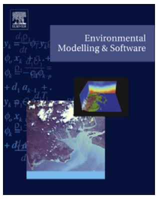
Environmental Modelling & Software 92 (2017) 142–151

Contents lists available at ScienceDirect

Environmental Modelling & Software

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



Replicating complex agent based models, a formidable task

Elizabeth Donkin [a,*], Peter Dennis [a], Andrey Ustalakov [b,1], John Warren [a,2], Amanda Clare [b]

[a] Institute of Biological Environmental and Rural Sciences (IBERS), Aberystwyth University, Penglais Campus, SY23 3DA, UK
[b] Department of Computer Science, Aberystwyth University, Penglais Campus, SY23 3DB, UK

ARTICLE INFO

Article history:
Received 1 December 2016
Received in revised form
20 January 2017
Accepted 27 January 2017

Keywords:
Agent-based model
Individual-based model
Model replication
NetLogo
Open code
Repast

ABSTRACT

Promoting replication of models is unarguably a positive step for agent based modelling, as replication promotes rigorous testing. Model replication remains rare, yet is vital to assessing the repeatability of existing agent based models. Notably, more work is needed to assess cross platform and language replication, which represent potential sources of variability between model results. An existing, complex agent based model was replicated using two widely used platforms (NetLogo and Repast). When results generated by the models were compared, the findings differed not only in magnitude but the trends produced by the data, resulting in different conclusions being drawn from each set of model predictions. The variation between the models is believed to be a result of the complexity of encoding a substantial theoretical model in particular programming languages. This highlights the express need to document replication of existing models in order to fully understand the potential limitations to replication.

© 2017 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Software availability

Name of the Model: Repast Model (plantInsectPolyculture)
Developers: Andrey Ustalakov, CA Technologies, Oxford, UK; Elizabeth Donkin, IBERS, Aberystwyth University, UK.
Software Required: Repast Simphony 2.2.0
Program Language: JAVA
Code Availability: https://github.com/lizzydonkin/plantInsectPolyculture
Platform Availability: free download http://repast.sourceforge.net/download.html
Name of the Model: NetLogo Model (plantInsectPolycultureNetLogo)
Developer: Elizabeth Donkin, IBERS, Aberystwyth University, UK.
Software Required: NetLogo 5.1.0
Program Language: NetLogo
Code Availability: https://github.com/lizzydonkin/plantInsectPolycultureNetLogo
Platform Availability: free download https://ccl.northwestern.edu/netlogo/download.shtml

1. Introduction

Agent based models (ABM) are widespread throughout the biological, ecological and environmental sciences (Bithell and Brasington, 2009; Bradhurst et al., 2016; Grimm, 1999; Grimm et al., 2005; Matthews et al., 2007). They provide valuable insight into often complex agent based systems and have become an indispensable research tool (Thiele and Grimm, 2015). Despite the value of ABMs, developed models are rarely repeatedly applied. The majority of researchers develop models from scratch rather than testing and building on the work of others (Thiele and Grimm, 2015; Wilenksy and Rand, 2007).

The importance of developing a culture of model replication in the sciences is an idea gaining traction (Axtell et al., 1996; Collins et al., 2015; Thiele and Grimm, 2015; Wilenksy and Rand, 2007). Replication exposes models to external scrutiny and facilitates robustness analysis that is rarely implemented by the original model developers. This results in more reliable models that have a body of literature supporting their predictions (Axelrod, 1997; Axtell et al., 1996; Easterbrook, 2014; Thiele and Grimm, 2015).

Replication and reuse of models is unarguably a positive

---

* Corresponding author.
E-mail addresses: eld38@aber.ac.uk, donkin.e@gmail.com (E. Donkin).
[1] Present address: 11 Oasis Business Park, Eynsham, OX29 4TP, UK.
[2] Present address: University of Natural Resources and the Environment, Vudal Campus, East New Britain, Papua New Guinea.

http://dx.doi.org/10.1016/j.envsoft.2017.01.020
1364-8152/© 2017 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

development for agent based modelling (Easterbrook, 2014; Thiele and Grimm, 2015). However, in the current research environment there are complications hampering a wide spread adoption of model replication. Agent based models representing natural systems can be, and often are, highly complex (Evans et al., 2013; Sun et al., 2016; Thiele and Grimm, 2015). To facilitate replication the documentation associated with complex models needs to be detailed, and source code made freely available. This level of information is often lacking for published ABMs and although detailed frameworks for model descriptions exist and repositories for code have been established, models are still published without this information (Rollins et al., 2014; Thiele and Grimm, 2015).

As the importance of replication is realised, there is a real need to address whether it is practically possible to replicate a complex model from the information typically provided. There is also a real need to address how and when replication can and should be implemented, and highlight potential pitfalls (Axtell et al., 1996; Hales et al., 2003). One such complication can be cross platform and language replication. In an ideal world, every researcher in agent based modelling would adopt the same programming language and development platform. However, this is unrealistic. Researchers will adopt a method of programming and developing ABMs that suits their ability and resources. Programming languages and technology evolve relatively quickly and current languages may not be the most popular, effective or practical in the near future (Meyerovich and Rabkin, 2013).

There are few publications documenting ABM replication across platforms and programming languages, or indeed documenting model replication at all. Those that have been published have dealt with relatively simple models (Axtell et al., 1996; Bajracharya and Duboz, 2013; Railsback et al., 2006; Wilenksy and Rand, 2007). Replication of these simple models across platforms has been found to be problematic in practical terms, making the process of replication a lengthy and complex task (Wilenksy and Rand, 2007). However, of greater consequence is the indication that the process can introduce variation in results through disruption of the original model design when made to fit a different language (Bajracharya and Duboz, 2013). This could potentially prevent the successful replication of complex models, as investigating all possible interacting consequences of coding choices would be a near impossible task.

Here we attempt replication and testing of a previously published agro-ecological, agent based model. The model simulates pest insects within an agricultural setting. Insect behaviour is regularly modelled using ABM, due to the ability to incorporate fine scale movement behaviour (Almeida et al., 2010; Perez and Dragicevic, 2010). This particular model was chosen as relevant to our research, and we wished to adapt it to further explore the parameter space. However, the model is typical of many complex ABMs simulating multi-species interactions. We replicated the model based on the information provided in the published article, using two established agent based modelling platforms. The process of replication is discussed, with particular focus on the potential pitfalls. Source code for the two reproduced models are presented for further use and development.

2. Materials and methods

An agent based model, designed and implemented by Potting et al. (2005) was recreated in two agent based modelling platforms: Repast Simphony 2.2.0 and NetLogo 5.1.0. The original model was developed in Visual Basic and the original source code is not available (R Potting, personal communication). To assess how accurately the model could be replicated from the existing information, the original model was recreated as closely as possible using NetLogo, with only the information provided in Potting et al. (2005). To assess the potential for translational differences between models the NetLogo version of the model was recreated in Java within Repast Simphony, using the NetLogo code and description. The model replications were implemented by the same authors.

Repast Simphony is a group of free and open source modelling platforms, that make use of inter-language libraries (North et al., 2013). The Repast environment provides a graphic user interface whilst users develop models in a general purpose programming language such as Java, as was used in our replication. NetLogo is also a free and open source simulation environment that utilises a programming language, also named NetLogo, designed specifically for developing ABMs (Tisue and Wilensky, 2004). Models are developed in a graphical environment where the user controls individual agents, termed 'turtles' in the NetLogo literature, in a grid environment of patches (for detailed reviews of the platforms see: Tisue and Wilensky, 2004; North et al., 2013).

The original model description is available in Potting et al. (2005). Our interpreted, replicated NetLogo and Repast version is described below using the ODD protocol outlined by Grimm et al. (Grimm et al., 2010, 2006). Assumptions and discrepancies with the original description are outlined.

2.1. Model description

2.1.1. Overview

2.1.1.1. Purpose. The purpose of this model is to understand how insect behavioural ecology, in regards to foraging ability and behaviour of different pest insect species, could impact the efficacy of polyculture planting strategies for preventing damage to crops from pest insect species.

2.1.1.2. State variables and scales. The model simulates time as discrete time steps with a fixed duration. The model environment is comprised of a grid of 100 × 100 cells with a fixed location, each cell representing a plant of a certain type. Each grid cell, or plant, can be occupied by an infinite number of insects, or remain unoccupied. Upon model initialisation the environment is created with a given proportion of each plant type and a given spatial arrangement.

The plant types are distinguished by the variable 'plant specific flight tendency' and whether their quality can be detected by an insect in flight. 'Plant specific flight tendency' describes the probability of an insect leaving the plant to forage or remaining and feeding. Low values of 'plant specific flight tendency' describe a favourable plant and vice versa. Plant types are defined in Table 1. State variables for the global environment, insects and plants are summarised in Table 2. The three possible spatial arrangements of the crop and non-crop plants are described in Fig. 1.

Insects within the model can have one of three search strategies: visual, olfactory or contact. The starting density of the population is 500 insects, all of which are univoltine and are identical in their search strategy.

2.1.1.3. Process overview and scheduling. The following processes

Table 1
Plant types and their corresponding values for 'plant specific flight tendency'.

Plant type	Plant specific flight tendency	Detected in flight
Favourable		
crop	0.5	yes
trap crop	0.05	yes
Unfavourable		
deterrent	1	no
repellent	1	yes

Download English Version:

https://daneshyari.com/en/article/4978146

Download Persian Version:

https://daneshyari.com/article/4978146

Daneshyari.com