

Agent-Based Modeling and Simulation

The ODD Protocol


Dr. Alejandro Guerra-Hernández

Instituto de Investigaciones en Inteligencia Artificial
Universidad Veracruzana
*Campus Sur, Calle Paseo Lote II, Sección Segunda No 112,
Nuevo Xalapa, Xalapa, Ver., México 91097*

`mailto:aguerra@uv.mx`
`https://www.uv.mx/personal/aguerra/abms`

Doctorado en Inteligencia Artificial 2024



- ▶ These slides are based on the book of Railsback and Grimm [5], chapter 3.
- ▶ Any difference with this source is my responsibility.
- ▶ This work is licensed under [CC-BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/) 
- ▶ To view a copy of this license, visit:

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Formulating an ABM

- ▶ It means progressing from the heuristic part of modeling, in which we first think about the problem, data, ideas, and hypotheses, to the first **formal** and **rigorous** representation of the model.
- ▶ We try to write it down in words, diagrams, equations, etc., which requires us to make a series of decisions about the **model's structure**.
- ▶ It is important to realize that a model simply does not exist before it has been **formulated explicitly** so people can **understand** it.

Purposes

- ▶ To make ourselves, the model's authors, **think explicitly** about all parts of the model.
- ▶ To **communicate** the model to our colleagues or supervisors, which usually leads to further discussions and modifications of the formulation.
- ▶ As the basis for the model's **implementation**.
- ▶ To **publish** results, including a complete and accurate description of the model.



The problem

- ▶ What **characteristics** of an ABM do we need to **describe**?
- ▶ How do we describe them **concisely** yet **clearly**?
- ▶ Ultimately, how do we **think** about the design of an ABM?
- ▶ **Answer.** Adhere to the **ODD Protocol** [1, 2, 3].



Why so early?

- ▶ We use the protocol throughout the **rest** of this course.
- ▶ To learn from the very start to **think** about ABMs in a **systematic** and **organized** way.



Learning Objectives

- ▶ To develop a firm understanding of the **Overview** and **Details** elements of ODD.
- ▶ To develop an introductory understanding of the **Design concepts** element.
- ▶ To understand, from its ODD description, the model we will **program** and use in the next two sessions.

Replicability

- ▶ It can be **difficult** to keep all of the ABM's characteristics in mind.
- ▶ Many ABMs in the literature are **incomplete**, which makes it impossible to **reimplement** them and **replicate** their results.
- ▶ Replication is key to science: models that cannot be reproduced are **unscientific**.
- ▶ Often, the description of the ABMs is **lengthy** and **hard** to read, even if the model in question is simple.
- ▶ How can we describe ABMs in a way that is **easy** to understand yet complete?
- ▶ **Answer. Standardization.** You know in advance what kind of information will be presented and what order will it appear in.



ODD components

Overview	1. Purpose and patterns
	2. Entities, state variables, and scales
	3. Process overview and scheduling
Design concepts	4. Design concepts <ul style="list-style-type: none">- Basic Principles- Emergence- Adaptation- Objectives- Learning- Prediction- Sensing- Interaction- Stochasticity- Collectives- Observation
	5. Initialization
	6. Input data
	7. Submodels



Purpose

- ▶ A clear and concise statement of the **question** or **problem** addressed by the model:
 - ▶ What **system** we are modeling?
 - ▶ What we are trying to **learn** about it?
- ▶ It is impossible to make any decisions about the model **without** purpose.
- ▶ Knowing a model's purpose is like having a **roadmap** to the rest of the model description.
- ▶ To describe the purpose in a way that is useful for formulating and understanding the model, is essential to be as **specific** as possible.



Tricks

- ▶ Sketching the **key graph** of the results section of the first publication you want to produce: what inputs (the x-axis) drive what outputs (the y-axis), in what units and with what kinds of variation (error bars, histograms, etc.).
- ▶ Saying what systems and problems the model is **not designed for** can often also be helpful and important for constraining the design.
- ▶ Using the word “Specifically,…” early on the statement of the purpose is a good way to **focus** the modeler and reader on exactly what the model is for.

Patterns

- ▶ This element addresses the question, what **system** are we modeling?
- ▶ Specifying the criteria you will use to decide when your model is **realistic enough** to be useful for your purpose.
- ▶ We can assume that a model is useful for its purpose when it **reproduces** a variety of patterns that are driven by the same processes that are important for the problem that the model is designed for.
- ▶ Equally important is knowing what patterns a model **does not reproduce**, as those patterns provide clues about what is missing or unrealistic about the model.
- ▶ Sources: abstractions, data, literature, expert opinion.



Entities

- ▶ What **kinds of things** are represented in the model?
 - ▶ One or more types of **agents**;
 - ▶ The **environment** where the agents live and interact.
- ▶ The environment is often broken into **local** units or **patches**; and the **global** environment that affects all agents.

State variables I

- ▶ Entities are characterized by their state variables which specifies how the model specifies their **state** at any time.
- ▶ An agent state is defined by:
 - Properties.** Or attributes, e.g., size, age, savings, memory, etc. The NetLogo agent variables.
 - Behavioral attributes.** Or strategies, e.g., searching behavior, bidding strategy, learning algorithm, etc. The procedures an agent can execute.
- ▶ State variables **do not include** quantities that can be deduced or calculated from the states of the agent and its environment, e.g., the distance to a given service, etc.



State variables II

- ▶ Some variables are **static** and do not change. Still, they are different among agents. Otherwise, there is **no need** for the variable, e.g., if only females are represented in the model.
- ▶ Many ABMs are **spatially explicit** and the space is often **heterogeneous**. The space can be continuous or discrete (patches).
- ▶ **Patches** are also characterized by state variables too, e.g., position, color, etc.

Global environment

- ▶ The global environment refers to **variables** that vary in **time**, but not necessarily in space, e.g., temperature, tax rates, etc.
- ▶ These global variables are provided by data or sub-models **external** to the ABMs.



Variables and patterns

- ▶ There is an important factor affecting the **choice of variables**: the patterns we are using to establish the desired level of structural realism.
- ▶ We may need to add variables just so we can **quantify** these patterns and how well the model matches observations, as discussed later concerning the **observation** design concept.

Temporal scale

- ▶ How time is represented:
 - Temporal extent. How long a time is simulated?
 - Temporal resolution. How the passage of time is simulated? The time step size, e.g, day, week, year, etc.
- ▶ The processes and changes happening at times shorter than a time step are only summarized and represented by how they make state variables jump from one time step to the next.
- ▶ The temporal extent is usually determined by system-level phenomena produced by the model, whereas temporal resolution is usually determined by the agent-level phenomena driving the model internally.

Spatial scale

- ▶ How space is represented?
 - ▶ **Spatial extent.** The total size of the space.
 - ▶ **Spatial resolution.** The size of the patches.
- ▶ The **right** spatial resolution of an ABM depends on key behaviors, interactions, and phenomena.
- ▶ Spatial relationships and effects within a grid cell are **ignored**; only spatial effects among cells are represented.

Process overview

- ▶ The previous element of the ODD is about **the structure of the model**.
- ▶ This element deals with the **dynamics** of the model: The processes that **change** the state variables of model entities.
- ▶ Every process, with one important exception, describes the **behavior** or dynamics of the model's entities.
- ▶ What are the entities doing?
- ▶ What behaviors do the agents execute as simulated time proceeds?
- ▶ What updates and changes happen in their environment?
- ▶ **Succinct** descriptions of these issues. Processes usually are treated as sub-models in the final part of the protocol.



Observer processes

- ▶ The only processes that are not linked to one of the model's entities are **observer** processes:
 - ▶ Observing and record what the model entities do, and
 - ▶ Why and when they do it, and
 - ▶ What patterns emerge from it all.
- ▶ We need to specify observation processes that do things like display the model's status on graphical displays and plots and write statistical summaries to output files.
- ▶ The ODD protocol includes a description of observer processes because the way we observe a model –the kind of **data** we collect from it and how we look at those data– can strongly affect how we **interpret** the model and what we learn from it.



Scheduling I

- ▶ The **schedule** defines the order in which the processes are executed by the computer.
- ▶ An ABM's schedule, when described well, provides a concise yet complete **outline** of the whole model (the go procedure).
- ▶ A model's schedule can be thought of as a sequence of actions; an **action** specifies (a) which model entities execute (b) which processes, in (c) what order.
- ▶ **Example:**

```
1 | ask turtles [move]
```

The ask statement causes the turtles to execute move **one at a time**, and in **randomized order**.



Scheduling II

- ▶ Some models rely on **other kinds of scheduling**, e.g., nonrandom execution order, asynchronous updating, discrete events, etc.
- ▶ Some schedules are simple enough to be described by simply **listing** the model processes in plain words.
- ▶ Others, more complicated, would require the use of **pseudo-code**.

Justification

- ▶ This section describes how the model implements a set of **basic concepts** that are important for designing ABMs.
- ▶ Such concepts provide a **standardized** way to think about very important characteristics of ABMs that **cannot** be described well using other conceptual frameworks, e.g., differential equations.
- ▶ **Examples.** Emergent outcomes, adaptive decisions, collectives, etc.
- ▶ For each one of the eleven design concepts in the ODD protocol, a set of questions is proposed to make **design decisions** about them conscious.
- ▶ Not all of these concepts are **relevant** for all ABMs.



Basic principles

1. What general **concepts, theories, hypotheses, or modeling** approaches underlie the model's design? How is the model related to **previous thinking** about the problem it addresses?
2. How were these principles **incorporated** in the model's design? Does the model **implement** the principles in its design; or address them as a **study topic**, e.g., by evaluating and proposing alternatives to them?

Emergence

3. What are the model's important **results** and **outputs**? Which of them **emerge** from mechanistic representation of the adaptive behavior of individuals, and which are **imposed** by rules that force the model to produce certain results?



Adaptation

4. What adaptive behaviors do agents have, and why? In what ways can they **respond to changes** in their environment and themselves? What **decisions** do they make?
5. How are these behaviors modeled? Do **adaptive traits** (models of adaptive behavior) assume agents choose among alternatives by explicitly considering which is the most likely to increase some specific objective (**direct objective-seeking**), or do traits simply force agents to reproduce behavior patterns observed in real systems (**indirect objective-seeking**)?



Objectives

6. For direct objective-seeking, what **measure** is used to rate decision alternatives? This measure is the agent's internal model of how it would benefit from each **choice** it might make, e.g., fitness, utility, etc. What elements of **future success** are in the objectives measure, e.g., survival, profit, etc.? How does the objective measure represent processes that link adaptive behaviors to important variables of the agents and their environment?
7. How were the variables and mechanisms in the objective measure chosen considering the model's **purpose** and the real system it represents? How is the agent's current **internal state** considered in the modeling decisions? e.g., mortality, bankruptcy, etc., Does the objective measure **change** as the agents change?



Learning

8. Do individuals **change** their adaptive traits over time as a consequence of their experience? If so, **how**?



Prediction

9. How do agents predict **future conditions** (environmental and internal) in their adaptive traits? What assumptions about, or mechanisms of, the real individuals being modeled were the basis for how prediction is modeled?
10. How does simulated prediction make use of **mechanisms** such as memory, learning, or environmental cues? Or is prediction “tacit”: only implied in simple adaptive traits, i.e., only implied in simple rules for adaptive behavior?



Sensing

11. What variables of their **environment** and **themselves** are agents assumed to sense and therefore be able to consider in their behavior? What is the basis of these assumptions.
12. What sensing **mechanisms** are modeled explicitly, and which sensed variables are agents instead assumed simply to “know”?
13. With what **accuracy** or uncertainty are agents assumed to “know” or sense which variables? Over what distances (in geographic, network, grid, or other space)?

Interaction

14. How do model's agents interact? Do they interact **directly** with each other, e.g., does one agent directly change the state of others? Or is interaction **mediated**, such as via competition for a resource?
15. With which **other** agents does an agent interact?
16. What **real** interaction mechanisms were the model's representation of interaction based on? At what spatial and temporal scales do they occur?

Stochasticity

17. How are stochastic processes (based on **pseudo-random** numbers) used in the model and why? Are stochastic processes used:
- ▶ To **initialize** the model?
 - ▶ Because it is believed important for some processes to be variable but unimportant to represent the **causes** of variability?
 - ▶ To reproduce observed behaviors using **empirically** determined probabilities?

Collectives

18. Are collectives –agregations of agents that affect the state or behavior of member agents and are affected by their members– represented in the model?
19. If so, how are collectives represented? Do they emerge from the traits of agents, or are agents given traits that impose the formation of collectives? Or are the collectives modeled as another type of agent with its own traits and state variables?



Observation

20. What **outputs** of the model are needed to observe its internal dynamics as well as its system-level behavior? What **tools**, e.g., graphics, file output, data on individuals, etc., are needed to obtain these outputs?
21. What outputs and **analyses** are needed to **test** the model against the criteria of usefulness –usually, a set of **patterns**– defined in the Purpose and Patterns element? What outputs are needed to solve the problem the model was designed for?



Initialization

- ▶ We must describe how we **set up** the model World at the beginning of the simulation, because results of the model often **depend** on these **initial conditions**.
- ▶ **Examples**. Number of agents created and the initial values given to their state variables, e.g., location, size, etc.; and how the initial values of the environment variables are set.
- ▶ Sometimes we want a model's result to **depend** on its initial conditions, e.g., to understand how the system responds to a new situation; In other cases, we want to make model results **independent** of the initial conditions.
- ▶ To make results **reproducible**, we have to specify the initial state of all the state variables of all entities in the model.



Input Data

- ▶ Models often include environmental variables, e.g, temperature or market price that change over time and are **read** into, instead of simulated within, the model.
- ▶ These inputs are usually read in from data files as the model **executes**.
- ▶ Input here does not refer to **parameter** values or initialization data, which are also sometimes read in from files at the start of a simulation.



Submodels

- ▶ Up to this point, the ODD protocol has specified the **skeleton** of the model: its entities, state variables, and the names all of the processes and how they are scheduled.
- ▶ Here, we put flesh on the bones: all the **major processes** in the model are considered submodels.
- ▶ You can think of a submodel as a model of one process in the ABM; submodels are often almost completely **independent** of each other and can be designed and tested independently.
- ▶ We must describe **all** equations, logical rules, and algorithms that constitute the submodels. We also need to document **why** we formulated the submodels as we did, e.g., literature, assumptions, parameter values, test, and calibration.



Corridors

- ▶ Many animals **disperse** –leave their home location and move long distances for purposes such as mating—at some point in their life.
- ▶ Dispersing animals respond to the **landscape**, avoiding some features and being attracted to others.
- ▶ These behavioral responses to the landscape can channel their movement into pathways referred to as **corridors** –Linear elements in the landscape that facilitate dispersal; e.g., include hedgerows, fences, and vegetation along roads.
- ▶ However, our perception of corridors certainly is limited because we can't see the landscape through the animals' eyes. Could it be that some places where we see high numbers of dispersing animals are **virtual corridors**? Subtle emergence?



A Model for Corridors

- ▶ Pe'Er, Saltz, and Frank [4] chose an extremely simple system in which it is easy to observe real individuals in corridors: **mate-finding** by butterflies.
- ▶ Many butterfly species adopt a **hilltopping** strategy to mate: They simply move uphill until they are concentrated on hilltops where they can meet and mate.
- ▶ The proposed model is an example of how **simple** a model can be and still capture the essence of a system with regard to a certain question.
- ▶ Because the model is so simple, we avoid brainstorming to proceed directly to the ODD protocol formulation.



Visually [4]

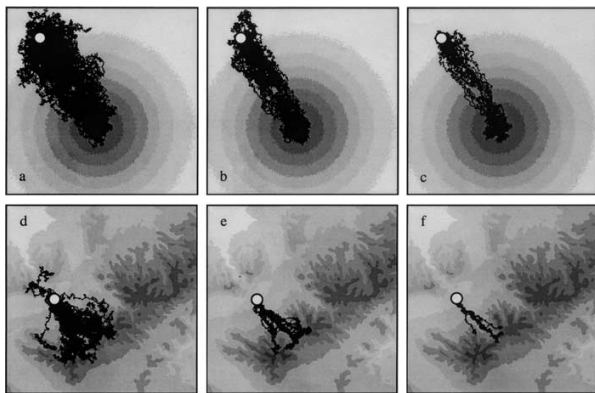


Figure 1. The trajectories of 100 individual butterflies (black marks) moving 1000 time steps from a single source patch (white circle) toward the mountain summit/s on (a-c) a virtual landscape with one summit and on (d-f) a realistic landscape of 7500×7500 m (cell size = 25 m). We used different probabilities of the butterflies moving upward (q): (a,d) $q = 0.2$; (b,e) $q = 0.5$; and (c,f) $q = 0.8$. Gray shading reflects elevation (darker is bigger).



Purpose

- ▶ The model was designed to **explore** questions about virtual corridors.
- ▶ Under what conditions do the interactions of butterfly hilltopping **behavior** and landscape **topography** lead to the emergence of virtual corridors, that is, relatively narrow paths along which many butterflies move?
- ▶ How does **variability** in the butterflies' tendency to move uphill affect the emergence of virtual corridors?
- ▶ Only **general patterns** are used as criteria for its usefulness: that butterflies can reach hilltops, and that their movement has a strong stochastic element representing the effects of factors other than elevation.

Entities, Variables, and Scales

- ▶ Entities: **butterflies** and square **patches** of land.
- ▶ The patches make up a square grid landscape of 150×150 patches, and each patch has one state variable: its **elevation**.
- ▶ Butterflies are characterized only by their **location** –the x- and y-coordinates of the center of their patch.
- ▶ Patch size and the length of one time step in the simulation are not specified because the model is **generic**.
- ▶ But, when **real landscapes** are used, a patch corresponds to 25×25 m². Simulations last for 1000 time steps; the length of one time step is not specified but should be about the time it takes a butterfly to move 25–35 m.



Process Overview and Scheduling

- ▶ There is only one process in the model: **movement** of the butterflies.
- ▶ On each time step, each butterfly moves once. The order in which the butterflies execute this action is unimportant because there are **no interactions** among the butterflies.

Design concepts I

- ▶ The **basic principle** addressed by this model is the concept of virtual corridors –pathways used by many individuals when there is nothing particularly beneficial about the habitat in them.
- ▶ An **alternative view** to the assumption that the corridor itself is attractive in some way.
- ▶ This concept is addressed by seeing when corridors **emerge** from two parts of the model: the adaptive movement behavior of butterflies and the landscape they move through.
- ▶ This **adaptive behavior** is modeled via a simple empirical rule that reproduces the behavior observed in real butterflies: moving uphill.



Design concepts II

- ▶ This behavior is based on the understanding (**not included** in the model) that moving uphill leads to mating, which conveys fitness (success at passing on genes, the presumed ultimate objective of organisms).
- ▶ Because the hilltopping behavior is assumed *a priori* to be the objective of the butterflies, the concepts of **Objectives** and **Prediction** are not explicitly considered.
- ▶ There is no **learning** in the model.
- ▶ **Sensing** is important in this model: butterflies are assumed able to identify which of the surrounding patches has the highest elevation, but to use no information about elevation at further distances.



Design concepts III

- ▶ Although real butterflies stop to visit each other on the way uphill, it was decided it is not important to include such **interactions** in the model.
- ▶ **Stochasticity** is used to represent two sources of variability in movement that are too complex to represent mechanistically. Real butterflies do not always move directly uphill, likely because of:
 1. Limits in the ability of the butterflies to sense the highest area in their neighborhood, and
 2. Factors other than topography (e.g., flowers that need investigation along the way) that influence movement direction.

Design concepts IV

- ▶ This variability is represented by assuming butterflies do not move uphill every time step; sometimes they move **randomly** instead.
- ▶ Whether a butterfly moves directly uphill or randomly at any time step is modeled stochastically, using a **parameter** q that is the probability of an individual moving directly uphill instead of randomly.
- ▶ To allow **observation** of virtual corridors, we will define a specific **corridor width** measure that characterizes the width of a butterfly's path from its starting patch to a hilltop.



Initialization

- ▶ The topography of the landscape (**elevation** of each patch) is initialized when the model starts.
- ▶ Two kinds of **landscapes** are used in different versions of the model:
 1. A simple artificial topography, and
 2. The topography of a real study site, imported from a file containing elevation values for each patch.
- ▶ The butterflies are initialized by creating five hundred of them and setting their **initial location** to a single patch selected randomly.

Input Data

- ▶ The environment is assumed to be **constant**, so the model has no input data.

Sub-models I


- ▶ The **movement** submodel defines exactly how butterflies decide whether to move uphill or randomly.
- ▶ First, to **move uphill** is defined specifically as moving to the neighbor patch that has the highest elevation; if two patches have the same elevation, one is chosen randomly.
- ▶ **Move randomly** is defined as moving to one of the neighboring patches, with equal probability of choosing any patch.
- ▶ **Neighbor patches** are the eight patches surrounding the butterfly's current patch.



Sub-models II

- ▶ The **decision** of whether to move uphill or randomly is controlled by the parameter q , which ranges from 0.0 to 1.0 (a global variable: all butterflies use the same value).
- ▶ On each time step, each butterfly draws a random number from a **uniform distribution** between 0.0 and 1.0. If this random number is less than q , the butterfly moves uphill; otherwise, the butterfly moves randomly.

Visually

97.4	93.2	91.7
98.4		94.6
99.3	97.2	96.9



Benefits

- ▶ Describing a model on **paper** is perhaps the most important part of modeling: very few benefits of modeling can be achieved without it.
- ▶ For ABMs, it is especially important to use **standard concepts** and **formats** to describe and design models: these models are complex, so we need a clear, standard way to think and write about them.
- ▶ The standard **languages** of differential equations and statistical modeling cannot describe ABMs, so instead we use the ODD protocol and its design concepts.
- ▶ Without such a standard, ABMs are often too incompletely described to be **replicated**, which makes them unscientific.



Generic and Hierarchical

- ▶ ODD is generic and hierarchical.
- ▶ **Generic** means that it can be used to describe any ABM in any field of application or discipline.
- ▶ **Hierarchical** means that it starts with an overview of a model's structure, scales, processes, and scheduling, so we can understand the model's basics, before presenting the details needed to understand how processes are actually represented.
- ▶ In between, important concepts underlying the model's design are explained, for example: what key behaviors in the model are emergent instead of being imposed? To what extent and why is stochasticity included? How do we observe model behavior to better understand how model output emerges?



A Framework

- ▶ The ODD protocol provides a very useful framework to **think** about and formulating ABMs.
- ▶ Once a model is written down in ODD format, it is very clear how to translate it into a NetLogo **program**.

Referencias I

- [1] V Grimm et al. “A standard protocol for describing individual-based and agent-based models”. In: *Ecological Modelling* 198.1–2 (2006), pp. 115 –126. ISSN: 0304-3800. URL: <http://www.sciencedirect.com/science/article/pii/S0304380006002043>.
- [2] V Grimm et al. “The ODD protocol: A review and first update”. In: *Ecological Modelling* 221.23 (2010), pp. 2760 –2768. ISSN: 0304-3800. URL: <http://www.sciencedirect.com/science/article/pii/S030438001000414X>.
- [3] V Grimm et al. “The ODD protocol for describing agent-based and other simulation models: A second update to improve clarity, replication, and structural realism”. In: *Journal of Artificial Societies and Social Simulation* 23.2 (2020), p. 7.
- [4] G Pe’Er, D Saltz, and K Frank. “Virtual corridors for conservation management”. In: *Conservation Biology* 19.6 (2005), pp. 1997–2003.
- [5] SF Railsback and V Grimm. *Agent-Based and Individual-Based Modeling*. Second. Princeton, NJ, USA: Princeton University Press, 2019.

