Agent Simulations in the Web Browser: A Survey of Web Browser-Based Agent Platforms

Ramesh Maddegoda[1], Asoka S. Karunananda[2]
[1]Independent Researcher,
Columbus, OH 43240, United States of America
[2] Department of Computational Mathematics, University of Moratuwa,
Moratuwa, Sri Lanka
email : [1]ramesh.maddegoda@gmail.com, [2] asokakaru@uom.lk

***Abstract*— Agent Based Modelling and Simulation (ABMS) is a very powerful technique used in many domains in order to model and simulate the actions and interactions of autonomous agents in complex systems. In the recent years, with the advancements in the frontend technologies, some of the ABMS tool developers released web browser-based ABMS tools which allow users to develop, view, share and execute agent-based simulations in any device, including the mobile devices, through a web browser. However, the existing surveys of ABMS tools mainly focus on the desktop based ABMS tools and do not provide comparisons and evaluations focused on web browser-based ABMS tools. To address this issue, we came-up with a comparison and evaluation criteria with a strong emphasis on web browser-based features. This paper presents a comparative up-to-date review of several publicly available web browser-based ABMS tools.**

Keywords*— *Agent Based Modelling and Simulation (ABMS) Tools, Swarm Intelligence, Modelling Complex Systems, Multi-Agent Systems

I. INTRODUCTION

Agent Based Modelling and Simulation (ABMS) is a promising computation modelling and simulation approach, where a system is modelled as a collection of autonomous decision making entities called agents. Each individual agent is following a basic set of rules to evaluate it's situation, making decisions and executing actions. As a result, higher level system properties emerge from the interactions between the lower level entities. The concepts of ABMS are highly inspired by the biological and sociological systems and it combines the elements of complexity science, multi-agent systems, game theory, emergence, computational sociology and evolutionary programming. ABMS is widely used in many scientific domains including biology [1], ecology [2] and social science [3].

The ABMS community has developed numerous agent based modelling tools that enable individuals to develop agent based models and simulations. Most of these tools are distributed as desktop applications and the users have to install them as software applications on computers. At the same time if an ABMS developer wants to share an agent based model with other individuals, it is necessary for those who want to view and execute the model also to have the same ABMS tool installed in their computers. As a results it is necessary to make sure that the recipients of the model are also using the same operating system, given that different ABMS tools are compatible with different operating systems. With the popularity of the smart phones, tablets and devices such as Chromebooks, there is a growing population of users who primarily use mobile operating systems and Chromebooks for their day to day activities. However, most of the ABMS tools are only supporting conventional computer operating systems and therefore it is impossible to view and execute the models built using most of the ABMS tools on mobile operating systems and Chromebooks. Also there are other reasons that discourage users from installing software on their computers such as concerns on malware and personal inconvenience.

With the advancements of the frontend web technologies, now it is possible to build web browser-based web applications that provide a look and feel, an acceptable level of performance and usability similar to the desktop applications. In recent years there were several web browser-based ABMS tools released by ABMS tool developers with the ability to execute models on web browsers. Some of those tools also provide web browser-based features to create agent based models in addition to the ability to view and execute models. As a result now it is possible for ABMS developers to share their models with others individuals, regardless of the operating systems or devices used by the recipients. Because these web browser-based ABMS tools are compatible with most of the common web browsers used in both personal computers and mobile devices.

II. RELATED SURVEYS OF AGENT BASED MODELLING AND SIMULATION TOOLS

In recent years several researchers have conducted surveys of ABMS tools. The paper "Agent Based Modelling and Simulation tools: A review of the state-of-art software" [4] presented by Sameera Abar, Georgios K. Theodoropoulos, Pierre Lemarinier and Gregory M.P. O'Hare provides a comprehensive comparative literature survey of ABMS tools and this is one of the most detailed surveys done on the subject. They have evaluated over eighty ABMS tools and provided results by comparing license/ availability, source code, type of agent based on it's interaction behaviour, programming language/API and Integrated Development Environment (IDE) used, compiler, operating system and implementation platform, modelling strength, scalability, ABMS scope or application domain. The main intention of this paper is to help scientists and engineers to quickly assess how they might choose and properly apply ABMS to their own research applications.

The "Survey of Agent Based Modelling and Simulation Tools" [5] by R.J. Allan outlines more than forty existing ABMS tools based on information found online and based on personal experiences. Also this report explores the applications of ABMS in the domains of biology and medicine, physics and chemistry, security, cyber security, the environment, social and economic modelling, supply network and transport optimisations. And this report focuses on executing ABMS on high performance computing systems.

The paper "Evaluation of free Java-libraries for social-scientific agent based simulation" [6] by Robert Tobias and Carole Hofmann compares four freely available programming libraries for support of social scientific agent based computer simulations. They have evaluated RePast, Swarm, Quicksilver and VSEit software and concluded that RePast software as a clear winner compared to the other software considered for their evaluation.

Kalliopi Kravari and Nick Bassiliades have published the article "A Survey of Agent Platforms" [7] to provide a comparative review of twenty four existing agent platforms. The authors have developed a comparison and evaluation criteria based on platform properties, usability, operating ability, pragmatics and security management. Also this article provides classifications for agent platforms in order to help readers to understand which agent platforms broadly exhibit similar properties and in which situations which choices should be made. These classifications include programming languages used by ABMS tools, FIPA compliance, application domain and the support for semantic web technologies.

"A Survey of Programming Languages and Platforms for Multi-Agent Systems" [8] by Rafael Bordini et al. presents a survey of programming languages and development tools for multi-agent systems. They have conducted this survey by considering the programming languages (declarative, imperative, and hybrid) and integrated development environments used by multi-agent systems. And finally they explore the frameworks that are not so strongly tied to a particular programming language.

Cynthia Nikolai and Gregory Madey have presented the paper "Tools of the Trade: A Survey of Various Agent Based Modelling Platforms" [9] which examines over fifty ABM platforms and characterizes each based on five important characteristics users consider when choosing a toolkit. These characteristics are: language required to program a model and to run a simulation, operating system required to run the toolkit, type of license that governs the toolkit, primary domain for which the toolkit is intended, and types of support available to the user.

Many of the above surveys provide very comprehensive and detailed comparisons of ABMS tools. Some of the above surveys have considered few web browser-based ABMS tools also in their research. However, the main focus of above surveys were not to evaluate the features specific to web browser-based ABMS tools. Our intention was to conduct a survey of web browser-based ABMS tools in order to fill this gap and provide a supplement to the existing surveys of ABMS tools.

III. Overview of the Web Browser-Based Agent Platforms

At the time of this survey, following web browser-based ABMS tools were publicly available on the internet and considered for the survey.

A. AgentBase

AgentBase.org [10], [11] allows users to build agent based models that run in the web browser. It follows the Agent oriented Programming model of NetLogo [12]. Models in AgentBase use the AgentBase Library, and are written in CoffeeScript, which is instantly interpreted as JavaScript. This AgentBase library was derived from AgentScript and it provides a rich set of resources for moving and drawing agents, neighbour detection, and many other features.

B. AgentScript

AgentScript.org [13] provides a minimalist agent based modelling system based on NetLogo semantics. It has a Model View Control (MVC) architecture and it is a modern ES6 module based JavaScript implementation.

C. Behaviour Composer

Behaviour Composer [14], [15] is a web based tool available on modelling4all.org to support teachers, learners and researchers, including those with little or no programming experience, to build, share, and discuss agent based computer models using a visual tool. When a user executes a model built using the Behaviour Composer, it automatically converts the model in to a NetLogo file and loads the model in to a web browser-based view powered by NetLogo Web [16].

D. Insight Maker

Insight Maker [17], [18] is a modelling and simulation tool that runs in web browser. It provides a rich set of features for system dynamics modelling and agent based modelling. Also insightmaker.com provides a rich collection of models called Insights, published by Insight developers.

E. Mesa

Mesa [19], [20] is an agent based modelling framework for Python. Mesa allows users to build agent based models, visualize them using a web browser-based interface and analyse the results using Python related data analysis tools. With the recent popularity of Python based tools, Mesa provides a low learning curve for the users who are already familiar with Python.

F. NetLogo Web

NetLogo Web [16] is a version of the NetLogo modelling environment [12] that runs entirely in the web browser. NetLgo is one of the most widely used and feature rich ABMS tool. At the moment NetLogo Web provides only a subset of the features available in NetLogo desktop version. However, NetLogo Web is a highly usable and stable ABMS environment which provides most of the necessary features.

G. StarLogo Nova

StarLogo Nova [21] is the online version of StarLogo TNG [22], which is a widely known ABMS language. It provides a web browser-based environment to build and

simulate models. StarLogo Nova mainly focuses on education with an effort to introduce programming skills to younger students. It uses a programming language of coloured blocks that fit together like puzzle pieces. StarLogo Nova is used by users with various backgrounds to create games, simulations and to study diverse concepts in science and maths, such as epidemiology, ecology, geometry and computational thinking.

IV. Comparison and Evaluation Criteria

Based on our literature review on previous surveys of ABMS tools, we identified that the comparison and evaluation criteria of previous surveys have not addressed some of the important accepts specific to web browser-based ABMS tools. Therefore we came-up with our comparison and evaluation criteria based on 1) Constraints, web browser compatibility and supported devices; 2) Licence, cost, current development status and affiliations; 3) Programming languages and terminology; 4) Access control and model sharing; 5) Availability of classic agent based models, and 6) Learning resources, user community and model repositories. The following sections of this paper review the selected ABMS tools with the help of this comparison and evaluation criteria.

V. Constraints, Web Browser Compatibility and Supported Devices

One of the main advantages of web browser-based ABMS tools is the ability to share and execute agent based simulations on web browsers regardless of the operating system or the device of the user. In addition to the ability to execute agent based simulations, some of the ABMS tools included in this survey provide features to create and edit agent based models only using a web browser. For example, AgentBase, Behaviour Composer, Insight Maker, NetLogo Web and StarLogo Nova allow users to create and edit models on web browser without any additional tools. On the other hand, AgentScript and Mesa are primarily software libraries to be used with programming languages such as JavaScript and Python and they do not provide in-built features to edit the models only using a web browser. When using AgentScript or Mesa, it is expected a developer to create and host the agent based simulations as a web application, so other individuals can view and execute agent based simulations through web browsers. However, since Mesa is a Python package, it is possible to import Mesa through a Jupyter Notebook [23]. A Jupyter Notebook is an open source web application that enables users to create and share documents that contain live code, equations, visualizations and text. When Mesa is imported in a Jupyter Notebook, it provides an interactive environment where users can create, edit and execute agent based models and simulations in a web browser.

Compared to the desktop versions of ABMS tools, the web browser-based ABMS tools provide a limited set of features due to the constraints exist with web browsers. For example, NetLogo desktop version provides a 3D (three-dimensional) view where agent based modelling and simulation can be executed in a 3D environment. However, the NetLogo Web version only supported a 2D (two-dimensional) view at the time of writing this paper. Another example is the GIS (Geographic Information Systems) support available as an extension to the NetLogo desktop version. At the time of this survey, NetLogo Web did not have a support for extensions such as GIS extension. Table 1 provides a comparison of ABMS tools in terms of 2D, 3D and GIS support.

Table 1 2D, 3D and GIS Support

ABMS Tool	View Agent Based Simulations in 2D	View Agent Based Simulations in 3D	GIS Support
AgentBase	✓	-	-
AgentScript	✓	✓	✓
Behaviour Composer	✓	-	-
Insight Maker	✓	-	-
Mesa	✓	-	✓ (with mesa-geo GIS extension)
NetLogo Web	✓	-	-
StarLogo Nova	✓	✓	-

To test the compatibility with different web browsers, it was decided to evaluate each ABMS tool with four widely used web browsers available on desktop computers and laptops. The ability to create models, ability to edit models and ability execute simulations of each tool were tested with different web browsers. For AgentScript, only the ability to execute simulations was tested, because AgentScript is a software library that does not provide any interactive editing tools to create or edit models using a web browser. The Mesa ABMS tool, which is another software library, was tested after importing Mesa in an interactive Jupyter Notebook. All the ABMS tools considered for this survey passed this web browser compatibility test with Google Chrome, Microsoft Edge, Mozilla Firefox and Safari web browsers.

With the growing use of the mobile phones, tablets and devices such as Chromebooks, there is a very high probability that a given user wants to view an agent based simulation on a mobile device or a Chromebook. Therefore it was decided to test all of these ABMS tools by viewing similar models on different types of mobile devices and Chromebooks. Therefore multiple Android phones, Android tablets, iPhones, iPads and Chromebooks were used for this test. All the ABMS tools considered for this survey passed this mobile browser and device compatibility test with Google Chrome and Mozilla Firefox web browsers on both Android and iOS (iPhone and iPad) devices, Safari web browser on iOS (iPhone and iPad) devices and Google Chrome web browser on Chromebook devices.

VI. Licence, Cost, Current Development Status and Affiliations

The type of the software licence of each ABMS tool, the cost, current development status and affiliations are very important factors when selecting an ABMS tool. These factors are very critical to the long-term success of an ABMS project which utilizes an ABMS tool. All the ABMS tools considered for this survey were available free of charge, at the time of this survey. However, there are differences in the software licences used and these differences in the licenses are important for developers who wish to distribute derived work from the source code of these ABMS tools. Table 2 presents a comparison of licences, current development status related information and affiliations for each ABMS tool.

Table 2 Licence, Current Development Status and Affiliations

ABMS Tool	Licence	Current Development Status and Affiliations
Agent Base	GNU General Public License, version 3	Latest Update: Feb, 2017 Source Code: https://github.com/wybo/agentbase Copyright (c) Wybo Wiersma, 2014 Affiliation: University of Oxford
Agent Script	GNU General Public License, version 3	Latest Update: Oct, 2020 Source Code: https://github.com/backspaces/agentscript Copyright (c) Owen Densmore, RedfishGroup LLC, 2012-2020 Affiliation: RedfishGroup LLC
Behaviour Composer	BSD License	Latest Update: May, 2015 Source Code: https://code.google.com/archive/p/modelling4all/source Written by Dr. Ken Kahn, Copyright (c) University of Oxford, 2014 Affiliation: University of Oxford
Insight Maker	Insight Maker Public License (a custom licence based on the Affero GPL)	Latest Update: October, 2020 Source Code: https://github.com/scottfr/insightmaker Copyright (c) Scott Fortmann-Roe, 2010-2020 Affiliation: insightmaker.com
Mesa	Apache License, Version 2.0	Latest Update: October, 2020 Source Code: https://github.com/projectmesa/mesa Copyright (c) Core Mesa Team, 2020 Affiliation: Project Mesa
NetLogo Web	GNU General Public License, version 2	Latest Update: September, 2020 Source Code: https://github.com/NetLogo/NetLogo Copyright (c) Uri Wilensky, 1999-2014 Affiliation: Northwestern University Center for Connected Learning and Computer-Based Modelling
StarLogo Nova	License information is not publicly available	Latest Update: November, 2018 StarLogo Nova source code and copyright details were not publicly available at the time of writing this paper Affiliation: Massachusetts Institute of Technology (MIT) Scheller Teacher Education Program

VII. Programming Languages and Terminology

The programming language used in an ABMS system in order to develop agent models is also a key deciding factor when selecting an ABMS tool. In some of the domains where agent based modelling and simulations are highly useful, such as biology and social science, the domain experts are not very familiar with the traditional programming languages used for computer programming. In such cases, visual programming approaches provided by some of the ABMS tools are preferred and some of these visual programming approaches are very convenient for users to successfully develop agent based models and simulations, with limited or no programming experience. Table 3 provides a comparison of the different programming languages used in the selected ABMS tools.

Table 3 Programming Languages Used in Selected ABMS Tools

ABMS Tool	Programming Language	Level of Traditional Programming Skills Required
AgentBase	CoffeeScript	High
AgentScript	JavaScript	High

Behaviour Composer	Provides a visual composer with visual elements such as prototypes, micro behaviours and attributes Also it is possible to add NetLogo code through micro-behaviours	Low
Insight Maker	Enables users to build models graphically and supports a rich set of diagramming features.	Low
Mesa	Python	High
NetLogo Web	NetLogo	Medium
StarLogo Nova	A blocks-based programming language, which uses blocks to put together puzzle-like pieces	Low

It was observed that many of the ABMS tools included in this survey were inspired by NetLogo and provides NetLogo like semantics for agents. Table 4 shows the terminology used by different ABMS tool to represent agents.

Table 4 Agent Based Modelling Terminology Used in Different ABMS Tools

ABMS Tool	Agent Based Modelling Related Terminology
AgentBase	Provides NetLogo like semantics for agents: *Turtles*, *Patches* and *Links*
AgentScript	Provides NetLogo like semantics for agents: *Turtles*, *Patches* and *Link*
Behaviour Composer	Agent are represented as *Prototypes*. *Attributes* and *Micro-behaviours* can be added to the *Prototypes*
Insight Maker	Agent Based Models are primarily built using *States*, *Transitions*, *Actions*, *Variables* and *Agent Populations*
Mesa	Agents are represented with the *Agent* class
NetLogo Web	Four types of agents are available: *Turtles* are agents that move around in the world, *Patches* have locations with coordinates, *Links* are connecting turtles and *Observers* give instructions to other agents
StarLogo Nova	Agents are represented as *Turtles*

VIII. Access Control and Model Sharing

Some of the ABMS tools require users to create a user account, in order to use the ABMS tool. These user account are used for access control and provide a persistent storage in some of the tools. Also this determines the ways a user can share models with other users. Table 5 compares the access control, storage and sharing related features of ABMS tools.

Table 5. Access Control, Storage and Sharing Related Features of ABMS Tools

ABMS Tool	User Account Required?	Can Store Models in an Online Storage?	Ways to Share Models
AgentBase	Yes	Yes	As a URL
AgentScript	No	No	As a URL
Behaviour Composer	No	No	As a URL to be automatically opened with NetLogo Web, embedded as iFrame or embedded as a link in a web site
Insight Maker	Yes	Yes	Embed Insight (model) in a website, export as a file or share as a *Storytelling* (step-by-step walk-through of the model.)
Mesa (with Jupyter notebook)	Depends on the access control of the Jupyter notebook	Depends on the access control of the Jupyter notebook	As a Jupyter notebook or as a URL
NetLogo Web	No	No	As a URL or export as a NetLogo file
StarLogo Nova	Yes	Yes	As a URL

IX. Availability of Classic Agent Based Models

All of the ABMS tools considered for this survey were available with a collection of sample agent based models. In addition to that some of the popular ABMS tools have community based online model repositories with agent based models created by community members. The availability of classic agent based models built using a given ABMS tool increases confidence of the users on the abilities of the ABMS tool. Table 6 shows the classic agent based models used to compare ABMS tools on the availability of classic agent based models.

Table 6 . List of Classic Agent Based Models Used for the Survey

Model	Description
Boids	*Boids* is an artificial life program, developed by Craig Reynolds [24] in 1986, which simulates the flocking behaviour of birds

Life	*Game of Life* is a cellular automaton devised by John Horton Conway [25] in 1970. It is an example of a zero-player game where the evolution is determined by its initial state, requiring no further input
Fire	*Forest Fire* model [26] simulates the spread of a fire through a forest. This model is defined as a cellular automaton on a grid with cells. A cell can have one of the three states: empty, occupied by a tree, or burning
Epidemic	*Epidemic* model [27] is a simplified simulation of the spread of an infectious disease in a closed population
Ants	*Ants* model [28] simulates a colony of ants forage for food. This model is an interaction between food sources, ants and chemicals dropped by ants
Segregation	*Schelling's Model of Segregation* is an agent based model developed by economist Thomas Schelling [29]
Predator and Prey	*Predator and Prey* model explores the stability of a predator and prey ecosystem. Wolf Sheep Predation [30] is a specific version this model which involves wolves and sheep
Prisoner's Dilemma	*The Prisoner's Dilemma* [31] is a popular example of a game analysed in game theory that shows why two completely rational individuals might not cooperate, even if it appears that it is in their best interests to do so

Table 7 compares the public availability of above mentioned classic agent based models built using the ABMS tools at the time of conducing this survey.

Table 7 Availability of Selected Classic Agent Based Models Built Using the ABMS Tools

ABMS Tool	Boids	Life	Fire	Epidemic	Ants	Segregation	Predator and Prey	Prisoner's Dilemma
AgentBase	✓	✓	-	-	✓	-	-	-
AgentScript	✓	-	✓	-	✓	-	-	-
Behaviour Composer	-	-	-	✓	-	-	✓	-
Insight Maker	-	✓	✓	✓	-	✓	✓	✓
Mesa	✓	✓	✓	-	-	✓	✓	✓
NetLogo Web	✓	✓	✓	✓	✓	✓	✓	✓
StarLogo Nova	✓	-	✓	✓	✓	✓	✓	✓

X. Learning Resources, User Community and Model Repositories

Availability of rich learning resources and user community are some major considerations when selecting an ABMS tool. Also the availability of model repositories helps users to view and learn from the models developed by other individuals. Table 8 presents the availability of learning resources, user community and model repositories.

Table 8 Learning Resources, User Community and Model Repositories

ABMS Tool	Amount of Tutorials and Documentation	Dedicated User Community Forum	Model Repositories
AgentBase	Small	-	Only a small number of models distributed with the tool
AgentScript	Small	-	Only a small number of models distributed with the tool
Behaviour Composer	Medium	✓	Several models and micro behaviours are available with Modelling4All Project
Insight Maker	Medium	✓	A large number of models created by users are available through the Explore Insight page of the insightmaker.com
Mesa	Small	-	Only a small number of models distributed with the tool
NetLogo Web	Large	✓	Over 1000 models created by users are available through multiple repositories including NetLogo User Community Models [32] and Modeling Commons [33]
StarLogo Nova	Large	✓	A large number of models created by users are available through the Star Logo Nova projects search

XI. Conclusion

Based on the findings of this survey it appears that each ABMS tool has both strengths and weaknesses based on the requirements of the user. All of the ABMS tools provide the ability to view and execute agent based models and simulations on web browsers. And all of these tools (except AgentScript) allow users to create and edit models on web browser. And every single ABMS tool selected for this survey supports 2D views on web browsers. However, only AgentScipt and StarLogo Nova provide 3D views on web browsers. If an agent based model requires support for GIS, it is advisable to use AgentScipt or Mesa (with mesa-geo GIS extension) tools which support GIS. All selected ABMS tools were available free of charge at the time of this survey and as a result there were no costs associated with using these tools.

The programming languages used to develop agent based models and simulations with these tools were compared and presented in this paper. This comparison helps users to select ABMS tools with the preferred

programming language. Also there are some ABMS tools available with visual programming environments to help users with little or no programming experience to effectively build agent based models and simulations. On the other hand, experienced developers may prefer to use a JavaScript, CoffeeScript or Python based ABMS tools with the flexibility to integrate those tools with existing software stacks. Furthermore, we reviewed the access controls available in each tool and ways to share models with others.

Finally, we compared the availability of classic agent based models, learning resources, user community forums and model repositories for each ABMS tool that can heavily influence a decision to select an ABMS tool for a long-term project. We believe that the comparisons provided by this survey will provide guidelines for individuals planing to use web browser-based ABMS tools. Also we hope this survey will provide a supplement to the existing surveys of ABMS tools.

REFERENCES

[1] Bora, S., Emek, S.: Agent-Based Modeling and Simulation of Biological Systems. In: Modeling and Computer Simulation, Dragan Cvetković, IntechOpen (2018)

[2] Grimm V., Railsback S.F.: Agent-Based Models in Ecology: Patterns and Alternative Theories of Adaptive Behaviour. In: Billari F.C., Fent T., Prskawetz A., Scheffran J. (eds) Agent-Based Computational Modelling. Contributions to Economics, pp 139--152. Physica-Verlag HD (2006)

[3] Klein, D., Marx, J., Fischbach, K.: Agent-Based Modeling in Social Science, History, and Philosophy. An Introduction. In: Historical Social Research 43 (2018)

[4] Abar, S.,Theodoropoulos, G., Lemarinier, P., O'Hare, G.:Agent Based Modelling and Simulation tools: A review of the state-of-art software. In: Computer Science Review (2017)

[5] Allan, R.J.: Survey of Agent Based Modelling and Simulation Tools. In: Technical Report DL-TR-2010-007, Science and Technology Facilities Council (2009)

[6] Tobias, R., Hofmann, C.: Evaluation of free Java-libraries for social-scientific agent based simulation. In: Journal of Artificial Societies and Social Simulation, vol. 7 (2004)

[7] Kravari, K., Bassiliades, N.: A Survey of Agent Platforms. In: Journal of Artificial Societies and Social Simulation. 18 (2015)

[8] Bordini, R., Lars, B., Dastani, M., Seghrouchni, A., Gómez-Sanz, J., Leite, J., O'Hare, G., Alexander, P., Ricci, A.: A Survey of Programming Languages and Platforms for Multi-Agent Systems. In: Informatica 30 (2006)

[9] Nikolai, C., Madey, G.: Tools of the Trade: A Survey of Various Agent Based Modeling Platforms. In: Journal of Artificial Societies and Social Simulation, vol. 12 (2009)

[10] Wiersma W.: AgentBase: Agent Based Modeling in the Browser. In: Jager W., Verbrugge R., Flache A., de Roo G., Hoogduin L., Hemelrijk C. (eds) Advances in Social Simulation 2015. Advances in Intelligent Systems and Computing, vol 528. Springer, Cham (2017)

[11] AgentBase, http://agentbase.org/

[12] Tisue, S., Wilensky, U.: NetLogo: A simple environment for modeling complexity. In: International Conference on Complex Systems (2004)

[13] AgentScript, http://agentscript.org/

[14] Kahn, K. Noble, H.: The Modelling4All project a web-based modelling tool embedded in Web 2.0. In: Proceedings of the 2nd International Conference on Simulation Tools and Techniques for Communications, Networks and Systems, SimuTools (2009)

[15] Modelling4All Project, http://m.modelling4all.org/

[16] NetLogo Web, https://www.netlogoweb.org/

[17] Fortmann-Roe, S.: Insight Maker: A general-purpose tool for web-based modeling & simulation. In: Simulation Modelling Practice and Theory, vol. 47, pp. 28--45 (2014)

[18] Insight Maker, https://insightmaker.com/

[19] Masad, D., Kazil, J.: Mesa: An Agent-Based Modeling Framework. In: Python in Science Conference (2015)

[20] Mesa: Agent-based modeling in Python 3+, https://mesa.readthedocs.io/en/master/

[21] StarLogo Nova, https://www.slnova.org/

[22] Klopfer E., Scheintaub H., Huang W., Wendel D.: StarLogo TNG: Making Agent-Based Modeling Accessible and Appealing to Novices. In: Komosinski M., Adamatzky A. (eds) Artificial Life Models in Software. Springer, London (2009)

[23] Project Jupyter, https://jupyter.org/

[24] Reynolds, C. W.: Flocks, Herds, and Schools: A Distributed Behavioral Model. In: Computer Graphics, 21(4) (SIGGRAPH '87 Conference Proceedings), pp. 25--34. (1987)

[25] Gardner, M.: Mathematical Games: The fantastic combinations of John Conway's new solitaire game "Life". In: Scientific American, vol. 223, pp. 120--123 (1970)

[26] Wilensky, U.: NetLogo Fire model, Center for Connected Learning and Computer-Based Modeling, Northwestern University, http://ccl.northwestern.edu/netlogo/ models/Fire

[27] Yang, C., Wilensky, U.: NetLogo epiDEM Basic model, Center for Connected Learning and Computer-Based Modeling, Northwestern University, http://ccl. northwestern.edu/netlogo/ models/epiDEMBasic

[28] Wilensky, U.: NetLogo Ants model, Center for Connected Learning and Computer-Based Modeling, Northwestern University, http://ccl.northwestern.edu/netlogo/ models/Ants

[29] Schelling, T.C.: Dynamic models of segregation. In: The Journal of Mathematical Sociology, vol. 1, pp. 143--186 (1971)

[30] Wilensky, U.: NetLogo Wolf Sheep Predation model, Center for Connected Learning and Computer-Based Modeling, Northwestern University, http://ccl.northwestern. edu/netlogo/ models/ WolfSheepPredation

[31] Prisoner's Dilemma, https://plato.stanford.edu/entries/prisoner-dilemma/

[32] NetLogo User Community Models, http://ccl.northwestern.edu/netlogo/models/ community/index.cgi

[33] Modeling Commons, http://modelingcommons.org