2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C) | 978-1-6654-2484-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/MODELS-C53483.2021.00105

Domain-Specific Modelling Languages for Participatory Agent-Based Modelling in Healthcare

Thomas Godfrey
Department of Informatics
King's College London
London, United Kingdom
thomas.1.godfrey@kcl.ac.uk

***Abstract*—Hospitals need to look into operational interventions to keep up with increasing patient numbers, and the threats of infectious diseases. Changes such as new ward layouts, staff allocations, testing schemes, admission pathways, etc., need to be considered to improve patient flow and safety. Agent-Based Modelling (ABM) has promising potential for simulating these types of interventions in hospital settings without interfering with real-world operations. While this approach to simulation has seen great academic support, the acceptance of these simulations in practice is still relatively low. Typical software development processes, and the use of general-purpose programming languages, make simulation implementations largely inaccessible to their intended users. Explaining model behaviour becomes challenging, and domain users are unable to modify or develop their own models without handing over to software developers. We propose that participatory modelling, aided by a suite of domain-specific modelling languages can help resolve these issues. We outline a set of DSMLs to involve users directly in model development with the aim of improving implementation comprehensibility, improving model development times, and improving overall simulation acceptance in a healthcare setting.**

***Index Terms*—Domain-Specific Modelling Languages, Agent-Based Modelling, Simulation, Healthcare, Participatory Modelling**

I. Problem

Hospitals are under pressure to keep up with increasing patient attendances [1], and the risks presented by infectious diseases such as COVID-19 [2]. Simulations such as agent-based models (ABM) have shown promising potential as a tool to assess operational interventions in these settings, by providing a cost-effective, safe alternative to *in vivo* experiments. However, the practicalities of developing ABMs makes real-world adoption challenging. Development of models in general-purpose programming languages is tedious, time-consuming, and largely inaccessible to the domain stakeholders. It is uncommon for software engineers to be sufficiently familiar with clinical processes, and it is similarly uncommon for clinical stakeholders to have experience in software development, resulting in an issue of communication between the two parties.

Participatory modelling [3] can be used to help bridge communication issues between these domains. Typically, this involves close collaboration between developers and the domain users in the design and implementation of models. These could include informal models such as sketches and flowcharts. or semi-formal approaches such as UML [4] for class diagrams, interaction diagrams etc. These models can then be translated into a concrete implementation in code. In informal models, this translation can be error-prone and subject to ambiguities. UML can also be too general to facilitate effective communication of the domain. Hospital-centred concepts cannot be directly represented in UML alone, and the language's notations and technical formalisms can be unapproachable to non-technical users. This can limit the participation of healthcare professionals, hinder their understanding of the models being developed, and limit the utility of the models for their purposes.

II. Related Work

A. Agent-Based Simulations in Healthcare

Developing a simulation involves developing a model, and therefore an abstraction of, a real-world system in order to provide insight into system behaviour or to test an intervention to how the system operates [5]. The critical nature of hospital processes makes implementing changes *in vivo* challenging and so simulation is an attractive approach in this domain. The most popular simulation techniques in healthcare include discrete event simulation (DES) [6], monte carlo simulation (MCS) [7], and agent-based modelling (ABM) [8].

Agent-based models capture a system as a set of agents within a set environment. Agents are individuals, such as staff or patients, who have autonomy in system processes and go through an iterative process of perceiving the environment, deliberating about an action, and then enacting that action in pursuit of goals either independently or as a group [9]. While the model of a single agent can be relatively simple, collections of socially interacting agents in the environment can produce complex emergent behaviour. The ability for agent-based modelling to capture these emergent behaviour is one of its driving advantages [10].

We believe ABM is the most appropriate approach for modelling healthcare systems. Monte carlo simulations are probabilistic and are well suited for investigations such as cost-benefit analysis [8], however the models are typically fairly simplistic and are not suitable for capturing a system as a whole. Rather, they are better suited for modelling individual stochastic processes within larger systems [11]. Healthcare systems contain a wide variety of staff types,

978-1-6654-2484-4/21/$31.00 ©2021 IEEE
DOI 10.1109/MODELS-C53483.2021.00105

therefore requiring a large degree of heterogeneity between the actors represented in a model. DES models typically capture actors as normative homogeneous entity groups. In contrast, ABM allows the modeller to capture properties at an individual level, allowing each agent to have their own characteristics and behaviour.

ABM uses a bottom-up approach to modelling that captures systems as a product of the people involved and their individual responsibilities. This is a natural description of a system which may be more accessible to users [12]. Agents in ABMs are defined by human-centred terms such as 'behaviour', 'decision-making', or in the case of BDI-agents, 'beliefs', 'desires', 'intentions' [13]. These terms can be more accessible to domain users compared to differential equations or probability functions found in other simulation approaches such as MCS or DES.

B. Simulation Acceptance in Healthcare

While simulation has seen good uptake in domains such as manufacturing and military defence, the same is not true in healthcare [14]. Young identifies the unique cultural factors in healthcare, where stakeholders tend to be more reluctant to move away from their existing 'known and safe' systems [15]. Lowery et al. also highlight how the technical nature of simulation can create barriers to adoption [16].

The degree to which users view simulation with credibility can contribute significantly to acceptance [17]. For healthcare professionals to be able to assess the credibility of a model, there needs to be transparency in model implementations and outputs. Liu et al. present a model that attempts to address this by allowing the user to define 'sensors' on agent and environment properties, exposing micro-level ABM behaviour during simulation runs [18]. While this approach improves the transparency for how agent properties may be contributing to observed model outputs, it is not possible to trace how agent behaviours that utilise those properties are implemented. It is difficult for users to assess whether the agent implementation is reflective of real world behaviour. The domain users must trust that the simulation developers have correctly understood and translated the domain processes into a technical implementation.

C. Domain-Specific Modelling Languages

In the majority of cases, ABMs are developed in either a GPL or in an agent-based platform such as Repast Simphony and NetLogo [19], [20]. These platforms use high-level statements to fulfil ABM requirements such as scheduling, message passing, and learning, while abstracting away from the precise implementation details underpinning them. While such languages can improve development productivity and understandability [21], the technical knowledge required to use them is still relatively high. In the healthcare domain, stakeholders are unlikely to have sufficient experience with programming and so may struggle with the technical nature of the languages used in these frameworks.

Domain-Specific Modelling Languages (DSML) can be used to model a system using terms and notations found in the target domain [22]. DSMLs are defined by their concrete syntax, abstract syntax, and semantics [23]. The *concrete syntax* is the grammar of the language, defining how the language appears to the user. The *abstract syntax* is often represented in an abstract syntax tree (AST), and is effectively a meta-model defining what concepts are captured in the language. The *semantics* then describe how the language should be 'understood' when executed.

DSMLs are purposefully more restrictive than General-Purpose Languages (GPL) such as Java [24] making them smaller and more focused languages. This can potentially make them more comprehensible and accessible to users. For non-technical users, languages should expose as little technical complexity as possible while still offering sufficient flexibility for the users to develop the scope of models they are interested in. Languages such as SCAMP [25] have successfully demonstrated the potential for DSMLs for non-programmers in the social science domain, but, to the best of our knowledge, there have not been any DSMLs targeted specifically for modelling healthcare systems.

III. Proposed Solution

A. Domain-Specific Modelling Languages

To tackle the issues outlined above, we will develop a set of high-level DSMLs to assist with participatory modelling of healthcare-based ABMs. DSMLs will allow healthcare professionals to capture models of the domain in familiar notations during simulation design and automatically translate those models into a platform-specific implementation. We hope to improve user engagement in the development of simulations, improve their understanding of how the models are created, and allow the users to modify and create simulations themselves.

To develop these DSMLs, we will use Jetbrains MPS [26]. MPS is a relatively flexible IDE for developing DSLs. The DSLs developed in MPS use a projectional editor thus avoiding the complexities and limitations of parsing. The user is able to interact directly with a projection of the language abstract syntax tree which allows for a more diverse range of notations to be used in the DSL, including non-textual concrete syntaxes. We hope that using a variety of both graphical and textual notations in our languages will improve the visual expressiveness, and accessibility, of the DSMLs.

Different case study models will require different modelling needs. For example, in models of infection risk, the layout of the environment and relative position of agents within that environment will be important concepts to configure in the DSMLs. However, in models of diagnostic decision-making, spatial layout may be less important. A single DSML would be too cumbersome to cover the entire scope of hospital domain concepts. We therefore separate our DSMLs into a hierarchy of libraries that can be instantiated as required (Fig. 1).

At the bottom level is the agent-based platform written in Repast Simphony, into which our DSMLs inject generated

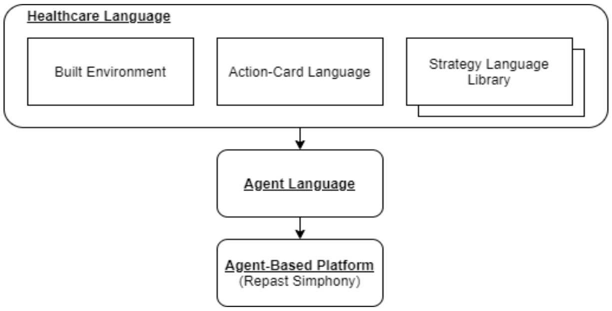

Fig. 1. Architecture of the DSML languages.

code. Atop this, we have our first DSML, the Agent Language. The Agent Language is effectively an ‘actor language’ designed to encapsulate the core concepts of agents in ABMs [27]. This language includes the concepts of `Agents`, numeric `Attributes` associated with that agent (such as energy level etc.), `Behaviour` (a type of action performed by the agent), `Role` (the responsibilities of the agent, used to define what `Behaviours` are available to them). `Behaviours` are triggered by receipt of a message, as defined in that `Behaviour`’s `MessageTrigger`. A `Behaviour` contains a list of basic `BehaviourElements` which can include setting the agent to `Stay` in a location until some `Condition` on the agent’s `Attribute`(s) evaluates to true, `Move` to another location or `SendMessage` to another agent to trigger a receiving agent’s `Behaviour`. The meta-model for this language is shown in Fig. 2.

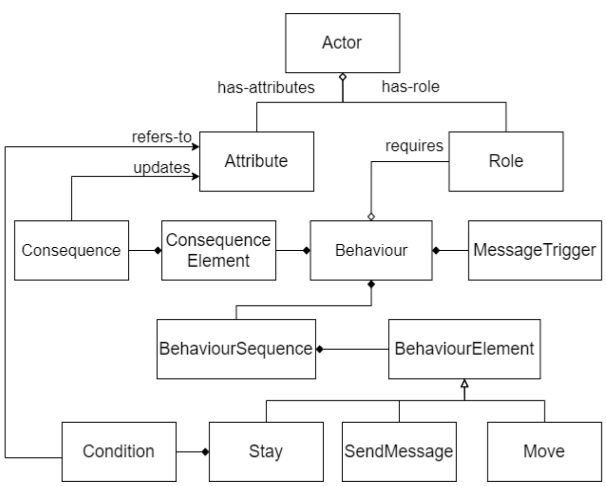

Fig. 2. Meta-model for the Agent Language.

We then have a modular set of languages at the top level in a collection called the ‘Healthcare Language’. These languages will cover concepts such as patient properties, staff types, hospital built environment, decision-making strategies, and specific communication channels. The final design of each of these languages is yet to be finalised, and will be developed in collaboration with the relevant hospital staff. However, we present an initial vision for these languages below.

The Action-Card Language will contain the concepts for the notion of action cards. Action cards are flowchart-like information visualisations available on the intranet in NHS hospitals. They define the flow of operations that need to be followed by staff depending on the presenting condition of patients. This includes the treatments that patients need to receive and the individual actions staff need to undertake. The Action-Card DSML will be designed to closely resemble these action cards, and so will be the first of our graphical languages. A sketch example of the concrete syntax for this DSML is shown in Fig. 3. Each `Action` in the `Action Card` is visualised as a rectangle, and contains a title as well as defining properties including the `Staff` type who performs the action, a boolean `Requires Patient` flag (to denote whether the process requires the patient to be present with the staff member(s)), the `Location` the action is performed in, and a set of `Action Steps` selected from a pre-defined list. The `Action Steps` are used to link `Actions` through conditional statements with `Branch`es associated with each conditional outcome. For example, in Fig. 3 there is an `Action Step` to give the patient a lateral flow test. The outcome of the test could be either positive or negative, and in each case a `Branch` is used to refer to the relevant proceeding `Action`.

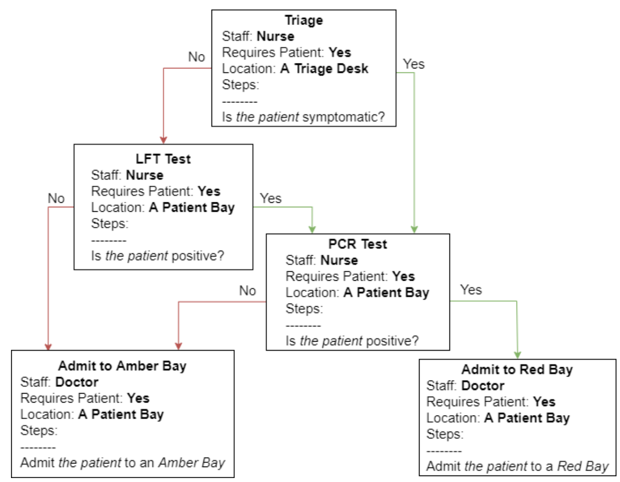

Fig. 3. Sketch of the Action-Card Language.

The Strategy Languages will contain distinct DSMLs for different auxiliary actions. This can include the strategies that staff make in selecting the next patient they need to see, how they select what action they should perform with that patient, or the specific details of how they communicate with other staff. The built environment will contain the concepts for defining the hospital environment including the types and position of `Locations` used in the Action-Card Language.

All languages will compile down into the language beneath, either directly or through weaving with other DSMLs at the same level. For example, for each `Staff` present in the Action-Card Language, a `Role` will be constructed in the Agent Language. Each of the `Actions` associated with that `Staff` will be translated into `Behaviours` for that `Role`. `Branch`es between `Actions` will be translated into `SendMessage` and `MessageTrigger` pairs in the `Branch` source and target `Behaviours`.

B. Agent-Based Modelling

As we highlighted in section II, we believe that agent-based modelling will be the most suitable form of simulation for the healthcare domain. We have decided to use the agent-based platform, Repast Simphony [19]. Repast provides the core agent-level functionality we require in our modelling while being generic enough to not constrain us to a particular domain. Scheduling, agent aggregations, logging, graphical interfaces and data visualisation are all provided in the platform. Simulation logging and visualisations will help the domain experts verify model behaviours during simulation execution. Repast Simphony also has good integration with R packages (such as SPARTAN [28] or RRepast [29]) which can help with semi-automated sensitivity analysis, determining the minimum number of required runs, and validating model outputs against real-world data. Lee et al. suggest that ABM platforms with features such as data analysis and visualisation can be effective for participatory modelling, allowing better involvement of stakeholders in model development [30].

We intend for the developed ABMs to be used for testing 'what-if' scenarios relating to hospital operations and the risks of infectious diseases. This can include alterations to patient care pathways, patient attendance profiles, new spatial layouts, staff allocations etc. Outputs from the developed simulations will be used to support decision making on operational changes in the hospital, and to provide an alternative view of hospital processes for different staff to encourage analysis and discussion.

C. Participatory Modelling

By integrating DSMLs with participatory modelling we aim to encourage more effective communication between stakeholders, better understanding of the model and domain, and more efficient development of ABMs. Our research will follow an action research approach to test this aim [31]. We will be conducting iterative cycles of problem-solving and reflection. Our problem-solving cycles will involve collaborating with hospital staff on a case study ABM simulation that fits their needs at the time. These case studies will be developed with participatory modelling as a focus and we will evaluate to what degree our DSMLs were able to enhance the properties of participatory modelling. Namely, the understanding and sharing of domain knowledge, and assessing the impact of our solutions to the stakeholders' problems [3]. We will investigate the degree to which DSMLs can effectively capture domain concepts, improve comprehensibility of ABMs, and encourage an appropriate level of trust in simulation results.

IV. Plan for Evaluation and Validation

For this project, we are concerned with the application of DSML technology in pursuit of the goal to *improve trust and acceptance of ABM simulation in the healthcare domain*. To test this goal, we will work in collaboration with St. Thomas Hospital in London. We view our goal as being composed of two objectives:

- To improve stakeholder understanding of the ABM simulations we develop
- To demonstrate the utility of simulation as a tool to stakeholders

The above objectives demonstrate our assumption that the trust stakeholders have in simulation is dependent on their understanding of how the simulation itself works, and that the acceptance of simulation is dependent on the clinical benefit afforded by it. During the project, we will continue to revise these assumptions as interventions and case studies develop.

While the research focus of this work will be on the design and evaluation of DSMLs in assisting participatory agent-based modelling, stakeholders in the problem domain are concerned with the utility and problem-solving potential of the simulations we develop. We seek to balance these viewpoints by developing distinct case study models in the hospital to demonstrate their clinical benefit. We aim to identify a key use case for which ABM would be suitable, develop the model within a reasonable turnaround time, validate the model, and present results to the stakeholders.

Our first objective concerns the ability for healthcare professionals to understand the simulations that we produce. By this, we mean that staff such as NHS managers, infection control staff, flow coordinators, consultants, will be able to identify suitable studies for ABM, will be able to understand how model outputs are produced, and will be able to modify simulation implementations for their own case studies. Currently, these staff at St. Thomas Hospital are not familiar with the use of simulation in their work. Although staff use IT systems for patient records and communications, there is not currently any simulation modelling in place for them to test 'what-if' scenarios, etc.

The effectiveness of the DSMLs we develop directly influences the effectiveness to which we reach our first objective. We will go through cycles of developing the DSMLs for case study simulations, and then evaluate the language quality and usability. To evaluate language quality, we will utilise the adapted cognitive dimensions framework introduced by Izquierdo and Cabot in their Collaboro meta-modelling tool [32]. The metrics are used to evaluate the quality of a DSML's concrete syntax including the expressiveness of language concepts, the discriminality between concept representations, and the avoidance of using the same visual element for more than one concept.

Beyond this, we will also evaluate the usability of the DSMLs by conducting systematic experiments of real-world usage. We will base these experiments on the evaluation framework set out by Miranda et al. [33]. This framework includes experiments, conducted with language users, for comparing the efficacy, efficiency, comprehensibility, and perceived usability of different DSMLs. For example, one experiment involves providing users with the definition of a language concept, and asking them to identify which concrete syntax element represents that concept. In another experiment, the users are given a task to model a particular scenario using the language, and are then given a questionnaire to assess their perception of the language's usability. We will use this evaluation framework for each significant iteration of DSML we develop, comparing the new language version to the previous.

Our second objective is concerned with demonstrating the clinical utility of simulation in a healthcare setting. This objective involves an investigation into the feasibility and efficacy of integrating ABM simulation as an intervention tool for hospitals. We will assess this by monitoring if: the models we produce were developed within the users' required turnaround time, if the model outputs are explainable to the users and are generalisable for their purposes, and if the model was able to meaningfully impact clinical practise. It is not sufficient for a model to be 'correct' if it takes a prohibitive amount of time to develop, does not provide explainable results, and is over-fit to a specific test environment, thus restricting the predictive power of the model.

We have weekly meetings with clinical teams to discuss their clinical needs, and provide feedback of our research findings. This offers opportunities for us to discuss model developments and the outputs of simulation experiments. These informal discussions will be coupled with individual interviews with staff to form a subjective analysis of the perceived clinical benefit of the models we develop. We will conduct model validations to determine if micro-level and macro-level model behaviour match real-world observations [34]. We will also conduct one-factor-at-a-time sensitivity analysis to measure the robustness of model outputs according to implementation inputs and assumptions [35].

V. Expected Contributions

Our main contribution will be a family of DSMLs designed for healthcare professionals to develop their own agent-based models of hospital processes. These DSMLs will be developed in close collaboration with the intended users to ensure the languages are fit for purpose. This includes ensuring that language concepts have sufficient coverage over the domain space to facilitate the scope of models the users are interested in. We will also focus on usability concerns including comprehensibility and expressiveness of the concrete syntaxes as described in section IV.

We will develop a number of case study ABMs as informed by the needs of the healthcare professionals at the time. These will be developed with participatory modelling as a focus, ensuring that the users are involved at all stages of development and evaluation. A concrete contribution of this work, therefore, will be a set of ABMs of distinct hospital-based case studies.

A research contribution of this work will be to test the hypothesis that 'improved visibility and explainability of model implementations will improve the trust that simulation users have for the model'. We will also test the hypothesis that 'the use of DSML will improve efficiency of model development'. We hope to demonstrate the feasibility of a full-scale DSML for healthcare-focused agent-based modelling, and identify the most appropriate form these DSMLs should take to be suitable for healthcare users.

VI. Current Status

A. Progress So Far

The first step of this project was to identify the requirements for DSMLs in the hospital domain. As a research group, we have previous experience in developing a range of ABM case studies allowing us to assess the scope of studies that healthcare professionals are interested in. We were able to identify the domain concepts that re-occur across studies and define a vision for a family of DSMLs to cover those concepts. Initial design details for these DSMLs are shown in section III.

So far, the agent-based platform implementation and the Agent Language have been developed and successfully used in an initial case study. This case study involved testing the efficacy of a new COVID-19 testing scheme in the St. Thomas' emergency department (ED) at different virus prevalence rates. The study was an opportunity to build an initial version of our DSMLs through discussions with ED and infection control staff to understand their requirements.

B. Next Steps

The next step is to implement the Action-Card Language in time for Winter 2021. We will engage in developing a new case study simulation during the Winter in collaboration with the St. Thomas' ED. It is expected that this will be an extension of the COVID-19 testing scheme study described above, but the specifics of the study will be formalised once the needs of the ED are assessed at the time. We are also hoping to engage with the Intensive Care Unit (ICU) in a separate case study. This new environment will provide an opportunity to identify which, if any, domain concepts are unique to either the ED or ICU, and allow us to evaluate if our current DSMLs are appropriate for both of these domains.

Decisions relating to the addition or modification of our DSMLs will be informed by the results of our evaluation questions, outlined in section IV. This evaluation cycle will be completed during the Summer of 2022. The cycle of development and evaluation will be repeated over the following year, with the specifics of which case studies, and areas of the hospital, we engage with being decided as needs arise. I plan to submit this thesis no later than April 2024.

Acknowledgment

I would like to thank the CIDR and Emergency Department staff at St. Thomas' Hospital for their engagement with this project. I would also like to thank my supervisors, Steffen Zschaler, Jonathan Edgeworth, and Simon Miles for their invaluable support and guidance. Funding was provided by KCL DRIVE-Health CDT (drive-health.org).

REFERENCES

[1] Department of Health, *The NHS Plan: a plan for investment, a plan for reform*. Department of Health, 2000.

[2] F. Zhou, T. Yu, R. Du, G. Fan, Y. Liu, Z. Liu, J. Xiang, Y. Wang, B. Song, X. Gu *et al.*, "Clinical course and risk factors for mortality of adult inpatients with covid-19 in wuhan, china: a retrospective cohort study," *The Lancet*, vol. 395, no. 10229, pp. 1054–1062, 2020.

[3] A. Voinov and F. Bousquet, "Modelling with stakeholders," *Environmental Modelling & Software*, vol. 25, no. 11, pp. 1268–1281, 2010.

[4] J. Rumbaugh, I. Jacobson, and G. Booch, *The Unified Modeling Language Reference Manual (2nd Edition)*. Pearson Higher Education, 2004.

[5] R. E. Shannon, "Systems simulation; the art and science," Tech. Rep., 1975.

[6] M. M. Günal and M. Pidd, "Discrete event simulation for performance modelling in health care: a review of the literature," *Journal of Simulation*, vol. 4, no. 1, pp. 42–51, 2010.

[7] S. Raychaudhuri, "Introduction to monte carlo simulation," in *2008 Winter simulation conference*. IEEE, 2008, pp. 91–100.

[8] K. Katsaliaki and N. Mustafee, "Applications of simulation within the healthcare context," *Journal of the operational research society*, vol. 62, no. 8, pp. 1431–1451, 2011.

[9] M. Wooldridgey and P. Ciancarini, "Agent-oriented software engineering: The state of the art," in *International workshop on agent-oriented software engineering*. Springer, 2000, pp. 1–28.

[10] A. T. Crooks and A. J. Heppenstall, "Introduction to agent-based modelling," in *Agent-based models of geographical systems*. Springer, 2012, pp. 85–105.

[11] M. S. Rauner, K. Heidenberger, and E.-M. Pesendorfer, "Model-based evaluation of diabetic foot prevention strategies in austria," *Health care management science*, vol. 8, no. 4, pp. 253–265, 2005.

[12] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," *Proceedings of the national academy of sciences*, vol. 99, no. suppl 3, pp. 7280–7287, 2002.

[13] A. S. Rao and M. P. Georgeff, "Modeling rational agents within a bdi-architecture." in *Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference (KR91)*, J. Allen, R. E. Fikes, and E. Sandewall, Eds. San Mateo: Morgan Kaufmann Publishers, 1991, pp. 473–484.

[14] S. C. Brailsford, T. Bolt, C. Connell, J. H. Klein, and B. Patel, "Stakeholder engagement in health care simulation," in *Proceedings of the 2009 Winter Simulation Conference (WSC)*. IEEE, 2009, pp. 1840–1849.

[15] T. Young, J. Eatock, M. Jahangirian, A. Naseer, and R. Lilford, "Three critical challenges for modeling and simulation in healthcare," in *Proceedings of the 2009 Winter Simulation Conference (WSC)*. IEEE, 2009, pp. 1823–1830.

[16] J. C. Lowery, "Introduction to simulation in health care," in *Proceedings of the 28th conference on Winter simulation*, 1996, pp. 78–84.

[17] B. S. Onggo, L. Yilmaz, F. Klügl, T. Terano, and C. M. Macal, "Credible agent-based simulation–an illusion or only a step away?" in *2019 Winter Simulation Conference (WSC)*. IEEE, 2019, pp. 273–284.

[18] Z. Liu, D. Rexachs, E. Luque, F. Epelde, and E. Cabrera, "Simulating the micro-level behavior of emergency department for macro-level features prediction," in *2015 Winter Simulation Conference (WSC)*. IEEE, 2015, pp. 171–182.

[19] M. J. North, N. T. Collier, J. Ozik, E. R. Tatara, C. M. Macal, M. Bragen, and P. Sydelko, "Complex adaptive systems modeling with repast simphony," *Complex adaptive systems modeling*, vol. 1, no. 1, pp. 1–26, 2013.

[20] U. Wilensky, "Netlogo (and netlogo user manual)," *Center for connected learning and computer-based modeling, Northwestern University. http://ccl. northwestern. edu/netlogo*, 1999.

[21] J. Gray, K. Fisher, C. Consel, G. Karsai, M. Mernik, and J.-P. Tolvanen, "Dsls: the good, the bad, and the ugly," in *Companion to the 23rd ACM SIGPLAN conference on Object-oriented programming systems languages and applications*, 2008, pp. 791–794.

[22] M. Mernik, J. Heering, and A. M. Sloane, "When and how to develop domain-specific languages," *ACM computing surveys (CSUR)*, vol. 37, no. 4, pp. 316–344, 2005.

[23] F. P. Andrés, J. De Lara, and E. Guerra, "Domain specific languages with graphical and textual views," in *International Symposium on Applications of Graph Transformations with Industrial Relevance*. Springer, 2007, pp. 82–97.

[24] K. Arnold, J. Gosling, and D. Holmes, *The Java programming language*. Addison Wesley Professional, 2005.

[25] H. V. D. Parunak, "Social simulation for non-hackers," *K.H. Van Dam, N. Verstaevel (eds.) 22nd International Workshop on Multi-Agent-Based Simulation (MABS 2021). Springer (2021)*.

[26] V. Pech, A. Shatalin, and M. Voelter, "Jetbrains mps as a tool for extending java," in *Proceedings of the 2013 International Conference on Principles and Practices of Programming on the Java Platform: Virtual Machines, Languages, and Tools*, 2013, pp. 165–168.

[27] G. Agha, "An overview of actor languages," *ACM Sigplan Notices*, vol. 21, no. 10, pp. 58–67, 1986.

[28] K. Alden, M. Read, J. Timmis, P. S. Andrews, H. Veiga-Fernandes, and M. Coles, "Spartan: a comprehensive tool for understanding uncertainty in simulations of biological systems," *PLoS computational biology*, vol. 9, no. 2, p. e1002916, 2013.

[29] A. P. García and A. Rodríguez-Patón, "Analyzing repast symphony models in r with rrepast package," *bioRxiv*, p. 047985, 2016.

[30] J.-S. Lee, T. Filatova, A. Ligmann-Zielinska, B. Hassani-Mahmooei, F. Stonedahl, I. Lorscheid, A. Voinov, J. G. Polhill, Z. Sun, and D. C. Parker, "The complexities of agent-based modeling output analysis," *Journal of Artificial Societies and Social Simulation*, vol. 18, no. 4, 2015.

[31] D. E. Avison, F. Lau, M. D. Myers, and P. A. Nielsen, "Action research," *Communications of the ACM*, vol. 42, no. 1, pp. 94–97, 1999.

[32] J. L. C. Izquierdo and J. Cabot, "Collaboro: a collaborative (meta) modeling tool," *PeerJ Computer Science*, vol. 2, p. e84, 2016.

[33] T. Miranda, M. Challenger, B. T. Tezel, O. F. Alaca, A. Barišić, V. Amaral, M. Goulao, and G. Kardas, "Improving the usability of a mas dsml," in *International workshop on engineering multi-agent systems*. Springer, 2018, pp. 55–75.

[34] F. Klügl, "A validation methodology for agent-based simulations," in *Proceedings of the 2008 ACM symposium on Applied computing*, 2008, pp. 39–43.

[35] G. Ten Broeke, G. Van Voorn, and A. Ligtenberg, "Which sensitivity analysis method should i use for my agent-based model?" *Journal of Artificial Societies and Social Simulation*, vol. 19, no. 1, p. 5, 2016.