Case Studies for a Markov Chain Approach to Analyze Agent-Based Models

Florian Kitzler
Technische Universität Wien, florian.kitzler@tuwien.ac.at

Martin Bicher
Technische Universität Wien, martin.bicher@tuwien.ac.at

Case Studies for a Markov Chain Approach to Analyze Agent-Based Models

Florian Kitzler[1], Martin Bicher[2]

[1, 2] Vienna University of Technology, Institute for Analysis and Scientific Computing, Karlsplatz 13, 1040 Vienna, Austria
{florian.kitzler[1], martin.bicher[2]}@tuwien.ac.at

Abstract. Agent-Based Models have become a widely used tool in social sciences, health care management and other disciplines to describe complex systems from a bottom-up perspective. Some reasons for that are the easy understanding of Agent-Based Models, the high flexibility and the possibility to describe heterogeneous structures. Nevertheless problems occur when it comes to analyzing Agent-Based Models. This paper shows how to describe Agent-Based Models in a macroscopic way as Markov Chains, using the random map representation. The focus is on the implementation of this method for chosen examples of a Random Walk and Opinion Dynamic Models. It is also shown how to use Markov Chain tools to analyze these models. Our case studies imply that this method can be a powerful tool when it comes to analyzing Agent-Based Models although some further research in practice is still necessary.

Keywords: Agent-Based Model, Markov Chain, Random Map Representation

1. Introduction

Agent-based modeling has become a widely used modeling technique and is nowadays used in many fields such as social sciences, computer science, healthcare management and economics [1]. One of the many advantages over other modeling techniques is the high flexibility of agent-based models. Another big merit is the possibility to describe heterogeneous structures. Especially these two features make it possible for agent-based models to deal with huge data sources and model very complex systems. The availability of more powerful computational tools helps to simulate such complex models but there are still limitations when it comes to analyzing agent-based models. Very complex agent-based models have a high number of parameters and usually a lot of them are not known exactly. To parameterize or calibrate the model, a lot of simulation runs are necessary. This leads to a high amount of computational resources and time. Another big issue is the validation of the model. That means to find out if the right model was used to satisfy your needs. Hereby a problem arises as appropriate methods hardly exist for agent-based models. The aforementioned problems underline the need for analysis-methods for agent- based models. Before we briefly explain our approach it is necessary to give short introductions to agent-based models and their connection to Markov chains. The latter poses the key-tool for our analysis-method.

2. Introduction to Agent-Based Modeling and Markov Chains

In this section we will give a short introduction to agent-based models (short ABMs) and Markov chains (short MCs). ABMs are models of systems composed of autonomous, interacting entities called agents. These agents live together in the same environment and can change their attributes after interacting with other agents or the environment. The collection of all attributes that describe the agent is called individual state of the agent. In this work we are just regarding stochastic ABMs. This means that random effects influence the simulation results. Stochastic models that are easier to analyze are MCs. A MC is a stochastic process that satisfies the Markov property that is also called "memorylessness" [2]. This means that the process "forgets" all about the past and the

further evolution only depends on the current state. We first define the state space of our stochastic process. That is a finite set of states our process can be in. A time-discrete stochastic process is defined as a sequence of random variables with state space. The variable always stands for the time at which we observe our process. We define as the one-step transition probability from state at time step to state in the next time step. The transition probabilities can be collected in a transition matrix, where the entry in row number i and column number j correspond to. The transition matrix contains all information about the MC and is important to analyze its transient and asymptotic behavior. After starting in a certain state the MC evolves following the transition probabilities. We now want to know in which state the process is when we observe it at a certain time. Given an initial distribution vector we can calculate the distribution at time with the formula The initial distribution is given as a row vector with being the probability that the MC starts in state at time. So the calculation of the state distribution at time is just n times multiplying with the transition matrix. Under certain conditions on the transition matrix a unique limit distribution exists with being the asymptotic probability for the process to be in state. Under these conditions the limit distribution can be found solving a system of linear equations, see [2]. As intended by the mentioned formulas MCs are quite simply to analyze in the contrary to ABMs. Hence a direct comparison between these techniques would improve the analysability of the ABMs.

3. Agent-Based Models as Markov Chains

This section will show that this comparison is possible as we give a step by step instruction on how to create a MC from a given ABM [3], [4].

3.1 Step1: Identifying the State Space

The first step is to find all possible states of the MC. Therefore we start with a configuration of the ABM. That is a possible combination of individual state values for each agent.

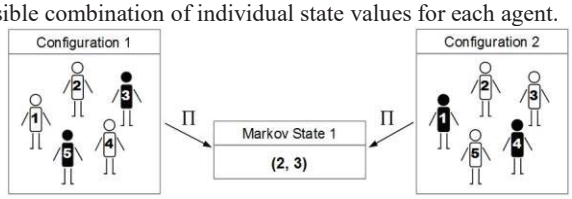

Fig. 1. Two configurations that map on the same Markov state. The first number represents the number of black agents, the second number represents the number of white agents. The state space of the MC is given as with number of black agents number of all white agents.

Let be the individual states space of an agent and the total number of agents. We define the configuration space as the set of all configurations of the ABM. Our aim is the macroscopic analysis of the model. Hence we need to find an aggregation mapping from our configuration space to a space. Usually this is done defining that the i-th row of counts all agents with state. The image will furthermore pose for the state space of our MC. This idea is illustrated in Fig. 1 for an ABM with only two individual states represented by the colours white and black.

3.2 Step 2: Calculate the Transition Matrix

The second step is to find all possible transitions between our Markov states and calculate the transition probabilities. Fig. 2 shows the practice of how to find possible transitions.

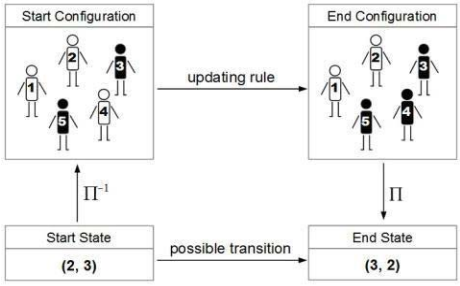

Fig. 2. Schematic description of the approach of calculating the transition matrix.

We first start with a chosen Markov state and we want to know which state can be reached in the next time step. Then we select a configuration of the ABM that maps on our chosen Markov state. It has to be mentioned that it is generally not trivial what configuration has to be chosen if more than one of them maps on the same Markov state but in the investigated examples this is irrelevant. With the regarded configuration we simulate one time step of the ABM using the updating rule. The updating rule contains all information about the ABM for example the movement of the agents or interacting rules. All agents update their attributes according to that rule and we observe a new configuration after the time step.
The last step is to map our new configuration again onto the state space with and we have found one possible state transition. If we repeat this several times for each Markov state we can approximately calculate the transition matrix for our MC.
Following these two steps a MC can be developed matching the macroscopic dynamics of the ABM. Hence the ABM can be investigated analysing the MC.

4. Results

We finally want to compare some results of the ABM and the corresponding MC for some test cases. Therefore we look at the transient behavior of the model at a specific time. We fix an initial distribution and calculate the transient distribution for the MC. For the ABM we realize a Monte-Carlo-Simulation where the start values for the agents follow the same initial distribution.

4.1 1D Random Walk

We will take a look at the results of a 1D random walk model. In this case we have a 1-dimensional lattice with a finite number of sites. Furthermore a number of agents are distributed on them. Hence the only attribute of an agent is its site on the lattice. The agent can move left and right following some rules. If the agent is alone on a site he has to move on to one of the neighboring sites with the same probability. If another agent is on the same site, there is a probability of 0.6 to stay. First we need to define the states of the MC and how the map works. We are considering 5 sites and 2 agents. The individual state of an agent is just an integer between 1 and 5 that holds his position on the numbered lattice. A state of the MC always contains 5 positive integers that sum up to the total number of agents, in our case to 2. If we use the symmetry of this model we can reduce the total number of Markov states to only 9. In Fig. 3 we can see the so-called transition-diagram of our MC. The arrows show all possible one-step transitions with probability greater than 0 between the Markov states represented by circles. Without knowing the exact transition probabilities we can start analyzing our process.

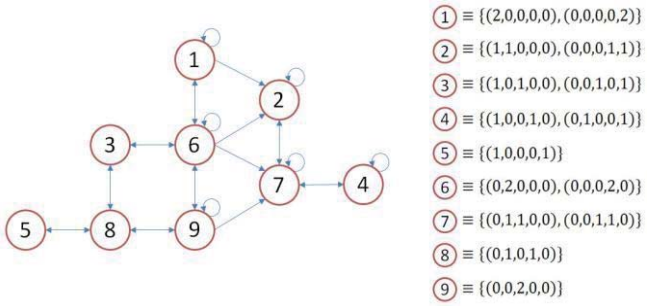

Fig. 3. Transition-diagram of a 1D random walk with 2 agents that can move on 5 sites lattice. The circles represent the Markov states and the arrows represent all possible one-step transitions. On the right hand side we see the corresponding agent configurations.

We call a closed communicating class. Once the process enters one of those states, it stays in the class forever. As is reachable from every other state in a finite number of time steps we neglect all other states for the asymptotic behavior.

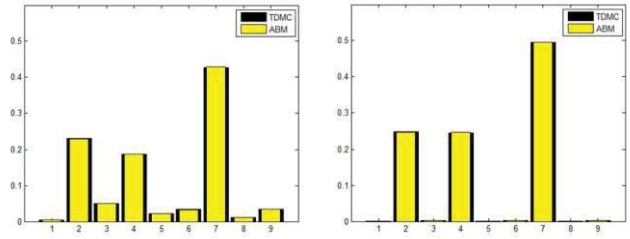

Fig. 4. Comparison of the ABM (bright) with the time-discrete MC (dark). The process always starts in state 5. On the left we see the transient distribution after 10 time steps, on the right after 20 time steps. For the Monte-Carlo-Simulation of the ABM we used 100.000 iterations.

The results in Fig. 4 show that the limit distribution is nearly reached after just 20 time steps. In this case we could calculate the transition probabilities by hand using the movement rules of the random walk. In our simulation the stochastic process always started in state 5 corresponding to the configuration. The exact limit distribution can be calculated solving a system of linear equations. It is independent of the initial distribution with, and. We can also calculate the expected value of the time we first enter the closed communicating class. When we start in state 5 this absorbing time is reached after 7.08 time steps.

4.2 Opinion Dynamics Model

Another type of model we were investigating is a two-state opinion dynamics models in which the agents are able to change their states after interacting with other agents. We were looking at different interaction network types and compared the results with regard to the Euclidean norm of the error vector. Analyzing a simulation run of the ABM with 20 agents we received a total difference to the Markov chain of.

Conclusion and Outlook

Our case studies show that this approach works well for the investigated examples. Using statistic tests we can show that the results of the ABM and the MC follow the same distribution. The movement and interacting models we considered can be used in more applied models as sub models. Some further research on bigger and more applied models is still needed. Acknowledgments. K-Projekt DEXHELPP is supported by BMVIT, BMWFW and the state of Vienna via COMET – Competence Centers for Excellent Technologies. Programme COMET is processed by FFG.

References

1. Axelrod, R.: Agent-Based Modeling as a Bridge between Disciplines. In: Tesfatsion, L., Judd, K.L. (eds.): Handbook of Computational Economics, Agent-Based Computational Economics, Volume 2, 1565-1584
2. Serfozo, R.: Probability and its Applications: Basics of Applied Stochastic Processes.
3. Banish, S., Lima, R., Araújo, T.: Agent Based Models and Opinion Dynamics as Markov Chains
4. Izquierdo, L. R., Izquierdo, S. S., Galán, J. M., Santos, J. I.: Techniques to Understand Computer Simulations: Markov Chain Analysis. In: Journal of Artificial Societies and Social Simulation 12(1)6 http://jasss.soc.surrey.ac.uk/12/1/6.html, (2009)