

# Week 2

## Agent-Based Modeling as a Method



# Review

- What is “modeling and simulation”?
- Course Design
- Objectives
- Readings
- Software
- Assignments



# Outline

- Introduction to chaos and complexity
- Why use agent-based models?
  - Advantages
  - Pitfalls
- Brief introduction to NetLogo



# I. CHAOS AND COMPLEXITY



# “Chaos”

- What do you think of?



# Chaos Theory

- Theory of mathematics
- Nonlinear systems
  - Simultaneous equations
- **Insight:** apparently “random” systems may be deterministic



# Nonlinearity

- Linear

$$y = a + bx + e$$

- Nonlinear

$$\begin{aligned}\frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= x(y - z) - y \\ \frac{dz}{dt} &= xy - \beta z\end{aligned}$$



# Problem with nonlinearity

- Cannot break down equations into constituent parts
- “Solutions” hard to come by
  - Depend upon initial conditions
  - Depend upon simultaneous values





# Mathematical chaos

- Deterministic equations
- Erratic behavior *over time*
  - Focus on “dynamics”
- Same system may appear:
  - Stable, predictable
  - Random, changing, unpredictable
  - *Depending upon “initial” conditions*



# Why does chaos arise?

- Sensitivity to initial conditions
  - The “Butterfly Effect” (Lorenz)
  - Small initial differences lead to large changes over time
  - Flap of the wings changes the path of a typhoon



# Attractors

- Graphical technique to illustrate coherence of chaotic systems
- Types
  - Point
  - Limit cycle
  - Strange



# Implications

- Simple systems can produce chaos
  - Saperstein: “the prediction of unpredictability”
  - Rumsfeld:

“As we know, there are known knowns; there are things we know we know. We also know there are known unknowns; that is to say we know there are some things we do not know. But there are also unknown unknowns—the ones we don’t know we don’t know.”

- Even if chaotic “locally,” there is “global” stability



# Implications

- Sensitivity to initial conditions
  - slight differences may lead to vastly different outcomes
- Differences appear random, but aren't
  - Even deterministic systems may appear “noisy” or random
- Hard to study mathematically or statistically
  - Difficult (impossible?) to infer rules or governing laws from chaotic data
  - Old methods focus on stasis, ignore dynamics
  - We require simulation methods



# Chaos Theory

- Questions?



# Some questions to consider

- Why do riots occur?
- How much vaccination is enough to prevent the spread of the flu?
- Why is there residential segregation in Norfolk?
- How does Google work?
- Why are bell-bottoms and tattoos popular (again)?



# Notice . . .

- Each example is:
  - Decentralized, leaderless
  - Massively parallel
  - Interdependent decision-making
  - Spatial
  - “Predictably unpredictable”





# “Organization”

- What do we mean?
  - “An organization” (entity)
  - Versus “organization” (state of being)



# But physics tell us ....

- “Entropy”
- Second Law of Thermodynamics



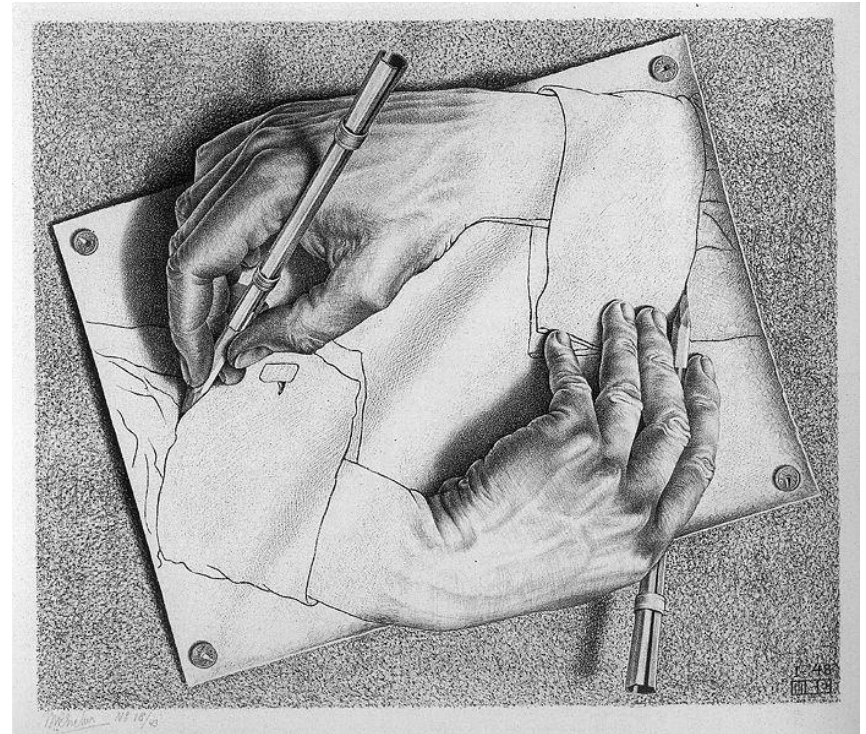
# Self Organization

- Waldrop 1992: “matter’s incessant attempts to organize itself into ever more complex structures, even in the face of the incessant forces of dissolution” p. 102
  - “Autopoiesis”



# Self Organization

- Organization “for free”
  - No external guidance
  - No leadership
  - Contra Second Law

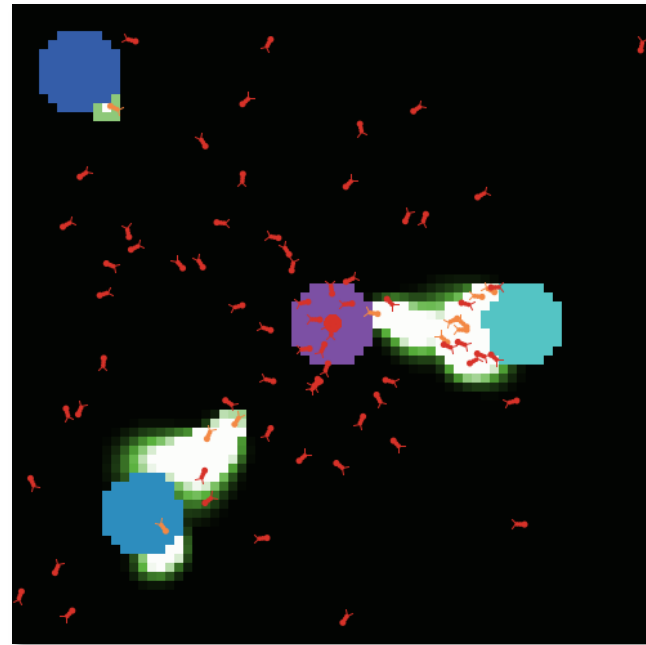
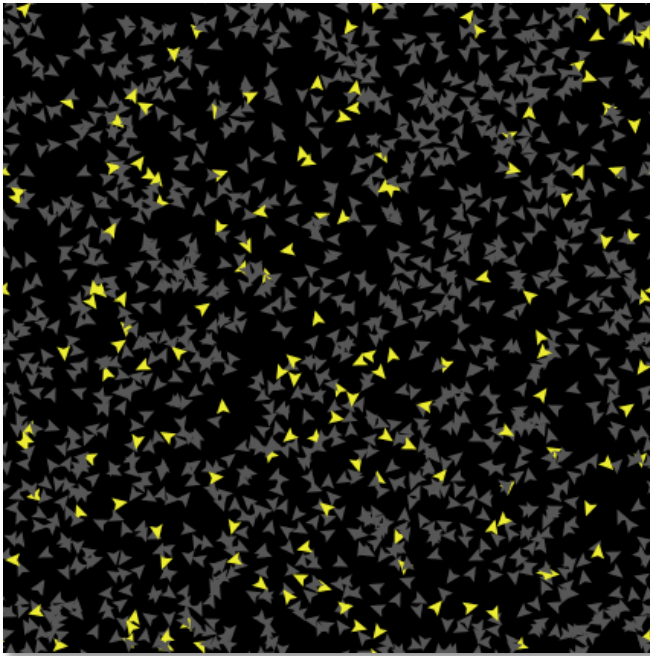


M.C. Escher, *Drawing Hands* (1948)



# Self Organization

- Examples



# “Emergence”

- Definitions
  - Gilbert and Troitzsch
  - Pepinsky, pp. 373-4: “those phenomena that appear at an aggregate level, based not on specific micro-level interactions of agents but rather on the complex and often unpredicted effects of many such interactions”
- Irreducible phenomena of systems that paradoxically arise from micro-level interactions
- **GOAL OF ABM: simulate emergence from “the bottom up” (micro-level interactions)**



# Second-Order Emergence

- Probably unique to social systems
- Agents recognize emergent structures
- Emergence thus affects agent behavior
  - “Double hermeneutic” (Giddens)





# Implications

- The phenomenon of emergence requires us to model interactions across levels of analysis
  - Agents and structures
- Mathematics is no help





# Chaos and Complexity

- Questions?



## II. WHY USE ABM?



# Uses of ABM

- Gilbert & Troitzsch's seven uses
  - Gain a better understanding
  - Prediction
  - Substitute for human expertise
  - Training
  - Entertainment
  - **Discovery**
  - **Formalization**



# Why use ABM? (1)

- Theory: fidelity to actual system
  - Nonlinearity
    - Cannot be understood analytically
    - Unpredictable
  - Space
    - Physical (GIS)
    - Network
  - (Bounded?) rationality of actors (Simon)
  - Dynamics rather than equilibria



# Why use ABM? (2)

- Methodology
  - Formalization: expose hidden assumptions
    - Pepinsky, p. 374: “Simulation forces the researcher to examine deeply the assumptions that she makes about the environment, the agents, the rules and parameters.”
  - Relationships between levels of analysis
    - Emergence
    - The agent-structure debate (Wendt)
  - Quasi-experimentation
  - Counterfactuals (Fearon)
  - “micro foundations”



# Why use ABM? (3)

- Agent learning
  - Inverts rational choice assumption
  - Change through learning
  - Appeal of ABM to constructivists
    - Hoffmann
    - Lustick

	Rational Choice	ABM
<i>Assumptions</i>	Interests	Behavior
<i>Inferences</i>	Behavior	Interests



# ABM for IS

- Core concerns of IR
  - Change through actor learning
    - Gilpin (1981): “interaction,” “systemic” and “system” change
  - Reciprocal influence of agents and structures
    - Wendt (1987)
  - Counterfactual reasoning
    - Fearon (1991)



# IS examples

- Pepinsky, p. 371
  - Political behavior
  - Arms races
  - Trade and economics
  - Ethnic conflict
  - Escalation
  - War and conquest
- Lots of others we'll read this semester





# Why use ABM?

- Questions?



# Logic of Simulation

- Combination of induction and deduction
  - *Deduction*: formal specification of
    - Actors
    - Variables
    - Rules of behavior
  - *Induction*: analyze inductively
    - Data generated by model
    - Alternative “histories” of the modeled system
- Computational social science
  - “Third way” (Axelrod 1997)
  - “From the bottom up” (Epstein and Axtell 1996)



# Epistemology

- “Underspecified and incomplete”
- Two issues
  - Explanation vs. Prediction
  - Point vs. Trend Prediction



# Explanation vs. Prediction

- Gilbert and Troitzsch
  - ABM best for “discovery” and explanation
  - Not good for prediction
- *Prediction* is not the only epistemological criterion
  - Scientific realism vs. positivism
  - Theories that predict well but are based on the wrong explanation are problematic



# Point vs. Trend Prediction

- Types of prediction
  - ABM good for “trend” prediction
    - Explore the statistical distribution of outcomes in the model
    - Provide a good sense of probabilities of events
  - ABM not good for “point” prediction
    - Due to sensitivity of macro parameters to micro values



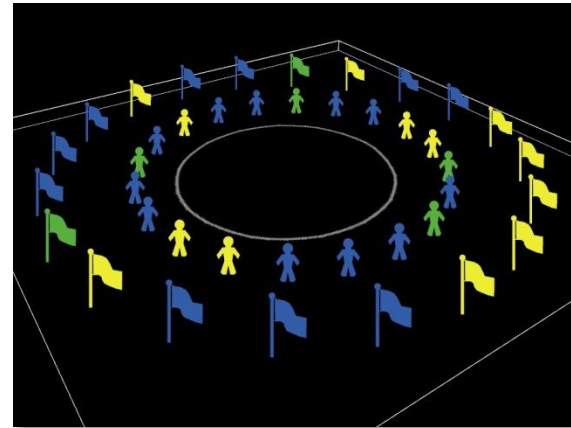
# “Model”

- All agent-based models:
  - Represent the environment of the system
  - Stipulate relevant actors
  - Specify rules (algorithms) and parameters
  - Explore behavior over (simulated) time
- NOTE: does not require assumption of linearity
  - Or indeed of any functional form
  - Contra regression analysis
- Clausewitz: “The map is not the territory”



# Notes about vocabulary

- “Target” vs. “Model”
- Simulation
  - “creating an artificial representation [model] of a real world system [target] in order to manipulate and explore the properties of that system.” Pepinsky, p. 369



# Notes about vocabulary

- “Inputs”
  - Similar to “independent” variables”
- “Outputs”
  - Similar to “dependent” variables
- “Parameters”
  - Not “variables” because we are not sampling
- Time vs. simulated time
  - Time in the model may or may not correspond to actual time
- Data
  - “Empirical”: from the target
  - “Meta”: from the model





# Epistemology

- Questions?



# Steps of Research

1. Puzzle
2. Definition of target
3. Observations of target
4. Assumptions and design the model
  - a. “Environment”
  - b. “Agents”
  - c. Parameters and rules
  - d. Time
5. Simulate
6. Record data
7. *Verification*
8. *Validation*
9. *Sensitivity analysis*



# Pitfalls (1)

- Incorrect model specification
  - Agents: “thick ontological presuppositions” (Pepinsky)
  - “In that there is little consensus about the nature of the international system, simulation may not be an acceptable research tool for this enterprise.” p. 379
  - Importance of **validation**
- Reliance on strong assumptions
  - Space: “strict locality of interactions”
  - Relevant agents



# Pitfalls (2)

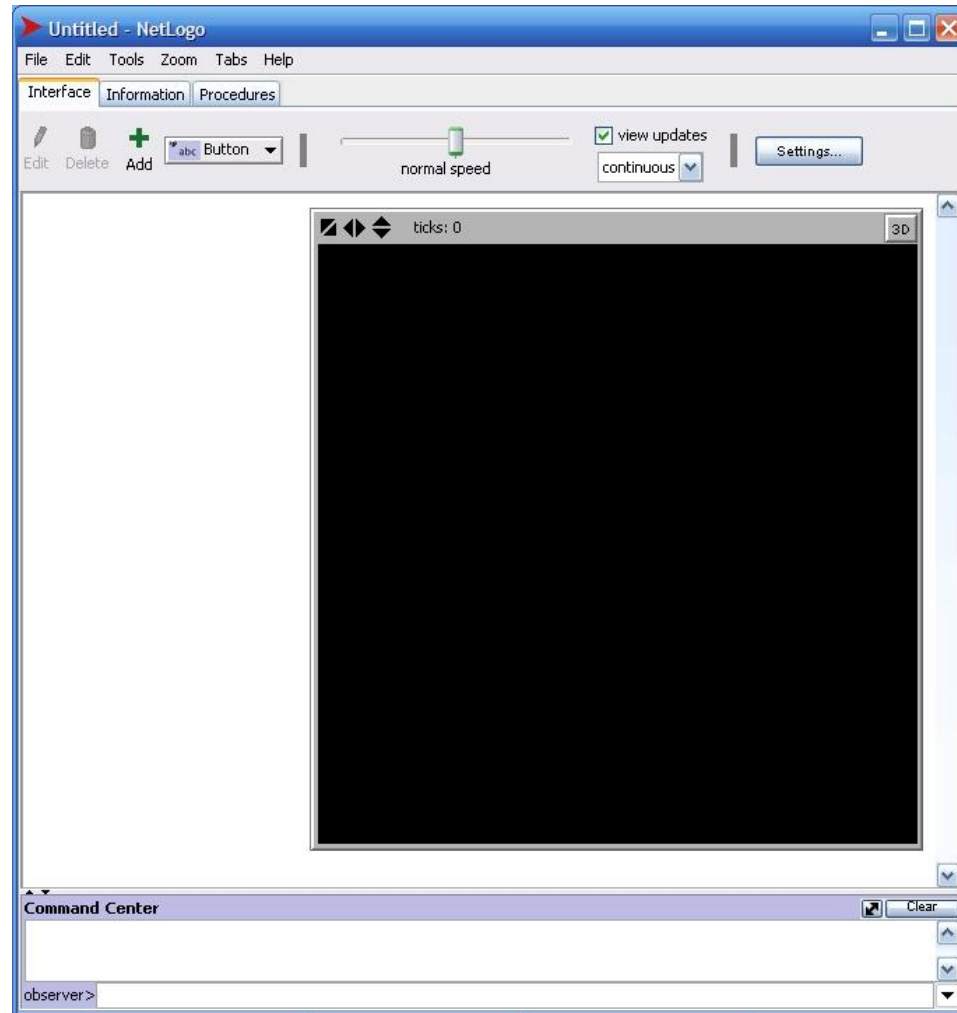
- Sensitivity of parameter specification: Chaos
  - Importance of **verification** and **sensitivity analysis**
- Quasi-experimentation: testing the *model*, not the *target* itself
- Epistemological and ontological claims tend to be obscured by the methodology
- Lack of consensus on epistemology of simulation



# III. INTRODUCTION TO NETLOGO

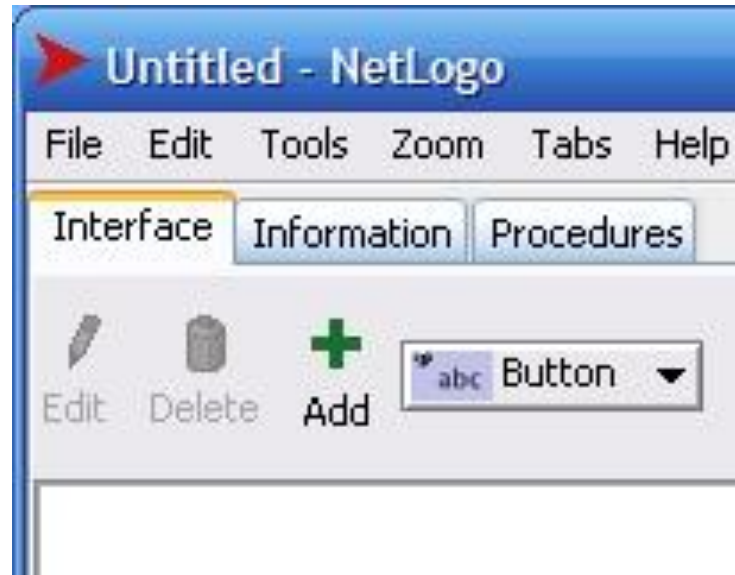


# Interface Elements

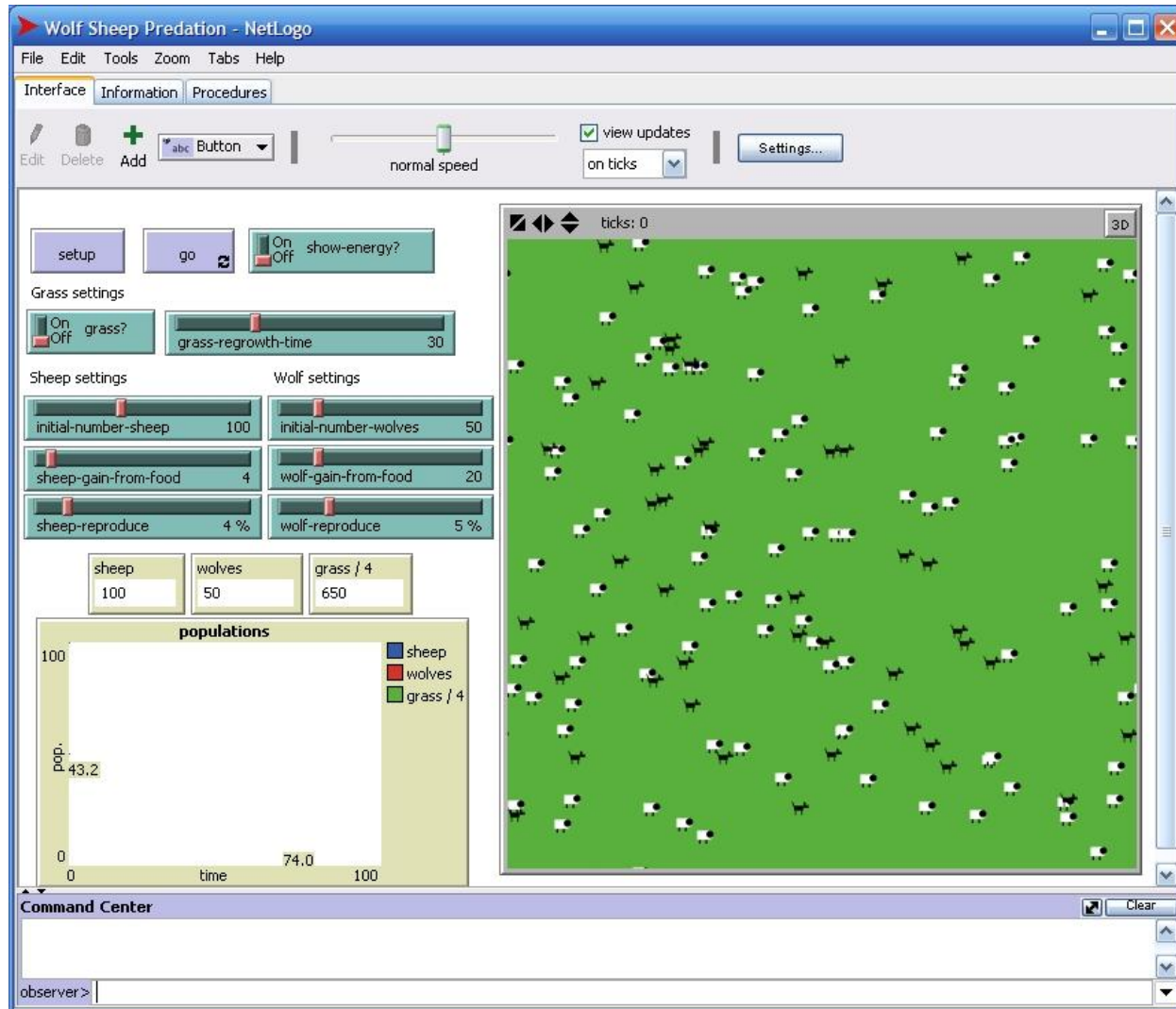


# Elements of NetLogo

- Interface tab
- Procedures tab
- Information tab

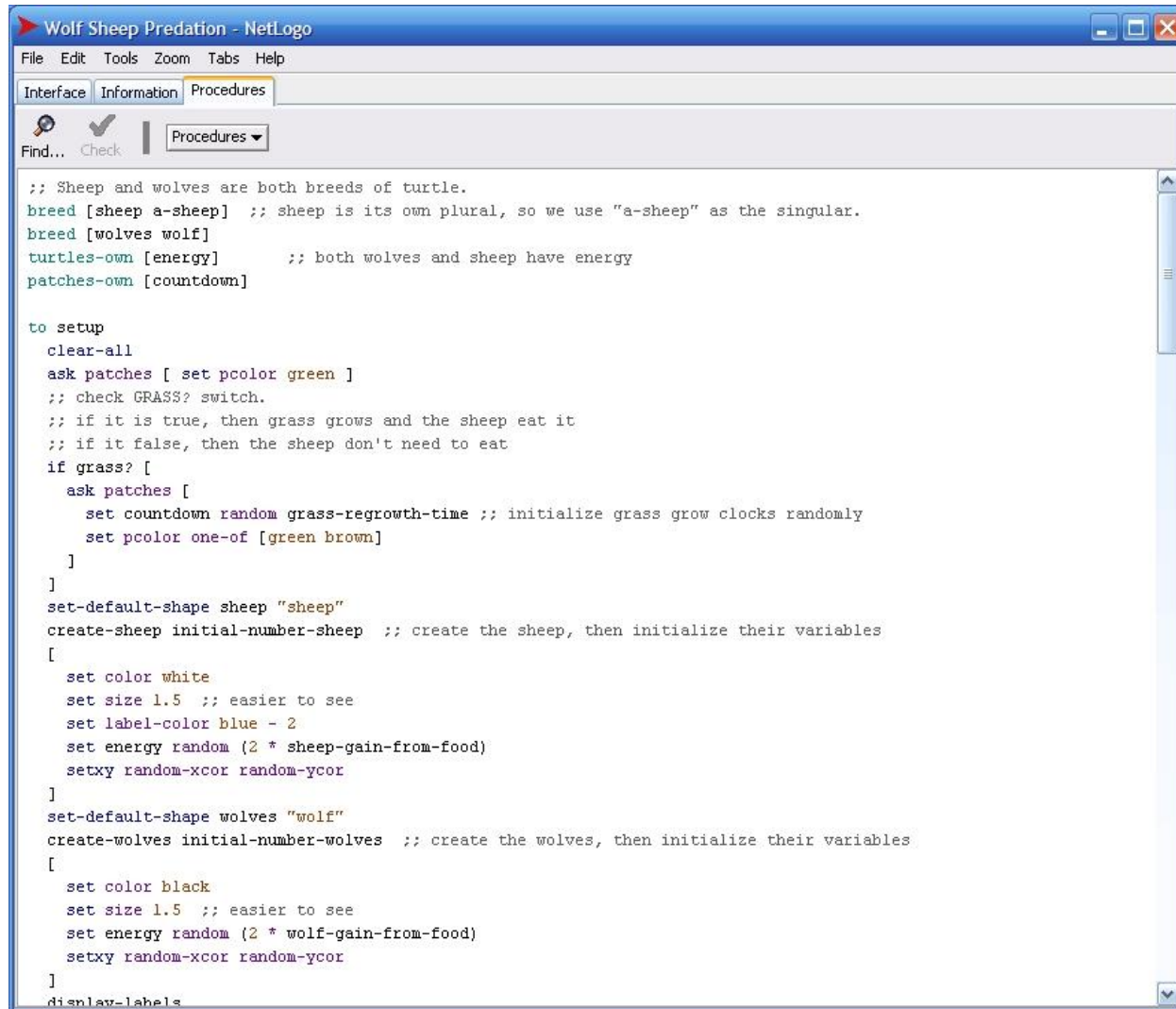


# Interface Tab





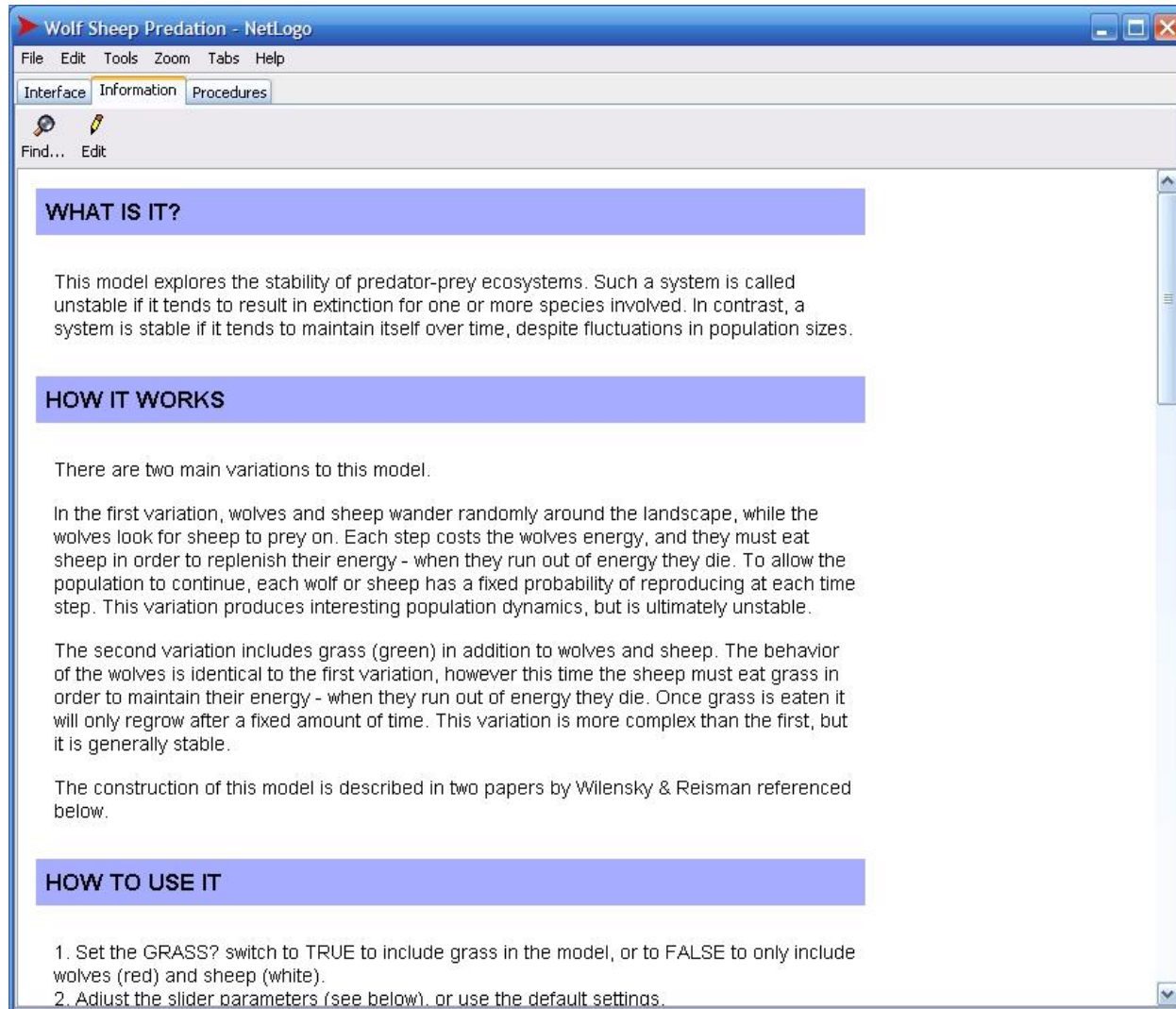
# Procedures Tab



```
;; Sheep and wolves are both breeds of turtle.
breed [sheep a-sheep] ;; sheep is its own plural, so we use "a-sheep" as the singular.
breed [wolves wolf]
turtles-own [energy]      ;; both wolves and sheep have energy
patches-own [countdown]

to setup
  clear-all
  ask patches [ set pcolor green ]
  ;; check GRASS? switch.
  ;; if it is true, then grass grows and the sheep eat it
  ;; if it false, then the sheep don't need to eat
  if grass? [
    ask patches [
      set countdown random grass-regrowth-time ;; initialize grass grow clocks randomly
      set pcolor one-of [green brown]
    ]
  ]
  set-default-shape sheep "sheep"
  create-sheep initial-number-sheep ;; create the sheep, then initialize their variables
  [
    set color white
    set size 1.5 ;; easier to see
    set label-color blue - 2
    set energy random (2 * sheep-gain-from-food)
    setxy random-xcor random-ycor
  ]
  set-default-shape wolves "wolf"
  create-wolves initial-number-wolves ;; create the wolves, then initialize their variables
  [
    set color black
    set size 1.5 ;; easier to see
    set energy random (2 * wolf-gain-from-food)
    setxy random-xcor random-ycor
  ]
  display-labels
```

# Information Tab



**Wolf Sheep Predation - NetLogo**

File Edit Tools Zoom Tabs Help

Interface Information Procedures

Find... Edit

## WHAT IS IT?

This model explores the stability of predator-prey ecosystems. Such a system is called unstable if it tends to result in extinction for one or more species involved. In contrast, a system is stable if it tends to maintain itself over time, despite fluctuations in population sizes.

## HOW IT WORKS

There are two main variations to this model.

In the first variation, wolves and sheep wander randomly around the landscape, while the wolves look for sheep to prey on. Each step costs the wolves energy, and they must eat sheep in order to replenish their energy - when they run out of energy they die. To allow the population to continue, each wolf or sheep has a fixed probability of reproducing at each time step. This variation produces interesting population dynamics, but is ultimately unstable.

The second variation includes grass (green) in addition to wolves and sheep. The behavior of the wolves is identical to the first variation, however this time the sheep must eat grass in order to maintain their energy - when they run out of energy they die. Once grass is eaten it will only regrow after a fixed amount of time. This variation is more complex than the first, but it is generally stable.

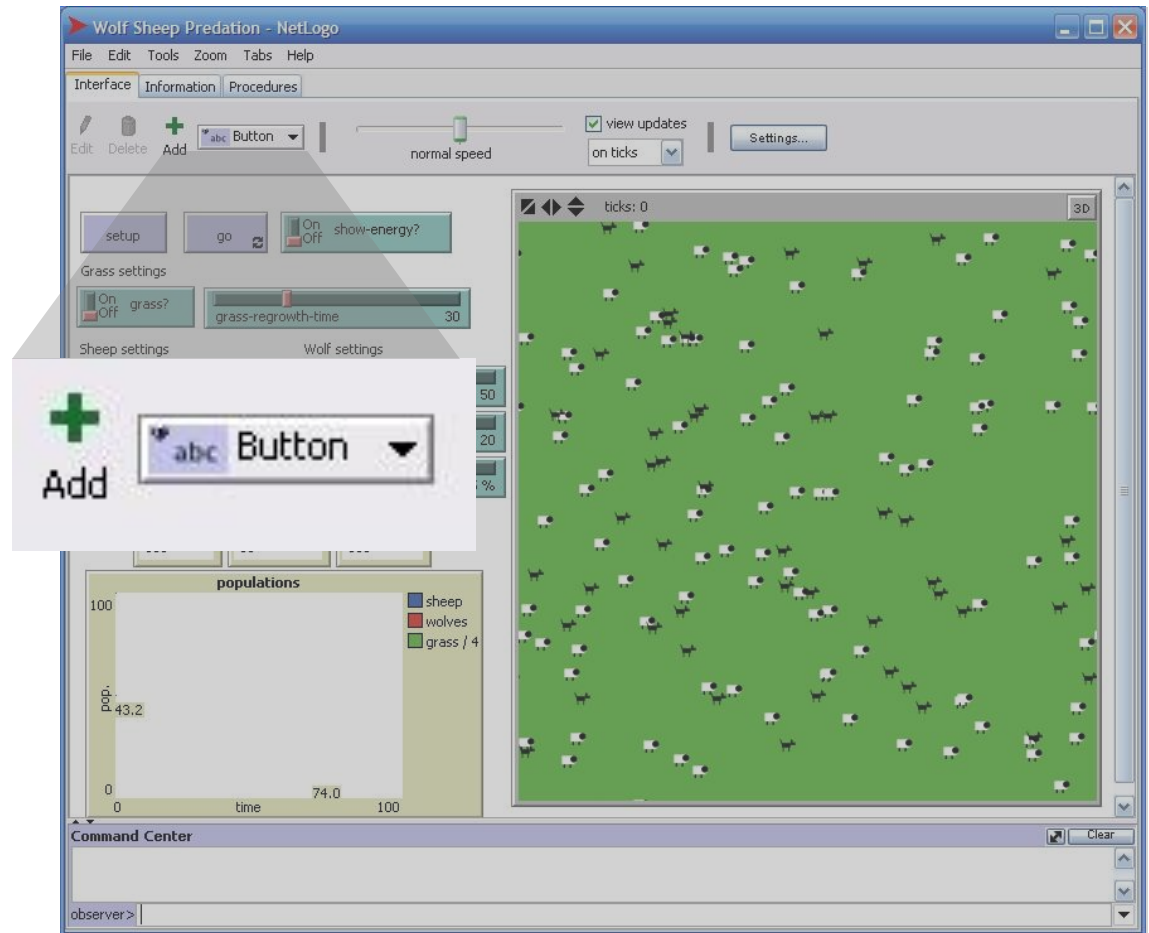
The construction of this model is described in two papers by Wilensky & Reisman referenced below.

## HOW TO USE IT

1. Set the GRASS? switch to TRUE to include grass in the model, or to FALSE to only include wolves (red) and sheep (white).
2. Adjust the slider parameters (see below), or use the default settings.

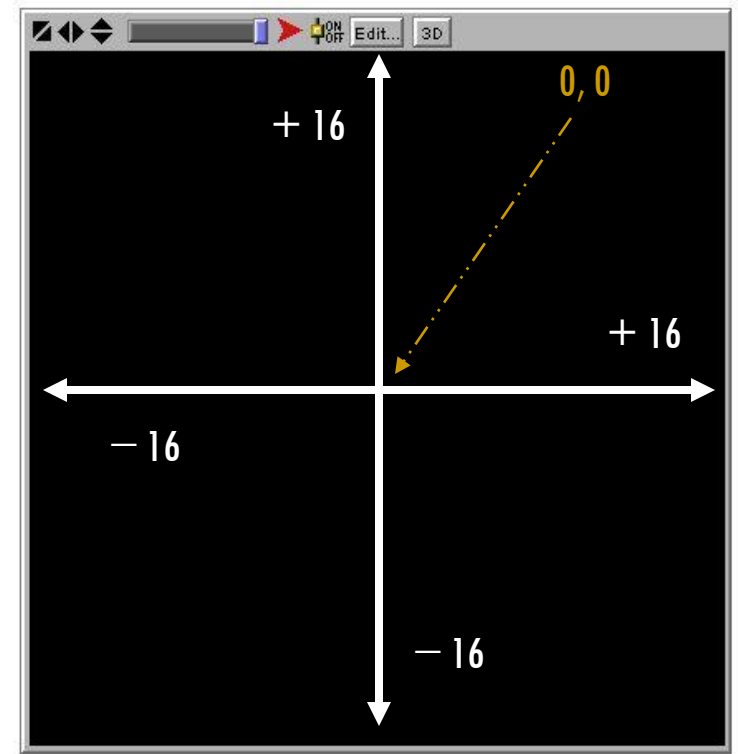
# Graphical User Interface Elements

- Buttons
- Sliders
- Switches
- Choosers
- Monitors and plots



# The NetLogo “World”

- A matrix
  - 33 x 33 square patches
  - **Cartesian coordinates**
- World is customizable
  - Size, Scale
  - Shape of patch (square versus hexagon)
  - Wrapping (none, top-bottom, left-right, all)
  - Origin



# Introduction

- Questions?

