2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)
October 8-12, 2022, Macau, China

2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC) | 978-1-6654-6880-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ITSC55140.2022.9921892

Empirical Learning of Decision Parameters for Agent-Based Model*

Bing Song, Gang Xiong, Fenghua Zhu, Xuke Wu, Yisheng Lv, Peijun Ye†

Abstract— **Agent-Based Model (ABM) is a widely used tool to analyze distributed systems. However, the decision-making parameters are difficult to determine, since ABM is a kind of micro model and such parameters, varying from person to person, cannot be measured conveniently in real traffic systems. For this problem, this paper introduces reinforcement learning to empirically and efficiently calculate the micro parameters of ABM. By a parameterization of the individual interactions, our new approach is able to decouple the dependence for a given agent upon his "social neighbors", and thus can accelerate the learning process. Experiments on inter-city traveling of population indicate that the proposed method is effective for the micro parameter computation.**

I. Introduction

Agent-Based Model (ABM) is an explicit model which can describe the behavior patterns of individuals through a micro perspective. Moreover, it can simulate the relationship between individuals and environment. This makes ABM a widely used tool to study socio-ecological systems such as analyzing traffic situation[1-4], traveling behavior of population[5-8], social computation[9-11], population synthesis[12-15], and analyzing behavior of groups[16,17]. In particular, many researchers use ABM to analyze COVID transmission [18-20].

One main challenge of ABM is the model parameter computation. As ABM usually models individuals or minor groups in real social systems, such microscopic features like preferences, cognitive knowledge patterns, etc. are rarely measurable. Even a few of them could be achieved via classic psychological tests, the results from a minor group of testees may usually bring sample bias and may not be representative enough for the overall population. Therefore, it is essential to develop a feasible method to efficiently calculate the agent micro decision parameters so that the subsequent travel simulation is consistent with realistic systems.

In history, agent parameter computation is usually named as model calibration and validation. It refers to calibrating agents parameters which are difficult to validate in macro state by simulation output and realistic output. Due to the heterogeneity of agents and the complex interactions between agents and environment, values of the same microscopic parameter are mostly various for different agents. These parameters are difficult to calibrate at the macro level, because the dynamic properties and causal relationship at the micro level are difficult to analyze directly.

There are several different approaches in current work on parameter calibration. Nicholas et al. proposed a sensitivity-based method [21]. They propose a global sensitivity analysis to combine the two sensitivities to learn how microscopic parameters influence the macroscopic output of the system. Fagiolo et al. proposed an indirect reasoning method [22,23]. They used a simple proxy to approximate the relationship between ABM input and output, so that the parameter space could be searched more quickly. This works well when the scale is small. Simone et al. proposed the generalized method of moments (GMM) [24], but the selection of moments may cause deviations in the calibration of parameters. The same problem is also exists in the method of simulated moments (MSM) [25]. Classical Bayesian theory is introduced into the parameter calibration problem [26,27], which effectively solves the problem of bias due to the choice of moments, but the work of Canova et al. [28,29] proved that the choosing of prior distribution may produce artificial curvature. Lamperti et al. proposed a machine learning approach [30], which requires to select an appropriate prior distribution to ensure validity, and still has a long computational process. Ye et al. introduced the idea of mean field in physics, and calibrated the microscopic parameters from the state transfer from the macroscopic perspective [31]. On this basis, this paper further introduces Reinforcement Learning (RL) to analyze the state transfer at macro level, and quantifies the interaction of agents at micro level to calibrate the parameters. Experiments on population migration scenarios demonstrate the effectiveness of our method.

This paper mainly has three contributions:

1) Introduce Reinforcement Learning to calculate the state transfer probability, and calibrate microscopic parameter with state transfer probability.

2) Propose to quantifies the interaction between agents as the agent's own parameters to represent explicitly and reduce the computational burden.

*The research was supported by Chinese Guangdong Basic and Applied Basic Research Foundation under Grant (2019B1515120030, 2020B0909050001). National Natural Science Foundation of China under Grants U1909204, U1811463, 62076237, T2192933, 61872365& U19B2029, Youth Innovation Promotion Association Chinese Academy of Sciences under Grant 2021130.

B. Song, F. Zhu, X. Wu, Y. Lv, P. Ye are with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China. They are also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, 100049, China. (e-mail: songbing2020@ia.ac.cn; fenghua.zhu@ia.ac.cn; wuxuke2020@ia.ac.cn; yisheng.lv@ia.ac.cn; peijun.ye@ia.ac.cn)

G. Xiong is with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China, The Beijing Engineering Research Center of Intelligent Systems and Technology, Institute of Automation, Chinese Academy of Sciences, China, and The Cloud Computing Center, Chinese Academy of Sciences, Dongguan 523808, China.

† Corresponding Author.

3) From the experiment of inter-city traveling of population, prove that our method behaves well in accuracy and computational complexity.

II. Related work and problem statement

A. A.Review of Classical Methods

The purpose of calibration is to adjust the microscopic parameters of the agent so that the output of system is controlled within an acceptable error range to simulate stylized fact[32,33]. However, because ABM has complex construction, it is hard to observe the microscopic parameters of the actual system, and direct parameter calibration is not feasible. Naturally, minimizing the distance between the output of the simulated system and actual statistics enables ABM to approximate reality to achieve the expected effect of the calibration. This inspired the idea of moment-based methods. The Generalized Method of Moments(GMM) has been applied in some financial market problems[24]. However, the moment cannot be accurately known in GMM, so the Monte Carlo simulation method is used to approximate it. This means that the effect of approximation affects the selection of moments and thus the accuracy of the final parameter calibration. The Method of Simulated Moments(MSM) is to select a vector of parameter values and then run ABM to generate a simulated time series, and compute the distance function which measures simulated moments and real-world sampled moment data[25]. Then, through minimizing distance, the micro parameters will be calibrated, which can be represented as:

$$argmin\, Dist(X^R, X^S, \theta)$$

where X^R is the sampling moment of realistic data, and $X^S$is the simulated moments. $Dist$ is the distance function of these two moments. The calibrated parameter θ is searched in parameter space to minimize distance function. However, the accuracy of calibration depends on the choosing of moments, and searching process increases the computational burden of method.

Grazzini et al. introduced Bayesian theory to calibration[27]:

$$p(\theta|Y^R) \propto L(\theta; Y^R) \cdot p(\theta)$$

$$L(\theta; Y^R) = p(Y^R|\theta)$$

where $L(\theta; Y^R)$ represents likelihood function, $p(\theta)$ is the prior distribution and $p(\theta|Y^R)$ is the posterior distribution of parameters. $Y^R$is observed statistics. According to maximize $p(\theta|Y^R)$, parameter θ of agents will be calibrated.

Bayesian method solved the drawback of moment-based methods that calibration accuracy depends on the choosing of moments. And Bayesian method focuses on the whole distribution but not specific moments, which improve the efficiency of Bayesian method.

This frame involves three steps: *1)* simulating the model, *2)* computing $L(\theta; Y^R)$ with a θ. *3)* sampling parameters from $p(\theta|Y^R)$. These steps have great computational burden. The complexity of ABM leads to the requirement for efficient sampling methods.

With the development of machine learning(ML), researchers started to introduce ML to ABM. Lamperti et al. Proposed to calibrate with ML and intelligent iterative sampling[30]. This method draws a points pool of parameters first, then sample and runs ABM with the result of sampling. Points will be labeled according to the output of ABM. Run the surrogate learning algorithm and predict labels over the pool. Then sample from unlabeled points and label them after running ABM. Iteration will stop till the 'budget' defined by users is reached. Before this procedure, preliminary settings should be chosen, including surrogate algorithm, sampler and measurement of surrogate performance.

This approach calibrate parameters by drawing pool of parameter combinations. Thus, it solves the problem of searching in parameter space. In particular, decision trees are introduced to classify and regress, and the set of decision trees is build to make the surrogate approximate.

The decision tree surrogate needs heuristic search so the computational burden is still great if ABM scale increases.

Calibrating from a fundamental perspective in detail provides a different perspective. Yu et al. proposes AMETS (Agent-based Model for Emission Trading Scheme)[34] model to describe the emerging process from micro to macro level, and heterogeneity among agents. Parameters in this model are divided into two parts. Calibration of first parts uses FORECAST model[35] and for the second part they design functions and calibrate with collected statistics. This helps the results of AMETS achieve high accuracy.

B. Problem Statement

In Ye's work[27], the behavior of agents is considered as a high-order Markov Decision Process(MDP)[36].Then from Markov theory, the transfer of ABM's macro state will have Markov property, which is represented into:

$$\begin{cases} x(t) = T(t) \cdot x(t-1) \\ x(0) = x_0 \end{cases} \tag{1}$$

This is the macro state transfer equation, where $x(t) = [x_1(t), x_2(t) \cdots x_N(t)]^T$ is macro state of ABM, $x_i(t)$ is the number of agents with the i-th micro state at time t. N is the states number. x_0 is initial state of the ABM system. $T(t) \in \mathrm{R}^{N\times N}$ is the state transfer matrix. $T_{ij}(t)$ stands for the state transfer probability from i-th micro state to j-th micro state at time t.

Let $y(t)$represent the macro observation of ABM at time t. Assuming that the measurement is linear, then the M-dimensional vector $y(t)$ will be:

$$y(t) = B(t) \cdot x(t) \tag{2}$$

where $B(t) \in R^{M\times N}$ is the measurement matrix. Let $\hat{y}(t)$ stand for the realistic observation of actual ABM system, then we can compare these two observation and get a function to measure the macro output of ABM system:

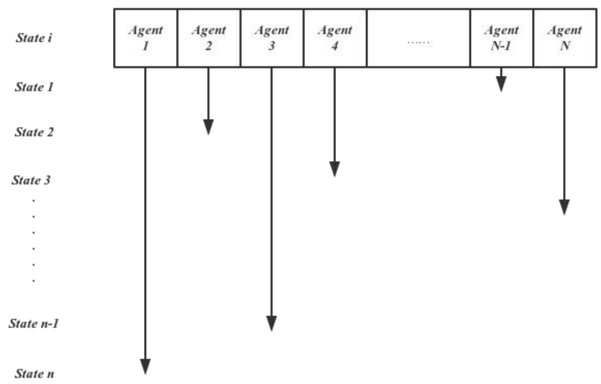

Fig. 1.State transfer after Reinforcement Learning.

$$J = \sum_{t=1}^{K} J(t) = \sum_{t=1}^{K} [y(t) - \hat{y}(t)]^T \cdot V \cdot [y(t) - \hat{y}(t)] \quad (3)$$

where K stands for the number of steps, V is coefficient matrix of the importance of each metric.

Then $T(t)$ can be solved through minimizing J. The macroscopic state transfer probability of the system is the statistical representation of the microscopic actions of multi-agents, and the actions of the agents are determined by their own parameters and environmental information. This inspires the idea of this paper that uses the macroscopic state transition probability to calibrate the ABM parameters.

III. Decision Parameter computation using reinforcement learning

On the basis of macro state transfer, this paper further proposes to use Reinforcement Learning to calibration. Inspired by macroscopic state transfer, this paper further proposes to use reinforcement learning to calibrate the microscopic parameters of ABM. Specifically, we first use reinforcement learning to learn the behavior patterns of the agent. Considering the multi-faceted micro-parameters of the agent to set the reward function of action, so that the action of the agent can simulate the decision result of the agent in the real scene to the greatest extent. For the learned agent, its state transfer probability is calculated to make it equal to the transfer probability expressed by the microscopic parameters, so as to calibrate the microscopic parameters of the agent.

The state transfer is as shown in Fig.1. Assume N agents at time $t-1$ are with the i-th micro state, then at time t, those agents will transfer to different states. The transfer step is decided through Reinforcement Learning. After RL, agents will transfer to the state that will get the most reward. Due to the heterogeneity of agents, the reward and their transfer targets will be different. Let n_{ij} stand for the number of agents that transfer from i-th state to j-th state, so the corresponding transfer probability will be represented by the following equation:

$$T_{ij} = \frac{n_{ij}}{N} \quad (4)$$

From a microscopic point of view, there are roughly four factors that can affect the state change of an agent: parameters related to the state change of agent itself, interaction with other agents, decision parameters and methods of agents, and environmental influence. For ABM, there are complex agent interactions inside the system. A powerful tool for simulating the decision-making process and selection preferences of agents under this interactive relationship is Multi-Agent Reinforcement Learning (MARL). However, this method is computationally demanding because it requires to observe the joint state of the agents and consider joint actions and rewards. With the number of agents increasing, the computational cost will become barely affordable. In order to simplify the calculation, this paper parameterizes the interaction between multiple agents. Specifically, the agents can be expressed as $a\{t,\theta_i(t), Nei(t), W_i, Env\}$. Here, $\theta_i(t)$ is agent parameters itself, that is its individual factors which will be considered in the decision making process. $Nei(t)$ is factors from other agents which have interaction with itself. W_i is weight parameters which measures importance of factors. Env is factors of environment. Thus we can express T_{ij} in microscopic state by the expression through these factors:

$$T_{ij} = (a\{t,\theta_j(t), Nei(t), W_j, Env\} | \\ a\{t-1,\theta_i(t-1), Nei(t-1), W_i, Env\}) \quad (5)$$

And reward function in RL is also from these factors:

$$R = f(\theta_i(t), Nei(t), W_i(t), Env) \quad (6)$$

Through learning, the simulated decision-making process of agents is constantly approximate the 'stylized facts'. At micro level, action policy of agents can simulate decision-making process. At macro level, the state transfer probability can be calculated after one time step, then T_{ij} will be used to calibrate micro parameters. The process that parameterized the complex interaction will reduce computational complexity possibly.

The pseudo code of whole process is as shown in Algorithm.

The Reinforcement Learning is for simulating the decision-making process of agents in real system by designing the reward function of actions through considering parameters that affect the agent's state transfer.

For each step, the corresponding reward value is calculated and the Q matrix is updated. The corresponding q value consists of two parts: the immediate reward and the future reward. The attenuation coefficient γ in the future reward indicates the degree of importance that the agent attaches to the future reward. As the q value contains future reward, if the future reward is not large, the second part of the q value is relatively small, resulting in a small q value, then the probability of the agent selecting action a next time in state s is correspondingly decrease. The greedy coefficient ε in decision process is to prevent action selection from falling into a local optimum. If the greedy coefficient is not set, and the action with the highest current profit is selected in each decision, other q values cannot be selected, which makes it

Parameter Computation using Reinforcement Learning

Input: Number of agents, Greedy coefficient ε, Number of iteration N_{ite}.

Output: Calibrated parameter *para.*

Initialize Q, S
for *agent* **in** *Agents* **do**
  for *i* **in** N_{ite} **do**
    if *Random* < ε **then**
      $a \leftarrow Max(Q(s))$
    else
      $a \leftarrow Randomaction$
    end if
    $R(a) \leftarrow f(\theta_i(t), Nei(t), W_i(t), Env)$
    $Q(s) \leftarrow (1-\sigma) \cdot Q(s) + \sigma \cdot (R + r \cdot maxQ(s'))$
    $s \leftarrow s'$
  end for
end for
$T_{ij}^{Macro} \leftarrow \frac{n_{ij}}{N}$
$T_{ij}^{Micro} \leftarrow (a\{t\} | a\{t-1\})$
$Para \leftarrow T_{ij}^{Macro} = T_{ij}^{Micro}$ ▷Parameter Calibration

difficult for the agent to explore other actions and miss better strategies.

After the training is completed, the converged Q matrix is obtained. According to the Q matrix, the state transfer of agents in the next time step can be determined, so as to calculate the state transfer probability of the entire multi-agent system. From a microscopic point of view, the agent's own parameters can be used to represent the state transfer probability of another expression. The values of the two expressions are equal to complete the calibration of the microscopic parameters of agents.

IV. CASE STUDY: INTER-CITY TRAVELING OF POPULATION

A. Experiment Setting

In order to verify the method proposed in this paper, this section selects inter-city traveling of population, one of the classic application scenarios of ABM, for computational experiments. For comparison, we conduct experiments with the data used in the paper [15,31] and compare with machine learning based surrogate method and mean-field method.

The experiment selects China's population traveling behavior in the decade 2000-2010 to simulate. As one of the most populous countries in the world, China's population system is a representative multi-agent system. The data in this experiment mainly has three categories: census data, population sample and annual statistics. The census data is the fifth national census in 2000. Individuals in this data has 7 basic attributes. These are gender, age, residence city, ethnicity, registration province, marital status, and birth status. The population sample data is also disaggregated sample which is from census in 2000. It has 1,180,111 records, including personal and social attributes of individuals, and some other detailed information under the premise of protecting personal information. These records accounts 0.95% of the whole national population records. The census data and population sample are for generating the initial state, which is population in 2000 in this experiment. The annual statistics is from totally 361 cities, and records information such as average annual income, birth rate, death rate and other demographic characteristics of these cities.

In RL, we set a Q matrix. For corresponding to city number, the Q matrix has 361 rows and 361 columns. Then the elements in Q matrix is q value. For example, the (i, j) in Q matrix is transfer probability from i-th city to j-th city. This matrix is for training to learn the decision-making of agents. After this matrix converges, in the i-th row, the number of column which has the largest q is city number that the agent will transfer to in next step. The computation of Q matrix with iteration is:

$$Q(s,a) = (1-\sigma) \cdot Q(s,a) + \sigma \cdot (R + r \cdot maxQ(s')) \quad (7)$$

where σ is learning rate, $r \in (0,1)$ is the attenuation factor, $maxQ(s')$ stands for the future reward. R is reward computed from the following equation:

$$R = k - (E_{or} - E_{des}) \quad (8)$$

where k is an artificial constant. E_{or} is the evaluation of value in origin state and E_{des} in destination state. The value evaluation can be computed from the following equation:

$$E = \alpha \cdot \theta_i(t) + \beta \cdot Nei(t) + \gamma \cdot W_i(t) + (1-\alpha-\beta-\gamma) \cdot Env \quad (9)$$

where α, β, γ is the weight of each parameter. The total of the 4 weights is 1.

In this experiment, $\theta_i(t)$ is represented by registration parameter S_{reg}, which can be expressed as:

$$S_{reg} = \begin{cases} 1 & If\ the\ agent\ has\ a\ local\ reg \\ \dfrac{k}{dist(RestCity, RegCity)} & Otherwise \end{cases}$$

This function is from the famous Schelling's model [37]. where *dist* is a function that computes the distance between the agent's residential city and registration city. Constant k should satisfy the condition that $k \leq dist(RestCity, RegCity)$ so that $S_{reg} \leq 1$. Here we set

$$k = \min_{RestCity \neq RegCity} dist(RestCity, RegCity)$$

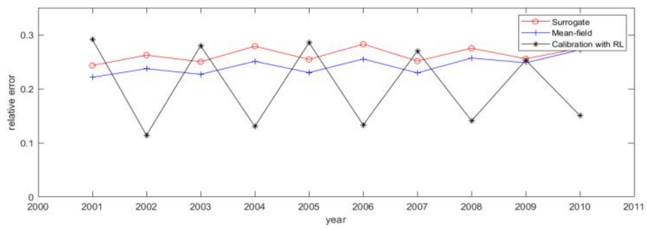

Fig. 2. Results of three methods of calibration.

$Nei(t)$ is represented by family parameter S_{fam}, which can be calculated by:

$$S_{fam} = \frac{n_{same}}{n}$$

where n_{same} means the number of family members that are in the same city as the agent, and n means the number of agent's all family members. $W_i(t)$ is represented by income parameter S_{income}. This is decided by agent's residential city and year. Env is represented by ethnic parameter S_{eth}, which can be expressed as:

$$S_{eth} = 1 - \frac{rank_{ResCity}}{n_{city}}$$

where $rank_{ResCity}$ means the rank of proportion of ethnic groups in agent's residential city and n_{city} means the number of cities. Here the ethnic parameter S_{eth} is designed for minorities.

The decision-making parameters W_i and environment influence parameters Env are complicated in realistic system. Many factors can influence the process of state transfer. To simplify the calculation, we select income factors that have a greater impact in the personal decision-making process as agent's decision-making parameters, and select ethnic settlement factors that have a greater impact in the social environment as environmental parameters.

B. *Experiment Results*

In the experiment, one agent is set to represent 10000 actual people. For evaluate the effectiveness of proposed method, we set relative error which can be calculated by:

$$error = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{n_{si} - n_{ai}}{n_{ai}}\right|$$

where N is the total city number, which is 361. n_{si} is the population number in the i-th city in simulation system and n_{ai} is in actual system. For baselines, we set experiment results shown in Ye's work [31] as baselines, and Mean-Field calibration method is also proposed in their work. The comparison is shown in Fig. 2 and Table I.

It can be seen that our method achieves lower relative errors, which indicates that our method performs better in the parameter calibration task of population migration experiments. In some years, such as 2002, 2004 and other even-numbered years, our method can greatly reduce the relative error, and in other years, the errors of three methods

TABLE I. RELATIVE ERRORS OF THREE METHODS OF CALIBRATION

	2001	**2002**	**2003**	**2004**	**2005**
Calibration with RL	29.15%	11.40%	27.95%	13.03%	28.52%
Mean-field Calibration	22.11%	23.71%	22.67%	25.07%	22.99%
Surrogate Calibration	24.30%	26.21%	24.99%	27.86%	25.42%
	2006	**2007**	**2008**	**2009**	**2010**
Calibration with RL	13.30%	26.96%	14.07%	25.28%	15.04%
Mean-field Calibration	25.50%	22.93%	25.68%	24.81%	27.20%
Surrogate Calibration	28.25%	25.12%	27.49%	25.54%	27.49%

are close. So it can be seen that that our method is better on the whole. From the trend of data, we can intuitively find that if the simulation time is increased by a few more years, the fluctuation of the errors of our method will gradually decrease and the performance will be better than the other two methods, which can be known from the data in 2009. And the results show that our method is not so robust in these 10 years, but it will be more robust in ten more years.

The running time of our methods is 2,257 seconds, and mean-field calibration needs 2,158 seconds. The two methods have similar time cost, and the difference in running time is only about 100 seconds. If we consider every related agents of an agent, that will add many new parameters because agents have their own factors. Our method considers these factors and represents them into only 4 parameters. This is the reason why our method can reduce the computational complexity. The process that parameterizing the interaction between agents simplifies the complexity of calculation to a certain extent. If we do not simplify the interaction but consider it directly, the running time will be greater. Moreover, this process keeps accuracy without ignoring the interaction between agents at the same time.

V. CONCLUSION

One of the main challenge for Agent-Based Model is its decision parameter learning for a reasonable and reliable simulation. While typical methods have limitations because ABM has complex structure especially if agents in simulation add, this paper introduces RL for parameter calibration. Reinforcement Learning is used to simulate the behavior of agents, and then we calculate state transfer probability. Then the microscopic parameters of agents are calibrated according to the state transfer probability. By a parameterization of the agent interactions, our method can further reduce the computational complexity with an acceptable accuracy.

One major limitation of our method is that the empirical setting of reward function in most cases cannot uniquely determine each parameter of the agent decision model. It only provides a constraint for a feasible domain of parameter set, parameters in feasible domain can satisfy constraints of calibration. Therefore, to achieve unique "optimal" values, we need to impose more heuristic rules in our future work, such as the maximum entropy or the minimum variation principles.

References

[1] P. Ye and F. Wang. "Hybrid Calibration of Agent-Based Travel Model using Traffic Counts and AVI Data". *IEEE International Conference on Intelligent Transportation Systems (ITSC 2017).* Oct. 16-19, 2017, Yokohama, Japan.

[2] Y. Xie, Y. Murphey, D. Kochhar. "Personalized Driver Workload Estimation Using Deep Neural Network Learning From Physiological and Vehicle Signals". *IEEE Transactions on Intelligent Vehicles*, 5(3): 439-448, 2020.

[3] Y. Chen, X. Wang, L. Li, F. Wang. "Traffic situational awareness research and development enhanced by social media data: the state of the art and prospects". *Chinese Journal of Intelligent Science and Technology*, 4(1): 1-13, 2022.

[4] Y. Li, X. Lu, J. Wang, K. Li. "Pedestrian Trajectory Prediction Combining Probabilistic Reasoning and Sequence Learning." *IEEE Transactions on Intelligent Vehicles* 5(3): 461-474, 2020.

[5] P. Ye, X. Wang, C. Chen, Y. Lin and F. Wang. "Hybrid agent modeling in population simulation: Current approaches and future directions." *Journal of Artificial Societies and Social Simulation*, vol. 19, no. 1, pp. 1−12, 2016.

[6] Z. Chen, Z. Zhan, Y. Lin, Y. Gong, T. Gu, F. Zhao, H. Yuan, X. Chen, Q. Li and J. Zhang. "Multi objective cloud workflow scheduling: A multiple populations ant colony system approach." *IEEE Transactions on Cybernetics,* vol. 49, pp. 2912−2926, 2019.

[7] L. Jared, J. Wen and P. L. Gurian. "Simulating the human-building interaction: Development and validation of an agent-based model of office occupant behaviors." *Building and Environment,* vol. 88, pp. 27 −45, 2015.

[8] H. Scott. "MayaSim: An Agent-Based Model of the Ancient Maya Social-Ecological System." *J. Artif. Soc. Soc. Simul*. vol. 16, 2013.

[9] P. Ye, T. Wang and F. Wang. "A Survey of Cognitive Architectures in the past 20 Years". *IEEE Transactions on Cybernetics*, 48(12): 3280-3290, 2018.

[10] P. Ye, X. Wang, W. Zheng, Q. Wei and F. Wang. "Parallel Cognition: Hybrid Intelligence for Human-Machine Interaction And Management". *Frontiers of Infor-mation Technology & Electronic Engineering*, 2022. (Accepted)

[11] P. Ye, S. Wang and F. Wang. "A General Cognitive Architecture for Agent-Based Modeling in Artificial Societies". *IEEE Transactions on Computational Social Systems*, 5(1): 176--185, 2018.

[12] P. Ye, B. Tian, Y. Lv, Q. Li and F. Wang. "On Iterative Proportional Up-dating: Limitations and Improvements for General Population Synthesis". *IEEE Transactions on Cybernetics*, 52(3): 1726-1735, 2022.

[13] P. Ye, F. Zhu, S. Sabri and F. Wang. "Consistent Population Synthesis with Multi-Social Relationships Based on Tensor Decomposition". *IEEE Transactions on Intel-ligent Transportation Systems*, 21(5): 2180-2189, 2020.

[14] P. Ye and X. Wang. "Population Synthesis using Discrete Copulas". *In Proceedings of IEEE International Conference on Intelligent Transportation Systems (ITSC 2018)*, Maui, Hawaii, USA, Nov. 4-7, 2018: 479-484.

[15] P. Ye, X. Hu, Y. Yuan and F.-Y. Wang. "Population synthesis based on joint distribution inference without disaggregate samples." *Journal of Artificial Societies and Social Simulation*, vol. 20, no. 4, pp. 16−16, 2017.

[16] H. Hu, G. Wen, W. Yu, J. Cao and T. Huang. "Finite-time coordination behavior of multiple euler − lagrange systems in cooperation-competition networks." *IEEE Transactions on Cybernetics,* vol. 49, pp. 2967−2979, 2019.

[17] P. Ye, X. Wang, G. Xiong, S. Chen and F. Wang. "TiDEC: A Two-Layered Integrated Decision Cycle for Population Evolution". *IEEE Transactions on Cybernetics*, 51(12): 5897-5906, 2021.

[18] C. Kerr, R. Cliff , M. Stuart. "Covasim: an agent-based model of COVID-19 dynamics and interventions." *medRxiv, 2020*.

[19] C. Silva, L. Petrônio, V. Paulo. "COVID-ABS: An agent-based model of COVID-19 epidemic to simulate health and economic effects of social distancing interventions." *Chaos, Solitons, and Fractals,* vol. 139, pp. 110088−110088, 2020.

[20] C. Erik. "An agent-based model to evaluate the COVID-19 transmission risks in facilities." *Computers in Biology and Medicine,* vol. 121, pp. 103827−103827, 2020.

[21] N. Magliocca, V. McConnell and M. Walls. "Integrating global sensitivity approaches to deconstruct spatial and temporal sensitivities of complex spatial Agent-Based Models." *J. Artif. Soc. Soc. Simul*. vol. 21, 2018.

[22] C. Gourieroux, A. Monfort, and E. Renault. "Indirect inference." *Journal of Applied Econometrics*, vol. 8, pp. 85−118, 1993.

[23] G. Fagiolo, M. Guerini, F. Lamperti, A. Moneta and A. Roventini. "Validation of Agent-Based Models in economics and finance." 2017.

[24] S. Alfarano, T. Lux and F. Wagner. "Estimation of Agent-Based Models: The case of an asymmetric herding model." *Computational Economics,* vol. 26, pp. 19−49, 2005.

[25] S. Chen, C. Chang and Y. Du. "Agent-based economic models and econometrics." *The Knowledge Engineering Review,* vol. 27, pp. 187−219, 2012.

[26] Grazzini, M. Jakob, G. Richiardi and M. Tsionas. "Bayesian estimation of Agent-Based Models." *Journal of Economic Dynamics and Control,* vol. 77, pp. 26−47, 2015.

[27] G. Andrew, J. Carlin, H. Stern, D. Dunson, A. Vehtari and D. B. Rubin. "Bayesian data analysis." 1995.

[28] F. Canova and L. Sala. "Back to square one: Identification issues in DSGE models." *Macroeconomics: Aggregative Models eJournal,* 2005.

[29] Fagiolo, Giorgio and A. Roventini. "Macroeconomic policy in DSGE and Agent-Based Models redux: New developments and challenges ahead." *Microeconomics: General Equilibrium & Disequilibrium Models eJournal,* 2016.

[30] F. Lamperti, A. Roventini and A. Sani. "Agent-Based Model calibration using machine learning surrogates." *Sciences Po publications,* 2017.

[31] P. Ye, Y. Chen, F. Zhu, Y. Lv, W. Lu and F. Wang. "Bridging the micro and macro: Calibration of Agent-Based Model using mean-field dynamics." *IEEE transactions on cybernetics,* 2021.

[32] Fagiolo, Giorgio, M. Guerini, F. Lamperti, A. Moneta and A. Roventini. "Validation of Agent-Based Models in economics and finance." *Simulation Foundations, Methods and Applications,* 2019.

[33] Gilbert, Nigel and S. C. Bankes. "Platforms and methods for agent-based modeling." *Proceedings of the National Academy of Sciences of the United States of America,* vol. 99, pp. 7197−7198, 2002.

[34] S. Yu, Y. Fan, L. Zhu and W. Eichhammer. "Modeling the emission trading scheme from an agent-based perspective: System dynamics emerging from firms' coordination among abatement options." *Eur. J. Oper. Res*. vol. 286, pp. 1113−1128, 2020.

[35] T. Fleiter, M. Rehfeldt, A. Herbst, R. Elsland, A. Klingler, P. Manz and S. Eidelloth. "A methodology for bottom-up modelling of energy transitions in the industry sector: The FORECAST model." *Energy Strategy Reviews,* vol. 22, pp. 237−254, 2018.

[36] Puterman, L. Martin. "Markov decision processes: Discrete stochastic dynamic programming." *Wiley Series in Probability and Statistics* 1994.

[37] T. Schelling. "Models of segregation." *The American Economic Review,* vol. 59, pp. 488−493, 1969.