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

Bayesian Optimization for Multi-Agent Routing in Markov Games

Zhenyu Shou, Xu Chen, Xuan Di *Member, IEEE*

***Abstract*— This paper aims to understand how to guide travelers' routing behavior toward a system optimum, using a bilevel game in a multi-agent traffic environment. With the goal to optimize some systematic objective (e.g., overall traffic condition) of city planners, we formulate a Stackelberg game with the upper level as the planner and the lower level as a multi-agent Markov game in which each rational and selfish traveler aims to minimize her travel cost. We employ a Bayesian optimization method on the upper level to solve for optimal controls of city planners and a mean field multi-agent deep Q learning approach to solve for optimal route choices of travelers on the lower level. We demonstrate the effect of two administrative measures, namely tolling and signal control, on the behavior of travelers on the Braess network and a large-sized real-world road network, respectively. Braess paradox, which is usually defined in static user equilibrium, is also defined and discovered in the context of the multi-agent Markov game.**

Key-words: Stackelberg Game, Bayesian Optimization (BO), Markov Routing Game

I. Introduction

With a growing number of agents relying on GoogleMaps or other navigation tools, dynamic routing games in a multi-agent system have been proposed to understand travelers' routing behavior. [1] develops a Markov routing game in which decentralized dynamic routing behavior of agents is modeled while accounting for traffic congestion arising from interactions among one another. In the Markov routing game, drivers' routing behavior is non-cooperative and individual goals could deviate from a system optimum. This raises one question of interest: *with selfish and rational travelers aiming to minimize their individual travel cost, how would a policymaker (aka the planner) guide the behavior of traveler toward some desirable outcome via planning or policy countermeasures?* This paper aims to understand how to guide travelers' routing behavior toward a system optimum. We formulate the interaction between travelers and the planner as a bilevel Stackelberg game and devise a Bayesian optimization scheme on the upper level to solve for optimal controls by the planner.

A. Related work

Bilevel network design problems (NDP) has been extensively studied in transportation planning [2]. A Stackelberg (or leader-follower) game is formulated in which the upper level player selects a control that impacts the routing behavior of lower level travelers while the lower level followers update their routing choice.

In the lower level problem using equilibrium constraints, static [3]–[5] or dynamic user equilibria (DUE) [6]–[8] are usually assumed and these equilibria can be directly solved from model-based optimization. In this paper, we assume no knowledge of system transition dynamics nor reward functions and thus, travelers' dynamic en-route choices are modeled as a Markov game and solved by multi-agent reinforcement learning (MARL). In other words, MARL is model-free and thus, agents need to explore and learn the environment for optimal driving policies, which is more challenging to solve for social optima.

MARL is a framework for modeling one's sequential decision-making processes while accounting for its interaction and competition with other agents. Applying it to routing a large number of agents on a road network has been largely understudied. [9] modeled multi-agent interaction but independent tabular Q-learning is implemented without considering information of other agents. To account for traffic congestion and stabilize training, [1] formulated the dynamic routing of multiple agents as a Markov game and then applied a mean field deep Q-learning algorithm to solve an equilibrium. Building on the framework developed in [1], in this paper, we will develop a bilevel optimization.

For the bilevel optimization, [10] first applied Bayesian optimization (BO) to MARL in the context of driver repositioning. However, applying BO to a Markov routing game (MRG) requires to model every agent's route choice on a graph, which is more challenging and is thus the focus of this paper.

B. Contributions of this paper

This paper aims to solve optimal controls for both the upper level planner and the lower level self-motivated agents whose goals differ. Simply increasing transportation infrastructure supply may lead to undesirable outcomes due to the existence of the Braess paradox. Thus, we propose a bilevel optimization with the upper level as the planner using countermeasures like tolling or traffic signal control to optimize some systematic objective (i.e., the average travel time of all agents), while the lower level as a multi-agent reinforcement learning (MARL) where each agent aims to minimize her own travel time. Since the bilevel optimization is challenging to solve, especially with MARL as the lower-level dynamic equilibrium, we propose a Bayesian optimization embedded with a multi-agent deep Q learning approach.

(Corresponding author: Xuan Di.)
Zhenyu Shou and Xu Chen are with the Department of Civil Engineering and Engineering Mechanics, Columbia University, New York City, NY 10027 USA (e-mail: zs2295@columbia.edu, xc2412@columbia.edu).
Xuan Di is with the Department of Civil Engineering and Engineering Mechanics, Columbia University, New York, NY, 10027 USA, and also with the Data Science Institute, Columbia University, New York, NY, 10027 USA (e-mail: sharon.di@columbia.edu).

2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC) | 978-1-6654-6880-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ITSC55140.2022.9922110

The remainder of the paper is organized as follows. Section II introduces the Markov routing game. Section III introduces the bilevel optimization where city planners interact with travellers in the Markov routing game. A case study about tolling on the Braess network is presented. Section IV presents optimal traffic signal control over a real-world large-scale road network. Section V concludes this study.

II. Markov routing game and multi-agent reinforcement learning algorithm

In this section, we will briefly introduce the Markov routing game (MRG) developed in [1]. The MRG is a partially observable Markov decision process represented by $\langle N, S, O, A, P, R, \gamma \rangle$. Each component is specified below:

- N. The total number of controllable agents;
- $\mathbf{s} \in S$. Environmental state $\mathbf{s}$ represents the distribution of travellers on road networks. $\mathbf{s}$ is not fully observable to agents.
- $\mathbf{o} \in O$. Each agent $i \in \{1, 2, \cdots, N\}$ draws a private observation, denoted as $o_i = (n, t)$. n is the node and t is the time step.
- $\mathbf{a} \in A$. The action set consists of all route choices at each node on road networks.
- P. The joint action among agents triggers a state transition $\mathbf{s} \rightarrow \mathbf{s}'$ based on the transition probability $P(\mathbf{s}'|\mathbf{s}, \mathbf{a})$.
- $r \in R$. The reward can be negative travel cost such as travel time and distance in the context of route choice.
- γ. The discount factor of the future reward.

Note that in MARL, the high dimension of joint action space among agents would make the Q-value function infeasible to evaluate. To tackle this challenge, [1] applies the mean field approximation [11] to evaluate the Q-value function, which is formulated as: $Q_i = Q(o_i, a_i, \bar{a}_i)$. $\bar{a}_i$ is defined as a mean action, representing the traffic flow on the link that is chosen by agent i. The mean field multi-agent deep Q-learning (MF-MA-DQL) algorithm to solve MRG is summarized in Algorithm (1).

Algorithm 1 MF-MA-DQL

```
Input: exploration parameter ϵ = ϵ_0, learning rate η = η_0
Initialize a DQN Q(o, a, ā|θ), a target network Q(o, a, ā|θ^−)
for episode ← 1 to T do
    while s is not terminal do
        Each agent selects action using ϵ-greedy policy
        Update state s → s′ and observe mean action ā_i
        Store (o_i, a_i, o_i′, r_i, ā_i) into replay buffer
    end while
    Sample a batch from replay buffer
    Update parameter θ and optimal policy of agents
    Decay ϵ and η
    Update parameter θ^− every τ periods, i.e., θ^− ← θ
end for
```

III. Bilevel optimization for multi-agent route choice

In this section, we propose a leader-follower (Stackelberg) game between city planners and travelers in order to understand how to guide routing behaviors of travellers. On the upper level, city planners (i.e., the leader) impact traveller' routing behavior through operational measures. The lower level is the MRG among travellers (i.e., the follower). We also develop an algorithm based on Bayesian optimization and MF-MA-DQL to solve the bilevel game.

We first aim to investigate the effect of tolling a critical link on the overall traffic network. Note that from the perspective of travelers, the overall travel cost is the summation of their travel time and the toll charge paid out of their pocket. Without loss of generality, we assume that the travel time and toll charge could directly be added without unit conversion. In other words, we assume travelers value their time and monetary cost equally. We study the effect of tolling a link on the Braess network, as presented in Figure (1). Travelers move from origin node n_0 to terminal node n_3, and the travel time on link l_{21}, i.e., $\Delta t_{21} = \alpha$. We regard α as the toll charge on link l_{21}. A larger α means that the toll is higher, indicating that there might be fewer travelers choosing the link; while a smaller α means that the toll is lower, indicating that more travelers may choose this link. We aim to find an optimal value of α with which some overall systematic performance, denoted as f, is optimized.

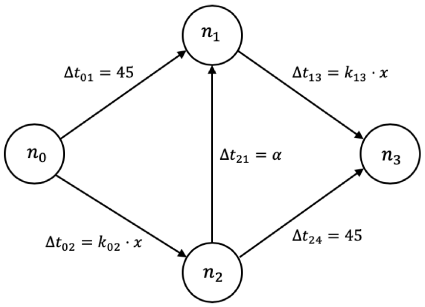


Fig. 1: Braess network

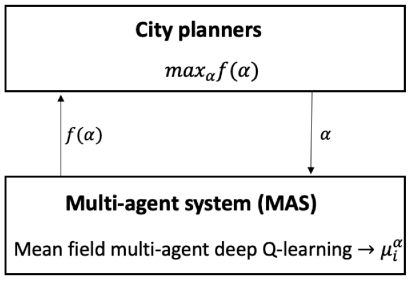


Fig. 2: Bilevel optimization

As aforementioned, behavior of travelers change with the adjustment in α. In this study, we assume that agents are perfectly rational and leave bounded rationality [12] for future research. Recall that our goal is to find the optimal α, therefore we formulate this as a bilevel optimization problem, as presented in Figure (2). The upper level is the planner aiming to maximize the systematic performance f via the control parameterized by α. In the lower level MARL, each agent $i \in \{1, 2, \cdots, N\}$ derives her optimal policy μ_i^α by the developed MF-MA-DQL algorithm presented in the previous section. Note that we use superscript α in the notation

of optimal policy, i.e., μ_i^α, to signify the dependency of optimal policy on the control parameter α. With the optimal behavior of all agents, the planner observes the systematic performance $f(\alpha)$ and adjusts α until reaching optimum.

Due to the unknown complex structure of f over α, traditional gradient based methods are not applicable. In addition, evaluation of f at one value of α using MARL is computationally expensive. Therefore, we resort to Bayesian optimization (BO) [13] that does not require the gradient information and is especially efficient to optimize some objective that is expensive to evaluate. The procedure of BO is as follows. First, BO places a statistical model on the objective function f, such as a Gaussian process. Second, BO devises an acquisition function such as upper confidence bound (UCB) [14] to decide where to evaluate the next, i.e., to choose an α based on the statistical model. Third, BO updates the statistical model based on the newly evaluated α, and the process repeats. The pesudo-code of BO is listed in Algorithm (2) [10].

Algorithm 2 Bayesian Optimization

1: Initialize a Gaussian process prior on f
2: Evaluate f at n_0 different α according to certain rules
3: Set computational budget K and $n = n_0$
4: **for** $n \leftarrow n_0$ to K **do**
5: Update posterior probability distribution on f based on all evaluated α
6: Calculate an acquisition function
7: Locate the α_n which maximizes the acquisition function
8: Evaluate f: use MF-MA-DQL to solve MRG given α_n
9: **end for**
10: Return α_n which maximizes f

A. Numerical example

We now apply the bilevel optimization to the Braess network presented in Figure (1). The rationale of adopting the Braess network in this section is as follows. First, the Braess network is simple, and usually an analytical solution could be derived, meaning that we could compare our numerical solution to its analytical counterpart. Second, recall that our goal is to find an optimal α with which the overall systematic performance (i.e., average travel time of all agents in the Braess network) is optimized. The existence of Braess paradox in the Braess network makes our goal nontrivial.

1) Lower level: We first validate the developed MF-MA-DQL algorithm in two cases, a single-batch demand and a multi-batch demand. In both cases, we initialize a multilayer perceptron (MLP) with three hidden layers $(32, 16, 8)$ to approximate the centralized Q function. ReLU is the activation function used between hidden layers. Learning rate $\eta = 10^{-3}$. Exploration parameter ϵ is initially set as 0.1, i.e., $\epsilon_0 = 0.1$, and linearly decreases to 0.01.

The single-batch demand level initially at origin node n_0 is set as 40, and $k_{02} = k_{13} = 1$. We further assume $\alpha = 100$, i.e., a very large number, indicating that link l_{21} is not supposed to be used by any agent. The convergence of the MF-MA-DQL algorithm is presented in Figure (3). The y-axis is the average travel time of all 40 agents from origin node n_0 to terminal node n_3. The x-axis shows the number of episodes. The derived optimal policy shows that 20 agents choose route $n_0 \rightarrow n_1 \rightarrow n_3$ and the other 20 choose route $n_0 \rightarrow n_2 \rightarrow n_3$.

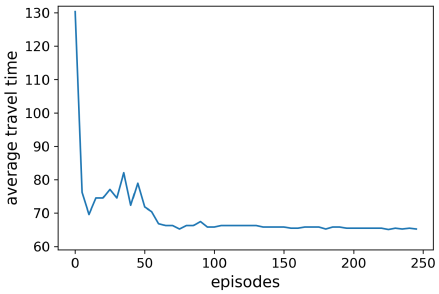


Fig. 3: Convergence of MF-MA-DQL with single-batch demand

The analytical solution for this case is derived as follows. Note that the route $n_0 \rightarrow n_1$ strictly dominates the one $n_0 \rightarrow n_2 \rightarrow n_1$, because $\Delta t_{02} + \Delta t_{21} > \Delta t_{01}$ always holds. Similarly, route $n_2 \rightarrow n_3$ strictly dominates the route $n_2 \rightarrow n_1 \rightarrow n_3$. Therefore, rational agents would not use l_{21}. Consequently, there are only two reasonable routes from n_0 to n_3, namely $n_0 \rightarrow n_1 \rightarrow n_3$ and $n_0 \rightarrow n_2 \rightarrow n_3$. Assuming there are y agents choosing the former and $40 - y$ agents choosing the latter, at equilibrium, if both routes are used, the travel time on both routes should be the same. Mathematically, $45 + k_{13} \cdot y = k_{02} \cdot (40 - y) + 45$, where the left hand side is travel time on the former route and the right hand side is travel time on the latter route. We have: $y = \dfrac{40 \cdot k_{02}}{k_{02} + k_{13}}$. Plugging $k_{02} = k_{13} = 1$, $y = 20$, meaning that at equilibrium, 20 agents choose the former route and 20 choose the latter. Travel time for all agents at equilibrium is 65. Both travel time and optimal policy from previous numerical solution agree well with their analytical counterparts from the analytical solution. Thus this case validates the effectiveness of the developed MF-MA-DQL algorithm.

The aforementioned scenario is symmetric, because travel time on both routes is $45+x$, where x is the flow (i.e., number of agents) choosing the route. To further test the effectiveness of the MF-MA-DQL algorithm in asymmetric scenarios, we break the link symmetry by keeping $k_{02} = 1$ and varying k_{13}. Specifically, we test the MF-MA-DQL algorithm with 10 different values of k_{13}, namely $k_{13} \in \{1, 2, \cdots, 10\}$. Other parameters remain unchanged. When $k_{02} = 1$, we have $y = \frac{40}{1+k_{13}}$. The travel time for all agents is $45 + \frac{40 \cdot k_{13}}{1+k_{13}}$. Note that the analytical solution applies to both integer and fractional k_{13}. The comparison of average travel time between the numerical solution (i.e., using MF-MA-DQL) and analytical solution is presented in Figure (4). The y-axis is the average travel time of all agents after convergence under a given k_{13}. All red dots (i.e., numerical solution) are on the blue curve (i.e., analytical solution), indicating a very good agreement between the numerical and analytical solution. This validates the effectiveness of the developed MF-MA-DQL algorithm in asymmetric cases.

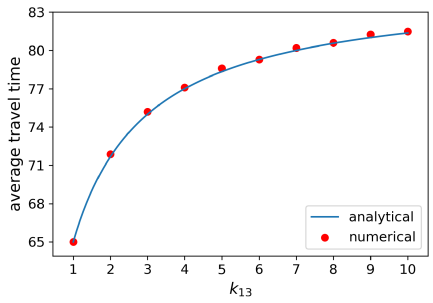


Fig. 4: Comparison between numerical solution and analytical solution with varying k_{13}

In the case of multi-batch demand, the initial travel demand at origin node n_0 is 40. In addition, another 20 agents will depart from n_0 at time $t = 10$. For notation simplicity, we call 40 agents who depart from n_0 at $t = 0$ the first 40 agents and 20 agents who depart from n_0 the latter 20 agents. In this case, the route choice of the first 40 agents impacts that of the latter 20 agents, because the first 40 agents already occupy some links when the 20 agents enter the network and thus travel time on those links could be larger. We further assume $\alpha = 1$ in this case.

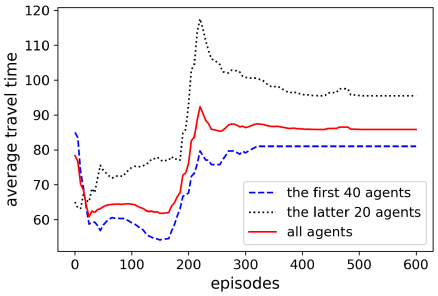
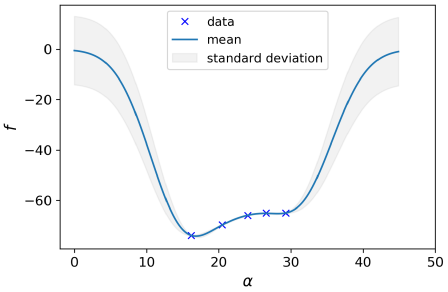


Fig. 5: Convergence of MF-MA-DQL with multi-batch demand

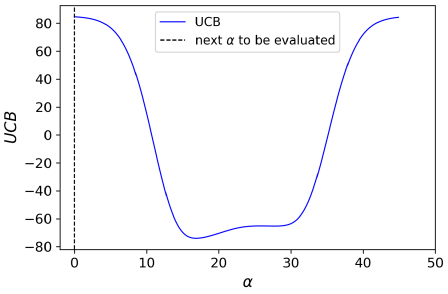
Figure (5) presents the average travel time of the first 40 agents, the latter 20 agents, and all 60 agents, respectively, versus the number of training episodes. The average travel time of the first 40 agents decreases fast, then bounces back and forth below 60, and finally increases fast to around 80. As for the latter 20 agents, their average travel time and optimal policies are strongly affected by the first 40 agents. After the average travel time of the first 40 agents converges after 300 episodes, it takes another 100 episodes for the average travel time of the latter 20 agents to converge. This is as expected because the first 40 agents impacts of the latter 20 agents, but not vice versa. After 400 episodes, the average travel time of the latter 20 agents converges to 95, which is consistent with the analytical solution.

2) Bilevel network design problem: After validating the MF-MA-DQL algorithm in the lower level MARL, we now run bilevel optimization with $k_{02} = k_{13} = 1$ and α as the control variable of the upper level planner. With a single-batch demand, we aim to find an optimal α with which the average travel time of 40 agents is optimized. To be precise, the negative average travel time of these 40 agents is taken as the upper level objective f. The range of α is set as $[0, 45]$.

BO is applied to solve the bilevel problem. As for the initial point of BO, we evaluate f at five randomly sampled α's. Figure (6(a)) presents the posterior probability distribution of f conditioned on these five initially evaluated α's. We select UCB as the acquisition function, plotted in Figure (6(b)).



(a) Posterior probability distribution of f conditioned on the initial evaluated five α's



(b) Acquisition function

Fig. 6: Posterior probability distribution and acquisition function at the 0^{th} iteration

The computational budget is set to be 15 iterations. The posterior probability distribution of f along with the analytical solution is plotted in Figure (7). We then compare it to its analytical solution, of which the derivation is detailed below. From node n_0 to node n_3, there are in total three possible routes, namely, route $n_0 \to n_1 \to n_3$, route $n_0 \to n_2 \to n_1 \to n_3$, and route $n_0 \to n_2 \to n_3$. Assuming there are y_1 agents choosing the first route, y_2 agents choosing the seconds route, and $40 - y_1 - y_2$ choosing the last route, at equilibrium, travel time on these routes is equal, if all used. Therefore, we have $45 + y_1 + y_2 = y_2 + (40 - y_1 - y_2) + (y - 1 + y_2) = y_2 + (40 - y_1 - y_2) + 45$ and $y_1 = \alpha - 5$, $y_2 = 50 - 2\alpha$. In addition, the constraints on the agent number choosing a route are $0 \le y_1 \le 40$ and $0 \le y_2 \le 40$. These constraints yield $5 \le \alpha \le 25$. Actually, with $\alpha < 5$, it can be easily seen that only route $n_0 \to n_2 \to n_1 \to n_3$ is used by agents; while with $\alpha > 25$, route $n_0 \to n_2 \to n_1 \to n_3$ is not used by any agent. Therefore, the objective f versus α could be solved as

$$f(\alpha) = \begin{cases} -(80+\alpha), & \alpha < 5 \\ -(90-\alpha), & 5 \le \alpha \le 25 \\ -65, & \alpha > 25. \end{cases}$$

The analytical solution is plotted as the red dotted line in Figure (7). The overall good agreement between the

analytical solution and the numerical solution validates the bilevel optimization.

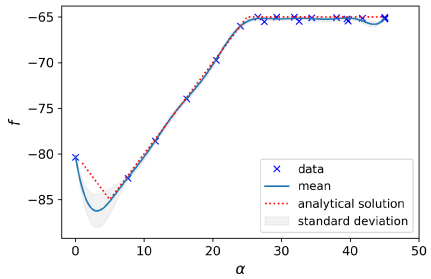


Fig. 7: Posterior probability distribution of f

Let us analyze the emergence of Braess paradox. With $\alpha = 0$ (i.e., no toll), the optimal route choice, from both the numerical and analytical solution, for all agents is to use route $n_0 \rightarrow n_1 \rightarrow n_2 \rightarrow n_3$, and the average travel time of all agents is 80; with $\alpha > 25$ (i.e., a large toll), the optimal route choice is half of agents using route $n_0 \rightarrow n_1 \rightarrow n_3$ and the other half using route $n_0 \rightarrow n_2 \rightarrow n_3$, and the average travel time of all agents is 65. This indicates that a small toll (i.e., a shorter travel time) on link l_{12} actually increases the overall travel time of all agents, resulting in the emergence of Braess paradox. In other words, decreasing the travel cost on a link by either expanding the capacity of the link or reducing the toll charge on the link may not benefit the overall traffic condition. In contrast, decreasing the travel cost on a link may deteriorate the overall traffic condition by attracting too many travelers to the link. Thus the optimal toll pricing on link l_{12} is a travel time greater than or equal to 25, which yields the optimal systematic objective.

IV. Case Study

We apply the developed bilevel optimization to a real-world road network with 69 nodes and 166 links, as presented in Figure (8). Traffic on the road network is simulated in SUMO. We demonstrate the set-up of the traffic environment in the bilevel game and present numerical results.

A. Bilevel network design problem

1) Lower level: The traffic environment of the MRG on the lower level is similar to [1]. Vehicles on the road network consist of controllable agents and background traffic. Controllable agents learn from interactions with others and adapt their route choices. Background vehicles are non-strategic players who follow some prescribed travel pattern. In this case study, there are four groups of controllable agents: 1) three agents travel from node 14 to node 60, 2) three agents from node 15 to node 60, 3) three agents from node 48 to node 1, and 4) three agents from node 69 to node 1. With respect to the background traffic, there are in total around 1,600 vehicles in the south-north direction and 500 vehicles in the east-west direction within the simulation time period (i.e., 1000 seconds).

2) Upper level - signal control: With the controllable agents aiming to minimize their travel time and background traffic following a prescribed traffic profile, city planners can affect the route choice behavior of adaptive controllable agents by adjusting the traffic signals at intersections, i.e., signal control. The goal of city planners is to develop a proper signal control scheme so that the average travel time of all controllable agents is minimized. We assume that the planner only adjusts traffic signals on Broadway. In addition, we assume that the duration of green and red phases is 60 seconds and 30 seconds, respectively. The decision variable is the offset between green lights of two consecutive intersections along Broadway.

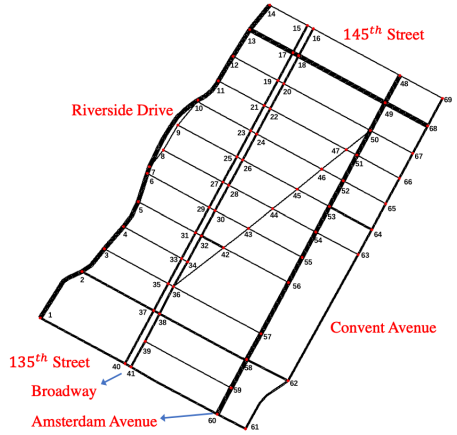


Fig. 8: Road network in SUMO

Denoting the negative average travel time of all controllable agents as f and the offset as α. With different values of α (i.e., the offset), controllable agents may experience different travel times. For example, with a proper α, controllable agents may take advantage of "green wave" on Broadway and thus spend less time reaching their destination, leading to a larger f. In contrast, with a poorly chosen α, controllable agents may need to stop frequently and spend more time on waiting at the intersection, resulting in a smaller f. The goal is to find an optimal α that maximizes f, i.e., $\alpha^* = \text{argmax} f(\alpha)$.

B. Numerical results

Figure (9) presents the distance between two consecutive evaluated α's in the BO algorithm. A smaller distance means that BO chooses to evaluate similar α's, indicating that the algorithm approaches convergence. The BO algorithm is stopped when the distance is smaller than a threshold value of 2.5 for four times in a row. With five randomly selected α's as a starting point, BO reaches convergence after 9 additional iterations.

With the illustrated convergence of both levels, the final result of the posterior probability distribution of the objective f (i.e., the negative average travel time) is shown in Figure (10). The x-axis is the offset α. The y-axis is the objective f. The mean and the standard deviation of the Gaussian process fitting based on the data are also plotted. As one could see, the standard deviation is small around

evaluated α's while large when there is no nearby evaluated α's. The final result suggests that $\alpha^* = 4$. With the optimal α, the optimal objective is around -200, meaning that with an offset of 4, the average travel time of all controllable agents is 200 seconds.

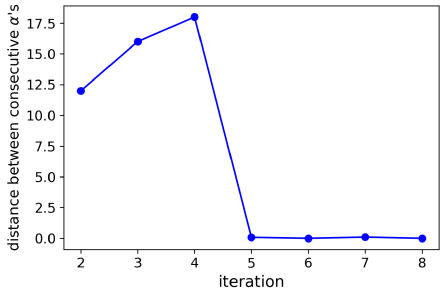


Fig. 9: Convergence of BO

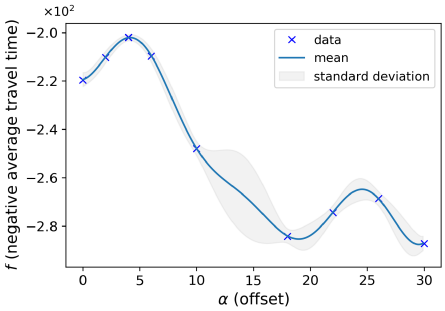


Fig. 10: Posterior probability distribution of f at the 8^{th} iteration

To provide more insights, we decompose the average travel time of all controllable agents into two components, namely average waiting time (at intersections) and average cruising time, presented in Figure (11). In general, a smaller α (e.g., $\alpha < 10$) yields a smaller average waiting time and a smaller average cruising time, while a larger α results in a higher average waiting time and a higher cruising time. This could be partially explained as follows. With a smaller α, vehicles could take advantage of the "green wave" and enjoy a smaller waiting time and better traffic condition. The lowest waiting time and the lowest cruising time are achieved when the offset is set as 4 seconds, which is exactly the previously derived optimal α^*.

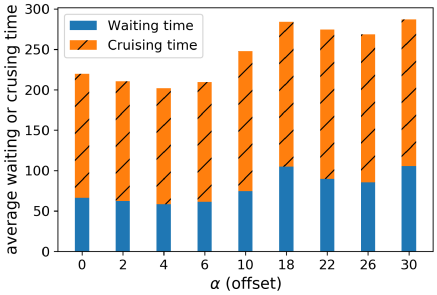


Fig. 11: Decomposition of average travel time

V. Conclusion

This paper develops a bilevel optimization with the lower level as MARL and the upper level solved by BO. We demonstrate the effect of two countermeasures, namely tolling (see Section III) and signal control (see Section IV), on the behavior of travelers and show that the systematic objective of the planner can be optimized by a proper control.

In the future, we would like to explore the following directions: (1) The existence and uniqueness of an equilibrium for dynamic routing games is not provided in this paper, due to its complicated structure. This is a challenging problem at the intersection of reinforcement learning and game theory. (2) Learning may lead to many implausible Nash equilibria if one's belief about opponents' play or information is inaccurate. Under what conditions learning leads to a desired Nash equilibrium needs to be investigated.

Acknowledgements

This work is partially sponsored by the National Science Foundation under CAREER award number CMMI-1943998.

References

[1] Z. Shou, X. Chen, Y. Fu, and X. Di, "Multi-agent reinforcement learning for markov routing games: A new modeling paradigm for dynamic traffic assignment," *Transportation Research Part C: Emerging Technologies*, p. 103560, 2022.

[2] H. Yang and M. G. H. Bell, "Models and algorithms for road network design: a review and some new developments," *Transport Reviews*, vol. 18, no. 3, pp. 257–278, 1998.

[3] X. Di, H. X. Liu, and X. J. Ban, "Second best toll pricing within the framework of bounded rationality," *Transportation Research Part B*, vol. 83, pp. 74–90, 2016.

[4] X. Di, R. Ma, H. X. Liu, and X. J. Ban, "A link-node reformulation of ridesharing user equilibrium with network design," *Transportation Research Part B: Methodological*, vol. 112, pp. 230–255, 2018.

[5] X. Chen and X. Di, "Ridesharing user equilibrium with nodal matching cost and its implications for congestion tolling and platform pricing," *Transportation Research Part C: Emerging Technologies*, vol. 129, p. 103233, 2021.

[6] M. W. Levin and S. D. Boyles, "Intersection auctions and reservation-based control in dynamic traffic assignment," *Transportation Research Record*, vol. 2497, pp. 35 – 44, 2015.

[7] K. Han, Y. Sun, H. Liu, T. L. Friesz, and T. Yao, "A bi-level model of dynamic traffic signal control with continuum approximation," *Transportation Research Part C: Emerging Technologies*, vol. 55, pp. 409–431, 2015.

[8] R. Ma, X. J. Ban, and W. Szeto, "Emission modeling and pricing on single-destination dynamic traffic networks," *Transportation Research Part B: Methodological*, vol. 100, pp. 255–283, 2017.

[9] A. L. C. Bazzan and R. Grunitzki, "A multiagent reinforcement learning approach to en-route trip building," in *2016 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2016, pp. 5288–5295, iSSN: 2161-4407.

[10] Z. Shou and X. Di, "Reward design for driver repositioning using multi-agent reinforcement learning," *Transportation Research Part C*, vol. 119, no. 102738, 2020.

[11] Y. Yang, R. Luo, M. Li, M. Zhou, W. Zhang, and J. Wang, "Mean Field Multi-Agent Reinforcement Learning," in *International Conference on Machine Learning*, Jul. 2018, pp. 5571–5580.

[12] X. Di, H. X. Liu, J.-S. Pang, and X. J. Ban, "Boundedly rational user equilibria (BRUE): Mathematical formulation and solution sets," *Transportation Research Part B: Methodological*, vol. 57, pp. 300–313, Nov. 2013.

[13] P. I. Frazier, "A Tutorial on Bayesian Optimization," *arXiv:1807.02811 [cs, math, stat]*, Jul. 2018, arXiv: 1807.02811.

[14] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: no regret and experimental design," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ser. ICML'10. Haifa, Israel: Omnipress, Jun. 2010, pp. 1015–1022.