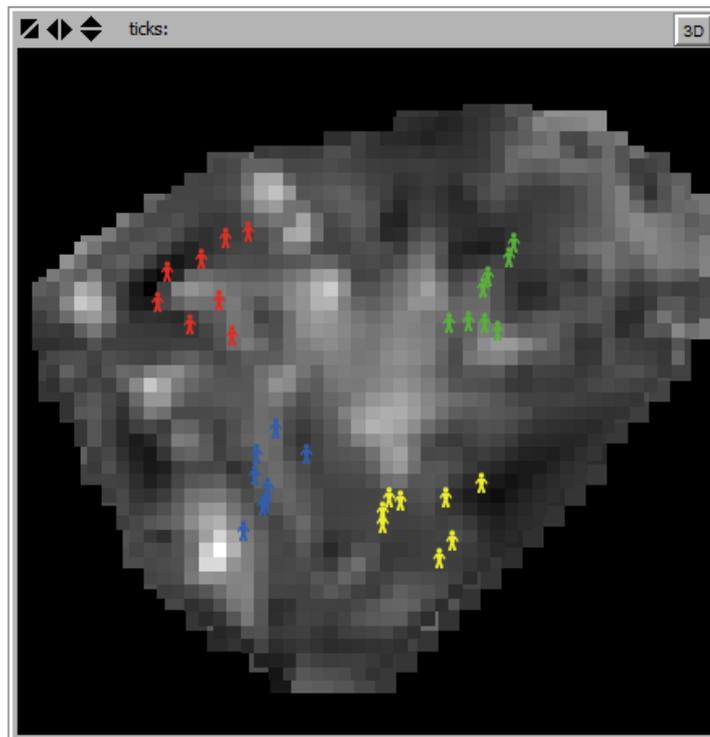




**British  
Geological Survey**  
NATURAL ENVIRONMENT RESEARCH COUNCIL

# Agent Based Modelling: Initial assessment for use on soil bioaccessibility

Environmental Modelling Programme  
Internal Report OR/16/022





# Agent Based Modelling: Initial assessment for use on soil bioaccessibility

D Beriro; M Cave; J Wragg; A Hughes

The National Grid and other Ordnance Survey data © Crown Copyright and database rights 2016. Ordnance Survey Licence No. 100021290 EUL.

## *Keywords*

Agent Based Model; Reast-simphony; NetLogo; Insight-maker; Northampton.

## *Front cover*

Output from test e NetLogo model for Northampton.

## *Bibliographical reference*

D BERIRO; M CAVE; J WRAGG; A Hughes.. 2016. Agent Based Modelling: Initial assessment for use on soil bioaccessibility. *British Geological Survey Internal Report*, OR/16/022. 33pp.

Copyright in materials derived from the British Geological Survey's work is owned by the Natural Environment Research Council (NERC) and/or the authority that commissioned the work. You may not copy or adapt this publication without first obtaining permission. Contact the BGS Intellectual Property Rights Section, British Geological Survey, Keyworth, e-mail [ipr@bgs.ac.uk](mailto:ipr@bgs.ac.uk). You may quote extracts of a reasonable length without prior permission, provided a full acknowledgement is given of the source of the extract.

Maps and diagrams in this book use topography based on Ordnance Survey mapping.

## BRITISH GEOLOGICAL SURVEY

The full range of our publications is available from BGS shops at Nottingham, Edinburgh, London and Cardiff (Welsh publications only) see contact details below or shop online at [www.geologyshop.com](http://www.geologyshop.com)

The London Information Office also maintains a reference collection of BGS publications, including maps, for consultation.

We publish an annual catalogue of our maps and other publications; this catalogue is available online or from any of the BGS shops.

*The British Geological Survey carries out the geological survey of Great Britain and Northern Ireland (the latter as an agency service for the government of Northern Ireland), and of the surrounding continental shelf, as well as basic research projects. It also undertakes programmes of technical aid in geology in developing countries.*

*The British Geological Survey is a component body of the Natural Environment Research Council.*

*British Geological Survey offices*

### **BGS Central Enquiries Desk**

Tel 0115 936 3143 Fax 0115 936 3276  
email [enquiries@bgs.ac.uk](mailto:enquiries@bgs.ac.uk)

### **Environmental Science Centre, Keyworth, Nottingham NG12 5GG**

Tel 0115 936 3241 Fax 0115 936 3488  
email [sales@bgs.ac.uk](mailto:sales@bgs.ac.uk)

### **Murchison House, West Mains Road, Edinburgh EH9 3LA**

Tel 0131 667 1000 Fax 0131 668 2683  
email [scotsales@bgs.ac.uk](mailto:scotsales@bgs.ac.uk)

### **Natural History Museum, Cromwell Road, London SW7 5BD**

Tel 020 7589 4090 Fax 020 7584 8270  
Tel 020 7942 5344/45 email [bgs london@bgs.ac.uk](mailto:bgs london@bgs.ac.uk)

### **Columbus House, Greenmeadow Springs, Tongwynlais, Cardiff CF15 7NE**

Tel 029 2052 1962 Fax 029 2052 1963

### **Maclean Building, Crowmarsh Gifford, Wallingford OX10 8BB**

Tel 01491 838800 Fax 01491 692345

### **Geological Survey of Northern Ireland, Department of Enterprise, Trade & Investment, Dundonald House, Upper Newtownards Road, Ballymiscaw, Belfast, BT4 3SB**

Tel 028 9038 8462 Fax 028 9038 8461

[www.bgs.ac.uk/gsni/](http://www.bgs.ac.uk/gsni/)

### *Parent Body*

### **Natural Environment Research Council, Polaris House, North Star Avenue, Swindon SN2 1EU**

Tel 01793 411500 Fax 01793 411501  
[www.nerc.ac.uk](http://www.nerc.ac.uk)

Website [www.bgs.ac.uk](http://www.bgs.ac.uk)

Shop online at [www.geologyshop.com](http://www.geologyshop.com)

# Contents

<b>Contents.....</b>	<b>i</b>
<b>Summary .....</b>	<b>ii</b>
<b>1 Introduction.....</b>	<b>3</b>
<b>2 What is ABM? .....</b>	<b>3</b>
<b>3 Software trialled .....</b>	<b>5</b>
3.1 Repast-simphony .....	5
3.2 NetLogo .....	8
3.3 Insight-maker.....	15
<b>4 Recommendations for future work.....</b>	<b>19</b>
4.1 Summary and next steps .....	19
4.2 Details of the example to be developed.....	19
<b>Glossary.....</b>	<b>21</b>
<b>References .....</b>	<b>22</b>
<b>Appendix 1 NETLOGO ASSOCIATED Scripts.....</b>	<b>23</b>
<b>Appendix 2 REPAST-SIMPHONY ASSOCIATED Scripts .....</b>	<b>27</b>

## FIGURES

Figure 1. Example of the flow zombie tutorial .....	6
Figure 2. Screenshot of the modified geography example.....	7
Figure 3. Example screen shot of a NetLogo model .....	10
Figure 4. Northampton As Bioaccessibility Read in to NetLogo .....	11
Figure 5. Northampton As exposure model in operation .....	12
Figure 6. Cumulative As exposure for each of the human agents in each of the four groups (coloured lines) and the mean cumulative exposure for each group (dashed black line).....	13
Figure 7. Box and whisker plot of the exposure for each of the 4 groups after 100 time steps ...	13
Figure 8. Probability density function plot of the exposure for each of the 4 groups after 100 time steps .....	14
Figure 9. Example model created using “primitives” and “links” .....	16
Figure 10. Examples of model output .....	17

## TABLES

Table 1. Sample of relevant environmental applications using ABM. ....	4
Table 2. SWOT analysis for repast-simphony .....	8

## Summary

This report describes the testing of Agent Based Modelling implementations in three different software packages: Repast-simphony, NetLogo and Insight-maker. These software have been evaluated against their capability to simulate the exposure of people as agents moving across Arsenic contaminated soils. Two of the three tested software (Repast-simphony and NetLogo) are recommended for assessment on more complex problems. An outline work plan is presented for future work.

# 1 Introduction

BGS holds a significant number of spatial datasets including soil geochemistry for England. BGS also has a track record in measuring and modelling the human ingestion bioaccessibility of potentially harmful substances (PHS) in soil, in particular undertaking laboratory experiments to simulate the human digestive system (*in vitro*). Bioaccessibility is an estimate of the amount of PHS released from soil once ingested (a common and accepted characteristic of children who play in gardens and parks). This is important because human health risk assessments for the presence of natural or anthropogenic PHS in soil are overly conservative, which can lead to unnecessary costly remediation as part of land use planning and redevelopment. Consequently, BGS research on the bioaccessibility of PHS has led to an internationally recognised body of knowledge on the fate and transport of PHS in soil and in the human body. However it is felt that there are limitations to this work and it would benefit from being able to “mobilise” the digestive system, i.e. move people (and their stomachs) around a contaminated landscape so that PHS intake and uptake and its interaction with the environment can be modelled. The ability to move people as receptors around a landscape has the potential to improve the understanding of the impact of decision-making with respect to contaminated land and its use and clean up.

The activity reported here is part of the Integrated Modelling and Uncertainty team’s tasks during FY2015-6. The work is designed to answer the question “Is Agent Based Modelling (ABM) useful for extending the work of bioaccessibility and environmental interaction?”. In order to explore the potential of ABM to support research on the human ingestion bioaccessibility of PHS in soil, three software applications for undertaking ABM have been trialled (NetLogo, repast-simphony and insight-maker) on a simple example and assessed for further work.

## 2 What is ABM?

Defined as “actions and interactions of autonomous agents with a view to assessing their effects of the system as a whole”. “Agents have behaviours, often described by simple rules, and interactions with other agents, which in turn influence their behaviours” (Macal and North, 2010). Put simply, it is the computational representation of things moving around a space. It radically differs from other modelling approaches in that the agents move around a landscape so have a position:  $x$ ,  $y$  and possibly  $z$ . They follow rules on how they move and, importantly, how they engage with each other and the environment in which they exist.

It has a wide range of uses where a behaviour or movement can be attributed to anything within a system. A recent paper (Farmer and Foley, 2009) has espoused ABM for use in economic modelling. One of the authors use ABM to simulate stock market trading by simulating different trading behaviours and types of institutions. They do admit that a global scale economic model would be beneficial, but an ambitious undertaking.

However ABM is not without its challenges, particularly for Socio-Economic systems (Filatova et al., 2013). Here the authors identify four challenges to successfully undertaking ABM of socio-economic systems: 1. Modelling agent’s behaviour; 2. Sensitivity analysis, verification and validation; 3. Coupling socio-demographic, ecological and biophysical models and 4. Spatial representation. These are all relevant issues to the work described below.

As we have seen ABM is not new, there have been a number of applications of agent based approaches over the last decades, and the approach itself has evolved. Table 1 summarises the most relevant applications to the environmental and human health field. The table also includes papers that are appropriate in terms of movement of humans.

**Table 1. Sample of relevant environmental applications using ABM.**

<b>Application</b>	<b>Summary</b>	<b>Reference/link</b>
Movement of people around streets and within buildings	Using different examples: Tate Gallery and Notting Hill Carnival the movement of people as agents is modelled. Different approaches for modelling movement such as stepwise and random walk are used.	Batty, 2003 <a href="http://www.casa.ucl.ac.uk/working_papers/paper61.pdf">http://www.casa.ucl.ac.uk/working_papers/paper61.pdf</a>
Fibrosis in the lung	Using ABM to replicate the degeneration of lung tissue from exposure to particulate matter.	Brown et al., 2011
Appropriate software for city-based movement	Three agent modelling software are evaluated to understand the most appropriate approach for simulating agent movement in cities; seven criteria are applied and three case studies are used to test the approach: evacuation, neighbourhood development and trip origin.	Crooks et al, 2008 <a href="http://discovery.ucl.ac.uk/15174/1/15174.pdf">http://discovery.ucl.ac.uk/15174/1/15174.pdf</a>
Car driver behaviour	The response of car drives to real-time information is studied using a hypothetical commute and by feeding in travel information and seeing their response.	Dia et al., 2002
Agent based pedestrian model	Development of the code STREETS which enables pedestrian movement to be simulated.	Schelhorn et al., 1999. <a href="http://www.casa.ucl.ac.uk/streets.pdf">http://www.casa.ucl.ac.uk/streets.pdf</a>
Farming	How different aspects of farming (technology, economics, environmental change and policy interventions) affect a community of farmers and their output.	Schreinemachers and Berger, 2011.
Water resource allocation	Linkage of a groundwater model to ABM systems is described and an example represented.	Castilla-Rho et al., 2015

## 3 Software trialled

### 3.1 REPAST-SIMPSONY

Repast-Simphony is an open source agent-based modelling tool kit, originally developed (North et al., 2013) in 2001. It operates using a modified Java IDE (Eclipse Luna) to implement agent-based modelling. Launched from Eclipse, Repast-Simphony is a separate visualisation module that is also capable of a range of user modifications e.g. access to Google Earth maps for GIS agent-based modelling. Repast Simphony is a cross platform Java-based modelling system that runs under Microsoft Windows, Apple Mac OS X, and Linux. It supports the development of extremely flexible models of interacting agents for use on workstations and computing clusters (using Logo-style C++ computer language). Repast Simphony models can be developed in several different forms including the ReLogo dialect of Logo, point-and-click statecharts, Groovy or Java all of which can be interleaved.

More information on Repast Simphony including documentation and downloads can be found at: [http://repast.sourceforge.net/repast\\_simphony.php](http://repast.sourceforge.net/repast_simphony.php). The use of the software in the literature suggests it has a moderate level of uptake (Scopus search 'repast AND "agent based"': 168 hits since 2013).

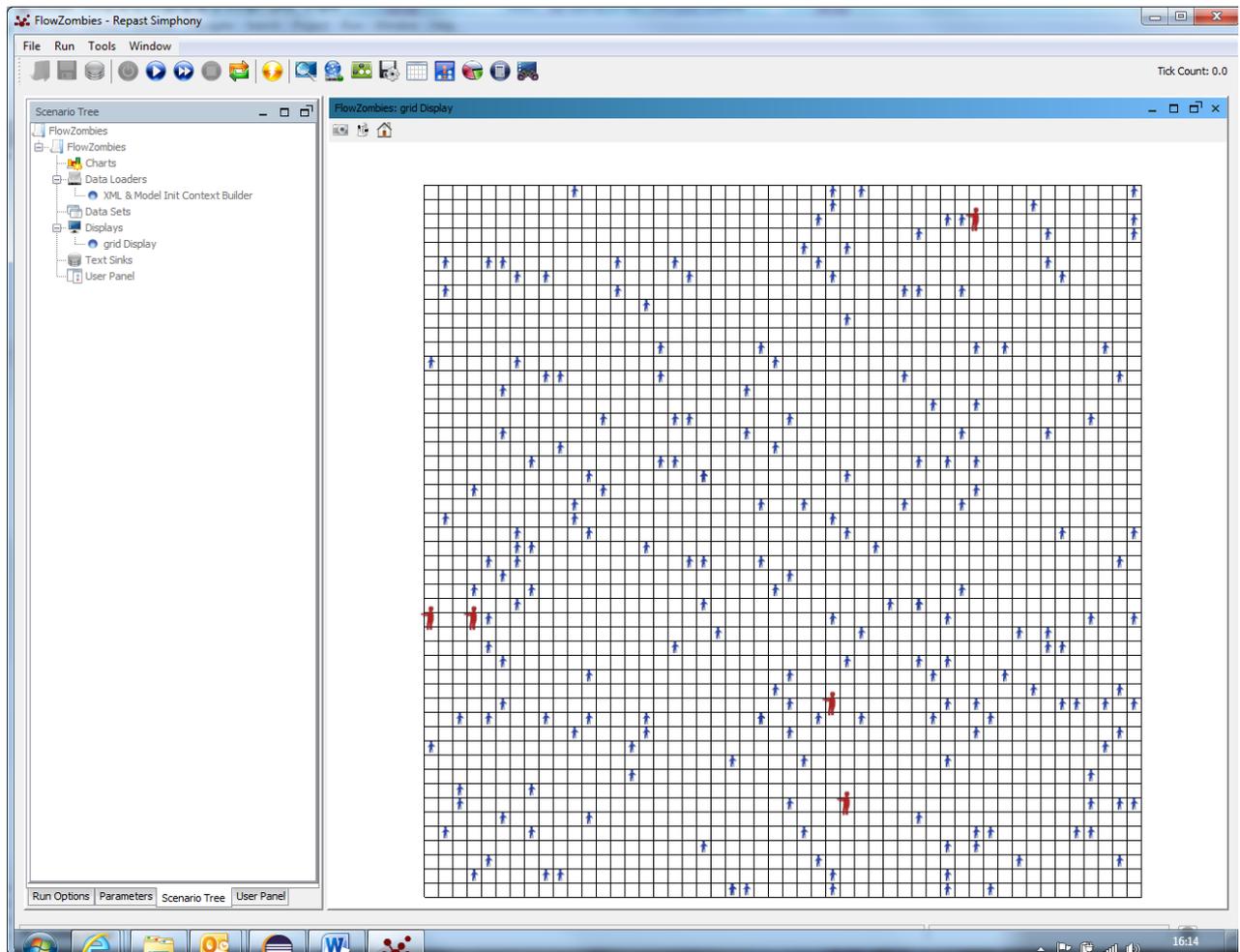
Familiarisation with Repast Simphony was achieved two ways: 1) implementation of a demonstration model (FlowZombies) using Groovy and Java based programming; and 2) implementation and customisation of a second demonstration model (Geography). The Zombie flow model was implemented directly from the following user guides downloaded with the software or in the documentation section on the website.

The FlowZombies example is simple agent-based model involving zombies chasing humans and humans running away from zombies. When Human energy levels are low they stop moving and the Zombie infects the Human and turn them into a Zombie. The Humans and Zombie are both agents.

The Groovy FlowZombie example was initiated using a user guide led implementation of flow chart programming where each item in the flow chart was assigned attributes. These Flow charts were initiated by Repast Simphony in its own Java-like language.

The Java FlowZombie example was initiated using a user guide led implementation of a Java programme which utilised a Repast Simphony API (a collection of function based libraries).

The output of both FlowZombie models is a gridded display of Zombie chasing and infecting humans (Figure 1).

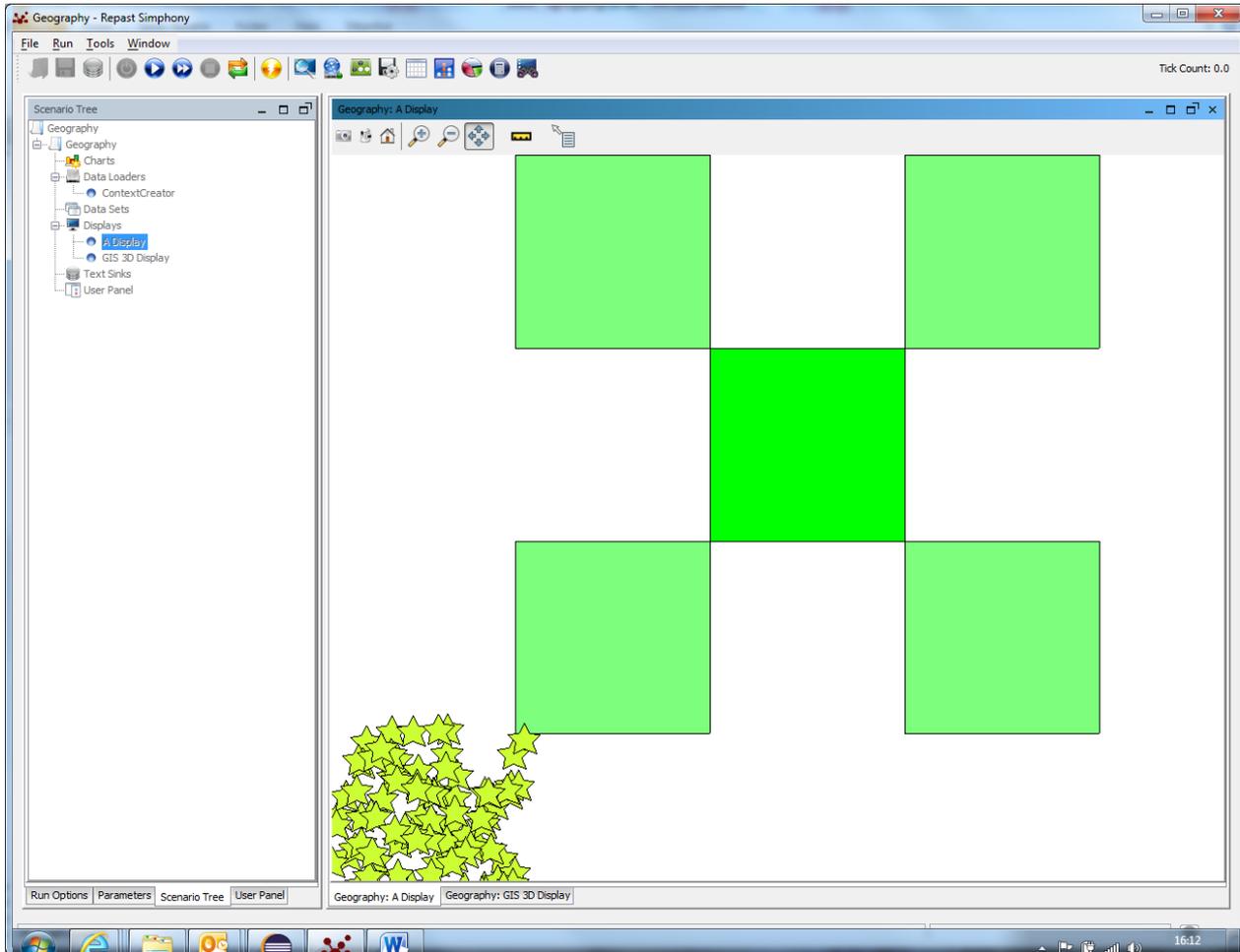


**Figure 1. Example of the flow zombie tutorial**

The secondary demonstration model, Geography has been developed. The Context Builder for this model describes it as:

*“...mobile GisAgents move around the Geography with a random motion and are represented by point locations. ZoneAgents are polygons that represent certain geographic areas. WaterLine agents represent water supply lines from Lake Michigan that supply the Chicago area. When a ZoneAgent intersects a WaterLine, the ZoneAgent will have access to fresh drinking water. GisAgents that are within a certain distance from the ZoneAgent boundary will also have access to water. Agents that are not in proximity to a Zone with a water supply will not have access to water (they will be thirsty). BufferZoneAgents are for visualization to illustrate the extend of the boundary around a ZoneAgent.”*

This example was modified to represent a more realistic example of As uptake. Mobile agents (stars in Figure 2) were moved by a random walk from the bottom left to the top right of the agent domain. Two shapefiles were added to the example as ZoneAgents: one to represent buildings and the other representing a “patch” of contamination. The agents are moved by the random walk method and then “bounced” off the walls of the buildings before crossing the contaminated ground collecting exposure. Whilst the example was relatively crude, it demonstrated the capability of repast-simphony.



**Figure 2. Screenshot of the modified geography example**

### **3.1.1 Repast-simphony Lessons Learnt**

A summary of the lessons learnt are presented in the SWOT diagram in Table 2.

**Table 2. SWOT analysis for repast-simphony**

<p><b>Strengths</b></p> <p>Agent-based models can be programmed from scratch in universal programming languages.</p> <p>Library of models to test and extract useful code.</p> <p>Can be used for to create GIS models.</p> <p>Can bring in shapefile and display (in geographical context).</p> <p>Different geographical contexts: grid, space and geographical.</p> <p>Rules on where agents can move achieved by using logic</p>	<p><b>Weaknesses</b></p> <p>Dearth of user community despite apparent popularity.</p> <p>Advanced Java programming skills required</p> <p>Visualisation module unstable.</p> <p>Reliance of open source coding and visualisation software.</p>
<p><b>Opportunities</b></p> <p>Universal programming language so easily conveyed to programming community and ABM specialists.</p> <p>Applicable to other research lines in and outside BGS.</p> <p>Upskill team in Java programming.</p> <p>Specialised modelling not common to all organisations.</p> <p>Use pre-programmed models as code resource to build on.</p>	<p><b>Threats</b></p> <p>Time required to upskill BGS staff.</p> <p>The level of Java programming is highly advanced and that team will find it difficult to meet demand within resources allocated.</p> <p>Time resources need to programme a good model will exceed resources available to team.</p>

### 3.2 NETLOGO

NetLogo is an agent-based programming language and integrated modelling environment.

NetLogo was designed, in the spirit of the Logo programming language, to be "low threshold and no ceiling"<sup>1</sup>. Logo is an educational programming language, designed in 1967 by Daniel G. Bobrow, Wally Feurzeig, Seymour Papert and Cynthia Solomon. The language was originally conceived to teach concepts of programming related to the computer language Lisp and only later to enable what Papert called "body-syntonic reasoning" where students could understand (and predict and reason about) a turtle's motion by imagining what they would do if they were a turtle.

NetLogo teaches programming concepts using agents in the form of turtles, patches, links and the observer. NetLogo was designed for multiple audiences in mind, in particular: teaching children in the education community and for domain experts without a programming background to model related phenomena. Many scientific articles have been published using NetLogo ( web of science search on NetLogo on 30/3/2016 gives 320 results) .

---

<sup>1</sup> Low threshold and no ceiling concept means that programmers with limited skills can start using the language, but that it is flexible enough for much more complex problems to be tackled.

The NetLogo environment enables exploration of emergent phenomena. It comes with an extensive models library including models in a variety of domains, such as economics, biology, physics, chemistry, psychology, system dynamics. NetLogo allows exploration by modifying switches, sliders, choosers, inputs, and other interface elements. Beyond exploration, NetLogo allows authoring of new models and modification of existing models.

NetLogo is freely available from the NetLogo website. It is in use in a wide variety of educational contexts from elementary school to graduate school. Many teachers make use of NetLogo in their curricula.

NetLogo was designed and authored by Uri Wilensky, director of Northwestern University's Center for Connected Learning and Computer-Based Modelling.

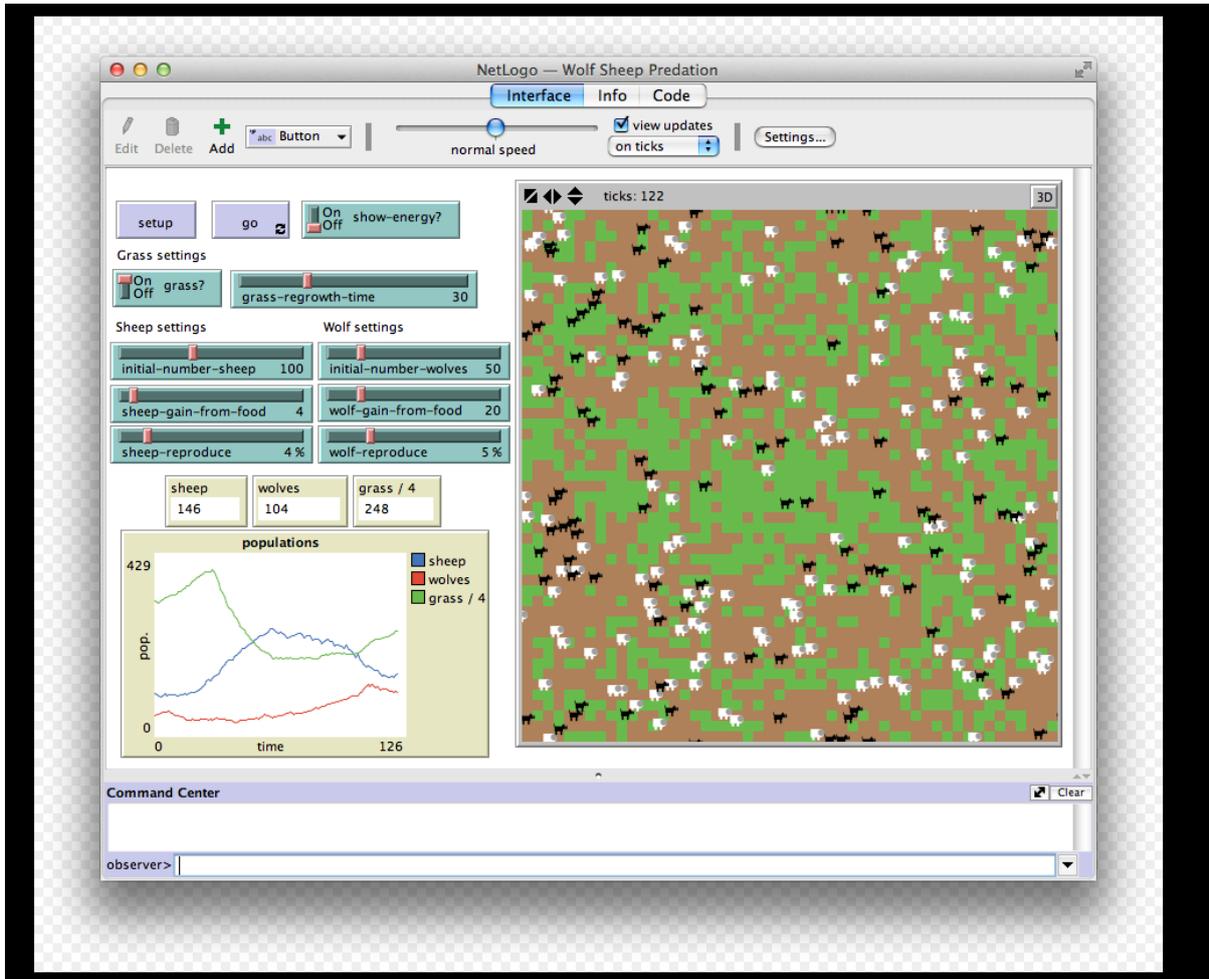
### **3.2.1 User group and support**

NetLogo has a number of user groups and discussion lists. The program itself is written in Java and there is a “NetLogo-devel” for developing the actual program (<https://groups.google.com/forum/#!forum/netlogo-devel>). There is a group for asking programming questions on the NetLogo language on the “Stack Overflow” site (<http://stackoverflow.com/questions/tagged/netlogo>). NetLogo also has a twitter feed (@NetLogo) for release announcements and NetLogo-related news.

There are also extensive libraries of NetLogo models. The program comes with a wide variety of example models covering the fields of art, biology, chemistry and physics, computer science, earth science, games, mathematics, networks, social science, system dynamics. Users are encouraged to upload their completed and tested models to the site. There is also a site called “NetLogo Modelling Commons” which NetLogo user can sign up to share models (<http://modelingcommons.org/account/login>). It currently contains over 1000 models.

### **3.2.2 Program environment**

The program consists of a programming window and an associated graphical user interface (GUI) (Figure 3). The GUI which contains a pictorial view of the application being modelled (movement of agents and the state of the patches the agents are moving within and around). As well as this, there are buttons which control specific subroutines of the underlying program and sliders which can be used to change the value of inputs to the program as well plots and data reporting boxes to monitor the progress of the model.



**Figure 3. Example screen shot of a NetLogo model**

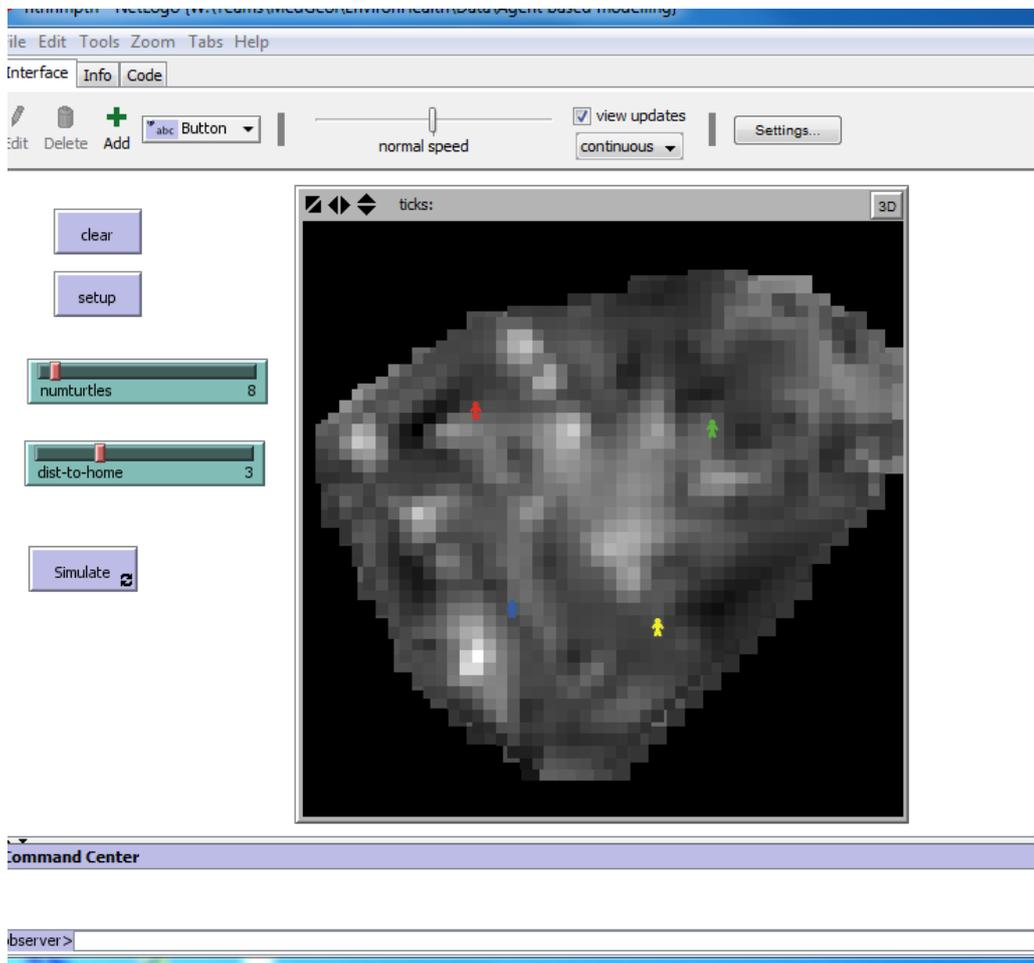
### 3.2.3 Preliminary assessment of NetLogo for Human Health Risk assessment

The initial aim was to look at the feasibility of both generally using ABM within BGS and the NetLogo language specifically as a tool for human health risk assessment with respect to contaminated land.

A simple scenario was tested. The town of Northampton has soils developed from the underlying ironstone formations which have elevated As concentrations. The soils were the subject of a study in which the bioaccessibility of the As in the soils was modelled from bioaccessibility measurements, soil properties and total As concentrations (Cave et al 2013). The idea was to take the spatial distribution of As bioaccessibility from the study, read this into NetLogo, then to have a set of human agents moving over different sections of the sample space (Northampton As bioaccessibility map) whilst being exposed to the variable concentrations of As and building up an exposure over time.

#### 3.2.3.1 READING IN THE SPATIAL DATA

NetLogo has a built in function for importing spatial data either as a raster or shapefiles. In this case a raster file of the As bioaccessibility was read into the NetLogo environment (Figure 4) with white representing high bioaccessible As and black representing low bioaccessible As.



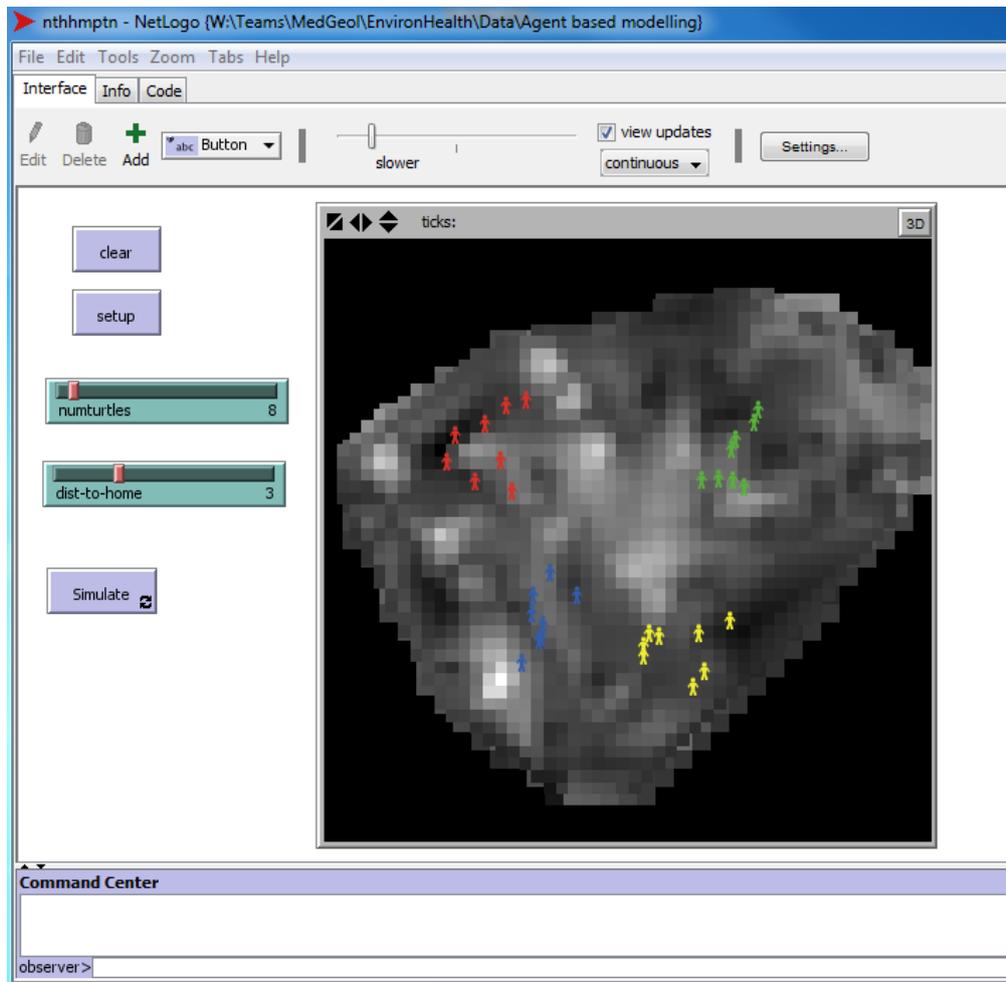
**Figure 4. Northampton As bioaccessibility imported into NetLogo**

### 3.2.3.2 AGENT MOVEMENT AND EXPOSURE

Four separate populations of human subjects containing 8 humans were set up in different locations in Northampton, shown as the different coloured figures in Figure 4. The program is set up so that each population moves in a random walk but no more than a set distance from the starting point. Each human agent starts out with a zero exposure to As and each time it move onto a new square their exposure is increased by a given fraction of the As bioaccessibility of the square they are occupying. The programming script for the model shown in the Appendix 1.

### 3.2.3.3 OPERATION OF THE MODEL

Figure 5 shows the model in operation for each population with the agents moving in their designated areas at no more than 3 units away from their starting locations.

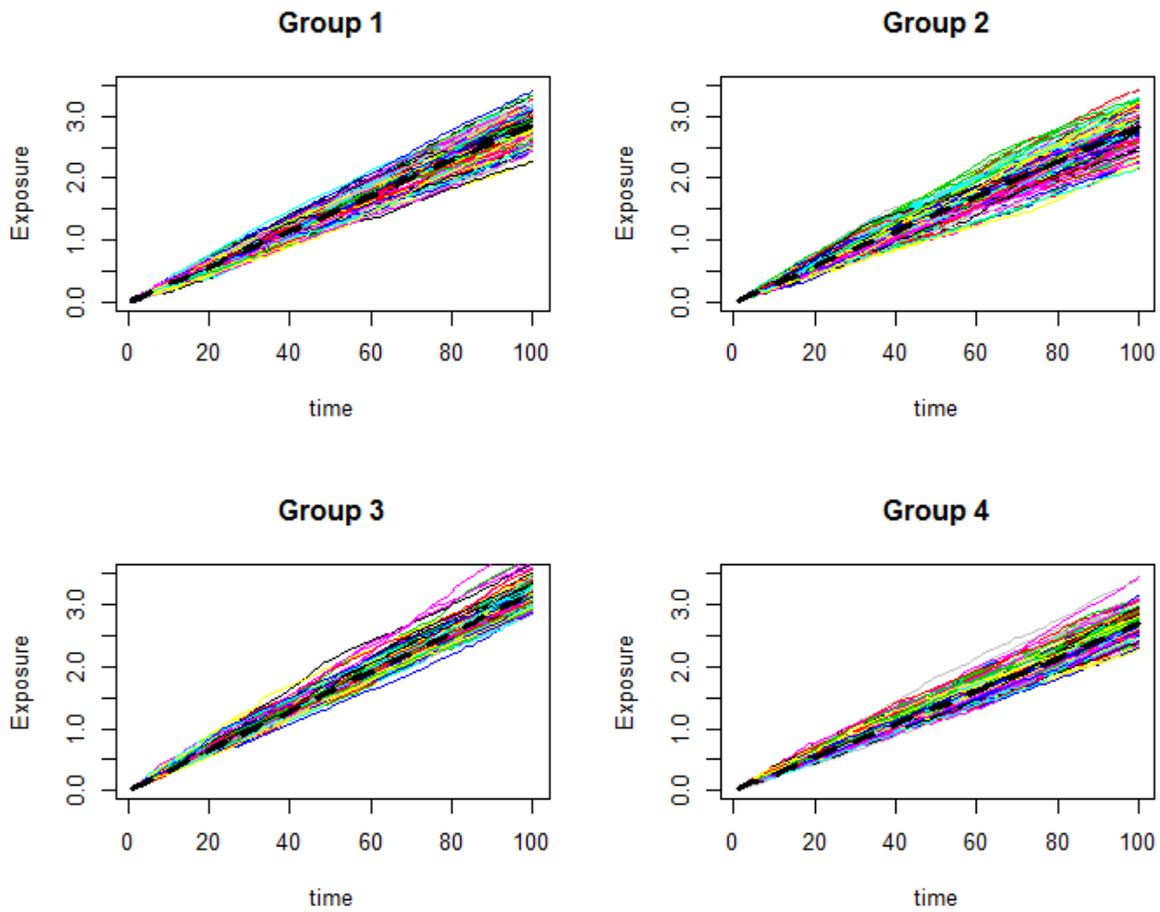


**Figure 5. Northampton As exposure model in operation**

#### 3.2.3.4 MONITORING OF THE MODEL OUTPUTS

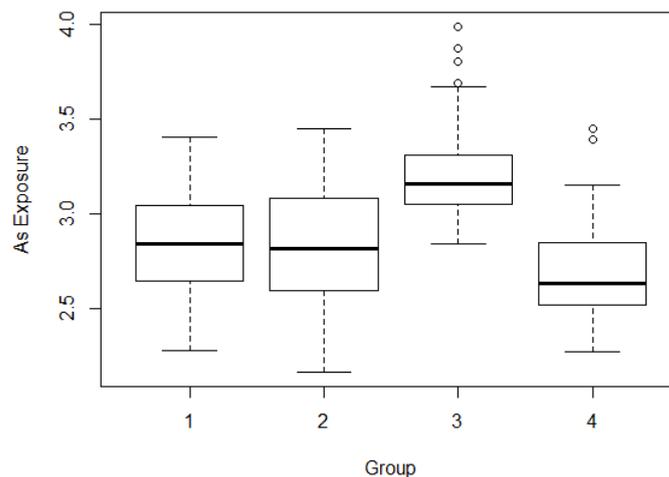
Whilst the NetLogo model allows the exposure received by each human agent in each area to be plotted in graphs there is an additional feature of NetLogo which gives additional functionality to controlling inputs and collecting and analysing data. There is a package written in the R programming language called “RNetLogo” which allows an R program to run a NetLogo model to modify the input parameters and collect the outputs for further statistical analysis.

As an example an R program was written to run the Northampton NetLogo model setting 100 agents at each of the 4 locations and collecting the data from 100 time steps. The R script for this is shown in Appendix 2.



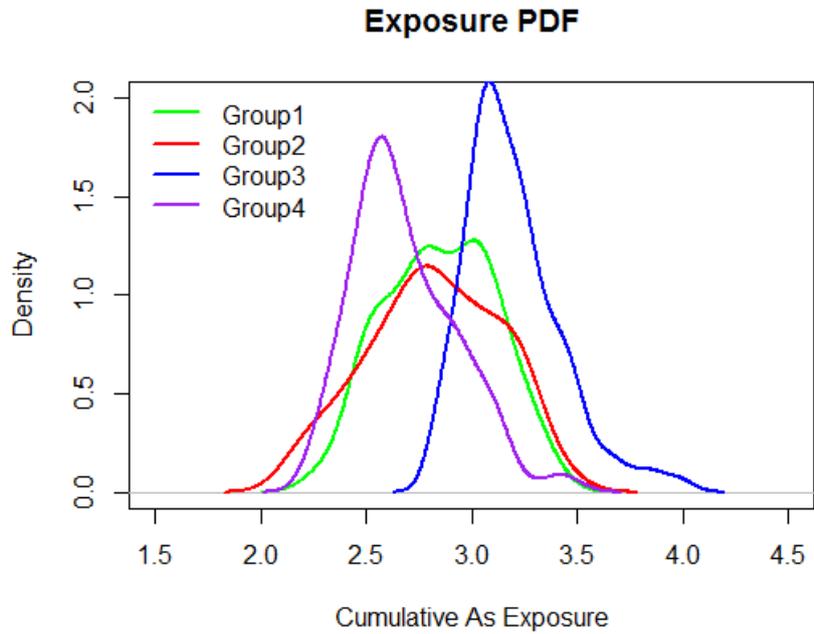
**Figure 6. Cumulative As exposure for each of the human agents in each of the four groups (coloured lines) and the mean cumulative exposure for each group (dashed black line)**

Figure 6 shows the cumulative exposure for each of the 100 human agents in each of the 4 locations.



**Figure 7. Box and whisker plot of the exposure for each of the 4 groups after 100 time steps**

Figure 7 summarises the final exposure in each of the four groups in terms of a boxplot and Figure 8 shows the same data but in the form of a probability density function.



**Figure 8. Probability density function plot of the exposure for each of the 4 groups after 100 time steps**

**3.2.4 Conclusions**

The NetLogo language appears to be a well-established tool in the field of ABM with a strong user base and a good record of peer reviewed work in the literature. Within the context of this scoping study the NetLogo environment has worked well on the small demonstration project and shows promise to be a very useful tool for human health risk assessment from contaminated soils.

**Table 3. SWOT analysis for NetLogo**

<b>Strengths</b>	<b>Weaknesses</b>
<p>Agent-based models can be programmed from scratch in universal programming languages.</p> <p>Library of models to test and extract useful code.</p> <p>Can be used for to create GIS models.</p> <p>Can bring in shapefile and display (in geographical context).</p> <p>Different geographical contexts: grid, space and geographical.</p> <p>Strong user community.</p> <p>Significant published literature using NetLogo</p>	<p>R programming skills required</p>

<b>Opportunities</b>	<b>Threats</b>
<p>Universal programming language so easily conveyed to programming community and ABM specialists.</p> <p>Applicable to other research lines in and outside BGS.</p> <p>Specialised modelling not common to all organisations.</p> <p>Use pre-programmed models as code resource to build on.</p>	<p>Time required to upskill BGS staff.</p> <p>Relies on open source products</p>

### 3.3 INSIGHT-MAKER

Insight-maker: insightmaker.com is a web-based model building tool which has an ABM capability. Aimed at making modelling more accessible to a wider range of users, the user needs to have an account which can be used to build and run models. To understand the capabilities of the software the ABM Tutorial was undertaken: [insightmaker.com/node/3790](https://insightmaker.com/node/3790). This tutorial was implemented which can be accessed at: <https://insightmaker.com/insight/49684/ABM-Tutorial>

A relatively simple interface is designed to allow the user to set up a model using a flow chart-based approach (Figure 9). This shows the creation of a simple example using primitives (activities used to build a model) and links (which join up the actions). Once the model has been run the output can be displayed in a number of different ways: time series plots and the distribution of the agents (Figure 10).

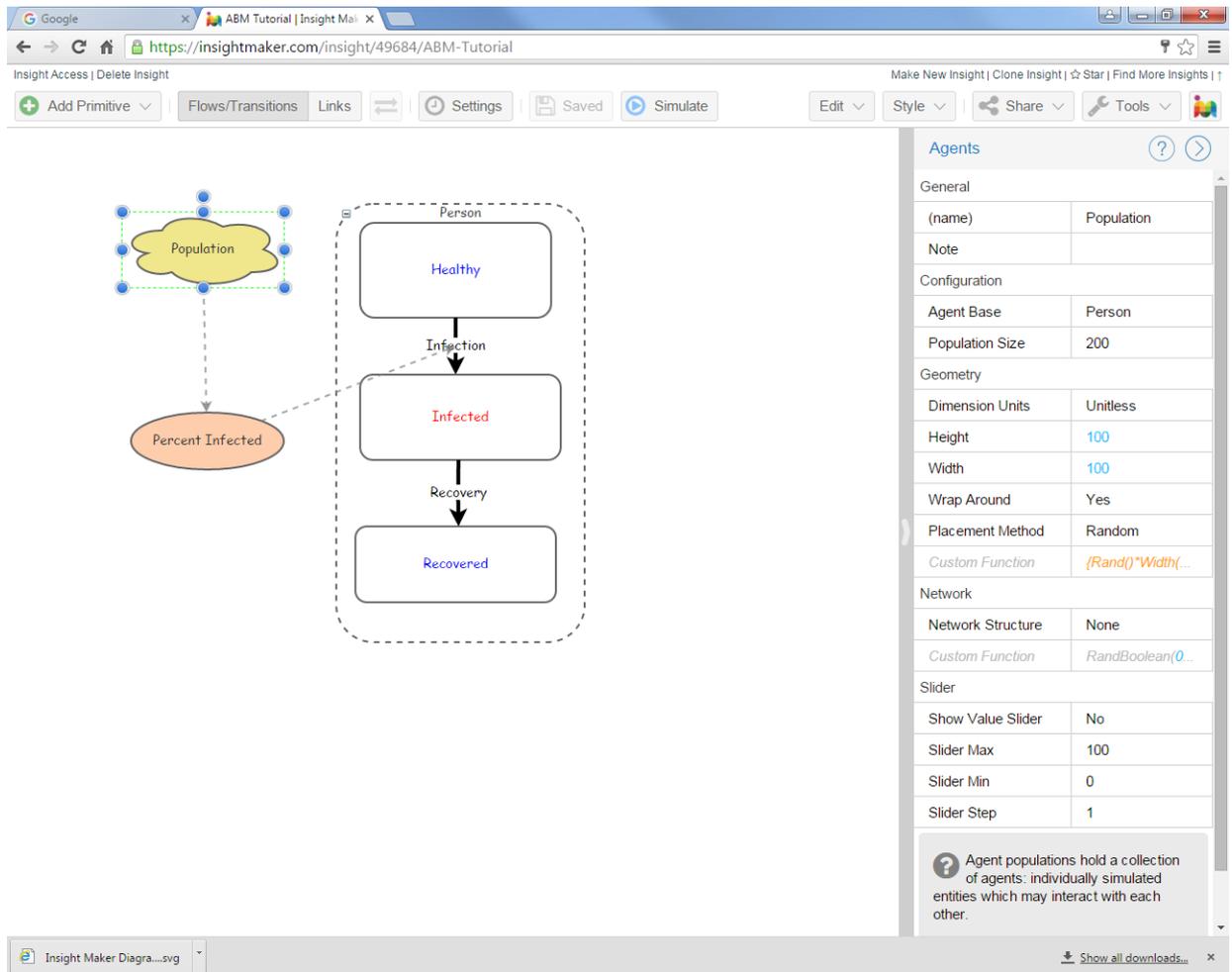


Figure 9. Example model created using “primitives” and “links”



**Figure 10. Examples of model output**

Whilst Insight-maker offers a simple and effective way of producing models, it does have a number of limitations. It is web-based which limits the processing power, it requires models to be stored on a server and is aimed at “quick and dirty” solutions. Whilst it does what it sets out to do, namely lower the barrier to entry for modelling, this is achieved at the expense of scalability to increasingly complex problems.

**Table 4. SWOT analysis for Insight-maker**

Strengths	Weaknesses
<p>Agent-based models can be quickly and easily programmed from scratch in universal programming languages.</p> <p>Library of models to test and extract useful code.</p> <p>Relatively strong user community.</p>	<p>Lack of scalability to more complex problems</p>

<b>Opportunities</b>	<b>Threats</b>
<p>Universal programming language so easily conveyed to programming community and ABM specialists.</p> <p>Applicable to other problems inside and outside BGS.</p>	<p>Code resides in web-based repository</p> <p>Website may be no longer supported</p>

## 4 Recommendations for future work

### 4.1 SUMMARY AND NEXT STEPS

By exploring the examples to a) understand how they work and b) attempt to answer the question “Is Agent Based Modelling (ABM) useful for extending the work of bioaccessibility and environmental interaction?” then the following conclusion are reached:

- a. Ease of use: Insight-maker > NetLogo > Repast-simphony
- b. Flexibility: Repast-simphony>NetLogo>Insight-maker

However what has become apparent is that there are two clear favourites for further use: repast-simphony and NetLogo. Both offer the flexibility to undertake complex problems with ABM.

It is clear that by the introduction of agents as representatives of real-world entities it can make the assessment of human health assessment potentially much easier. There are significant benefits in the conceptual understanding of the modelling and how this can be exploited to explain the process to end-users. Examples of Local Authorities (LAs) would benefit from modelling of contaminated land problems with ABM.

#### 4.1.1 Next steps

The work packages planned for FY 16/17 are as follows:

1. Literature review to determine gaps in applying ABM to human health effects of exposure to contaminated land
2. Identify and characterise groups doing ABM including which software they use e.g.
  - a. UCL
  - b. Warwick
  - c. Lancaster
  - d. Risk Solutions Ltd
  - e. Other groups – who are they
3. Identify the use of external groups as mentors as well as collaborators
4. Do simple Northampton example (houses, roads and parks along with a range of agents – children of different ages and adults)
5. Build on human exposure matrix initiated as part of the work reported here
6. Produce database / spreadsheet of available environmental and exposure data
7. Improve Java knowledge within the group before next sprint<sup>2</sup>
8. Undertake sprints early in next FY (June/July) to capitalise on current momentum
9. Identify potential recipients of new BGS skills and ABM problem solving e.g. Université de Lorraine (Nancy, Fr)

### 4.2 DETAILS OF THE EXAMPLE TO BE DEVELOPED

The “real-world” ABM model should be based on a geographical context relying on attributed and gridded raster data representative of the local environment of Northampton (or similar area). External data are expected to include:

- Basemap e.g. OS Open Map
  - Buildings
  - Public open space
  - Roads

---

<sup>2</sup> Sprint is the process of implanted Agile working, i.e intense periods (up to three weeks) working as a team on one problem (see for example - scrummethodology.com).

- Gardens
  - Total and bioaccessible As concentrations
  - Soil type based on parent material maps or superficial deposits (e.g. clay, sand, loam)
  - Organic carbon content of soils

Some assistance from a BGS GIS specialist is expected to be required to assist with the creation of the data layers. The attributed geographical data will be used to reflect environmental constraints on the movement and flow of people.

The agents used in the model are expected to include children of different age classes and adults of the same age class. These age classes should be derived from the Environment Agency Contaminated Land Exposure Assessment (CLEA) model used to estimate chronic exposure to PHS in soil in humans. Doing so will help to ensure that the outputs are in a language that risk-based land management and land-use planning professionals can understand.

The behaviour of the agents should also be derived from CLEA. For the ingestion pathway this would include the amount of soil ingested per day based on a given number of hours spent within a particular land-use scenario e.g. public open space. Additional data sources may also be called on that are not generic e.g. actual number of visits to a park per year.

The aim of the model is to test whether ABM can simulate the exposure to As in humans from soil as well as the industry standard deterministic model (CLEA). If the answer is “yes” then “what if” scenarios would be explored e.g. what if a child only plays in one park or another, or in a park for 50% of the time and in a garden for 50% of the time. The results would then be used to explore land-use planning scenarios e.g. developing park #1 and extending park #2 or where the most suitable location is for park #3.

The potential results are believed to be innovative and relevant to not only to regulators e.g. planners and contaminated land officers, but also consultant business advising their clients on how a particular area of land might be used. The ABM would also be a beneficial way of exploring how existing BGS dataset, products and applied science might be made to be more impactful. Outputs of this work are expected include publication in a high impact journal and an application for funding under a relevant scheme e.g. Innovate UK, NERC Innovation Projects or commercially funded research. Links with existing project such as NERC KE Fellowships and other national initiative such as the Future Cities Catapult will be explored.

# Glossary

API – Application Programming Interface: method by which data can be accessed by calls to a programme

Choosers: NetLogo term meaning drop down menu for its interface

IDE – Integrated Development Environment: Term for interface used as programming language development environment to enable code to be written and tested within a single piece of software

KE – Knowledge Exchange: transferring research experience to industry whilst understanding their issues

Lisp: Programming language developed as a “mathematical formalism for reasoning about the use of recursion equations as a model for computation” in the 1950s.

NERC – Natural Environment Research Council ([www.nerc.ac.uk](http://www.nerc.ac.uk))

Sliders: NetLogo term meaning sliding controls for its interface

Stack Overflow: website which contains discussion forum for different software ([stackoverflow.com](http://stackoverflow.com))

Switches: NetLogo term meaning radio boxes for its interface

## References

- Batty, M., 2003. Agent-based pedestrian modelling. *Advanced spatial analysis: The CASA book of GIS*, 81.
- Brown, B. N., Price, I. M., Toapanta, F. R., DeAlmeida, D. R., Wiley, C. A., Ross, T. M., ... & Vodovotz, Y., 2011. An agent-based model of inflammation and fibrosis following particulate exposure in the lung. *Mathematical biosciences*, 231(2), 186-196.
- Castilla-Rho, J. C., Mariethoz, G., Rojas, R., Andersen, M. S., & Kelly, B. F. J., 2015. An agent-based platform for simulating complex human-aquifer interactions in managed groundwater systems. *Environmental Modelling & Software*, 73, 305-323.
- Cave, M. R., Wragg, J. & Harrison, H. 2013. Measurement modelling and mapping of arsenic bioaccessibility in Northampton, UK. *Journal of Environmental Science & Health Part A*, 48, 629-640.
- Crooks, A., Castle, C., & Batty, M., 2008. Key challenges in agent-based modelling for geo-spatial simulation. *Computers, Environment and Urban Systems*, 32(6), 417-430.
- Dia, H., 2002. An agent-based approach to modelling driver route choice behaviour under the influence of real-time information. *Transportation Research Part C: Emerging Technologies*, 10(5), 331-349.
- Farmer, J. D., & Foley, D., 2009. The economy needs agent-based modelling. *Nature*, 460(7256), 685-686.
- Filatova, T., Verburg, P. H., Parker, D. C., & Stannard, C. A., 2013. Spatial agent-based models for socio-ecological systems: challenges and prospects. *Environmental modelling & software*, 45, 1-7.
- Macal, C. M., & North, M. J., 2010. Tutorial on agent-based modelling and simulation. *Journal of simulation*, 4(3), 151-162.
- North, M. J., Collier, N. T., Ozik, J., Tatara, E. R., Macal, C. M., Bragen, M., & Sydelko, P., 2013. Complex adaptive systems modeling with Repast Symphony. *Complex adaptive systems modeling*, 1(1), 1.
- Schelhorn, T., O'Sullivan, D., Haklay, M., & Thurstain-Goodwin, M., 1999. *STREETS: An agent-based pedestrian model*.
- Schreinemachers, P., & Berger, T., 2011. An agent-based simulation model of human-environment interactions in agricultural systems. *Environmental Modelling & Software*, 26(7), 845-859.
- Turner, A., & Penn, A., 2002. Encoding natural movement as an agent-based system: an investigation into human pedestrian behaviour in the built environment. *Environment and planning B: Planning and Design*, 29(4), 473-490.

# Appendix 1 NETLOGO ASSOCIATED Scripts

## NetLogo Script for the Northampton model

### NetLogo Script

```

extensions [gis]
globals [elevation-dataset]
patches-own [elevation]
turtles-own [exposure group]

to setup
  let xHm [-7 6 -5 3]
  let yHm [6 5 -5 -6]
  set-default-shape turtles "person"
  set      elevation-dataset      gis:load-dataset      "C:/Users/mrca/My
Documents/Nthpton_As.asc"
  gis:paint elevation-dataset 0
  gis:apply-raster elevation-dataset elevation
  let min-elevation gis:minimum-of elevation-dataset
  let max-elevation gis:maximum-of elevation-dataset
  ask patches
  [ if (elevation <= 0) or (elevation >= 0)
    [set pcolor scale-color black elevation min-elevation max-elevation]
  ]
  ask patches
  [ if pcolor = black
    [set elevation 0]
  ]
  create-turtles numturtles [
    set exposure 0
    set xcor -7
    set ycor 6
    set group 1
    set color red
  ]
  create-turtles numturtles [
    set exposure 0
    set xcor 6
    set ycor 5
    set group 2
    set color green
  ]
  create-turtles numturtles [
    set exposure 0
    set xcor -5
    set ycor -5
    set group 3
    set color blue
  ]
  create-turtles numturtles [
    set exposure 0
    set xcor 3

```

```
    set ycor -6
    set group 4
    set color yellow
  ]
end
to go
ask turtles with [group = 1][
  ifelse distancexy -7 6 > dist-to-home
  [facexy -7 6]
  [rt random 360]           ; set random heading
  forward 1                 ; advance one step
]
ask turtles with [group = 2][
  ifelse distancexy 6 5 > dist-to-home
  [facexy 6 5]
  [rt random 360]          ; set random heading
  forward 1                ; advance one step
]
ask turtles with [group = 3][
  ifelse distancexy -5 -5 > dist-to-home
  [facexy -5 -5]
  [rt random 360]          ; set random heading
  forward 1                ; advance one step
]
ask turtles with [group = 4][
  ifelse distancexy 3 -6 > dist-to-home
  [facexy 3 -6]
  [rt random 360]          ; set random heading
  forward 1                ; advance one step
]
ask patches [
  let my-turtles turtles-here
  if any? my-turtles [
    let lvls elevation / 100
    ; print lvls
    ask my-turtles
    [set exposure (exposure + lvls)]
  ]
]
End.
```

## R Script for the NetLogo Northampton Model

```
library(RNetLogo)
# this part starts up the NetLogo language
nl.path <-"C:/Program Files/NetLogo 5.3/app"
NLStart(nl.path,gui=TRUE, nl.obj=NULL, is3d=FALSE)
# now get the file path for the model we ant to run
model.path<-file.choose()
NLLoadModel(model.path)
# This part sets up the model setting the number of turtles at each site
# and the maximum distance from home the agents are allowed to move
NLCommand("clear-all")
NLCommand("set numturtles 100")
NLCommand("set dist-to-home 3")
NLCommand("setup")
# Now we need to set up some dataframes to store the arsenic exposure
```

```

# for each agent in each group
df1<-data.frame(nms=as.character(1:100))
df2<-data.frame(nms=as.character(1:100))
df3<-data.frame(nms=as.character(1:100))
df4<-data.frame(nms=as.character(1:100))
# this loop runs the NetLogo Program over 100 cycles collecting the exposure
# from each agent in each group at each cycle
for(i in 1:100){
NLDoCommand(1, "go")
Exp1<- NLGetAgentSet("exposure","turtles with [group = 1]")
df1<-cbind(df1,Exp1)

Exp2<- NLGetAgentSet("exposure","turtles with [ group = 2 ]")
df2<-cbind(df2,Exp2)

Exp3<- NLGetAgentSet("exposure","turtles with [ group = 3 ]")
df3<-cbind(df3,Exp3)

Exp4<- NLGetAgentSet("exposure","turtles with [ group = 4 ]")
df4<-cbind(df4,Exp4)
}
# now the results can be plotted out
#####
# This plots the cumulative exposure for each agent in each group
# along with an average value for each group
par(mfrow=c(2,2))
avg1<-sapply(df1[,2:101],FUN=mean)
plot(1:100,avg1,type="l", ylim=c(0,3.5),lty=2,lwd=4,
      xlab="time",ylab="Exposure", main="Group 1")
for( i in 1:100){
  lines(1:100,df1[i,2:101],col=83+i)
}
lines(1:100,avg1,lty=2,lwd=3)

avg2<-sapply(df2[,2:101],FUN=mean)
plot(1:100,avg2,type="l", ylim=c(0,3.5),lty=2,lwd=2,
      xlab="time",ylab="Exposure", main="Group 2")
for( i in 1:100){
  lines(1:100,df2[i,2:101],col=83+i)
}
lines(1:100,avg2,lty=2,lwd=3)

avg3<-sapply(df3[,2:101],FUN=mean)
plot(1:100,avg3,type="l", ylim=c(0,3.5),lty=2,lwd=2,
      xlab="time",ylab="Exposure", main="Group 3")
for( i in 1:100){
  lines(1:100,df3[i,2:101],col=83+i)
}
lines(1:100,avg3,lty=2,lwd=3)

avg4<-sapply(df4[,2:101],FUN=mean)
plot(1:100,avg4,type="l", ylim=c(0,3.5),lty=2,lwd=2,
      xlab="time",ylab="Exposure", main="Group 4")
for( i in 1:100){
  lines(1:100,df4[i,2:101],col=83+i)
}
lines(1:100,avg4,lty=2,lwd=3)
#####
# This does a box and whisker plot for the final exposures over
# 100 cycles
par(mfrow=c(1,1))
alldat<-cbind(df1[,101],df2[,101],df3[,101],df4[,101])
boxplot(alldat, xlab="Group",ylab="As Exposure")
#####
#This does a probability density function of the final exposure in each

```

```
# group
plot(density(alldat[,1]),main="Exposure PDF",
      xlab="Cumulative As Exposure",col="green",
      ylim=c(0,2),xlim=c(1.5,4.5),lwd=2)
lines(density(alldat[,2]),col="red",lwd=2)
lines(density(alldat[,3]),col="blue",lwd=2)
lines(density(alldat[,4]),col="purple",lwd=2)
legend("topleft",c("Group1","Group2","Group3","Group4"),
      col=c("green","red","blue","purple"),bty="n",lty=1,lwd=2)
#####
# this quits the NetLogo program
NLQuit()
.
```

## Appendix 2 REPAST-SIMPSONY Scripts

## ASSOCIATED

GisAgent class which moves agents

```

public class GisAgent {
    private String name;
    private boolean water = false;
    private double waterRate;
    private Grid<Object> grid;
    private double exposure;
    private double expFactor = 0.5;
    public GisAgent(String name, Grid<Object> grid) {
        this.name = name;
        this.grid = grid;
    }
    @ScheduledMethod(start = 1, interval = 1, priority =
ScheduleParameters.FIRST_PRIORITY)
    public void step() {
        randomWalk();

        // The agent drinks and gets thirsty.
        water = false;
        waterRate = 0;
    }

    /**
     * Random walk the agent around.
     */
    private void randomWalk(){
        Context context = ContextUtils.getContext(this);
        Geography<GisAgent> geography =
(Geography)context.getProjection("Geography");
        //geography.moveByDisplacement(this, RandomHelper.nextDoubleFromTo(-0.0005,
0.0005),
        //          RandomHelper.nextDoubleFromTo(-0.0005, 0.0005));

        double xMove = RandomHelper.nextDoubleFromTo(-0.5, 1);
        double yMove = RandomHelper.nextDoubleFromTo(-0.5, 1);
        Geometry geometry = geography.getGeometry(this);
        Coordinate coordinate = geometry.getCoordinate();
        double xAgent = coordinate.x;
        double yAgent = coordinate.y;
        double xNew = xAgent + xMove;
        double yNew = yAgent + yMove;
    }
}

```

```

    if(xNew < 0 || xNew > 100){
        xMove = 0;
    }
    if(yNew < 0 || yNew > 100){
        yMove = 0;
    }

    geography.moveByDisplacement(this,xMove, yMove);
    geometry.getCoordinate();
    xAgent = coordinate.x;
    yAgent = coordinate.y;
    System.out.println(" Agent is: "+name+" at x: "+xAgent+" and y: "+yAgent);
    grid.moveTo(this, (int)xAgent, (int)yAgent);
    GridPoint pt = grid.getLocation(this);
    //System.out.println(" Agent is : "+name+" at at grid x: "+pt.getX()+" and
y: "+pt.getY());
    }
    public String getName() {
        return name;
    }
}

```

Method which checks for coincident agents and “rebounds” moving agents off buildings

```

private void checkForAgents(){
    Context context = ContextUtils.getContext(this);
    Geography geography = (Geography)context.getProjection("Geography");

    // Find all features that intersect the zone feature
    IntersectsQuery query = new IntersectsQuery(geography, this);

    Parameters parm = RunEnvironment.getInstance().getParameters();
    double zoneDistance = (Double)parm.getValue("zoneDistance"); //
meters

    // method checks if the polygon is present at same location as
GisAgent
    // if yes then the method send the agent back to 0,0 and the
displacement walk starts again
    // need to produce rule that means the agent reacts more
realistically with polygon
    GeographyWithin within = new GeographyWithin(geography,
zoneDistance, this);
    GeometryFactory fac = new GeometryFactory();
    for (Object obj : within.query()) {
        if (obj instanceof GisAgent){
            System.out.println(" Shapefile is: "+this.name);
            GisAgent agent = (GisAgent)obj;
            if ( this.name == "Building"){

                // If the zone finds a GisAgent, set the agent water rate
from the zone

                geography.moveByDisplacement(agent, -0.5, -0.5); // Move
away from

            }else{
                agent.increaseExposure();
                // Increase exposure
            }
        }
    }
}

```

}  
}  
}