Integrating optimisation and agent-based modelling

Peter George Johnson
Tina Balke
Centre for Research in Social Simulation, University of Surrey, Guildford, UK
Lars Kotthoff
Cork Constraint Computation Centre, University College Cork, Cork, Ireland

KEYWORDS

agent-based modelling, optimisation, incentive design, photovoltaic, policy modelling

ABSTRACT

A key strength of agent-based modelling is the ability to explore the upward-causation of micro-dynamics on the macro-level behaviour of a system. However, in policy contexts, it is also important to be able to represent downward-causation from the macro and meso-levels to the micro, and to represent decision-making at the macro level (i.e., by governments) in a sensible way. Though we cannot model political processes easily, we can try to optimise decisions given certain stated goals (e.g., minimum cost, or maximum impact). Optimisation offers one potential method to model macro-level decisions in this way. This paper presents the implementation of an integration of optimisation with agent-based modelling for the example of an auction scenario of government support for the installation of photovoltaic solar panels by households. Auction type scenarios of this kind, in which large groups of individuals or organisations make bids for subsidies or contracts from government, are common in many policy domains.

Introduction

Agent-based modelling (ABM) is an increasingly popular technique in the social sciences to evaluate the effect of policies and other instruments that affect groups of people. This is a result of the fact that ABM is well suited to exploring the macro-level behaviour of a system resulting from the micro-dynamics of the agents represented. Many heterogenous and autonomous agents and their interaction with one another, as well as the environment they act in, are relatively easily represented and the resulting models represent upward-causation well. However, in policy contexts it is common for downward-causation to also play a central role, and thus it is important for this to be included in any policy relevant ABM. Thus, a policy ABM may need to include a 'policymaker' type agent, and a process through which its decisions affect other agents. Representing the decision-making process of such a policymaker agent is likely to be difficult, reflecting the political and complex nature of policy-making. One option is to focus on a small selection of central goals a policymaker may have, such as minimising the cost of a policy, maximising the effect, or optimising with respect to another indicator (e.g., environmental quality).

Optimisation technologies are well-established in computer science and artificial intelligence to select the optimal element(s) or solution(s) (with regard to some goals) from some set of available alternatives. In many policy examples, there is a choice of possible decisions to make and actions to take given the current state of the system. Typically, these decision and action alternatives can be distinguished by the "value" they add to the overall goals of the policy makers. In such cases, it is desirable to make the decision that optimises the value with respect to the goal. This is an optimisation problem.

A typical example of an optimisation problem in this context is where a set of companies are putting in bids for contracts to provide a certain service that should be realised as a result of a policy decision. Usually, cost is the primary optimisation criterion – the service should be provided as cheaply as possible. However, the adequacy of the service has to be ensured as well. If only one company can get the contract, the problem is usually easy to solve – the lowest bidder that maintains the required standards gets the contract.

However, in many scenarios, it is not that simple. The service to provide may be complex and consist of several sub-services. Companies bid to provide those sub-services and it is the task of the policy maker to ensure that everything necessary is provided in the end. Instead of simply minimizing the cost, other considerations need to be taken into account now, making the problem much harder to solve. This is where optimisation technology is useful and can make life much easier for the policy maker.

This paper presents a method for the integration of this type of optimisation with an ABM built in NetLogo. This powerful method offers policy modellers a desirable option for representing decision-making at the macro-level. To illustrate our approach the example case of government support for household installation of photovoltaic (PV) solar panels is used. In this case, households/agents make bids for government support, and the optimisation selects a subset of these and 'gives' out support in such a way that the goal set out by the user (e.g. minimization of costs) is achieved. This type of auction scenario is particularly common in many policy domains.

Although it might be possible to implement optimisation routines in NetLogo by embedding them in an agent's behaviour rules, it should be noted that this would prove highly computationally demanding, thus reducing the model speed considerably. More fundamentally, it would be time inefficient owing to the extensive coding required. It is thus much more efficient to use a stand-alone optimiser and integrate the two.

The paper is structured as follows. In the next section we introduce the individual components of our integrated approach, namely ABM and optimisation. Next, we describe previous work and outline the novelty of our approach against it. Then, the case study of PV adoption in the Emilia Romagna region is presented alongside details of the ABM. In the next section we focus on the main contribution of this paper by describing and discussing the proposed integration approach itself. Finally, some initial results of the final integrated model are presented. The paper closes with a brief conclusion section.

Methodological Background

Agent-based modelling

[3, 7, 15] provide overviews of ABM and their use in the social sciences. ABM is a form of simulation modelling in which multiple agents act and interact within an environment. Agents can represent any decision-making unit (e.g., person, household, firm) and are autonomous, can communicate with each other, and are typically heterogeneous. The agents act and interact within an environment which may represent an abstract conceptual space (e.g., social or opinion space), a real physical or geographic space (e.g., a building or country), or have no real meaning (i.e., when agents are in a network connected by links). Typically the agent behaviour rules, attributes and the environment are setup using empirical data or theory, and the model is then run. The emergent macro-level patterns that are the output of the model are then compared and analysed alongside the micro-level rules. Broadly, ABMs can be used first, to explore and explain the mechanism of a theory of individual behaviour on the whole system, or second, to describe and forecast the behaviour of a system, or third, in a participatory context to explore a system and its behaviours with stakeholders. ABM is most often used when the researchers: (i) are interested in modelling interactions and feedback between actors, and actors and their environment, (ii) believe heterogeneity of actors is important in the system, (iii) are interested in the spatial dynamics of a system, (iv) believe path dependence may be an important element in the system, (v) believe actors in the system have behaviours that change, or adapt over time, and/or (vi) want to use an intuitive and flexible modelling approach for participatory modelling [10].

Whilst ABM are well suited, and typically used, to represent the decision-making and behaviour of many heterogeneous micro-level agents, it is less common to represent meso and macro-level actors. At the meso-level, examples might include firms, or government agencies, whereas at the macro-level examples could include governments or nations. However, it is difficult to endow these agents with sophisticated behaviours (beyond profit/utility maximisation or heuristics), especially when interacting with a large number of micro-level agents. Indeed, if the behaviour is more complex, or the number of micro-level agents is large, the process is likely to be computationally demanding. This reflects the problem that the meso or macro-level agent has to perceive, use and manipulate information from all of the agents in the simulation.

The combination of the facts that meso and macro-level behaviour is worthy of representing, and that it can prove difficult to implement in NetLogo means further development is an important avenue for the continued evolution of ABM for policy and social science applications.

Optimisation

Optimisation aims to find the solution to a problem that is optimal with respect to an application-specific criterion. It is applicable in a wide range of contexts and a well-known technique in many sciences. Examples include minimising waste in a production pipeline that can manufacture a range of different items from the same raw material for a set of orders, minimising the cost of travel while visiting a list of locations, or maximising the value while minimising expenditure when bidding on a list of items with associated cost and value.

In a policy-making context, optimisation problems can arise in a number of scenarios. The budget allocated to achieve a certain objective may need to be distributed across different policy instruments in an optimal fashion. The provision of a service may rely on several companies putting in bids to provide this service for a certain fee. A regional development project may want to minimise the negative impact on the environment.

Optimisation is a large field in its own right, and a survey of the different areas is beyond the scope of this paper. The interested reader is referred to [4] for an introduction. There are a number of mature software packages that implement optimisation technologies that can be used here.

It is important to remember that optimisation facilitates decision-making at the macro-level. Once, for example, the bids for a service are known, the decision based on them can be optimised. To get these bids, other techniques are required – in our case, agent-based modelling. Without this separate component, optimisation cannot do its work as it would have no data to base its decisions on. It is the integration of agent-based modelling and optimisation that can support the policy maker.

Previous work

There is a relatively limited literature on the integration of optimisation and ABM, and the authors are aware of no examples akin to the type of integration (i.e., optimisation used in a macro-level/policymaker agent) that is presented here. Broadly, there are two most common forms of integration of optimisation and ABM in previous works: (i) optimisation used as a calibration and validation tool for ABMs, and (ii) ABMs used to solve optimisation problems.

In the first form, which appears the most popular [6], [16] use optimisation to fine tune agent parameters in an ABM of marketing strategies. [12] and [8] again use optimisation in a similar way in their respective models of financial markets. [2] propose using 'adaptive dichotomic optimisation', which represents a form of single objective genetic algorithm optimisation, and demonstrate with an ABM of a financial market. [11] use optimisation in the same way to calibrate micro-level parameters, but also to help optimise the emergent behaviour in their model of disaster management (i.e., to minimise casualties, and other indicators), and in effect to explore the results of their model. [13] uses optimisation to calibrate their model of air traffic management and suggests the approach be used elsewhere. [9] use optimisation in a related form, in which an ABM and optimisation model of anti-pirating techniques for shipping companies are developed and used to validate and complement each other within a larger tool.

In the second form, [1] present a review of the uses of ABMs, and broadly agent approaches, to help solve optimisation problems. They identify two types of agents in this sense, first physical agents that may represent physical entities such as workers or machines, and second, 'functional' agents, that represent nothing in the physical world, but are a piece of software used to carry out subtasks of the ABM. They also identify two structures: one in which many agents self-organise to solve a problem, or a second in which there are 'mediator' type agents which set an optimised plan, that may be refined by the 'worker' agents. This type of integration is less formal than the first and is essentially the application of ABM to scheduling or resource allocation problems. Examples include [17], [5], and [14].

Finally, [5] present the embedding of optimisation in micro-level agents to aid resource allocation problems, using an example taken from the food industry. This is one of the more similar integrations to that which is described in this paper, however there are some crucial differences. First and most importantly, the actors represented are at the micro-level, rather than at the macro-level (i.e., policymakers). Second, the intention of the model is to solve some production / resource problem, rather than to represent decision-making accurately. Third, the implementation is done using one piece of software, rather than integrating an existing ABM with optimisation software.

What separates these previous works and this paper is that the approaches use either, optimisation to analyse the results of the simulation after it has been run, embed it in the micro-agent decision making architecture, or use agent type approaches on common optimisation problems. The approach described in this paper allows a run-time integration of optimisation and ABM, where the optimisation component (representing a macro-level policy agent) can communicate with, and influence, and thereby optimise macro-level decisions in the simulation during its execution.

Case Study

Photovoltaic in the Emilia Romagna region

The example used in this paper is that of the 'ePolicy social simulator' developed as part of the ePolicy FP7 project on engineering the policy life-cycle[1] which uses the Emilia Romagna region in Italy as its case study. Italy has few fossil fuel resources and relies heavily on imported natural gas, which is why (together with a general sentiment of the Italian population against nuclear power) renewable energy has long been a topic of interest in Italy. Of the different renewable energy technologies available, PV panels have been of particular interest in Italy due to climate and economic conditions that have resulted in a steep rise in capacity in Italy in the last couple of years. The regional government in Emilia Romagna is particularly interested in the potential for new technologies to contribute to energy production. This interest in PV serves as the basis of the use of Emilia Romagna as the case study.

The ePolicy social simulator

The ePolicy social simulator, which is an ABM, has been developed to serve as a component of a wider decision support system (DSS) for policy makers and analysts working on energy policy. It is intended to be used to answer the policy question, "What are the effects of different policy instruments on PV system diffusion in the Emilia Romagna region?". For Emilia Romagna, two specific regional policy instruments have been identified in collaboration with regional policy-makers: investment grants and interest-rate support (for interest on loans to purchase a system). The ABM simulates the behaviour of households in reaction to these policy support schemes. The agents' behaviour rule simulates the consideration of, and decision to install, PV and is based upon a household survey and interview data from Italy. When agents decide to install PV, they may make a bid to the regional government to apply for support either in the form of grants or interest-rate support. This form of application is motivated by similar schemes that the Emilia Romagna region has run in the past. The optimiser then has to decide which bids to fund, given the overall budget and power capacity target and any other goals. The synthetic population of agents in the ABM is setup using population data from Italy. The environment of the ABM in which the agents interact in is setup using GIS data of the Emilia Romagna region. The outputs of the ABM are the aggregate costs of installations, aggregate power generated by PV, and total number of installations.

[1] See `http://www.epolicy-project.eu/` for more details.

Integration approach

The integrated approach is implemented in a Java-based user interface which unifies and encapsulates both the ABM and the optimiser[2]. While NetLogo is used for the ABM, the lpsolve software[3] provides the optimiser. Both provide application programming interfaces (APIs) that we utilise in the user interface.

The purpose of the user interface is to transparently make use of the relevant technologies without burdening the user with additional parameters and setup steps. Indeed, there is no indication of how the specified problem is being solved underneath. This also allows for easy and transparent integration of alternative solving methods.

Figure 1 describes the integration of the ABM and optimisation. Using the user interface, the user first defines the scenario they wish to explore. The scenario options include selecting the region (Emilia Romagna or just its capital, Bologna), budget and its distribution over time (first come, first served, even, ramp up or down), intended PV power supply, optimisation criterion (minimising the budget spent, maximising the power production, maximising the participation, i.e. the number of funded bids), and their beliefs about the status of national level policy instruments (feed-in tariffs and income tax liability reductions). The simulator and its data are then loaded, and the optimiser is informed of the user's choices on optimisation criterion, budget and budget distribution.

Now the simulation begins. For each time period (one year), household decisions are made, and a list of households who wish to bid for regional government support is generated along with the bid amount and the size of the PV installation. This list of bids is then passed to the optimiser which builds an optimisation model based on the bids and the parameters specified by the user.

The optimisation model is a variant of the so-called knapsack problem – given a set of items, each with a certain weight and value, and a knapsack with a certain capacity, maximise the value of the items put in the knapsack without exceeding its capacity. In the case of maximizing the power produced for example, the weight of the items is the cost, the value the power produced, and the capacity of the knapsack is the budget.

Knapsack-type problems are very common in the optimisation domain and can be solved very efficiently in practice even though they are hard to solve in general. In our experiments, the optimisation problems were solvable within a fraction of a second for all scenarios and optimisation criteria.

The optimiser solves the optimisation problem and generates a list of funded bids. This funded list is then passed back to the ABM where the households implement the decisions and install PV, resulting in costs to the policy instrument, and increased power generation. This process is repeated until the simulation has reached the desired final year (currently 2020). Finally, the produced power, spent budget, count of funded bids (i.e., installations) is provided to the user.

Initial simulator results

Table I presents a comparison of the key outputs that the different optimisation objectives and methods achieve on the data of an auction the Emilia-Romagna region ran in 2004, with a budget of 3,200,000 Euro and a target power of 1,200 kW. All scenarios use the first come, first served budget distribution and have both grant and interest payment incentives enabled. The values are the ones achieved after the first simulation step.

The ordering approach corresponds to what the Emilia-Romagna region did in the 2004 auction to determine the winning bids. All bids are ranked according to criteria that depend on the optimisation function. Then the list of bids is traversed in this order, with each bid funded if there are sufficient funds available and the power target has not been reached. For example, when maximising the power production, the bids are ranked by the ratio of power production of the installation divided by its cost.

We are showing only the results after the first time step of the simulation because the results across all time steps are not comparable. The optimisation takes into account all the information available at each step and makes the optimal decision for that. This is what we aim to demonstrate here. Ideally, we would be able to make decisions based on the information for the entire simulation – while funding a particular bid at time step n may appear to be a good decision, it is possible that at time step $n+1$ a better bid will be made that should be funded instead. However, it is impossible for us to do this – decisions have to be made for each time step because earlier decisions will affect the course of the simulation and later time steps.

The approaches that do not use optimisation tend to leave more "slack" at each time step, i.e. not all the bids that could be funded are. This means that more of the budget is available in subsequent time steps, allowing to fund potentially better bids. It is because of this that the approaches that do not use optimisation may achieve overall better results than the optimisation approach. This is, however, purely by chance – optimisation is overall likely to obtain better results, especially when the number of time steps is large and decisions are spread out across them. Here, we compare the numbers after the first simulation step to remove this random element.

Table I clearly shows that using optimisation technology is worthwhile. The simple first come, first serve approach for deciding which bids are funded achieves significantly less power production and fewer funded bids than the other approaches. Ranking the bids depending on the optimisation criterion achieves much better

[2]Supplementary material in form of a more detailed description of the implementation, experimental results, and where the software can be downloaded be found at `http://www.epolicy-project.eu/sites/default/files/public/D5.3.pdf`.

[3]See `http://lpsolve.sourceforge.net/` for details.

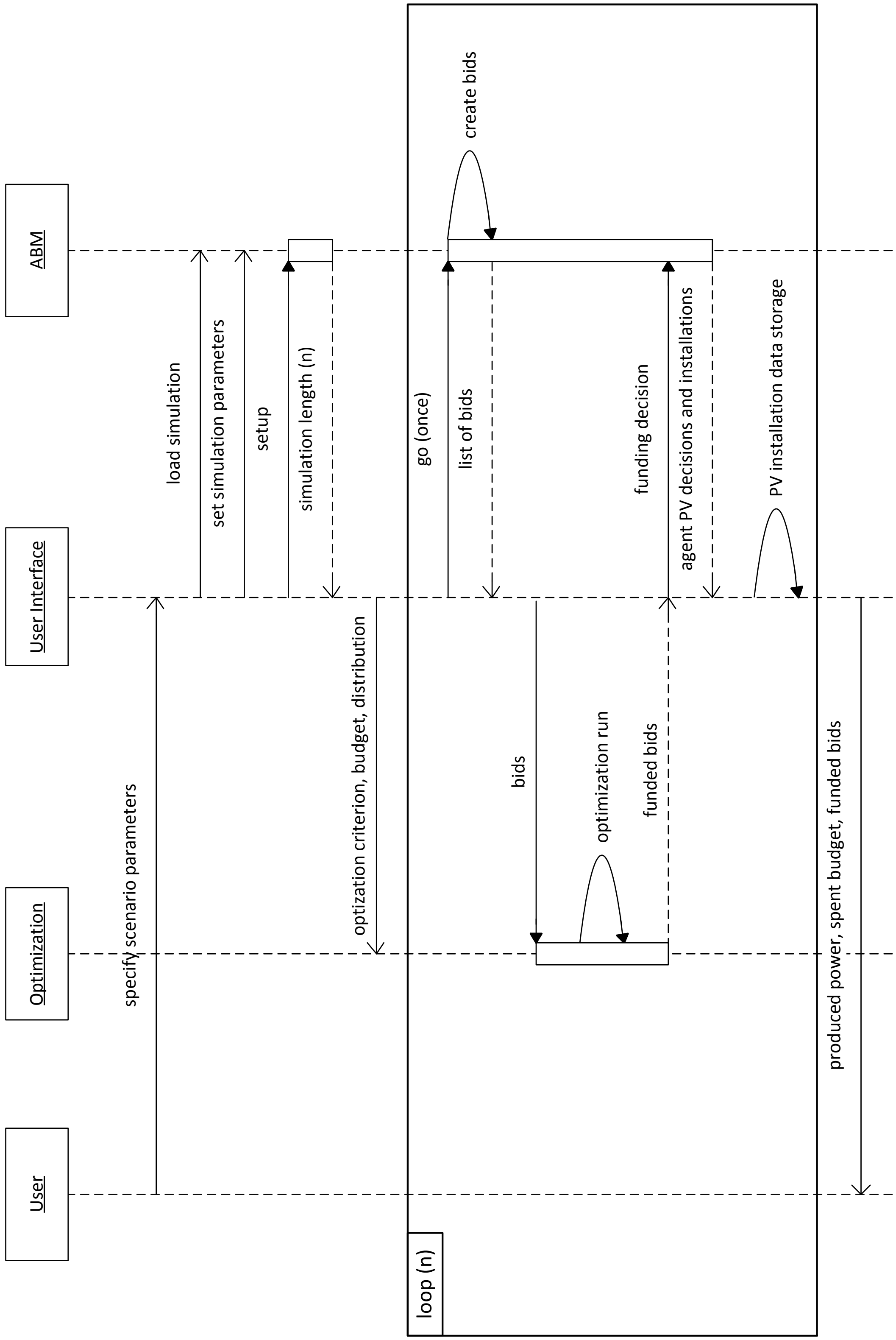

Fig. 1: UML sequence diagram of the integration between user, user interface, optimisation component, and ABM.

	Total power (kW)	Total allocated funds (Euro)	Total funded bids
first come, first serve	4,350	3,191,303.73	1,750
order to maximise power	7,050	3,129,700.30	2,175
order to minimise expenditure	1,350	526,087.50	150
order to maximise participation	6,300	3,175,091.00	2,425
optimise to maximize power	7,125	3,199,996.67	825
optimise to minimize expenditure	1,200	466,252.5	125
optimise to maximize participation	6,300	3,198,519.88	2,425

TABLE I: Comparison of key outputs for different optimisation objectives and methods for the first simulation step. Both incentive instruments (i.e. investment grants and interest-rate support) were enabled and first come, first served budget distribution is used.

results. Yet, using optimisation is able to improve even further on this. These results clearly demonstrate the benefit of using optimisation technology for ABM.

Note that in the case of optimising to maximize participation, the order and the optimisation approach achieve the same number of funded bids, but the order approach at a lower cost. This is because the optimisation approach considers only the single objective of maximizing the participation – there are several assignments of funds to bids that achieve the same participation and the optimisation approach happened to choose the one with a higher cost.

This is not an inherent limitation of the optimisation approach – to take the cost into account as well, we can model the problem as a so-called multi-objective optimisation problem. This class of problems is similarly well studied in the literature and can be solved efficiently in practice. We do not consider this approach here simply for the sake of simplicity of the exposition – multi-objective optimisation problems are by their very nature more complex to define.

Conclusion

We have made a first step towards integrating agent-based modelling and optimisation technologies at runtime. In many scenarios, a special type of agent is required to make downward-causation decisions that affect the other agents in a simulation. Such decisions need to take into account the current state of the simulation and optimise for a criterion. Optimisation technology is a natural choice for implementing this process. Instead of integrating the optimisation as part of the decision-making of a 'governmental agent' within the ABM, our prototype linked the ABM and an existing solver for optimisation questions and allowed them to communicate during the course of the simulation.

In our case study for funding photovoltaic installations in Italy's Emilia-Romagna region, our integrated simulation has shown significant improvements on previous results by using the optimisation component instead of less sophisticated approaches.

Although we present the integrated ABM-optimisation approach with the help of the Emilia Romagna PV adoption case study in this paper, our approach is not limited to this case study and can be applied to a large variety of topics. It is in particular useful when macro-level decision-making and its influence on the agent decisions are of importance for the simulation. With our approach one can both use the ABM to understand and analyse the agent decision-making behaviour at the micro-level and at the same time to make optimisation decisions at the macro-level. As noted before, because of the interaction of the two components during the execution of the simulation, the macro-level decisions influence the agents at the micro-level and therefore influence the whole simulation result. This interaction between the different levels allows for the study of the complete system, as well as the interaction between levels.

Our future work can be divided into two streams. The first stream is the extension of the ABM and the exploration of other scenarios for the Emilia Romagna case study. In particular, applying the same methodology to an entire country, such as Italy, would be of practical interest.

The second stream concerns the generalization of the integration methodology and its application to other domains, as well as refining the current interactions between the components. In particular, multi-objective optimisation problems and the effect of optimising multiple, independent decisions at each time step on the entire simulation can be explored.

Acknowledgements

This research was funded under the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement number 288147 (ePolicy).

References

[1] M. Barbati, G. Bruno, and a. Genovese. Applications of agent-based models for optimization problems: A literature review. *Expert Systems with Applications*, 39(5):6020–6028, Apr. 2012.

[2] B. Calvez and G. Hutzler. Adaptative Dichotomic Optimization: a New Method for the Calibration of Agent-Based Models. In *European Simulation and Modelling Conference (ESM 2007)*, pages 415–419, 2007.

[3] E. Chattoe-Brown. Why Sociology Should Use Agent Based Modelling. *Sociological Research Online*, 18(3), Aug. 2013.

[4] E. K. P. Chong and S. H. Zak. *An Introduction to Optimization.* Wiley-Interscience, 3 edition, Feb. 2008.

[5] P. Davidsson, J. A. Persson, and J. Holmgren. On the Integration of Agent-Based and Mathematical Optimization Techniques. In *Agents and Mulit-agent systems: Technologies and Applications.* 2007.

[6] C. Franklin. *Multi-objective Optimisation using Agent-based Modelling.* PhD thesis, Stellenbosch University, 2012.

[7] N. G. Gilbert. *Agent-based Models.* Sage Publications Ltd., 2008.

[8] M. Gilli and P. Winker. A global optimization heuristic for estimating agent based models. *Computational Statistics & Data Analysis*, 42(3):299–312, Mar. 2003.

[9] M. Jakob and P. Michal. Agents vs . Pirates : Multi-Agent Simulation and Optimization to Fight Maritime Piracy. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems Inno- vative Applications Track (AAMAS 2012)*, 2012.

[10] P. G. Johnson. *The SWAP Model: Policy and Theory Applications for Agent-based Modelling of Soil and Water Conservation Adoption*. PhD thesis, University of Surrey, 2014. forthcoming.

[11] G. Narzisi, V. Mysore, and B. Mishra. Multi-Objective Evolutionary Optimization of Agent-based Models: An Application to Emergency Response Planning. In *Proceedings of Computational Intelligence*, 2006.

[12] A. Rogers, M. Group, and C. Science. Multi-objective Calibration for Agent Based Models. In *Proceedings of 5th Workshop on Agent-Based Simulation*, 2004.

[13] S. Schutte. Optimization and Falsification in Empirical Agent-Based Models, Jan. 2010.

[14] K. Socha and Kisiel-Dorohinicki. Agent-based evolutionary multiobjective optimisation. In *Proceedings of the 2002 Congress on Evolutionary Computation.*, volume 1, pages 109–114. Ieee, 2002.

[15] F. Squazzoni, W. Jager, and B. Edmonds. Social Simulation in the Social Sciences: A Brief Overview. *Social Science Computer Review*, Dec. 2013.

[16] T. Terano and K. Naitoh. Agent-based modeling for competing firms: from balanced-scorecards to multiobjective strategies. In *Proceedings the 37th Annual Hawaii International Conference on System Sciences*. IEEE, 2004.

[17] G. Weichhart, M. Affenzeller, A. Reitbauer, and S. Wagner. Modelling of an Agent-Based Schedule OPtimization System. In *Proceedings of the Fourth Congress on Evolutionary Computation*, number Penya 2002, 2002.