A Standard Protocol for Describing Individual-Based and Agent Based Models

Article *in* Ecological Modelling · September 2006

DOI: 10.1016/j.ecolmodel.2006.04.023 · Source: OAI

Volker Grimm
Helmholtz-Zentrum für Umweltforschung

Uta Berger
Technische Universität Dresden

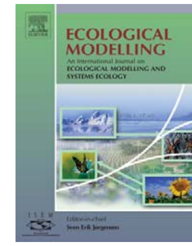
Finn Bastiansen
Rifcon

Sigrunn Eliassen
University of Bergen

available at www.sciencedirect.com

ScienceDirect

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



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*
[e] *Centre for Ecology and Hydrology, Winfrith Technology Centre, Dorchester DT2 8ZD, UK*
[f] *108 Roe Drive, Port Moody, British Columbia V3H 3M8, Canada*
[g] *Institute of Marine Research, Box 1870, Nordnes, N-5817 Bergen, Norway*
[h] *Netherlands Institute of Ecology, Centre for Limnology, Rijksstraatweg 6, 3631 AC Nieuwersluis, Netherlands*
[i] *Hebrew University of Jerusalem, Institute of Life Sciences, Department of Evolution, Systematics and Ecology, Givat Ram Campus, Jerusalem 91904, Israel*
[j] *Lang, Railsback & Associates, 250 California Ave., Arcata, CA 95521, USA*
[k] *Max Planck Institute for Evolutionary Anthropology, Deutscher Platz 6, 04103 Leipzig, Germany*
[l] *Universität Potsdam, Institut für Biochemie und Biologie, Maulbeerallee 2, 14469 Potsdam, Germany*
[m] *Université des Sciences et Technologies de Lille, Station Marine de Wimereux, Ecosystem Complexity Research Group, CNRS—UMR 8013 ELICO, 28 Avenue Foch BP 80, F-62930 Wimereux, France*
[n] *USGS/Florida Integrated Science Centers and Department of Biology, University of Miami, P.O. Box 249118, Coral Gables, FL 33124, USA*

ARTICLE INFO

Article history:
Received 8 November 2005
Received in revised form 24 April 2006
Accepted 26 April 2006
Published on line 14 June 2006

Keywords:
Individual-based model
Agent-based model
Model description
Scientific communication
Standardization

ABSTRACT

Simulation models that describe autonomous individual organisms (individual based models, IBM) or agents (agent-based models, ABM) have become a widely used tool, not only in ecology, but also in many other disciplines dealing with complex systems made up of autonomous entities. However, there is no standard protocol for describing such simulation models, which can make them difficult to understand and to duplicate. This paper presents a proposed standard protocol, ODD, for describing IBMs and ABMs, developed and tested by 28 modellers who cover a wide range of fields within ecology. This protocol consists of three blocks (Overview, Design concepts, and Details), which are subdivided into seven elements: Purpose, State variables and scales, Process overview and scheduling, Design concepts, Initialization, Input, and Submodels. We explain which aspects of a model should be described in each element, and we present an example to illustrate the protocol in use. In addition, 19 examples are available in an Online Appendix. We consider ODD as a first step for establishing a more detailed common format of the description of IBMs and ABMs. Once initiated,

* *Corresponding author.* Tel.: +49 341 235 2903; fax: +49 341 235 3500.
E-mail address: volker.grimm@ufz.de (V. Grimm).

doi:10.1016/j.ecolmodel.2006.04.023

the protocol will hopefully evolve as it becomes used by a sufficiently large proportion of modellers.

1. Introduction

Simulation models that describe individual organisms or, more generally, "agents", have become a widely used tool, not only in ecology (DeAngelis and Gross, 1992; DeAngelis and Mooij, 2005; Grimm, 1999; Grimm and Railsback, 2005; Huse et al., 2002; Shugart et al., 1992; Van Winkle et al., 1993) but also in many other disciplines dealing with complex systems made up of autonomous entities, including the social sciences (Epstein and Axtell, 1996; Gilbert and Troitzsch, 2005), economics (Tesfatsion, 2002), demography (Billari and Prskawetz, 2003), geography (Parker et al., 2003), and political sciences (Axelrod, 1997; Huckfeldt et al., 2004). Individual-based models (IBMs) allow researchers to study how system level properties emerge from the adaptive behaviour of individuals (Railsback, 2001; Strand et al., 2002) as well as how, on the other hand, the system affects individuals. IBMs are important both for theory and management because they allow researchers to consider aspects usually ignored in analytical models: variability among individuals, local interactions, complete life cycles, and in particular individual behaviour adapting to the individual's changing internal and external environment.

However, the great potential of IBMs comes at a cost. IBMs are necessarily more complex in structure than analytical models. They have to be implemented and run on computers. IBMs are more difficult to analyze, understand and communicate than traditional analytical models (Grimm et al., 1999). Particularly critical is the problem of communication. Analytical models are easy to communicate because they are formulated in the general language of mathematics. Their description usually is complete, unambiguous and accessible to the reader. In contrast, published descriptions of IBMs are often hard to read, incomplete, ambiguous, and therefore less accessible. Consequently, the results obtained from an IBM are not easily reproduced (Hales et al., 2003). Science, however, is based on reproducible observations. Solving the problem of how to communicate IBMs can only increase their scientific credibility (Ford, 2000; Lorek and Sonnenschein, 1999).

There are two main and interrelated problems with descriptions of IBMs: (1) there is no standard protocol for describing them and (2) IBMs are often described verbally without a clear indication of the equations, rules, and schedules that are used in the model.

A standard protocol for the description of IBMs would make reading and understanding them easier because readers would be guided by their expectations. Gopen and Swan (1990) explain how understanding is facilitated when writers take readers' expectations into account: readers are better able to absorb information if it is provided in a familiar, meaningful structure. For example, when we read a sentence we expect context at the beginning and the point to be stressed at the end. Likewise, when we read a paper describing an analytical model, we expect to see several equations and definitions of the variables, then a table of parameter values. But when we start reading an IBM-based paper we start without the expectation of a familiar structure. As a consequence, we have to read the entire model description in every detail, even if at first we only want to have a general idea of the model's purpose, structure and processes. This makes reading many IBM descriptions cumbersome and inefficient. Moreover, if we want to critically assess a model or re-implement it, wholly or in part, we must tediously transform the verbal model description into explicit equations, rules, and schedules before implementing it in our own program. Even the many clear and useful IBM descriptions that certainly exist do not entirely solve the communication problem because different authors use different protocols. Thus, a priori, we do not know where in the model description we should expect to find particular information.

Lengthy verbal descriptions are the second reason why many IBM descriptions are so cumbersome. Often we find a mixture of general considerations, verbal descriptions of processes, and lengthy justifications of the specific model formulations chosen. All this makes it hard to extract the information relevant for understanding and implementing the model. But this need not be. Three very successful IBMs, which have been re-used and modified in numerous follow-up models, describe their basic model processes in equations: the JABOWA forest model of Botkin et al. (1972) and Shugart (1984), which gave rise to a full pedigree of so-called "gap models" (Liu and Ashton, 1995); the fish cohort model of DeAngelis et al. (1980), which initiated large research projects using IBMs (Tyler and Rose, 1994; Van Winkle et al., 1993); and the fish school model of Huth and Wissel (1992, 1994), which was independently re-implemented and modified several times (e.g., Inada and Kawachi, 2002; Kunz and Hemelrijk, 2003; Reuter and Breckling, 1994). The success of these three models seems to a large degree to be due to the fact that their extensive use of the language of mathematics allowed them to be easily reproduced.

We conclude that what we badly need is a standard protocol for describing IBMs which combines two elements: (1) a general structure for describing IBMs, thereby making a model's description independent of its specific structure, purpose and form of implementation (Grimm, 2002) and (2) the language of mathematics, thereby clearly separating verbal considerations from a mathematical description of the equations, rules, and schedules that constitute the model. Such a protocol could, once widely used, guide both readers and writers of IBMs.

In this article we propose a standard protocol for describing IBMs (including agent-based models, multi-agent simulation, or multi-agent systems; see Discussion). The basic idea of the protocol was proposed by Grimm and Railsback (2005) and then discussed during an international workshop on individual-based modelling held in Bergen, Norway, in the spring of 2004. Most participants of that workshop are among the authors of this article, leading to 28 authors from seven different countries. The work of the authors covers a wide range of fields within ecology (e.g., marine, terrestrial, plant, animal, behaviour, population, forest, theory, conservation, etc.), and

Overview	Purpose
	State variables and scales
	Process overview and scheduling
Design concepts	Design concepts
Details	Initialization
	Input
	Submodels

Fig. 1 – The seven elements of the ODD protocol, which can be grouped into the three blocks: Overview, Design concepts, and Details.

the authors have altogether been involved in the writing of more than 200 IBM-based papers.

We agreed to test and refine the standard protocol proposed by Grimm and Railsback (2005) by applying it to our own models: every author, or team of co-authors, rewrote one of their existing model descriptions using the new standard protocol. The set of 19 models used in this test differs widely in scope, structure, complexity, and implementation details (see Online Appendix). As a result of the test applications, the protocol was slightly revised.

Here, we first present the standard protocol, which Grimm and Railsback (2005) refer to as the PSPC + 3 protocol. The abbreviation "PSPC" referred to the initials of first four elements of the protocol (purpose, structure, process, concepts) and "+3" referred to the remaining three elements. In the revised protocol, however, the names of some elements have been changed. We are therefore using a new acronym, "ODD", which stands for the three blocks of elements 'Overview', 'Design concepts', and 'Details' (Fig. 1).

Then we present an example application of the protocol, and summarize our experience with test applications in a list of frequently asked questions which provides practical hints for using the protocol. Finally we discuss both our experience with the test applications and ODD's potentials and limitations and how it could contribute to further unification of the formulation and implementation of IBMs.

2. The ODD protocol

The basic idea of the protocol is always to structure the information about an IBM in the same sequence (Fig. 1). This sequence consists of seven elements that can be grouped in three blocks: Overview, Design concepts, and Details (as a mnemonic, this sequence can be referred to as the ODD sequence). The overview consists of three elements (purpose, State variables and scales, process overview and scheduling), which provide an overview of the overall purpose and structure of the model. Readers very quickly can get an idea of the model's focus, resolution and complexity. After reading the overview it should be possible to write, in an object-oriented programming language, the skeleton of a program that implements the IBM described. This skeleton includes the declaration of all objects (classes) describing the models entities (different types of individuals or environments) and the scheduling of the model's processes.

The block or element "Design concepts" does not describe the model itself, but rather describes the general concepts underlying the design of the model. The purpose of this element of the protocol is to link model design to general concepts identified in the field of Complex Adaptive Systems (Grimm and Railsback, 2005; Railsback, 2001). These concepts include questions about emergence, the type interactions among individuals, whether individuals consider predictions about future conditions, or why and how stochasticity is considered. By referring to such general design concepts, each individual-based and agent-based model is integrated into the larger framework of the science of Complex Adaptive Systems.

The third part of ODD, Details, includes three elements (initialization, input, submodels) that present the details that were omitted in the overview. In particular, the submodels implementing the model's processes are described in detail. All information required to completely re-implement the model and run the baseline simulations should be provided here. If space in a journal article is too limited, Online Appendices or separate publications of the model's details should be provided.

The logic behind the ODD sequence is: context and general information is provided first (Overview), followed by more strategic considerations (Design concepts), and finally more technical details (Details). We can help readers understand our IBMs by always using this structure: a standard protocol that provides the information in an order that allows the reader to easily build on their previous understanding. Below, the seven elements of ODD are described. A template document of the ODD protocol is provided in the Online Appendix.

2.1. *Purpose*

The purpose of a model has to be stated first because without knowing it, readers cannot understand why some aspects of reality are included while others are ignored. Usually, the context and purpose of a model are provided in the introduction of an article, but it is nevertheless important to have a clear, concise and specific formulation of the model's purpose because it provides a guide for what to expect in the model description that follows. Thus, this element informs about why you need to build a complex model, and what, in general and in particular, you are going to do with your model.

2.2. *State variables and scales*

What is the structure of the model system? For example, what kind of low-level entities (e.g., individuals, habitat units) are described in the model? How are they described? What hierarchical levels exist? How are the abiotic and biotic environments described? What is the temporal and spatial resolution and extent of the model system?

First, the full set of state variables should be described. The term 'state variables' refers to low-level variables that characterize the low-level entities of the model, i.e. individuals or habitat units. For example, individuals might be characterized by a number of characteristics: age, sex, social rank, location, parents; habitat units might be characterized by location, soil type, predation risk (for a certain species), percentage cover.

It is important not to confuse low-level state variables with auxiliary, or aggregated, variables, such as population size or average food density in a given area. Auxiliary variables contain information that is deduced from low-level entities and their low-level state variables. Population size, for example, is simply the number of individuals; age structure is a histogram taken from the age of all individuals; average food density is the average of the amount of food in every habitat unit in a given region. In contrast, low-level state variables cannot be deduced from other low-level state variables, because they are elementary properties of model entities. Age, sex and location, for example, cannot be deduced from any other variable but are elementary properties of an individual. In other words, auxiliary variables aggregate information from model entities, whereas low-level state variables describe elementary properties of the model's entities.

If the set of (low-level) state variables is large, as is the case with many IBMs, it should preferably be presented in a table in which the variables are grouped according to the entities represented in the model (e.g., individuals, habitat units, abiotic environment). Another option is to use class diagrams of the Unified Modeling Language (UML; Fowler, 2003). Once readers know the full set of (low-level) state variables, they have a clear idea of the model's structure and resolution, such as the level of detail the individuals are described with. It is daunting to find how difficult it is to extract the full set of state variables from many existing IBM descriptions.

Second, the higher-level entities should be described: for example a population consisting of individuals, a community consisting of populations, or a landscape consisting of habitat units.

Finally, in addition to the state variables, the scales addressed by the model should be stated, i.e. length of time steps and time horizon, size of habitat cells (if the model is grid-based), and extent of the model world (if the model is spatially explicit). The reason why these scales have been selected should briefly be explained, because choosing the scale is a fundamental decision determining the design of the entire model. The dimensions must be clearly defined for all parameters and variables in the tables, to avoid confusion and inconsistencies and allow model reproduction. With spatially explicit models that include spatial heterogeneity, a figure representing the model area in a typical configuration can be useful.

2.3. *Process overview and scheduling*

To understand an IBM, we must know which environmental and individual processes are built into the model; examples are food production, feeding, growth, movement, mortality, reproduction, disturbance events, and management. At this stage, a verbal, conceptual description of each process and its effects is sufficient because the main purpose of this element of ODD is to give a concise overview. If the number of processes included in the model is large, a table listing the processes might be useful.

In addition, the scheduling of the model processes should be described. This deals with the order of the processes and, in turn, the order in which the state variables are updated. More specific questions include: How is time modelled in the IBM—using discrete time steps, continuous time, or both? Is dynamic scheduling used for events that happen quickly compared to the model's time step and are highly dependent on execution order (Grimm and Railsback, 2005)? What model processes or events are grouped into actions that are executed together? Do these actions produce synchronous or asynchronous updating of the state variables? How are actions that actually happen concurrently in nature executed in the model? What actions are on a fixed schedule, and in what order? Are some actions executed in random order? What is the basis for these scheduling decisions?

In many cases it will be convenient to visualize scheduling by using flow charts. Freeware software is available for producing flow charts, and some accepted conventions of drawing flow charts should be followed. Flow charts must, however, correspond literally to the flow of processes in the model, otherwise they make it virtually impossible to re-implement the model. In fact, for dynamic scheduling (e.g., Zeigler et al., 2000) flow charts might actually hinder understanding; pseudo-code describing the structure of the simulation program is an alternative (see, for example, Pitt et al., 2003).

2.4. *Design concepts*

The design concepts provide a common framework for designing and communicating IBMs. They are explained in more detail in Grimm and Railsback (2005) and in the Appendix "Design concepts" in the Online Archive; this Appendix also includes a more detailed checklist of questions regarding design concepts. Here we only provide a short checklist which should be followed when describing (and designing) an IBM. Those items of the checklist that do not apply should simply be left out in the model description; an example would be if the model includes no collective agents, such as a herd or family group. The sequence of the checklist items – in contrast to the seven elements of ODD – is not meant to be compulsory but may be shuffled if considered necessary.

Emergence: Which system-level phenomena truly emerge from individual traits, and which phenomena are merely imposed?

Adaptation: What adaptive traits do the model individuals have which directly or indirectly can improve their potential fitness, in response to changes in themselves or their environment?

Fitness: Is fitness-seeking modelled explicitly or implicitly? If explicitly, how do individuals calculate fitness (i.e., what is their fitness measure)? In agent-based models that do not address animals or plants, instead of fitness other "objectives" of the agents should be considered here (e.g. economic revenue, pollution control).

Prediction: In estimating future consequences of their decisions, how do individuals predict the future conditions they will experience?

Sensing: What internal and environmental state variables are individuals assumed to sense or "know" and consider in their adaptive decisions?
Interaction: What kinds of interactions among individuals are assumed?
Stochasticity: Is stochasticicity part of the model? What are the reasons?
Collectives: Are individuals grouped into some kind of collective, e.g. a social group?
Observation: How are data collected from the IBM for testing, understanding, and analyzing it?

2.5. *Initialization*

This deals with such questions as: How are the environment and the individuals created at the start of a simulation run, i.e. what are the initial values of the state variables? Is initialization always the same, or was it varied among simulations? Were the initial values chosen arbitrarily or based on data? References to those data should be provided. Communicating how IBMs are initialized can be important if peers want to re-implement the IBM and reproduce the simulation experiments reported.

2.6. *Input*

The dynamics of many IBMs are driven by some environmental conditions which change over space and time. A typical example is precipitation, which may vary over time (seasons, years) and space (different spatial patterns of rainfall in different regions), and management, e.g. harvesting regimes (management might also be addressed in the section "simulation experiments", which usually will follow the model description). All these environmental conditions are "input", i.e. imposed dynamics of certain state variables. The model output gives the response of the model to the input. Readers need to know what input data are used, how they were generated and how they can be generated or obtained. To really achieve full reproducibility it might be necessary to provide (in online archives) the input files that you used yourself, including even the random number used as seed.

2.7. *Submodels*

Here, all submodels representing the processes listed above in "Process overview and scales" are presented and explained in detail, including the parameterization of the model. But, given the space limitations of journals, how can we make the detailed model description easy to understand, easy to use for re-implementing the model, and nevertheless complete? The answer partly depends on the complexity of the model, but in general we propose that two versions of the detailed model description be written:

1. The mathematical "skeleton" of the model. This skeleton consists of the model equations and rules and one or more tables presenting the model parameters and their dimensions. Verbal explanations of the equations and rules should be kept to a minimum: parameters have of course to be explained, but longer explanations of why this specific model formulation was chosen, how the parameters were determined, etc., do not belong here. If the list of equations and rules is too long, it should be presented in an Online Appendix.
2. A full model description. This version has exactly the same structure as the "skeleton" (i.e., the same subtitles and equation numbers), but now each equation and parameter is verbally explained in full detail and deals with questions such as: What specific assumptions are underlying the equations and rules? How were parameter values chosen? How were submodels tested and calibrated? Ideally, the two versions of the detailed model description could be presented in the same document, with the more detailed verbal descriptions hidden to readers in version one but visible in version two. (This technique is partly used in the HTML model description of Deutschman et al. (1997) where readers can chose links providing more detailed information.)

For most IBMs, the second version will be too long to be included in a journal paper. Grimm and Railsback (2005) suggest two solutions to this problem. One is to use the online or electronic archives of the journal; an increasing number of journals are providing online archives. The other is to publish the full model description (version two) in an extra paper or a technical report which is accessible via the Internet.

3. Sample application of ODD

Here we present a sample application of ODD to an individual-based population model of the alpine marmot, *Marmota marmota* (Grimm et al., 2003; Dorndorf, 1999). For reasons of space limitations, we here chose a relatively simple model that describes many processes empirically by using probabilities, for example 'mortality'. The Online Appendix contains examples of much more complex models that represent many processes mechanistically. The following example is a revised version of a model description given in Grimm et al. (2003).

3.1. *Purpose*

The purpose of the model is to understand how the social behaviour of the marmots – in particular territoriality, reproductive suppression, and hibernation as a group – affects population dynamics and in particular extinction risk if populations are small.

3.2. *State variables and scales*

The model comprises four hierarchical levels: individual, territory, (meta)population, and environment. Individuals are characterized by the state variables: identity number, age, sex, identity of the territory where the individual lives, and social rank. Newborns have the additional state variable weaning weight, which affects their mortality. Individuals which have not completed their first winter are referred to as juveniles; 1-year-olds as yearlings, and all others as adults. Apart from this, social rank is the main attribute which tells the difference between dominant and subdominant adults (Table 1).

Table 1 – Overview of processes, parameters and default values of parameters of the marmot model

Parameter	Value
Number of territories	22
Age of sexual maturity (years)	2
Winter mortality	
Mean of the winter strength distribution (days)	117
Standard deviation of the winter strength distribution (days)	10.2
Mean of the territory quality distribution (days)	0
Standard deviation of territory quality distribution (days)	8.4
Mean of the weaning date distribution (days)	185.5
Standard deviation of the weaning date distribution (days)	6.6
Winter mortality of floaters	0.9
Recolonization	
Dispersal probability at age 2	0.2
Dispersal probability at age 3	0.7
Dispersal probability at age 4	0.5
Dispersal probability at age 5	1
Probability to inherit a vacant dominant position at home	0.2
Probability to occupy a vacant dominant position in the neighbourhood	0.3
Probability to occupy a vacant dominant position further away than 500 m	0.5
Eviction	
Eviction probability of dominant animal	0.15
Reproduction	
Reproduction probability of a dominant female	0.64
Mean of the litter size distribution	3.3
Standard deviation of the litter size distribution	1.43
Sex ratio in a litter	0.58
Summer mortality	
Summer survival of juveniles	0.9
Summer survival of yearlings	0.94

If not otherwise specified, these default values are used (from Grimm et al., 2003, after Dorndorf, 1999). Dimensionless parameters are either numbers or probabilities; for parameter tables with a stronger focus on the dimensions of the parameters, see examples in the Online Appendix.

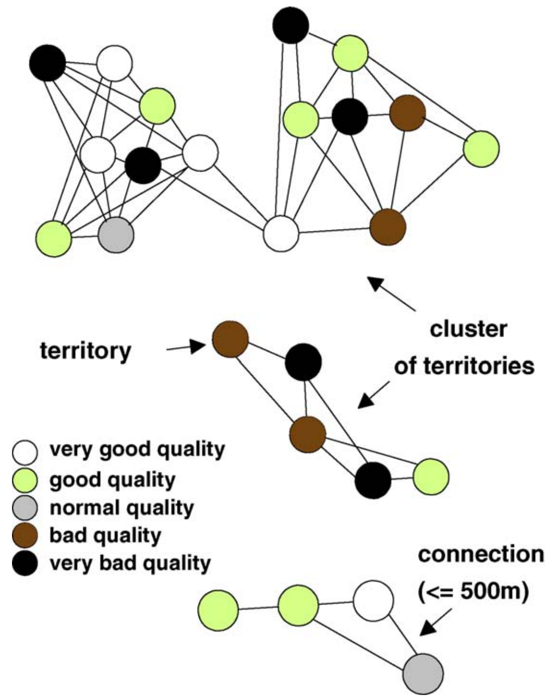


Fig. 2 – Spatial arrangement of territories in the model. Territories which are closer than 500 m to each other are linked by lines, indicating the chance of subdominants recolonizing vacant dominant positions within this neighbourhood without undertaking long-distance dispersal. The different grey scales of the territories indicate different habitat qualities of the territories (from Grimm et al., 2003, after Dorndorf, 1999).

A territory may be occupied by a social group of marmots and contains one hibernaculum used by this group during winter. A territory is characterized by the state variables: identity number, the number and list of individuals present, and its quality. If the number of individuals is zero, the territory is referred to as 'empty', i.e. space which has become vacant due to the extinction of a social group. Thus, territories may be recolonized just like empty patches in metapopulations. 'Quality' is an attribute characterizing habitat heterogeneity with respect to the harshness of overwintering conditions, indicated by the date in spring when a territory becomes snow-free.

The population is composed of several territories or social groups, respectively. Populations are characterized by size, the number of social groups, and the number and list of territories. In addition, a "floater pool" keeps track of both all subdominants which have left their home territory and dominants which have been evicted. The spatial structure is taken into account by specifying the linkages to neighbouring territories. A neighbouring territory is defined as a territory within the distance of 500 m. The number of linkages may vary between zero and six (Fig. 2). Clusters of neighbouring territories compose a local metapopulation. Several clusters make up the regional metapopulation of the alpine marmot (Fig. 2). As distances between clusters are greater than 500 m, only dispersing subdominants will cross this distance. On this spatial scale beyond 500 m the model is not spatially explicit but the dispersers may reach any cluster of territories within the model area. This restricts the extent of the area that can be described by the model to several square kilometres.

The highest hierarchical level in the model is the abiotic environment and its fluctuations. Since the severity of winter, indicated by the date when territories become snow-free, is the most important aspect in the life of marmots, the abiotic environment in the model is characterized by this date. The date when a territory becomes snow-free is referred to as 'winter strength'; it is drawn from a normal distribution and modulated by the quality of the territories.

3.3. *Process overview and scheduling*

The model proceeds in annual time steps. Within each year or time step, seven modules or phases are processed in the following order: winter mortality, eviction, inheritance, dispersal,

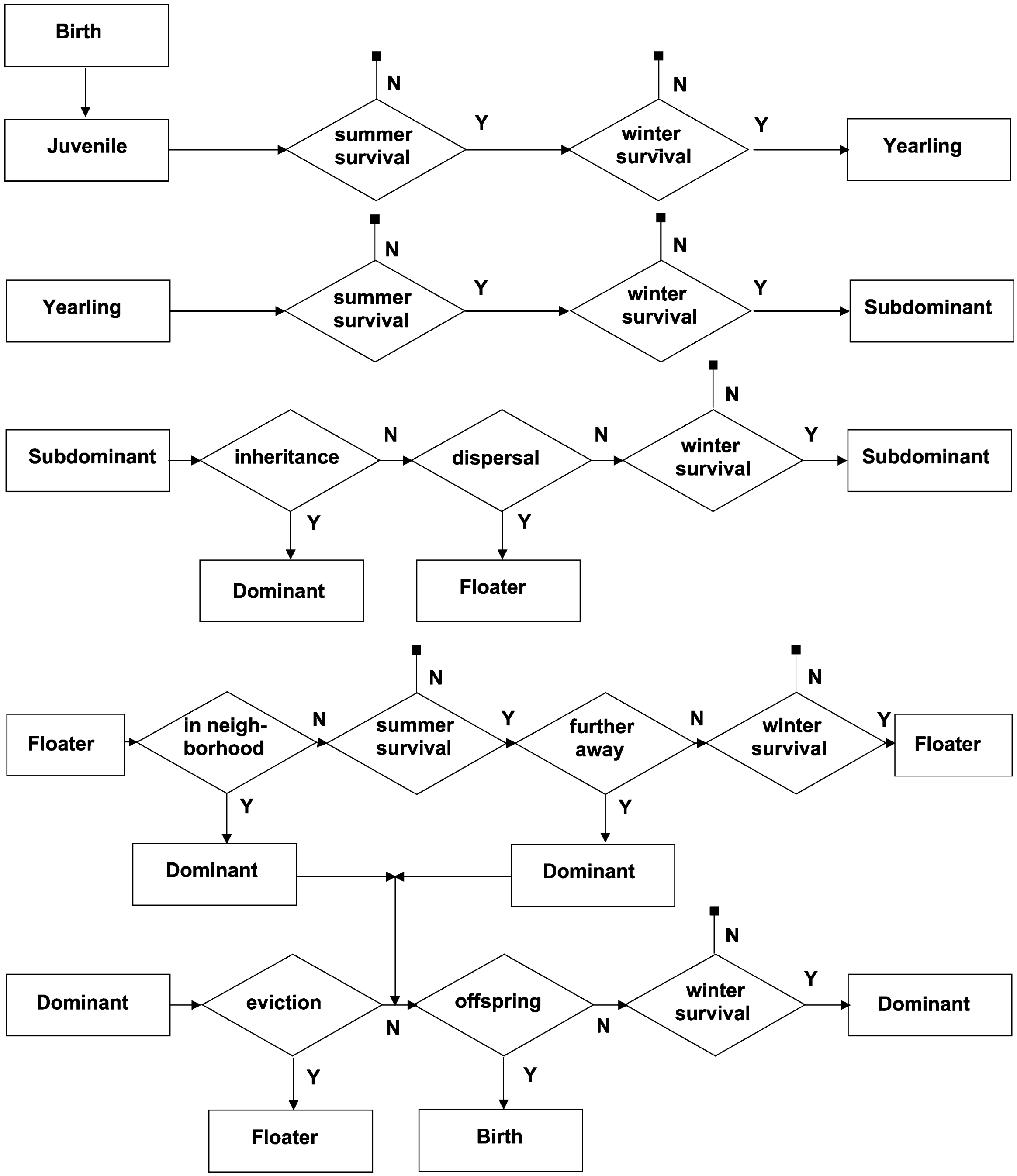


Fig. 3 – Life history of the model marmots showing the transitions between different age and social classes, as well as the processes which cause these transitions (from Grimm et al., 2003, after Dorndorf, 1999).

re-colonization of vacant dominant positions, reproduction, and summer mortality. Within each module, individuals and territories are processed in a random order. The individuals life cycle is depicted in Fig. 3.

3.4. *Design concepts*

Emergence: Population dynamics emerge from the behaviour of the individuals, but the individual's life cycle and behaviour are entirely represented by empirical rules describing, for example, mortality and dispersal rates as probabilities. Adaptation and fitness-seeking are thus not modelled explicitly, but are included in the empirical rules.
Sensing: Individuals are assumed to know their own sex, age, and social rank so that they apply, for example, their age-specific dispersal probabilities.
Interaction: Three types of interactions are modelled implicitly: winter mortality decreases with group size, alpha individuals suppress reproduction of subdominants, and after changes in the alpha male position in the current year,

the alpha female does not reproduce. One interaction is modelled explicitly: subdominants and individuals from the floater pool can try to evict alpha individuals.

Stochasticity: All demographic and behavioural parameters are interpreted as probabilities, or are drawn from empirical probability distributions. This was done to include demographic noise and because the focus of the model is on population-level phenomena, not on individual behaviour. Winter strength was taken from a truncated normal distribution in order to include environmental noise (i.e., variation of the population's growth rate driven by fluctuations of abiotic conditions). Likewise, habitat quality was taken from a truncated normal distribution in order to include spatial heterogeneity.

Observation: For model testing, the spatial distribution of the individuals was observed process by process. For model analysis, only population-level variables were recorded, i.e. group size distribution, population size over time, and time to extinction (using the "$\ln(1-P_0)$ plot" of Grimm and Wissel, 2004).

3.5. Initialization

Each territory was initially occupied with a 5-year-old couple of dominants and both a 1-year-old male and female subdominant. The evaluation of each simulation run started in the first year when the number of model adults was equal to the number of adults observed in the first year of the field study.

3.6. Input

In general model analysis, each year winter strength is drawn from a normal distribution with an empirically determined mean and standard deviation (mean = 117 days of the year for territories in the study area, s = 10.2 days). This overall winter strength is modified by differences in overwintering conditions among territories, i.e., from a normal distribution with a mean of zero and a standard deviation of 8.4 days. This means that territories which have a higher quality than the mean become snow-free a certain number of days earlier than specified by the overall winter strength, whereas territories of lower quality become snow-free later.

3.7. Submodels

Winter mortality: For dominant marmots, winter mortality – interpreted as the probability of dying in a certain winter – is determined from the long-term data set by logistic regression:

$$P_{\text{ter}} = [1 + \exp(6.82 - 0.286A - 0.028\text{WS} + 0.395\text{SUBY})]^{-1} \quad (1)$$

where A is the age, WS winter strength, and SUBY is the number of subdominants (including yearlings) present in a group. Eq. (1) states that the winter mortality of dominants increases with the severity of overwintering conditions and with age, but decreases with the number of subdominants and yearlings.

Similarly, winter mortality for subdominants (including yearlings) is:

$$P_{\text{sub}} = [1 + \exp(7.545 - 0.038\text{WS})]^{-1} \quad (2)$$

For juveniles, we found in addition a significant influence of weaning weight on winter mortality:

$$P_{\text{new}} = [1 + \exp(-1.014 - 0.024\text{WS} + 0.008\text{WW} + 0.613\text{SUB})]^{-1} \quad (3)$$

with WW being the weaning weight (see below, *Reproduction*) and SUB the number of subdominants (excluding yearlings). Thus, the place where sociality comes into play in our model is in Eqs. (1) and (3) via the variable SUB.

Two additional model rules take into account further processes affecting mortality. Firstly, in groups without subdominants and yearlings, the dominant couple had a higher risk of mortality than specified by Eq. (1). Whether the first dominant marmot (which is chosen randomly) dies or survives is determined according to the mortality specified in Eq. (1). If it dies its partner has an increased probability of dying of $P' = 0.66$. If this partner dies as well, the newborns – if present – will also die in turn. To avoid that this rule introduces a higher total mortality than specified in Eq. (1), for the case that the first partner survives the mortality of the second partner had to be modified (see Online Appendix).

The second model modification concerning winter mortality introduces the probability P_C, which takes into account the extinction of entire social groups due to local catastrophes during winter. We use a value of $P_C = 0.004$. Finally, we assume a winter mortality of the floaters which failed to take over a new territory during the summer as $P_{\text{floatwinter}} = 0.9$.

Eviction: Dominant positions may become vacant not only due to winter mortality but also because the existing dominant animal has been evicted by a subdominant group member or a floater. We assume that dominant individuals are evicted with a probability of $P_{EV} = 0.15$ and that all evicted animals enter the floater pool.

The following three modules of the model describe how dominant positions which became vacant due to winter mortality and eviction are reoccupied by subdominants or dispersers.

Inheritance: The oldest subdominant animal has a probability of $P_{IN} = 0.22$ of taking the dominant position. If this animal fails or if there is no subdominant in the territory, the dominant position remains vacant and can be taken over by a floater (see below).

Dispersal: Most of the subdominants willing to disperse leave their home territory in spring. The probability of leaving depends on age and is directly taken from Table 2. Dispersed animals are compiled in a list called the "floater pool". This list is used to handle the assignment of free dominant positions to floaters. Note that the floater pool contains both true floaters which disperse beyond 500 m and are subject to dispersal mortality during summer, and animals which will take over a dominant position in the neighbourhood.

Recolonization: In the model, recolonization is implemented by the following suite of rules. The first rule decides with a probability of $R_N = 0.5$ whether a vacant dominant position is reoccupied by a marmot that comes from a neighbouring territory. If this is the case, the floater pool is searched (in a random order) for such an animal, and if no animal is found the dominant position remains vacant. After repeating this procedure for each vacant dominant position, the remaining animals in the floater pool are treated as true floaters and have a dispersal mortality of $P_D = 0.3$, i.e. about 30% of the remaining floaters die before the next model rules are applied.

The next rule is analogous to the first rule, but this time each of remaining true floaters is allowed to occupy an available dominant position with a probability of $R_F = 0.5$. Finally, the last rule of this module checks territories where the dominant positions are still unoccupied for the presence of sexually mature animals. If one is found, the oldest subdominant animal moves into the dominant position.

Reproduction: Only when a dominant male and female are present in a territory reproduction can take place. The probability of a dominant female having offspring is 0.64 (Hackländer and Arnold, 1999). The mean litter size (L) is 3.3 and standard deviation is 1.43. The mean weaning weight (WW_{mean}) is 536 g (S.D. = 126.3 g) but decreases with litter size. Therefore a regression model is used to assess a mean weaning weight depending on litter size L ($WW_{mean} = 680.23 - 35.24L$, $R^2 = 0.143$, $P < 0.001$). In the model, litter size and weaning weight are drawn from normal distributions (in the case of litter size, discretized and truncated to the interval [1,6]) with the means and standard deviations specified. The sex of offspring is determined by chance with a bias of 0.58 towards males. We assume that no reproduction occurs if the holder of a male dominant position has changed during the current year.

Summer mortality: Summer mortality rates are only known from the field for juveniles and yearlings. Summer mortality of resident adults is low but hard to quantify. The summer mortality of adults is thus indirectly and implicitly taken into account in the probabilities of eviction and dispersal mortality. Newborns and yearlings die during summer with a probability of 0.11 and 0.07, respectively.

4. Practical hints for using ODD

During the test of the protocol, several questions arose that are not answered by the description of the protocol itself. The following list of questions is thus organized in the style of "Frequently Asked Questions" (FAQ). We plan to maintain this list on a webpage devoted to ODD. The evolving FAQ could be the basis of future developments of the protocol.

4.1. *Are scenarios, simulation experiments, and sensitivity analysis part of the protocol?*

No. The protocol is designed to describe the basic model. It corresponds to the "Materials" part of an article presenting empirical work. We recommend including a section entitled "simulation experiments" following the description of the model. This section would correspond to the classical "Methods" part of research articles. Simulation models are experimental systems (Peck, 2004), and scenarios, sensitivity or uncertainty analysis, etc. are all just that: simulation experiments that are carefully designed to test a certain hypothesis. This hypothesis or purpose of the experiment should clearly be stated.

4.2. *Should the elements of ODD always be presented in the given sequence?*

Yes, definitely. This is the main idea of the protocol: first providing a comprehensive overview; then explaining the design concepts underlying the model, and finally presenting all details that are necessary to fully understand and – in principle – re-implement the model. The sequence of the design concepts, however, may be changed, if considered necessary.

4.3. *Where do I describe parameterization and tests of the submodels?*

In the element "Submodels". If parameterization was not very complex, it might be sufficient to present the source of the parameters in the table listing the parameters. If parameterization was a major issue, it might be best to describe it briefly in the article and give details in an Online Appendix. The same applies to tests of the submodels, e.g. comparing them to independent implementations using, for example, spreadsheets (Grimm and Railsback, 2005).

4.4. *What about the source code and the executable program?*

Even the most carefully prepared verbal model description is likely to contain a few ambiguities that make it difficult, or even impossible, to independently re-implement the model (Edmonds and Hales, 2003; Rouchier, 2003). We therefore recommend that the source code, or parts of it, be provided in an Online Archive. So far this has not been done very often, partly because authors might want to keep their code proprietary, partly because there are so many different programming languages, compilers, software platforms, and operation systems that usually only a minority of readers will be able to fully understand the code or even run it on their own computers. It should, however, be possible to communicate how the three elementary parts of a model have been coded: the declaration of the model's entitities, the scheduling of processes, and the very rules and equations that have been used to represent the processes. Even if we, for example, do not understand Java, it should be possible to check in a Java program how the three elementary parts of the model have been implemented. The minimum requirements for this would be: comments that identify the three elementary parts, the meaning of the program variables, and the purpose of methods, functions, and procedures.

In addition, it would be good practice to provide an executable version of the program that is capable of performing all or the most important simulation experiments that are described in the article. All initialization, input, and output files that are required to run the program should be included. For a detailed discussion of the costs and benefits of providing the executable program, see Grimm (2002).

4.5. Why not use the Unified Modeling Language (UML)?

UML is indeed a powerful tool to describe object-oriented software in a unifying format (Fowler, 2003). However, the full UML is quite complex and includes numerous types of diagrams that are not at all easy to develop or understand. UML was designed and is developed by professional software engineers. The purpose of ODD, however, was that it can easily be written and understood by ecologists, who usually are not software engineers. Ultimately, something similar to UML should be developed for individual-based and agent-based models: a visual declarative language that is easy to use and can directly be compiled to computer code (tools for translating code to UML and vice versa exist). We recommend reading introductory texts of UML and using the most basic and simple type of diagram, the "class diagram" (see examples in Online Appendix).

4.6. How to deal with different journal formats?

Journals have different format requirements for headlines, number of headline levels, etc. We recommend trying to use the elements labels ("Purpose", "State Variables and Scales") as headlines, because this provides a clear visual guide to readers. If journals are particular about headlines, the elements names should be highlighted by other means.

4.7. In models including human agents, where do we describe memory and behavioural strategies?

Anything that is used to distinguish individuals is considered a low-level state variable. Memory clearly is represented by such variables. A behavioural strategy is not part of the individual's state if all individuals use the same strategy. If individuals can have different, but fixed strategies, then a variable indicating the strategy used by an individual would be a state variable, and the set of strategies would be submodels. If behavioural strategies vary continuously, then the variables and parameters specifying the behaviour of an individual are the state variables characterizing behaviour.

4.8. I find it difficult to clearly describe "scheduling"

Of all elements of a model description, "scheduling" is the least developed one and, in fact, is simply left out in many descriptions. Verbal descriptions are usually not sufficient to describe the ordering of processes in a model. Flow charts certainly are useful and easy to grasp, but for any scheduling deviating from a linear sequence of processes, pseudo code that exactly corresponds to the code used for simulations should be provided (plus the code itself).

5. Discussion

Regarding the communication and development of individual-based or agent-based models, the current situation is poignantly described by Hales et al. (2003): "Researchers tend to work in isolation, designing all their models from scratch and reporting their results without anyone else reproducing what they found." (Section 1.2). Reproducing results, however, is a *conditio sine qua non* for making simulation models a more rigorous tool for science: "Since almost all simulations are not amenable to formal analysis, the only way they can be verified is via the experimentation of running simulations. If we are to be able to trust the simulations we use, we must independently replicate them." (Edmonds and Hales, 2003, Section 12.2). A similar point is made by Aber (1997).

The ODD protocol is designed as a tool to facilitate the communication and replication of IBMs and agent-based models (ABMs). We consider the protocol as a first step for establishing a more detailed common format of the description of IBMs and ABMs. The test applications of ODD presented in the Online Appendix show that it does not immediately solve all problems of communicating IBMs or ABMs, but is a step in the right direction.

Originally, we expected that the protocol as proposed by Grimm and Railsback (2005) would make the test model descriptions (Online Appendix) quite similar, but this was less so than expected for two reasons. First, the original formulation of the protocol used a terminology, for example "state variables", that was not explicitly defined and therefore variously interpreted in the test applications. We tried to remove this terminological ambiguity in the revised formulation of the protocol. Second, the test situation was somewhat unnatural: existing descriptions of sometimes very complex models were rearranged and slightly revised, but not newly written from scratch. However, we expect that model descriptions will be more homogeneous if written anew, following the protocol presented above.

Still, as can be seen from the example above and in the Online Appendix, differences in the style of the presentation are likely to remain. We have to accept this at the current stage, because the protocol has to compromise between being general enough to include all kinds of individual-based or agent-based models and being specific enough to fulfil its purpose. In particular, the protocol is not specific enough to "force" a more strict use of the language of mathematics. It is, however, a good exercise to take an existing model description, which usually is a mixture of rules, equations, and lengthy explanation, and to keep only the factual description of the model and leave out all motivations, explanations, and justifications.

Besides the current limitations of ODD, however, also the benefits of the protocol became obvious in the test applications. The most important benefits were:

- The model description became easier to write. It was no longer necessary to waste a lot of time thinking about how to structure the text, because the protocol had made those decisions for the authors so that they simply could follow the template.
- The model description became more complete because the protocol reminded the authors of important details that they might have otherwise forgotten to include in the documentation.
- The model description became easier to understand. In one case, for example, the protocol suggested a context for describing a concept that had been confusing to the reviewers of the original paper (emergence). If ODD had been used

before, the review process would have been smoother and the final description would have been clearer.

- The protocol is not only useful for individual-based or agent-based models, but for bottom-up simulation models in general, for example grid-based models. Two of the test applications (“Biological control”, “Rangeland management”) are not individual-based; here, those design concepts that did not apply were simply ignored.

Once ODD is used by a sufficiently large proportion of modellers, the next step would perhaps be to develop more specific formats for the seven elements of the protocol. For example, UML class diagrams could become standard for giving an overview of state variables and processes; a certain format of pseudo-code describing process scheduling could be developed; a certain style for representing model rules could be established; or we could even identify a limited set of “behavioural primitives” (Ginot et al., 2002) that might be modelled in alternative but compatible ways.

If ODD develops as we envision it, we might after, say, 5–10 years come to the point where the following vision of the IBM developers “software heaven” becomes reality: “modelers could describe their IBM on paper using some kind of language that (1) people can understand intuitively, (2) is widely used throughout ecology, (3) provides ‘shorthand’ conventions that minimize the effort to describe the IBM rigorously and completely, and (4) can be converted directly into an executable simulator without the possibility of programming errors. After converting the model description into an executable simulator, the modelers then could turn the simulator into a *simulation laboratory* by attaching experimentation tools: probes to collect data; displays to show results visually; controls that automatically generate, execute, and interpret . . . analysis experiments” (Grimm and Railsback, 2005, p. 271).

We are planning to maintain an ODD webpage (which will be accessible via http://www.ufz.de/oesatools/odd), to regularly evaluate the usage of the protocol, to collect questions and suggestions of users, and to publish new “releases” of the protocol, which should, however, be compatible with earlier releases.

Acknowledgements

We would like to thank two anonymous reviewers for helpful comments.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.ecolmodel.2006.04.023.

REFERENCES

Aber, J.D., 1997. Why don’t we believe the models? Bull. Ecol. Soc. Am. 78, 232–233.

Axelrod, R., 1997. The Complexity of Cooperation: Agent-based Models of Competition and Collaboration. Princeton University Press, Princeton, NJ.

Billari, F., Prskawetz, A., 2003. Agent-Based Computational Demography: Using Simulation to Improve Our Understanding of Demographic Behaviour. Springer Physica-Verlag, Heidelberg.

Botkin, D.B., Janak, J.F., Wallis, J.R., 1972. Some ecological consequences of a computer model of forest growth. J. Ecol. 60, 849–873.

DeAngelis, D.L., Cox, D.K., Coutant, C.C., 1980. Cannibalism and size dispersal in young-of-the-year largemouth bass: experiment and model. Ecol. Model. 8, 133–148.

DeAngelis, D.L., Gross, L.J., 1992. Individual-Based Models and Approaches in Ecology. Chapman and Hall, New York.

DeAngelis, D.L., Mooij, W.M., 2005. Individual-based modeling of ecological and evolutionary processes. Annu. Rev. Ecol. Evol. Syst. 36, 147–168.

Deutschman, D.H., Levin, S.A., Devine, C., Buttel, L., 1997. Scaling from trees to forests: analysis of a complex simulation model. Science 277, 1688, Available at: http://www.sciencemag.org/feature/data/deutschman/index.htm.

Dorndorf, N., 1999. Zur Populationsdynamik des Alpenmurmeltiers: Modellierung, Gefährdungsanalyse und Bedeutung des Sozialverhaltens für die Überlebensfähigkeit. Ph.D. Thesis. Philipps-Universität Marburg, Germany.

Edmonds, B., Hales, D., 2003. Replication, replication, and replication: some hard lessons from model alignment. J. Artif. Soc. Soc. Simul. 6, http://jasss.soc.surrey.ac.uk/6-4/11.html.

Epstein, J., Axtell, R., 1996. Growing Artificial Societies. Social Science from the Bottom Up. Brookins Institution Press/The MIT Press.

Ford, E.D., 2000. Scientific Method for Ecological Research. Cambridge University Press, Cambridge.

Fowler, M., 2003. UML Distilled: A Brief Guide to the Standard Object Modeling Language, third ed. Addison–Wesley Professional.

Gilbert, N., Troitzsch, K., 2005. Simulation for the Social Scientist, second ed. Open University Press, Milton Keynes.

Ginot, V., Le Page, C., Souissi, S., 2002. A multi-agents architecture to enhance end-user individual-based modelling. Ecol. Model. 157, 23–41.

Gopen, G.D., Swan, J.A., 1990. The science of scientific writing. Am. Sci. 78, 550–559.

Grimm, V., 1999. Ten years of individual-based modelling in ecology: What have we learned, and what could we learn in the future? Ecol. Model. 115, 129–148.

Grimm, V., 2002. Visual debugging: a way of analyzing, understanding, and communicating bottom-up simulation models in ecology. Nat. Res. Model. 15, 23–38.

Grimm, V., Dorndorf, N., Frey-Roos, F., Wissel, C., Wyszomirski, T., Arnold, W., 2003. Modelling the role of social behavior in the persistence of the alpine marmot *Marmota marmota*. Oikos 102, 124–136.

Grimm, V., Railsback, S.F., 2005. Individual-Based Modeling and Ecology. Princeton University Press, Princeton, NJ.

Grimm, V., Wissel, C., 2004. The intrinsic mean time to extinction: a unifying approach to analyzing persistence and viability of populations. Oikos 105, 501–511.

Grimm, V., Wyszomirski, T., Aikman, D., Uchmanski, J., 1999. Individual-based modelling and ecological theory: synthesis of a workshop. Ecol. Model. 115, 275–282.

Hackländer, K., Arnold, W., 1999. Male-caused failure of female reproduction and its adaptive value in alpine marmots (*Marmota marmota*). Behav. Ecol. 10, 592–597.

Hales, D., Rouchier, J., Edmonds, B., 2003. Model-to-model analysis. J. Artif. Soc. Soc. Simul. 6, http://jasss.soc.surrey.ac.uk/6-4/5.html.

Huckfeldt, R., Johnson, P.E., Sprague, J.D., 2004. Political Disagreement: The Survival of Diverse Opinions within

Communication Networks. Cambridge University Press, Cambridge, UK.
Huse, G., Giske, J., Salvanes, A.G.V., 2002. Individual-based modelling. In: Hart, P.J.B., Reynolds, J. (Eds.), Handbook of Fish and Fisheries. Blackwell, Oxford, pp. 228–248.
Huth, A., Wissel, C., 1992. The simulation of the movement of fish schools. J. Theor. Biol. 156, 365–385.
Huth, A., Wissel, C., 1994. The simulation of fish schools in comparison with experimental data. Ecol. Model. 75–76, 135–146.
Inada, Y., Kawachi, K., 2002. Order and flexibility in the motion of fish schools. J. Theor. Biol. 214, 371–387.
Kunz, H., Hemelrijk, C.K., 2003. Artificial fish schools: collective effects of school size, body size, and form. Artif. Life 9, 237–253.
Liu, J., Ashton, P.S., 1995. Individual-based simulation models for forest succession and management. Forest Ecol. Manage. 73, 157–175.
Lorek, H., Sonnenschein, M., 1999. Modelling and simulation software to support individual-oriented ecological modelling. Ecol. Model. 115, 199–216.
Parker, D.C., Manson, S.M., Janssen, M.A., Hoffmann, M.J., Deadman, P., 2003. Multi-agent systems for the simulation of land-use and land-cover change: a review. Ann. Assoc. Am. Geogr. 93, 314–337.
Peck, S.L., 2004. Simulation as experiment: a philosophical reassessment for biological modeling. Trends Ecol. Evol. 19, 530–534.
Pitt, W.C., Box, P.W., Knowlton, F.F., 2003. An individual-based model of canid populations: modelling territoriality and social structure. Ecol. Model. 166, 109–121.
Railsback, S.F., 2001. Concepts from complex adaptive systems as a framework for individual-based modelling. Ecol. Model. 139, 47–62.
Reuter, H., Breckling, B., 1994. Selforganization of fish schools: an object-oriented model. Ecol. Model. 75/76, 147–159.
Rouchier, J., 2003. Re-implementation of a multi-agent model aimed at sustaining experimental economic research: the case of simulations with emerging speculation. J. Artif. Soc. Soc. Simul. 6, http://jasss.soc.surrey.ac.uk/6-4/7.html.
Shugart, H.H., 1984. A Theory of Forest Dynamics: The Ecological Implications of Forest Succession Models. Springer-Verlag, New York.
Shugart, H.H., Smith, T.M., Post, W.M., 1992. The potential for application of individual-based simulation models for assessing the effects of global change. Annu. Rev. Ecol. Syst. 23, 15–38.
Strand, E., Huse, G., Giske, J., 2002. Artificial evolution of life history and behavior. Am. Nat. 159, 624–644.
Tesfatsion, L., 2002. Agent-based computational economics: growing economies from the bottom up. Artif. Life 8, 55–82.
Tyler, J.A., Rose, K.A., 1994. Individual variability and spatial heterogeneity in fish population models. Rev. Fish Biol. Fisheries 4, 91–123.
Van Winkle, W., Rose, K.A., Chambers, R.C., 1993. Individual-based approach to fish population dynamics: an overview. Trans. Am. Fish. Soc. 122, 397–403.
Zeigler, B.P., Praehofer, H., Kim, T.G., 2000. Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems, second ed. Academic Press.