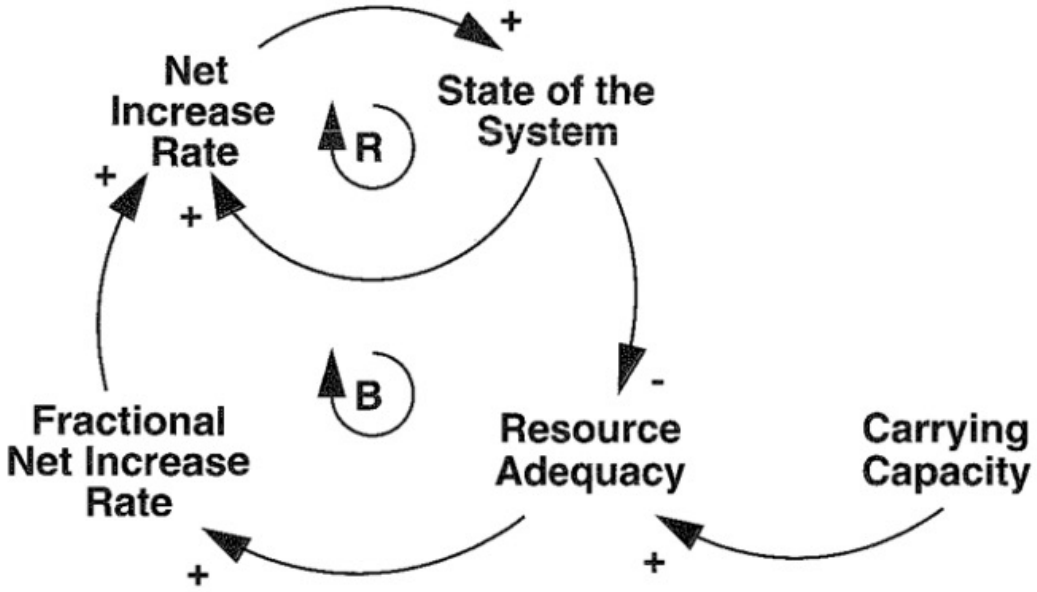


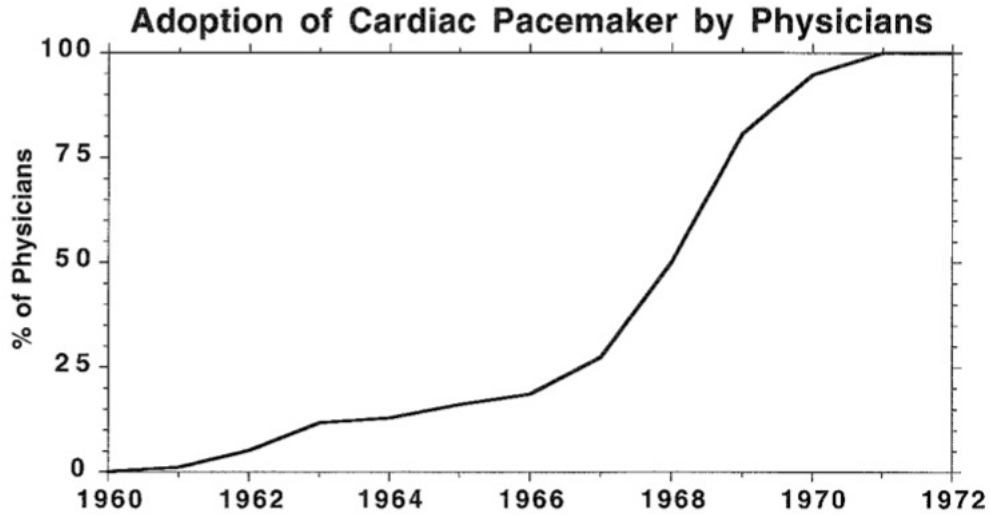


Other common modes: S-shaped Growth

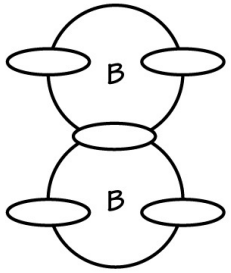








System Archetypes (Senge, 1990)

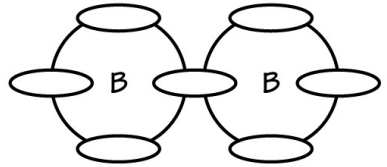


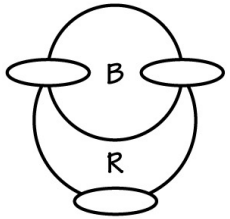
Drifting Goals

Two balancing loops that strive to close the gap between a goal and current reality.

Escalation

Two or more players who manage their own balancing loop in response to the threatening actions of others.



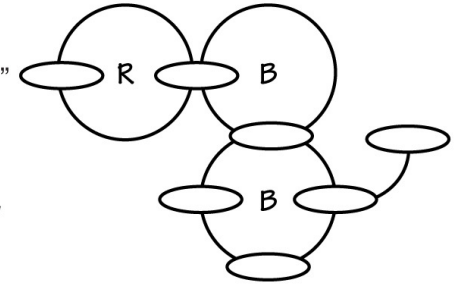


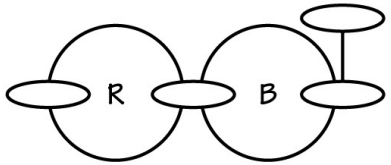
Fixes That Fail

Efforts to bring something into balance create consequences that reinforce the need to take more action.

Growth and Underinvestment

A "Limits to Success" structure with a specific system constraint—namely, an investment policy balancing loop.



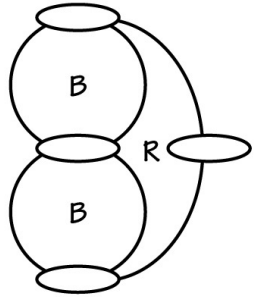


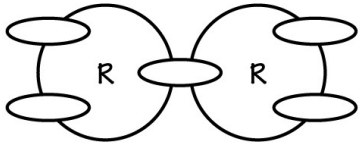
Limits to Success

A reinforcing loop creates pressure in the system that is relieved by one or more balancing loops that slow growth.

Shifting the Burden

Two balancing loops compete for control in "solving" a problem symptom, while a reinforcing side-effect of one solution makes the problem worse.



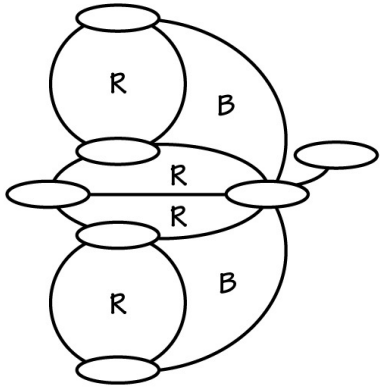


Success to the Successful

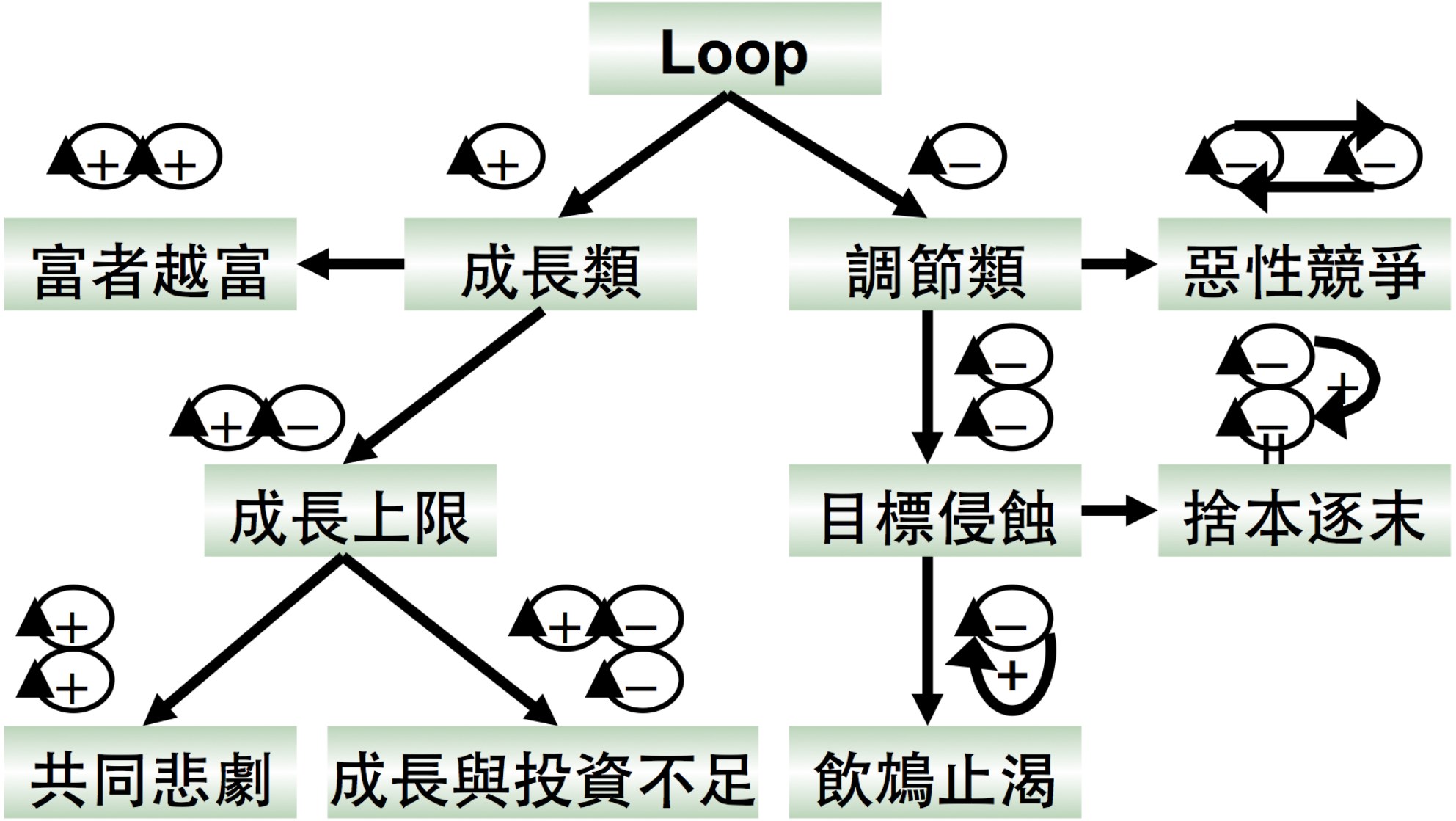
Two reinforcing loops compete for a common, limited resource.

Tragedy of the Commons

Two or more reinforcing activities whose sum total strains a limited resource and creates balancing consequences for all.



System Archetypes (Senge, 1990)



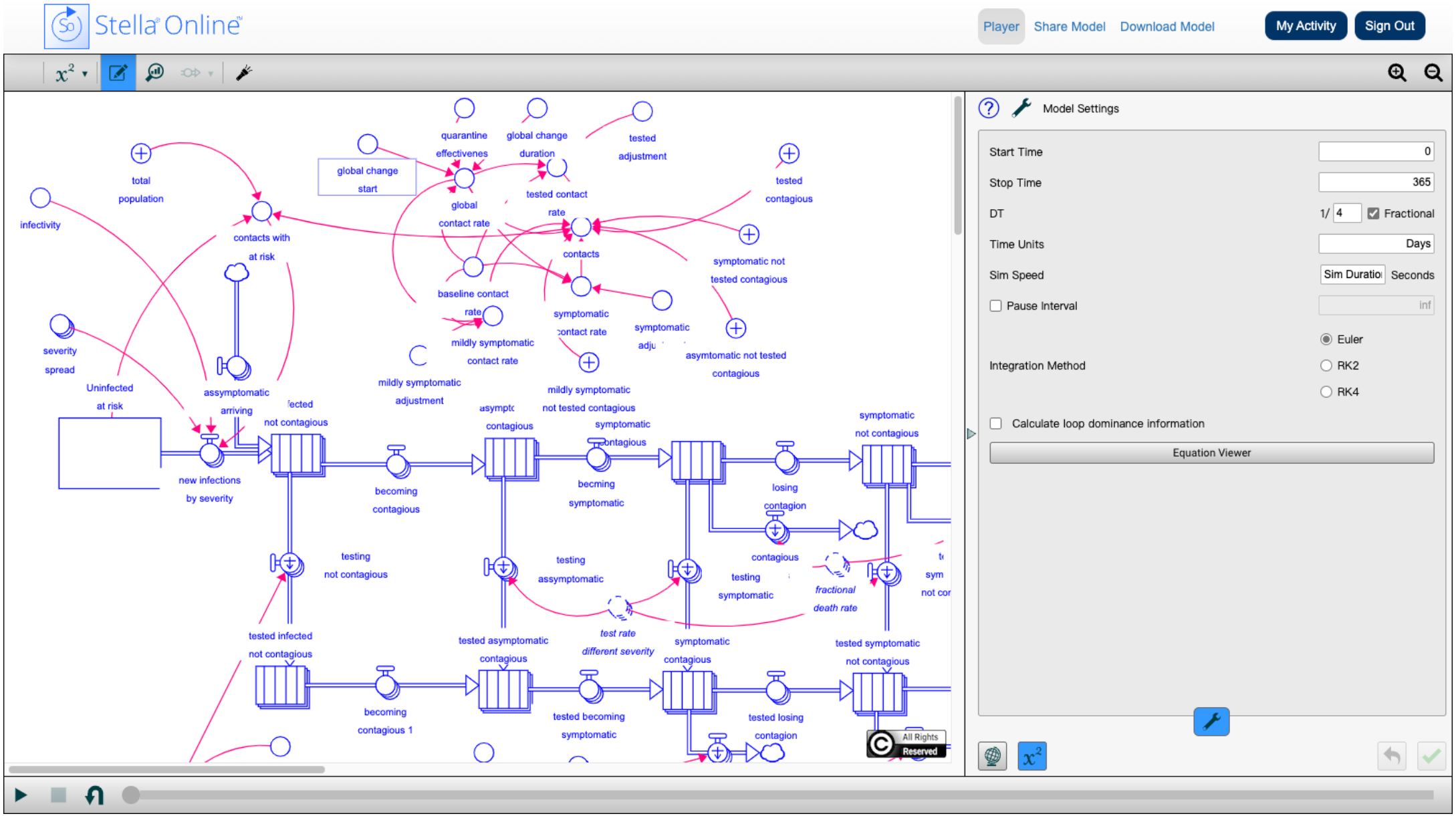
Do system archetypes help?

Is it possible to learn effectively in complex settings without simulation? Ca n the use of problem structuring methods, elicitation techniques, and other qualitative systems methods overcome the impediments to learning? If intu ition is developed highly enough, if systems thinking is incorporated in prec ollege education early enough, or if we are taught how to recognize a set of "system archetypes" (Senge, 1990),will we be able to improve our intuition about complex dynamics enough to render simulation unnecessary?

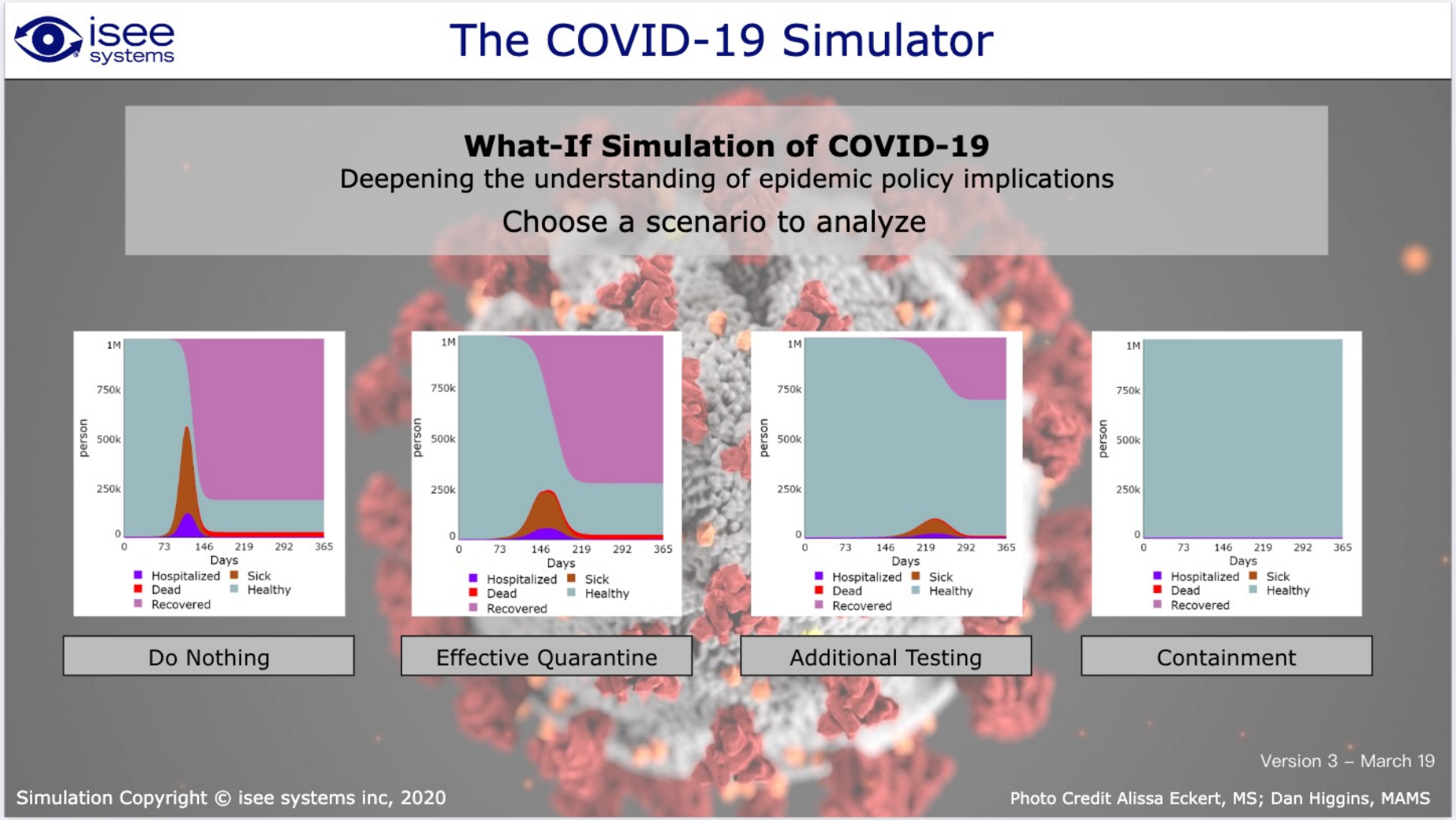
The answer is clearly no…it will still be necessary to develop formal models, solved by simulation to learn about such systems.

by Sterman (2000)

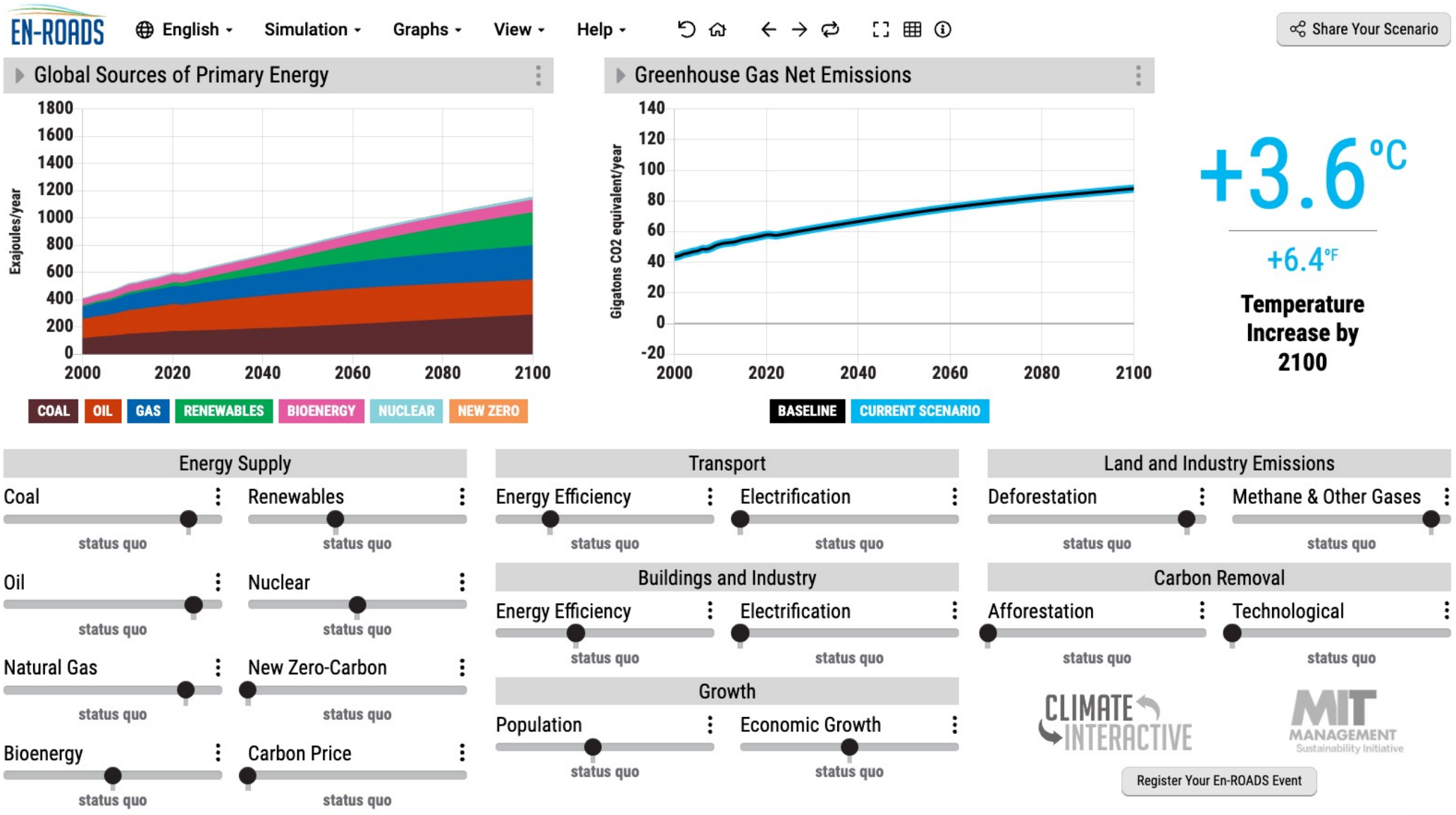
The real goal: A complex system (1/3)



The real goal: A complex system (2/3)



The real goal: A complex system (3/3)



Goals for today

Learning dynamic equations as WHAT models

E.g., psychological states change over time

Learning system dynamics as HOW models

Structure generates behavior

More familiarity with & confidence in modeling

As you see so many good/bad models today

Neural Ordinary Differential Equations

1 Introduction

Models such as residual networks, recurrent neural network decoders, and normalizing flows build complicated transformations by composing a sequence of transformations to a hidden state:

$$\mathbf{h}_{t+1} = \mathbf{h}_t + f(\mathbf{h}_t, \theta_t) \tag{1}$$

where $t \in \{0 \dots T\}$ and $\mathbf{h}_t \in \mathbb{R}^D$. These iterative updates can be seen as an Euler discretization of a continuous transformation (Lu et al., 2017; Haber and Ruthotto, 2017; Ruthotto and Haber, 2018).

What happens as we add more layers and take smaller steps? In the limit, we parameterize the continuous dynamics of hidden units using an ordinary differential equation (ODE) specified by a neural network:

$$\frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t), t, \theta) \tag{2}$$

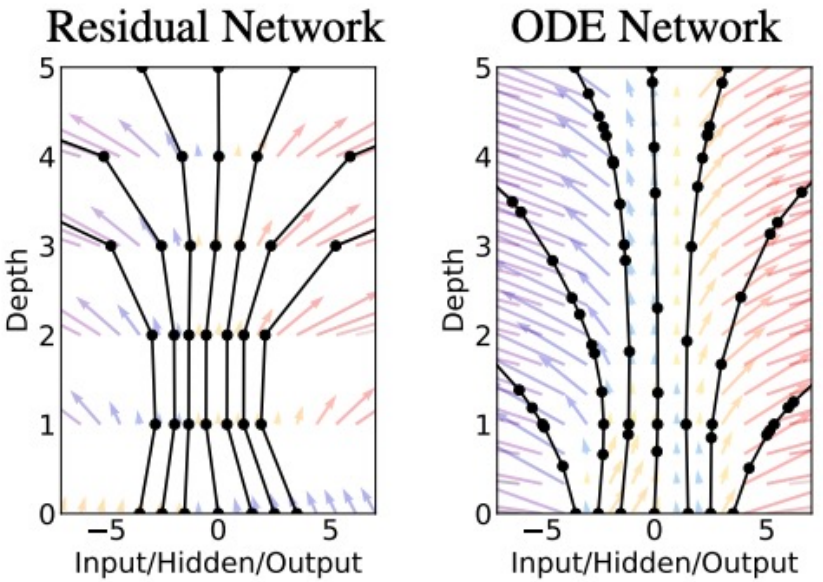


Figure 1: *Left:* A Residual network defines a discrete sequence of finite transformations. *Right:* A ODE network defines a vector field, which continuously transforms the state. *Both:* Circles represent evaluation locations.

Starting from the input layer $\mathbf{h}(0)$, we can define the output layer $\mathbf{h}(T)$ to be the solution to this ODE initial value problem at some time T. This value can be computed by a black-box differential equation solver, which evaluates the hidden unit dynamics f wherever necessary to determine the solution with the desired accuracy. Figure 1 contrasts these two approaches.

Game Over