Designing and Documenting Models: The ODD Protocol

EES 4760/5760

Agent-Based and Individual-Based Computational Modeling

Jonathan Gilligan

Class #4: Tuesday, Sept. 3 2019

Housekeeping at Start of Class:

Getting Started:

Today’s slides are at https://ees4760.jonathangilligan.org/Slides/Class_04

- Download and save the Butterfly model from https://ees4760.jonathangilligan.org/models/Class_04/butterfly_odd.nlogo
- Open NetLogo and load “butterfly_odd.nlogo”

Designing and Documenting Models

Designing and Documenting Models

ECOLOGICAL MODELLING 198 (2006) 115–126

ELSEVIER

available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/ecolmodel

ECOLOGICAL MODELLING

A standard protocol for describing individual-based and agent-based models

Volker Grimm[a,*], Uta Berger[b], Finn Bastiansen[a], Sigrunn Eliassen[c], Vincent Ginot[d], Jarl Giske[c], John Goss-Custard[e], Tamara Grand[f], Simone K. Heinz[c], Geir Huse[g], Andreas Huth[a], Jane U. Jepsen[a], Christian Jørgensen[c], Wolf M. Mooij[h], Birgit Müller[a], Guy Pe'er[i], Cyril Piou[b], Steven F. Railsback[j], Andrew M. Robbins[k], Martha M. Robbins[k], Eva Rossmanith[l], Nadja Rüger[a], Espen Strand[c], Sami Souissi[m], Richard A. Stillman[e], Rune Vabø[g], Ute Visser[a], Donald L. DeAngelis[n]

[a] UFZ Umweltforschungszentrum Leipzig-Halle GmbH, Department Ökologische Systemanalyse, Permoserstr. 15, 04318 Leipzig, Germany

[b] Zentrum für Marine Tropenökologie, Fahrenheitstr. 6, 28359 Bremen, Germany

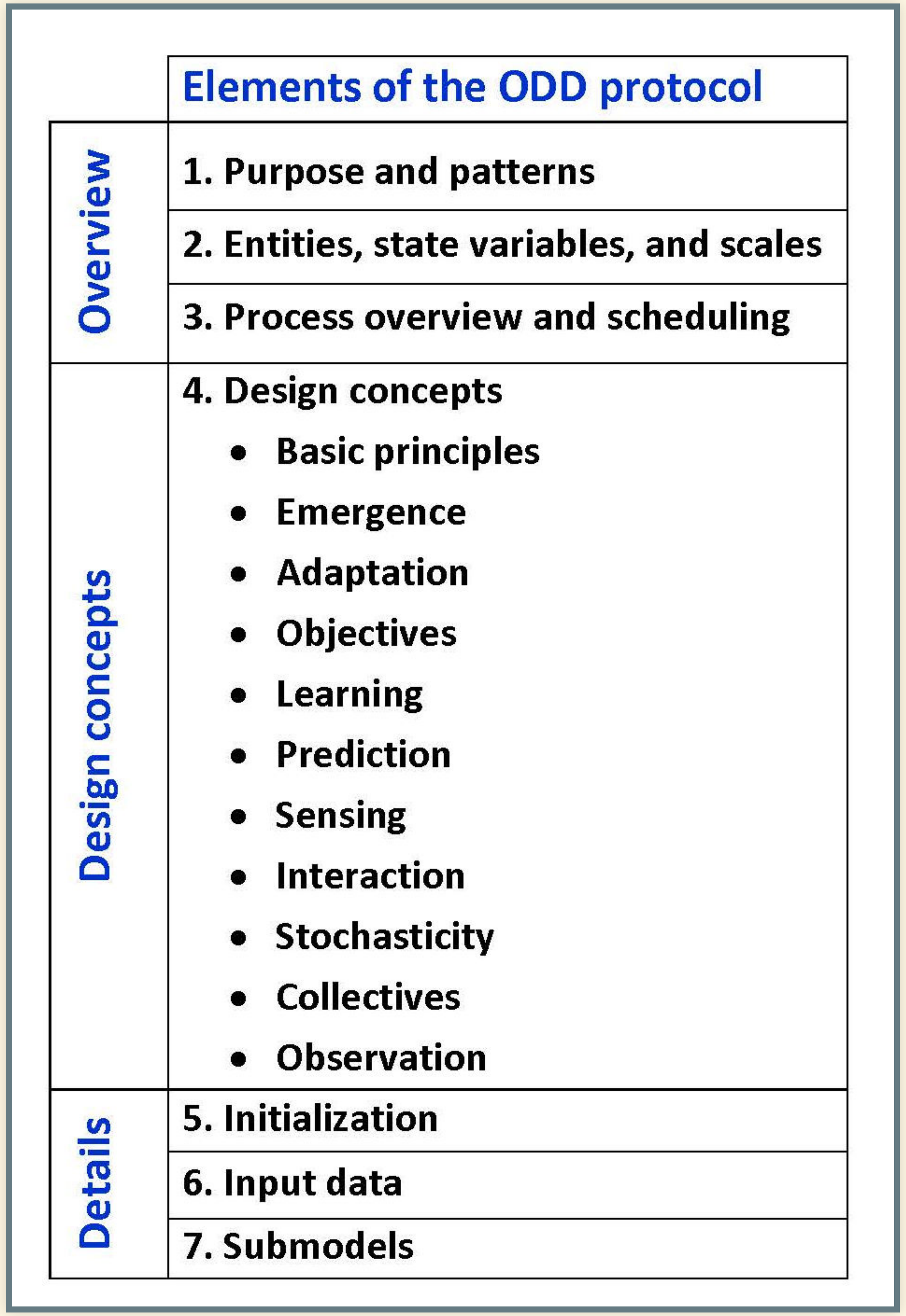
[c] University of Bergen, Department of Biology, P.O. Box 7800, N-5020 Bergen, Norway

[d] INRA, Unité de Biométrie, Domaine St.-Paul, 84 814 Avignon Cedex 9, France

Design

- Don’t start writing code until you know what you’re trying to do.
  - Big picture
    - What is the **purpose** of your model?
    - What **things** does your model use?
    - How do those things **behave**?
  - Design concepts
    - How do you represent the **things** in your model?
    - How do you implement their **behavior**?
    - How will your **things** and **behaviors** realize your **purpose**?
    - What **data** will you collect from your model?
    - How will you **use that data** to achieve your **purpose**?

Overview, Design Concepts, and Details



	Elements of the ODD protocol
Overview	1. Purpose and patterns
	2. Entities, state variables, and scales
	3. Process overview and scheduling
Design concepts	4. Design concepts • Basic principles • Emergence • Adaptation • Objectives • Learning • Prediction • Sensing • Interaction • Stochasticity • Collectives • Observation
Details	5. Initialization
	6. Input data
	7. Submodels

General

Detailed

ODD in perspective:

- Write overview and major parts of design concepts *first*
- As you write the model code, revisit and revise design concepts.
- Much of the details will emerge in the course of programming.
- When you are finished, write a complete ODD. This will be the major documentation for your model.

ODD Outline

1. Purpose & Patterns

Questions:

- What is the **purpose** of the model?
  - What are you trying to do?
  - ***Suggestion***: What will the *key graph* of your results look like?
- **Patterns:** What aspects are we modeling?
  - How will you judge whether it's realistic enough for the purpose?
  - What can you neglect?

2. Entities, State Variables, Scale

- What kinds of **entities** are in the model?

  Agents, collectives, spatial units, global environment, …
- What attributes (state-variables) characterize the entities?

  Age, sex, wealth, mood, opinion, soil type, land costs, rainfall, market price, …
  - You only need state-variables if they vary (different entities, different times, …)
  - Use the minimum number you need
- What are the temporal and spatial resolutions and extents of the model?

3. Process Overview and Scheduling

- How do states change?
- What entities do what, and in what order?
    - Schedule:
        1. Which entities take actions?
        2. What actions do they take?
            - Details of actions often go in **submodels**
        3. In what order do they take them?

4. Design Concepts

4. Design Concepts

There are **11 design concepts** (see Table 3.1).

Textbook has one chapter for each.

Outline of Design Concepts

- **Basic Principles:** Basis of model in general concepts and theories
- **Emergence:** What emerges as the model runs?
  (phenomena not imposed or directly programmed)
- **Adaptation** How do agents respond to changes in their environment?
  What decisions do they make, and how do they decide?
  Do they seek objectives directly (*deliberatly*) or indirectly (*mimic natural behavior*)?
- **Objectives (Fitness):** Goals of agents? What determines survival?
  Do objectives change as agent changes?
- **Learning:** Do individuals change behavior as they gain experience?
- **Prediction:** How do agents predict consequences of their decisions?
  (learning, memory, environmental cues, programmed assumptions)
- **Sensing:** What do agents know or perceive when making decisions?
  (Is sensing process itself explicitly modelled, or do they *just know*?)
- **Interaction:** What forms of interaction among agents are there?
- **Stochasticity:** Is there randomness in model? *Randomness must be justified!*
- **Collectives:** Grouping of individuals (Herds, social networks, …)
- **Observation:** How are data collected from model for analysis?

Details

5. Initialization

- What is the initial state of the model world?
- Time $t = 0$ of a simulation run
- In detail:
  - How many entities, of what type, are there initially?
  - What are the exact values of their state variables?
    (Or how were they set at random?)
  - Is initialization always the same,
    or does it vary from one simulation run to the next?
  - Are initial values chosen arbitrarily, or based on data?
  - References to those data should be provided.

6. Input data

- Does the model use input from external sources to represent processes that change over time?
  - data files, other models, human interaction
- If so, what data?
  - Where did they come from?
  - Provide references, citations.

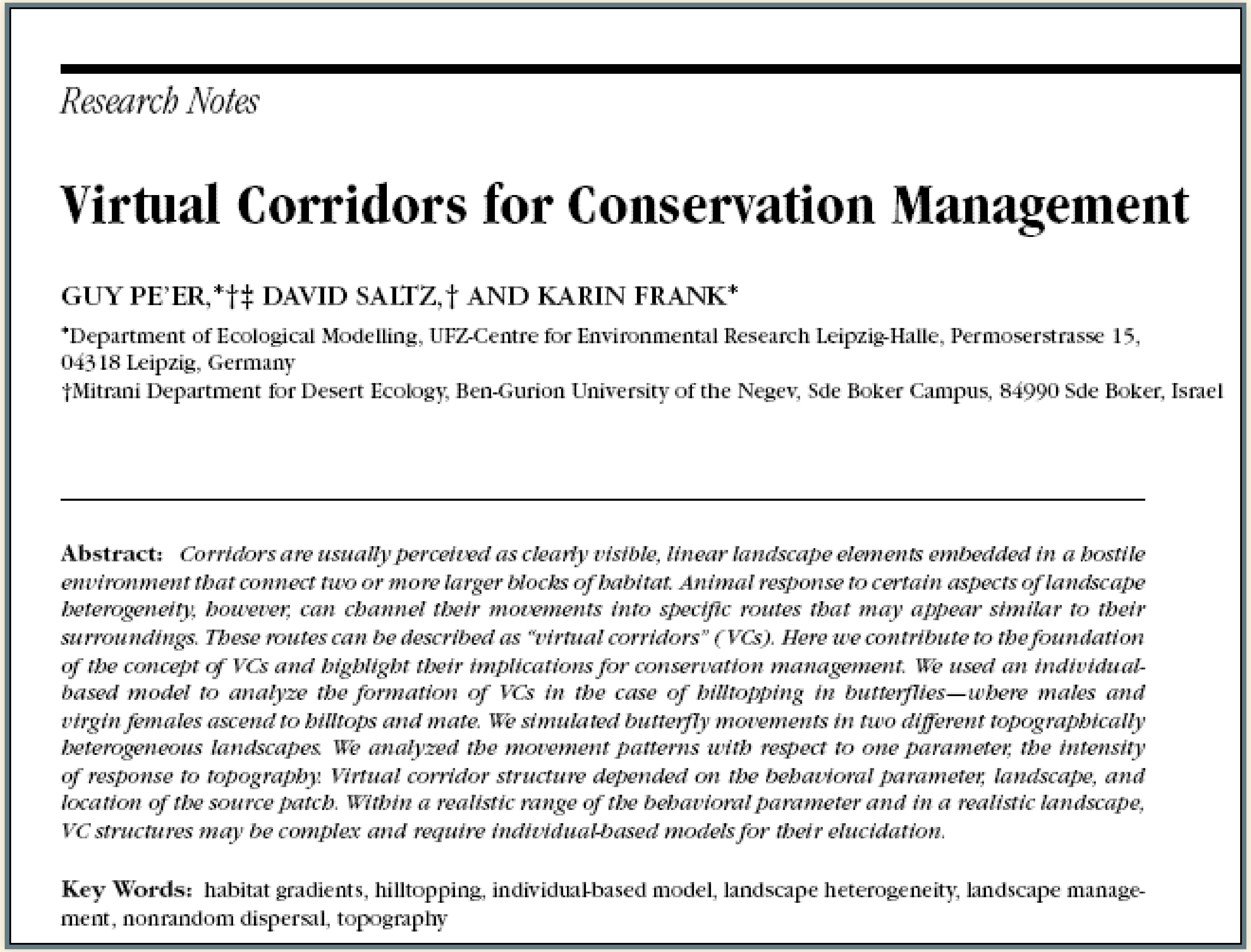
7. Submodels

If the **process scheduling** step contains a list of processes or actions, explain, in detail what **submodels** represent those processes or actions.

- What are the model parameters?
- How were the submodels designed or chosen?
- How were they tested?

Example: Virtual Corridors for Conservation Management

Example: Virtual Corridors for Conservation Management



Research Notes

Virtual Corridors for Conservation Management

GUY PE'ER,*†‡ DAVID SALTZ,† AND KARIN FRANK*

*Department of Ecological Modelling, UFZ-Centre for Environmental Research Leipzig-Halle, Permoserstrasse 15, 04318 Leipzig, Germany
†Mitrani Department for Desert Ecology, Ben-Gurion University of the Negev, Sde Boker Campus, 84990 Sde Boker, Israel

Abstract: *Corridors are usually perceived as clearly visible, linear landscape elements embedded in a hostile environment that connect two or more larger blocks of habitat. Animal response to certain aspects of landscape heterogeneity, however, can channel their movements into specific routes that may appear similar to their surroundings. These routes can be described as "virtual corridors" (VCs). Here we contribute to the foundation of the concept of VCs and highlight their implications for conservation management. We used an individual-based model to analyze the formation of VCs in the case of hilltopping in butterflies—where males and virgin females ascend to hilltops and mate. We simulated butterfly movements in two different topographically heterogeneous landscapes. We analyzed the movement patterns with respect to one parameter, the intensity of response to topography. Virtual corridor structure depended on the behavioral parameter, landscape, and location of the source patch. Within a realistic range of the behavioral parameter and in a realistic landscape, VC structures may be complex and require individual-based models for their elucidation.*

Key Words: habitat gradients, hilltopping, individual-based model, landscape heterogeneity, landscape management, nonrandom dispersal, topography

Pe'er, G., D. Saltz, and K. Frank, "Virtual corridors for conservation management," Conservation Biology **19**, 1997 (2005).

Butterfly Model in NetLogo

Open NetLogo and load “butterfly_odd.nlogo”

- Code section is blank, but ODD is filled in on “Info” tab.
    - You will fill in the code based on ODD while reading Chapter 4
- Click on “Edit” (pencil icon) to see what Info tab looks like when you edit it.
    - For details on editing “Info” tab, open NetLogo User Guide from the NetLogo Help menu and go to “Info Tab Guide” in the “Reference Section”

Purpose

- Ecologists observe that as butterflies move uphill, they concentrate into narrow and well-definied *virtual corridors* rather than following any old path to the top of the hill.
- Explore the concept of *virtual corridors:*

  Can concentrations of migrating animals emerge spontaneously from movement behavior and topography, instead of being a special habitat?
- Specifically, How does the concentration of hill-topping butterflies emerge from:
  - How butterflies move uphill
  - Landscape topography

Entities, State Variables, and Scales

- **Landscape:**
  - Square grid cells, with one *state variable*: **elevation**.
- **Butterflies:**
  - Have one *state variable*: **location** (discrete: which patch they’re in)

Entities, State Variables, and Scales

- **Spatial Scale:**
  - 150 × 150 cells
  - Corresponds to 25 × 25 meters in real landscape
- **Time Scale:**
  - Simulations last 1000 ticks
  - Tick length is unspecified (time for a butterfly to move one cell).

Process Overview and Scheduling

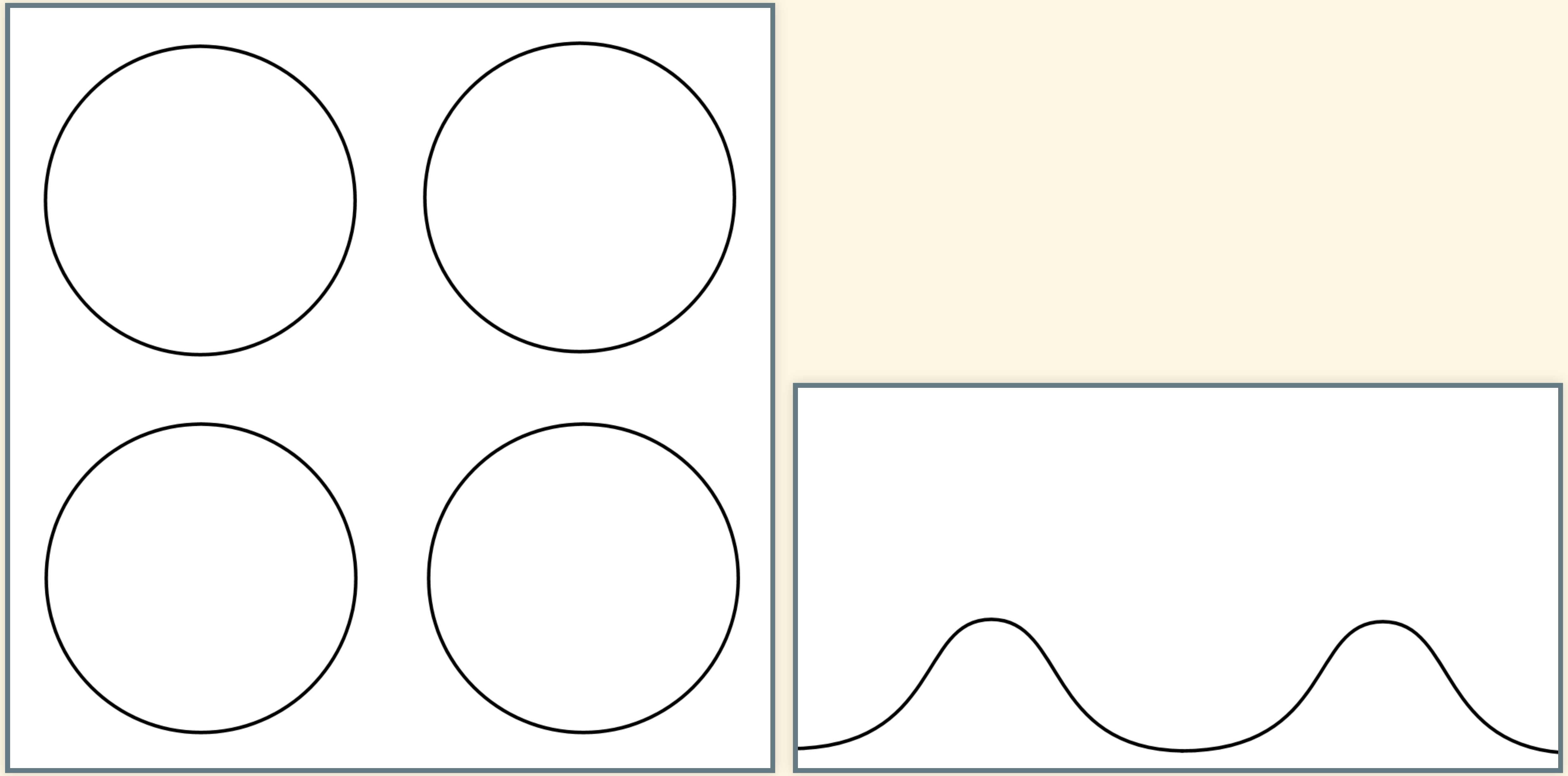
- **Only one process:** butterfly movement
  - On each tick, each butterfly moves once
  - The order in which butterflies move is unimportant because they don’t interact

Design concepts (important ones)

- **Basic Principles:** The concept of *virtual corridors*
- **Emergence:** results (concentration of butterflies in corridors) emerge from movement rule and topography
- **Sensing:** Butterflies can sense elevation in current and 9 surrounding cells
- **Interaction:** None
- **Stochasticity:** Used to represent reasons why butterflies do not move straight uphill
- **Observation:** We need a way to measure of butterfly concentration

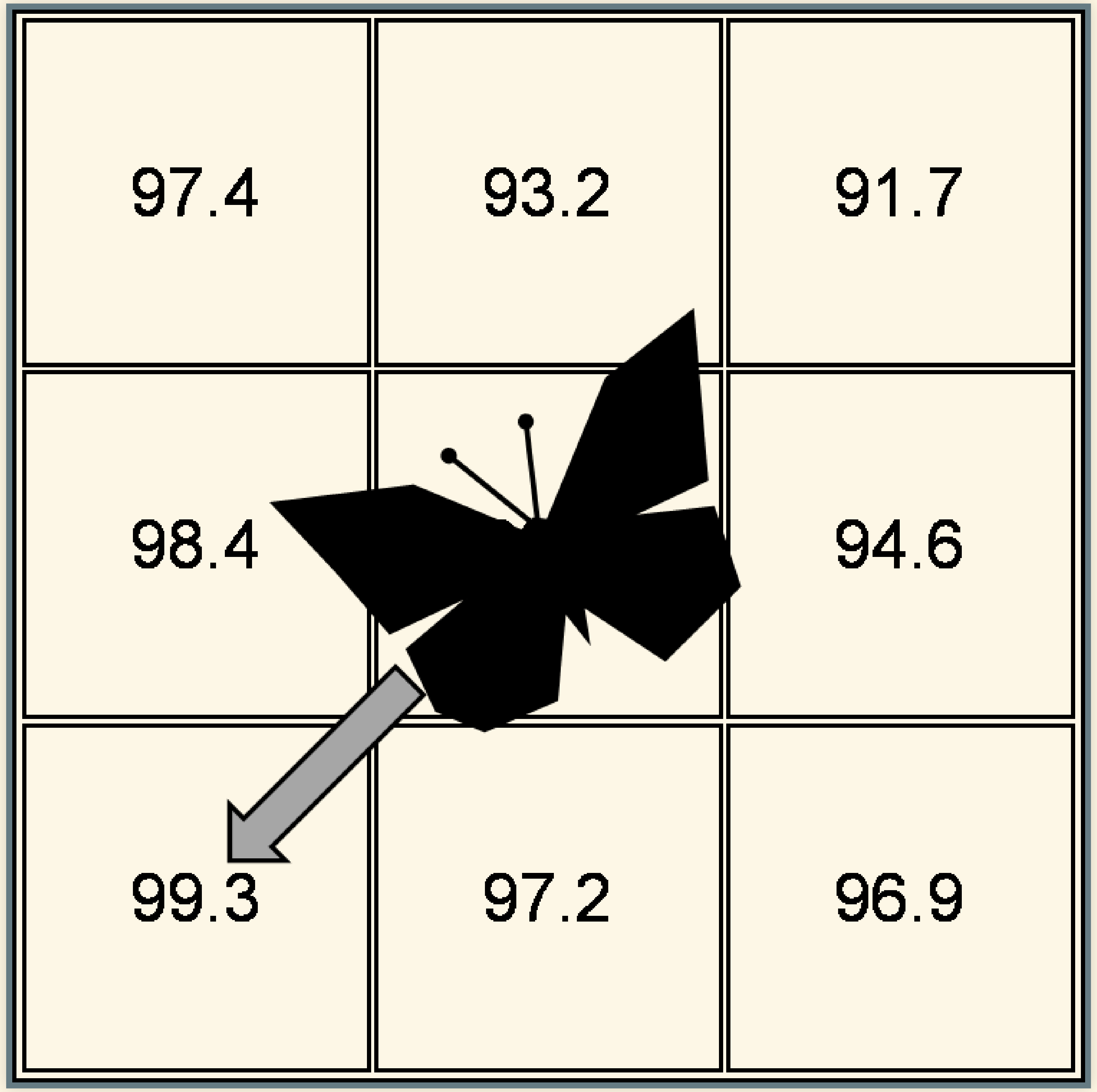
Initialization

- **Landscape:** cell elevations set to flat landscape with four hills
- **Butterflies:** 500 are created and placed in one cell



Submodel: Butterfly movement

- Global parameter *q* is probability that butterfly moves straight uphill, vs. moving to random neighbor cell.



97.4	93.2	91.7
98.4		94.6
99.3	97.2	96.9

Next Steps

Next Steps

For Thursday, you will follow the book and write code to implement the butterfly model.