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

Efficient Calibration of Agent-Based Traffic Simulation Using Variational Auto-Encoder*

Peijun Ye, Fenghua Zhu, Yisheng Lv, Xiao Wang and Yuanyuan Chen

***Abstract*— In agent-based traffic simulation, calibration is an essential stage before the models applied to reproduce the individual/group travel behaviors. While traditional methods suffer from a high computational complexity, this paper proposes an improved method to alleviate the computational burden for large-scaled simulations. Specifically, we introduce variational auto-encoder to compress the original agent state vector into a lower dimensional hidden space, where the state transfer probability is calculated fast. Then the probability is mapped into the original space through a decoder, to achieve the agent travel parameters. The dynamic calibration method is tested with other baselines in urban travel demand analysis. Experiment results demonstrate that our method brings about 19% elevation of efficiency with the same accuracy of calibration.**

I. INTRODUCTION

Urban transportation is a typical cyber-physical social system that integrates behaviors from travelers, schedulers, drivers and their interactions via networked machines. For the study of such distributed systems, Agent-Based Model (ABM) provides scientists and engineers a useful tool for their modeling, analysis and experiment. It characterizes behavioral modes of heterogeneous individuals/groups and grows emergent systemic features by their interaction, communication and learning from each other and from local surrounded environments. This micro-macro way, up to now, might be the "best" method that links the overall system dynamics with its potential micro causalities [1]. Thus, ABM is constantly studied by the transportation research community [2]–[7].

When ABM is created to describe realistic system phenomena, two steps are essential for its modeling and analysis. The first is the generation of a synthetic population that is grounded on the real target population. Research on this problem has yielded many fruitful results [8]–[10]. The second is to build an intelligent agent for each individual that characterizes his cognitive and decision-making features [11], [12]. In such a step, quantitative calibration is necessary to validate the results as reliable. Traditionally, this process refers to the adjustment of the model, keeping its output approximate to realistic "stylized facts" with an acceptable error. Such "stylized facts" usually exploit aggregate statistics from direct measurements of the real traffic system. Representatives in the line of work include Generalized Method of Moments (GMM) [13], Method of Simulated Moment (MSM) [14], Simulated Minimum Distance (SMD) [15] and Simulated Maximum Likelihood (SML) [16]. In the calibration, a distance between the aggregation of ABM output and the actual measured counterpart is minimized as

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} D(Y^R, Y^A, \boldsymbol{\theta})$$

where $Y^R, Y^A, \boldsymbol{\theta}$ stand for the selected moments from realistic data, the selected moments from artificial results and the parameter set respectively. D is an arbitrarily defined distance function. In practice, the moment-based optimization methods are easy to control since they directly treat the evaluation criterion as the optimized objective. However, the calibration requires many bottom-up trial-and-error iterations as there is typically no analytical closed forms of ABM.

A second approach for ABM calibration exploits the Bayesian estimation, which avoids complex numerical approximation to the disaggregate model [17]. In this framework, modelers need to infer the "most likely" parameters with a given observed statistics Y^R by solving

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} p\left(\boldsymbol{\theta} \middle| Y^R\right) = \arg\max_{\boldsymbol{\theta}} L(Y^R, \boldsymbol{\theta}) \cdot p(\boldsymbol{\theta})$$

where $p(\boldsymbol{\theta})$ and $p\left(\boldsymbol{\theta} \middle| Y^R\right)$ are called the prior and posterior distributions and $L(Y^R, \boldsymbol{\theta})$ is the likelihood. Generally, with enough training and trials, Bayes theorem guarantees the asymptotic optimality of calibration even if $p(\boldsymbol{\theta})$ is arbitrarily set. However, calculation of the likelihood over different $\boldsymbol{\theta}$ in its feasible space is much computationally expensive.

A third approach for ABM calibration introduces machine learning techniques, which successfully avoids redundant computations in model optimization. This approach estimates the agent parameters by using particular learning algorithms or by training a surrogate of the target system. Classic work of this category is surrogate learning, where a surrogate model is constructed to be the approximation of original ABM system via heuristic sampling and supervised learning [18]. While the method greatly reduces necessary running times of ABM, it suffers from the sampling bias so that the calibrated parameters are very likely to fall into a local optimum.

Recently, a novel method using mean-filed approach is proposed to solve the above problems. In contrast with the previous ones, it is an on-line calibration method from macro to micro levels [19]. The mean-field calibration starts

*This work was supported in part by National Natural Science Foundation of China under Grant 62076237, Grant T2192933, Grant U1811463, Grant U1909204, Grant 61906191, and in part by Youth Innovation Promotion Association Chinese Academy of Sciences under Grant 2021130.

Peijun Ye, Fenghua Zhu, Yisheng Lv and Yuanyuan Chen are all with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China. (Email: peijun_ye@hotmail.com)

Xiao Wang is with Qingdao Academy of Intelligent Industries, Qingdao, 266109, China.

2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC) | 978-1-6654-6880-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ITSC55140.2022.9922234

with inference of agent state transfer probabilities via the aggregated system model. Then, each state transfer probability is used to construct a micro transfer equation, from which the agent behavioral parameters are analytically or numerically solved. Albeit the mean-field calibration closes the gap between macro system metrics and micro agent models, its deficiency lies in the possible high dimensions of state space for large-scaled simulation, which elicits a complicated equation solving. To improve its performance, this paper introduces auto-encoder to compress the original state vector into a lower dimensional hidden space. Such a compression will greatly accelerate the model solving process and thus removes the bottleneck of its computation. Comparative experiments have indicated that the introduction of variational auto-encoder brings about 19% elevation of the computational performance, with the same level of calibration accuracies.

The organization of this paper is as follows. Section II elucidates our proposed method in details. Section III provides the experiment results. Specifically, we concentrate on the training of auto-encoder and the calibration effects of agent models. The paper concludes at last with some further discussions to shed light on potential future work.

II. DYNAMIC CALIBRATION BASED ON VARIATIONAL AUTO-ENCODER

Given a multi-agent system for traffic simulation, assume each agent is defined by a set of attributes and each value combination of all the attributes is defined as a state. An agent can stand for an individual or a group of traveler(s), an individual or a group of vehicle(s), etc.. To limit the scope, an agent in this paper represents a driver whose travel mode selects the private car. For simplicity, we assume each agent corresponds to a vehicle (this can be easily extended by introducing a vehicle/person travel ratio) and the attributes are all discrete (continuous ones can be pre-processed by a piece-wise discretization). Define a system state vector

$$x\left(t\right)=\begin{bmatrix}x_1(t) & \cdots & x_N(t)\end{bmatrix}^T\in\mathbb{R}^N$$

as the number of agents in each state at time index t. Here, a specific combination of the agent attributes (¡location=node 23, age=34¿ for example) is defined as an agent state. N is the total number of agent states. $x_i(t)$ stands for the number of agents in state i. For ease of computation, $x\left(t\right)$ here is defined as a real vector. When applied in real systems, it needs to be rounded to represent the agent number. The state transfer equation between adjacent time steps is

$$\left\{\begin{array}{l}x(t)=T(t)\cdot x(t-1)\\ x(0)=x_0\end{array}\right. \tag{1}$$

where x_0 is the initial distribution of agents. $T(t)\in\mathbb{R}^{N\times N}$ is the state transfer matrix in which the cell in the i-th row and j-th column is denoted as $T_{ij}(t)$. $T_{ij}(t)$ means the aggregate transfer probability from the agent state j to i. For system observation, let

$$y(t)=O[x(t)]$$

be the M-dimensional vector of detected metrics. Here we consider a linear measurement

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

where $B(t)\in\mathbb{R}^{M\times N}$ is called a measurement matrix. Denote the actual observation from realistic system as $\hat{y}(t)$. The performance of the multi-agent system is defined as

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

where K is the total number of observed steps and the positive symmetric matrix V represents the importance of each metric.

As a discrete calculus of variation problem, the optimal trajectory of $x\left(t\right)$ can be calculated by Euler equation

$$\left\{\begin{array}{l}\frac{\partial J(t)}{\partial x(t)}|_{x=x^*}=2[B(t)]^TV[B(t)x^*(t)-y(t)]=0\\ x^*(0)=x_0\end{array}\right.$$

Therefore, we need to solve

$$\left\{\begin{array}{l}B\left(t\right)x^*\left(t\right)-\hat{y}\left(t\right)=B\left(t\right)T\left(t\right)x^*\left(t-1\right)-\hat{y}\left(t\right)=0\\ x^*(0)=x_0\end{array}\right.$$

Since it is an under-determined system, we need to incorporate further constraints for its unique solution. The problem here is relaxed into an optimization as

$$\begin{aligned}T^*\left(t\right)&=\arg\min_{T}\|B(t)T(t)x^*(t-1)-\hat{y}(t)\|_2+\lambda\|T(t)\|_1\\ &\quad s.t.\ 0\leq T_{ij}(t)\leq 1\end{aligned} \tag{4}$$

where λ is a regularization parameter.

Eq. (1) to Eq. (4) define a system model where the overall agent state transfer probability, $T_{ij}(t)$, is computable from the observation in an on-line mode. For real transportation systems, however, a lot of factors such as social networks, personal travel costs, are all influential to individual/group's travels. This elicits a large dimension of $x\left(t\right)$, which makes the system model solving much time consuming. To this end, we introduce the variational auto-encoder (VAE) to compress the state space so that the computational efficiency could be improved. The structure of our VAE neural network is illustrated in Fig. 1, where the encoder learns a compressed representation in a latent space while the decoder completes an inverse task. At first, the system state vectors from adjacent time steps are encoded to achieve their embedded representations. This process relies on an encoder network with shared parameter values. Then, the state transfer equation in latent space is solved given the embedded states $z\left(t\right)$ and $z\left(t-1\right)$. Estimation of the latent state transfer probability is similar as $T\left(t\right)$

$$\begin{aligned}P^*\left(t\right)&=\arg\min_{P}\|P(t)z(t-1)-z(t)\|_2+\lambda\|P(t)\|_1\\ &\quad s.t.\ 0\leq P_{ij}(t)\leq 1\end{aligned} \tag{5}$$

As the dimension of compressed representation is much lower than the original one, such an optimization takes far less computational time to get a result. The achieved $P\left(t\right)$

is then flattened to the decoder, which transforms the latent state transfer probabilities into the original space. As such, $T_{ij}(t)$ is calculated in an efficient way.

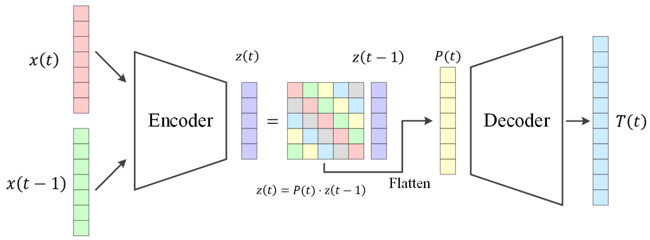

Fig. 1. Structure of VAE network. In each time step, adjacent states are encoded in the latent space and a state transfer probability matrix $P(t)$ is computed. Then the $P(t)$ is reconstructed into the original space through the decoder to get the agent state transfer probability $T(t)$.

As a supervised learning, training of the VAE network involves forward computation and backward propagation. The forward computation is explained in the last paragraph, as an "encoding, latent transfer probability solving, and decoding" process. Since the decoding step finally reconstructs state transfer probabilities in the original space, the backward propagation exploits a "real" label by directly solving Eq. (4). We adopt the loss of mean squared error as

$$loss = \mathbb{E}_{x \sim D(x)} \left[\left\| T(t) - T^{*}(t) \right\|_2 \right]$$

where $D(x)$ is the training data set.

Given a reasonable solution $T_{ij}(t)$, we further calibrate the agent model by establishing an equation between its micro behavioral parameters and the state transfer probability. Generally, there are four aspects that impact agent's travel behaviors: 1) individual states. This type of attributes can be calculated from his endogenous psychological/physiological states or his surrounded environment, such as the private car ownership, personal estimation of traffic situation, etc; 2) dependent states. This type of attributes mainly refers to the agent's social relations, which are determined by other agents. For example, a community rally or social appointment may cause individual's travel, and the arrangement of such events relies on the states of one's social neighbors. Therefore, simulating this kind of travel behaviors needs to investigate the agent's neighbors in his social network; 3) decision-making models. The agent decision-making models or algorithms map a given unified state (including both the individual and dependent states as a whole) into a specific action/strategy. Such models simulate human's behavior selection over his personal knowledge base. And the model parameters, which we will calibrate in this paper, may vary from agent to agent to reflect the heterogeneity of individuals; 4) stochastic factors. To simulate the randomness of human behaviors, a multi-agent system usually adopts a stochastic number generator over particular distributions (like the Gaussian distribution) so that some perturbations could be introduced in the decision-making process. By considering these four aspects, the state transfer probability of a particular agent can be denoted as

$$\begin{aligned} T_{ij}(t) = & T \Big\{ a\Big[t, Nei(t), W = w_j, \Theta(t), \xi\Big] \\ & \Big| \, a\Big[t-1, Nei(t-1), W = w_i, \Theta(t-1), \xi\Big] \Big\} \end{aligned} \tag{6}$$

Here $\Theta(t)$ stands for the parameter set of decision-making algorithms, and the stochastic factors, ξ, are assumed to be independent and identically distributed (i.i.d.). $Nei(t)$ means the aggregate state of the agent's neighbors and W means his individual states. Eq. (6) provides a basis for calculating state transfer probability of each agent, where the right side is determined by his decision-making model (note that the decision-making model could be represented in various forms such as explicit mathematical equations, a neural network or a knowledge base, etc..) and the left side is estimated by the VAE network before. Therefore, the model parameters can be analytically or numerically solved from this equation.

III. EXPERIMENTS FOR AGENT-BASED TRAFFIC SIMULATION

To validate the VAE calibration method proposed before, this section will conduct computational experiments for urban traffic simulation. It is essential for the travel demand prediction and the traffic control/management strategy test. The experiments also incorporate several previous approaches as benchmarks to comparatively verify the advantages of our VAE method.

A. Data Source and Experiment Setting

The test scenario is set to be the central district of Chengdu, a western city of China. The abstract road network contains about 37 intersections as nodes, 112 arterial roads as links. The studied region covers about 20 km^2, including 54 residential areas, 42 schools and hospitals, 25 large hotels, malls and commercial centers, 18 government office and central business districts, 5 libraries, stadiums and sports centers, 4 leisure plazas and parks, 4 tourism sights (see Fig. 2).

To analyze the individual travel demand, we develop an agent-based simulation system based on the Simulation of Urban Mobility (SUMO) platform, and an external calibrator using the proposed VAE method [20]. The framework of our experiment system is illustrated as Fig. 3. In every calibration interval, the SUMO simulation environment collects the number of agents in each micro state and sends aggregate results to the VAE calibrator. The calibrator computes the agent state transfer probabilities via a pre-trained VAE network and returns to the SUMO system. Then the simulation system changes each agent's travel parameters according to its received results. For evaluation, a certain number of sensors are set in the observable links to compare calibrated link flows with corresponding observed ones.

Our data source of experiment includes three types. The first type is a basic synthetic population, which generates

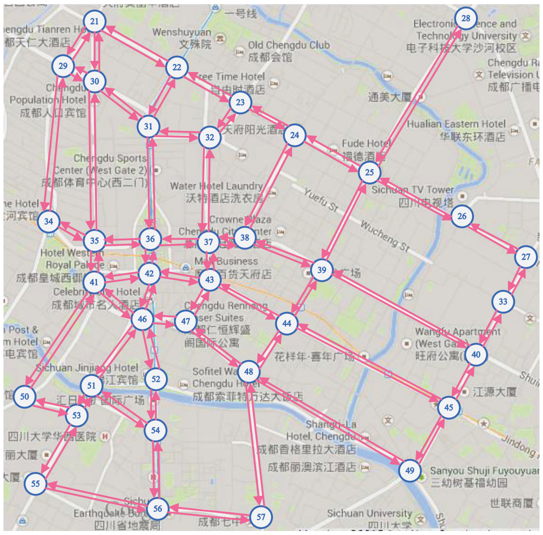

Fig. 2. The abstract road network covering about 37 intersections and 112 arterial roads.

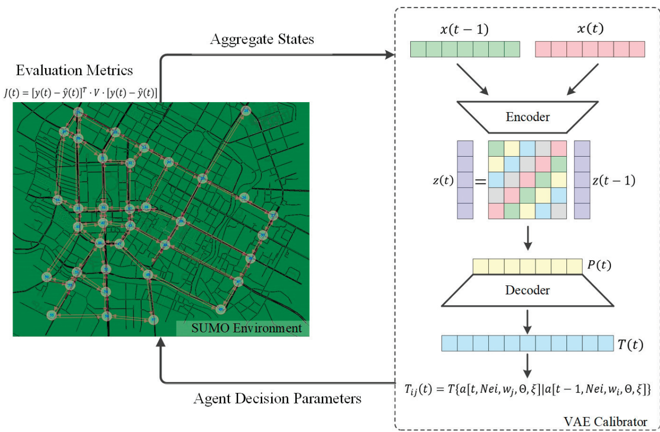

Fig. 3. Framework of the Experiment System.

initial states of all the agents before simulation. The population synthesis uses about 3,300 households with 10,000 individuals as a seed (accounting for 1.02‰of the total target population) and the investigations of family, schools and enterprises as constraints. The final virtual agents involve these three types of social relationships with their travels impacted by family members and school/corporation locations. Details of the synthesis can be referred to in [21]. The second type is the actual daily traffic counts detected from observable links. These data are collected from 225 loop detectors embedded in the road surface. The loop detectors record the number of passed vehicles every minute and send results to the transportation management center. Since the observable links do not cover the whole road network, the link traffic flows can only be deemed as a partial view of the total travel demand. Yet they could be used to reconstruct the overall OD flows between every node pair. The third type is active taxi locations from GPS devices installed in 13,608 taxies. When the GPS devices are turned on, they keep transmitting the location coordinates (represented as latitude and longitude) to the management center at intervals of 10 seconds. We select a raw dataset from a weekday, which contains 3,202,442 records of traffic counts and 77,645,666 records from taxis.

The data pre-processing before computational experiments involves three aspects. First, due to the possible malfunction of GPS devices, a small proportion of taxi records upload zero location coordinates. Similar problems may also take place in loop detectors, resulting in some missing data items in particular records. For these deficient records, we simply exclude them from our dataset. Second, some records seem obviously abnormal (such as the speed of 200 km/h). Such outliers are removed as well. Third, duplicate records may arise due to some transmission problems. For this kind, only one record would be retained and other duplicates are removed. After the data clean, about 0.06% traffic counts and 2.48% taxi records are filtered, which indicates that the noise ratio stays at an extremely low level. In the final cleaned datasets, the loop detector records mainly provide four metrics: volume, speed, occupancy and headway. The taxi GPS devices give latitude, longitude, direction, and speed. GPS locations are further converted into links through map matching and the links are connected to form the vehicle travel paths.

Our studied time is from 6:00 to 24:00, taking every 15 minutes as a calibration cycle. For each cycle, the calibrator estimates the total traffic flow between a given origin and destination node pair (OD pair). This estimation is according to Eq. (5) via the agent numbers $x(t-1)$ from SUMO and the pre-trained VAE network. Then, the calibrator searches all the travel paths between a given origin and destination node pair (OD pair), and computes the path selection probability of each agent. The computation is based on Eq. (6)

$$T_{ij}(t) = \sum_k T_k \Big\{ a[t, Nei, W=j, \Theta, \xi] \,\Big|\, a[t-1, Nei, W=i, \Theta, \xi] \Big\}$$

where T_k means the assignment probability of the k-th path from node i to node j. To get a unique solution, we further set each T_k to be proportional to the taxi path

$$\frac{T_k \Big\{ a[t, Nei, W=j, \Theta, \xi] \,\Big|\, a[t-1, Nei, W=i, \Theta, \xi] \Big\}}{\sum_k T_k \Big\{ a[t, Nei, W=j, \Theta, \xi] \,\Big|\, a[t-1, Nei, W=i, \Theta, \xi] \Big\}} = \frac{N_k \Big\{ W=j \,\Big|\, W=i \Big\}}{\sum_k N_k \Big\{ W=j \,\Big|\, W=i \Big\}}$$

where $N_k\{W=j|W=i\}$ stands for the number of taxi trajectories through path k.

B. Experiment Results

The computational experiments are conducted for 5 times with identical settings. We use the same amount of data to train our VAE model and test the ultimate calibration effect. Fig. 4 shows the training error of our VAE network, where the MSE loss is adopted. As can be seen, the loss gets nearly 240% at the beginning and rapidly drops while the training goes on. It decreases to nearly zero at the 120-th epoch around. The error curve also indicates that our training of VAE is steady as it is monotonic without oscillation.

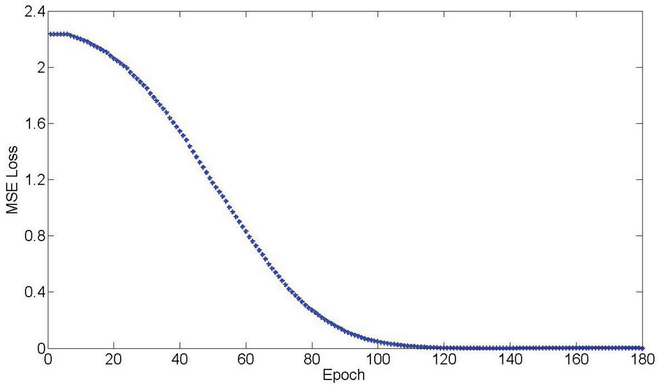

Fig. 4. Training Error of VAE Network.

Fig. 5 shows the mean average errors (MAE) of 5 experiments. The time intervals are numbered from 1 to 72 as the coordinates of X axis, with each index standing for a 15-minute time span. Each error point is computed by

$$MAE = \frac{1}{L}\sum_{i=1}^{L}\left|\frac{ActCount_i - SynCount_i}{ActCount_i}\right|$$

where $ActCount_i$ and $SynCount_i$ are the observed link traffic counts from real system and the detected traffic counts from our SUMO simulation. L is the number of observable links. We also include two previous calibration methods, machine learning surrogate and mean-field, as the comparative benchmarks. Clearly, the overall MAE of surrogate method is about 6% larger than the other two. And their errors all decline rapidly at the beginning of simulation and then keep stable with weak oscillations. This is because that the vehicles in the early morning are too few, so that even a small deviation will bring a large relative error. With the grown of travel demand, the errors keep stable at about 16%. However, while the proposed VAE method achieves a similar overall accuracy as the mean-field approach (about 13%), it suffers from greater oscillations. Fig. 6 illustrates the Root Mean Squared Errors (RMSE) of the three methods. Similarly, the surrogate approach gets the worst overall performance with such indicator being about 2500. By contrast, mean-field and VAE are 2123 and 2156, respectively, which shows a slight elevation. And the VAE model in this metric is also less stable than mean-field.

We further investigate the computational performance that is listed in Table I. For running time, the surrogate calibration takes about 26 hours while mean-field calibration costs about 23 hours. The reason behind lies in the iterative trial-and-error process in surrogate calibration whereas the mean-field method can dynamically conduct simulation and calibration. By contrast, our VAE method requires only 18.5 hours to complete the calibration with simulation. This reflects that the latent encoding of state variables can effectively reduce the computational complexity and thus lead to an efficient calibration. In general, our method brings about 19% improvement in the efficiency.

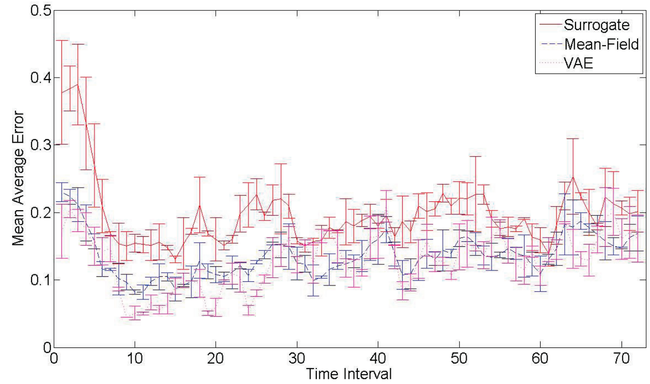

Fig. 5. Mean Average Errors of Calibration for Traffic Demand Prediction.

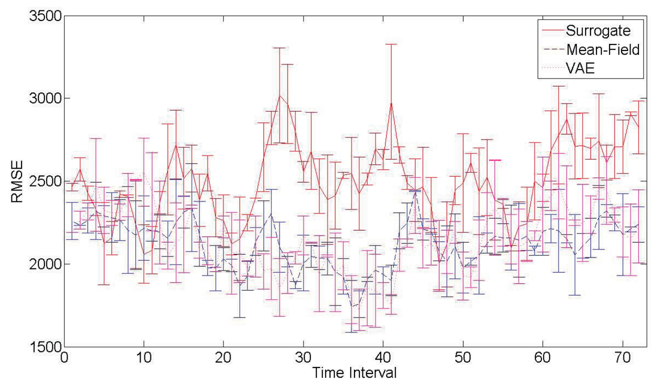

Fig. 6. RMSE of Calibration for Traffic Demand Prediction.

TABLE I

COMPUTATIONAL PERFORMANCE OF THREE CALIBRATION METHODS

	Surrogate	Mean-Field	VAE
Aver. Cal. &Sim. Time	26 H 03 Min.	22 H 54 Min.	18 H 27 Min.
Environment	Software: SUMO 1.3.1 + Python 3.6 OS: Windows 10 (x64) CPU: Intel Core i5-9400 (8 cores, 2.9 GHz) GPU: Nvidia GeForce GTX 1660 Ti (6 GB) RAM: 16 GB		

IV. CONCLUSIONS AND DISCUSSIONS

Agent-Based model provides a useful tool to analyze urban transportation systems where travelers, schedulers, drivers with their interactions mutually drive the system's evolution. Using an emergent paradigm, such type of models can simulate the systemic dynamics and predict travel demand for the test of traffic control/management strategies. This paper addresses the calibration of ABM, proposing an efficient method to dynamically calculate the agent decision parameters. The novel method introduces the variational auto-encoder to learn a latent representation of the system state. And such latent encoding can greatly reduce the dimension of system and accelerate its state transfer probability computation. Computational experiments based on actual traffic data have indicated that the proposed VAE method brings about 19% elevation of efficiency with the same accuracy of previous calibrations.

The VAE calibration in this paper, in essence, exploits the knowledge implied in the neural network. It can be deemed as a pre-training mode to learn state transfer features from samples in advance. This kind of priori will certainly improve the computational performance in subsequent dynamic calibration. In addition, the latent encoding by VAE further accelerates this process. And that is why our experiments achieve remarkable elevation in efficiency. However, greater compression of state vector may lose greater information, leading to a decrease of calibration accuracy. Therefore, we need seek a balance between the efficiency and accuracy. The thought of inverse learning might shed light on our future work [22].

Experiment results in this paper have already demonstrated that VAE model suffers from more fierce oscillation than mean-field approach. A potential reason for this phenomenon might be the lack of adaptability of VAE network after its training completed. Thus, if the samples for training include some bias that deviates from the test data, then the learned model may not be generalizable in other test cases. In the future work, using incremental learning or sampling for complex networks may be a feasible solution for such a problem [23].

REFERENCES

[1] P. Ye, X. Wang, W. Zheng, et al.. Parallel Cognition: Hybrid Intelligence for Human-Machine Interaction and Management. Frontiers of Information Technology & Electronic Engineering, 2022. (In Press)

[2] P. Ye and F.-Y. 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.

[3] F.-Y. Wang, P. Ye and J. Li. Social Intelligence: The Way We Interact, The Way We Go. IEEE Transactions on Computational Social Systems, 2019, 6(6): 1139–1146.

[4] P. Ye, X. Wang, G. Xiong, et al.. TiDEC: A Two-Layered Integrated Decision Cycle for Population Evolution. IEEE Transactions on Cybernetics, 2021, 51(12): 5897–5906.

[5] F.-Y. Wang, J. Guo, G. Bu and J. Zhang. Mutually Trustworthy Human-Machine Knowledge Automation and Hybrid Augmented Intelligence: Mechanisms and Applications of Cognition, Management, and Control for Complex Systems. Frontiers of Information Technology & Electronic Engineering, 2022. (In Press).

[10] P. Ye, B. Tian, Y. Lv, et al.. On Iterative Proportional Updating: Limitations and Improvements for General Population Synthesis. IEEE Transactions on Cybernetics, 2022, 52(3): 1726–1735.

[6] T. Liu, B. Tian, Y. F. Ai, et al.. Parallel Reinforcement Learning-Based Energy Efficiency Improvement for A Cyber-Physical System. IEEE/CAA Journal of Automatica Sinica, 2020, 7(2): 617–626.

[7] Z. Liu, N. Wu, Y. Qiao, et al.. Performance Evaluation of Public Bus Transportation by Using DEA Models and Shannon's Entropy: An Example from A Company in A Large City of China. IEEE/CAA Journal of Automatica Sinica, 2021, 8(4): 779–795.

[8] P. Ye, X. Hu, Y. Yuan, et al.. Population Synthesis Based on Joint Distribution Inference without Disaggregate Samples. Journal of Artificial Societies and Social Simulation, 2017, 20: 4.

[9] 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.

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

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

[13] S. Alfarano, T. Lux and F. Wagner. Estimation of Agent-Based Models: The Case of an Asymmetric Herding Model. Computational Economics, 2005, 26(1): 19–49.

[14] S.-H. Chen, C.-L. Chang and Y.-R. Du. Agent-Based Economic Models and Econometrics. The Knowledge Engineering Review, 2012, 27(2): 187–219.

[15] A. Fabretti. On the Problem of Calibrating an Agent Based Model for Financial Markets. Journal of Economic Interaction and Coordination, 2013, 8(2): 277–293.

[16] J. Kukacka and J. Barunik. Estimation of Financial Agent-Based Models with Simulated Maximum Likelihood. FinMaP-Working Paper, Financial Distortions and Macroeconomic Performance, Kiel University, Germany, 2016.

[17] J. Grazzini, M. G. Richiardi and M. Tsionas. Bayesian Estimation of Agent-Based Models. Journal of Economic Dynamics and Control, 2017, 77: 26-47.

[18] F. Lampertia, A. Roventinib and A. Sani. Agent-Based Model Calibration using Machine Learning Surrogates. Journal of Economic Dynamics & Control, 2018, 90: 366–389.

[19] P. Ye, Y. Chen, F. Zhu, et al.. Bridging the Micro and Macro: Calibration of Agent-Based Model Using Mean-Field Dynamics. IEEE Transactions on Cybernetics, 2021 (Accepted).

[20] D. Krajzewicz, J. Erdmann, M. Behrisch, et al.. Recent Development and Applications of SUMO—Simulation of Urban Mobility. International Journal on Advances in Systems and Measurements, 2012, 5(3&4): 128–138.

[21] P. Ye, F. Zhu, S. Sabri, et al.. Consistent Population Synthesis with Multi-Social Relationships Based on Tensor Decomposition. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(5): 2180–2189.

[22] M. Menner, K. Berntorp, M. N. Zeilinger, et al.. Inverse Learning for Data-Driven Calibration of Model-Based Statistical Path Planning. IEEE Transactions on Intelligent Vehicles, 2021, 6(1): 131–145.

[23] D.-D. Zhou, B. Hu, Z.-H. Guan, et al.. Sampled-Data State Estimation for Complex Networks With Partial Measurements. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2020, 50(11): 4787–4795.