Received: 20 April 2024 | Accepted: 27 September 2024

DOI: 10.1111/mice.13354

COMPUTER-AIDED CIVIL AND INFRASTRUCTURE ENGINEERING WILEY

RESEARCH ARTICLE

An adversarial diverse deep ensemble approach for surrogate-based traffic signal optimization

Zhixian Tang[1] | **Ruoheng Wang**[1] | **Edward Chung**[1] | **Weihua Gu**[1] | **Hong Zhu**[2]

[1]Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University, Hong Kong, China

[2]College of Transportation Engineering, Tongji University, Shanghai, China

Correspondence
Edward Chung, Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University, Hong Kong, China.
Email: edward.cs.chung@polyu.edu.hk

Funding information
Hong Kong Research Grants Council – General Research Fund, Grant/Award Number: 15224423; Innovation and Technology Commission - Mainland-Hong Kong Joint Funding Scheme, Grant/Award Number: MHP/038/23; National Key Research Program, Grant/Award Number: 2023YFE0209300

Abstract

Surrogate-based traffic signal optimization (TSO) is a computationally efficient alternative to simulation-based TSO. By replacing the simulation-based objective function, a surrogate model can quickly identify solutions by searching for extreme points on its response surface. As a popular surrogate model, the ensemble of multiple diverse deep learning models can approximate complicated systems with a strong generalizability. However, existing ensemble methods barely focus on strengthening the prediction of extreme points, which we found can be realized by further diversifying base learners in the ensemble. The study proposes an adversarial diverse ensemble (ADE) method for online TSO with limited computational resources, comprising two stages: In the offline stage, base extractors are diversified with unlabeled data by a designed adversarial diversity training algorithm; in the online stage, base predictors are trained in parallel with limited labeled data, and the ensemble then serves as the surrogate model to search for solutions iteratively for TSO. First, it is demonstrated that the prediction accuracy on extreme points, and associated solution quality, can be constantly improved with base learners' diversity enhanced by ADE. Case studies of TSO conducted on a four-intersection arterial further demonstrate the superior solution quality and computational efficiency of the ADE surrogate model in a wide range of traffic scenarios. Moreover, a large-scale online TSO experiment under dynamic traffic demand proves ADE's effectiveness in practical applications.

1 | INTRODUCTION

1.1 | Background

Traffic on surface streets crosses at intersections and flows on the network. Traffic signals are widely used to allocate time resources to different movements at the intersections toward smooth and efficient traffic flow. Therefore, traffic signal optimization (TSO) is critical to enhancing the efficiency of the urban traffic network.

In TSO, it is of vital significance that the objective function can accurately evaluate real-world signal performance. Microscopic traffic simulators replicate the traffic by emulating the individual vehicles' behaviors and thus

provide accurate performance estimates for signal timing. Using microscopic traffic simulation outputs as the objective function, studies have been focusing on a simulation-based optimization approach for the TSO problem (Baldi et al., 2017; P. Li et al., 2010; Stevanovic et al., 2007).

The traffic demand is dynamic, especially in the morning and evening peaks. To cater to real-time traffic demand, the simulation-based optimization starts over once the real-time demand data are captured. Therefore, all the simulations are conducted online. Since traffic demand is dynamic over time, a timely return of optimal signal timing is required. However, simulation is time-consuming because of the behaviors of numerous individual vehicles and their interactions in the network. Meanwhile, the search for the optimal solution involves a great number of simulations. Consequently, the expensive-to-evaluate nature impedes its applications in TSO under the tight time budget. To address this issue, researchers are focusing on adopting a surrogate model to approximate and then replace the simulation-based objective function. Surrogate models are usually constructed using simple mathematical structures, rendering better time efficiency over the underlying simulation-based function. Therefore, the extreme point (i.e., the optimal solution) of a surrogate model can be quickly identified. The extreme point is the surrogate models' belief in a good solution of the simulation-based function. Thus, the extreme point is then evaluated by simulation to verify its solution quality and update the surrogate model. Examples of applications of surrogate modeling in transportation system optimizations can be seen in optimal pricing (Xiqun Chen et al., 2014; Osorio & Atasoy, 2021), optimal dedicated lane allocation (Z. Li et al., 2022), network design (Xiqun Chen et al., 2015), model calibration (Cheng et al., 2019), and TSO (Xiao Chen et al., 2019; Ito et al., 2019; Leandro & Luque, 2023; Liang et al., 2021; Osorio & Bierlaire, 2013; Osorio & Chong, 2015; Zheng & Li, 2023; Zheng et al., 2019).

In recent years, deep learning models, a powerful tool for modeling complex systems, have been serving as popular surrogate models for prediction tasks in transportation systems because of their capabilities of coping with high nonlinearity and dimensions in transportation systems, for example, Dharia and Adeli (2003), Jiang and Adeli (2005), K. Tang et al. (2021), Zou and Chung (2024), and Zou et al. (2024). Refer to Y. Wang et al. (2019) for a review. Deep learning models can learn to generalize the characteristics of the simulation-based objective function using a limited number of evaluated samples, to form a response surface that approximates the landscape of the simulation-based objective function. Meanwhile, the heavy request for simulation output during optimization is replaced by efficient prediction of deep learning models. In this way, optimization can be accelerated by avoiding costly simulations.

Meanwhile, recent studies applying reinforcement learning (RL) realized noticeable performance in TSO problems. RL methods have extended the TSO to thousands of road networks (C. Chen et al., 2020), enabled more flexible traffic signal strategies (Long et al., 2022), and been applied in varied data environments (R. Zhang et al., 2021). Refer to Wei et al. (2021) and Noaeen et al. (2022) for a detailed survey. Despite these advancements, the practical deployment of RL in TSO is still limited by its challenges. Leveraging an offline training and online execution paradigm, RL models face potential issues with out-of-distribution traffic scenarios. They may fail to adapt to unseen traffic conditions in real time, which can lead to detrimental outcomes in certain situations (Gu et al., 2022). In contrast, the surrogate-based approach is a plug-and-play tool for TSOs. It adapts to real-time traffic scenarios by iterative online refinement of solution. Therefore, it can be safer and more robust to varied traffic scenarios. It can also compensate for deep RL approaches as a safety module by providing a robust candidate solution.

1.2 | Research objectives, challenges, and contributions

Surrogate models are essentially prediction models. The exceptional performance in prediction tasks highlights their significant potential for surrogate-based optimizations. Compared to traditional surrogate models, this potential becomes even more promising in future infrastructure environments featuring massive historical data from various sources and types. Therefore, this study focuses on deep learning surrogate-based online TSO. However, this approach presents two main challenges.

The first challenge is that the time and simulation budgets are very limited due to the online nature of this problem. Online TSO requires timely update of signal timing to cater to the latest traffic demand. As a result, the tight time budget restricts the number of online simulation runs. Nevertheless, with limited training data, a single deep learning model may not generalize well in unseen areas. At the end of the training, they can converge to any hypothesis of the simulation-based objective function in unseen areas. All of those hypotheses can fit the training data perfectly yet bring huge errors in unseen areas (Sagi & Rokach, 2018). An ensemble of multiple base learners may have better generalizability over a single learner because the ensemble tends to cancel out the errors to make more accurate predictions. This mechanism works better when base learners converge to diverse hypotheses and produce

diverse errors, instead of making similar mistakes. Therefore, it is widely accepted that the prediction performance of an ensemble jointly depends on the prediction performance and diversity of its base learners (Dietterich, 2000; T. Zhou et al., 2018). Nevertheless, the additional training for ensemble diversity can take more time than regular ensembles.

The second challenge is that deep learning models, even an ensemble of them, cannot predict well on their extreme points, leading to poor solution quality and reduced optimization efficiency. For decades, the community of ensemble learning has been focusing on overall prediction performance in the entire sample space. On the contrary, the surrogate-based optimization derives the solution by searching for the extreme points on the response surface of a surrogate model, therefore emphasizing the accurate prediction of extreme points. However, the performance on the extreme points may not be consistent with overall prediction performance. Experiments on deep learning surrogate-based TSO (Gora et al., 2018) have shown that, with nearly 10,000 training data points, despite an error of merely 2% on a test set of random samples, the error can reach 12.9% on a test set of extreme points. In reality, the limited training data resulting from a tight time budget can further exacerbate the ensemble performance on extreme points. The poor solution quality of the extreme points will lead to the misidentification of promising solutions, thus disrupting the optimization. However, this problem is rarely investigated by existing studies.

Upon delving deeper into this issue, since an optimization algorithm proactively searches for the extreme points in the entire sample space, the performance on extreme points can be improved only when the non-promising extreme points are eliminated on the entire response surface. In this study, it is found that this can be realized by further boosting ensemble diversity.

The main research question is to develop a deep ensemble model that addresses these two challenges and thereby enabling it to efficiently solve surrogate-based online TSO problems.

In this study, an adversarial diverse ensemble (ADE) is proposed. In ADE, each base learner consists of an extractor extracting diverse features from an input solution to TSO and a predictor predicting the objective value of the solution based on the diversified features. The tight time budget limits the online training time. Thus, a two-stage training framework comprising offline diversity pre-training and online predictor training is introduced as follows: The first stage aims to constantly diversify the extractors using a proposed adversarial diversity training algorithm. This stage is trained offline with unlabeled data. In the second stage, the parameters of these diversified extractors are frozen. The input of training data is converted to diverse features by the diversified extractors. The predictor of each base learner is then trained to fit the features and true labels, with a time budget similar to that of ordinary ensemble learning.

Using ADE as the surrogate model in surrogate-based TSO, the improved prediction performance on extreme points can enhance the capability of identifying promising solutions, compared with ordinary ensembles. Consequently, the convergence of the optimization can be accelerated under a tight simulation budget. In addition, the superior time efficiency renders faster model training and thus saves the time budget. A timely return of signal timing enables quick responses to dynamic traffic demand in online TSO.

This paper contributes to the literature in three aspects.

1. Ensemble diversity training. It introduces a two-stage framework that separates diversity training and predictor training. A novel adversarial diversity training algorithm is developed to boost diversity without any labeled data.
2. Prediction on extreme points. This study is the first to observe and demonstrate that prediction performance of deep ensembles on extreme points can be enhanced by achieving a substantial level of ensemble diversity.
3. Online surrogate-based TSO efficiency. The improved prediction performance on extreme points accelerates the identification of promising solutions, thereby enabling more efficient online TSO.

The rest of the paper is organized as follows. The next section provides a review of surrogate modeling, surrogate-based TSO, and ensemble learning. In the third section, the underlying simulation-based TSO problem is given, and the framework of surrogate-based optimization is introduced. The fourth section provides the details of the proposed ADE. The fifth section validates the prediction performances of ADE in a traffic performance prediction problem. The performance of ADE in the TSO problem under static traffic demand is examined in the sixth section to show and explain its advantages as a surrogate model. It is followed by an online TSO with dynamic traffic demand in a large-scale network in the seventh section. Finally, conclusions and future works are presented in the eighth section.

2 | LITERATURE REVIEW

This section provides a literature review of related studies. First, a brief survey of surrogate modeling is provided. Then, the research efforts on surrogate-based TSO are

presented. To form a position of the proposed ADE method, the fundamentals and recent trends in ensemble learning are also introduced.

2.1 | Surrogate modeling

Surrogate modeling aims to leverage cheap-to-evaluate models to approximate simulation-based functions, such that the heavy simulations can be replaced by surrogate models' predictions. In past decades, some models are widely used to build surrogate models, including polynomial functions, Gaussian process (GP, also called Kriging), radial basis functions (RBFs), support vector machine (SVM; Forrester & Keane, 2009), and deep learning models. In general, surrogate models are mainly applied to three types of tasks, prediction, sensitivity/uncertainty analysis, and optimization (Kudela & Matousek, 2022). Different tasks ask for different capabilities of the surrogate models.

The difficulties of prediction tasks mainly lie in the high-dimensional (time-series, image, etc.) input with varied feature relevance and different data distributions in training set and test set. Deep learning models have presented overwhelming advantages in various prediction tasks, such as disaster forecasting (Zahura et al., 2020), which has to be simulated for a long time otherwise.

Predictions may not be the end of their applications. Sensitivity analysis is one of their downstream tasks. It is particularly crucial when system reliability is dependent on the random factors of the input variables (Sudret & Der Kiureghian, 2000). To address this, a polynomial chaos expansion (PCE) model was proposed to build surrogate models that approximate the sensitivity of expensive simulation models (Sudret, 2008). A sparse PCE model was later developed to reduce the computational burden when the dimensionality increased to 20 (Blatman & Sudret, 2010). Additionally, Kriging models are widely used in reliability analysis due to their ability to provide analytical estimates of prediction error (Gaspar et al., 2017; Sundar & Shields Michael, 2019).

Optimization is another type of downstream task. Surrogate-based optimization has been adopted across various disciplines, such as protein design (Borkowski et al., 2020), structural design (Bisbo & Hammer, 2020), and aerodynamic optimization (Tao & Sun, 2019). On the other hand, the research community has been investigating advanced surrogate modeling techniques. Notable examples include a polynomial-chaos-based Kriging model that approximates global trends using PCE and captures local variance with GP (Schobi et al., 2015) and a parallel Bayesian optimization method leveraging approximations from multiple GPs (J. Wang et al., 2020).

In optimization problems, it is crucial for the extreme points of surrogate models to accurately fit the true function. With limited data, a single surrogate model may not generalize well in its extreme areas. Recent advances have focused on using an ensemble of multiple base surrogate models to alleviate this problem. These methods enhance diversity among base models by employing different model structures, training data bagging (J. Y. Li et al., 2021), and weight allocation (Goel et al., 2007). However, these traditional methods do not always guarantee sufficient diversity. It is still likely that the base models make the same mistakes in the same areas.

2.2 | Surrogate-based TSO

For decades, the research community has been focusing on analytical modeling of traffic dynamics, using approaches such as shock wave theory (Wu & Liu, 2011), queueing theory (Osorio & Bierlaire, 2009), and cell transmission model (Pohlmann & Friedrich, 2010) for signal optimization. In those models, traffic dynamics is usually simplified, especially when the real traffic does not flow in a well-arranged manner, such as at intersections with turning bays and lane-drop areas. By simulating the vehicles individually, microscopic traffic simulation can provide an accurate evaluation of signal performance that closely reflects real-world conditions. Thus, it can be an ideal objective function for TSO. However, simulation-based objective functions are expensive to evaluate. To relieve the computational burden, research communities have been focusing on surrogate modeling that approximates the simulation-based objective function. There are mainly two streams of surrogate-based TSO. The first stream combines the surrogate model and analytical model in the objective function. The second stream relies on using the surrogate model as the sole objective function.

Both surrogate and analytical models are inexpensive to evaluate, and their respective advantages are taken into account in the studies of the first stream. Analytical models are established with physical components of the underlying traffic system. Therefore, they can provide a good global approximation. In contrast, fitting the data samples from microscopic simulation, surrogate models are expected to predict more accurately in specific local areas that are densely sampled (Osorio & Bierlaire, 2013). The objective function is formulated as a linear combination of the two models, the preference of which can be adjusted to cater to the needs of global and local search. Originating from the pioneering work of Osorio and Bierlaire (2013), this method has been applied to TSO problems with various scenarios and objectives, including signal control in large-scale networks (Osorio & Chong, 2015), dynamic

signal control (Chong & Osorio, 2017), travel time-robust signal control (Xiao Chen et al., 2019), and energy-efficient signal control (Osorio & Nanduri, 2015). In this stream of studies, the surrogate model is usually a simple quadratic polynomial to maintain analytical tractability. As a cost of using such a simple structure, the surrogate model can only fit well in the local area out of its limited fitting ability. Furthermore, the simple structure has difficulty handling discrete variables, such as signal phase sequences, and highly nonlinear relations, such as offset. Consequently, the above studies only optimize green splits, which are relatively linear to signal performances.

On the contrary, the surrogate models used in the second stream are generic functions with superior fitting ability (Osorio & Chong, 2015). For example, an RBF model and a GP model (also called Kriging model) were applied to serve as the surrogate models in TSO (Leandro & Luque, 2023). A GP model was incorporated into Bayesian optimization to solve arterial signal coordination problems (Ito et al., 2019). Liang et al. (2021) proposed a surrogate-assisted cooperative TSO for large-scale networks. In this work, the large network was first decomposed into sub-networks, and then an RBF model was developed to approximate the signal performance function in the sub-networks. RBF model and GP model were also used to tackle multi-objective TSO considering stochasticity in simulations (Zheng & Li, 2023; Zheng et al., 2019). Tay and Osorio (2022) proposed a GP model to approximate the whole large-scale network globally. The GP model was enhanced by problem-specific information from analytical models to obtain prior beliefs before any simulation was conducted. Gong et al. tackled the same problem with a trust region Bayesian optimization method using local GP models (Gong et al., 2024). To defend cyberattacks on signal systems, a robust GP surrogate-based TSO was proposed to realize an equilibrium in the attack-defense game (Bao et al., 2023; Zheng et al., 2023).

However, traditional surrogate models face certain shortcomings. First, their potential for scalability to large historical datasets and high-dimensional inputs is limited (Liu et al., 2020). Second, it is still an open problem to model discontinuous functions using traditional surrogate models (Boursier Niutta et al., 2018). These aspects hinder the applications of surrogate models in complicated TSO systems.

In recent years, deep learning models have been overwhelmingly popular in surrogate modeling due to their paramount abilities to cope with high dimensions, nonlinearity, and discontinuity. However, in a study of deep learning prediction for signal performance, it was reported that the prediction accuracy on the extreme points of deep learning models was far behind their overall performance (Gora et al., 2018). Due to the great prediction error, the extreme points of the deep learning surrogate models will turn out to be non-promising solutions, and a limited simulation budget will be wasted in validating those solutions. Although an attempt has demonstrated that with extensive training data, deep learning models can satisfactorily facilitate TSO (Gora et al., 2019), it is still not applicable in practice with limited training data. Therefore, despite the predominant applications of deep learning models in traffic prediction tasks, the issue of poor prediction on extreme points has been impeding their applications in traffic optimization problems.

2.3 | Diverse ensemble learning

An ensemble of multiple deep learning models (called base learners in the context of ensemble learning) can improve generalization. The vital reason is that the diversity of base learners drives them to make different mistakes for the same task. The mistakes tend to be canceled out by averaging the predictions. Therefore, this review introduces the studies focusing on diversifying ensembles.

Randomization is an intuitive and classic method to encourage ensemble diversity. It can be incorporated into different components of ensemble training. For example, bagging (Breiman, 1996) and random forests (RFs; Breiman, 2001) allocate random subsets of training data for different base learners. Random weight initialization and learning rate scheduling (Huang et al., 2017) tend to drive base learners to converge to different areas in the parameter space, which has been comprehensively demonstrated by Fort et al. (2019).

Randomization methods only trigger diversity at the initial stage of ensemble training, without continuous guidance in the following training process. Instead of betting on randomization, studies have been proposed to boost ensemble diversity proactively with diversity-encouraging loss functions. The selective approaches aim to select a subset of base learners or training data from a pool of them to enhance diversity. For example, a genetic algorithm based selective ensemble (GASEN) algorithm was developed to select a subset from a pool of base learners, with an objective function of the minimal correlation among them (Z. -H. Zhou et al., 2002). A DivE^2 model was introduced to create subsets of training data with specialized and complementary expertise (T. Zhou et al., 2018). A loss function consisting of inter-model and intra-model diversity guided the data assignment problem. Other approaches continuously train the existing base learners towards diversity. For example, for regression tasks, a genetic algorithm was used to evolve the weights of base learners to balance accuracy and diversity through multi-objective optimization (Opitz & Shavlik, 1995). To encourage diversity in classification

tasks, Pang et al. (2019) proposed diversifying the non-maximal predictions of the classes while maintaining the predicted class with maximal probability being the true class. It was also proposed that the features could be diversified to encourage the diversity of the base predictions (S. Zhang et al., 2020).

However, only a few studies focused on diversifying ensembles with limited training data. An intuitive strategy is to use limited labeled training data to pursue prediction accuracy and boost diversity through pseudo or unlabeled data. For example, an oppositional relabeling method was proposed to construct diverse datasets, driving the base learners to diverse predictions in unseen areas (Melville and Mooney, 2005). Ensemble diversity could also be promoted by differentiating the predictions of unlabeled data (M. -L. Zhang and Zhou, 2013). In these methods, the accuracy and diversity of the base learners are trained simultaneously. However, introducing pseudo or unlabeled data will inevitably increase the training time compared to ordinary ensembles, such as those using randomization methods.

While only a few ensemble studies have been proposed for limited training data scenarios, even fewer are further specifically designed for online training under a tight time budget. In this study, an ADE method is proposed to tackle the issue of limited time by separating the task of diversity and predictor training. An adversarial diversity training algorithm is developed to boost diversity offline with unlabeled data.

Some studies, such as selective ensemble, also separate the two tasks, yet those studies do not emphasize that the training of predictors follows the training of diversity, which therefore is still conducted online. This study distinguishes itself by addressing this particular aspect.

3 | PRELIMINARIES

In this section, first, the simulation-based online TSO problem is elaborated on. Then, a surrogate-based optimization framework is introduced to solve the problem.

3.1 | TSO problem

Online TSO with dynamic demand can be decomposed into fixed-time TSOs in small time windows. Therefore, fixed-time TSO serves as the basic unit for online dynamic TSO. The constrained simulation-based fixed-time TSO problem is formulated by Equations (1a) to (1g).

$$\min\ y = f_{\text{sim}}\ (c, \boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \dots, \boldsymbol{\delta}_I) \tag{1a}$$

Subject to: $\forall i \in \mathbb{N}_I = \{1, \dots, I\}$,

$$c_{\min} \le c \le c_{\max} \tag{1b}$$

$$\boldsymbol{\delta}_i = (\boldsymbol{g}_i, s_i, \theta_i) \tag{1c}$$

$$\boldsymbol{g}_i^{\min} \le \boldsymbol{g}_i \le c \tag{1d}$$

$$\|\boldsymbol{g}_i\| = c \tag{1e}$$

$$s_i \in S_i \tag{1f}$$

$$0 \le \theta_i \le c;\ \theta_1 = 0 \tag{1g}$$

In this optimization problem, traffic demand is assumed to be fixed, which is a common practice in simulation-based TSO (Osorio & Bierlaire, 2013; Tay & Osorio, 2022). In dynamic applications, the optimization starts over with the latest traffic demand. The objective shown in Equation (1a) is to minimize the traffic performance index y. The simulation-based objective function f_{sim} calls a simulation run for the evaluation of signal timing $(c, \boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \dots, \boldsymbol{\delta}_I)$. c is the variable of common cycle length. The range of cycle length is restricted by Equation (1b). $\boldsymbol{\delta}_i$ stands for the signal timing at intersection i. I is the number of signalized intersections. For each intersection, the green durations $\boldsymbol{g}_i$, phase sequence s_i, and offset θ_i are optimized as Equation (1c) indicates. The unit of signal timing is second. The constraints of the green durations are given in Equations (1d) and (1e). Amber and all-red times are included in $\boldsymbol{g}_i$. Phase sequence s_i is required to be in a set of feasible sequences S_i, as Equation (1f) indicates. s_i can be defined as either an integer variable (such as, for three-phase cases, its value can be 0 or 1) or a sequence (its value can be $\{0,\ 1,\ 2\}$ or $\{0,\ 2,\ 1\}$). In this study, s_i is defined as a sequence, but it is still regarded as a one-dimensional decision variable. Finally, Equation (1g) restricts the range of the offsets. The offset of the first intersection is set to be the datum value, zero.

Constraints (1d) to (1f) reflect a typical stage-based modeling of signal timing (Webster, 1958). The stage-based approach splits the cycle into several phases. Only compatible movements can be grouped into the same phase, ensuring the safety of the movements. All the possible permutations of the phases form the feasible set of phase sequences.

In this model, the dimension of decision variables is $d = 2I + \sum_i |\boldsymbol{g}_i|$. 1 for cycle length, $I-1$ for offsets, I for sequences, and $\sum_i |\boldsymbol{g}_i|$ for phase durations. For notational convenience, in the following contents, the variables $(c, \boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \dots, \boldsymbol{\delta}_I)$ are replaced by $\boldsymbol{x} \in \mathbb{R}^d$ and the feasible

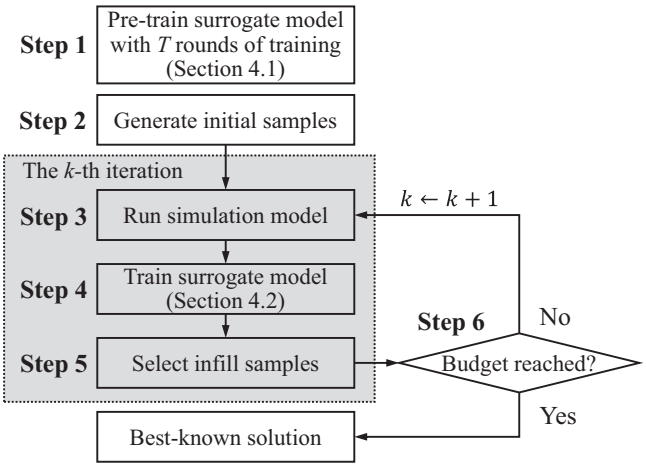


FIGURE 1 Framework of surrogate-based optimization.

region defined in Equations (1b) to (1g) is denoted by Ω. The optimal solution $\boldsymbol{x}^*$ of the problem in Equations (1a) to (1g) is defined in Equation (1h).

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}\in\Omega} f_{\text{sim}}(\boldsymbol{x}) \tag{1h}$$

The high dimension of decision variables and expensive-to-evaluate objective function bring computational challenges to this optimization problem. In the following section, a surrogate-based optimization framework is introduced to realize computationally efficient optimization.

3.2 | Framework of surrogate-based optimization

The framework of surrogate model-based optimization is presented in Figure 1.

In this study, an ADE model is proposed as the surrogate model. It requires an offline pre-training stage with T rounds of training (see Section 4.1) before it can be trained by the simulation data. Therefore, Step 1 is to pre-train the surrogate model.

In Step 2, the initial samples of signal timing, whose set is denoted as X_{init}, are selected in the feasible region Ω. Each sample is a vector of all decision variables specified in Equation (1a). To obtain an initial knowledge of the landscape of f_{sim}, the samples should be as evenly distributed as possible. An optimization-based method of design of experiment, maximin Latin hypercube sampling (Maximin LHS; Morris & Mitchell, 1995), is adopted to generate the initial samples.

Step 3 runs traffic simulations to evaluate the received samples from the last step (either Step 2 or Step 6) to obtain their true labels. All the samples evaluated at or before the kth iteration form a data pool $(X^k_{\text{pool}}, Y^k_{\text{pool}})$, which is defined in Equations (2a) and (2b).

$$X^k_{\text{pool}} = X_{\text{init}} \cup \left\{X^l_{\text{ext}}\right\}_{l=1}^{k} \tag{2a}$$

$$Y^k_{\text{pool}} = Y_{\text{init}} \cup \left\{Y^l_{\text{ext}}\right\}_{l=1}^{k} \tag{2b}$$

where Y is the set of true labels of X (scripts omitted). X^k_{ext} is the set of extreme points at the kth iteration, which will be introduced below.

In Step 4, the surrogate model f_{SM^k} will be trained or re-trained by $(X^k_{\text{pool}}, Y^k_{\text{pool}})$. The proposed ADE will serve as the surrogate model. For details of ADE, please refer to Section 4.

Updated with the latest data pool, the surrogate model will subsequently adjust its beliefs regarding promising areas. Step 5 searches the extreme points of the latest surrogate model f_{SM^k} The extreme point $\boldsymbol{x}^{\boldsymbol{k}}_{\text{ext}}$ is defined in Equation (2c). In the next iteration, the extreme points are the infill samples that will be evaluated by simulation to verify their solution quality and update the surrogate model. In the following contents, the terms extreme points and infill samples are used interchangeably in different context.

$$\boldsymbol{x}^{\boldsymbol{k}}_{\text{ext}} = \operatorname{argmin}_{\boldsymbol{x}\in\Omega} f_{\text{SM}^k}(\boldsymbol{x}) \tag{2c}$$

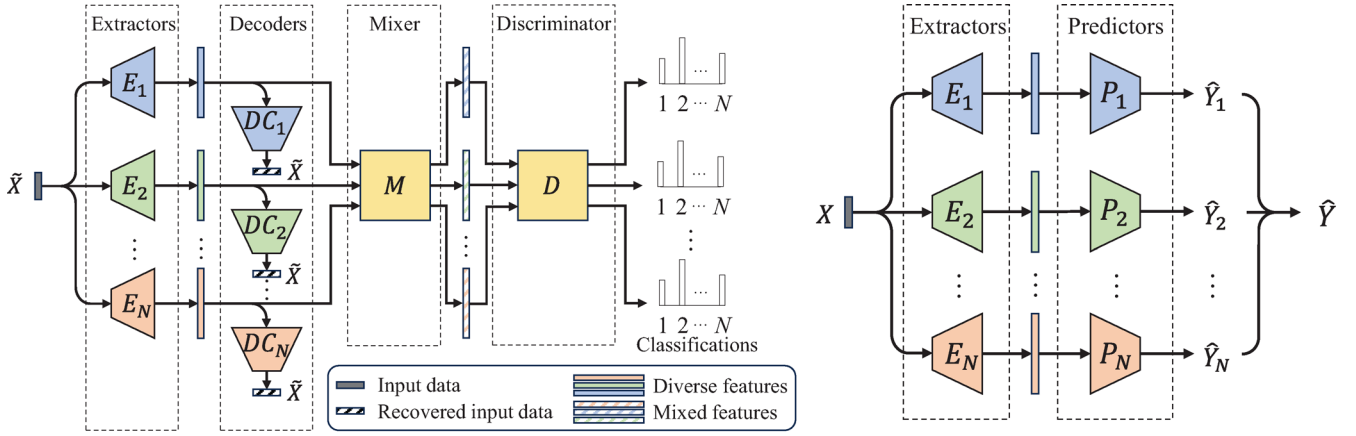
Differential evolution (DE; Das et al., 2016) serves as the optimizer for the search for extreme points in Equation (2c). Due to the stochastic nature of DE, the returned $\boldsymbol{x}^{\boldsymbol{k}}_{\text{ext}}$ may vary across multiple parallel search sessions, thus allowing for constructing a set of different infill samples, X^k_{ext}. Please refer to Appendix 1 for the detailed DE algorithm.

In Step 6, if the budget is reached, terminate and output the best-known solution in $(X^k_{\text{pool}}, Y^k_{\text{pool}})$; otherwise, go to Step 3 to evaluate the extreme points X^k_{ext} by simulation.

The surrogate-based optimization relies on a good solution quality of the extreme point $\boldsymbol{x}^{\boldsymbol{k}}_{\text{ext}}$. However, $\boldsymbol{x}^{\boldsymbol{k}}_{\text{ext}}$ may not necessarily be a promising solution to the original simulation-based objective function $f_{\text{sim}}(\boldsymbol{x})$. To tackle this issue, an ADE approach is proposed to improve the prediction performance on the extreme points and consequently to improve their solution quality.

4 | ADVERSARIAL DIVERSE ENSEMBLE

Figure 2 shows the network structure of the proposed adversarial diverse ensemble (ADE). In the inference phase (i.e., when implemented in predictions), the



(a) Offline training stage for diverse extractors

(b) Online training stage for predictors, and inference phase

FIGURE 2 Network structure of adversarial diverse ensemble (ADE).

ensemble consists of N base learners $\{E_j, P_j\}_{j=1}^N$, as Figure 2b illustrates. $\{E_j\}_{j=1}^N$ is a set of feature extractors that are trained to extract diverse features from the same input data. $\{P_j\}_{j=1}^N$ are the predictors receiving the features from their respective extractors and outputting predictions. In the forward calculation, for any given input $\boldsymbol{x}$, each E_j outputs a distinct feature $E_j(\boldsymbol{x})$. Then each predictor P_j receives $E_j(\boldsymbol{x})$ and outputs a base prediction $P_j(E_j(\boldsymbol{x}))$ as Equation (3) indicates. In this study, the final output of the ensemble is simply the average of base learners' predictions as Equation (3) shows.

$$\hat{y} = \frac{1}{N}\sum_{j=1}^{N}\hat{y}_j = \frac{1}{N}\sum_{j=1}^{N}P_j\left(E_j\left(\boldsymbol{x}\right)\right) \tag{3}$$

Simultaneous training of extractors and predictors is time-consuming for online tasks. Thus, a two-stage training is designed to separate the two training processes in a semi-supervised learning manner. The first stage is offline training of diverse extractors (i.e., Step 1 in Figure 1) with the assistance of decoders $\{DC_j\}_{j=1}^N$, a mixer M, and a discriminator D as Figure 2a shows. The second stage is online training of predictors illustrated in Figure 2b (i.e., Step 4 in Figure 1). The two stages are introduced below.

4.1 | Offline training of diverse extractors

Given a set of unlabeled input data $\tilde{X} = \{\tilde{\boldsymbol{x}}_{\boldsymbol{r}}\}_{r=1}^{\tilde{R}}$ randomly sampled in Ω, this stage aims to train diverse extractors $\{E_j\}_{j=1}^N$. It can be conducted offline because no data from simulation are required. The extractors are expected to generate features that are distinct from those produced by other extractors. Meanwhile, the generated features should retain all the information derived from the input data; otherwise, the predictors cannot learn an effective mapping from extracted features to labels in the second stage. The diversity is achieved through an adversarial diversity training algorithm, and information retention is ensured with the assistance of the decoders.

4.1.1 | Adversarial diversity training

In the ADE, as illustrated in Figure 2a, the extractors $\{E_j\}_{j=1}^N$ and the mixer M are the key players to interact with an unbiased judger, that is, a discriminator D. The mixer aims to assimilate the features extracted by different extractors, to hinder the discriminator from identifying which extractor a mixed feature originally comes from. On the contrary, the extractors aim to diversify features to defy the assimilation effect of the mixer, to help the discriminator identify the origin extractor of the mixed feature.

In the adversarial process, the increasing diversification capability of the extractors will enforce the mixer to strengthen its ability to confound the discriminator. Similarly, the enhanced mixer will also drive the extractors to generate increasingly distinct features, which can be more identifiable to the discriminator. Through multiple rounds of adversarial training, the diversification ability of the extractors will be progressively enhanced.

The forward calculation of the aforementioned process is elaborated on as follows. For any extractor E_j, the extracted feature $E_j(\tilde{\boldsymbol{x}})$ is fed into the mixer M and converted to a mixed feature $M(E_j(\tilde{\boldsymbol{x}}))$. The mixed feature is then fed into the discriminator D, which identifies which extractor generates the mixed feature $M(E_j(\tilde{\boldsymbol{x}}))$. The discriminator outputs a probability distribution $D(M(E_j(\tilde{\mathbf{x}}))) \in \mathbb{R}^N$. The ith element of it represents

the probability that the mixed feature $M(E_j(\tilde{\boldsymbol{x}}))$ comes from the ith extractor E_i.

In the following contents, let ϕ denote the weights of neural networks. The loss functions of adversarial training include diversification loss $\mathcal{L}_E^{\text{div}}(\phi_{E_j})$ and feature magnitude loss $\mathcal{L}_E^{\text{mag}}(\phi_{E_j})$ for extractors and mixing loss $\mathcal{L}_M(\phi_M)$ for the mixer.

1. Diversification loss:

$$\mathcal{L}_E^{\text{div}}\left(\phi_{E_j}\right) = \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\tilde{X}}\left[\mathcal{H}\left(D\left(M\left(E_j\left(\tilde{\boldsymbol{x}}\right)\right)\right), q_j\right)\right] \quad \forall j \in \{1, \ldots N\} \tag{4a}$$

$$q_j\ (i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases},\ i, j \in \{1, \ldots N\} \tag{4b}$$

q_j is a probability distribution representing the correct classification. $\mathcal{H}(q', q'')$ is the cross-entropy loss forcing a distribution q' to approach q''. This loss optimizes the weights of the extractor ϕ_{E_j} to diversify its output $E_j(\tilde{\boldsymbol{x}})$. When this loss converges well, the jth probability of the discriminator's output $D(M(E_j(\tilde{\boldsymbol{x}})))$ should be the largest to signify the correct identification.

2. Feature magnitude loss:

$$\mathcal{L}_E^{\text{mag}}\left(\phi_{E_j}\right) = \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\tilde{X}}\left[\frac{1}{|\tilde{\boldsymbol{x}}|}\|E_j\left(\tilde{\boldsymbol{x}}\right)\|_2^2\right] \quad \forall j \in \{1, \ldots N\} \tag{5}$$

$|\cdot|$ is the length of the vector and $\|\cdot\|_2^2$ stands for ℓ_2^2-norm. The purpose of this loss function is to restrict the magnitude of the diversified features. This term is used to counteract a tendency that the extractors generate distinguishable features solely by magnitude, which lacks meaningful diversity.

3. Mixing loss:

$$\mathcal{L}_M\ (\phi_M) = \frac{1}{t}\sum_{s=1}^{t}\mathcal{L}_M^s\left(\phi_M\right) \tag{6a}$$

$$\mathcal{L}_M^s\left(\phi_M\right) = \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\tilde{X}}\left[\frac{1}{N}\sum_{j=1}^{N}\mathcal{H}\left(D\left(M\left(E_j^s\left(\tilde{\boldsymbol{x}}\right)\right)\right), q\right)\right] \tag{6b}$$

$$q\ (i) = \frac{1}{N},\ \forall i \in \{1, \ldots N\} \tag{6c}$$

q is a target uniform distribution. The objective of $\mathcal{L}_M$ in Equation (6a) is to optimize the weights of the mixer ϕ_M to assimilate the diversified features. In this way, the discriminator is not able to make a judgment on the origin of a mixed feature $M(E_j(\tilde{\boldsymbol{x}}))$. It is realized by forcing the output distribution of the discriminator to be uniform as Equation (6b) indicates. E_j^s refers to the historical E_j at the sth round, and t stands for the current number of rounds. The mixer M has to assimilate the diversified features produced by all the historical extractors as Equation (6a) shows. In this way, the extractors in the next round cannot converge to any historical ones but will be trained to enhance diversity further.

It should be noted that, in this study, the discriminator is not designed to aid either the extractors or the mixer in achieving their respective objectives but serves as an impartial judge. Thus, the discriminator is not trained in any process but maintains its randomly initialized weights.

4.1.2 | Information retention

Adversarial diversity training facilitated by Equations (4) to (6) can help extractors generate diverse features. However, it is very likely that the extractors produce features with little or even no information retained from the input data. In this case, the predictors cannot learn to make sensible inferences using incomplete information. Therefore, a loss is introduced to retain information in the extracted features via an encoder–decoder learning paradigm. Taking an extractor E_j as an encoder, the decoder DC_j is introduced to recover the feature $E_j(\tilde{\boldsymbol{x}})$ back to the input data $\tilde{\boldsymbol{x}}$ by forcing $DC_j(E_j(\tilde{\boldsymbol{x}}))$ to approach $\tilde{\boldsymbol{x}}$. The loss function for information retention is described in Equation (7), where $\mathcal{G}$ is the mean squared error (MSE) loss.

$$\mathcal{L}_{E,DC}^{\text{info}}\left(\phi_{E_j}, \phi_{DC_j}\right) = \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\tilde{X}}\left[\mathcal{G}\left(DC_j\left(E_j\left(\tilde{\boldsymbol{x}}\right)\right), \tilde{\boldsymbol{x}}\right)\right] \quad \forall j \in \{1, \ldots N\} \tag{7}$$

4.1.3 | Strategy for coordinating multiple objectives

As introduced above, extractors and mixer are the two players in the adversarial diversity training. In a single round of adversarial training, one step is to train the extractors together with decoders by the loss functions of diversification $\mathcal{L}_E^{\text{div}}$, feature magnitude $\mathcal{L}_E^{\text{mag}}$, and information retention $\mathcal{L}_{E,DC}^{\text{info}}$. The other step is to train the mixer by misclassification loss function $\mathcal{L}_M$.

The training of extractors involves multiple objectives in a single objective optimization via a weighted-sum combination as Equation (8) shows.

$$\mathcal{L}_E = w_E^{\text{div}} \cdot \mathcal{L}_E^{\text{div}}\left(\phi_{E_j}\right) + w_E^{\text{mag}} \cdot \mathcal{L}_E^{\text{mag}}\left(\phi_{E_j}\right) + w_{E,DC}^{\text{info}} \cdot \mathcal{L}_{E,DC}^{\text{info}}\left(\phi_{E_j}, \phi_{DC_j}\right) \tag{8}$$

TABLE 1 Algorithm 1: Strategy for coordinating multiple objectives.

Global input: Threshold $[\alpha_E^{\text{mag}}, \alpha_{E,DC}^{\text{info}}]$. Basic weight $[\overline{w}_E^{\text{div}}, \overline{w}_E^{\text{mag}}, \overline{w}_{E,DC}^{\text{info}}]$. Decay function $f_d(\mathcal{L}; \alpha)$.
Local input: Loss function values $[\mathcal{L}_E^{\text{div}}, \mathcal{L}_E^{\text{mag}}, \mathcal{L}_{E,DC}^{\text{info}}]$ in last epoch.
For * in $[(\cdot)_E^{\text{mag}}, (\cdot)_{E,DC}^{\text{info}}]$:
  If $\mathcal{L}^* \geq \alpha^*$:
    $w^* = \overline{w}^*$
  Else:
    $w^* = \overline{w}^* \cdot f_d(\mathcal{L}^*; \alpha^*)$
  End If
End For
$w_E^{\text{div}} = \overline{w}_E^{\text{div}}$.
Output: $\mathcal{S} = \{w_E^{\text{div}}, w_E^{\text{mag}}, w_{E,DC}^{\text{info}}\}$.

The major objective $\mathcal{L}_E^{\text{div}}$ conflicts with $\mathcal{L}_E^{\text{mag}}$ and $\mathcal{L}_{E,DC}^{\text{info}}$, which requires a strategy to coordinate these objectives. Specifically, a weight-tuning strategy is designed in Table 1. In this strategy, the loss $\mathcal{L}_E^{\text{mag}}$ and $\mathcal{L}_{E,DC}^{\text{info}}$ will be maintained around their targets. It is realized by assigning a large initial weight to them for fast convergence and applying a decay function when they are below the targets. Once the targets are met, the optimization will focus more on diversification loss $\mathcal{L}_E^{\text{div}}$. The targets are set because extremely small $\mathcal{L}_E^{\text{mag}}$ and $\mathcal{L}_{E,DC}^{\text{info}}$ will restrict the representation ability of the extractors to produce more diverse features.

4.2 | Online training of predictors

In the training process above, the extractors are trained to produce diverse features. The extractors will be frozen in the second training stage.

The second stage trains the predictors to approximate the underlying simulation-based objective function via a small amount of online collected data (X, Y). $X = \{\boldsymbol{x}_r\}_{r=1}^R$, $R \ll \tilde{R}$. $Y = \{y_r\}_{r=1}^R$ is obtained through simulations. In the forward pass, the input data are converted to diverse features by the extractors. The features are then fed into the predictors to make predictions. It can be noticed that the training of each predictor is independent, enabling parallel training. The loss function is as Equation (9) shows.

$$\mathcal{L}_P(\boldsymbol{\phi}_{P_j}) = \mathbb{E}_{(\boldsymbol{x},y)\sim(X,Y)}\left[\mathcal{G}\left(P_j\left(E_j(\boldsymbol{x})\right), y\right)\right] \quad \forall j \in \{1, \ldots N\} \tag{9}$$

4.3 | Pseudo-code of ADE training

The detailed training process of ADE is summarized in Table 2. Algorithm 2.1 is for the pre-training of diversified extractors with unlabeled data. In one round of adversarial training, the training of extractors and mixer is implemented alternately. Algorithm 2.2 is for the parallel training of predictors with labeled data.

TABLE 2 Adversarial diverse ensemble (ADE) training.

Algorithm 2.1: Offline training for diverse extractors
Input: The number of base learners N. The number of adversarial training rounds T. Training data $\hat{X}$. Learning rate η. Maximal epochs $e_E^{\max}$, $e_M^{\max}$. The global input for Algorithm 1.
Initiate: $\boldsymbol{\phi}_{E_j}$, $\boldsymbol{\phi}_{DC_j}$, $\boldsymbol{\phi}_M$, $\boldsymbol{\phi}_D$.
For $t = 1$ to T:
  Initiate: $\mathcal{S} = [w_E^{\text{div}}, w_E^{\text{mag}}, w_{E,DC}^{\text{info}}] \leftarrow [\overline{w}_E^{\text{div}}, \overline{w}_E^{\text{mag}}, \overline{w}_{E,DC}^{\text{info}}]$.
  For $j = 1$ to N:
    For $e = 1$ to $e_E^{\max}$:
      $\boldsymbol{\phi}_{E_j} \leftarrow \boldsymbol{\phi}_{E_j} - \eta \cdot \nabla \mathcal{L}_E$.
      $\boldsymbol{\phi}_{DC_j} \leftarrow \boldsymbol{\phi}_{DC_j} - \eta \cdot w_{E,DC}^{\text{info}} \cdot \nabla \mathcal{L}_{E,DC}^{\text{info}}(\boldsymbol{\phi}_{E_j}, \boldsymbol{\phi}_{DC_j})$.
      Update $\mathcal{S}$ by Algorithm 1.
    End For
  End For
  For $e = 1$ to $e_M^{\max}$:
    $\boldsymbol{\phi}_M \leftarrow \boldsymbol{\phi}_M - \eta \cdot \nabla \mathcal{L}_M(\boldsymbol{\phi}_M)$.
  End For
End For
Output: $\{E_j\}_{j=1}^N$.

Algorithm 2.2: Online training for predictors
Input: $\{E_j\}_{j=1}^N$ from Algorithm 2.1. Training data $(X, Y) = (X_{\text{pool}}^k, Y_{\text{pool}}^k)$. Learning rate η. Maximal epochs $e_P^{\max}$.
Parallel For $j = 1$ to N:
  Initiate: $\boldsymbol{\phi}_{P_j}$. $e = 0$.
  While $\mathcal{L}_P(\boldsymbol{\phi}_{P_j}) > \alpha_P$ and $e < e_P^{\max}$:
    $\boldsymbol{\phi}_{P_j} \leftarrow \boldsymbol{\phi}_{P_j} - \eta \cdot \nabla \mathcal{L}_P(\boldsymbol{\phi}_{P_j})$.
    $e \leftarrow e + 1$.
  End While
End Parallel For
Output: $\{E_j, P_j\}_{j=1}^N$.

5 | PERFORMANCE VALIDATIONS

In this section, the performance of ADE as a *prediction model* is validated in a traffic signal performance prediction problem on the given study site. Specifically, ADE serves to predict a traffic simulator's output $f_{\text{sim}}(\boldsymbol{x})$. f_{sim} outputs average delay per vehicle (s/veh) from the simulation. ADE is trained by limited samples from $f_{\text{sim}}(\boldsymbol{x})$ and tries to predict any sample $\boldsymbol{x}$ in the space Ω. In the validations, first, the diversification performance of the proposed ADE is examined under different numbers of adversarial training rounds T. Different levels of diversity may lead to varied prediction performance. The overall prediction performance of ADE in the entire sample distribution is then validated. Particularly, the prediction performance on the extreme points is also

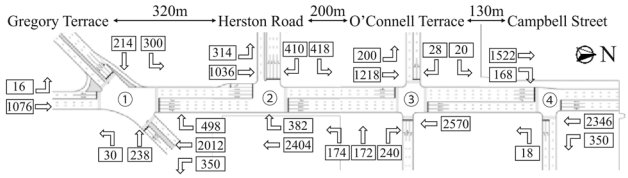


FIGURE 3 Study site and traffic scenario.

TABLE 3 Phase structure.

Intersection No.	Protected movements			Permitted movements
	Phase 0	Phase 1	Phase 2	
1	S-Th, S-LT, N-Th	N-Th, N-LT, N-RT	W-Th, W-LT, E-Th	E-LT
2	S-Th, N-Th	N-Th, N-RT, W-LT	W-LT, W-RT	S-LT
3	E-Th, E-LT, E-RT	E-LT, E-RT, W-RT	S-Th, N-Th, S-LT	W-LT
4	N-Th, N-LT	S-RT, E-LT	-	S-Th

Abbreviations: S, south; N, north; W, west; E, east; Th, through; LT, left-turn; RT, right-turn.

examined. To begin with, a description of the study site is presented.

5.1 | Study site

Experiments are conducted on a four-intersection arterial on Bowen Bridge Road, Brisbane, Australia. The traffic demand was collected in a morning peak. The road geometry and traffic demand are shown in Figure 3. SUMO, that is, Simulation of Urban Mobility, serves as the microscopic traffic simulator (Lopez et al., 2018). The simulator is set to be deterministic.

As mentioned in Section 3.1, this study adopts stage-based modeling for traffic signals. Table 3 presents the phase structure. Movements here are represented by the approach they come from and their flow direction. For example, S-Th stands for a through flow movement from the south approach. The permitted movements are discharged in any phase. The protected movements can only be discharged in their specific phase(s). The parameters of the TSO problem in Equation (1) are shown in Table 4. Each simulation lasts for 20 min, with additional 5 min warm-up. The yellow and all-red times at the end of each phase are 3 and 2 s, respectively.

TABLE 4 Parameters for the traffic signal optimization problem.

Parameter	Value
$c_{\max}, c_{\min}$	120, 60
$\boldsymbol{g}_i^{\min}$	$\{12, 12, \ldots, 12\}$
S_i	$\{(0,1,2),(0,2,1)\}$, if $\lvert\boldsymbol{g}_i\rvert = 3\{(0,1)\}$, if $\lvert\boldsymbol{g}_i\rvert = 2$

TABLE 5 Architecture of ADE.

Module	Input	Hidden	Output	Parameters
Extractor (each)	27	(128,2)	64	44,864
Decoder (each)	64	(128,2)	27	44,827
Predictor (each)	64	(128,3)	1	57,985
Mixer	64	(128,1)	64	33,088
Discriminator	64	–	7	455

5.2 | Experiment setting

5.2.1 | Learning setup

In this particular problem, the decision dimension $d = 19$. However, the input to the ensemble model is a 27d vector, including extra dimensions of sequence-based phase sequences, as explained in Section 3.1. The size of the diverse features is set to 64, as well as the length of the mixed features (see Figure 2 and Section 4.1.1 for the purpose of diverse and mixed features). The extractors, decoders, predictors, and mixer are built on multilayer perceptron. The discriminator uses a linear layer. The number of base learners is $N = 7$. The detailed architecture of the ADE is presented in Table 5.

The hyperparameters in Algorithms 1 and 2 are set as follows. $\alpha_E^{\text{mag}} = 1.0$, $\alpha_{E,DC}^{\text{info}} = 1.0$. $\bar{w}_E^{\text{div}} = 1.0$, $\bar{w}_E^{\text{mag}} = 5.0$, $\bar{w}_{E,DC}^{\text{info}} = 5.0$, $e_E^{\max} = 3000$, $e_M^{\max} = 500$. $f_d\ (\mathcal{L};\alpha) = \exp(-3\alpha/\mathcal{L})$. The following common settings are for all the tested methods for a fair comparison. Rectified Linear Unit (ReLU) is adopted as the activation function. Learning rate η is 0.001. $e_P^{\max}$ is 1000 for predictor training. Two strategies are applied to avoid overfitting: An l_2 regularization with a weighting factor of 0.1 is combined in the loss function for the training of predictors; an early-stopping threshold $\alpha_P = 0.5$ is to terminate the training. Adam is selected to be the optimizer (Kingma & Ba, 2014). The weights are initialized by the He-normal method (He et al., 2015).

Pytorch is adopted as the tool to establish the deep learning models and implement the training and inference.

All the experiments are conducted on a desktop with a 2.50 GHz 8 core CPU and 16 GB memory.

5.2.2 | Proposed models and baselines

To verify the effectiveness of ADE, the ablation study of loss functions in adversarial diversity training and the comparative study with other diversification techniques are conducted. The proposed ADE, ablated ADE, and other baselines are given below.

- ADE-T: The proposed ADE models trained with T, from 1 to 60, rounds of adversarial diversity training to showcase the evolving effects on diversification and prediction performance.
- ADE-i: An ablated ADE whose extractors simply use initialized weights without any training (i.e., no loss function involved).
- ADE-m: An ablated ADE whose extractors are merely trained by $\mathcal{L}_E^{\text{mag}}$, to just restrict the magnitude of the features.
- ADE-0: An ablated ADE whose extractors are trained by $\mathcal{L}_E^{\text{mag}}$ and $\mathcal{L}_{E,DC}^{\text{info}}$, to restrict the feature magnitude and retain information. However, adversarial diversity training is not involved (equivalent to zero round of adversarial diversity training).
- RND-p: An ensemble whose diversity is only triggered by randomizing initial weights. Each base learner uses the same network structure as the predictor module (except that the input size is 27) indicated in Table 5.
- RND-ep: It is similar to RND-p, except that it uses the same extractor-predictor structure as ADE.

During the training, the ADEs and their ablated variants are first pre-trained by an unlabeled dataset $\tilde{X}$ with a size of 300. RND-p and RND-ep do not need pre-training. The predictors of ADEs, ablated variants, and RND baselines are trained by a labeled dataset (X, Y) with a size of 50, in which X is generated by Maximin LHS and Y is the lables from f_{sim}. Unless otherwise specified, the results in the rest of the paper are aggregated from the 25 parallel experiments, using five sets of $\tilde{X}$ and five sets of (X, Y).

5.3 | Diversification performance

This section aims to examine the overall diversification performance in the entire sample space (i.e., feasible region Ω). The trained surrogate models are tested on a test dataset of 1000 data randomly sampled in Ω with true labels is used for the evaluation.

The average relative standard deviation (RSD, the larger, the more diverse) is an index of diversity. Please refer to Appendix 2 for the calculation of the index. The results of diversification are presented in Figure 4.

Figure 4 shows that for ADE-T, RSD kept increasing along with adversarial training rounds T. The trend became stable from about 50th round. RSD reached 28% at the 60th round, significantly larger than RND baselines (13% and 9%) and ADE-0 (12%). It demonstrates that adversarial diversity training can continuously diversify the base learners until stability.

The relatively large diversity of ADE-i and ADE-m came from the randomized initial weights of the extractors. Compared with ADEs, RND baselines cannot guarantee a high diversity using randomization technique as its diversity is not controllable during the training.

Different levels of diversity may lead to varied prediction performance. The following aims to examine the effects of diversity on ADE's prediction performances.

5.4 | Overall prediction performance

In this section, the average prediction performance across the entire sample space is examined. Ensemble mean absolute percentage error (eMAPE) is adopted to evaluate the ensemble prediction performance. The average MAPE of base learners (bMAPE) is used to evaluate base learners' prediction accuracy. Also refer to Appendix 2 for the calculation of the indices. In Figure 4, comparing ADE-0 and ADE-1, the slight diversity introduced by one round of diversity training could improve bMAPE. However, in the following rounds, the constantly increasing diversity deteriorated base learners' prediction performance until around the 15th round. This phenomenon occurred because the increasing diversity resulted in varied base predictions, some of which inevitably deviated from the ground truth.

ADE's ensemble performance (eMAPE) can be enhanced by larger diversity (RSD) and better accuracy of base learners (bMAPE). Compared to ADE-0, the decreased eMAPE of ADE-1 was contributed by both a larger RSD and lower bMAPE. Then, eMAPE increased until around the 10th round as a rapidly deteriorating bMAPE eclipsed the slowly growing RSD. In contrast, the effects of RSD became prominent after the 10th round, leading to a slowly dropping eMAPE.

Although ADE-i and ADE-m presented relatively large diversity, they could not predict well. Due to ablated information retention, their base predictors could not realize effective generalization on test data as reflected by a poor bMAPE. In contrast, RND baselines presented poor prediction performance out of their poor diversity.

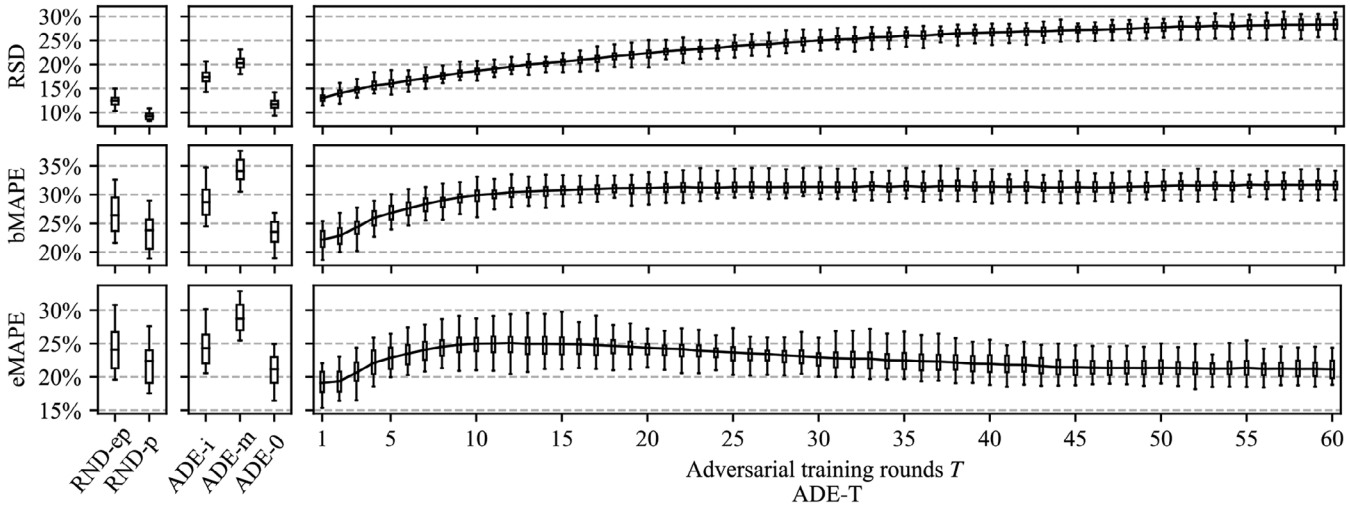


FIGURE 4 Performance validation on random test set.

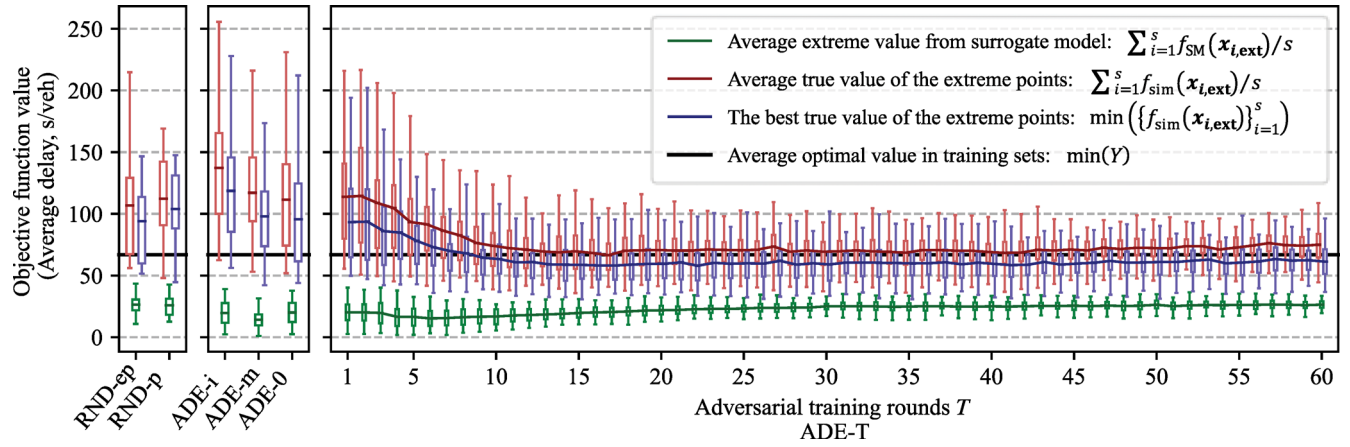


FIGURE 5 Performance validation on extreme points.

5.5 | Prediction performances on extreme points

While common prediction problems focus on overall prediction performance across the entire sample distribution, the nature of surrogate-based optimization problems necessitates accurate prediction for extreme points of the surrogate models. Therefore, in this section, surrogate models' prediction performance on their extreme points is examined.

The procedure of the experiment is described as follows. First, extreme minimum point $\boldsymbol{x}_{\mathbf{ext}}$ of a surrogate model f_{SM} is searched by the DE algorithm. DE may return different $\boldsymbol{x}_{\mathbf{ext}}$ in parallel search sessions. To reduce stochasticity in the results, extreme points from $s = 5$ parallel search sessions form a test set $X_{ext} = \{\boldsymbol{x}_{i,\mathbf{ext}}\}_{i=1}^{s}$. The average extreme value predicted by f_{SM}, $\sum_{i=1}^{s} f_{SM}(\boldsymbol{x}_{i,\mathbf{ext}})/s$, is presented by the green line in Figure 5. Samples in X_{ext} are then validated by true function f_{sim}. The average true value $\sum_{i=1}^{s} f_{sim}(\boldsymbol{x}_{i,\mathbf{ext}})/s$ is presented by the red line. Particularly, the minimal true value among $f_{sim}(\boldsymbol{x}_{i,\mathbf{ext}})$ is also presented in Figure 5 by a blue line. The boxes stand for the ranges of those values in 25 parallel experiments. As mentioned above, the surrogate model is trained by a 50-sample labeled training set (X, Y). The best-known value in the training set, $\min(Y)$, is also presented in Figure 5 as a reference, with the black line for the average best-known value in the five training sets.

Compared with the prediction performances on random samples, all the models presented more noticeable prediction errors on the extreme points as can be seen from the large absolute distance between the predicted values (green) and the true values (red). This is because it is inevitable that only those points biased towards the negative direction the most can be recognized as extreme points by DE. Although the predicted extreme values

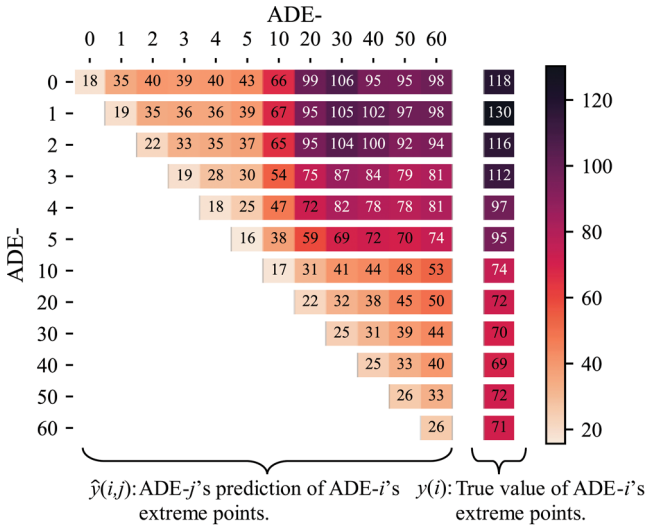


FIGURE 6 The evolution of predictions of extreme points.

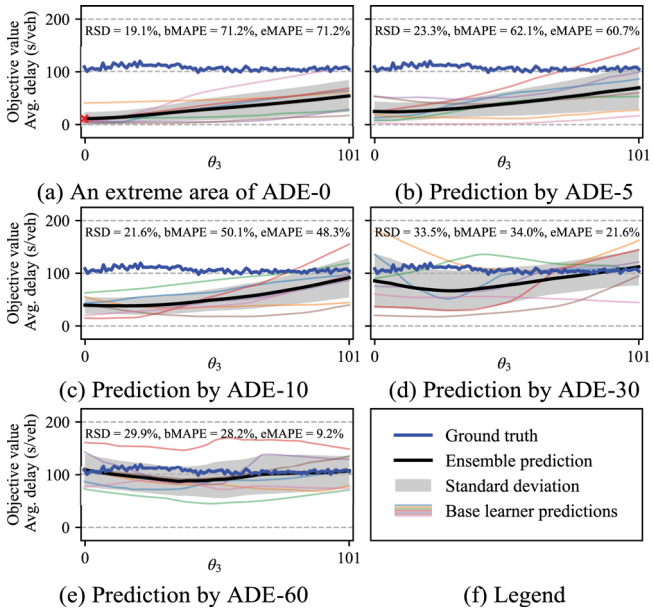


FIGURE 7 Effects of diversity on non-promising extreme point prediction.

varied little across the models, those models showed significantly different prediction performances and capabilities in identifying promising solutions.

With the increase of adversarial training rounds, ADE exhibited a declining trend in the average true values (red line), indicating a reduction in prediction errors on extreme points. The trend converged toward the average optimal value in the training set (black line). Moreover, starting from around the 8th round, ADE could generate the best true value (blue line) that surpassed even the best-known value in the training sets (black line). It demonstrates ADE's remarkable capability of creating promising solutions beyond the existing knowledge from the training data. It is enabled by improved prediction performance on the extreme points. The mechanism behind is explained in the following.

5.6 | Explaining prediction performance improvement on extreme points

Optimization algorithm searches the extreme points in the entire sample space. The extreme points can be either promising or non-promising ones. The prediction performance on extreme points can be improved only when the non-promising extreme points are eliminated as much as possible. Figure 6 presents such a process facilitated by adversarial diversity training. For the ADEs with no or only a few rounds of training, the extreme points of them were usually non-promising solutions as can be seen from the great gap between the predicted and true values. Taking the first row of Figure 6 as an example, ADE-0 predicted an average extreme value of 18, which appeared to be very promising yet was validated 118 by simulation. When diversity was gradually boosted by more training rounds, the prediction of those non-promising extreme points of ADE-0 was gradually corrected by its following ADEs, approaching their true values. Consequently, the non-promising extreme points of ADE-0 will not be extreme points of its following ADEs. The similar correction processes are also shown for ADE-1, ADE-2, and so on.

Figure 7 presents a representative example visualizing and explaining how diversity facilitates the correction process. First, a global extreme point $\boldsymbol{x}^0_{\mathbf{ext}}$ (marked by a red cross; see Appendix 3 for its information) on the high-dimensional response surface of ADE-0 is shown in Figure 7a. To visualize the extreme property of this point, an arbitrary dimension, θ_3 (i.e., the offset of the third intersection), is expanded to be the horizontal axis in the 2d figure.

Figure 7a indicates that the similar mistakes made by base learners of ADE-0 in this area led to this non-promising $\boldsymbol{x}^0_{\mathbf{ext}}$. Nevertheless, the ensemble prediction of this point was corrected slightly by ADE-5, as shown in Figure 7b, due to diversity and generalization. With five rounds of adversarial diversity training, the enhanced diversity enforced the base predictions to differ in this area. Such a diversification did not center on the predicted extreme value in Figure 7a. Since the base predictions were already extreme values, they could not produce predictions with even lower values. Therefore, as an effect of generalization, the diversification was realized by increasing some base predictions, approaching the ground truth. It demonstrates that although the ensemble prediction is simply an average of base predictions, it can still enforce the

correction process. This process was further facilitated in the following rounds illustrated in Figure 7c–e. As a result, the ensemble prediction of this area was gradually corrected, and such a non-promising extreme area diminished from ADE's belief.

It should be noted that the goal of the correction process is not to fit the ground truth perfectly. As long as the predicted value is over the threshold of being an extreme point, the non-promising extreme point is successfully removed from its belief. For example, Figure 7c illustrates that ADE-10 outputs about 40 for $\boldsymbol{x}^0_{\mathbf{ext}}$. Figure 6 indicates that the threshold of ADE-10 is 17, much lower than 40. In this way, $\boldsymbol{x}^0_{\mathbf{ext}}$ is not an extreme point of ADE-10 anymore.

While the base predictions of non-promising extreme points can be easily diversified, the remaining extreme points are the ones all the base learners are still confident about. That is, the base predictions of the remaining extreme points can resist the diversification and maintain their predicted values. Such extreme points are thus more likely to be promising solutions.

This diversity mechanism seems similar to how ensembles improve overall performance in the entire sample space. However, it necessitates a higher degree of overall diversity. This is because the degree of diversity can be uneven across different areas of the response surface, leaving some areas still undiversified. The non-promising extreme points can still exist in those areas. Therefore, the overall diversity should be substantially boosted to cover the whole sample space and to clear the non-promising extreme points as much as possible.

5.7 | Summary

In this section, three performance indices are evaluated for the proposed ADE, including overall diversification, prediction performances in the entire sample space and on extreme points. The following can be concluded from the results.

1. The adversarial diversity training can continuously diversify the base learners' predictions as the training rounds proceed.
2. The growing degree of ADE diversity has different effects on prediction performance on two test sets. On the random points set, a slight degree of diversity is preferable, whilst excessive diversity can be harmful.
3. On the extreme points set, the prediction performance is continuously enhanced by growing diversity until reaching stability. Consequently, the solution quality of the extreme points is also improved.

In the next section, the efficiency of ADEs with different degrees of diversity will be examined in TSO problems.

6 | EVALUATION IN STATIC TSO

Section 5 validates the performance of ADE as a *prediction model* in a traffic signal performance prediction problem under static demand. In this section, ADE serves as a *surrogate model* in surrogate-based TSO problems and is evaluated in terms of the optimization efficiency. TSO experiments on the same study site in Section 5 will be conducted. In these experiments, the traffic demand is static and the signal timing is fixed over time.

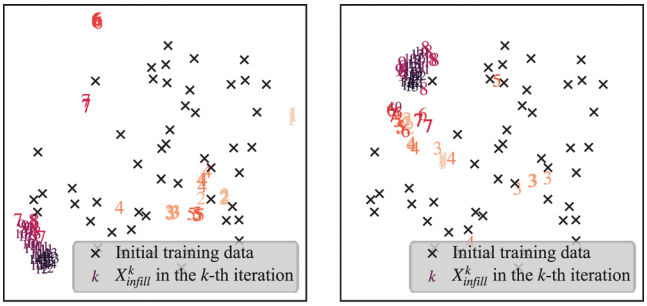
6.1 | Experiment settings

The optimization experiments in this section follow the framework in Figure 1. Specifically, diverse extractors are pre-trained by T rounds of adversarial diverse training in Step 1 and then fixed. The predictors are trained or re-trained iteratively in Step 4. The experiment settings are specified here. The settings for the study site are identical to those for the validations in Section 5. The same $\tilde{X}$ and (X, Y) in the validations are used to train the initial surrogate model. To search for better solutions continuously, five extreme points returned by the DE algorithm will be selected as the infill samples and evaluated by simulation at each iteration. The iterative optimization will go through 14 iterations. Therefore, at the end of the optimization, a total of $50 + 14 \times 5 = 120$ samples will be used to train the surrogate model and to search for the optimal solution.

To avoid redundant experiments, some representative models are selected to be evaluated in this section, including all the RND baselines, all the ablated ADEs, ADE-1 to ADE-5 with interval 1, and ADE-10 to ADE-60 with interval 10. Four widely used surrogate models are used as baselines for the comparison, including the GP model, RBF model, support vector regression (SVR), and RF.

6.2 | Results

Two different budgets are applied to evaluate the efficiency of optimization. First, a budget of 120 simulation runs is applied to analyze the models' performance in discovering better solutions. Then, the influence of the models' time efficiency on optimization results is analyzed under a budget of 120 s.



(a) ADE-1 (b) ADE-30

FIGURE 8 Distribution of infill samples.

6.2.1 | Results under a budget of 120 runs

The evaluated surrogate models are divided into two groups for the convenience of analyzing the results. The first group consists of ensembles that are not fully diversified, including all the RND baselines, all the ablated ADEs, and ADE-1 to ADE-5. ADE-10 to ADE-60 are regarded as fully diversified ensembles.

To begin with, typical examples of infill sample distribution are presented in Figure 8 to exhibit the patterns of the iterative optimization procedures. A t-distributed stochastic neighbor embedding (t-SNE) algorithm (van der Maaten & Hinton, 2008) is employed to visualize the high-dimensional distribution in a 2D figure. The location of number k stands for the location of the infill samples from the kth iteration. Samples from different iterations are also distinguished by colors. ADE-1 and ADE-30 are representatives of the two groups. In both cases, the infill samples at the early iterations scatter in the entire distribution to conduct a global search for promising solutions. From about the sixth round, the infill samples tend to gather in a local area to conduct local search. Despite a similar pattern, the two groups presented different performances.

Figure 9a shows the final best-known values at the 14th iteration. The fully diversified models can obtain better final solutions than the non-fully diversified ones. Comparing the convergence curve in Figure 9b,c, the fully diversified ADE-10 to ADE-60 converged significantly faster during the initial iterations (i.e., the global search). This rapid convergence contributed to achieving a superior final solution over the others. To further investigate the details of the fast convergence, the records of objective values of the infill samples are provided in Figure 10.

Figure 10 illustrates the true value $y^k_{\min}$ and predicted value $\hat{y}^k_{\min}$ of the best infill sample at each iteration, which are mathematically defined in Equation (10). Figure 10 indicates that ADE-1 was more likely to return the infill samples (i.e., extreme points) validated to be non-promising at the initial iterations as can be indicated by their large true values. On the contrary, the fully diversified ADE-30 could rapidly discover promising solutions, much more efficient than ADE-1, with less wasted simulation budget. This observation meets the expectation, given that the validation results on extreme points (Section 5.4) have shown that the fully diversified ADEs can identify solutions with better quality.

$$\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{\min}}, y^k_{\min} = \underset{\boldsymbol{x}\in X^k_{\text{infill}}}{\operatorname{argmin}} f_{\text{sim}}(\boldsymbol{x}), \min_{\boldsymbol{x}\in X^k_{\text{infill}}} f_{\text{sim}}(\boldsymbol{x}) \tag{10a}$$

$$\hat{y}^k_{\min} = f_{\text{SM}^k}\left(\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{\min}}\right) \tag{10b}$$

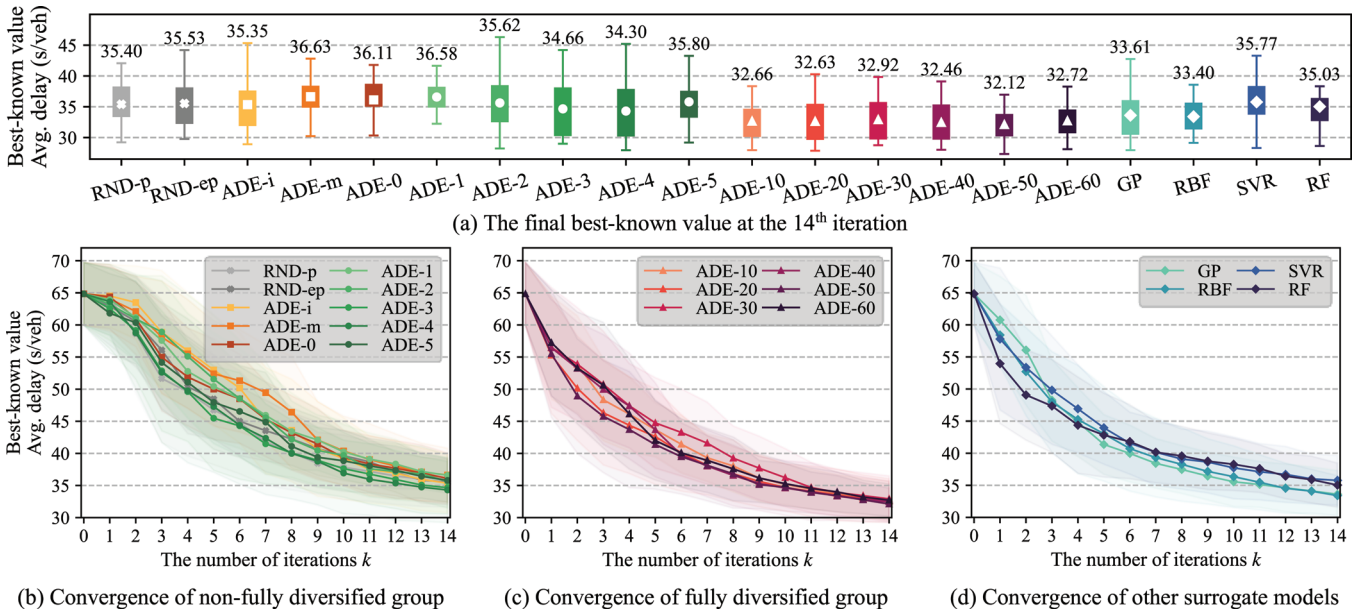
As all the non-promising solutions are gradually cleared by infill samples, the prediction errors on extreme points will decrease as Figure 10 illustrates. That is because, from the angle of backpropagation, the decreasing prediction errors will gradually lead to the fact that the surrogate models will not be dramatically updated by the infill samples. Consequently, surrogate models will not update their beliefs in promising areas. The following iterations will focus on the local search in the most promising area identified. In this process, the local area will be gradually densely populated by infill samples as Figure 8 shows.

In a densely sampled area, the prediction performance on extreme points may not vary a lot among the surrogate models. Therefore, ADE-1 and ADE-30 presented similar convergence rates in local search. However, since local search is conducted only in the promising area, the quality of the final best-known solution is still greatly dependent on the efficiency of finding a good promising area at the initial iterations.

Figure 9c presents the convergence rate of the optimizations using traditional surrogate models, including GP, RBF, SVR, and RF. The convergence rate of fully diversified ADEs was the same or faster than that of traditional surrogate models at the early iterations. At the end of the optimization, ADE presented slightly better best-known value over traditional surrogate models, which can be indicated by Figure 9a as well.

6.2.2 | Results under a budget of 120 s

In practice, it is more straightforward to restrict the time budget to return demand-responsive signal timing solutions in a short time window. The time efficiency depends on not only the convergence rate over the simulation budget but also the time required for training the base predictors, which is proportional to the required epochs to converge in backpropagation.



(a) The final best-known value at the 14th iteration

(b) Convergence of non-fully diversified group

(c) Convergence of fully diversified group

(d) Convergence of other surrogate models

FIGURE 9 Performances in optimization.

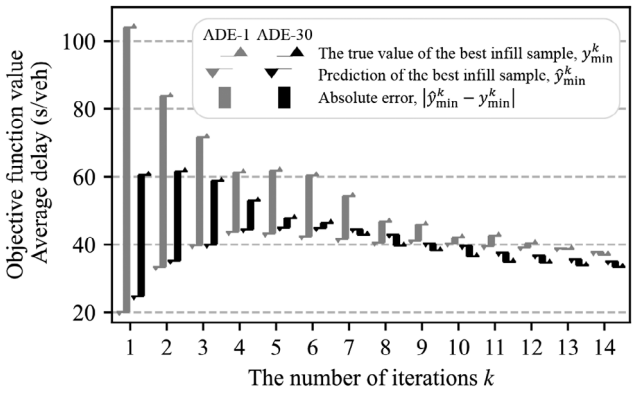


FIGURE 10 The predicted and true values of the best infill samples (see Equation 10 for detailed definitions).

The actual simulation time for one run is approximately 1 s. For convenience, assume each simulation takes exactly a second. If the training time is ideally zero, it can conduct 120 runs in 120 s, and the final best-known value will be the same as Figure 9a. Therefore, the time budget is set to be 120 s to analyze how training time impacts the final best-known value.

Table 6 presents the average time spent for predictor training, simulations, and DE algorithm in 120 s. It shows a consistent decrease in predictor training time along with the increased rounds of adversarial training. The reduced training time leaves more time for simulation runs, from an average of 98.32 runs by ADE-1 to 111.26 runs by ADE-60. Consequently, the difference of best-known value (i.e., average delay) under 120 s and 120 runs budget also presents a decreasing trend. Such a tighter gap of final best-known values maintains the solution quality under a tight time budget.

Some baseline surrogate models presented advantages in time efficiency of training and prediction because their

TABLE 6 Time allocation and performance under different budgets.

Model	Time cost in a 120 s budget			Average delay (s/veh)		
	Training	Simulation	DE	120 runs	120 s	Diff.
RND-ep	25.49	92.63	1.88	35.53	40.78	5.25
RND-p	19.56	98.29	2.15	35.40	37.79	2.39
ADE-i	24.62	92.34	3.03	35.35	40.73	5.38
ADE-m	22.72	93.96	3.32	36.63	39.88	3.25
ADE-0	18.49	97.59	3.92	36.11	39.13	3.02
ADE-1	18.21	98.32	3.47	36.58	40.23	3.65
ADE-2	18.66	97.91	3.43	35.62	39.17	3.55
ADE-3	18.63	97.60	3.77	34.66	37.60	2.94
ADE-4	13.57	102.56	3.87	34.30	35.66	1.36
ADE-5	13.15	103.07	3.78	35.80	37.86	2.06
ADE-10	9.91	106.07	4.03	32.66	33.97	1.30
ADE-20	7.38	108.47	4.15	32.63	33.45	0.82
ADE-30	5.62	110.05	4.34	32.92	33.53	0.62
ADE-40	4.90	110.73	4.37	32.46	32.86	0.40
ADE-50	4.53	111.08	4.39	32.12	32.77	0.65
ADE-60	4.31	111.26	4.44	32.72	33.20	0.48
GP	0.08	117.09	2.83	33.61	33.91	0.30
RBF	0.07	116.66	3.26	33.40	33.99	0.60
SVR	0.07	117.24	2.69	35.77	35.88	0.12
RF	2.40	114.01	3.58	35.03	36.37	1.34

Abbreviations: GP, Gaussian process; RBF, radial basis function; SVR, support vector regression; RF, random forest.

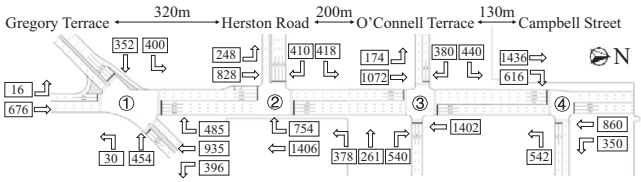


FIGURE 11 Traffic pattern with heavy turning flows.

model structures are much lighter than ADEs. Therefore, the difference in average delay under the 120 s budget is less than that of deep ensemble models (see column Diff.).

6.3 | Applicability in other traffic scenarios

This section presents TSO experiments in other traffic scenarios, aiming to evaluate ADE's applicability to a broader range of traffic patterns and demand.

Regarding the traffic demand described in Figure 3 as a basic scenario, two extra levels of traffic demand were tested with demand factors of 0.8 and 1.2 to examine the ADE's performance in lighter and heavier demand scenarios. The experiments above are all conducted on a through-traffic-predominant pattern in Figure 3. A significantly different traffic pattern is also tested as illustrated in Figure 11. In this pattern, there are heavy turning flows merging into and diverging from the arterial. Similarly, three demand factors of 0.8, 1.0, and 1.2 were tested in this pattern.

To further illustrate the efficiency of ADE in TSO problems, this experiment incorporates additional baselines for comparison including

1. Analytical models MAXBAND (Little et al., 1981) and LMP-BAND (Z. Tang et al., 2023). They are arterial signal coordination models, especially suitable for the four-intersection arterial site.
2. Rule-based Webster method (Webster, 1958), the most classic fixed-time control method; and max pressure (MP; Varaiya, 2013), a popular actuated control method;
3. Simulation-based optimization with an infinite budget (SBO-inf) and a limited budget of 120 simulation runs (SBO-lim). The solution of SBO-inf is a result of an exhaustive search. Therefore, it can be regarded as the true optimal solution and be the base of the optimization gap.

The results are shown in Table 7. The upper part presents the performances of various surrogate-based optimizations with two types of budget limit. Table 7 shows that fully diversified ADE-10 to ADE-60 outperformed the other ensemble surrogate models in all the tested scenarios. It demonstrates that the effects of ensemble diversity on optimization performance are not scenario-specific. This is unsurprising, given that the simulation-based objective function is not involved in the adversarial diversity training of extractors.

Compared with traditional surrogate models, fully diversified ADEs also presented comparable performance in most scenarios and realized slight average improvements by 3.6% (ADE-50 14.5% vs. GP 18.1%). Moreover, ADE models were more robust to various traffic scenarios, whereas the performance of traditional surrogate models, particularly SVR and RF, deteriorated under scenarios with a demand factor of 1.2.

The lower part of Table 7 presents the performances of analytical or rule-based TSO techniques. Compared with these TSO models, fully diversified ADE models improved the optimization gap by up to 9.2% (comparing LMP-BAND and ADE-50 under the 120 s budget). The improvement comes from an accurate replicate of traffic flow over the analytical models.

Webster and MP methods did not realize satisfactory performance, because coordination among adjacent signals is not considered in their calculations.

6.4 | Discussion

In the validation section, ADE-1 performs the best on the random points test set, while ADE-10 to ADE-60 remarkably improves extreme point prediction over ADE-1. From the optimization results, it can be concluded that the deep ensemble model's prediction performance on the extreme points, rather than the overall prediction performance, may contribute more to the performance in the surrogate-based optimization. At the broader level, this finding may also fit into other types of surrogate models that are prone to misrecognized promising solutions caused by poor generalization on extreme points.

Comparing analytical models and surrogate models, the latter outperforms the former by accurate traffic dynamics. Figure 12 shows examples of the vehicle trajectories generated by LMP-BAND and ADE-30. At the intersections at about 350 and 850 m, LMP-BAND did not anticipate a blockage by residual queues. In contrast, information from simulations enabled ADE-30 to evaluate signal timing accurately. ADE-30 identified a solution with a slight shift of offset to avoid the interruption by the residual queues.

It is found that ADE-50 performed better than ADE-60, whereas the same level of diversity was observed in Figure 4. It may be an effect from the other two loss functions in adversarial diversity training. An

TABLE 7 Average delay (s/veh) under various traffic scenarios. Optimization gaps are calculated based on SBO-inf.

Pattern	Two-way predominant flows						Heavy turning flows									
Demand factor	0.8		1		1.2		0.8		1		1.2		Average			
Budget[a]	Same runs	Same time	Same runs	Same time	Same runs	Same time	Same runs	Same time	Same runs	Same time	Same runs	Same time	Same runs	Gap	Same time	Gap
RND-ep	31.2	33.8	35.5	40.8	47.3	55.4	28.9	30.7	40.4	44.8	55.0	61.1	**39.7**	**31.9%**	**44.4**	**47.6%**
RND-p	31.1	33.3	35.4	37.8	47.3	52.3	29.1	30.2	39.8	42.9	54.0	58.4	**39.4**	**31.0%**	**42.5**	**41.1%**
ADE-i	29.7	31.2	35.3	40.7	47.4	54.2	28.7	30.7	39.7	43.1	55.0	58.0	**39.3**	**30.5%**	**43.0**	**42.8%**
ADE-m	29.0	31.0	36.6	39.9	46.7	53.2	29.2	29.8	41.1	45.6	52.4	57.6	**39.2**	**30.1%**	**42.9**	**42.3%**
ADE-0	29.6	32.3	36.1	39.1	44.9	52.8	28.5	29.2	40.2	44.3	49.5	55.3	**38.1**	**26.7%**	**42.2**	**40.1%**
ADE-1	29.6	32.1	36.6	40.2	43.2	50.3	29.5	31.0	40.1	43.3	50.3	54.9	**38.2**	**26.9%**	**42.0**	**39.4%**
ADE-2	30.2	32.9	35.6	39.2	43.9	50.9	29.6	30.7	39.6	42.4	48.9	54.8	**38.0**	**26.1%**	**41.8**	**38.9%**
ADE-3	29.9	32.7	34.7	37.6	43.7	49.7	29.0	30.9	36.1	40.3	49.6	55.2	**37.1**	**23.4%**	**41.1**	**36.4%**
ADE-4	29.8	31.7	34.3	35.7	45.1	51.8	29.4	30.6	39.9	41.5	47.4	50.3	**37.6**	**25.0%**	**40.2**	**33.7%**
ADE-5	28.4	30.8	35.8	37.9	43.5	48.4	28.7	29.4	38.3	40.5	47.4	51.5	**37.0**	**23.0%**	**39.8**	**32.0%**
ADE-10	27.3	28.1	32.7	34.0	42.2	45.9	27.2	28.0	36.1	37.6	41.9	43.8	**34.6**	**14.9%**	**36.2**	**20.4%**
ADE-20	27.6	28.1	32.6	33.4	40.1	43.9	26.3	26.5	34.5	35.3	41.7	44.1	**33.8**	**12.3%**	**35.2**	**17.0%**
ADE-30	27.2	27.7	32.9	33.5	41.1	43.6	27.2	27.6	33.7	34.1	41.7	43.1	**34.0**	**12.8%**	**34.9**	**16.0%**
ADE-40	28.8	29.1	32.5	32.9	41.1	43.8	27.3	27.5	33.1	33.5	40.0	40.5	**33.8**	**12.2%**	**34.5**	**14.7%**
ADE-50	27.9	28.3	32.1	32.8	41.9	43.9	27.2	27.4	32.8	33.3	40.4	41.2	**33.7**	**11.9%**	**34.5**	**14.5%**
ADE-60	29.6	30.1	32.7	33.2	41.5	43.2	28.1	28.5	34.0	34.5	40.6	40.8	**34.4**	**14.3%**	**35.1**	**16.4%**
GP	28.1	28.3	33.6	33.9	45.0	45.3	26.5	26.5	32.2	32.3	46.9	47.0	**35.4**	**17.6%**	**35.6**	**18.1%**
RBF	30.7	30.7	33.4	34.0	44.8	44.9	28.7	28.7	32.2	32.2	49.3	49.3	**36.5**	**21.3%**	**36.6**	**21.7%**
SVR	33.5	33.8	35.8	35.9	65.4	66.3	28.5	28.8	33.7	33.9	50.0	51.0	**41.1**	**36.7%**	**41.6**	**38.2%**
RF	30.9	31.2	35.0	36.4	67.6	68.0	29.8	30.2	39.3	39.6	53.7	55.1	**42.7**	**41.8%**	**43.4**	**44.2%**

Pattern	Two-way predominant flows			Heavy turning flows				
Demand factor	0.8	1	1.2	0.8	1	1.2	Average	Gap
MAXBAND	29.9	35.9	47.1	31.2	36.2	45.1	**37.6**	**24.8%**
LMP-BAND	29.0	34.5	47.3	32.1	35.7	45.0	**37.3**	**23.7%**
Webster	36.7	42.9	60.0	33.5	43.9	46.8	**44.0**	**46.1%**
MP	38.5	41.2	58.0	33.4	42.4	48.4	**43.6**	**44.9%**
SBO-lim	35.9	43.7	53.7	32.7	42.3	49.3	**42.9**	**42.6%**
SBO-inf	25.0	29.4	36.0	25.0	29.9	35.4	**30.1**	–

Abbreviations: GP, Gaussian process; MP, max pressure; RBF, radial basis function; SBO-inf, simulation-based optimization with an infinite budget; SBO-lim, limited budget of 120 simulation runs; SVR, support vector regression; RF, random forest.

[a]Budget: The optimizations using RND-ep, RND-p, ADE-i, ADE-m, ADE-0, ADE-1 to ADE-5, ADE-10, ADE-20, ADE-30, ADE-40, ADE-50, and ADE-60 are conducted with a fixed budget of 120 simulation runs or 120 s of wall clock time. MAXBAND, LMP-BAND, Webster, and MP solve the optimization using almost no time.

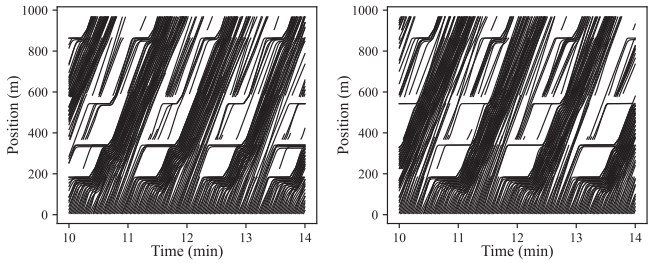


FIGURE 12 Vehicle trajectories. North to south direction.

early-stop strategy can be a rule of thumb to avoid that the pre-training is excessive and becomes harmful to ADE's performance.

Although the time cost for a simulation is assumed 1 s, the actual simulation time may be substantially reduced using a more powerful computer. In this case, the optimization efficiency can be further improved.

7 | LARGE-SCALE ONLINE TSO UNDER DYNAMIC DEMAND

The TSO experiments above are the cases with static demands, and the latency (i.e., 120 s time budget) of signal update caused by the surrogate-based optimization procedures is not reflected in the optimization results. However, the traffic demand is dynamic over time in the real world. On the one hand, it requires a timely return of signal timing solution from surrogate-based TSO with good solution quality. On the other hand, any latency can negatively impact traffic performance. This task can be even more challenging in networks with larger scale. This section aims to examine the performance of ADE in a large-scale online TSO under dynamic traffic demand.

7.1 | Study site

This experiment was conducted on a traffic network in Tongxiang, China. It has 16 signalized intersections (No. 1–16) and several major unsignalized intersections (No. 17–23) as Figure 13a shows. The dynamic demand was collected in a morning peak from 7 to 9 a.m. The simulation lasted for 120 min with additional 15 min for warm-up. The demand profile of the signalized intersections and the total traffic input to the network are shown in Figure 13b,c. Despite a drastic demand change, the fixed-time signal scheme currently implemented on this site is updated only at 8:30 a.m. The "lazy" update leads to congestions, which calls for a dynamic signal timing scheme that caters to real-time traffic demand.

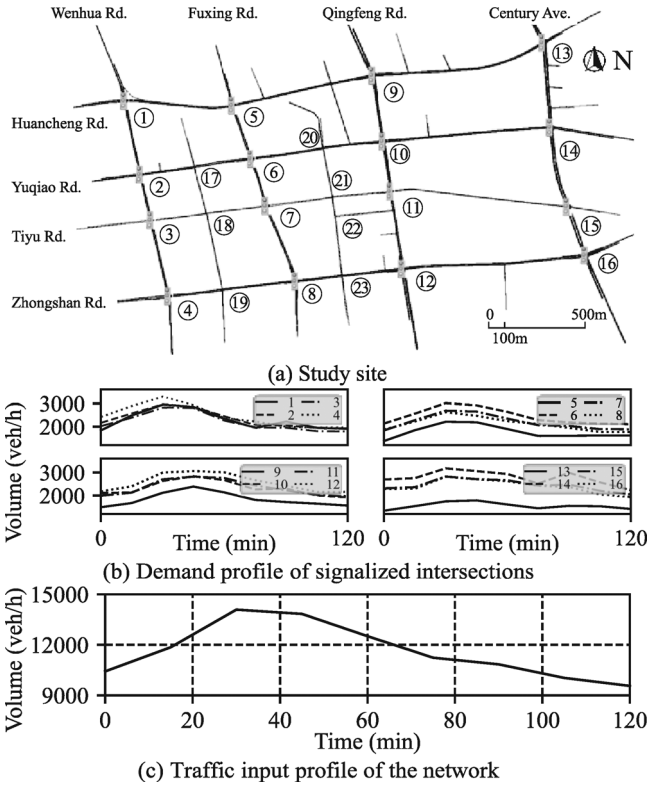


FIGURE 13 The large-scale study site and demand profile.

7.2 | Distributed surrogate modeling

In online surrogate-based TSO, the training data are very sparse, considering there are hundreds of decision variables for large-scale networks. In this case, it is difficult to establish a valid approximation of the simulation-based function. To tackle this problem, a distributed surrogate modeling approach is introduced.

In urban traffic networks, some traffic performance indices, such as the total delay, are the sum of the performances of each intersection. Taking advantage of this additivity property, the network total delay predicted by a surrogate model $f_{\mathrm{SM}}(\boldsymbol{x})$ can be expressed as the sum of local total delays predicted by local surrogate models $f_{i,\mathrm{SM}}(\boldsymbol{x_i})$ as Equation (11a) shows.

$$f_{\mathrm{SM}}(\boldsymbol{x}) = \sum_{i \in \mathbb{N}_I \cup \{-1,-2\}} f_{i,\mathrm{SM}}(\boldsymbol{x_i}) \tag{11a}$$

$$\boldsymbol{x_i} = \begin{cases} [c, \boldsymbol{\delta_i}, \boldsymbol{\delta}_{j_1}, \boldsymbol{\delta}_{j_2}, \ldots], if\ i \in \mathbb{N}_I \\ [c, \boldsymbol{\delta}_{j_1}, \boldsymbol{\delta}_{j_2}, \ldots], if\ i \notin \mathbb{N}_I \end{cases}; j_1, j_2, \ldots \in \mathbb{N}_i \tag{11b}$$

In this study, the local surrogate models are established for each signalized intersection $i \in \mathbb{N}_I = \{1, 2, \ldots, 16\}$. As Equation (11b) indicates, the input of $f_{i,\mathrm{SM}}$, $\boldsymbol{x_i}$, is the decision variables most relevant to the total delay of intersection i, including the common cycle length c, signal

TABLE 8 Input and output of local models.

Local model i	$\mathbb{D}_i$	$\mathbb{N}_i$	Local model i	$\mathbb{D}_i$	$\mathbb{N}_i$
1	{1}	{2,5}	10	{10}	{6,9,11,14}
2	{2}	{1,3,6}	11	{11}	{7,10,12,15}
3	{3}	{2,4,7}	12	{12}	{8,11,16}
4	{4}	{3,8}	13	{13}	{9,14}
5	{5}	{1,6,9}	14	{14}	{10,13,15}
6	{6}	{2,5,7,10}	15	{15}	{11,14,16}
7	{7}	{3,6,8,11}	16	{16}	{12,15}
8	{8}	{4,7,12}	−1	{17,18,19}	{2,3,4,6,7,8}
9	{9}	{5,10,13}	−2	{20,21,22,23}	{6,7,8,10,11,12}

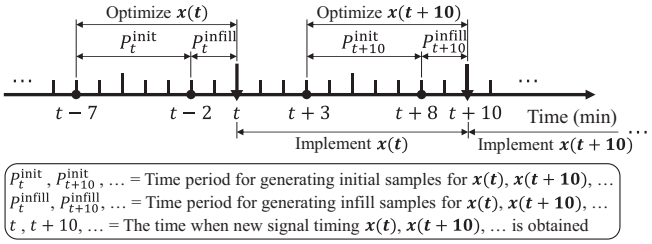


FIGURE 14 Optimization schedule.

timing of the ego intersection $\boldsymbol{\delta_i}$, and signal timing of its adjacent intersections $\boldsymbol{\delta_j}$. The set of adjacent intersections of intersection i is denoted by $\mathbb{N}_i$. The output of $f_{i,\text{SM}}$ is the total delay on the entry lanes of intersection i.

In the network shown in Figure 13a, there are two areas without traffic signal devices. The one includes unsignalized intersections No. 17 to 19, and the other includes No. 20 to 23. Two extra local surrogate models, numbered by $i = -1$ and $i = -2$, are built to predict the total delay for the two areas. These models output the total delay of their respective areas. The input is the signal timing of the intersections that envelop the area. Refer to Table 8 for the detailed input (i.e., signal variables of intersections in $\mathbb{N}_i$) and output (i.e., total delay of intersections in $\mathbb{D}_i$) of each local surrogate model.

7.3 | Optimization schedule

To respond to the dynamic traffic demand, the surrogate-based online TSO is operated following the schedule in Figure 14. It updates signal timing every 10 min. To cater to the traffic demand at $t-2$, the TSO starts over at $t-7$. It spends 5 min from $t-7$ to $t-2$ to obtain initial samples. The demand at $t-2$ is unknown at this stage. A naive strategy of assuming a constant rate of demand change is used for demand prediction at $t-2$. Then, it spends 2 min from $t-2$ to t to search for the solution by infill samples. The

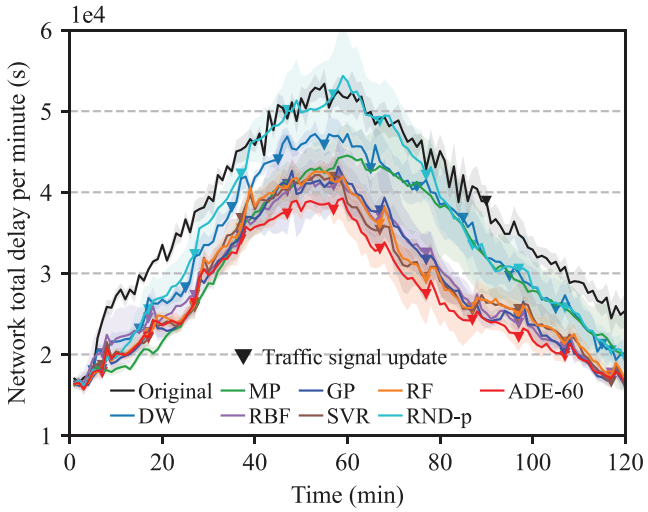


FIGURE 15 Results of dynamic total delay. Original, the field-implemented signal scheme; DW, dynamic Webster; MP, max pressure; GP, Gaussian process; RBF, radial basis function; SVR, support vector regression; RF, random forest; RND, randomization.

new signal timing is obtained and updated at t, and implemented from t to $t+10$. One simulation on this network spends about 4 s. Therefore, the simulation budget for each TSO session is about $(5+2)\times(60/4) = 105$ simulation runs. Signals derived from the Webster method are also updated every 10 min and thus called dynamic Webster (DW) in the following contents. The MP method is adopted during the warm-up period for all the experiments. LMP-BAND and MAXBAND models are not suitable in this site and thus not applied in this experiment.

7.4 | Results

Figure 15 illustrates the time-varying network total delay per minute (s) for each method. The intersections in this study site are relatively less coupled, given the larger distance between them. Therefore, self-regulating MP method demonstrated comparatively good performance when the demand was low. However, when the network became more and more congested, surrogate-based optimization began to present its advantages.

When traffic is heavier, it is important to clear the queues and avoid fast accumulation of queues. Otherwise, a spillback is very likely to happen. However, clearing an upstream queue may cause an accumulation of downstream queues. Therefore, it heavily relies on a systematic optimization to balance the queues across the intersections. Self-regulating controls, such as DW and MP, could not realize effective optimization because the control of the intersections was uncoordinated. In contrast, surrogate-based optimizations could optimize signal

schemes globally, outperforming self-regulating schemes. Despite of 120 s latency, the signal update was still timely as indicated by the good performances.

Among the surrogate models, ADE-60 presented better performance over traditional surrogate models (GP, RBF, SVR, and RF) and the non-fully diversified RND-p when the traffic became heavier, which is consistent with the results under heavy demand in Table 7. Such an advantage was maintained even when the demand notably dropped. That was because ADE-60 prevented a more severe congestion from happening, such that it did not need to recover from a more congested traffic state.

8 | CONCLUSION

Surrogate-based optimization has been a computationally efficient alternative to simulation-based optimization in TSO problems. However, as powerful tools in system modeling, deep learning models have not been extensively applied in surrogate-based TSO, because of their poor performance on the extreme points (Gora et al., 2018) and subsequent inefficient identification of promising solutions. A diverse ensemble of deep learning models is a classic approach to improve generalization in unseen areas. However, under tight simulation and time budget, ensembles with traditional diversity training are still not competent surrogate models in TSO problems due to unsolved poor performance on extreme points and low computational efficiency.

To solve the problems, an ADE approach is proposed. In ADE, each base learner consists of an extractor and a predictor. To reduce online training time, the training of ADE is separated into offline diversity training for extractors and online training for predictors. An adversarial diversity training algorithm is developed to diversify the extractors consistently.

Validation experiments have shown that tested on the set of extensive random samples, ADE continuously boosted overall diversity in the entire distribution, along with the increased number of adversarial training rounds. Prediction performance was improved by a slight degree of diversity, yet deteriorated by excessive diversity. On the contrary, when tested on the set of extreme points, fully diversified ADEs presented remarkably improved prediction performance, demonstrating enhanced capability of discovering promising solutions. Such improvements come from diminishing non-promising extreme points, which is facilitated by boosted overall diversity.

To examine the optimization efficiency of the ADE as a surrogate model, TSO experiments were conducted on a four-intersection arterial. The results indicated that fully diversified ADEs outperformed the non-fully diversified ones in terms of the final best-known value under a tight simulation budget because of their improved performances on extreme points. A decreasing trend of predictor training time along with the diversity training rounds further improved the time efficiency of ADE, maintaining optimization performance under a tight time budget. Experiments on other traffic scenarios demonstrated ADE's applicability in a wider range of traffic patterns and demand levels. In those scenarios, ADE decreased the optimization gap by 9.2% over analytical TSO models and 3.6% over traditional surrogate models, in terms of best-known value under the 120 s budget.

It is proved that ADE can be applied to large networks, which is enabled by a proposed distributive surrogate modeling. The enhanced optimization efficiency of ADE renders timely signal solutions with good quality, which facilitates noticeable performance improvement under dynamic traffic scenarios.

As a general method, the proposed ADE can serve as a surrogate model in other simulation-based optimization problems in broader disciplines. This study also provides an insight that the performance of a surrogate model in an optimization problem may mainly depend on the prediction performance on extreme points (also called decision prediction error in related attempts by Elmachtoub & Grigas, 2021; Gupta & Zhang, 2024) rather than the overall prediction performance.

Despite the promising results, there are several considerations that could improve the work. First, due to a lack of uncertainty term in Equation (2c), the optimization based on ADE can be easily stuck in the local optimum, instead of actively exploring uncertain areas, as can be indicated by Figure 8. Fortunately, ADE has the potential for good performance in uncertainty quantification (UQ) as diversity is the foundation of ensemble-based UQ. Future work can focus on ADE's UQ potential to improve its performance in surrogate-based optimization.

ADE provides a framework for various deep learning models to enhance the prediction of extreme points. Another stream of future work involves addressing different TSO problems using expertized deep learning models with the ADE framework. For example, self-organizing surrogate-based TSO can be developed with deep learning models receiving real-time vehicle arrival rate or connected vehicle data.

Last, future works will focus on applying ADEs into stochastic surrogate-based optimizations under uncertain demand and multi-objective optimizations in TSO problems.

REFERENCES

Baldi, S., Michailidis, I., Ntampasi, V., Kosmatopoulos, E., Papamichail, I., & Papageorgiou, M. (2017). A simulation-

based traffic signal control for congested urban traffic networks. *Transportation Science*, *53*(1), 6–20.

Bao, J., Zheng, L., & Ban, X. (2023). Biobjective robust network-wide traffic signal optimization against cyber-attacks. *Transportation Research Part C: Emerging Technologies*, *151*, 104124.

Bisbo, M. K., & Hammer, B. (2020). Efficient global structure optimization with a machine-learned surrogate model. *Physical Review Letters*, *124*(8), 086102.

Blatman, G., & Sudret, B. (2010). An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. *Probabilistic Engineering Mechanics*, *25*(2), 183–197.

Borkowski, O., Koch, M., Zettor, A., Pandi, A., Batista, A. C., Soudier, P., & Faulon, J.-L. (2020). Large scale active-learning-guided exploration for in vitro protein production optimization. *Nature Communications*, *11*(1), 1872.

Boursier Niutta, C., Wehrle, E. J., Duddeck, F., & Belingardi, G. (2018). Surrogate modeling in design optimization of structures with discontinuous responses. *Structural and Multidisciplinary Optimization*, *57*(5), 1857–1869.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Chen, C., Wei, H., Xu, N., Zheng, G., Yang, M., Xiong, Y., Xu, K., & Li, Z. (2020). Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(4), 3414–3421.

Chen, X., Osorio, C., & Santos, B. F. (2019). Simulation-based travel time reliable signal control. *Transportation Science*, *53*(2), 523–544.

Chen, X., Zhang, L., He, X., Xiong, C., & Li, Z. (2014). Surrogate-based optimization of expensive-to-evaluate objective for optimal highway toll charges in transportation network. *Computer-Aided Civil and Infrastructure Engineering*, *29*(5), 359–381.

Chen, X., Zhu, Z., He, X., & Zhang, L. (2015). Surrogate-based optimization for solving a mixed integer network design problem. *Transportation Research Record*, *2497*(1), 124–136.

Cheng, Q., Wang, S., Liu, Z., & Yuan, Y. (2019). Surrogate-based simulation optimization approach for day-to-day dynamics model calibration with real data. *Transportation Research Part C: Emerging Technologies*, *105*, 422–438.

Chong, L., & Osorio, C. (2017). A simulation-based optimization algorithm for dynamic large-scale urban transportation problems. *Transportation Science*, *52*(3), 637–656.

Das, S., Mullick, S. S., & Suganthan, P. N. (2016). Recent advances in differential evolution—An updated survey. *Swarm and Evolutionary Computation*, *27*, 1–30.

Dharia, A., & Adeli, H. (2003). Neural network model for rapid forecasting of freeway link travel time. *Engineering Applications of Artificial Intelligence*, *16*(7), 607–613.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In T. G. Dietterich (Ed.), *International workshop on multiple classifier systems*, Springer, Berlin, Heidelberg (pp. 1–15).

Elmachtoub, A. N., & Grigas, P. (2021). Smart "predict, then optimize." *Management Science*, *68*(1), 9–26.

Forrester, A. I. J., & Keane, A. J. (2009). Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, *45*(1), 50–79.

Fort, S., Hu, H., & Lakshminarayanan, B. (2019). *Deep ensembles: A loss landscape perspective*. arXiv preprint arXiv:1912.02757. https://arxiv.org/abs/1912.02757

Gaspar, B., Teixeira, A. P., & Guedes Soares, C. (2017). Adaptive surrogate model with active refinement combining Kriging and a trust region method. *Reliability Engineering & System Safety*, *165*, 277–291.

Goel, T., Haftka, R. T., Shyy, W., & Queipo, N. V. (2007). Ensemble of surrogates. *Structural and Multidisciplinary Optimization*, *33*(3), 199–216.

Gong, Y., Zhong, S., Zhao, S., Xiao, F., Wang, W., & Jiang, Y. (2024). Optimizing green splits in high-dimensional traffic signal control with trust region Bayesian optimization. Computer-Aided Civil and Infrastructure Engineering. Advance online publication. https://doi.org/10.1111/mice.13293

Gora, P., Brzeski, M., Karnas, K., Przybyszewski, P., Klemenko, A., Kochański, A., Dryja, H., Kukawska, M., Kopczyk, D., & Możejko, M. (2019). Solving traffic signal setting problem using machine learning. *2019 6th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, Kraków, Poland (pp. 1–10).

Gora, P., Brzeski, M., Możejko, M., Klemenko, A., & Kochański, A. (2018). *Investigating performance of neural networks and gradient boosting models approximating microscopic traffic simulations in traffic optimization tasks*. arXiv preprint arXiv:1812.00401. https://arxiv.org/abs/1812.00401

Gu, S., Yang, L., Du, Y., Chen, G., Walter, F., Wang, J., Yang, Y., & Knoll, A. (2022). *A review of safe reinforcement learning: Methods, theory and applications*. arXiv preprint arXiv:2205.10330. https://arxiv.org/abs/2205.10330

Gupta, R., & Zhang, Q. (2024). Data-driven decision-focused surrogate modeling. *AIChE Journal*, *70*(4), e18338.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile (pp. 1026–1034).

Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J. E., & Weinberger, K. Q. (2017). *Snapshot ensembles: Train 1, get M for free*. arXiv preprint arXiv:1704.00109. https://arxiv.org/abs/1704.00109

Ito, H., Tsutsumida, K., Matsubayashi, T., Kurashima, T., & Toda, H. (2019). Coordinated traffic signal control via Bayesian optimization for hierarchical conditional spaces. *2019 Winter Simulation Conference (WSC)*, National Harbor, MD (pp. 3645–3656).

Jiang, X., & Adeli, H. (2005). Dynamic wavelet neural network model for traffic flow forecasting. *Journal of Transportation Engineering*, *131*(10), 771–779.

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980. https://arxiv.org/abs/1412.6980

Kudela, J., & Matousek, R. (2022). Recent advances and applications of surrogate models for finite element method computations: A review. *Soft Computing*, *26*(24), 13709–13733.

Leandro, A., & Luque, G. (2023). Optimization of traffic light cycles using genetic algorithms and surrogate models. In A. Abraham, T.-P. Hong, K. Kotecha, K. Ma, P. Manghirmalani Mishra, & N. Gandhi (Eds.), *Hybrid intelligent systems* (pp. 607–617). Springer Nature Switzerland.

Li, J. Y., Zhan, Z. H., Wang, H., & Zhang, J. (2021). Data-driven evolutionary algorithm with perturbation-based ensemble surrogates. *IEEE Transactions on Cybernetics*, *51*(8), 3925–3937.

Li, P., Abbas, M. M., Pasupathy, R., & Head, L. (2010). Simulation-based optimization of maximum green setting under

retrospective approximation framework. *Transportation Research Record*, *2192*(1), 1–10.

Li, Z., Tian, Y., Sun, J., Lu, X., & Kan, Y. (2022). Simulation-based optimization of large-scale dedicated bus lanes allocation: Using efficient machine learning models as surrogates. *Transportation Research Part C: Emerging Technologies*, *143*, 103827.

Liang, Y., Ren, Z., Wang, L., Liu, H., & Du, W. (2021). Surrogate-assisted cooperative signal optimization for large-scale traffic networks. *Knowledge-Based Systems*, *234*, 107542.

Little, J. D., Kelson, M. D., & Gartner, N. M. (1981). MAXBAND: A program for setting signals on arteries and triangular networks. *Transportation Research Record*, (795), 40–46.

Liu, H., Ong, Y. S., Shen, X., & Cai, J. (2020). When Gaussian process meets big data: A review of scalable GPS. *IEEE Transactions on Neural Networks and Learning Systems*, *31*(11), 4405–4423.

Long, M., Zou, X., Zhou, Y., & Chung, E. (2022). Deep reinforcement learning for transit signal priority in a connected environment. *Transportation Research Part C: Emerging Technologies*, *142*, 103814.

Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y. P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., & Wiessner, E. (2018). Microscopic traffic simulation using SUMO. *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, HI (pp. 2575–2582).

Melville, P., & Mooney, R. J. (2005). Creating diversity in ensembles using artificial data. *Information Fusion*, *6*(1), 99–111.

Morris, M. D., & Mitchell, T. J. (1995). Exploratory designs for computational experiments. *Journal of Statistical Planning and Inference*, *43*(3), 381–402.

Noaeen, M., Naik, A., Goodman, L., Crebo, J., Abrar, T., Abad, Z. S. H., Bazzan, A. L. C., & Far, B. (2022). Reinforcement learning in urban network traffic signal control: A systematic literature review. *Expert Systems with Applications*, *199*, 116830.

Opitz, D., & Shavlik, J. (1995). Generating accurate and diverse members of a neural-network ensemble. *Advances in Neural Information Processing Systems*, *8*, Denver, CO.

Osorio, C., & Atasoy, B. (2021). Efficient simulation-based toll optimization for large-scale networks. *Transportation Science*, *55*(5), 1010–1024.

Osorio, C., & Bierlaire, M. (2009). A surrogate model for traffic optimization of congested networks: an analytic queueing network approach (Technical report TRANSP-OR090825). Transport and Mobility Laboratory, ENAC, EPFL.

Osorio, C., & Bierlaire, M. (2013). A simulation-based optimization framework for urban transportation problems. *Operations Research*, *61*(6), 1333–1345.

Osorio, C., & Chong, L. (2015). A computationally efficient simulation-based optimization algorithm for large-scale urban transportation problems. *Transportation Science*, *49*(3), 623–636.

Osorio, C., & Nanduri, K. (2015). Energy-efficient urban traffic management: A microscopic simulation-based approach. *Transportation Science*, *49*(3), 637–651.

Pang, T., Xu, K., Du, C., Chen, N., & Zhu, J. (2019). Improving adversarial robustness via promoting ensemble diversity. *International Conference on Machine Learning, PMLR*, Long Beach, CA (pp. 4970–4979).

Pohlmann, T., & Friedrich, B. (2010). Online control of signalized networks using the cell transmission model. *13th International IEEE Conference on Intelligent Transportation Systems*, Funchal, Portugal (pp. 1–6).

Sagi, O., & Rokach, L. (2018). Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery*, *8*(4), e1249.

Schobi, R., Sudret, B., & Wiart, J. (2015). Polynomial-chaos-based Kriging. *International Journal for Uncertainty Quantification*, *5*(2), 171–193.

Stevanovic, A., Martin, P. T., & Stevanovic, J. (2007). Vissim-based genetic algorithm optimization of signal timings. *Transportation Research Record*, *2035*(1), 59–68.

Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering & System Safety*, *93*(7), 964–979.

Sudret, B., & Der Kiureghian, A. (2000). *Stochastic finite element methods and reliability: A state-of-the-art report*. Department of Civil and Environmental Engineering, University of California.

Sundar, V. S., & Shields Michael, D. (2019). Reliability analysis using adaptive Kriging surrogates with multimodel inference. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, *5*(2), 04019004.

Tang, K., Cao, Y., Chen, C., Yao, J., Tan, C., & Sun, J. (2021). Dynamic origin-destination flow estimation using automatic vehicle identification data: A 3D convolutional neural network approach. *Computer-Aided Civil and Infrastructure Engineering*, *36*(1), 30–46.

Tang, Z., Yao, J., Tang, K., & Chung, E. (2023). *LMP-Band: A linked movement-pair progression model for urban arterial signal coordination control*. SSRN 4341456. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4341456

Tao, J., & Sun, G. (2019). Application of deep learning based multi-fidelity surrogate model to robust aerodynamic design optimization. *Aerospace Science and Technology*, *92*, 722–737.

Tay, T., & Osorio, C. (2022). Bayesian optimization techniques for high-dimensional simulation-based transportation problems. *Transportation Research Part B: Methodological*, *164*, 210–243.

van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(11), 2579–2605.

Varaiya, P. (2013). Max pressure control of a network of signalized intersections. *Transportation Research Part C: Emerging Technologies*, *36*, 177–195.

Wang, J., Clark, S. C., Liu, E., & Frazier, P. I. (2020). Parallel Bayesian global optimization of expensive functions. *Operations Research*, *68*(6), 1850–1865.

Wang, Y., Zhang, D., Liu, Y., Dai, B., & Lee, L. H. (2019). Enhancing transportation systems via deep learning: A survey. *Transportation Research Part C: Emerging Technologies*, *99*, 144–163.

Webster, F. V. (1958). *Traffic signal settings* (Technique Paper No. 39). Road Research Laboratory, London.

Wei, H., Zheng, G., Gayah, V., & Li, Z. (2021). Recent advances in reinforcement learning for traffic signal control: A survey of models and evaluation. *SIGKDD Explorations Newsletter*, *22*(2), 12–18.

Wu, X., & Liu, H. X. (2011). A Shockwave profile model for traffic flow on congested urban arterials. *Transportation Research Part B: Methodological*, *45*(10), 1768–1786.

Zahura, F. T., Goodall, J. L., Sadler, J. M., Shen, Y., Morsy, M. M., & Behl, M. (2020). Training machine learning surrogate models from a high-fidelity physics-based model: Application for real-time street-scale flood prediction in an urban coastal community. *Water Resources Research*, *56*(10), e2019WR027038.

Zhang, M.-L., & Zhou, Z.-H. (2013). Exploiting unlabeled data to enhance ensemble diversity. *Data Mining and Knowledge Discovery*, *26*(1), 98–129.

Zhang, R., Ishikawa, A., Wang, W., Striner, B., & Tonguz, O. K. (2021). Using reinforcement learning with partial vehicle detection for intelligent traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, *22*(1), 404–415.

Zhang, S., Liu, M., & Yan, J. (2020). The diversified ensemble neural network. *Advances in Neural Information Processing Systems*, *33*, Virtual (pp. 16001–16011).

Zheng, L., Bao, J., & Mei, Z. (2023). Urban traffic signal control robust optimization against risk-averse and worst-case cyberattacks. *Information Sciences*, *640*, 119067.

Zheng, L., & Li, X. (2023). Simulation-based optimization method for arterial signal control considering traffic safety and efficiency under uncertainties. *Computer-Aided Civil and Infrastructure Engineering*, *38*(5), 640–659.

Zheng, L., Xue, X., Xu, C., & Ran, B. (2019). A stochastic simulation-based optimization method for equitable and efficient network-wide signal timing under uncertainties. *Transportation Research Part B: Methodological*, *122*, 287–308.

Zhou, T., Wang, S., & Bilmes, J. A. (2018). Diverse ensemble evolution: Curriculum data-model marriage. *Advances in Neural Information Processing Systems*, *31*, Montréal, Canada.

Zhou, Z.-H., Wu, J., & Tang, W. (2002). Ensembling neural networks: Many could be better than all. *Artificial Intelligence*, *137*(1), 239–263.

Zou, X., & Chung, E. (2024). Traffic prediction via clustering and deep transfer learning with limited data. *Computer-Aided Civil and Infrastructure Engineering*, *39*(17), 2683–2700.

Zou, X., Chung, E., Zhou, Y., Long, M., & Lam, W. H. K. (2024). A feature extraction and deep learning approach for network traffic volume prediction considering detector reliability. *Computer-Aided Civil and Infrastructure Engineering*, *39*(1), 102–119.

How to cite this article: Tang, Z., Wang, R., Chung, E., Gu, W., & Zhu, H. (2025). An adversarial diverse deep ensemble approach for surrogate-based traffic signal optimization. *Computer-Aided Civil and Infrastructure Engineering*, *40*, 632–657. https://doi.org/10.1111/mice.13354

APPENDIX A: DE ALGORITHM

DE algorithm is a prevalent evolutionary algorithm utilized extensively in global optimization tasks. The critical steps in DE are the operations of mutation and crossover. The particular mutation strategy in this study is presented in Equation (A1). For any individual $\boldsymbol{x}_{i,G}$ in the Gth population, its mutant vector $\boldsymbol{v}_{i,G}$ is calculated as

$$\boldsymbol{v}_{i,G} = \boldsymbol{x}_{i,G} + F\left[\left(\boldsymbol{x}_{\text{best},G} - \boldsymbol{x}_{i,G}\right) + \sum_{j=1}^{J} \eta_j \left(\boldsymbol{x}_{r_j,G} - \boldsymbol{x}_{i,G}\right)\right] \tag{A1a}$$

TABLE A1 Algorithm 3: DE.

Outer loop

Input: Surrogate model f_{SM^k}. Scaling factor F. Crossover probability p_c. Population size N_p. Maximal number of generations $G_{\max}$.

Initiate: $G = 0$. Random population $\{\boldsymbol{x}_{i,G}\}_{i=1}^{N_p}$.

While $G \le G_{\max}$:

  Update $\{\boldsymbol{x}_{i,G}\}_{i=1}^{N_p}$ according to the inner loop.

  $G \leftarrow G + 1$.

End While

$X_{\text{all}} = \{\boldsymbol{x}_{i,G} | i = 1,2,\ldots,N_p; G = 0,1,\ldots,G_{\max}\}$.

$\boldsymbol{x}_{\text{ext}}^{k} = \underset{\boldsymbol{x}\in X_{\text{all}}}{\text{argmin}}\, f_{\text{SM}^k}(\boldsymbol{x})$.

Output $\boldsymbol{x}_{\text{ext}}^{k}$.

Inner loop

Input: Last population $\{\boldsymbol{x}_{i,G}\}_{i=1}^{N_p}$.

#Mutation

Obtain $\{\boldsymbol{v}_{i,G}\}_{i=1}^{N_p}$ according to Equation (A1).

Crossover

Obtain $\{\boldsymbol{u}_{i,G}\}_{i=1}^{N_p}$ according to Equation (A2).

Evaluate

Obtain $\{f_{\text{SM}^k}(\boldsymbol{u}_{i,G})\}_{i=1}^{N_p}$ and $\{f_{\text{SM}^k}(\boldsymbol{x}_{i,G})\}_{i=1}^{N_p}$.

Selection

$$\boldsymbol{x}_{i,G+1} = \begin{cases} \boldsymbol{u}_{i,G} & \text{if } f_{\text{SM}^k}(\boldsymbol{u}_{i,G}) \le f_{\text{SM}^k}(\boldsymbol{x}_{i,G}) \\ \boldsymbol{x}_{i,G} & \text{otherwise} \end{cases}, \forall i.$$

Output $\{\boldsymbol{x}_{i,G+1}\}_{i=1}^{N_p}$.

$$\eta_j = \begin{cases} 1, & \text{if } f_{\text{SM}^k}\left(\boldsymbol{x}_{r_j,G}\right) \le f_{\text{SM}^k}\left(\boldsymbol{x}_{i,G}\right) \\ -1, & \text{otherwise} \end{cases} \tag{A1b}$$

where $\boldsymbol{x}_{\text{best},G}$ is the optimal individual in the Gth population, $\{r_j\}_{j=1}^{J}$ are mutually exclusive random integers within population size. In this study, $J = 2$. F is a scaling factor, adjusting the contribution of the differential terms to $\boldsymbol{v}_{i,G}$. Crossover operation is presented in Equation (A2):

$$\boldsymbol{u}_{i,G}^{k} = \begin{cases} \boldsymbol{v}_{i,G}^{k}, & \text{if rand}(0,1) < p_c \\ \boldsymbol{x}_{i,G}^{k}, & \text{otherwise} \end{cases}, \forall k \tag{A2}$$

where p_c is the crossover probability, and k is the component index. The components of $\boldsymbol{x}_{i,G}$ are defined by Equation (1c).

The outer and inner loops of DE algorithm are presented in Table A1. In this study, $F = 0.5$, $p_c = 0.5$, $N_p = 50$, $G_{\max} = 30$.

APPENDIX B: CALCULATION METHOD OF INDICES

The calculation of eMAPE, bMAPE, and RSD is as Equation (B1) indicates.

$$\text{eMAPE} = \frac{100\%}{R'} \sum_{i=1}^{R'} \left|\frac{y - \hat{y}}{y}\right| \tag{B1a}$$

$$\text{bMAPE} = \frac{100\%}{NR'} \sum_{n=1}^{N} \sum_{i=1}^{R'} \left| \frac{y - \hat{y}_n}{y} \right| \tag{B1b}$$

$$\text{RSD} = \frac{100\%}{R'} \sum_{i=1}^{R'} \frac{\sqrt{\frac{1}{N} \sum_{n=1}^{N} (\hat{y}_n - \hat{y})^2}}{\hat{y}} \tag{B1c}$$

where y is the true label of test data, $\hat{y}_n$, $n = 1, 2, \ldots, N$, are base learners' predictions. $\hat{y}$ is the ensemble's prediction, obtained by Equation (3). R' is the number of test data. Subscript i is the index of y, $\hat{y}$, and $\hat{y}_n$ in their respective test set. For notational simplicity, the subscript i is omitted.

APPENDIX C: A GLOBAL EXTREME POINT OF ADE-0

The global extreme point of ADE-0 in Figure 7a is shown in Table C1. The dimension of θ_3 is the 1d local area in Figure 7.

TABLE C1 The global extreme point of ADE-0.

Variable	Value
c	101
$\boldsymbol{g}_1, \boldsymbol{g}_2, \boldsymbol{g}_3, \boldsymbol{g}_4$	{20, 69, 12}, {44, 16, 41}, {12, 12, 77}, {89, 12}
s_1, s_2, s_3, s_4	{0, 2, 1}, {0, 1, 2}, {0, 1, 2}, {0,1}
$\theta_1, \theta_2, \theta_3, \theta_4$	0, 0, 0, 10