

Scheduling Vehicles for On-Time Arrival using Route-Reservations

C. Menelaou, S. Timotheou, P. Kolios, C.G. Panayiotou and M.M. Polycarpou

Abstract—In this work, we address the problem of scheduling vehicle departures from their origin such that they will arrive at their destination on-time. For this problem, vehicles transmit to a central scheduler their origin and destination pair and the time that they require to arrive at their destination. The scheduler determines each vehicle's departure time as well as the path to be followed while making the appropriate route reservations on the selected path such that all scheduled vehicles avoid congested road segments. Due to the reservations, the scheduler can guarantee on-time arrival at the destination for each vehicle request. In this paper, the mathematical formulation of the proposed problem is presented, and an efficient algorithmic solution is derived. Microscopic simulation results demonstrate the substantial improvements obtained by applying the proposed algorithm in realistic scenarios.

I. INTRODUCTION

Traffic congestion is a major problem in modern cities with several adverse effects on the citizens quality of life as well as economy and environment. Unfortunately, the problem of congestion is expected to get worse as cities become bigger while the infrastructure remains fixed. A characteristic of congestion is that it typically occurs only during specific periods during the day (rush hours). More specifically, during rush hours, the number of vehicles that enter parts of the network is larger than the network's carrying capacity and as a result traffic congestion occurs. The motivation of this work is to distribute the peak demand over a larger period of time (i.e., some vehicles should enter the network earlier or later) such that the peak demand will not exceed the network's capacity. As a result, congestion will be avoided, and vehicles will arrive at their destination on-time, without excessive delays in the network. Such guarantees can be provided through the use of a reservation architecture as described in the sequel.

Several approaches have already been proposed to address the on-time arrival problem by determining each

vehicle's departure time and the associated route aiming to either maximize the vehicle's on-time arrival probability or to minimize the expected traversal time [1], [2]. The majority of the literature considers link-level dynamics, assuming that each link's travel time distribution is known [3]. However, this approach is not easy to implement especially during congested conditions since link travel time distributions can hardly be estimated [4]. The use of navigation systems onboard of connected vehicles can provide traffic state estimates and alternative routing choices in an effort to assist drivers in minimizing their travel time. However, such *uncoordinated* decisions can result in network state oscillations as drivers would opt to traverse through non-congested road segments seeking to minimize their travel times [5], [6] (selfish behavior). On the other hand, socially optimal strategies are more appropriate for alleviating the problem of congestion [7].

A solution to the problem is to prevent congestion altogether by restricting the number of vehicles in the network below its critical density [4]. This can be done by coordinating the departure times for each vehicle (i.e., apply demand management by controlling the time that vehicles enter the network) which can significantly improve the traffic flows and sustain travel times around those achieved assuming free-flow speed conditions [8], [9]. The main contribution of this work is that it proposes a reservation-based architecture that aims to compute the vehicles departure times and reserves their route to follow to reach their destination guaranteeing on-time arrival while ensuring a congestion-free operation.

A reservation-based architecture was also proposed in [10], [11], [12], however, the objective was to achieve the earliest possible arrival time at the destination. A key feature of that approach was that vehicles could be instructed to wait at their origin until a non-congested path was made available. In this work, the reservation architecture is similar; however, the optimization problem and objective are very different. In this approach, a vehicle sends to the central scheduler its origin-destination pair and its desired arrival time at the destination. The scheduler determines the time that the vehicle should depart from its origin and the path to follow, such that it will arrive at the destination on or before the desired arrival time. In this approach, the scheduler's objective is to minimize the difference between the departure and the desired arrival times such that congested links are avoided and travelers are not significantly inconvenienced (e.g., they do not arrive too early at their destination).

This work has been supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 739551 (KIOS CoE), the Government of the Republic of Cyprus through the Directorate General for European Programmes, Coordination and Development and through the Research Promotion Foundation (Project: CULTURE/BR-NE/0517/14).

C. Menelaou, S. Timotheou, P. Kolios, C.G. Panayiotou and M.M. Polycarpou are with the KIOS Research and Innovation Center of Excellence, and the Department of Electrical and Computer Engineering, University of Cyprus, {cmene102, timotheou.stelios, pkolios, christosp and mpolycar}@ucy.ac.cy

Furthermore, through the reservation architecture, the scheduler has a reasonable estimate of the *future* states of the network; thus it will route vehicles only through non-congested road segments.

The remaining of the paper is organized as follows. Section II briefly reviews the relevant literature and Section III mathematically describes the proposed reservation scheme and defines the on-time arrival problem (OTA). Section IV derives an algorithmic solution for the OTA problem which utilizes a backward route-reservation scheme that schedules vehicles through road-segments that are below their critical density. The performance of the proposed solution is investigated in Section V, demonstrating the gains achieved for a number of different metrics while also indicating that vehicles almost always arrive at their destination on-time. Concluding remarks are provided in Section VI.

II. RELATED WORK

Several papers address the problem of online traffic state estimation using for example routing techniques with deterministic or stochastic measurements to guide vehicles via paths that minimize the destination arrival time [13], [14]. There is also great interest in practical aspects of stochastic routing that aim at finding the least expected travel time paths or the most reliable paths, where the travel-time on each link is a random variable with an associated probability distribution [2], [1]. For the case of the most reliable paths, the objective is to reduce the risk of arriving late rather than to minimize the expected travel time [15]. For instance, some travelers tend to sacrifice travel-time to take a more reliable route when hard deadlines are considered. The issue with these approaches is that routing and scheduling decisions determine the traversal path without considering the adverse effects of congestion and the changes in traffic state due to unreliable estimates of the travel times [16]. The stochastic on-time arrival problem is formulated as a stochastic dynamic programming problem [17] and solved by determining the optimal path at each node based on the travel-time realized to that node [3]. This solution approach is computationally expensive making it non-practical for a real-case application since the detailed link-level dynamics should be taken into consideration for all routing decisions.

To address the computational complexity, recent efforts make their routing decisions based on macroscopic models as detailed in [18]. In this case, traffic dynamics are defined according to the three major mobility factors: speed, flow, and density which are used to build the Network Fundamental Diagram (NFD) [19]. The critical density can then be used to restrict external inflows across the network without considering the detail link-level models [20]. Apart from routing solutions, the possibility of spreading the traffic load across a larger set of road segments can significantly improve the overall traveling times [21]. Currently, perimeter control ap-

proaches constitute the state-of-the-art congestion mitigation mechanisms that utilize NFD dynamics to regulate the inflow along urban regions [22], [23]. External inflows are allowed to enter the network only if the critical density has not been reached. The significant advantage of that approach is that they are simple to implement within the existing road infrastructure since control is only applied at each region's boundaries. On the other hand, there is no control mechanism for flows generated within the region and unwanted queues may occur at boundaries which may obstruct the upstream network destinations [24] [25].

Route reservations have also been considered in the past. For example, the work in [26] investigates the effects of a slot allocation algorithm which can be used by an infrastructure manager to manage the demand (departure time allocation) for pre-specified paths. Furthermore, reservations adopted as an alternative to perimeter control have been considered in [27] for vehicle flows passing through a cordon in order to manage the capacity within the controlled regions. Furthermore, reservations have also been used in highway systems which provide higher priority access to lanes [28] and [29]. As already mentioned, this work derives a novel route-reservation architecture that can compute paths to solve the OTA problem allowing vehicles to arrive at their destinations on-time while eliminating traffic congestion.

III. PROBLEM FORMULATION

We assume that an urban area is modelled as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the sets \mathcal{V} and \mathcal{E} represent the road junctions (i.e., $\{v_i, v_j\} \in \mathcal{V}$) and the road-segments (i.e., $(i, j) \in \mathcal{E}$), respectively. Each road-segment $(i, j) \in \mathcal{E}$ is described by the parameters λ_{ij} and l_{ij} (km), denoting the number of lanes and its length, respectively. In addition, we assume that the road network under investigation constitutes an urban area with a well defined Network Fundamental Diagram (NFD) [30] with the following traffic parameters: ρ^C (veh/km), q^C (veh/h), ρ^J (veh/km), and u_f (km/h), representing the critical density, the maximum flow (capacity achieved only at critical density), jam density and free-flow speed, respectively. The segment's $(i, j) \in \mathcal{E}$ traffic dynamics are characterized by the parameters ρ_{ij}^C and ρ_{ij}^J , denoting the critical and jam densities, respectively. Note that to obtain the critical density of the road-segment (i, j) we use:

$$\rho_{ij}^C = (\rho^C / \rho^J) \rho_{ij}^J. \quad (1)$$

Hence, the critical density of each segment is proportional to the region's critical density. Furthermore, let variable $\rho_{ij}(k)$ veh/km denote the instantaneous density of each road segment at each time-slot $k \in \mathcal{T}$, where \mathcal{T} defines the time horizon of the problem. Considering Eq. (1) it is true that, for all $\rho_{ij}(k) \leq \rho_{ij}^C$ vehicles can be assumed to travel with free-flow speed u_f . On this premises, the number of time-slots that a vehicle is

require to traverse road segment (i, j) can be expressed as:

$$\bar{c}_{ij} = \lfloor l_{ij}/u_f/T \rfloor, \quad (2)$$

where T is the sampling interval and $\lfloor z \rfloor$ denotes the nearest integer to z .

As in [12], the proposed architecture keeps track of the accumulated number of vehicles reservations (i.e., $r_{ij}(k)$) of each road-segment (i, j) for time-slot k . This information provides an accurate estimate of the expected density of each road segment in order to be able to keep track of its admissibility state. A road-segment is assumed to be *admissible* at the discrete time-slot k if a vehicle starting from road junction i at time-slot k can traverse road segment (i, j) without making the accumulated reserved density larger than the critical density at any point within the transit time. Let variable $x_{ij}(k)$ denote the admissibility state taking the value $x_{ij}(k) = 1$ if segment $(i, j) \in \mathcal{E}$ is admissible and $x_{ij}(k) = 0$, otherwise. Mathematically the admissibility state can be defined as follows:

$$x_{ij}(k) = \begin{cases} 1, & \text{if } r_{ij}(\tau)/(\lambda_{ij}l_{ij}) \leq \rho_{ij}^C, \forall \tau = k, \dots, k + \bar{c}_{ij} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where the quantity $r_{ij}(\tau)/(\lambda_{ij}l_{ij})$ is the accumulated reserved density of road segment (i, j) at time $\tau \in [k, \dots, k + \bar{c}_{ij}]$. Given the admissibility state, the cost of traversing a road segment (i.e., $c_{ij}(k)$) can be defined as follows:

$$c_{ij}(k) = \begin{cases} \bar{c}_{ij}, & \text{if } x_{ij}(k) = 1 \\ \infty, & \text{if } x_{ij}(k) = 0 \end{cases} \quad (4)$$

On-Time Arrival (OTA) problem:

Given the origin-destination pair of the m -th vehicle (i.e., $O_m - D_m$, with $O_m, D_m \in \mathcal{V}$), the desirable destination arrival time $d_{D_m}^{des}$, and the reservation states $x_{ij}(k)$, $(i, j) \in \mathcal{E}$, $\forall k \in \mathcal{T}$, then, the OTA problem seeks to find the starting time s_m^* and the path p_m^* that minimize the difference between $d_{D_m}^{des} - s_m^*$. In other words, OTA finds the latest time that the m -th vehicle should start from its origin such that it will arrive at the destination on or before the desired arrival time.

To complete the problem formulation, let p_h , denoting the h -th path from source O_m to destination D_m , be defined as $p_h = (v_0^h, v_1^h), (v_1^h, v_2^h), (v_2^h, v_3^h), \dots, (v_{L_h-1}^h, v_{L_h}^h)$, where $v_j^h \in \mathcal{V}$ is the j -th visited node in the h -th path, with $v_0^h = O_m$, $v_{L_h}^h = D_m$, and L_h is the length of p_h . Additionally, let variable $d_{v_j^h}^h(s)$ be the arrival time at junction $v_j \in \mathcal{V}$ if a vehicle departs from its origin at $s \in \mathcal{T}$. Then, the arrival time to each node of the h -th

path can be expressed as:

$$\begin{aligned} d_{v_0^h}^h(s) &= s, \\ d_{v_1^h}^h(s) &= d_{v_0^h}^h(s) + c_{v_0^h, v_1^h}(d_{v_0^h}^h(s)) \\ &\vdots \\ d_{v_{L_h}^h}^h(s) &= d_{v_{L_h-1}^h}^h(s) + c_{v_{L_h-1}^h, v_{L_h}^h}(d_{v_{L_h-1}^h}^h(s)) \end{aligned} \quad (5)$$

Thus, for the m -th scheduled vehicle, the central-controller has to compute s_m^* and p_m^* that solve the problem (P₁) below:

$$(P_1) \quad \min_{s, p^h} J_T = d_{D_m}^{des} - d_{v_0}^h(s) \quad (6a)$$

s.t. Model Dynamics (1) – (5),

$$d_{D_m}^h(s) \leq d_{D_m}^{des}. \quad (6b)$$

Constraint eq. (6b) is added to ensure that vehicle m will not arrive after the desired time to the destination. Furthermore, the constraints in (1) - (5) define the model dynamics which consider each road segment's admissibility state.

Clearly, if at a given time there are no road segments that are at their capacity, the path that the m -th vehicle should follow is the shortest path from O_m to D_m and it should start at time $s_m^* = d_{D_m}^{des} - c_m^*$, where $c_m^* = l_m^*/u_f/T$ and l_m^* is the length of the shortest path. On the other hand, if there are links of the shortest path that are at their capacity, then the vehicle may have two options, either depart much earlier when all links of the shortest path are admissible (and arrive earlier and wait at the destination) or start a little earlier and take a longer path and arrive at the destination on time. Out of these possibilities, (P₁) will select the one that will allow the vehicle to depart as late as possible from the origin and still make it to the destination on time.

IV. OTA ALGORITHMIC SOLUTION

A solution to the OTA problem is obtained, based on dynamic programming [31], by constructing a time-space Graph (TSG). Algorithm 1 obtains an OTA solution taking into account the m -th request (O_m, D_m and $d_{D_m}^{des}$), the current number of reservations (i.e., $r_{ij}(k)$), and the current admissibility state (i.e., $x_{ij}(k)$) of each edge $(i, j) \in \mathcal{E}$ over the $k \in \mathcal{T}$ (Note, that the reservations and admissibility states can be easily expressed in the form of 2-D matrices with the columns representing link indices while the rows represent time indices). The constructed TSG is a directed acyclic graph where the space dimension contains all the indices of the nodes in $\mathcal{G}(\mathcal{V}, \mathcal{E})$ and the time dimension includes consecutive time slots in descending order (starting from the desired destination arrival time, $d_{D_m}^{des}$ and going backwards in time). In this way, each node in the space-time network represents the node where a vehicle arrives at the specific time-slot k . Once we have all nodes we can construct the TSG by inserting the edges that connect two nodes in reverse direction of the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ and backwards in

time on the TSG, with edges inserted only if there is a physical connection between the two adjacent nodes on \mathcal{G} , with its associated travel time cost reflecting the node on TSG that the vehicle will arrive.

The edge insertion procedure is accomplished based on two states. The first is the admissibility state, where an edge is considered as admissible if $x_{ij}(k) = 1$ according to Eq. (3). The second is the reachability state which defines if the newly inserted edge of TSG is reachable from the destination or not, meaning that there is a path that connects the destination node with the starting node of the related edge. Both states can be determined simply using variable, $d_{v_i}(k) v_i \in \mathcal{V}$, which denotes the arrival time at each node since the constructed graph does not contain a cycle. Therefore, in case that $d_{v_i}(k) = \infty$ then the edge is both not reachable and not admissible, while in the case that $d_{v_i}(k) = k$, the edge is both reachable and admissible. In case that both conditions are satisfied, (line 13 of Algorithm 1), an edge (i, j) is added on TSG, (lines 14-15). The whole process repeats until the time-slot that O_m becomes reachable, i.e. $d_{O_m}(k) < \infty$ for any i and k (edge (i, O_m)), (lines 8-20). In that case, the algorithm converges and returns the identified path by tracing back the nodes from O_m to D_m with the vehicle's departure time be equal with $s_m^* = d_{O_m}(k)$, (line 11).

Therefore, the solution of Algorithm 1 provides the m -th vehicle's departure time (i.e., s_m^*) and its route to follow (i.e., p_m^*). This information is utilized to make the appropriate route reservations on each road segment at the expected traversal times. We emphasize that, to compute route reservations for each vehicle; we use the constant parameter (i.e., \bar{c}_{ij}) which defines the number of time-slots that a vehicle is required to traverse a road segment. Hence, by knowing the m -th vehicle path to follow and its departure time the expected traversal time for each road-segment can be calculated assuming each segment requires \bar{c}_{ij} time slots to be traversed. Hence, the reservation status is updated during the expected transit times. In a similar manner, from $r_{ij}(k)$ we also update the admissibility state of each road segment (i.e., $(i, j) \in \mathcal{E}$).

Initially $r_{ij}(k) = 0$ and $x_{i,j}(k) = 1$ for all $(i, j) \in \mathcal{E}$ and $k \in \mathcal{T}$. Furthermore, it is assumed that vehicle requests are collected over an interval and are sorted in descending order based on the desired arrival time. Then, they are processed by the TSG algorithm sequentially starting from the latest desired arrival time to the earliest.

The algorithm 1 results in an optimal solution in the discretized space-time domain that it operates and executes in pseudo-polynomial time since the state space for the m -th request to be solved is not known until the first time that the algorithm visits the origin node in which $d_{O_m}(k) < \infty$, meaning that the algorithm converges with complexity $O((d_{D_m}^{des} - s_m^*)|\mathcal{E}|)$. Note that, in this algorithm we have to find a solution within a fixed interval $d_{D_m}^{des} T_{sub}$ where T_{sub} is the time a vehicle has submitted its request. If no such solution is found

Algorithm 1 TSG Alogrithm

```

1: Input:  $\mathcal{G}(\mathcal{V}, \mathcal{E})$ ,  $r_{ij}(k)$ ,  $O_m$ ,  $D_m$ ,  $x_{ij}(k)$ ,  $d_{D_m}^{des}$ ,  $c_{ij}(k) \forall k \in \mathcal{T}$ ;
2: Initialization:
3:  $t = d_{D_m}^{des}$ ;
4:  $d_{v_i}(t) = \infty, \forall t \in \mathcal{T}, v_i \in \mathcal{V}$ ;
5:  $d_{D_m}(t) = t, \forall t \in \mathcal{T}$ 
6:  $s_m^* = -\infty$ 
7: Algorithm Execution:
8: while  $t > s_m^*$  do
9:   for  $(i, j) \in \mathcal{E}$  do
10:    if  $((i == O_m) \text{ OR } (j == O_m)) \text{ AND } (d_{O_m}(t) > s_m^*)$  then
11:       $s_m^* = d_{O_m}(t)$ ;
12:    else
13:      if  $(x_{ij}(t) == 1) \text{ and } (d_{v_i}(t) < \infty)$  then
14:         $d_{v_j}(t) = d_{v_i}(t) - c_{ij}$ ;
15:         $previous[v_j][d_{v_j}(t)] = v_i$ ;
16:      end if
17:    end if
18:  end for
19:   $t = t - 1$ ;
20: end while
21: Trace back  $p_m^*$  and  $previous[O_m][s_m^*]$ ;
22: Reservations-Admissibility status Update:
23: Update Reservations( $p_m^*, s_m^*$ );
24: Update Admissibility( $p_m^*, s_m^*$ );
25: Output:  $p_m^*, s_m^*$ ;

```

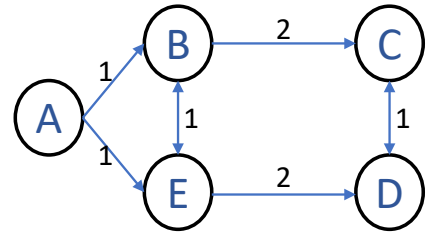


Fig. 1: An example network of $\mathcal{G}(\mathcal{V}, \mathcal{E})$.

the algorithm returns failure. The optimum solution can be derived considering that each node in TSG is reachable only if the reachability state of all predecessor nodes forming the minimum path from destination to that particular node starting from the destination at the corresponding time. Hence, if a node is reachable through path p , then all nodes forming p are also reachable (with the minimum cost) and the *optimal substructure property* applies [31]. The reachability of all states examined for decreasing k and thus the optimal solution is found at time $d_{O_m}^*$ which in turn represent the latest time at which a vehicle should start from O_m to reach D_m on-time or on earlier time considering the admissibility states of the edges in $\mathcal{G}(\mathcal{V}, \mathcal{E})$.

Illustrative Example

To better understand the proposed procedure, consider the example illustrated in Figure 1 (a) where edge lengths reflect the traversal times for specific road segment while the critical density of all edges in the graph is equal with 1 veh/edge. In this example, initially, no reservations are made and two vehicles request a path from A to D desiring to arrive at D at time slots $d_{D_1}^{des} = 9$ and $d_{D_2}^{des} = 10$, respectively. These requests are first sorted in descending order and thus the second request will be executed first by Algorithm 1.

Figure 2 (a) shows the TSG graph that is constructed by executing the first sorted request. The space dimension of each node indicates the junction index while the time dimension indicates the node created over time (with the time index starting from $d_{D_2}^{des} = 10$). The reachability of each node is assessed from variable $d_{v_i}(k)$ where for the case of D_2 for all time-slots is reachable and thus $d_{D_2}(k) < \infty \forall k \in \mathcal{T}$. As illustrated in the figure, in the first column edges emerge only from the destination node (e.g., D_2) since all other nodes are not reachable at time-slot $k = 10$. Similarly, in the second column edges emerge from nodes D_2 and C since at time-slot $k = 9$ they have been reached from the destination (e.g., D_2). Note that the black solid-line edges are those that are added to construct the TSG which has a feasible path from the destination to the specific node. As Fig. 2 (a) shows, at the fourth column is the first time index that the origin is reached with the algorithm converging at this time index. In that way, the grid-shaded nodes represent the nodes consisting of the path p_2^* (i.e., $A \rightarrow E \rightarrow D_2$, also denoted with the solid green line) where the latest departure time is $s_2^* = 7$. Next, the algorithm updates the reservations based on the obtained solution and the admissibility state of those edges changes as follows $x_{EA}(7) = x_{DE}(8) = x_{DE}(9) = 0$.

Subsequently, the algorithm re-executes the TSG procedure for the other request (e.g., $d_{D_1}^{des} = 9$) where the associated TSG is depicted in Figure 2 (b). For that case, the time index begins at the 9-th time-slot (according to vehicle's request) while due to reservations made from the first vehicle we can observe that the shortest path is not a feasible solution due to the non-admissible states that emerge for some particular time-slots. Hence, the first time that the originating junction is reached at the 5-th time-slot (fifth column) where two alternative solutions exist (denoted with green and red solid lines, respectively) and the algorithm selects as a solution the p_1^* (i.e., $A \rightarrow B \rightarrow C \rightarrow D$, green line) with the $s_1^* = 5$. Note that, both solutions have equivalent objective value $J_T = 4$ but, their length differs. More specifically, if the vehicle follows the green path, then the duration of its travel time will be 4 time-slots and will arrive at the destination exactly on time. Otherwise, if vehicle follows the red path, the duration of its travel time will be 3 time-slots and will arrive at the destination 1 time-slot

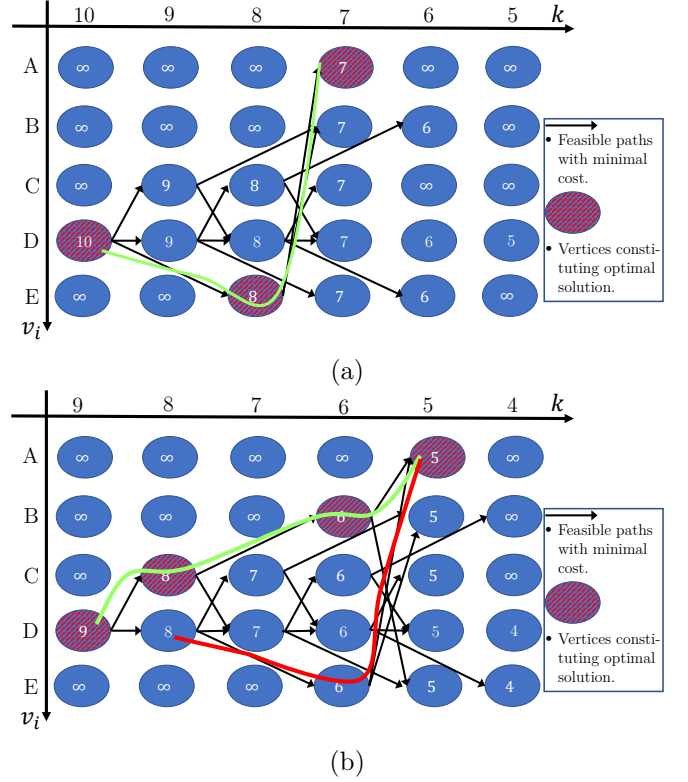


Fig. 2: The direct acyclic graph that generated from TSG procedure to solve (a) the first vehicles request (b) the second vehicle request.

earlier, thus the vehicle will wait at the destination for 1 time-slot. In other words, if the second vehicle would like to arrive at the destination at $k = 9$, it cannot leave from its origin at $k = 6$ because will produce congestion at the edge (E, D) at the time-slot $k = 8$.

V. PERFORMANCE EVALUATION

A. Simulation Setup

To evaluate the performance of the proposed solution we consider an 1.8 km^2 non-signalized urban region of the downtown San Francisco, as illustrated in Fig. 3. The area consists of 99 road junctions and 208 single-lane road segments with lengths varying from 100 m, to 400 m. The network was imported in the SUMO micro-simulator [32], and the Krauss car following model [33] was used. The car-following model parameters are set as follows: vehicle length 5 m, maximum speed 15 m/s, acceleration 2.5 m/s^2 , deceleration 4.5 m/s^2 , and minimum-gap-distance 2.5 m while no vehicle overtaking is allowed. The simulation time-step in SUMO was set to 0.1s while the time step of the algorithm was set equal to $T = 1\text{s}$. A critical density of $\rho_{ij}^C = 33 \text{ veh/km/lane}$ and a free-flow speed of $u_f = 10.0 \text{ m/s}$ is used to calculate each segment's travel time. All simulations were performed for 2 hours. The vehicle desired arrival times are requested only during the first simulation hour in which requests

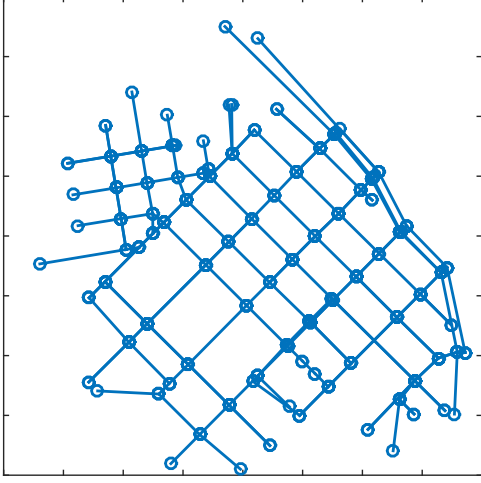


Fig. 3: San Francisco road network under consideration.

are uniformly distributed during 6 time intervals. Specifically, all desired arrival times were distributed uniformly in the time intervals of 8:00, 8:10, \dots 9:00 am. Hence, the network is loaded only during the first simulation hour while the second is used to empty the network and record some measurements. Finally, a total of ten Monte Carlos simulations are constructed (10 realizations) for varying flow rates from 1000 – 8000 *veh/h*.

The proposed algorithm described in Section IV is compared against the case where no control mechanism is applied (i.e., NC) where vehicles traverse from their origin to their destination along the shortest distance path. The travel time for each path is calculated assuming free-flow speed conditions while the departure time for each vehicle is assigned assuming that vehicles will take their shortest time path plus a uniformly distributed time budget (between 0-3 min) that is allocated to each vehicle to depart in advance to its shortest time path. Note that within the simulated network, the average trip length is around 1.5 min. Also, for the OTA results, it is assumed that all vehicles comply to the derived schedule. Furthermore, note that all vehicles schedules are obtained based only on the reservation estimates, rather than the actual state of each road segment, however, deviations between the reservations and actual state exists due to the underlying uncertainty involved in the microscopic simulation. Finally, in the results presented hereafter only vehicles that have completed their journeys during the 2-hour simulation time are considered.

B. Simulation Results

Figs. 4, and 5 show the vehicle average travel time and the number of vehicles that manage to reach their destination during the simulation time, respectively. The scattered plots in Figs. 4 depict the mean travel time of each realization, while the lines represent the mean travel time for all realizations. Similarly, the scattered plots in Fig. 5 depicts the number of vehicles that manage to

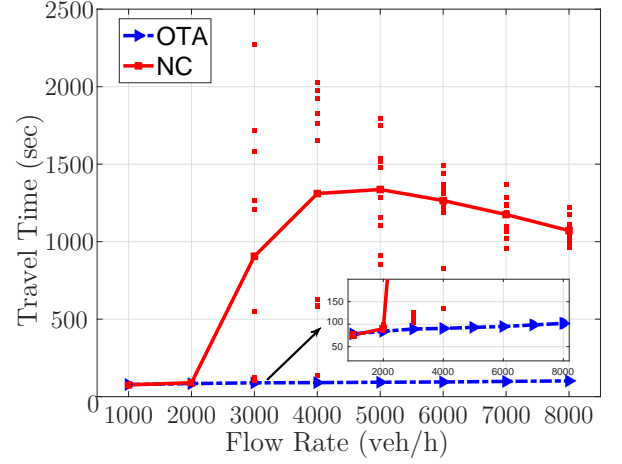


Fig. 4: Travel time for different simulation scenarios with varying demand flow rate.

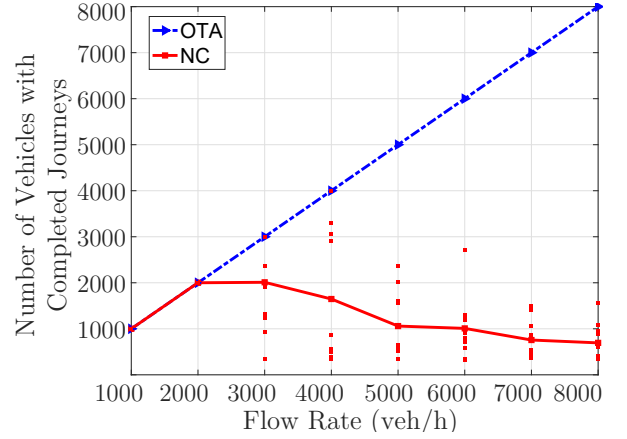


Fig. 5: Number of vehicles with completed journeys for different simulation scenarios with varying demand flow rate

reach their destination within the simulation time for each realization, while the lines represent the average number of vehicles that completed their journey. As illustrated in Figures 4 and 5 (as expected), at low flow rates both OTA and NC approaches perform equally well. However, at higher flow rates it is evident that OTA outperforms NC in terms of travel times since OTA avoids congestion. Additionally, Fig. 5 shows that for the case of OTA all vehicles manage to reach their destination, unlike the case of the non-controlled case where a significant number of vehicles cannot manage to reach their destination due to the formation of severe traffic congestion.

The scatterplot of Fig. 6 represents the number of vehicles that have reached their destination after their desired arrival time for all realizations obtained by each Monte Carlo run. Similarly, the scatterplot in Fig. 7 shows the average time by which vehicles exceeded their desired arrival time for each realization while the line represents the mean value for all realizations. Both figures

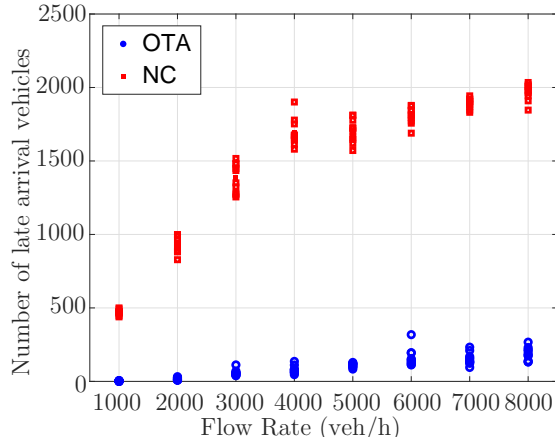


Fig. 6: Number of late arrival vehicles.

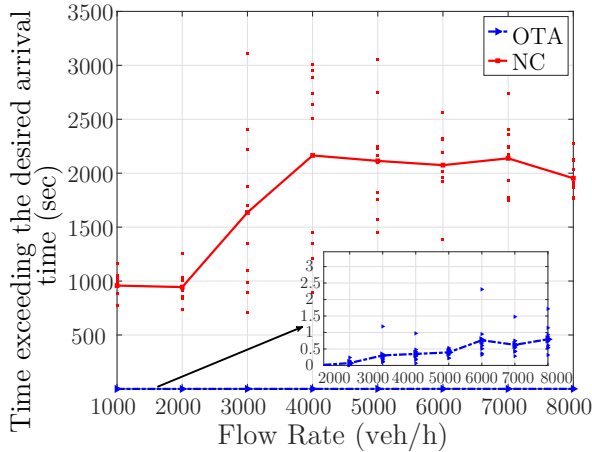


Fig. 7: The time that vehicles exceeding their desired arrival time.

clearly show that the OTA approach manages to schedule most of the vehicles on-time with those exceeding their desired arrival time have negligible delays. Note that, those delays occur due to the uncertainty in the micro-simulation environment. On the other hand, for the non-controlled case, it is observed that at the highest flow-rates the congestion is unavoidable with many vehicle arriving late while a large number of them cannot reach the destination within the simulation period.

Fig.8 illustrates the average waiting observed at the destination. In this figure, we measure the time that vehicles arrive earlier than their desired arrival time. As expected, with higher flow rates, the waiting time tends to be higher than with lower flow rates, meaning that travelers arrive much earlier than their desired time. This phenomenon occurs since an increase in demand results in more vehicles having to traverse the network in the presence of non-admissible segments. Even so, the vehicles arrive earlier than the desired time, while the waiting time at the destination is sustained within acceptable levels (around 3 min on average for the case of the highest flow rate scenario).

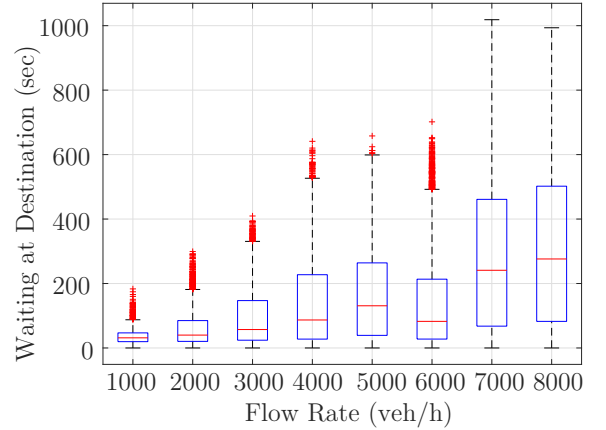


Fig. 8: The waiting time at destination for all considered flow rates measured as the difference between the desired and the actual arrival time.

VI. CONCLUSIONS

This work proposes a novel route-reservation scheme that aims to schedules vehicles to arrive at their destination at their desired time while at the same time, traffic congestion is eliminated by restricting the density of all road segments below their critical value. In this framework, the on-time arrival (OTA) problem is examined and solved by developing a dynamic programming algorithm that solves the OTA problem in pseudo-polynomial time. Simulation results demonstrate that under the proposed solution, the on-time arrival for all vehicle requests is guaranteed, while also demonstrate the substantial improvements gained in terms of network operation and the experienced travel times, especially during high flow rates.

Future research avenues include the examination of the proposed approached in a more aggregated manner where reservations are made over bigger subareas rather than over individual links, an action that can potentially reduce modeling complexity and improve its computational efficiency. Furthermore, another possible future avenue is not to solve the problem on a vehicle-by-vehicle basis but to group requests and solve the related problem using flow-based optimization techniques. Another important aspect that requires investigation is the fairness of the proposed approach. In this work, vehicle request are sorted in descending order, something that may benefit vehicles that desire to arrive in later times as the algorithm serves them first, an action that maybe unfair for those that wish to arrive earlier.

REFERENCES

- [1] E. D. Miller-Hooks and H. S. Mahmassani, "Least expected time paths in stochastic, time-varying transportation networks," *Transportation Science*, vol. 34, no. 2, pp. 198–215, 2000.
- [2] Y. M. Nie and X. Wu, "Shortest path problem considering on-time arrival probability," *Transportation Research Part B: Methodological*, vol. 43, no. 6, pp. 597 – 613, 2009.

- [3] S. Samaranayake, S. Blandin, and A. Bayen, "A tractable class of algorithms for reliable routing in stochastic networks," *Procedia-Social and Behavioral Sciences*, vol. 17, pp. 341–363, 2011.
- [4] C. F. Daganzo, "Urban gridlock: macroscopic modeling and mitigation approaches," *Transportation Research Part B: Methodological*, vol. 41, no. 1, pp. 49–62, 2007.
- [5] W. S. Vickrey, "Congestion theory and transport investment," *The American Economic Review*, pp. 251–260, 1969.
- [6] J. R. Correa, A. S. Schulz, and N. E. Stier-Moses, "Selfish routing in capacitated networks," *Mathematics of Operations Research*, vol. 29, no. 4, pp. 961–976, 2004.
- [7] S. Çolak, A. Lima, and M. C. González, "Understanding congested travel in urban areas," *Nature communications*, vol. 7, no. 10793, 03 2016.
- [8] C. F. Daganzo, "The uniqueness of a time-dependent equilibrium distribution of arrivals at a single bottleneck," *Transportation science*, vol. 19, no. 1, pp. 29–37, 1985.
- [9] E. J. Gonzales and C. F. Daganzo, "The evening commute with cars and transit: Duality results and user equilibrium for the combined morning and evening peaks," *Procedia-Social and Behavioral Sciences*, vol. 80, pp. 249–265, 2013.
- [10] C. Menelaou, S. Timotheou, P. Kolios, C. G. Panayiotou, and M. P. Polycarpou, "Controlling road congestion via a low-complexity route reservation approach," *Transportation Research Part C: Emerging Technologies*, vol. 81, pp. 118–136, 2017.
- [11] C. Menelaou, S. Timotheou, P. Kolios, C. G. Panayiotou, and M. M. Polycarpou, "Minimizing traffic congestion through continuous-time route reservations with travel time predictions," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 1, pp. 141–153, March 2019.
- [12] C. Menelaou, S. Timotheou, P. Kolios, and C. G. Panayiotou, "Improved road usage through congestion-free route reservations," *Journal of the Transportation Research Board*, vol. 2621, pp. 71–80, 2017.
- [13] I. Kaysi, M. Ben-Akiva, and H. Koutsopoulos, "An integrated approach to vehicle routing and congestion prediction for real-time driver guidance," Ph.D. dissertation, Doctoral dissertation, Massachusetts Institute of Technology Cambridge, Mass, 1993.
- [14] D. E. Kaufman and R. L. Smith, "Fastest paths in time-dependent networks for intelligent vehicle-highway systems application*," *Journal of Intelligent Transportation Systems*, vol. 1, no. 1, pp. 1–11, 1993.
- [15] L. Yang and X. Zhou, "Optimizing on-time arrival probability and percentile travel time for elementary path finding in time-dependent transportation networks: Linear mixed integer programming reformulations," *Transportation Research Part B: Methodological*, vol. 96, pp. 68–91, 2017.
- [16] L. Du, S. Chen, and L. Han, "Coordinated online in-vehicle navigation guidance based on routing game theory," in *Transportation Research Board 94th Annual Meeting*, no. 15-3613, 2015.
- [17] H. Frank, "Shortest paths in probabilistic graphs," *Operations Research*, vol. 17, no. 4, pp. 583–599, 1969.
- [18] V. Knoop, S. Hoogendoorn, and J. Van Lint, "Routing strategies," in *Transportation Research Board 94th Annual Meeting*, no. 15-3613, 2015.
- [19] L. Immers and S. Logghe, *Traffic flow theory*, Department of Civil engineering Section Traffic and Infrastructure, Belgium, 05 2003, course H 111.
- [20] N. Geroliminis and C. F. Daganzo, "Macroscopic modeling of traffic in cities," in *TRB 86th annual meeting*, no. 07-0413, 2007.
- [21] M. Yildirimoglu, M. Ramezani, and N. Geroliminis, "Equilibrium analysis and route guidance in large-scale networks with mfd dynamics," *Transportation Research Part C: Emerging Technologies*, vol. 59, pp. 404–420, 2015.
- [22] Y. Ji and N. Geroliminis, "On the spatial partitioning of urban transportation networks," *Transportation Research Part B: Methodological*, vol. 46, no. 10, pp. 1639–1656, 2012.
- [23] Y. Ji and N. Geroliminis, "Large-scale urban networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2141–2154, 2015.
- [24] A. Kouvelas, M. Saeedmanesh, and N. Geroliminis, "Feedback perimeter control for heterogeneous urban networks using adaptive optimization," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*. IEEE, 2015, pp. 882–887.
- [25] A. Kouvelas, D. Triantafyllos, and N. Geroliminis, "Enhancing model-based feedback perimeter control with data-driven online adaptive optimization," *Transportation Research Part B: Methodological*, vol. 96, pp. 26–45, 2017.
- [26] R. De Feijter, J. J. Evers, and G. Lodewijks, "Improving travel time reliability by the use of trip booking," in *Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE*, vol. 1. IEEE, 2003, pp. 205–210.
- [27] Y. Zhao, K. Triantis, D. Teodorović, and P. Edara, "A travel demand management strategy: The downtown space reservation system," *European Journal of Operational Research*, vol. 205, no. 3, pp. 584–594, 2010.
- [28] K. Liu, E. Chan, V. Lee, K. Kapitanova, and S. H. Son, "Design and evaluation of token-based reservation for a roadway system," *Transportation Research Part C: Emerging Technologies*, vol. 26, pp. 184–202, 2013.
- [29] W. Liu, H. Yang, and Y. Yin, "Efficiency of a highway use reservation system for morning commute," *Transportation Research Part C: Emerging Technologies*, vol. 56, pp. 293–308, 2015.
- [30] N. Geroliminis and J. Sun, "Properties of a well-defined macroscopic fundamental diagram for urban traffic," *Transportation Research Part B: Methodological*, vol. 45, no. 3, pp. 605–617, 2011.
- [31] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, et al., *Introduction to algorithms*. MIT press Cambridge, 2001, vol. 2.
- [32] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo-simulation of urban mobility-an overview," in *SIMUL 2011, The Third International Conference on Advances in System Simulation*, 2011, pp. 55–60.
- [33] S. Krauss, P. Wagner, and C. Gawron, "Metastable states in a microscopic model of traffic flow," *Physical Review E*, vol. 55, no. 5, p. 5597, 1997.