A Framework for Centralized Traffic Routing in Urban Areas

Matyáš Švadlenka\textsuperscript{1,2}, Lukáš Chrpa\textsuperscript{1}
\textsuperscript{1}Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague
\textsuperscript{2}Faculty of Electrical Engineering, Czech Technical University in Prague
svadlmat@fel.cvut.cz, chrpaluk@cvut.cz

Abstract

Dealing with the ever-increasing demand for traffic management is one of the main challenges of the 21st century. The issue is much more apparent in urban areas during rush hours. Traffic congestion causes economic losses due to delays and increased fuel consumption and, on top of that, is a major health risk. Intelligent centralized traffic routing is an important concept aiming at reducing traffic congestion in urban areas by more effectively utilizing road networks.

In this demo, we present a framework that, in a nutshell, integrates techniques for intelligent centralized traffic routing into the well-known SUMO simulator, so these techniques can be evaluated in realistic settings on real/realistic datasets. In particular, the framework automatically identifies “problematic” urban regions by analyzing historical traffic data, then simplifies the road networks by precomputing promising routes (for each considered traffic flow), and finally, leverages a planning-based approach to generate routes. Our framework is evaluated on a real dataset from Dublin’s metropolitan area.

1 Introduction

Nowadays, traffic in urban areas is one of the major economic problems due to losses from traffic accidents and travel delays. In rush hours, the problem with traffic jams, road congestion, and travel delays is far more apparent. To give an example, the cost of congestion in London has exceeded £5 billion in 2020 in lost time and fuel consumption\textsuperscript{1} and has become a major health threat [Chang \textit{et al}., 2019]. With the continuing growth of global urbanization, the problem of traffic congestion is expected to be exacerbated. Hence there is a need for intelligent traffic management techniques that can, at least to some extent, mitigate the problem. The introduction of highly innovative techniques such as \textit{Connected Autonomous Vehicles (CAV)} has the potential to revolutionize the field as we have more options for designing intelligent traffic management techniques [Vallati and Chrpa, 2018].

Effective traffic management has to consider a global perspective on the controlled (urban) region to make good decisions about traffic signal configurations or vehicle routes. Distributed (decentralized) approaches for traffic signal control [Xie \textit{et al}., 2012] or dynamic vehicle routing [Wang and Niu, 2019] achieved promising results and are now being used in practice. Centralized approaches, in contrast, take the (global) perspective of a traffic control center and aim to optimize traffic for the whole controlled region. Notably, centralized approaches allow us to consider both the dynamic system optimal (DSO) principle [Merchant and Nemhauser, 1978] and the dynamic user optimal (DUO) principle [Friesz \textit{et al}., 1989] and to trade-off in between them. Centralized approaches, based on automated planning [Pozanco \textit{et al}., 2021; Gulić \textit{et al}., 2016], have been used, in cooperation with the SUMO simulator [Lopez \textit{et al}., 2018]. The generation of traffic light control strategies has also been tackled by using hybrid PDDL+ and numeric planning [Antoniou \textit{et al}., 2019; Vallati \textit{et al}., 2016; McCluskey and Vallati, 2017].

In the traffic routing problem, we face the alternative of decentralized and centralized computation mechanisms [Shahi \textit{et al}., 2020]. The former refers to accomplishing routing or rerouting tasks in closed vehicles or infrastructures, with benefits ranging from leveraged computation capacity to communication loads [Du \textit{et al}., 2014; Claes \textit{et al}., 2011]. The centralized approaches aim to provide the optimal routes for each vehicle from the global perspective, with the clear benefit of having a holistic vision of the controlled region. Yamashita \textit{et al}., [2005] proposed a method that collects data about intended routes of the vehicles, based on the collected data it predicts future traffic intensity in the area, and the prediction is broadcasted back to the vehicles that update their routes according to the prediction. Recently, planning-based techniques for centralized traffic routing have been proposed and shown promising results [Chrpa \textit{et al}., 2019; Vallati \textit{et al}., 2021; Svadlenka and Chrpa, 2023; Svadlenka \textit{et al}., 2023].

In this demo, we present a framework that integrates techniques for intelligent centralized traffic routing into the well-known SUMO simulator [Lopez \textit{et al}., 2018] (initial work on the framework was recently published [Svadlenka and Chrpa, 2023]). The framework aims to evaluate central-
ized traffic routing techniques in realistic traffic conditions on real/realistic datasets. The framework provides the following functionality: (i) it automatically identifies “problematic” urban regions by analyzing historical traffic data, (ii) simplifies the road networks by precomputing promising routes (for each considered traffic flow) [Svadlenka et al., 2023], and (iii) leverages a PDDL-based planning approach to generate vehicle routes [Chrpa et al., 2019]. Our framework is (preliminarily) evaluated on a real dataset from Dublin’s metropolitan area [Gueriau and Dusparic, 2020].

2 Centralized Traffic Routing Framework

The problem of effective centralized traffic routing is computationally demanding and hence it might not be feasible to tackle it in the whole metropolitan urban areas [Chrpa and Vallati, 2023]. Thus it is necessary to simplify the problem by focusing only on the most critical traffic flows in the most critical parts of (the whole) network.

The architecture of our framework for centralized traffic routing, depicted in Figure 1, is divided into three main parts: region identification component (left), smart routing component (right), and the SUMO simulator itself [Lopez et al., 2018] (bottom left). Note that the architecture extends the one presented in [Svadlenka and Chrpa, 2023] by considering existing (real) datasets from larger metropolitan areas.

The region identification component processes a given SUMO scenario to identify “problematic” regions that are prone to traffic congestion. It analyzes historical statistical traffic data from the scenario (travel time, arrived and departed vehicles, etc.) in the considered road network. For each road segment, we compute Congestion index as $1 - \frac{t_f}{t_a}$ (where $t_f$ is a free flow travel time, and $t_a$ is an actual travel time). Then, we apply Gravitational Clustering [Wright, 1977] to identify the “problematic” regions that are prone to congestion. Gravitational Clustering is parametrized by the number of iterations and by the “merging radius”. The identified “problematic” regions, as the result of the clustering, can be visualized in the framework, so an expert can either approve the result (i.e., the regions are reasonable) or change the parametrization of the clustering and generate a new result. The approved regions are then converted with Netedit (which is a part of the SUMO framework) into the graph-like format that we use in the routing component.

The smart routing component periodically takes vehicles from the scenario in advance of their arrival (30 seconds in this work) and filters out those that do not drive through any of the identified regions. For each episode that is specified by a given identified “problematic” region and a given 30-second timespan, we initially identify traffic flows specified by the point of origin, where the vehicles enter the region, and the point of destination, where the vehicles leave the region. For each traffic flow, we in a preprocessing step identify a road sub-network containing “smart” routes that are diverse enough and within a specified suboptimality bound [Svadlenka et al., 2023]. In particular, a variant of the A* algorithm [Hart et al., 1968] is used to identify routes that are within the specified suboptimality bound [Svadlenka and Chrpa, 2023] and then by using the DBSCAN algorithm [Ester et al., 1996] we cluster the routes based on their similarity, which is measured by Jaccard Index (that measures similarity of sets). For each cluster, we consider only the shortest route as it achieved the most promising results [Svadlenka et al.,]
2023]. Each episode is then compiled into a PDDL model that, in a nutshell, encodes a planning task in which the vehicles, from considered traffic flows, have to be routed through the region such that the traffic intensity on road segments is minimized. Note that vehicles of each traffic flow can be routed only on pre-identified “smart” routes in the preprocessing and other (non-routed) vehicles are taken into account in the form of the occupancy of the road segments. More details about the PDDL model can be found in [Chrpa et al., 2019]. To solve a PDDL planning task (to generate vehicle routed through the region in the given episode) we use the Mercury planner [Domshlak et al., 2015]. To solve an episode, we imposed a time limit of 27 seconds (for “smart” routing and Mercury together) and, in the case, no solution is produced, the original routes of the vehicles are assigned.

In the final step, all planned vehicle routes are converted back into SUMO’s representation (by replacing the parts of the routes of the vehicles going through the “problematic” regions). SUMO hence simulates how our centralized traffic routing affects the traffic both in the “problematic” regions as well as in the whole metropolitan area. The results of the simulation are exported into a file consisting of various statistics (e.g., average travel time, average travel distance).

### 3 Demonstration & Preliminary Results

The demonstration will focus on showing the functionality of our framework that aims at improving traffic conditions by centralized traffic routing. Resources and video demonstration can be found in the following repository².

For the demonstration, we consider a dataset from the metropolitan area of Dublin [Gueriau and Dusparic, 2020]. To emphasize the benefit of centralized traffic routing techniques, we have considered data from morning rush hour (between 7 am and 9 am), in which the traffic in the region is the most intense. In total, the data contain 45 181 vehicles out of which 6 177 were routed by centralized routing.

In our experiments, we have parametrized the Gravitational Clustering algorithm we use to identify “problematic” regions such that the merging radius was set to 8 meters (i.e., how close the clusters have to be to be merged into a larger cluster) and the number of iterations was set to 150. To speed up the convergence of the algorithm we have considered the weights of initial clusters (each cluster corresponds to a single road segment) as 10 times the value of the Congestion Index for particular road segments. For the “smart” route preprocessing, we have generated at most 3000 routes that are at most 1.2 times longer than the shortest route (for each traffic flow). Only traffic flows, in which the preprocessing identified at least 10 routes such that each route consisted of at least 3 road segments, were further processed by the DBSCAN clustering algorithm that was parametrized by setting the minimum distance (measured by Jaccard Index) between “diverse” routes (that cannot be in the same cluster) to 0.26 and the minimum cluster size of 2 routes. The routing of vehicles (of considered traffic flows) is done by an off-the-shelf planner Mercury [Domshlak et al., 2015] (as mentioned in the previous section). The experiments were run Intel i7-8700 (3.2 GHz) and 4GB of RAM.

We have identified three “problematic” regions (in the whole Dublin metropolitan area) in which we have applied our Centralised Traffic Routing (CTR) approach (they are shown in the video presentation). Table 1 shows the results of the simulation (in a whole area). As can be seen, the use of CTR (in “problematic” regions) can improve the average travel time by about 22.5% with respect to the original traffic data. Of course, such a plain comparison might not reflect reality as many drivers routinely use modern SATNAV systems that dynamically adjust the route based on current traffic conditions. Hence, we have considered the “DUO” option in SUMO (setting the rerouting for each 30 seconds to align with the duration of the planning episode in the CTR approach) that, roughly speaking, considers the use of such SATNAV systems. Unsurprisingly, DUO improves the results by further 35%. A plain combination of CTR and DUO led to the same results as SUMO always replaced the routes generated by CTR with the DUO ones. Interestingly, when focusing only on the central “problematic” region, the average travel time for CTR was about 10% better than for DUO.

### 4 Conclusions

This demonstration presents a framework that, in a nutshell, integrates centralized traffic routing techniques into the SUMO simulator [Lopez et al., 2018] in order to evaluate the techniques on real datasets. The framework has a modular architecture, so it allows embedding a new technique (e.g. for centralized traffic routing) without considerable implementation overheads.

The preliminary results achieved in the Dublin metropolitan area (in the morning rush hour) indicate that the use of centralized traffic routing techniques might complement the existing decentralized techniques that are used in modern SATNAV systems.

Future work will include the development and integration of more efficient centralized routing techniques, effective combination with DUO techniques, and integration of existing approaches for traffic light control (e.g. see [McCluskey and Vallati, 2017]).

### Table 1: Results of the simulation on Dublin dataset (rush hours). DUO (Dynamic User Optimal) refers to decentralized dynamic routing. CTR refers to our Centralised Traffic Routing approach. All quantities are averages per vehicle.

<table>
<thead>
<tr>
<th></th>
<th>travel distance (m)</th>
<th>speed (m/s)</th>
<th>travel time (s)</th>
<th>waiting time (s)</th>
<th>time loss (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>1917.04</td>
<td>6.59</td>
<td>749.68</td>
<td>482.92</td>
<td>559.17</td>
</tr>
<tr>
<td>CTR</td>
<td>1909.99</td>
<td>6.76</td>
<td>581.55</td>
<td>324.42</td>
<td>397.48</td>
</tr>
<tr>
<td>DUO</td>
<td>1812.64</td>
<td>7.67</td>
<td>326.30</td>
<td>116.55</td>
<td>161.31</td>
</tr>
</tbody>
</table>

²https://github.com/lchrpa/CTR-IJCAI24-demo.git
Acknowledgments

This research is supported by Czech Science Foundation (project no. 23-05575S), by the European Union under the project ROBOPROX (reg. no. CZ.02.01.01/00/22_008/0004590), and by the Grant Agency of the Czech Technical University (project no. SGS24/115/OHK3/2T/37).

References


