A Distributed Gradient Approach for System Optimal Dynamic Traffic Assignment

Mehrzad Mehrabipour and Ali Hajbabaie, *Member, IEEE*

***Abstract*— This study presents a distributed gradient-based approach to solve system optimal dynamic traffic assignment (SODTA) formulated based on the cell transmission model. The algorithm distributes SODTA into local sub-problems, who find optimal values for their decision variables within an intersection. Each sub-problem communicates with its immediate neighbors to reach a consensus on the values of common decision variables. A sub-problem receives proposed values for common decision variables from all adjacent sub-problems and incorporates them into its own offered values by weighted averaging and enforcing a gradient step to minimize its objective function. Then, the updated values are projected onto the feasible region of the sub-problems. The algorithm finds high quality solutions in all tested scenarios with a finite number of iterations. The algorithm is tested on a case study network under different demand levels and finds solutions with at most a 5% optimality gap.**

***Index Terms*— Distributed, system optimal, dynamic traffic assignment, sub-problem, decomposition.**

I. Introduction

DYNAMICc Traffic Assignment (DTA) is a well-studied research area to determine time-dependent traffic flows by minimizing the cost of the system or individual users in transportation networks. DTA deployment has brought many benefits to a wide range of applications over the past decades. Network design [1]–[4], traffic operations [5], congestion pricing [6], evacuation planning [7], and traffic management systems [8] are some of DTA applications.

DTA with accurate network loading models has many decision variables and constraints to encompass its spatial/temporal scales and the number of origin-destination (OD) pairs. The number of decision variables and constraints can further increase based on the utilized network loading concept. For instance, a cell transmission model (CTM)-based DTA will have more decision variables and constraints than a link performance function-based DTA, as CTM divides each link into several shorter links, which increases the number of decision variables.

Centralized approaches solve the optimization model with no decomposition, distribution, or parallelism. Central optimization frameworks do not scale with the size of DTA, especially when complex network loading concepts, such as CTM are in use [5], [9]–[12]. Decomposition approaches convert the model into sub-problems and a master problem. Each subproblem has less computational complexity in comparison with the original model because the subproblem contains only a fraction of the variables and constraints of the original model. Therefore, decomposition approaches can improve scalability; however, non-spatial decomposition approaches face computational complexity growth in the sub-problems as the size of the original problem grows. Besides, all existing decomposition techniques require solving the master problem (a central component) to aggregate the solutions found by sub-problems and expanding the network will increase the complexity of the master problem as well. Therefore, while decomposition approaches can provide optimality bounds and scale better than the centralized approaches, they eventually become intractable when the size of the problem grows [13]–[18].

This paper presents a distributed gradient-based approach (DGA) to overcome the discussed drawbacks of existing decomposition approaches. The proposed methodology distributes network-level DTA into several intersection-level DTA sub-problems. Constraints and the objective function are exclusively distributed among sub-problems.

It allocates a computational node to each sub-problem and does not require a central component or a master problem. Therefore, it scales better than the decomposition algorithms, whose complexity depend on the network size. The computational complexity of sub-problems does not depend on the number of nodes and links in this approach as a result of the spatial decomposition; however, it is a function of the number of OD pairs. This intersection-level distribution is well-suited for urban network planning purposes because extending network by adding more intersections (nodes and links) will not change the computational complexity of the methodology and its structure.

The approach starts with setting initial values for decision variables. These values can be found by simulating the network with a path-based CTM and sending the demand through the shortest paths found by Dijkstra's algorithm. Then, the approach performs three steps to update the value of decision variables iteratively at each sub-problem: (1) it first incorporates the value of common decision variables among sub-problems by taking a weighted average, (2) the approach adjusts the value of the decision variables from the first step by moving them towards the negative direction of the gradient of the objective function at each sub-problem to minimize the objective, and (3) it projects the values of step 2 onto the set

Manuscript received October 25, 2020; revised May 13, 2021, September 13, 2021, and February 18, 2022; accepted March 3, 2022. The Associate Editor for this article was S. C. Wong. *(Corresponding author: Ali Hajbabaie.)*

The authors are with the Department of Civil, Construction, and Environmental Engineering, North Carolina State University, Raleigh, NC 27695 USA (e-mail: mmehrab2@ncsu.edu; ahajbab@ncsu.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TITS.2022.3163369, provided by the authors.

Digital Object Identifier 10.1109/TITS.2022.3163369

of constraints of each sub-problem to maintain feasibility. The approach iterates among these three steps until the values of shared decision variables from sub-problems reach consensus: there is an acceptable disagreement on the generated values of the decision variables. The sub-problems communicate based on a hypothetical information exchange graph that consists of nodes and directional links. Each node and link represent a sub-problem and the communication between sub-problems, respectively. A weight is assigned to each link for incorporating information among sub-problems. The approach uses these weights for taking a weighted average in the first step.

It can be shown theoretically that the algorithm eventually converges to the optimal solution of the problem. Note that the approach can work without necessarily starting with a feasible solution. Furthermore, the approach is applicable to problems with nonlinear and quadratic objective functions since it needs either the gradient or sub-gradients of the objective function.

In the remainder of this paper, we present a review of relevant literature to solve SODTA problems. Then, we describe the problem formulation and the solution technique in separate sections. The discussion is continued by presenting a case study network and numerical results. Finally, the concluding remarks and directions for future research are presented.

II. Literature Review

Ziliaskopoulos [9] formulated SODTA with one destination using CTM as a linear and convex program. He solved the program centrally using the Simplex method for a small network with 10 cells. Beard and Ziliaskopoulos [5] further improved this formulation to jointly optimize signal timing parameters and traffic assignment considering multiple OD pairs. This formulation was a mixed-integer linear program and was solved centrally for only two intersections. Chiu and Zheng [10] also modified the CTM-based SODTA to address emergency and disaster conditions. They presented a linear formulation with multiple OD pairs and optimized it for a simplified network of 40 cells using the interior point method to find paths and departure schedules for muti-priority groups. Aziz and Ukkusuri [12] proposed a non-linear and quadratic CTM-based SODTA formulation to minimize emission and travel time with a single destination. They solved the model centrally for a network with 88 cells. Zheng and Chiu [11] proposed a network flow algorithm to solve Ziliaskopoulos's [9] formulation and overcome the scalability issue of the previous centralized approaches. They first proved that SODTA was equivalent to the earliest arrival flow problem. Then, they created a time-expanded network to solve the earliest arrival flow problem. They applied the algorithm to a network with 2012 cells. Even though the algorithm was more efficient than the previous techniques, it was restricted to single destination problems since the earliest arrival flow problem was not equivalent to SODTA for multiple OD pairs. Therefore, the algorithm was not applicable to SODTA with multiple OD pairs. All these approaches used a centralized solution approach, which did not scale well with temporal and spatical scales of the CTM-based traffic assignment problems. Therefore, they were applicable to smaller-size problems with a limited amount of available memory and time.

Long *et al.* [19] showed that the link-based formulation was more computationally efficient compared to path-based counterparts. However, solving the problem for larger cases still required more efficient algorithms. They presented various formulations to model SODTA using Link Transmission Mode (LTM) to optimize route and departure choice decisions. Different combinations of models including first-in-first-out (FIFO) and non-holding-back constraints were presented. The problems without FIFO constraints were solved using a commercial solver and those with FIFO constraints were optimized using the branch and bound algorithm for a network of twelve nodes. Long and Szeto [20] also studied LTM for the SODTA problem considering FIFO conditions. Vehicle holding back was also addressed using mixed-integer constraints. The proposed formulation with FIFO constraints was a mixed-integer linear program that had more computational complexity compared to linear problems. The problem was solved using a branch and bound algorithm via commercial software.

Decomposition approaches were developed to decrease the computational complexity of transportation problems [21]–[24] and SODTA. Li *et al.* [16] presented a decomposition algorithm based on the Dantzig-Wolfe principle to solve Ziliaskopoulos's [9] formulation. The algorithm consisted of a sub-problem and a master problem. The sub-problem generated link flows using a minimum-cost-flow problem, and the master problem found the best convex combination of all generated flows. The number of decision variables in the master problem increases over iterations, and the computational complexity of the sub-problems and the master problem increase by adding more nodes and links to the network. Moreover, the algorithm cannot be applied to cases where the ratio of free-flow speed to backward propagation speed is not an integer value. Lin *et al.* [15] also used the Dantzig-Wolfe principle to develop a heuristic-based decomposition algorithm to calibrate the flow capacity of a User Equilibrium (UE) DTA with a single destination and CTM traffic dynamics. They created sub-problems by decomposing the dual formulation for bi-level capacity calibration problem. The master problem considered the relaxed constraints in sub-problems and combined their solutions. Similar to Li *et al.*'s [16] study, the growth in the computational complexity of the master problem and sub-problems is unavoidable. Ramadurai and Ukkusuri [25] proposed an approach with acyclic sub-networks for ODs and activity sequences to solve a user equilibrium DTA problem with a point queue model. The route choice, location, time, and duration of an activity were decision variables. In the approach, sub-network equilibrium was achieved by changing the flow of paths. The optimality condition was not guaranteed in this simulation-based approach.

Jafari *et al.* [14] decomposed the UE static traffic assignment problem spatially by creating sub-networks to represent sub-problems. The sub-problems found the traffic assignment solution within a sub-network given the regional demand from a master problem. The master problem solved the static traffic assignment problem for an aggregated network using modified travel times obtained from the sub-problems. They needed to create an aggregated network with arbitrary arcs that contained all paths based on the original network. They applied a method of sensitivity analysis by Jafari and Boyles [26] to find required parameters for arbitrary links and relative changes in path flows by demand fluctuations at each iteration. The approach by Boyles [27] provided the basis for network aggregation to create the master problem and model sub-networks. Boyles's [27] method could find the cost function of arbitrary arcs in the aggregated network as well. In Jafari *et al.*'s [14] approach, considering more sub-networks leads to increasing the complexity of the aggregated network and the master problem. Moreover, the algorithm is restricted to acyclic networks and requires an initial feasible solution that satisfies UE conditions.

Mehrabipour *et al.* [13] developed an OD-based decomposition approach using the Dantzig-Wolfe principle to overcome some drawbacks in previous approaches. The approach solved a CTM-based SODTA problem and consisted of a master problem and sub-problems. Each sub-problem found cell flows for one OD pair, and the master problem combined the generated flows considering the unseen constraints in sub-problems on total flow over all OD pairs. This approach increased the scalability of centralized approaches while guaranteeing optimality conditions. However, the computational complexity of the master problem grew with the number of iterations. Besides, the computational complexity of the master and sub-problems increased by adding more nodes and links to the network.

A. The Summary of Contributions

The contributions of this paper are as follow:

1. The proposed approach is completely distributed. The sub-problems work cooperatively without requiring a central optimization unlike existing decomposition approaches developed in [13]–[18], [28]–[30] Therefore, the proposed approach is scalable with the size of the network.
2. The approach finds high quality solutions to the SODTA problem with a reasonable number of iterations. The required assumptions for convergence do not restrict the application of the approach to specific network properties such as limiting the ratio of free-flow speed to backward propagation wave speed [16] and having specific network geometry [14], [18].
3. The computational complexity of sub-problems is independent of the number of nodes and links in the network since we create each sub-problem by a spatial distribution of the objective function and the set of constraints unlike the previous approaches: (e.g., [14]–[18], [28]–[30]).

III. Problem Formulation

SODTA is formulated as a linear program utilizing the CTM traffic dynamics introduced by Daganzo [31], [32]. In cell transmission model, a road is discretized into homogenous segments called cells. Homogenous cells have the same length and free-flow speed. Also, the study period is divided into time intervals named time steps. The length of each cell is equivalent to the distance that a vehicle can travel in one-time step. Note that CTM relates flow and density in each cell using non-linear equations. Some examples of formulations based on CTM can be found in [33]–[36]. CTM can also be extended to model transit flows [37] and energy consumption in transportation networks [38]. We used the linearized form of the CTM-based SODTA formulation introduced by Beard and Ziliaskopoulos [5] and modified the set of OD pairs to reduce the computational complexity. The cell-based constraints developed in [39] can be incorporated in our formulation to resolve priority issues at merge cells in the future.

TABLE V (see Appendix) presents the sets, decision variables, parameters, notation, and terms used throughout the paper. Let C, T, C_{OD}, and $S(i)$ respectively denote the set of cells, time steps, OD pairs, and cells successor to cell $i \in C$. This formulation includes two sets of decision variables. The first set is the number of vehicles $x_i^{t,od}$ in cell $i \in C$ at time step $t \in T$ associated with OD pair $(o, d) \in C_{OD}$, and the second set is the number of vehicles $y_{ij}^{t,od}$ flowing from cell $i \in C$ to successor cell $j \in S(i)$ at time step $t \in T$ associated with OD pair $(o, d) \in C_{OD}$. Traffic signal timing parameters are input to this formulation, and equation (1) finds the variable saturation flow rate f_i^t at intersection cell $i \in C_I$ for time step $t \in T$ based on the signal timing parameters. We use C_I and F_i to denote the set of intersection cells and constant saturation flow rate of cell $i \in C_I$. The signal status g_i^t is a binary parameter defined for all intersection cells $i \in C_I$ and time steps $t \in T$. When the signal is green, it will be one, and zero otherwise. The variable saturation flow rate is equal to the constant saturation flow rate if the signal in the intersection cell is green.

$$f_i^t = g_i^t F_i \quad \forall t \in T,\ i \in C_I \tag{1}$$

We define the minimization of total travel time as the objective function in expression (2). The total travel time is found by summing the number of all vehicles $x_i^{t,od}$ in all network cells except for sink cells $i \in CC_S$ for all OD pairs $(o, d) \in C_{OD}$ over all time steps $t \in T$ and multiplying the result by τ (i.e., the duration of each time step). We can eliminate the time step duration τ from the objective function since it has a constant value.

$$\text{Min } \sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{t\in T} \sum\nolimits_{i\in C\backslash C_S} \tau x_i^{t,od} \tag{2}$$

Constraints (3), (4), and (5) show the conservation of flow for ordinary, source, and sink cells, respectively. An increase or decrease in the number of vehicles $x_i^{t+1,od} - x_i^{t,od}$ between time steps $t \in T$ and $t+1 \in T$ is equal to the difference of the total inflow and outflow of cell $i \in C$ at time step $t \in T$ for OD pair $(o, d) \in C_{OD}$.

$$\sum\nolimits_{k\in P(i)} y_{ki}^{t,od} - \sum\nolimits_{j\in S(i)} y_{ij}^{t,od} = x_i^{t+1,od} - x_i^{t,od}$$
$$\forall t \in T, \quad i \in C\{C_S, C_O\}, \quad (o, d) \in C_{OD} \tag{3}$$

$$\mathcal{D}_i^{t,od} - \sum\nolimits_{j\in S(i)} y_{ij}^{t,od} = x_i^{t+1,od} - x_i^{t,od}$$
$$\forall t \in T, \quad i \in C_O, \quad (o, d) \in C_{OD} \tag{4}$$

$$\sum\nolimits_{i \in P(j)} y_{ij}^{t,od} = x_j^{t+1,od} - x_j^{t,od} \quad \forall t \in T, \quad j \in C_S, \quad (o,d) \in C_{OD} \tag{5}$$

Constraint (6) limits the total outgoing flow $\sum_{j \in S(i)} y_{ij}^{t,od}$ from cell $i \in C$ to its successor cells $j \in S(i)$ to the occupancy $x_i^{t,od}$ of the cell at time step $t \in T$ with OD pair $(o,d) \in C_{OD}$.

$$\sum\nolimits_{j \in S(i)} y_{ij}^{t,od} \leq x_i^{t,od} \quad \forall t \in T, \ i \in C, \ \ (o,d) \in C_{OD} \tag{6}$$

Constraints (7) and (8) respectively ensure that the total outgoing flow from and the total incoming flow to a cell are limited to the constant saturation flow rate of the cell. We use F_i to denote the constant saturation flow rate of cell $i \in C$.

$$\sum\nolimits_{(o,d) \in C_{OD}} \sum\nolimits_{j \in S(i)} y_{ij}^{t,od} \leq F_i \quad \forall t \in T, \ i \in C \tag{7}$$

$$\sum\nolimits_{(o,d) \in C_{OD}} \sum\nolimits_{i \in P(j)} y_{ij}^{t,od} \leq F_j \quad \forall t \in T, \ j \in C \tag{8}$$

The total incoming flow $\sum_{(o,d) \in C_{OD}} \sum_{i \in P(j)} y_{ij}^{t,od}$ to cell $j \in C$ at time step $t \in T$ should be less than or equal to the available capacity $M_j - \sum_{(o,d) \in C_{OD}} x_j^{t,od}$ of that cell as shown by constraint (9). Note that M_j denotes the maximum number of vehicles that cell $j \in C$ can accommodate. We use δ to denote the ratio of free-flow speed to the backward propagation speed.

$$\sum\nolimits_{(o,d) \in C_{OD}} \sum\nolimits_{i \in P(j)} y_{ij}^{t,od} \leq \delta \left(M_j - \sum\nolimits_{(o,d) \in C_{OD}} x_j^{t,od} \right) \quad \forall t \in T, \quad j \in C \tag{9}$$

Constraint (10) limits the total outgoing flow $\sum_{(o,d) \in C_{OD}} \sum_{j \in S(i)} y_{ij}^{t,od}$ from intersection cell $i \in C_I$ to variable saturation flow rate f_i^t at time step $t \in T$. We sum the flow over all links between cell $i \in C_I$ and its successor cell $j \in S(i)$ over all OD pairs to find the total outflow $\sum_{(o,d) \in C_{OD}} \sum_{j \in S(i)} y_{ij}^{t,od}$. Equation (1) finds the variable saturation flow rate f_i^t given the signal status g_i^t for intersection cell $i \in C_I$ and time step $t \in T$.

$$\sum\nolimits_{(o,d) \in C_{OD}} \sum\nolimits_{j \in S(i)} y_{ij}^{t,od} \leq f_i^t \quad \forall t \in T, \ i \in C_I \tag{10}$$

Constraints (11) and (12) are used to ensure that the decision variables are nonnegative.

$$x_i^{t,od} \geq 0 \quad \forall t \in T, \ i \in C, \ \ (o,d) \in C_{OD} \tag{11}$$

$$y_{ij}^{t,od} \geq 0 \quad \forall t \in T, \ i \in C\{C_S\}, \ j \in S(i), \ (o,d) \in C_{OD} \tag{12}$$

The summary of the proposed optimization model follows:

$$\text{Min} \sum\nolimits_{(o,d) \in C_{OD}} \sum\nolimits_{t \in T} \sum\nolimits_{i \in C \setminus C_S} \tau x_i^{t,od}$$
$$\text{s.t. } (3) - (12)$$

IV. Methodology

We present a distributed gradient-based methodology to solve the SODTA problem. The discussions are continued in five subsections: (1) the distribution of SODTA problem formulation, (2) initialization, (3) the procedure of distributed gradient-based update, (4) termination criteria, and (5) convergence properties.

In the formulation distribution section, we partition the cell-based SODTA formulation among sub-problems with an intersection-level segmentation so that the number of sub-problems is equal to the number of intersections. Each sub-problem contains some parts of the objective function and constraints of the original SODTA formulation that have decision variables corresponding to the cells and links within the intersection assigned to the sub-problem. The sub-problems do not share any constraint, and the objective function is distributed among sub-problems so that the summation of the objective functions is equivalent to the objective function of the original SODTA problem.

In the initialization step, we find initial values for the decision variables of each sub-problem. We start by generating the shortest paths for each OD pair using Dijkstra's algorithm [40] and sending the demand to the network through these paths using a path-based CTM simulation [13], [41]. We initialize the approach with the occupancy and flow values that are the outputs of the simulation. This simulation is a cell transmission model developed by Daganzo (1990) and used to simulate the movement of vehicles across the network. Then, the value of the decision variables of each sub-problem is updated over iterations. The updated value for a variable will be the weighted average of all presented values for that variable. We find the weighted average at the current iteration for all sub-problems using the value of decision variables from either the initialization step or the previous iteration. Then, the values of variables are updated considering the negative direction of the gradient of the objective function of each sub-problem (to minimize it) and project the value of the decision variables on the set of constraints at each sub-problem to maintain feasibility. The approach iterates until the conflict among the proposed values from the sub-problems is within an acceptable threshold. Note that having a unique optimal solution is not guaranteed in our approach. Fig. 1 shows the overall framework of the methodology, and we provide more details about each part of the figure in the rest of this section.

A. The Distribution of SODTA Formulation

The first step of the methodology is to decompose the network-level SODTA problem formulation into intersection-level sub-problems. It is not necessary to perform the decomposition around the intersections. Grouping intersections into one subproblem is also possible. Including more intersections in one sub-problem reduces the number of shared decision variables, which consequently reduces the conflict and the required number of iterations. However, the decision space of each sub-problem will grow substantially, which leads to more computational effort at each iteration. Therefore, even though the approach can find optimal or near-optimal solutions in a fewer number of iterations, each iteration can become more complex with a longer runtime. Each sub-problem will have decision variables associated with one intersection. The summation of the objective functions from each sub-problem is equivalent to the objective function of the original SODTA problem, and the union of the constraints from each sub-problem is equivalent to the

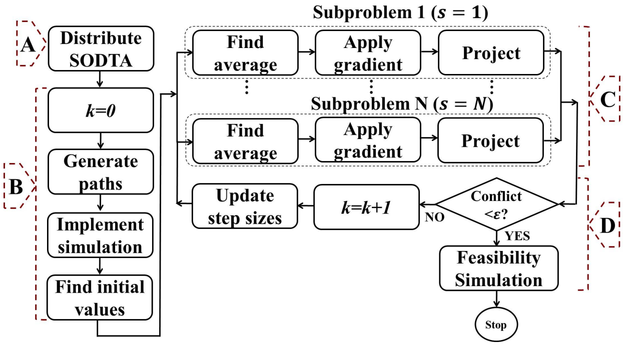


Fig. 1. The framework of the distributed gradient-based methodology.

original constraint set. Note that the sub-problems do not share any constraints. This intersection-level decomposition is well-suited for solving problems in urban street networks since assigning more intersections to the network will not change the architecture of the methodology and its computational complexity. Regional partitioning is used in [7] for static traffic assignment and in [42] for SODTA with route and departure time variables. More general regional partitioning examples can be found in [43]–[45]. The boundary of each intersection should be selected such that each intersection (subproblem) has more or less the same number of decision variables to balance out the run time and reduce overhead delays. A general formulation for sub-problem $s \in N$ is shown below. This formulation differs from the original formulation in the set of cells for which the constraints are defined. The sets are defined in Table V. A formulation for sub-problem $s \in N$ is shown in (13)-(23).

$$\text{Min} \sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{t\in T} \sum\nolimits_{i\in Ces^s} x_i^{t,od} \tag{13}$$

$$\sum\nolimits_{k\in P(i)} y_{ki}^{t,od} - \sum\nolimits_{j\in S(i)} y_{ij}^{t,od} = x_i^{t+1,od} - x_i^{t,od} \quad \forall t \in T,\ \ i \in Ceos^s,\ (o,d) \in C_{OD} \tag{14}$$

$$\mathcal{D}_i^{t,od} - \sum\nolimits_{j\in S(i)} y_{ij}^{t,od} = x_i^{t+1,od} - x_i^{t,od} \quad \forall t \in T,\ \ i \in Co^s,\ (o,d) \in C_{OD} \tag{15}$$

$$\sum\nolimits_{i\in P(j)} y_{ij}^{t,od} = x_j^{t+1,od} - x_j^{t,od} \quad \forall t \in T,\ \ j \in Cs^s, (o,d) \in C_{OD} \tag{16}$$

$$\sum\nolimits_{j\in S(i)} y_{ij}^{t,od} \le x_i^{t,od} \quad \forall t \in T,\ i \in Ces^s,\ (o,d) \in C_{OD} \tag{17}$$

$$\sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{j\in S(i)} y_{ij}^{t,od} \le F_i \quad \forall t \in T,\ i \in Ces^s \tag{18}$$

$$\sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{i\in P(j)} y_{ij}^{t,od} \le F_j \quad \forall t \in T,\ \ j \in Ceo^s \tag{19}$$

$$\sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{i\in P(j)} y_{ij}^{t,od} \le \delta(M_j - \sum\nolimits_{\forall(o,d)\in C_{OD}} x_j^{t,od}) \quad \forall t \in T,\ \ j \in Ceo^s \tag{20}$$

$$\sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{j\in S(i)} y_{ij}^{t,od} \le f_i^t \quad \forall t \in T,\ i \in Ci^s \tag{21}$$

$$x_i^{t,od} \ge 0 \quad \forall t \in T,\ i \in C^s,\ (o,d) \in C_{OD} \tag{22}$$

$$y_{ij}^{t,od} \ge 0 \quad \forall t \in T,\ i \in C^s, j \in S(i),\ (o,d) \in C_{OD} \tag{23}$$

B. Initialization

The approach starts with initial values for all decision variables at the first iteration, see Fig. 1. The initial solutions do not have to be feasible for the original SODTA formulation. We first implement Dijkstra's algorithm [40] to generate the shortest paths for all OD pairs. Note that we assume that the network is under free-flow condition. Then, we use the path-based CTM simulation introduced by Ukkusuri *et al.* [41] to find occupancy $x_i^{t,od}$ for cell $i \in C^s$ at time step $t \in T$ with OD pair $(o,d) \in C_{OD}$ and flow $y_{ij}^{t,od}$ for links between cell $i \in C^s$ and its successor cell $j \in S(i)$ at time step $t \in T$ with OD pair $(o,d) \in C_{OD}$ for all sub-problems $s \in N$.

C. Distributed Gradient Update

This procedure updates the decision variables of all sub-problems in iteration $k+1 \in K$ using three main steps:

Step 1: Each sub-problem optimizes the values of its decision variables. Therefore, decision variables that are in common between several sub-problems will have various values. This step sets the value of these decision variables in each sub-problem equal to their weighted average. The weights in each sub-problem are determined such that they satisfy the required conditions for convergence.

Step 2: The approach moves the computed values for sub-problem $s \in N$ from *Step 1* towards the negative direction of the gradient of the objective function of the corresponding sub-problem to minimize the total travel time within the region assigned to the sub-problem.

Step 3: The approach projects the decision variable values in sub-problem $s \in N$ from *Step 2* onto the set of constraints of the sub-problem to make the values feasible for that sub-problem.

Note that the formulation is not changed for updating the flow and occupancy values. The adjustment on the flow values are handled by the gradient update procedure We describe these steps with a mathematical representation in the rest of this section. We first introduce four definitions for the information exchange graph (Definition 1), neighbors of a sub-problem (Definition 2), the gradient of a function (Definition 3), and the projection operator (Definition 4).

Definition 1 (Information Exchange Graph): The information exchange graph $G = (N, A)$ contains nodes and directional links that belong to sets N and A, respectively. Node $s \in N$ represents a sub-problem, and link $(s, \mathscr{s}) \in A$ shows the transfer of information from sub-problem $s \in N$ to sub-problem $\mathscr{s} \in N$. If there is a variable in common between sub-problems $s, \mathscr{s} \in N$, directional links $(s, \mathscr{s}) \in A$ and $(\mathscr{s}, s) \in A$ are needed. There is also a self-arc at each node i.e., $\{(s, \mathscr{s}) : s = \mathscr{s}, \forall s, \mathscr{s} \in N\}$. This arc represents the use of information in the computation of sub-problem $s \in N$ that is generated by itself. Weight value $\mathscr{w}_{s\mathscr{s}}$ for the link going from node $s \in N$ to node $\mathscr{s} \in N$ is also assigned to link $(s, \mathscr{s}) \in A$ to be used for incorporating information from sub-problem $s \in N$ for decision variables in sub-problem $\mathscr{s} \in N$. Assumption 1 determines the value of weights and is required to prove the convergence of the approach. We can use identical values for wights on all links $(s, \mathscr{s}) \in A$ if $s \neq \mathscr{s}$. Weights on self-arcs $s \in N$ satisfy equation $\mathscr{w}_{ss} = -\sum_{\mathscr{s} \in C^{\mathfrak{n}^s}: s \neq \mathscr{s}} \mathscr{w}_{\mathscr{s}s}$. Numerical examples for the weights are provided for the simple example in this section and the test network in the result section.

Assumption 1: Let us use θ and θ' to show a lower and an upper bound for weights $\mathscr{w}_{s\mathscr{s}}$ on all links $(s, \mathscr{s}) \in N$, respectively. The information exchange graph $G = (N, A)$ and weight $\mathscr{w}_{s\mathscr{s}}$ on link $(s, \mathscr{s}) \in N$ should satisfy the following conditions:

(1) Information exchange graph $G = (N, A)$ should be connected.

(2) Weights must satisfy $\mathscr{w}_{s\mathscr{s}} = \mathscr{w}_{\mathscr{s}s}$ for all $s, \mathscr{s} \in N$

(3) If weight $\mathscr{w}_{s\mathscr{s}} \neq 0$, we have $\theta \leq \mathscr{w}_{s\mathscr{s}} \leq \theta'$ for all $s, \mathscr{s} \in N$ such that $\theta, \theta' > 0$

(4) Weights on self-arcs $s \in N$ must satisfy equation $\mathscr{w}_{ss} = -\sum_{\mathscr{s} \in C^{\mathfrak{n}^s}: s \neq \mathscr{s}} \mathscr{w}_{\mathscr{s}s}$

Definition 2 (The Neighbors of a Sub-Problem): The neighbors of sub-problem $s \in N$ are sub-problems $\mathscr{s} \in N : \mathscr{s} \neq s$ that offer estimation for at least one decision variable of sub-problem $s \in N$. In other words, sub-problem $\mathscr{s} \in N$ is a neighbor of sub-problem $s \in N$ if and only if $\mathscr{w}_{\mathscr{s}s} > 0$. We define the set of neighbors of sub-problem $s \in N$ including itself by $\mathfrak{n}^s$.

In general, any node $s \in N$ may be connected to any other node $\mathscr{s} \in N$ in the information exchange graph G even if the sub-problems (nodes) are not immediate neighbors in the original (physical) network if their corresponding sub-problems share a decision variable. This fact will not affect the distributed structure of the methodology because the approach uses the information from the previous iteration for the exchange process, not the current iteration. However, the structure of SODTA formulation and intersection-based distribution lead to the presence of links only between immediate neighbors in the information exchange graph. The reason is that the immediate neighbors share decision variables corresponding to the links between any two intersections (regions).

Definition 3 (Gradient of a Function): Let $\mathcal{F}_s(\boldsymbol{x})$, $\mathcal{G}_s \in \mathbb{R}^n$, and X_s respectively denote the objective function value of sub-problem $s \in N$ given vector $\boldsymbol{x}$, the gradient of the objective function $\mathcal{F}_s(\boldsymbol{x})$, and the feasible region of sub-problem $s \in N$. The gradient $\mathcal{G}_s$ satisfies inequality (24) for all vectors $\boldsymbol{z}$, $\boldsymbol{x} \in X_s$.

$$\mathcal{F}_s(\boldsymbol{z}) + \mathcal{G}'_s(\boldsymbol{x} - \boldsymbol{z}) \leq \mathcal{F}_s(\boldsymbol{x}) \tag{24}$$

Definition 4 (Projection Operator): We use the projection operator $\boldsymbol{P}_X[\boldsymbol{z}]$ to find the projection of vector $\boldsymbol{z}$ onto a closed convex set X using Euclidean norm as shown in (25).

$$\boldsymbol{P}_X[\boldsymbol{z}] = argmin_{x \in X} \|\boldsymbol{z} - \boldsymbol{x}\| \tag{25}$$

A projection is a linear transformation, and the projection operator is used to map any vector onto a closed convex set. By solving $argmin_{x \in X} \|z - x\|$, we can apply this operator to map vector of $x \in X$ on set X.

We now describe all three steps for iteration $k + 1 \in K$ assuming that the value of occupancy and flow decision variables are available (either from the initialization step or previous iteration $k \in K$ for all sub-problems $s \in N$). We first update the value of decision variables as described in *Steps 1* and *2*. We define auxiliary parameters $\mathscr{p}_i^{t,od} : t \in T, i \in C^s, (o, d) \in C_{OD}$ and $\mathscr{q}_{ij}^{t,od} : t \in T, i \in C^s, j \in S(i), (o, d) \in C_{OD}$ for updating the value of cell occupancy and flow decision variables, respectively. If a decision variable appears in only one sub-problem, we find its auxiliary parameter using either equation (26) or (27). Equations (26) and (27) find the auxiliary parameters for occupancy and flow variables, respectively. Including a weighted average in these equations is not required because the decision variable is optimized exclusively. We only need to move the value of the decision variable from the initialization step or previous iteration $k \in K$ at sub-problem $s \in N$ towards the negative direction of the gradient of the objective function. We denote the gradient of the objective function of sub-problem $s \in N$ by $\boldsymbol{\mathcal{G}}_s$ and the step size by γ^{k+1}. The gradient of objective function of sub-problem $s \in N$ respect to $x_i^{t,od}$ and $y_{ij}^{t,od}$ are shown by $\mathcal{G}_{s,i}^{t,od}$ and $\mathcal{G}_{s,ij}^{t,od}$, respectively.

We use equation (28) for the occupancy variable and (29) for the flow variable to find their auxiliary parameters when a decision variable appears in more than one sub-problem. For instance, the weighted average for decision variable $x_i^{t,od} : t \in T, i \in C^s, (o, d) \in C_{OD}$ at sub-problem $s \in N$ is $\sum_{s, \mathscr{s} \in n^s, j \in C^{n^s}: i=j} \mathscr{w}_{s\mathscr{s}} x_j^{t,od}$, which takes weighted average of the generated values for this variable by itself and its neighbors from previous iteration $k \in K$. Then, the value of $x_i^{t,od}$ generated by sub-problem $s \in N$ at iteration $k \in K$ is added to term $\alpha^{k+1} \sum_{s, \mathscr{s} \in n^s, j \in C^{n^s}: i=j} \mathscr{w}_{s\mathscr{s}} x_j^{t,od} - \gamma^{k+1} \mathcal{G}_s$ to find auxiliary parameter $\mathscr{p}_i^{t,od}$. We also use α^{k+1} to denote the step size used for consensus among decision variables at iteration $k + 1 \in K$. The same approach is used to find other auxiliary parameters at iteration $k + 1 \in K$ as shown in equation (29).

$$\text{If } x_i^{t,od} : t \in T, \quad i \in C^s, \quad (o, d) \in C_{OD} \tag{26}$$

is only in sub-problem $s \in N$:

$$\mathscr{p}_i^{t,od} = x_i^{t,od} - \gamma^{k+1} \mathcal{G}_{s,i}^{t,od}$$

If $y_{ij}^{t,od} : t \in T, i \in C^s, j \in S(i), (o, d) \in C_{OD}$ is only in sub-problem $s \in N$:

$$\mathscr{q}_{ij}^{t,od} = y_{ij}^{t,od} - \gamma^{k+1} \mathcal{G}_{s,ij}^{t,od} \tag{27}$$

If $x_i^{t,od} : t \in T, i \in C^s, (o, d) \in C_{OD}$ is several sub-problems in addition to $s \in N$:

$$\mathscr{p}_i^{t,od} = x_i^{t,od} + \alpha^{k+1} \sum\nolimits_{s, \mathscr{s} \in n^s, j \in C^{n^s}: i=j} \mathscr{w}_{s\mathscr{s}} x_j^{t,od} - \gamma^{k+1} \mathcal{G}_{s,i}^{t,od} \tag{28}$$

If $y_{ij}^{t,od} : t \in T, i \in C^s, j \in S(i), (o, d) \in C_{OD}$ is in several sub-problems in addition to $s \in N$:

$$\mathscr{p}_i^{t,od} = y_{ij}^{t,od} + \alpha^{k+1} \sum\nolimits_{s, s \in n^s, j \in C^{n^s}: i=j} \mathscr{w}_{s\mathscr{s}} y_{ij}^{t,od} - \gamma^{k+1} \mathcal{G}_{s,ij}^{t,od} \tag{29}$$

Then, we project the value of auxiliary parameters onto the constraints set of sub-problem $s \in N$ as discussed in *Step 3* and using the projection operator described in Definition 4. We find new values for decision variables at iteration

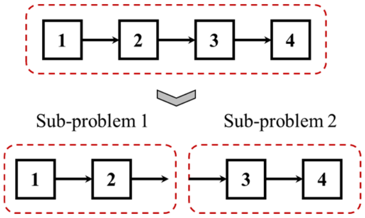


Fig. 2. The distribution of a link with 4 cells to sub-problems 1 and 2.

$k+1 \in K$ by solving the following optimization program for sub-problem $s \in N$.

$$\begin{aligned} \text{Min } & \sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{t\in T} \sum\nolimits_{i\in C^s} \left\| x_i^{t,od} - \mathscr{p}_i^{t,od} \right\|^2 \\ & \quad + \sum\nolimits_{(o,d)\in C_{OD}} \sum\nolimits_{t\in T} \sum\nolimits_{i\in C^s, j\in S(i)} \left\| y_{ij}^{t,od} - \mathscr{q}_{ij}^{t,od} \right\|^2 \\ \text{s.t. } & \text{Constraints (2)-(12) for sub-problem } s \in N \end{aligned} \tag{30}$$

We continue the procedure of updating the value of auxiliary parameters and decision variables for sub-problem $s \in N$ over iterations $k \in K$ until the termination criterion is satisfied.

Equation (31) shows the update procedure in vector notation. The vector of decision variables in sub-problem $s \in N$ at iteration $k+1$ is denoted by $\mathcal{X}_s^{k+1}$, and X_s is the feasible region of sub-problem $s \in N$. Let $\mathscr{w}_{s\mathscr{s}}$ and $\boldsymbol{\mathcal{G}}_s$ denote the weight for link $(s, \mathscr{s}) \in N$ in information exchange graph $G = (N, A)$ and the gradient of the objective function of sub-problem $s \in N$, respectively. Step sizes at iteration $k+1 \in K$ are α^{k+1} and γ^{k+1}.

$$\mathcal{X}_s^{k+1} = P_{X_s}\left[\mathcal{X}_s^k + \alpha^{k+1} \sum\nolimits_{\mathscr{s}\in \mathfrak{n}^s} \mathscr{w}_{\mathscr{s}s} \mathcal{X}_{\mathscr{s}}^k - \gamma^{k+1} \boldsymbol{\mathcal{G}}_s\right] \tag{31}$$

Fig. 2 shows a small network of four cells that is distributed to two sub-problems. We also illustrate the three steps of the update procedure in Fig. 3 using vector notation to visualize this procedure for this simple example. Each sub-problem contains those constraints and parts of the objective function that have the decision variables corresponding to cells and links within the region assigned to that sub-problem. Note that the constraints and objective function can be distributed following a different structure as long as the explained conditions in section A are satisfied. In this simple example, the sub-problems share the decision variables corresponding to the link between cells 2 and 3. Adding more links between cells 2 and 3 will not change the information exchange graph nor the performance of the approach because this one link forces the flow decision variables to appear in both sub-problems, and these sub-problems share information using two directional arcs in the information exchange graph.

Fig. 3.a shows information exchange graph $G = (N, A)$, where $N = \{1, 2\}$ and $A = \{(1, 2), (2, 1), (1, 1), (2, 2)\}$. In Fig. 3. .b-d, each red circle represents the feasible region of one sub-problem, and vector $z^* \in \mathrm{X}^*$ denotes the vector of optimal solutions. Vector $z^* \in \mathrm{X}^*$ is within the feasible region of the original problem as well. The number of variables in common among sub-problems will not change this region. Note that sub-problems do not need to share all decision variables. In other words, if there is at least one variable in common, the sub-problems share information.

The vector of initial values for decision variables for sub-problems 1 and 2 at iteration $0 \in K$ are denoted by $\mathcal{X}_1^0$ and $\mathcal{X}_2^0$, respectively. Fig. 3.b shows the incorporation of the value of decision variables and the computation of weighted average values for sub-problems 1 and 2 using parameters a and b, respectively, as described in *Step 1*. For example, we explain how to compute the value of a. We first multiply weight $\mathscr{w}_{11}$ on self-arc (1, 1) on node 1 in information exchange graph G with the vector of values for decision variables $\mathcal{X}_1^0$ in sub-problem 1 at iteration $0 \in K$, that is $\mathscr{w}_{11}\mathcal{X}_1^0$, and find term $\mathscr{w}_{21}\mathcal{X}_2^0$ the same way. We then find the value of a by summing terms $\mathscr{w}_{11}\mathcal{X}_1^0$ and $\mathscr{w}_{21}\mathcal{X}_2^0$ and multiplying the result by step size α^1. The value of b is found following the same procedure. Fig. 3.c presents Step 2 of the update procedure. We first sum the values of a and b with $\mathcal{X}_1^0$ and $\mathcal{X}_2^0$, respectively. Then, we add $-\gamma^1\mathcal{G}_1$ and $-\gamma^1\mathcal{G}_2$ to the value of $\mathcal{X}_1^0 + a$ and $\mathcal{X}_2^0 + b$ to move them towards the negative direction of the objective function gradient and find c and d, respectively. Fig. 3.d presents the third step, where we project the value of c and d on the feasible region of sub-problems 1 and 2 to find the vector of new values for decision variables $\mathcal{X}_1^1$ and $\mathcal{X}_2^1$ at iteration $1 \in K$, respectively. In the next section, we explain the required assumption for determining the value of step sizes and weights.

D. Convergence Properties

It can be shown that the solution of DGA converges to the optimal solution of the SODTA formulation using similar techniques proposed in [46], [47]. Due to space limitations, we only show the assumptions and the final theorem here.

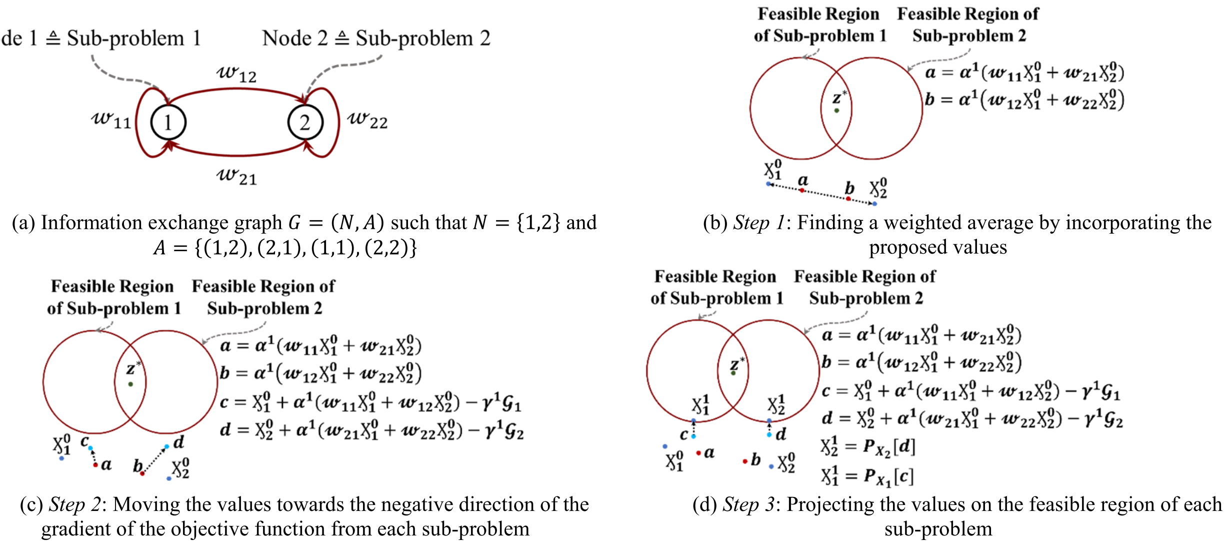
Assumption 2: The step sizes α^k and γ^k should satisfy these conditions: (1) $\sum_{k=1}^{\infty} \alpha^k = \infty$ and $\sum_{k=1}^{\infty} \gamma^k = \infty$, (2) $\sum_{k=1}^{\infty} (\alpha^k)^2 < \infty$ and $\sum_{k=1}^{\infty} (\gamma^k)^2 < \infty$, (3) $\sum_{k=1}^{\infty} (\alpha^k)^2 (\gamma^k)^2 < \infty$, (4) $\sum_{k=1}^{\infty} \frac{(\gamma^k)^2}{\alpha^k} < \infty$, and (5) $\sum_{k=1}^{\infty} \min(\alpha^k, \gamma^k) = \infty$.

Theorem 1: Vectors $\mathcal{X}_s^{k+1}$ generated by sub-problems $s \in N$ using the proposed distributed gradient-based methodology converge to a common optimal vector $z^* \in X^*$, i.e., $\lim_{k\to\gamma} \mathcal{X}_s^{k+1} = z^*$, $\forall s \in N$ if Assumption 1 and Assumption 2 hold.

E. Termination Criterion

The methodology finds the optimal solution when there is no disagreement among the values of decision variables found by different sub-problems. Only some of the flow variables for the boundary links between intersections belong to more than one sub-problem. The optimal solution is found when sub-problems are in agreement on the value of these decision variables, or in other words when the left-hand side of inequality (21) is zero. Since urban streets have a different number of lanes, the value of flow decision variables should be normalized to compute the disagreement by dividing the flow variable by the maximum capacity of receiving or sending cells. We set the termination criterion to reach a disagreement of at most ε.

If the termination criterion is not met, we update step sizes α^{k+1} and γ^{k+1} and go back to the distributed gradient-based


Node 1 ≜ Sub-problem 1
Node 2 ≜ Sub-problem 2
$\mathscr{w}_{12}$
$\mathscr{w}_{11}$
1
2
$\mathscr{w}_{22}$
$\mathscr{w}_{21}$
Feasible Region of Sub-problem 1
Feasible Region of Sub-problem 2
$a = \alpha^1(\mathscr{w}_{11}\mathrm{X}_1^0 + \mathscr{w}_{21}\mathrm{X}_2^0)$
$b = \alpha^1(\mathscr{w}_{12}\mathrm{X}_1^0 + \mathscr{w}_{22}\mathrm{X}_2^0)$
$c = \mathrm{X}_1^0 + \alpha^1(\mathscr{w}_{11}\mathrm{X}_1^0 + \mathscr{w}_{12}\mathrm{X}_2^0) - \gamma^1 \mathcal{G}_1$
$d = \mathrm{X}_2^0 + \alpha^1(\mathscr{w}_{21}\mathrm{X}_1^0 + \mathscr{w}_{22}\mathrm{X}_2^0) - \gamma^1 \mathcal{G}_2$
$\mathrm{X}_2^1 = P_{X_2}[d]$
$\mathrm{X}_1^1 = P_{X_1}[c]$
z^*
X_1^0
X_2^0
X_1^1
X_2^1
a
b
c
d

(a) Information exchange graph $G = (N, A)$ such that $N = \{1,2\}$ and $A = \{(1,2), (2,1), (1,1), (2,2)\}$

(b) *Step 1*: Finding a weighted average by incorporating the proposed values

(c) *Step 2*: Moving the values towards the negative direction of the gradient of the objective function from each sub-problem

(d) *Step 3*: Projecting the values on the feasible region of each sub-problem

Fig. 3. The procedure of distributed gradient-based update for an iteration.

update step.

$$\sum\nolimits_{t\in T}\sum\nolimits_{od\in OD}\sum\nolimits_{i\in C^s, j\in S(i), e\in C^{\mathscr{s}\in n^s}, f\in S(e): i=e, j=f, s\neq \mathscr{s}} \times \left| \frac{y_{ij}^{t,od}}{\max\left(M_i, M_j\right)} - \frac{y_{ef}^{t,od}}{\max\left(M_e, M_f\right)} \right| \leq \varepsilon \quad (32)$$

This termination criterion is checked using a distributed communication paradigm. Each sub-problem checks inequality (32), and if it satisfies the termination criterion, it stops updating the value of variables. The sub-problems compute the termination at the same time. The value of ε does not dependent on the network size since it is determined for each intersection. Finding the feasible solution is possible regardless of the value for ϵ. Moreover, we select the duration of study so that it is long enough for all vehicles can exit the network.

The vector of optimal solutions for the subproblems may not be a feasible solution to SODTA. Therefore, a feasible solution is found using CTM simulation. The input to the CTM is turning ratios that are found using the output of projections on the feasible region of subproblems. The output of the simulation is feasible flows for the entire network.

V. Test Network

We tested DGA on a Nguyen and Dupuis [48] network and portion of the downtown Springfield network in Illinois. We used the modified Nguyen and Dupuis [48] network by Long and Szeto [20]. This network contains 17 nodes, 23 links, and four OD pairs, and its cell representation has 57 cells, 63 links, 4 OD pairs. The network is decomposed into two regions creating two sub-problems in DGA. The two regions have 30 and 27 cells. They also have 36 and 31 links, respectively, from which 4 links are shared and the average number of decision variables in each subproblem is 2700 variables. Similar demand level, jam density, and flow capacity in [19] have been used. The network consists of 20 intersections with one-way and two-way streets, as shown in Fig. 4. All intersections are signalized with predefined signal timing parameters. In this figure, roads are shown with links, and the circles show the intersection number.

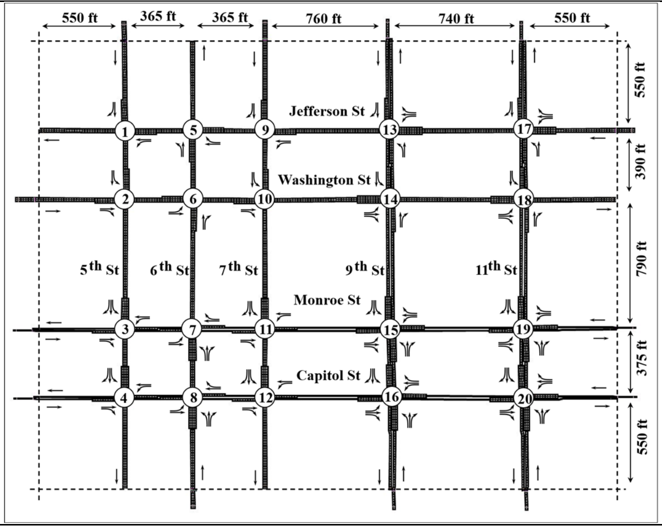
We considered 15 OD pairs and three demand profiles for this test network, representing undersaturated, saturated, and oversaturated conditions. We assumed demand for ODs is known. The proposed technique in [49], [50] may be used to estimate and calibrate the demand in the future. A test network with 40 (4 × 10) intersections and 25 ODs is also tested to show how the approach scales. This network is created by duplicating the network of 20 intersections. We used three demand profiles to represent different saturation levels. According to Assumption 1, one set of values for weights can be $\mathscr{w}_{s\mathscr{s}} = \mathscr{w}_{\mathscr{s}s} = 1$ for all $s, \mathscr{s} \in N$ in which $s \neq \mathscr{s}$ and $\mathscr{w}_{ss} = -\sum\nolimits_{\mathscr{s}\in C^{\mathrm{n}^s}: s\neq \mathscr{s}} \mathscr{w}_{\mathscr{s}s}$ for weights on self-arcs $s \in N$. We also set the step size rules as $\alpha^k = {}^1\!/_{k^{0.55}}$ and $\gamma^k = {}^1\!/_k$, that satisfy Assumption 2. For 20 intersection network, the termination is set to reach a disagreement of 0.5 for each sub-problem. It should be noted that networks with 20 and 40 intersections respectively result in 4,218,000 and 14,120,000 decision variables, representing a large optimization program.

VI. Results

A. Benchmarking the Methodology

1) Comparison With a Central Approach: We applied DGA and a central approach to the case study network of 20 intersections. TABLE I shows the total travel time and the total computation time by our approach and the central approach under three different demand patterns. The SODTA formulation has more than 4 million decision variables in this case study, and we could not find the optimal solution by running CPLEX for cases with less than 150 GB of memory. DGA reduced the decision variables of each sub-problem to 243,600 on average. Each pair of sub-problems shares 6,000 variables for one-directional street and 12,000 variables for two-directional streets. No feasibility occurred in this table.

The case study of 20 intersections –Springfield, IL



CTM characteristics	
The duration of each time step (sec)	6
The number of cells in each link	2,3 4
The total number of cells	316
The total number of links	387
Free-flow speed (mph)	25
Cell length (ft)	220
The capacity of cells except source and sink cells (veh)	9, 24, 36
Saturation flow rate except for source and sink cells (veh/ts/ln)	3,6, 9
Capacity and Saturation flow rate for source & sink cells	1000

sec: second, mph: mile per hour, ft: feet, veh: vehicle, ts: time step

Fig. 4. The case study of 20 intersections, its CTM characteristics, and demand patterns.

TABLE I

OBJECTIVE FUNCTION AND COMPUTATION TIME FOR 20-INTERSECTION NETWORK

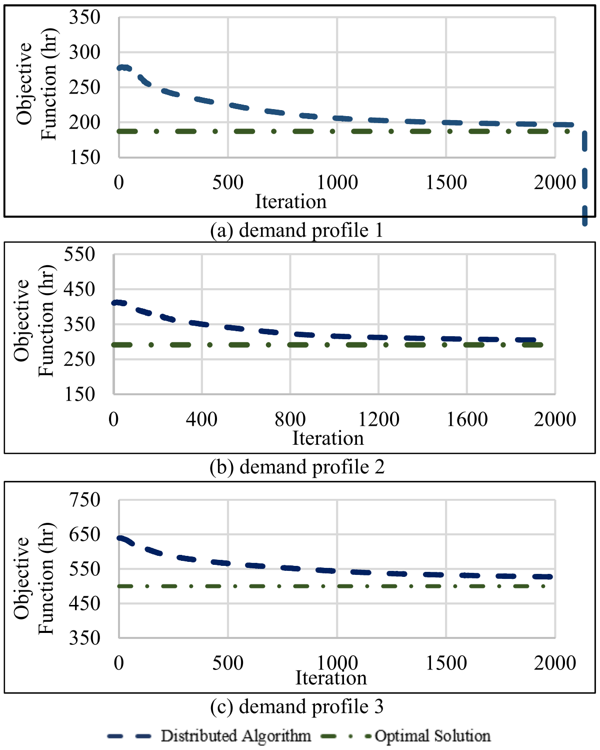
Approach/Gap	Demand Profile	Memory (GB)	Objective Function: Total Travel Time (hr)	Run-time (hr)
CPLEX	1	150	187.06	70.54
DGA	1	5	197.34	1.60
Difference (%)	1	-	5.49	97.73
CPLEX	2	150	291.29	73.59
DGA	2	5	307.30	1.81
Difference (%)	2	-	5.50	97.54
CPLEX	3	150	499.85	90.56
DGA	3	5	524.84	2.01
Difference (%)	3	-	5.01	97.78

DGA required only 5 GB of memory to generate the solutions. Note that the distributed approach is implemented with a parallel architecture. It also found the solutions with at most 5% optimality gap in less than 2.01 hours, which translates to about 97 % shorter runtime than CPLEX.

We allowed DGA to run for 4000 iterations (about 50% more iterations). The optimality gap of DGA is reduced from 5.49% to 2.80%, 5.50% to 2.49%, 5.01% to 3.35%. The runtime of DGA is changed from 1.60 to 3.98, 1.81 to 4.87, and 2.01 to 4.89 hours, and still, the computation time is substantially less than that of CPLEX (90% less). The new results show that the difference of runtime with CPLEX is still large even though the optimality gap is very small.

Moreover, the computation time of the algorithm proposed in this paper is 0.76 seconds, which is 33.34% less than the lowest CPU time for all tested formulations by Long *et al.* [19]. The number of variables for LTM and CTM-based formulations are 1,656 and 5,400. The CPU time to solve four different CTM-based formulations by Long and Szeto [20] varies between 1.14 seconds to 36.32 seconds.

Fig. 5.a-c shows the objective value of DGA and the optimal objective value for three demand profiles over iterations. The distributed approach reduces the value of the objective function towards the optimal value over iterations. Note that the optimal solution from CPLEX had the well-known holding back problem due to the linearization of the minimization functions in CTM. However, the solutions from the distributed approach did not have this issue since we found the solutions by simulating the network using non-linear CTM equations as presented in [51]. In all three figures, the optimality gap was at most 5% when we stopped the approach.



(a) demand profile 1

(b) demand profile 2

(c) demand profile 3

Fig. 5. The objective functions for the network of 20 intersections.

2) Comparison With a Danzig-Wolfe Decomposition-Based Algorithm (DWDA): We compared the solutions and perfor-

TABLE II
BENCHMARK WITH DANZIG-WOLFE DECOMPOSITION ALGORITHM FOR THE NETWORK OF 20 INTERSECTIONS

Demand	Approach/ Gap	Number of variables in sub-problems	Iterations	Objective Function	Optimality gap (%)	Run-time (hr)	Memory (GB)
1	DWDA	281,200	11	197.69	0.05	0.92	20
	DGA	≤ 146,880	1706	197.34	0.05	1.60	5
	Diff. (%)	-48	15409	0.18	0.00	74	-75
2	DWDA	281,200	53	308.42	0.05	2.34	20
	DGA	≤ 146,880	1708	307.30	0.05	1.81	5
	Diff. (%)	-48	3123	0.36	0.00	-23	-75
3	DWDA	281,200	101	529.29	0.05	8.83	20
	DGA	≤ 146,880	1715	524.84	0.05	2.01	5
	Diff. (%)	-48	1599	0.84	0.00	-77	-75

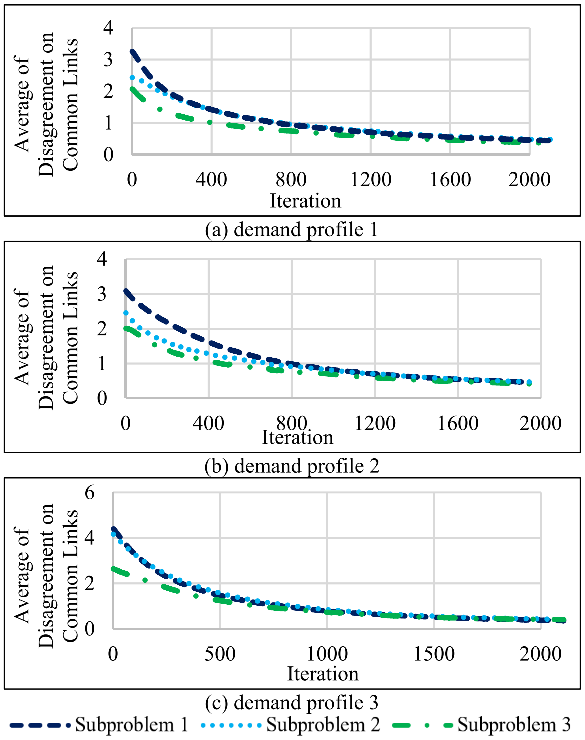
mance of the DGA to an OD-based decomposition approach developed in [13]. TABLE II shows different characteristics and performance measures for both approaches. Danzig-Wolfe Decomposition-based Algorithm (DWDA) has a master problem and several sub-problems that are solved iteratively with a stopping criterion of a 5% gap between the upper bound and lower bound of the approach. Each sub-problem has all decision variables and constraints for a SODTA formulation with one OD pair. Since we have 15 ODs, the number of sub-problems is 15. New extreme points are generated by sub-problems and added to the master problem solution pool. Therefore, the complexity of the master problem increases over iterations.

DGA has an intersection-based decomposition, and the number of sub-problems is 20 due to having 20 intersections in the network. This approach does not have any central component or master problem. The number of decision variables differs slightly in each sub-problem of DGA depending on the number of nodes and links and is at least 48% less than the number of variables in the sub-problems of DWDA. The number of iterations in DWDA is at most 101 while this number is 1715 for DGA. Even though we have more number of iterations in DGA, the computation time of each iteration is much less. The total run-time of DGA is 74% more than DWDA in undereducated demand pattern. However, when most of the decision variables have non-zero values in the oversaturated condition, the runtime of DGA is improved by 77% compared to DWDA. Moreover, since we have 48% fewer variables in DGA, we only required 5 GB of memory though we need at least 20 GB of memory to run DWDA.

B. The Performance of the Methodology

Fig. 6 presents the disagreement on the value of shared decision variables for three sub-problems with their neighbors for three demand profiles over iterations. In demand profile 1, when each sub-problem has a disagreement value of less than 0.5, the algorithm is terminated.

The algorithm reached a 5.5% optimality gap for demand profile 1 at iteration 1706, see Fig. 6.a. The algorithm reached a disagreement value of 0.5 for the second demand pattern at iteration 1708 with the 5.5% optimality gap, see Fig. 6.b. The approach also reached a disagreement value of 0.5 at iteration 1715 iterations for the third demand pattern with a similar gap of 5.0%. As expected, the disagreement increased with the demand level. However, the gap is the same for all scenarios once the approach is terminated.



(a) demand profile 1

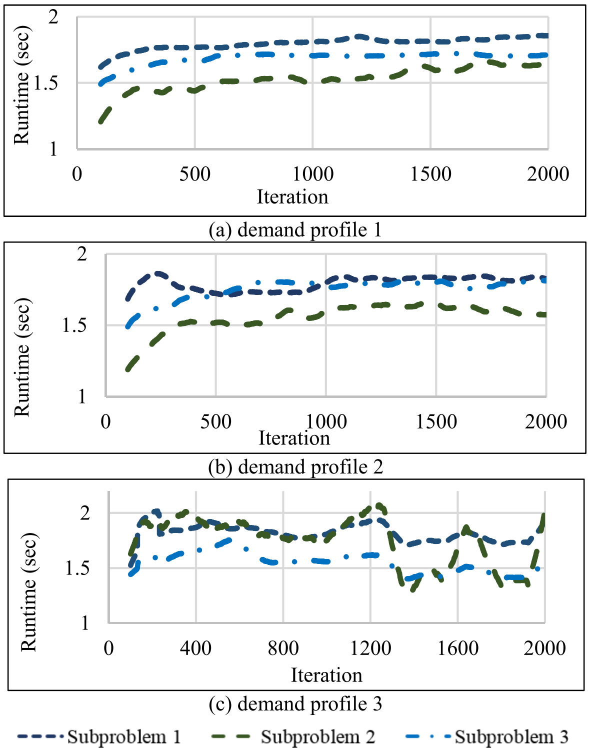
(b) demand profile 2

(c) demand profile 3

Fig. 6. Disagreement value ε over iterations in 20-intersection network.

Fig. 7.a - c shows the run-time of three sub-problems over iterations for the three demand profiles. We assigned each sub-problem to a different computational node using a multi-thread platform. The sub-problems are independent and optimized synchronously. Each sub-problem represents one intersection with an almost equal number of variables and constraints with other sub-problems. Therefore, the run-time is approximately the same among different nodes, which reduces overhead delays. The run-time of each sub-problem is relatively similar for all three demand profiles, and having a more congested sub-problem did not significantly affect the computation time; however, the total run-time differs by at most 20% between demand profiles due to the additional number of required iterations.

TABLE III shows the computation time for each section of the approach under three demand patterns. The runtime for the initialization step is negligible for all demand profiles. The distributed gradient update consists of step 1: computing weighted averages, step 2: moving values towards the negative direction of the gradient, and step 3: projecting values on the feasible region of sub-problems. Finding the weighted average needs the least CPU time, and projection has the highest computation time. The projection of the value of variables is implemented for all sub-problems at the same time, and its CPU time has increased by increasing the demand. This can happen because more number of decision variables has non zero values as the demand increases, and more computational effort is required to find variables' optimal values. No spe-



(a) demand profile 1

(b) demand profile 2

(c) demand profile 3

Fig. 7. The run-time of three sub-problems 20-intersection network.

TABLE III
BREAKDOWN OF RUNTIMES FOR THE NETWORK OF 20 INTERSECTIONS

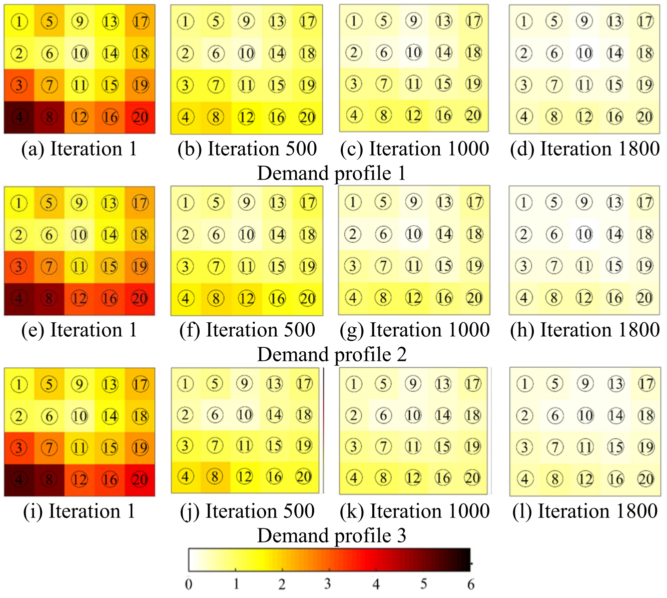
Demand Level		1	2	3
Initialization (hr)		0.0000	0.0000	0.0000
Distributed gradient update (hr)	Step 1	0.0023	0.0024	0.0022
	Step 2	0.0035	0.0036	0.0033
	Step 3	1.5525	1.7643	1.9689
Termination criterion (hr)		0.0351	0.0346	0.0338

cific trend has appeared for other steps including termination criterion calculation.

Fig. 8.a-d shows the disagreement on the value of decision variables for each sub-problem with its neighbors at iterations 1, 500, 1000, 1800 for three demand profiles. Each sub-problem is shown with a number following the same layout shown in Fig. 4. The spectrum shows a range of colors depending on the value of the disagreement. Darker color represents a higher disagreement value. Fig. 8.a-d present the disagreement at each sub-problem for the under-saturated demand profile. The colors become lighter as the number of iterations increases, which shows that the conflict on the proposed value by each sub-problem with its neighbors decreases. Fig. 8.e-h and Fig. 8.i-l display the same pattern for the second and third demand profiles, respectively.

C. The Effects of the Number of ODs on Convergence

Fig. 9 shows the impact of the number of OD pairs (15, 20, and 40) on the convergence of our approach in the test network of 20 intersections and semi-saturated demand. The algorithm reached the termination criteria in 1708, 2157, and 2319 iterations with 5%, 4%, and 5% optimality gaps for 15, 20, and 40 ODs, respectively. Increasing the number of OD pairs led to more iterations for convergence; however, the number of iterations does not increase as fast as the number of decision variables. Specifically, increasing the number of ODs from 20 to 40 doubles the number of decision variables but only increases the number of iterations by 7%.



(a) Iteration 1 (b) Iteration 500 (c) Iteration 1000 (d) Iteration 1800
Demand profile 1

(e) Iteration 1 (f) Iteration 500 (g) Iteration 1000 (h) Iteration 1800
Demand profile 2

(i) Iteration 1 (j) Iteration 500 (k) Iteration 1000 (l) Iteration 1800
Demand profile 3

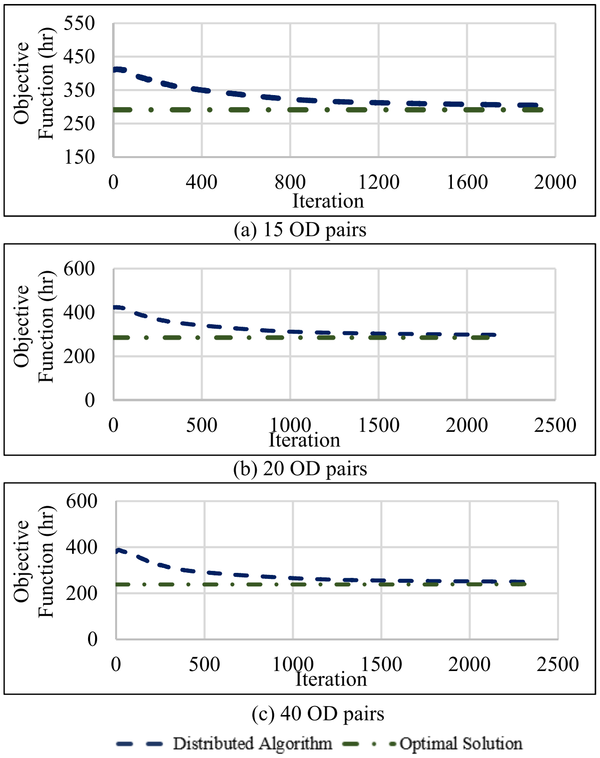
Fig. 8. The value of disagreement at each sub-problem.

D. The Effects of the Study Period Duration on Convergence

In this section, we evaluated the effect of increasing the duration of the study period on convergence. We tested three loading periods of 150, 300, and 450 time steps, as shown in Fig. 10. The three cases required 1420, 1708, and 2081 iterations with 5%, 4%, and 4% optimality gaps to reach the termination criterion. As we increased the loading time, the number of required iterations to meet the termination was increased. Increasing the loading time created more congestion. Therefore, the number of iterations was increased by increasing the loading period. However, the rate of increase in the required iterations was much less than the increase in the number of decision variables.

E. The Effects of the Network Size on Convergence

We also studied the effects of network size on convergence by looking at networks of 10, 20, and 40 intersections with similar characteristics, as shown in Fig. 11. An increase in network size increased the number of decision variables from 1,674,000 to 4,218,000, and 14,120,00. DGA was converged in 807, 1708, and 1710 iterations with 2%, 4%, and 5% optimality gaps for networks with 10, 20, 40 intersections, respectively. The loading period, demand, and the number of OD pairs were the same in all cases. The number of iterations for convergence was increased by increasing the network size; however, at a rate much slower than the increase in the network size. Increasing the size of the network from 20 intersections to 40 increases the number of required iterations by only two, which shows the scalability of the proposed methodology.

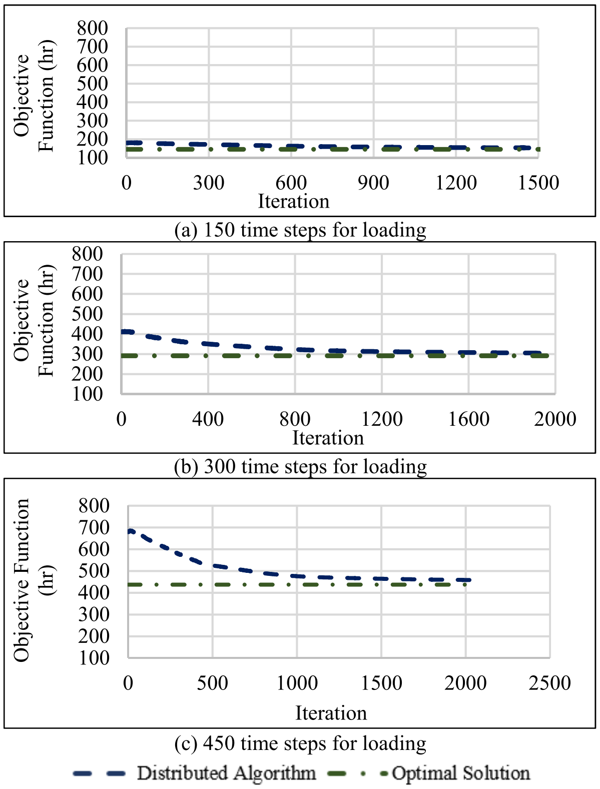


(a) 15 OD pairs

(b) 20 OD pairs

(c) 40 OD pairs

Fig. 9. The effects of the number of OD pairs on DGA convergence.



(a) 150 time steps for loading

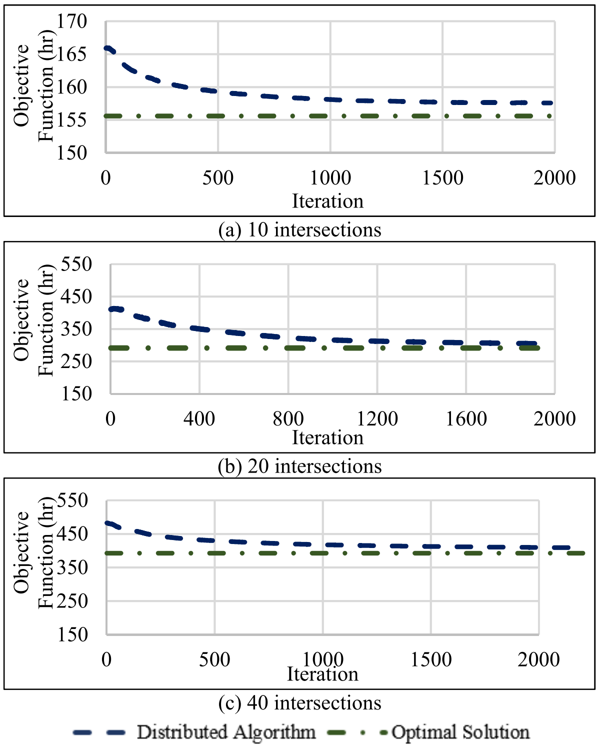
(b) 300 time steps for loading

(c) 450 time steps for loading

Fig. 10. The effect of study period duration on DGA convergence.

F. The Effect of Network Size on Decision Space

We also tested the approach on a network with 40 intersections with 632 cells, 780 links, 400 time steps, and 25 OD pairs that bring the total number of decision variables to 14,120,000. We used the same termination criterion that was applied to other cases. By increasing the intersections from 20 to 40, the number of decision variables increased from ∼4 million to ∼14 million (more than a factor of 3). We need to mention that the literature that includes models of networks with thousands of intersections and OD pairs use either exit flow functions, point queue models, or link-performance functions. These approaches are aggregated and have significantly fewer decision variables and do not provide the accuracy that is required for traffic operation purposes. In this paper, we use the CTM model, which is more accurate but at the expense of additional complexity. The network of 40 intersections is significantly larger than comparable studies that have used the cell transmission model [10], [12], [52]. The studies solve the problem for 5805 to 489,700 decision variables [11], [16].



(a) 10 intersections

(b) 20 intersections

(c) 40 intersections

Fig. 11. The effect of network size on DGA convergence.

Increasing the number of links from 387 to 780 and nodes from 316 to 632 does not change the complexity of each sub-problem due to the intersection-based distribution of the formulation. However, increasing the study period from 400 to 500 time steps and OD pairs from 15 to 25 lead to having more variables. Even though the variables are increased by 70%, we can find the solutions with at most 5.70% optimality gap in at most 5.34 hours, as shown in TABLE IV.

G. The Effect of the Network Size on Run-Time

Fig. 12 shows the total run-time for sub-problems for the network of 20 intersections with 15 and 30 ODs and the network of 40 intersections with 25 ODs. Increasing the number of OD pairs from 15 to 30 in the network of 20 intersections has increased the number of decision variables by 50% and run-time at each iteration by 71% on average.

TABLE IV
INCREASING THE NETWORK SIZE FROM 20 TO 40 INTERSECTIONS

Demand	Intersections	Optimality gap (%)		Runtime (hr)		Number of Iterations	
		DGA	DWDA	DGA	DWDA	DGA	DWDA
1	20	5.90	5.06	1.60	0.92	1704	11
	40	5.68	NA	5.19	NA	1710	NA
2	20	5.28	5.88	1.81	2.34	1708	53
	40	5.68	NA	5.26	NA	1711	NA
3	20	5.93	5.91	2.01	8.83	1715	101
	40	5.70	NA	5.34	NA	1716	NA

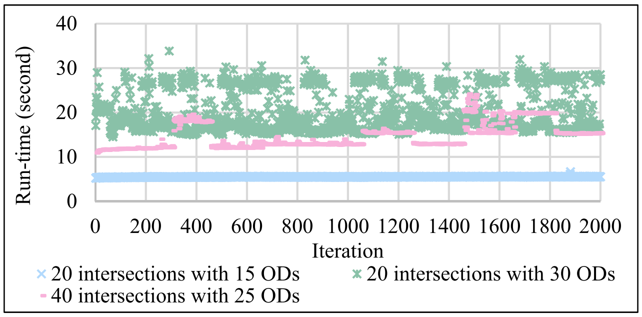


Fig. 12. The run-times of the proposed methodology.

When the size increases, the number of variables in each sub-problem increases only by the number of OD pairs in this case. The number of cells, links, and time steps remains constant. By increasing the number of intersections from 20 with 15 ODs to 40 intersections with 25 ODs, the number of decision variables in each sub-problem has increased by 70%. The run-time to generate solutions at each time step has increased by 52% on average.

VII. Conclusion

We introduced a gradient-based methodology that distributed SODTA into several independent sub-problems. This methodology overcomes the main drawback of existing decomposition techniques by having a fully distributed framework that does not require a centralized component. Each sub-problem represents one intersection. The set of constraints and the objective function of the CTM-based SODTA problem is distributed among sub-problems such that each sub-problem has some parts of the SODTA formulation with decision variables corresponding to the intersection region assigned to that sub-problem.

DGA updates the decision variables at each iteration by combining the proposed values for the shared decision variables by finding their weighted average. The approach also considers the gradient of the objective function at each sub-problem in updating these values. Then, DGA projects the computed values on the feasible region of each sub-problem. The update procedure continues until it reaches a consensus

TABLE V
NOTATIONS

Sets	
T	The set of time steps
C	The set of network cells
C_O	The set of source cells
C_S	The set of sink cells
C_I	The set of intersection cells
C_{OD}	The set of OD pairs
$P(i)$	The set of predecessors of cell $i \in C$
$S(i)$	The set of successors of cell $i \in C$
C_D	The set of diverge cells
N	The set of sub-problems
Cl	The set of links that their heads $i \in C^s$and tails $j \in S(i)$ belong to sub-problems $e \in C^{\mathscr{s} \in n^s}$ and $f \in S(e)$ such that $i = e, j = f, s \neq \mathscr{s}$
n^s	The set of neighbors of sub-problem $s \in N$ including itself
C^s	The set of cells in sub-problem $s \in N$
Co^s	The union of the set of cells in sub-problem $s \in N$ and the set of source cells, i.e., $C_O \cap C^s$
Cs^s	The union of the set of cells in sub-problem $s \in N$ and the set of sink cells, i.e., $C_S \cap C^s$
C_D^s	The set of diverge cells in sub-problem $s \in N$
$Ceos^s$	The set of cells in sub-problem $s \in N$ except for the set of source cells and the set of sink cells i.e., $C^s/C_O \cup C_S$
Ces^s	The set of cells in sub-problem $s \in N$ except for the set of sink cells, i.e., C^s/C_S
Ceo^s	The set of cells in sub-problem $s \in N$ except for the set of source cells, i.e., C^s/C_O
Ci^s	The union of the set of cells in sub-problem $s \in N$ and the set of intersection cells, i.e., $C_I \cap C^s$
X_s	The set of constraints of subproblem $s \in N$
X	The feasible region of SODTA problem
X^*	The set of optimal solutions
Decision variables	
$x_i^{t,od}$	The number of vehicles in cell $i \in C$ at time step $t \in T$ for OD pair $(o,d) \in C_{OD}$
$y_{ij}^{t,od}$	The number of vehicles flowing from cell $i \in C$ to downstream cell $j \in S(i)$ at time step $t \in T$ for OD pair $(o,d) \in C_{OD}$
Parameters	
τ	The duration of time step $t \in T$
$\mathcal{D}_i^{t,od}$	The entry demand level at source cell $i \in C_O$ at time step $t \in T$ for OD pair $(o,d) \in C_{OD}$
F_i	The saturation flow rate at cell $i \in C$
M_i	The maximum number of vehicles that cell $i \in C$ can accommodate
δ	The ratio of free-flow speed to the backward propagation speed
g_i^t	A binary parameter to define signal status at intersection cell $i \in C_I$ at time step $t \in T$. Zero and one values indicate red and green signals, respectively
$R_{ij}^{t,od}$	The turning ratio of the link between diverge cell $i \in C_D$ and its successor cell $j \in S(i)$ at time step $t \in T$ for OD pair $(o,d) \in C_{OD}$
f_i^t	The variable saturation flow rate of intersection cell $i \in C_I$ at time step $t \in T$
Notations and Terms	
K	The total number of iterations
k	The iteration counter
$\mathbb{X}_s^k$	The column vector of values for the decision variables of sub-problem $s \in N$ at iteration $k \in K$
$\boldsymbol{x}$ and $\boldsymbol{z}$	The column vectors
$\boldsymbol{x}'$	The transpose of vector $\boldsymbol{x}$
$\boldsymbol{x}'\mathbf{z}$	The dot product of two vectors $\boldsymbol{x}$ and z
$\boldsymbol{z}^* \in X^*$	A realization of the vector of optimal solution
$\lVert\boldsymbol{x}\rVert = (\boldsymbol{x}'\boldsymbol{x})^{\frac{1}{2}}$	The standard Euclidean norm
$\boldsymbol{P}_X[\boldsymbol{x}]$	The projection of vector $\boldsymbol{x}$ on set X
$\mathcal{F}$	The objective function of SODTA problem
$\mathcal{F}_s(\boldsymbol{x})$	The objective function of sub-problem $s \in N$ at vector $\boldsymbol{x}$
$\mathcal{F}^*$	The optimal objective function
G	The information exchange graph with a set of nodes N and a set of

TABLE V

(Continued.) NOTATIONS

$= (N, A)$	directional links A
$(s, \mathscr{s}) \in A$	A link in information exchange graph $G = (N, A)$
$\mathscr{w}_{s\mathscr{s}}$	The weight on the link going from node s to node $\mathscr{s}$ in information exchange graph $G = (N, A)$
θ and θ'	The lower bound and the upper bound for weight values $\mathscr{w}_{s\mathscr{s}}$: $s\mathscr{s} \in N$
$\mathscr{w}$	The weighted graph Laplacian matrix with weight $\mathscr{w}_{s\mathscr{s}}$ entries assigned to links $(s, \mathscr{s}) \in N$ in information exchange graph $G = (N, A)$
$\boldsymbol{\mathcal{G}}_s$	The gradient of the objective function $\mathcal{F}_s$ of sub-problem $s \in N$
$\mathcal{G}_{s,i}^{t,od}$	The gradient of objective function of sub-problem $s \in N$ respect to $x_i^{t,od}$
$\mathcal{G}_{s,ij}^{t,od}$	The gradient of objective function of sub-problem $s \in N$ respect to $y_{ij}^{t,od}$
α^k and γ^k	The step sizes at iteration $k \in K$
ε	The consensus error used for termination criterion
$\mathscr{p}_i^{t,od}$	The auxiliary parameter for vehicles in cell $i \in C$ at time step $t \in T$ with OD pair $(o, d) \in C_{OD}$
$\mathscr{q}_{ij}^{t,od}$	The auxiliary parameter for vehicles going from cell $i \in C$ to downstream cell $j \in S(i)$ at time step $t \in T$ with OD pair $(o, d) \in C_{OD}$
$\mathbb{A} \backslash \mathbb{B}$	All elements in set $\mathbb{A}$ except for the ones in set $\mathbb{B}$

on the proposed values for shared decision variables among sub-problems. DGA was tested on a Nguyen and Dupuis [48] network and a portion of the downtown Springfield network in Illinois. The run time of this approach was 33.34% less than the lowest run time for tested formulations on Nguyen and Dupuis [48] network by Long and Szeto [20]. DGA required 96% less memory and 97% less time compared to CPLEX to find solutions with a 5% optimality gap. In comparing DGA with a Danzig-Wolfe Decomposition-based Algorithm, the runtime was improved by 77% in oversaturated condition.

Adapting DGA to solve various types of DTA formulations like the DTA model proposed for emergency conditions [53] and the DTA model incorporated with signal timing optimization [54]–[56] can be considered as future research directions. Further research on different network partitioning scenarios presented in [57] and their effects on the convergence and run-time of our approach is needed. In a future study, we will consider multi-modal transportation systems similar to the model proposed by Bevrani *et al.* [58]. A more general formulation with multi-commodity flows similar to the presented model by Bevrani *et al.* [59], [60] can also be employed for generalizing this approach. We will also study the possibility of estimating the expected number of iterations to achieve a solution within a small gap in future studies.

APPENDIX NOTATIONS

See Table V.

REFERENCES

[1] A. Karoonsoontawong and S. T. Waller, "Integrated network capacity expansion and traffic signal optimization problem: Robust bi-level dynamic formulation," *Netw. Spatial Econ.*, vol. 10, no. 4, pp. 525–550, 2010.

[2] L. Hajibabai and Y. Ouyang, "Integrated planning of supply chain networks and multimodal transportation infrastructure expansion: Model development and application to the biofuel industry," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 28, no. 4, pp. 247–259, Apr. 2013.

[3] L. Hajibabai, Y. Bai, and Y. Ouyang, "Joint optimization of freight facility location and pavement infrastructure rehabilitation under network traffic equilibrium," *Transp. Res. B, Methodol.*, vol. 63, pp. 38–52, May 2014.

[4] A. Mirheli and L. Hajibabai, "Hierarchical optimization of electric vehicle charging infrastructure design and facility logistics," 2021, *arXiv:2105.10557*.

[5] C. Beard and A. Ziliaskopoulos, "System optimal signal optimization formulation," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1978, no. 1, pp. 102–112, Jan. 2006.

[6] W. Shen and H. M. Zhang, "On the morning commute problem in a corridor network with multiple bottlenecks: Its system-optimal traffic flow patterns and the realizing tolling scheme," *Transp. Res. B, Methodol.*, vol. 43, no. 3, pp. 267–284, Mar. 2009.

[7] W. Shen, Y. Nie, and H. M. Zhang, "Dynamic network simplex method for designing emergency evacuation plans," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2022, no. 1, pp. 83–93, Jan. 2007.

[8] S. Peeta and S. Bulusu, "Generalized singular value decomposition approach for consistent on-line dynamic traffic assignment," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1667, no. 1, pp. 77–87, Jan. 1999.

[9] A. K. Ziliaskopoulos, "A linear programming model for the single destination system optimum dynamic traffic assignment problem," *Transp. Sci.*, vol. 34, no. 1, pp. 37–49, 2000.

[10] Y.-C. Chiu and H. Zheng, "Real-time mobilization decisions for multi-priority emergency response resources and evacuation groups: Model formulation and solution," *Transp. Res. E, Logistics Transp. Rev.*, vol. 43, no. 6, pp. 710–736, Nov. 2007.

[11] H. Zheng and Y.-C. Chiu, "A network flow algorithm for the cell-based single-destination system optimal dynamic traffic assignment problem," *Transp. Sci.*, vol. 45, no. 1, pp. 121–137, 2011.

[12] H. M. A. Aziz and S. V. Ukkusuri, "Integration of environmental objectives in a system optimal dynamic traffic assignment model," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 27, no. 7, pp. 494–511, Aug. 2012.

[13] M. Mehrabipour, L. Hajibabai, and A. Hajbabaie, "A decomposition scheme for parallelization of system optimal dynamic traffic assignment on urban networks with multiple origins and destinations," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 34, no. 10, pp. 915–931, Oct. 2019.

[14] E. Jafari, V. Pandey, and S. D. Boyles, "A decomposition approach to the static traffic assignment problem," *Transp. Res. B, Methodol.*, vol. 105, pp. 270–296, Nov. 2017.

[15] D.-Y. Lin, V. Valsaraj, and S. T. Waller, "A Dantzig–Wolfe decomposition-based heuristic for off-line capacity calibration of dynamic traffic assignment," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 26, no. 1, pp. 1–15, 2011.

[16] Y. Li, S. T. Waller, and T. Ziliaskopoulos, "A decomposition scheme for system optimal dynamic traffic assignment models," *Netw. Spatial Econ.*, vol. 3, no. 4, pp. 441–455, 2003.

[17] T. Larsson and M. Patriksson, "Simplicial decomposition with disaggregated representation for the traffic assignment problem," *Transp. Sci.*, vol. 26, no. 1, pp. 4–17, Feb. 1992.

[18] T. Larsson, M. Patriksson, and C. Rydergren, "A column generation procedure for the side constrained traffic equilibrium problem," *Transp. Res. B, Methodol.*, vol. 38, no. 1, pp. 17–38, Jan. 2004.

[19] J. Long, C. Wang, and W. Y. Szeto, "Dynamic system optimum simultaneous route and departure time choice problems: Intersection-movement-based formulations and comparisons," *Transp. Res. B, Methodol.*, vol. 115, pp. 166–206, Sep. 2018.

[20] J. Long and W. Y. Szeto, "Link-based system optimum dynamic traffic assignment problems in general networks," *Oper. Res.*, vol. 67, no. 1, pp. 167–182, Jan. 2019.

[21] L. Hajibabai and D. Saha, "Patrol route planning for incident response vehicles under dispatching station scenarios," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 34, no. 1, pp. 58–70, Jan. 2019.

[22] M. Tajalli and A. Hajbabaie, "Traffic signal timing and trajectory optimization in a mixed autonomy traffic stream," *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 18, 2021, doi: 10.1109/TITS.2021.3058193.

[23] A. Mirheli, M. Tajalli, L. Hajibabai, and A. Hajbabaie, "A consensus-based distributed trajectory control in a signal-free intersection," *Transp. Res. C, Emerg. Technol.*, vol. 100, pp. 161–176, Mar. 2019.

[24] S. M. A. B. Al Islam and A. Hajbabaie, "Distributed coordinated signal timing optimization in connected transportation networks," *Transp. Res. C, Emerg. Technol.*, vol. 80, pp. 272–285, Jul. 2017.

[25] G. Ramadurai and S. Ukkusuri, "B-dynamic: An efficient algorithm for dynamic user equilibrium assignment in activity-travel networks1," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 26, no. 4, pp. 254–269, May 2011.

[26] E. Jafari and S. D. Boyles, "Improved bush-based methods for network contraction," *Transp. Res. B, Methodol.*, vol. 83, pp. 298–313, Jan. 2016.

[27] S. D. Boyles, "Bush-based sensitivity analysis for approximating subnetwork diversion," *Transp. Res. B, Methodol.*, vol. 46, no. 1, pp. 139–155, Jan. 2012.

[28] A. Gibert, "A method for the traffic assignment problem," London Bus. School London, Transp. Netw. Theory Unit, Tech. Rep. LBS-TNT-95, 1968.

[29] T. Leventhal, G. Nemhauser, and L. Trotter, "A column generation algorithm for optimal traffic assignment," *Transp. Sci.*, vol. 7, no. 2, pp. 168–176, May 1973.

[30] M. Florian, I. Constantin, and D. Florian, "A new look at projected gradient method for equilibrium assignment," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2090, no. 1, pp. 10–16, Jan. 2009.

[31] C. F. Daganzo, "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory," *Transp. Res. B, Methodol.*, vol. 28, no. 4, pp. 269–287, 1994.

[32] C. F. Daganzo, "The cell transmission model, Part II: Network traffic," *Transp. Res. B, Methodol.*, vol. 29, no. 2, pp. 79–93, 1995.

[33] R. Mohebifard and A. Hajbabaie, "Dynamic traffic metering in urban street networks: Formulation and solution algorithm," *Transp. Res. C, Emerg. Technol.*, vol. 93, pp. 161–178, Aug. 2018.

[34] M. Mehrabipour and A. Hajbabaie, "A cell-based distributed-coordinated approach for network-level signal timing optimization," *Comput. Civ. Infrastruct. Eng.*, vol. 32, no. 7, pp. 599–616, 2017.

[35] M. Mehrabipour, "Real-time network-level signal timing optimization," M.S. thesis, Washington State Univ., Pullman, WA, USA, Tech. Rep., 2018.

[36] S. M. A. B. A. Islam, A. Hajbabaie, and H. M. A. Aziz, "A real-time network-level traffic signal control methodology with partial connected vehicle information," *Transp. Res. C, Emerg. Technol.*, vol. 121, Dec. 2020, Art. no. 102830.

[37] S. M. A. B. Al Islam and A. Hajbabaie, "An enhanced cell transmission model for multi-class signal control," *IEEE Trans. Intell. Transp. Syst.*, early access, Aug. 11, 2021, doi: 10.1109/TITS.2021.3101838.

[38] S. M. A. B. Al Islam, H. M. A. Aziz, and A. Hajbabaie, "Stochastic gradient-based optimal signal control with energy consumption bounds," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 5, pp. 3054–3067, May 2021.

[39] W.-H. Lin and H. Liu, "Enhancing realism in modeling merge junctions in analytical models for system-optimal dynamic traffic assignment," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 4, pp. 838–845, Dec. 2010.

[40] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice hall, 1995.

[41] S. V. Ukkusuri, L. Han, and K. Doan, "Dynamic user equilibrium with a path based cell transmission model for general traffic networks," *Transp. Res. B, Methodol.*, vol. 46, no. 10, pp. 1657–1684, Dec. 2012.

[42] R. Zhong, J. Xiong, Y. Huang, A. Sumalee, A. H. F. Chow, and T. Pan, "Dynamic system optimum analysis of multi-region macroscopic fundamental diagram systems with state-dependent time-varying delays," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 4000–4016, Sep. 2020.

[43] M. Tajalli, M. Mehrabipour, and A. Hajbabaie, "Network-level coordinated speed optimization and traffic light control for connected and automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 11, pp. 6748–6759, Nov. 2021.

[44] R. Mohebifard, S. M. A. B. A. Islam, and A. Hajbabaie, "Cooperative traffic signal and perimeter control in semi-connected urban-street networks," *Transp. Res. C, Emerg. Technol.*, vol. 104, pp. 408–427, Jul. 2019.

[45] R. Mohebifard and A. Hajbabaie, "Distributed optimization and coordination algorithms for dynamic traffic metering in urban street networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1930–1941, May 2019.

[46] K. Srivastava, A. Nedić, and D. M. Stipanović, "Distributed constrained optimization over noisy networks," in *Proc. 49th IEEE Conf. Decis. Control (CDC)*, Dec. 2010, pp. 1945–1950.

[47] K. Srivastava, "Distributed optimization with applications to sensor networks and machine learning," Ph.D. dissertation, Univ. Illinois Urbana-Champaign, Champaign, IL, USA, Tech. Rep., 2012.

[48] S. Nguyen and C. Dupuis, "An efficient method for computing traffic equilibria in networks with asymmetric transportation costs," *Transp. Sci.*, vol. 18, no. 2, pp. 185–202, 1984.

[49] C. Antoniou, M. Ben-Akiva, and H. N. Koutsopoulos, "Nonlinear Kalman filtering algorithms for on-line calibration of dynamic traffic assignment models," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 4, pp. 661–670, Dec. 2007.

[50] X. Hu, Y.-C. Chiu, J. A. Villalobos, and E. Nava, "A sequential decomposition framework and method for calibrating dynamic origin—Destination demand in a congested network," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 10, pp. 2790–2797, Oct. 2017.

[51] R. Mohebifard and A. Hajbabaie, "Optimal network-level traffic signal control: A benders decomposition-based solution algorithm," *Transp. Res. B, Methodol.*, vol. 121, pp. 252–274, Mar. 2019.

[52] K. Doan and S. V. Ukkusuri, "Dynamic system optimal model for multi-OD traffic networks with an advanced spatial queuing model," *Transp. Res. C, Emerg. Technol.*, vol. 51, pp. 41–65, Feb. 2015.

[53] M. Di Gangi, "Modeling evacuation of a transport system: Application of a multimodal mesoscopic dynamic traffic assignment model," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1157–1166, Dec. 2011.

[54] L.-W. Chen and T.-Y. Hu, "Flow equilibrium under dynamic traffic assignment and signal control—An illustration of pretimed and actuated signal control policies," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1266–1276, Sep. 2012.

[55] A. Hajbabaie and R. F. Benekohal, "A program for simultaneous network signal timing optimization and traffic assignment," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2573–2586, Oct. 2015.

[56] A. Hajbabaie, "Intelligent dynamic signal timing optimization program," Dr. dissertation, Univ. Illinois Urbana-Champaign, Champaign, IL, USA, Tech. Rep., 2012.

[57] C. N. Yahia, V. Pandey, and S. D. Boyles, "Network partitioning algorithms for solving the traffic assignment problem using a decomposition approach," *Transp. Res. Rec.*, vol. 2672, no. 48, pp. 116–126, 2018.

[58] B. Bevrani, R. L. Burdett, A. Bhaskar, and P. K. D. V. Yarlagadda, "A capacity assessment approach for multi-modal transportation systems," *Eur. J. Oper. Res.*, vol. 263, no. 3, pp. 864–878, Dec. 2017.

[59] B. Bevrani, R. L. Burdett, A. Bhaskar, and P. K. D. V. Yarlagadda, "A multi commodity flow model incorporating flow reduction functions," *Flexible Services Manuf. J.*, vol. 32, no. 3, pp. 1–31, 2019.

[60] B. Bevrani, R. Burdett, A. Bhaskar, and P. K. D. V. Yarlagadda, "A multi-criteria multi-commodity flow model for analysing transportation networks," *Oper. Res. Perspect.*, vol. 7, Jan. 2020, Art. no. 100159.

Mehrzad Mehrabipour received the B.Sc. and M.Sc. degrees (Hons.) in industrial engineering in Iran in 2012 and 2014, respectively, and the M.Sc. degree in civil engineering from Washington State University in 2018. She is currently pursuing the Ph.D. degree with the Civil, Construction, and Environmental Engineering Department, North Carolina State University. She worked as an Instructor for about two years with the Industrial Engineering Department, Shahid Bahonar University. Her research interests are traffic assignment and operation research in congestion management.

Ali Hajbabaie (Member, IEEE) received the B.S. and M.S. degrees from the Sharif University of Technology, Tehran, Iran, in 2003 and 2006, respectively, and the M.S. degree in industrial engineering and the Ph.D. degree in civil engineering from the University of Illinois at Urbana-Champaign in 2011 and 2012, respectively. He is an Assistant Professor with the Civil, Construction, and Environmental Engineering Department, North Carolina State University. His research interests include traffic operations and control, connected and automated vehicles, and traffic flow theory. He was a recipient of the Best Paper Award from the Work Zone Traffic Control Committee of the Transportation Research Board. He is the Secretary of the work zone traffic control and the Chair of the Asset Management Subcommittee of the Traffic Signal Systems Committee of the Transportation Research Board.