



**HAL**  
open science

## Using the ODD protocol and NetLogo to replicate agent-based models

Volker Grimm, Uta Berger, Justin M. Calabrese, Ainara Cortés-Avizanda, Jordi Ferrer, Mathias Franz, Jürgen Groeneveld, Florian Hartig, Oliver Jakoby, Roger Jovani, et al.

### ► To cite this version:

Volker Grimm, Uta Berger, Justin M. Calabrese, Ainara Cortés-Avizanda, Jordi Ferrer, et al.. Using the ODD protocol and NetLogo to replicate agent-based models. *Ecological Modelling*, 2025, 501, pp.110967. 10.1016/j.ecolmodel.2024.110967 . hal-04824366

**HAL Id: hal-04824366**

**<https://hal.science/hal-04824366v1>**

Submitted on 6 Dec 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License

# Using the ODD protocol and NetLogo to replicate agent-based models

Volker Grimm<sup>1,2</sup>, Uta Berger<sup>3</sup>, Justin M. Calabrese<sup>1,4,5</sup>, Ainara Cortés-Avizanda<sup>6,1</sup>, Jordi Ferrer<sup>7</sup>, Mathias Franz<sup>8,9</sup>, Jürgen Groeneveld<sup>1</sup>, Florian Hartig<sup>10</sup>, Oliver Jakoby<sup>11</sup>, Roger Jovani<sup>12</sup>, Stephanie Kramer-Schadt<sup>13,14</sup>, Tamara Münkemüller<sup>15</sup>, Cyril Piou<sup>16</sup>, L. S. Premo<sup>17</sup>, Sandro Pütz<sup>18,1</sup>, Thomas Quintaine<sup>19</sup>, Christine Rademacher<sup>20</sup>, Nadja Rüger<sup>21,22,23</sup>, Amelie Schmolke<sup>11</sup>, Jan C. Thiele<sup>24,25</sup>, Julia Touza<sup>26</sup>, Steven F. Railsback<sup>27</sup>

1 Helmholtz Centre for Environmental Research – UFZ, Department of Ecological Modelling, Leipzig, Germany

2 University of Potsdam, Plant Ecology and Nature Conservation, Potsdam, Germany

3 TUD Dresden University of Technology, Department of Environmental Sciences, Dresden, Germany

4 CASUS - Center for Advanced Systems Understanding, Helmholtz-Zentrum Dresden-Rossendorf e.V. (HZDR), Earth System Science, Görlitz, Germany

5 Department of Biology, University of Maryland, College Park, MD, USA

6 EBD-CSIC, Estación Biológica de Doñana, Department of Conservation Biology and Global Change, Sevilla, Spain

7 Lycée International de Barcelona Bon Soleil, Barcelona, Spain

8 Max Planck Institute for Evolutionary Anthropology, Department of Primatology, Leipzig, Germany

9 Freie Universität Berlin, Institute of Biology, Berlin Germany

10 University of Regensburg, Faculty of Biology and Pre-Clinical Medicine, Group of Theoretical Ecology, Regensburg, Germany

11 RIFCON GmbH, Effect Modelling and Statistics, Hirschberg, Germany

12 EBD-CSIC, Estación Biológica de Doñana, Department of Ecology and Evolution, Sevilla, Spain

13 Leibniz Institute for Zoo and Wildlife Research IZW, Department of Ecological Dynamics, Berlin, Germany

14 Technische Universität Berlin, Institute of Ecology, Berlin, Germany

15 Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LECA, Grenoble, France

16 CIRAD, UMR CBGP, INRAE, IRD, Institut Agro Montpellier, Univ. Montpellier, Montpellier sur Lez, France

17 Washington State University, Department of Anthropology, Pullman, WA, USA

18 Federal Agency for Nature Conservation – BfN, Department of Terrestrial Monitoring, Leipzig, Germany

19 Université de Reims Champagne-Ardenne, Centre de Recherche et Formation en Eco-éthologie, Boult-aux-Bois, France

20 Technische Hochschule Nürnberg Georg Simon Ohm, Nürnberg, Germany

21 German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig, Leipzig, Germany

22 Leipzig University, Institute of Empirical Economic Research, Department of Economics, Leipzig, Germany

23 Smithsonian Tropical Research Institute, Balboa, Ancón, Panama

24 Georg-August-University of Goettingen, Department of Ecoinformatics, Biometrics and Forest Growth,  
Goettingen, Germany

25 Bremische Bürgerschaft, Bremen, Germany

26 University of York, Department of Environment and Geography and York Environmental Sustainability  
Institute, York, UK

27 Lang Railsback and Associates, Arcata, California, USA

Correspondence author: [volker.grimm@ufz.de](mailto:volker.grimm@ufz.de)

## 1 Abstract

2 Replicating existing models and their key results not only adds credibility to the original  
3 work, it also allows modellers to start model development from an existing approach rather  
4 than from scratch. New theory can then be developed by changing the assumptions or  
5 scenarios tested, or by carrying out more in-depth analysis of the model. However, model  
6 replication can be challenging if the original model description is incomplete or ambiguous.  
7 Here we show that the use of standards can facilitate and speed up replication: the ODD  
8 protocol for describing models, and NetLogo, an easy-to-learn but powerful software  
9 platform and language for implementing agent-based models. To demonstrate the benefits  
10 of this approach, we conducted a replication experiment on 18 agent-based models from  
11 different disciplines. The researchers **doing** the replications had no or little previous  
12 experience using ODD and NetLogo. Their task was to rewrite the original model description  
13 using ODD, implement the model in NetLogo and try to replicate at least one exemplary  
14 main result. They were also asked to produce, **if** time allowed, some initial new results with  
15 the replicated model, and to record the total time spent on the replication exercise.  
16 Replication was successful for 15 out of 18 models. The time taken varied between 2 and 12  
17 days, with an average of 5 days. ODD helped to systematically scan the original model  
18 description, while NetLogo proved easy and quick to learn, but difficult to debug when  
19 implementation problems arose. Although most of the models replicated were relatively  
20 simple, we conclude that even for more complex models it can be useful to use ODD and  
21 NetLogo for replication, at least for developing a prototype to help decide how to proceed  
22 with the replicated model. Overall, the use of **both**, standard approaches **such as ODD and**  
23 **easy to learn but powerful software such as NetLogo**, can promote coherence and efficiency  
24 within and between different models and modelling communities. Imagine if all modellers  
25 spoke ODD and NetLogo as a common language or *lingua franca*.

26 **Keywords:** model replication, standards, ODD protocol, NetLogo, agent-based model, theory  
27 development

## 28 1. Introduction

29 Agent-based simulation models are widely used in ecology, environmental and social  
30 sciences. Those models are typically spatially explicit and include autonomous and  
31 potentially unique decision-making of the central modelling units (the agents) in response to  
32 their environment. When these models are used to understand and predict the dynamics of  
33 real systems, they often behave in case-specific ways so that insights gained are difficult to  
34 generalize to other systems or environmental settings. Thus, a challenge in the field is to  
35 identify general principles and theories that explain how the behaviour of agents controls  
36 the emergence of structure and dynamics of real systems.

37 The aim of gaining general insights from agent-based models may be limited by the  
38 complexity and diversity of these systems (Antosz et al., 2023; Grimm et al., 2024;  
39 Wijermans et al., 2023), but we hold that there are also aspects of modelling practice and  
40 culture that hinders the development of more general, but still predictive, models:  
41 developing models more or less from scratch seems to be more valued by both individual  
42 modellers and the modelling community than starting from existing models.

43 There are several reasons for this. Modellers often learn more about the system in question  
44 during model development than from the final model. It has therefore been recommended  
45 to document not only the final model, but also the key analyses and decisions made during  
46 the modelling process (Ayllón et al., 2021; Grimm et al., 2014; Schmolke et al., 2010).  
47 Moreover, just using an existing model, for example by changing some settings and  
48 parameters, carries the risk of using the model beyond its original purpose and blindly  
49 trusting its results rather than trying to understand how they are generated.

50 Here, we suggest a middle way between developing models from scratch and just using  
51 existing models: starting new projects with model replication (Thiele and Grimm, 2015). In  
52 science, replication generally means repeating an existing study, based on the original  
53 description of the materials and methods used, and obtaining consistent results.  
54 Replicability is considered a cornerstone of the scientific method. It lends credibility because,  
55 in principle, conclusions become independent of the person, place and time involved in a  
56 study (Zhang and Robinson, 2021). This is relevant in ecology for both modelling and  
57 empirical studies (Filazzola and Cahill Jr, 2021; Fraser et al., 2020; Parker et al., 2016).

58 **Reproducing** the results of an original study, however, **achieves** more than **just confirming**  
59 **the original results** (Axtell et al., 1996; Gürcan et al., 2023; Zhong and Kim, 2010).  
60 **Reproducing a study also re-establishes the** methods of the original experiment have been  
61 established, so that they can now be used to modify the original question and the settings of  
62 the experiment, and thereby generate new insights. In modelling, for example, one could  
63 perform a Robustness Analysis, that is, try to systematically simplify the model to learn  
64 which model assumptions were essential to reproduce the observed patterns (Grimm and  
65 Berger, 2016). Or, the original results can be generalised by exploring additional scenarios,  
66 and additional, more in-depth simulation experiments can improve our understanding of the  
67 dynamics of the system under study (Carney and Davies, 2020; Hauke et al., 2020). The  
68 replication exercise can also reveal weaknesses in the assumptions, or implementation, of  
69 the original model (Edmonds and Hales, 2003), and replication can also be part of the  
70 attempt to facilitate **the** alignment and comparison of different models that **address** the  
71 same question **but give different answers** (Banitz et al., 2022; Hales et al., 2003). Such multi-  
72 **model tests, now often referred to as 'model intercomparison projects'** (MIP, e.g.  
73 **Warszawski et al. (2014)) are challenging but increasingly needed because we need to**  
74 **understand why the answers are different before our models can robustly support decision**  
75 **making.**

76 **A further direct benefit of replicating a modelling study is that** “the modeller starts from  
77 existing hypotheses and submodels and therefore enters the ‚modelling cycle‘ (Grimm and  
78 Railsback 2005) of iterative model formulation, implementation, simulation, and analysis  
79 much more quickly. This leaves more time for the critical but time-consuming task of in-  
80 depth model analysis, which is often performed only superficially because too much time  
81 had been spent developing the first conceptual model.” (Thiele and Grimm, 2015, p. 692).  
82 However, replicating models can be difficult if the model has not been described in detail,  
83 and even if it has, there may be subtle details in the implementation of the model in a  
84 particular programming language that are not documented (Grimm et al., 2020b).

85 **To standardize** model descriptions, the ODD protocol is increasingly used as a standard  
86 format for describing agent-based models, and is now also being used for **mathematical**  
87 **models** (Grimm et al., 2006; Grimm et al., 2010; Grimm et al., 2020b; Polhill et al., 2008); **as**  
88 **for standardized ABM descriptions including human behaviour, see Müller et al. (2014). The**  
89 **ODD protocol recommends certain structure of the model description and provides a list of**  
90 **items that should be explained.** The latest update of ODD (Grimm et al., 2020b) also  
91 recommends that the written model description should be explicitly linked to the code that  
92 implements the model, to further reduce ambiguity.

93 Another challenge in model replication is the variety of programming languages, operating  
94 systems, and software platforms used to implement models. Ideally, results should be  
95 independent of the details of a model's implementation, but exact matching is rarely  
96 possible, often due to the use of different random number generators or different floating-  
97 point algorithms. However, even achieving similar output distributions, or at least the same  
98 qualitative relationships between model input and output (Axtell et al., 1996), may fail.

99 Ideally, all models would be described in a format such as ODD and implemented in a  
100 generic software platform with a programming language that is easy to learn, understand,  
101 and execute. While this ideal seems unrealistic, we will show here that we are not far from  
102 it. We will show that translating ad hoc model descriptions into the ODD format and then  
103 implementing them using NetLogo<sup>1</sup> (Wilensky, 1999; Wilensky and Rand, 2015), a generic  
104 software platform and language originally developed for teaching and for agent-based  
105 modelling, is straightforward to learn and can make model replication easier and faster than  
106 is generally believed.

107 Model replication can, as we will show, be easy and fast even for those unfamiliar with ODD  
108 and NetLogo. Using ODD and NetLogo, we successfully reimplemented 15 of 18 models and  
109 replicated the original results. Below we list these models and summarise our experiences,  
110 including the time spent for the replication exercise, challenges and recommendations. We  
111 will suggest using ODD as a standard for model replication and NetLogo as a versatile and  
112 efficient tool for at least prototyping a replicated model.

113 Our results show that describing a model according to a standard such as the ODD protocol  
114 not only makes models transparent, but can also support replication, even for modellers  
115 previously unfamiliar with ODD and NetLogo and even if the original model did not use the  
116 ODD description. Replication should be even more likely to succeed if the original  
117 description follows ODD and is well linked to the accessible code. To our knowledge, this is  
118 the first study that tested the potential of ODD to support replication. We argue that  
119 standards like ODD thus contribute to Good Modelling Practice, which is increasingly needed  
120 and discussed as models are increasingly needed to support environmental and ecological  
121 decision making (Jakeman et al., 2024). Moreover, certain software platforms can actually  
122 become quasi-standards, as they are widely used within a certain community or domain,  
123 such as R for statistical analyses and modelling, Python for data science, and NetLogo for  
124 agent-based modelling.

---

<sup>1</sup> <https://ccl.northwestern.edu/netlogo/>

## 125 **2. Methods**

126 The replications presented here were conducted in 2009 to demonstrate the standardising  
127 power of the ODD protocol, which had been published only three years earlier in 2006.  
128 However, as ODD was quickly adopted by developers of individual and agent-based models  
129 (Grimm et al., 2010), there no longer seemed to be a strong need to advocate the use of the  
130 ODD protocol. There were also doubts as to whether NetLogo should really be  
131 recommended for wider use in science, given its origins as educational software and the  
132 perception that it would not be suitable for more complex models. Also, the importance and  
133 benefits of replication were not yet fully articulated or widely accepted.

134 Now, 15 years later, the situation has changed. The ODD protocol is widely used (Vincenot  
135 2018), and NetLogo has become the model language of choice in the social sciences, and is  
136 increasingly being used in ecology and other disciplines (Abar et al., 2017; Railsback et al.,  
137 2017; Vincenot, 2018). NetLogo has been shown **competent for** complex simulation models  
138 (e.g. BEEHAVE, Becher et al. 2014) **because its built-in primitives are highly efficient**, certain  
139 programming techniques/tricks can speed it up substantially (Railsback et al. 2017), and it  
140 can easily be run on High Performance Computing clusters (e.g., Ayllón et al. (2016),  
141 Gallagher et al. (2021)). One advantage of NetLogo is the ease with which models can be  
142 adopted and used by others, provided the model is well documented. For example, the  
143 honey bee colony model BEEHAVE (Becher et al., 2014) has been used in more than 25  
144 publications, about two-thirds of them without the involvement of any of the BEEHAVE  
145 developers (Groeneveld et al., 2024).

### 146 **2.1. ODD and NetLogo**

147 ODD model descriptions consist of seven elements (Table 1).

	ODD protocol	NetLogo elements
Overview	1. Purpose and patterns	Information tab
	2. Entities, state variables and scales	breeds, turtles-own, patches-own, globals
	3. Process overview and scheduling	„go“ procedure
Design concepts	4. Design concepts <ul style="list-style-type: none"> <li>- Emergence</li> <li>- Adaptation</li> <li>- Objectives</li> <li>- Learning</li> <li>- Prediction</li> <li>- Sensing</li> <li>- Interaction</li> <li>- Stochasticity</li> <li>- Collectives</li> <li>- Observation</li> </ul>	Information tab, primitives  Elements of the Graphical User Interface
Details	5. Initialization	„setup“ procedure
	6. Input data	File input
	7. Submodels	Procedures and reporters

148

149 **Table 1.** Partial correspondence of the elements of the ODD protocol and the software platform and  
150 programming language NetLogo.

151 The first three elements provide an overview, so that the reader does not need to read the  
152 entire model description to get an idea of what the model is and does; they also allow the  
153 reader to zoom in on the details of most interest. The fourth element describes important  
154 concepts underlying the design of the model; this acts as a checklist so that important design  
155 decisions are made explicit. The last three elements provide details of how the model is  
156 initiated, the data inputs representing external drivers, and the sub-models used to  
157 implement the model's processes.

158 NetLogo is both a programming language specifically designed for implementing agent-  
159 models (including cellular automaton-like models without mobile agents), and a software  
160 platform for programming and running models. It is the most widely used "offspring" of  
161 Logo, which was originally developed to introduce students or school children to modelling  
162 and simulation (Papert, 1985); it therefore still uses the "turtle" label for agents by default,  
163 but this can be easily changed to represent any autonomous agent.

164 NetLogo is free and open-source software released under a GNU General Public License  
165 (GPL), and runs on all major operating systems. It is well documented and comes with  
166 training material and a large library of example models, and it has been continuously  
167 maintained and developed since its first release in 1999. NetLogo is implemented in Java

168 and Scala, it can be run interactively with an easy-to-build Graphical User Interface, and it  
169 allows execution in headless mode on high-performance computing (HPC). NetLogo has  
170 been an unusually stable platform: most of our replicated models were implemented in  
171 version 4.0.5 of NetLogo in 2009 but we were able to run them all in the current version  
172 (6.4.0) with only a few simple update steps and almost no changes in code. NetLogo was  
173 specifically designed to facilitate programming agent-models. It includes powerful concepts,  
174 such as agentsets (set of turtles, patches or links with certain properties) and a large number  
175 of so-called primitives, which are procedures for representing tasks and algorithms which  
176 are needed in virtually all ABMs, such identifying the set of all neighbour grid cells,  
177 identifying all turtles within a certain radius, or moving in a certain direction and distance.

178 Interestingly, some NetLogo conventions closely correspond to the structure of ODD  
179 (Table 1). Virtually all NetLogo programs have a “setup” button, which corresponds to ODD's  
180 fourth element, “Initialisation”, and a “go” button, which corresponds to ODD's “Process  
181 Overview and Scheduling” element. NetLogo’s “procedures” and “reporters” (subroutines  
182 and functions) correspond to the ODD element “Submodels”.

## 183 **2.2. The models**

184 We conducted a replication exercise with 17 previously published models and one  
185 unpublished one. There were no restrictions on the choice of model for the replication  
186 exercise, except that the replicating person must not have been involved in the development  
187 of the model. We avoided overly complex models that would have taken more time to  
188 replicate and thoroughly test than would have been useful for a demonstration exercise. The  
189 18 models (Table 2) were from ecology, social sciences/game theory, microbiology,  
190 epidemiology, and behavioural ecology.

191 For the 17 published models, no original code was used initially, and only in two cases was  
192 the original code asked for and provided (Fielding, 2004; Wissel, 1992). Only four studies  
193 mention the programming language used: Java (Davis et al., 1999), C++ (Fielding, 2004);  
194 Voyons (Thiery et al., 1995), and C (Weiner et al., 2001).

195 **Table 2.** Overview of the 18 models that were used for the replication exercise in this study.

196

Reference	System and question	Replication success <sup>1</sup>	Gaps in original model description <sup>2</sup>	New Results	Time / days <sup>3</sup>
Fielding (2004)	Intraspecific competition and spatial heterogeneity alter life history traits in an individual-based model of grasshoppers.	Partly	Parameter, Code, Equations	No	12
Weiner et al. (2001)	The effects of density, spatial pattern, and competitive symmetry on size variation in simulated plant populations.	Yes	No	Yes	4.5
Hauert and Doebeli (2004)*	Limited evolution of cooperation in a spatially explicit snowdrift game	Yes	No	Yes	4
Wissel (1992)	Modelling the mosaic cycle of a Middle European beech forest	Yes	Rules, Scheduling, Code, Author	No	5
Deffuant et al. (2002)*	How can extremism prevail? A study based on the relative agreement interaction model.	Yes	No	No	4
Silvertown et al. (1992)*	Cellular automaton models of interspecific competition for space – the effect of pattern on process.	Yes	No	No	2
Axelrod (1997)*	The dissemination of culture: A model with local convergence and global polarization.	Yes	No	No	8
DeAngelis et al. (1979)	Cannibalism and size dispersal in young-of-the-year largemouth bass: experiment and model.	Yes	Parameter	Yes	5
Hauert et al. (2002)*	Volunteering as Red Queen mechanism for cooperation in public goods games.	Yes	No	No	5.5
Davis et al. (1999)	Environmental quality predicts parental provisioning decisions.	No	Rules, Equations	No	4.5
O'Keefe (2005)	The evolution of virulence in pathogens with frequency-dependent transmission.	Partly	Parameter, Equations	No	4
Kerr et al. (2002)*	Local dispersal promotes biodiversity in a real-life game of rock–paper–scissors.	Yes	Rules	No	2.25
Hansen et al. (2004)	Modelling the transmission cycle of the fox Tapeworm	Yes	No	Yes	8 N
Thiery et al. (1995)	A model simulating the genesis of banded vegetation patterns in Niger.	Yes	No	No	4
Jackson et al. (2008)*	The effect of social facilitation on foraging success in vultures: a modelling study	Yes	No	No	10 N
Ratz (1995)	Long-term spatial patterns created by fire: a model oriented towards boreal forests.	Yes	No	Yes	8.5
Ginovart et al. (2002)	Individual based simulations of bacterial growth in agar plates.	Yes	Parameters, Authors, Code, Equations	Yes	7 N
A.Costopoulos, R. Jobling (unpublished)*	How does the rate and magnitude of cultural innovation affect the ability of hominin groups to expand their species' ecological range?	No	Parameter ranges, Rules, Authors, Repetitions	Yes	10.5

<sup>1</sup> Yes: main results qualitatively reproduced; Partly: yes, but not for all scenarios; No: replication failed

<sup>2</sup> Parameter: settings incomplete or unknown; code: available but uncommented; rules/equations/scheduling: partly inconsistent or illogical; authors: were contacted and replied

<sup>3</sup> 1 day = 8 hours

N: Replicating person was beginner in programming and modelling and got help from colleagues

\*: Stylized model, i.e. not based on specific data or observed patterns

### 197 **2.3. Replication**

198 Those replicating the models were recruited from the Department of Ecological Modelling at  
 199 the UFZ, or were close collaborators. Their motivation was to learn about the usefulness and  
 200 potential of ODD and NetLogo as well as the replicability of modelling results. They were  
 201 mostly experienced modellers who routinely used at least one programming language for  
 202 their work, but only one of them has a background in computer science and software

203 engineering (JCT). Three replicating persons did not have any experience in programming  
204 and modelling, but got some help from experienced colleagues.

205 The replication tasks were: 1) rewrite the original model description according to the ODD  
206 protocol (original ODD version from 2006), 2) re-implement the model using NetLogo, 3) try  
207 to replicate at least one exemplary main result, 4) try to, if time allowed, achieve some first  
208 new results, either by running new scenarios or by modifying the original model  
209 assumptions, and 5) report on the time required for the whole exercise and on the  
210 challenges during the entire task.

### 211 **3. Results**

212 All 18 ODD model descriptions and the corresponding reports about replication of original  
213 results, new results, and general comments, are compiled in Supplementary Material S1. The  
214 NetLogo programs, in NetLogo versions 4.0.5 and 6.4.0, are in the Supplementary Material  
215 S2.

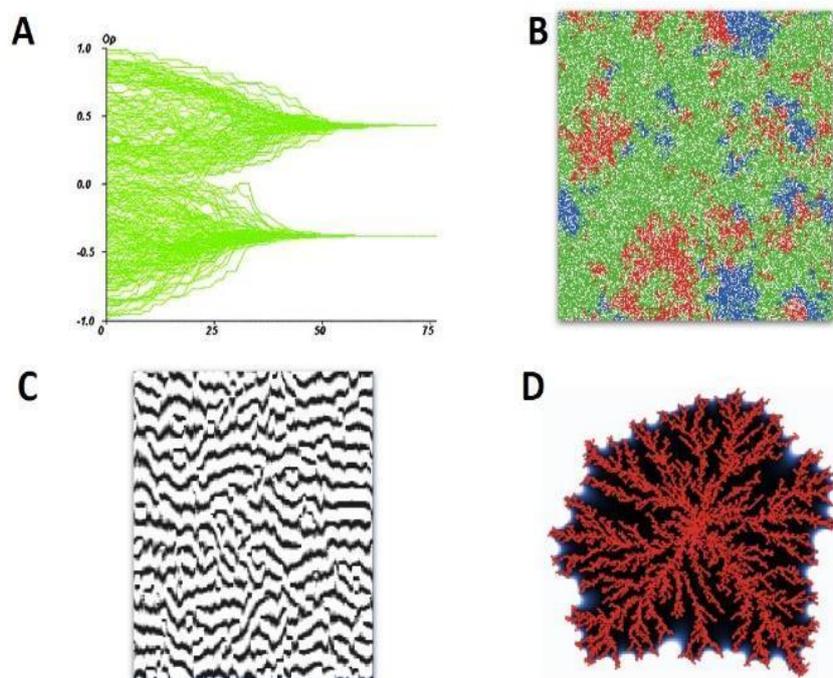
216 For 15 of the 18 models, exemplary main results were reproduced; for the two other  
217 models, only partial replication was achieved because gaps and ambiguities in the original  
218 model description could not be closed or resolved, respectively, and for one, albeit  
219 unpublished, model replication failed (Table 2). In the original model descriptions of eight of  
220 the models, some key information was missing, including parameter values (4 models),  
221 parameter ranges (1), model equations or rules (6), or scheduling of processes (1). In three  
222 cases help from the original authors was needed and thankfully provided. In Figure 1,  
223 example output of four of the replicated models is presented.

224 Those who commented on using ODD in the replication exercises were positive, in particular  
225 regarding the O-part as it helped to scan and rewrite the original model description in a  
226 systematic way. There was one complaint about some redundancy in ODD, but this has been  
227 discussed and justified in Grimm et al. (2010). In most exercises, NetLogo code was included  
228 in the ODD model description, but this is against one of the main aims of ODD to be  
229 independent of programming languages: ODD is meant as a verbal, or written, description.  
230 Still, having a few lines of code included can be acceptable if they serve more or less as  
231 pseudocode and help **specify** implementation details (see, for example, the ODD of the  
232 model BEEHAVE, Becher et al. (2014)).

233 NetLogo turned out to be easy and fast to learn, although this requires getting used to some  
234 new and unique concepts, such as “asking” sets of agents, which share a certain trait, to do

235 something instead of using loops and if-then conditions. Also the three novice programmers  
 236 succeeded, with some help of colleagues with experience in programming.

237

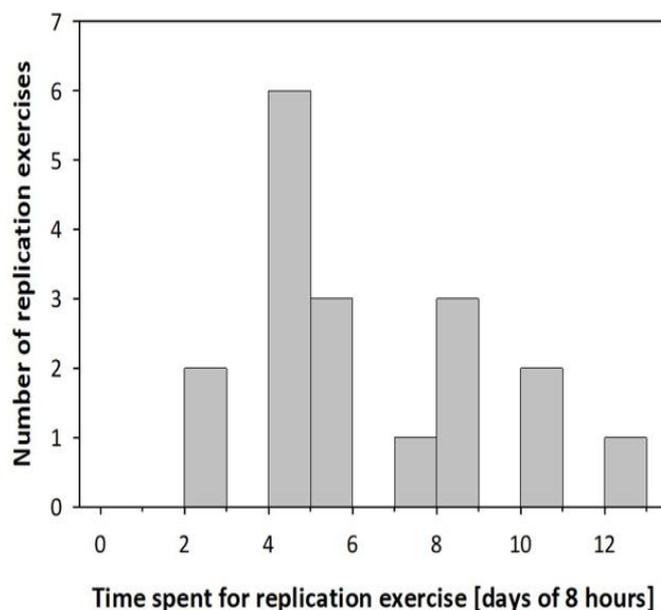


238

239 **Figure 1.** Output of replicated models that matched the output of four of the original models. A:  
 240 Opinion dynamics (opinion between -1 and +1 vs time, Deffuant et al., 2002); B: Rock-paper-scissor  
 241 bacterial community dynamics (spatial distribution of three different strains of bacteria, Kerr et al.  
 242 2002); C: Banded vegetation pattern emerging in semi-arid regions (black: wood vegetation, white:  
 243 other vegetation or bare soil, Thiery et al. 1995); D: Growth of a bacterial colony (Ginovart et al.,  
 244 2002). See supplements for details.

245 Virtually in all replication exercises, the programmer missed having an integrated debugger  
 246 that allows the program to be executed line-by-line while watching how variables are  
 247 changing, or to set breakpoints that would halt execution of the program in certain parts of  
 248 the code or under certain conditions. For debugging, NetLogo allows us to use aspects of the  
 249 Graphical User Interface for low-level debugging, e.g. by using different colours for different  
 250 states of the agents or by attaching labels to the agents. Such techniques are too limited for  
 251 full debugging, so one must use print statements or file output to observe if and how state  
 252 variables are changing. Another frequent complaint in 2009 was slow execution speed, but,  
 253 unlike the lack of a debugger, execution speed is no longer an issue. Now, NetLogo is widely  
 254 used successfully for large models (Railsback et al., 2017) because computers are much  
 255 faster, because HPC clusters or cloud computing are increasingly available, and because  
 256 NetLogo's built-in commands (primitives) use highly optimized algorithms.

257 In six replications, some new results were also produced (Table 2), which was an optional  
 258 task. The time invested in the replication exercise varied between 2 and 12 days (8 hours per  
 259 day) with a mean of 5 days (Figure 2) for the relatively simple models we used. In most cases  
 260 testing and debugging the code was the most time-consuming task.



261

262 **Figure 2.** Histogram of the time spent replicating the 18 models. The mean time spent is five days.

## 263 4. Discussion

264 The purpose of this study was to show that, using ODD and NetLogo, existing agent-models  
 265 can be replicated reliably and with relatively low effort. We have also shown in some of the  
 266 examples that once one has replicated a model, one can start new research by modifying the  
 267 model's assumptions and structure, or by running new simulation experiments that help to  
 268 better understand the original model or apply it to new parameter settings.

269 Replication is challenging because model descriptions usually show only the end result of  
 270 iterative model development. It can be difficult to understand the rationale for all the  
 271 model's assumptions without knowing the path by which the model was developed, and  
 272 it can be difficult to describe this rationale in sufficient detail for a model to be fully  
 273 understood. Also, verbal descriptions of models, even when attempting to be complete,  
 274 often contain ambiguities that can only be resolved by having access to the code that  
 275 implements the model.

276 We found the time taken to replicate to be surprisingly low, averaging just five days. This is  
 277 even more remarkable considering that those involved in the replication exercises had no or

278 very little prior experience with using NetLogo or ODD. Presumably, replication could be  
279 even faster with experience writing and testing NetLogo code.

280 Obviously, replicating more complex models than those used in this study will take longer.  
281 For example, a NetLogo replication of the complex and highly realistic trout model Instream  
282 (Railsback and Harvey, 2002) took about six months, but then the model was augmented  
283 with population genetics to study the evolution of specific traits under environmental  
284 change (Ayllón et al., 2018). In these six months, a deep understanding of the model  
285 assumptions and structure, and therefore of fish physiology and behaviour, as well as river  
286 flow dynamics and habitat quality, was gained.

287 Furthermore, the risk of misinterpreting the verbal model description increases for complex  
288 models. Donkin et al. (2017) first re-implemented a fairly complex ABM using NetLogo and  
289 were unable to replicate the original results at any level, i.e. numerical, distributional or  
290 relational (Axtell et al., 1996). They attributed these differences to a misunderstanding of  
291 how key mechanisms were interpreted without having the original code. However, when  
292 they attempted to replicate the results of their NetLogo implementation using Repast  
293 Symphony (North et al., 2013), the results again did not match. They attributed the  
294 difference to the different programming styles required for the two platforms. While  
295 NetLogo includes many high-level primitives for describing agent behaviour and decisions,  
296 Repast, which is a Java library, requires more elements of a model to be implemented from  
297 scratch.

298 We take the study by Donkin et al. (2017) as a caveat to be careful to fully understand how  
299 NetLogo's primitives work and, if in doubt, to test them carefully. The same applies, of  
300 course, to code you wrote yourself. Therefore, key behaviours should ideally be  
301 implemented in two different languages and produce the same results, to be sure that the  
302 implementation really does what the model description says it does, and that the behaviour  
303 of the submodel of that behaviour is fully understood (see an educational example in  
304 Chapter 12 of Railsback and Grimm (2019)).

305 The advantage of using ODD for replication is that it provides a checklist for verifying all the  
306 key features of a model, starting with the purpose, structure and scheduling of the model,  
307 and including all details of initialisation, file input and any sub-models. Model replication can  
308 fail at any of these levels, and indeed in our study, some examples reported missing  
309 information on parameter values and ranges, scheduling, initial settings or, most commonly,  
310 details of model equations and algorithms. Thus, the use of ODD could cause reprogramming

311 to be postponed or even cancelled, depending on the size of the information gap and  
312 whether it was possible to obtain the missing information from the original authors.

313 ODD is designed to contain all the information needed to replicate a model, but ambiguities  
314 are hard to avoid. Therefore, the latest ODD update (Grimm et al., 2020) recommends that  
315 the ODD description be closely aligned with the code, using the same names for variables,  
316 parameters and procedures, and including numbers of equations, model rules or algorithms  
317 as comments in the code. With the increasing popularity and acceptance of the FAIR  
318 principles (Wilkinson et al., 2016), more and more journals require not only data but also  
319 scripts and code to be made available, and it seems natural and very useful to actually  
320 provide a model description that is tightly integrated with the code.

321 In general, we recommend that every modeller should try to replicate a model at least once,  
322 because by learning how difficult it can be to replicate a model based on an incomplete or  
323 inconsistent model description, one also learns to improve one's own model description and  
324 code preparation. In Table 3, we summarise general recommendations for communicating a  
325 model and how it has been used, which may serve as a checklist by both model developers  
326 and model replicators. If such a checklist were widely followed, it would implicitly establish  
327 good modelling practice. Gürcan et al. (2023) provide a more detailed checklist for the  
328 replication of agent-based models, using the HUMAT model as an example (Antosz et al.,  
329 2022).

330 **Table 3.** A short checklist of questions modeller and those interested in model replication should ask  
331 themselves to make sure, for modellers, that their model can be both re-run and replicated, and for  
332 replicators that they have sufficient information to make replication work.

333

Replication question	Techniques to ensure replicability
Is the model description complete?  Would I be able to replicate my model solely based on the model description?	Use ODD. Write it by describing what your program does, not what you think the model is. Be specific, explicit, and complete about parameter settings and ranges, scheduling of actions, initial settings, data imported during runtime, and all details of the equations and algorithms used. Let others who do not know your model read the ODD. <b>Try to separately replicate complex submodels yourself in a different language (R, Python, C++, Julia etc.)</b>
Is the model description well-linked to the program implementing the model?	Use the same names for variables, parameters, and procedures in the ODD and in the program. Numbers of equations or algorithms in the ODD should be added as comments in the program.
Did you provide all files to re-run your model?	These include the source code and possibly files for data or parameter input, plus documentation of the software platform

	and operating system used, and which version.
Are the parameter settings underlying all scenarios and simulation experiments fully specified?	In NetLogo, include any BehaviorSpace experiment setups used, otherwise provide all parameter settings and scenario descriptions used for all results presented.
If your workflow included any scripts for, e.g. processing input data or output analysis, are those scripts included?	Include them. Often a lot of work goes into processing raw data or model output, so this is part of your achievements. Others might fail in using your model if the workflow is not fully documented or the scripts unavailable.

334

335 The original motivation for the replication exercises presented here was to suggest that ODD  
336 and NetLogo can be used as a common language or *lingua franca* in the modelling  
337 community. ODD has indeed become a widely used standard (Vincenot, 2018), and NetLogo  
338 also has thousands of users, but we are not suggesting that all modellers should use it  
339 exclusively. Rather, all modellers **might want to know** how to use NetLogo. Even when  
340 another platform is chosen for a particular model, NetLogo is a very efficient tool for  
341 prototyping and replicating (e.g., for code verification) parts or all of an agent-model.

342 The vision described in the Introduction, that all models would be described in a format such  
343 as ODD and implemented in a generic platform and programming language that is easy to  
344 learn, understand and execute, is in fact no longer a vision but to a large extent already a  
345 reality. ODD and NetLogo as standard tools have the potential to make modelling more  
346 coherent and efficient, to promote theory development by starting more projects from  
347 existing models rather than from scratch, and to make modelling more transparent and  
348 reliable and thus better suited to supporting decision making (Grimm et al., 2020a).

349 Both ODD and NetLogo are proof of concept that standards are indeed possible and  
350 necessary to improve modelling practice and lead to more coherence. Different models  
351 addressing the same question often produce incompatible predictions because the  
352 modellers have different preferences and backgrounds and use different expert opinions,  
353 calibration data and decision contexts (Alexander et al., 2017; Bahlburg et al., 2023; Banitz et  
354 al., 2022; Grimm, 2023). Standards like ODD do **not** resolve these differences per se, but they  
355 help making them explicit so that models and their underlying assumptions can be  
356 systematically compared and the model outcomes integrated (Wimmler et al., 2024).  
357 Standards are key to improving and establishing Good Modelling Practice, making models  
358 more fit for purpose (Hamilton et al., 2022) and aligning them with FAIR principles (Barton et

359 al., 2022). Therefore, the Open Modelling Foundation (OMF<sup>2</sup>), with more than 40 modelling  
 360 organisations as members, is currently working on the development and establishment of  
 361 standards covering all aspects of modelling and model applications.

## 362 Acknowledgements

363 This work was partially funded by the Center for Advanced Systems Understanding (CASUS),  
 364 which is financed by Germany's Federal Ministry of Education and Research (BMBF) and by  
 365 the Saxon Ministry for Science, Culture and Tourism (SMWK) with tax funds on the basis of  
 366 the budget approved by the Saxon State Parliament. **We would like to thank two anonymous**  
 367 **reviewers for their insightful comments.**

## 368 Supplementary material

369 S1 contains the reports on all replication  
 370 exercises: <https://doi.org/10.5281/zenodo.13983991> (Zenodo)

371 S2 contains the corresponding NetLogo programs, both the original versions in NetLogo 4.0.5  
 372 and a versions in NetLogo 6.4: <https://doi.org/10.5281/zenodo.13984010> (Zenodo)

## 373 References

- 374 Alexander, P., Prestele, R., Verburg, P.H., Arneth, A., Baranzelli, C., Batista e Silva, F., Brown,  
 375 C., Butler, A., Calvin, K., Dendoncker, N., 2017. Assessing uncertainties in land cover  
 376 projections. *Global change biology* 23, 767-781.
- 377 Antosz, P., Birks, D., Edmonds, B., Heppenstall, A., Meyer, R., Polhill, J.G., O'Sullivan, D.,  
 378 Wijermans, N., 2023. What do you want theory for?-A pragmatic analysis of the roles of  
 379 "theory" in agent-based modelling. *Environmental Modelling & Software* 168, 105802.
- 380 Antosz, P., Shults, F.L., Puga-Gonzalez, I., Szczepanska, T., 2022. HUM-e: An Emotive-Socio-  
 381 cognitive Agent Architecture for Representing Human Decision-Making in Anxiogenic  
 382 Contexts, Conference of the European Social Simulation Association. Springer, pp. 235-247.
- 383 Axelrod, R., 1997. The dissemination of culture - A model with local convergence and global  
 384 polarization. *Journal of Conflict Resolution* 41, 203-226.
- 385 Axtell, R., Axelrod, R., Epstein, J.M., Cohen, M.D., 1996. Aligning simulation models: A case  
 386 study and results. *Computational & mathematical organization theory* 1, 123-141.
- 387 Ayllón, D., Railsback, S.F., Gallagher, C., Augusiak, J., Baveco, H., Berger, U., Charles, S.,  
 388 Martin, R., Focks, A., Galic, N., Liu, C., van Loon, E.E., Nabe-Nielsen, J., Piou, C., Polhill, J.G.,  
 389 Preuss, T.G., Radchuk, V., Schmolke, A., Stadnicka-Michalak, J., Thorbek, P., Grimm, V., 2021.

---

<sup>2</sup> <https://www.openmodelingfoundation.org/>

- 390 Keeping modelling notebooks with TRACE: Good for you and good for environmental  
391 research and management support. *Environmental Modelling & Software* 136, 104932.
- 392 Ayllón, D., Railsback, S.F., Vincenzi, S., Groeneveld, J., Almodóvar, A., Grimm, V., 2016.  
393 InSTREAM-Gen: Modelling eco-evolutionary dynamics of trout populations under  
394 anthropogenic environmental change. *Ecological Modelling* 326, 36-53.
- 395 Bahlburg, D., Thorpe, S.E., Meyer, B., Berger, U., Murphy, E.J., 2023. An intercomparison of  
396 models predicting growth of Antarctic krill (*Euphausia superba*): The importance of  
397 recognizing model specificity. *Plos one* 18, e0286036.
- 398 Banitz, T., Schlüter, M., Lindkvist, E., Radosavljevic, S., Johansson, L.-G., Ylikoski, P., Martínez-  
399 Peña, R., Grimm, V., 2022. Model-derived causal explanations are inherently constrained by  
400 hidden assumptions and context: The example of Baltic cod dynamics. *Environmental*  
401 *Modelling & Software* 156, 105489.
- 402 Becher, M.A., Grimm, V., Thorbek, P., Horn, J., Kennedy, P.J., Osborne, J.L., 2014. BEEHAVE: a  
403 systems model of honeybee colony dynamics and foraging to explore multifactorial causes of  
404 colony failure. *Journal of Applied Ecology* 51, 470-482.
- 405 Carney, M., Davies, B., 2020. Agent-based modeling, scientific reproducibility, and  
406 taphonomy: A successful model implementation case study. *Journal of Computer*  
407 *Applications in Archaeology* 3, 182-196.
- 408 Davis, J.N., Todd, P.M., Bullock, S., 1999. Environment quality predicts parental provisioning  
409 decisions. *Proceedings of the Royal Society of London. Series B: Biological Sciences* 266,  
410 1791-1797.
- 411 DeAngelis, D.L., Cox, D.K., Coutant, C.C., 1979. Cannibalism and size dispersal in young-of-  
412 the-year largemouth bass: experiment and model. *Ecol. Modell* 8, 133-148.
- 413 Deffuant, G., Amblard, F., Weisbuch, G., Faure, T., 2002. How can extremism prevail? A study  
414 based on the relative agreement interaction model. *JASSS-The Journal of Artificial Societies*  
415 *and Social Simulation* 5.
- 416 Donkin, E., Dennis, P., Ustalakov, A., Warren, J., Clare, A., 2017. Replicating complex agent  
417 based models, a formidable task. *Environmental Modelling & Software* 92, 142-151.
- 418 Edmonds, B., Hales, D., 2003. Replication, replication and replication: Some hard lessons  
419 from model alignment. *Journal of Artificial Societies and Social Simulation* 6.
- 420 Fielding, D.J., 2004. Intraspecific competition and spatial heterogeneity alter life history  
421 traits in an individual-based model of grasshoppers\*. *Ecological Modelling* 175, 169-187.
- 422 Filazzola, A., Cahill Jr, J.F., 2021. Replication in field ecology: Identifying challenges and  
423 proposing solutions. *Methods in Ecology and Evolution* 12, 1780-1792.
- 424 Fraser, H., Barnett, A., Parker, T.H., Fidler, F., 2020. The role of replication studies in ecology.  
425 *Ecology and Evolution* 10, 5197-5207.
- 426 Gallagher, C.A., Grimm, V., Kyhn, L.A., Kinze, C.C., Nabe-Nielsen, J., 2021. Movement and  
427 seasonal energetics mediate vulnerability to disturbance in marine mammal populations.  
428 *The American Naturalist* 197, 296-311.

- 429 Ginovart, M., López, D., Valls, J., Silbert, M., 2002. Individual based simulations of bacterial  
430 growth on agar plates. *Physica A: Statistical Mechanics and its Applications* 305, 604-618.
- 431 Grimm, V., 2023. Ecology needs to overcome siloed modelling. *Trends in Ecology &*  
432 *Evolution*.
- 433 Grimm, V., Augusiak, J., Focks, A., Frank, B.M., Gabsi, F., Johnston, A.S.A., Liu, C., Martin, B.T.,  
434 Meli, M., Radchuk, V., Thorbek, P., Railsback, S.F., 2014. Towards better modelling and  
435 decision support: documenting model development, testing, and analysis using TRACE.  
436 *Ecological Modelling* 280, 129-139.
- 437 Grimm, V., Berger, U., 2016. Robustness analysis: Deconstructing computational models for  
438 ecological theory and applications. *Ecological Modelling* 326, 162-167.
- 439 Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand,  
440 T., Heinz, S., Huse, G., Huth, A., Jepsen, J.U., Jørgensen, C., Mooij, W.M., Müller, B., Pe'er, G.,  
441 Piou, C., Railsback, S.F., Robbins, A.M., Robbins, M.M., Rossmannith, E., Rüger, N., Strand, E.,  
442 Souissi, S., Stillman, R.A., Vabø, R., Visser, U., DeAngelis, D.L., 2006. A standard protocol for  
443 describing individual-based and agent-based models. *Ecol. Modell* 198, 115-126.
- 444 Grimm, V., Berger, U., DeAngelis, D.L., Polhill, J.G., Giske, J., Railsback, S.F., 2010. The ODD  
445 protocol: a review and first update. *Ecol. Modell* 221, 2760-2768.
- 446 Grimm, V., Berger, U., Meyer, M., Lorscheid, I., 2024. Theory for and from agent-based  
447 modelling: Insights from a virtual special issue and a vision. *Environmental Modelling &*  
448 *Software* 178, 106088.
- 449 Grimm, V., Johnston, A.S., Thulke, H.-H., Forbes, V., Thorbek, P., 2020a. Three questions to  
450 ask before using model outputs for decision support. *Nature communications* 11, 1-3.
- 451 Grimm, V., Railsback, S.F., Vincenot, C.E., Berger, U., Gallagher, C., DeAngelis, D.L., Edmonds,  
452 B., Ge, J., Giske, J., Groeneveld, J., Johnston, A.S.A., Milles, A., Nabe-Nielsen, J., Polhill, J.G.,  
453 Radchuk, V., Rohwäder, M.-S., Stillman, R.A., Thiele, J.C., Ayllón, D., 2020b. The ODD  
454 protocol for describing agent-based and other simulation models: A second update to  
455 improve clarity, replication, and structural realism. *Journal of Artificial Societies and Social*  
456 *Simulation* 23.
- 457 Groeneveld, J., Martinovič, T., Rossi, T., Salamon, O., Sara-aho, K., Grimm, V., 2024.  
458 Prototype Biodiversity Digital Twin: honey bees in agricultural landscapes. *Research Ideas*  
459 *and Outcomes* 10, e125167.
- 460 Gürçan, Ö., Szczepanska, T., Antosz, P., 2023. A Guide to Re-implementing Agent-Based  
461 Models: Experiences from the HUMAT Model, Conference of the European Social Simulation  
462 Association. Springer, pp. 519-531.
- 463 Hales, D., Rouchier, J., Edmonds, B., 2003. Model-to-model analysis. *Journal of Artificial*  
464 *Societies and Social Simulation* 6.
- 465 Hansen, F., Jeltsch, F., Tackmann, K., Staubach, C., Thulke, H.-H., 2004. Processes leading to a  
466 spatial aggregation of *Echinococcus multilocularis* in its natural intermediate host *Microtus*  
467 *arvalis*. *International journal for parasitology* 34, 37-44.
- 468 Hauert, C., De Monte, S., Hofbauer, J., Sigmund, K., 2002. Volunteering as red queen  
469 mechanism for cooperation in public goods games. *Science* 296, 1129-1132.

- 470 Hauert, C., Doebeli, M., 2004. Spatial structure often inhibits the evolution of cooperation in  
471 the snowdrift game. *Nature* 428, 643-646.
- 472 Hauke, J., Achter, S., Meyer, M., 2020. Theory development via replicated simulations and  
473 the added value of standards. *The journal of artificial societies and social simulation* 23.
- 474 Jackson, A.L., Ruxton, G.D., Houston, D.C., 2008. The effect of social facilitation on foraging  
475 success in vultures: a modelling study. *Biology Letters* 4, 311-313.
- 476 Jakeman, A.J., Elsworth, S., Wang, H.-H., Hamilton, S.H., Melsen, L., Grimm, V., 2024. Towards  
477 normalizing good practice across the whole modeling cycle: its instrumentation and future  
478 research topics. *Socio-Environmental Systems Modelling* 6, 18755-18755.
- 479 Kerr, B., Riley, M.A., Feldman, M.W., Bohannan, B.J.M., 2002. Local dispersal promotes  
480 biodiversity in a real-life game of rock-paper-scissors. *Nature* 418, 171-174.
- 481 Müller, B., Balbi, S., Buchmann, C.M., de Sousa, L., Dressler, G., Groeneveld, J., Klassert, C.J.,  
482 Le, Q.B., Millington, J.D.A., Nolzen, H., Parker, D.C., Polhill, J.G., Schlüter, M., Schulze, J.,  
483 Schwarz, N., Sun, Z., Taillandier, P., Weise, H., 2014. Standardised and transparent model  
484 descriptions for agent-based models: Current status and prospects. *Environmental  
485 Modelling & Software* 55, 156-163.
- 486 North, M.J., Collier, N.T., Ozik, J., Tataru, E.R., Macal, C.M., Bragan, M., Sydelko, P., 2013.  
487 Complex adaptive systems modeling with Repast Symphony. *Complex adaptive systems  
488 modeling* 1, 1-26.
- 489 O'Keefe, K.J., 2005. The evolution of virulence in pathogens with frequency-dependent  
490 transmission. *Journal of Theoretical Biology* 233, 55-64.
- 491 Papert, S., 1985. Different visions of logo. *Computers in the Schools* 2, 3-8.
- 492 Parker, T.H., Forstmeier, W., Koricheva, J., Fidler, F., Hadfield, J.D., Chee, Y.E., Kelly, C.D.,  
493 Gurevitch, J., Nakagawa, S., 2016. Transparency in ecology and evolution: real problems, real  
494 solutions. *Trends in Ecology & Evolution* 31, 711-719.
- 495 Polhill, J.G., Brown, D.G., Grimm, V., 2008. Using the ODD protocol for describing three  
496 agent-based social simulation models of land use change. *JASSS-The Journal of Artificial  
497 Societies and Social Simulation* 11, <http://jasss.soc.surrey.ac.uk/11/12/13.html>.
- 498 Railsback, S.F., Ayllón, D., Berger, U., Grimm, V., Lytinen, S.L., Sheppard, C.J.R., Thiele, J.C.,  
499 2017. Improving execution speed of models implemented in NetLogo. *Journal of Artificial  
500 Societies and Social Simulation* 20, art. 3.
- 501 Ratz, A., 1995. Long-term spatial patterns created by fire: a model oriented towards boreal  
502 forests. *International Journal of Wildland Fire* 5, 25-34.
- 503 Schmolke, A., Thorbek, P., DeAngelis, D.L., Grimm, V., 2010. Ecological models supporting  
504 environmental decision making: a strategy for the future. *Trends Ecol. Evol* 25, 479-486.
- 505 Silvertown, J., Holtier, S., Johnson, J., Dale, P., 1992. Cellular automaton models of  
506 interspecific competition for space--the effect of pattern on process. *Journal of Ecology*, 527-  
507 533.
- 508 Thiele, J.C., Grimm, V., 2015. Replicating and breaking models: good for you and good for  
509 ecology. *Oikos* 124, 691-696.

- 510 Thiery, J.M., D'Herbes, J.M., Valentin, C., 1995. A model simulating the genesis of banded  
511 vegetation patterns in Niger. *J. Ecol* 83, 497-507.
- 512 Vincenot, C.E., 2018. How new concepts become universal scientific approaches: insights  
513 from citation network analysis of agent-based complex systems science. *Proceedings of the*  
514 *Royal Society B-Biological Sciences* 285.
- 515 Warszawski, L., Frieler, K., Huber, V., Piontek, F., Serdeczny, O., Schewe, J., 2014. The inter-  
516 sectoral impact model intercomparison project (ISI-MIP): project framework. *Proceedings of*  
517 *the National Academy of Sciences* 111, 3228-3232.
- 518 Weiner, J., Stoll, P., Muller-Landau, H., Jasentuliyana, A., 2001. The effects of density, spatial  
519 pattern, and competitive symmetry on size variation in simulated plant populations.  
520 *American Naturalist* 158, 438-450.
- 521 Wijermans, N., Scholz, G., Chappin, É., Heppenstall, A., Filatova, T., Polhill, J.G., Semeniuk, C.,  
522 Stöppler, F., 2023. Agent decision-making: The Elephant in the Room-Enabling the  
523 justification of decision model fit in social-ecological models. *Environmental Modelling &*  
524 *Software* 170, 105850.
- 525 Wilensky, U., 1999. *NetLogo*. Northwestern University, Center for Connected Learning and  
526 Computer-Based Modeling, Evanston.
- 527 Wilensky, U., Rand, W., 2015. *An introduction to agent-based modeling: modeling natural,*  
528 *social, and engineered complex systems with NetLogo*. MIT press.
- 529 Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A.,  
530 Blomberg, N., Boiten, J.-W., da Silva Santos, L.B., Bourne, P.E., 2016. The FAIR Guiding  
531 Principles for scientific data management and stewardship. *Scientific data* 3, 1-9.
- 532 Wimmeler, M.-C., Bathmann, J., Vollhüter, J., Berger, U., 2024. *pyMANGA: A modular, open*  
533 *and extendable software platform for modeling of forest and vegetation dynamics.*  
534 *Environmental Modelling & Software*, 105973.
- 535 Wissel, C., 1992. Modelling the mosaic cycle of a Middle European beech forest. *Ecological*  
536 *Modelling* 63, 29-43.
- 537 Zhang, J., Robinson, D.T., 2021. Replication of an agent-based model using the replication  
538 standard. *Environmental Modelling & Software* 139, 105016.
- 539 Zhong, W., Kim, Y., 2010. Using model replication to improve the reliability of agent-based  
540 models, *Advances in Social Computing: Third International Conference on Social Computing,*  
541 *Behavioral Modeling, and Prediction, SBP 2010, Bethesda, MD, USA, March 30-31, 2010.*  
542 *Proceedings* 3. Springer, pp. 118-127.

543