Dynamic Vehicle Path-Planning in the Presence of Traffic Events

Tobias Meuser*‡, Ioannis Stavrakakis†, Antonio Fernández Anta‡, Ralf Steinmetz*
*Multimedia Communications Lab (KOM), Technische Universität Darmstadt, Darmstadt, Germany
Email: {firstName.lastName}@KOM.tu-darmstadt.de
†National and Kapodistrian University of Athens, Greece
Email: ioannis@di.uoa.gr
‡IMDEA Networks Institute, Madrid, Spain
Email: antonio.fernandez@imdea.org

***Abstract*—Advanced Driver Assistance Systems require a tremendous amount of sensor information to support the driver's comfort and safety. In particular, systems that provide (good) route options to a vehicle rely on information, such as traffic jams and road blockages, which is sensed by other (possibly distant) vehicles and distributed by a central server. This information is clearly dynamic and may be invalid by the time the vehicle arrives at the affected location. In this work, we develop an innovative approach to determine optimal routes (minimizing the costs like travel-time to their destination) for vehicles whose original route is adversely impacted by a (severe) road event. The route is in principle reassessed just before each upcoming road intersection (decision-point), considering updated information about the road event and an estimate of the remaining lifetime of the road event. A set of recursive equations is developed that yields the optimal decision for each vehicle at each decision-point, accounting for aspects such as the vehicle's destination, driver characteristics, etc. In practice, the decision may be taken by the vehicle itself (if all the needed information is transferred to it), or by the remote server and be communicated to the vehicle (if all needed private vehicle information is transferred to the server). A discussion is presented, along with some ideas, their assessment, and associated tradeoffs, aiming at reducing communications costs. Simulations show that our approach adapts to the considered event and finds routes of similar quality as a full-knowledge approach with limited communication overhead.**

***Index Terms*—demand-driven, vehicular networks**

I. Introduction

Advanced Driver Assistance Systems (ADAS) rely on sensor information to improve the driver's comfort and safety [1]. In today's vehicles, this information is mainly obtained from onboard sensors, and provides limited awareness, due to their limited range and physical constraints (like obstacles and weather conditions). Extended awareness beyond the local perception is today possible through the exchange of (local) vehicle's information over a wide area network such as the cellular network. Such extended awareness can support dynamic mechanisms to establish the current best paths by considering real-time road conditions, such as traffic jams, accidents and road blockages, beyond the current vehicle's locality.

In this paper, we develop an innovative approach to determine optimal paths (minimizing the costs to their destination) for vehicles whose original path is adversely impacted by a (severe) road event. The path is in principle reassessed at the end of each road segment just before each upcoming road intersection (decision-point), taking into account updated information about the road event and an estimate of the remaining lifetime of the road event (if still active). We model the road network as a graph whose vertices correspond to the end of a road segment just before entering the intersection (referred to as the associate intersection). An edge exists between two vertices for each road segment that exists between two associated intersections; no edge exists between vertices associated with the same intersection. For a given vehicle type, destination, and road segment, a set of recursive equations are developed yielding the cost of the best path from the specific decision-point (vertex in the graph) to the destination, given: (i) the real state of the road event reported by other vehicles in the network, (ii) a probabilistic knowledge about the remaining lifetime of the event, and (iii) the expected remaining travel-time of the vehicle to the affected road segment. At a decision-point (at the end of the current segment), the vehicle will select the adjacent road segment inducing the minimum cost from that to the destination.

The state of all road segments and road events in an area is assumed to be collected by a central server which can compute all optimal paths from any decision-point to any destination and for any type of vehicle (or driver). If a vehicle provides private information to the server (destination, type, etc), then the server can determine if a better path is available and notify the specific car accordingly. Alternatively, the server could provide all relevant road segment states to a vehicle, and the vehicle could determine its optimal path on its own. The latter approach requires substantial communication resources and might overload the cellular network [2] without even ensuring that a better path does exist. A discussion is presented to this end along with some ideas, their assessment, and associated tradeoffs, aiming at reducing communications costs.

The remainder of this paper is structured as follows: In Section II, we provide an overview of the advances made in the vehicular routing problem. In Section III, we describe the considered scenario. In Section IV, we describe our modeling of the timeliness of information in the network and the derivation of optimal paths considering this timeliness. Section V then uses the knowledge of the vehicle's decision-making to improve the communication between vehicles and backend. We evaluate our approach in Section VI, which focuses on the improvement of our approach compared to traditional routing techniques and the achieved reduction in data traffic. Finally, we provide some conclusions in Section VII.

II. Related Work

Vehicular path-planning is a significant challenge due to the unexpected appearance and disappearance of road events like jams and accidents. To alleviate the impact of these road events, vehicles exchange information via wide-range area networks, like the cellular network, in combination with local communication capabilities like Wifi [3], [4] to improve their planning. According to Gonzalez et al. [5], there are three categories of planning for vehicles, *Global Planning*, *Behavioral Planning*, and *Local Planning*. While *Local Planning* and *Behavioral Planning* are related to the specific behavior of the vehicle on the road, *Global Planning* deals with the process of path-planning towards the destination of the vehicle.

In this work, we focus on the *Global Planning* of the routes of the vehicles. In the literature for *Global Planning*, road events (influencing the goodness of a route) are either considered by the cost function [6] or indirectly by measuring the influence on the route like higher travel-times. In general, previous research proposes path-planning approaches for individual vehicles [7]–[9] or the whole network [3]. In [10], the authors propose a path-planning approach to find optimal paths. This work is then extended by Guo et al. [11] to consider the appearance of events in the road network. However, to our knowledge, the impact of event timeliness on path-planning approaches has not been analyzed, i. e., the sudden disappearance of events. The lifetime of road events itself is investigated by [12], but the authors do not use the gathered knowledge for path-planning purposes. In our work, we want to focus on this gap by developing an approach to use the expected event lifetime to improve path-planning.

We use the path-planning for the optimization of communication by only communicating information required for path-planning. A similar approach has been used in [13], in which the authors updated only parts of the map that influenced the path-planning. However, our approach is capable of handling short-lifetime events and reducing the load by considering the disappearance of events. This disappearance poses new research challenges towards the requirements of information. These requirements are highly context-sensitive, which is not considered by the current literature. Most current literature aims to reduce communication latency as much as possible [14], [15] without considering the applications' requirements. For communication mechanisms in vehicular networks, common metrics like delay are used [16], [17]. This is not always appropriate, as road-related information need to be provided in-time to enable vehicular path-planning.

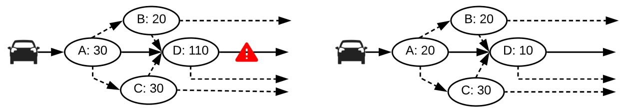
III. Scenario Overview

In this section, we provide an overview of our system and our assumptions. We consider a road scenario in which vehicles travel along their planned optimal paths $\vec{\phi} = (s_v, \ldots, s_d)$, defined as a sequence of road segments from origin s_v to destination s_d.

To find the optimal path, we use a directed graph whose vertices correspond to the end of a road segment just before entering the intersection (referred to as the associate intersection). An edge exists between two vertices for each road segment that exists between two associated intersections; no edge exists between vertices associated with the same intersection. Notice that one intersection may be present in multiple edges in the graph. The cost of each edge in the graph is determined by the cost of crossing the intersection and traversing the associated segment (with the edge) completely. We consider the costs of crossing intersections to account for possibly different times when traversing an intersection. For instance, a vehicle might be crossing the intersection rapidly if it goes straight, but may take more time if it turns left. We assume that every vehicle is aware of this graph representation of the road network, including all edge costs and average travel-times for an event-free road network.

In our road network, we assume that there can be an event blocking the normal traffic flow. Let e_s^t denote an event of type t (where t may refer to a traffic jam, an accident, etc) that occurs in segment s; let s_e denote the segment containing an event e. Additional event-specific meta-information (like the average lifetime $\bar{l}_t$) is assumed to be available. In our model developed in Section IV, we assume the appearance and disappearance of events to be instantaneous. Due to the occurrence of the event, the originally optimal paths are likely not to be optimal anymore. To adjust its path, each vehicle requires additional information like either road state or path suggestions, which it receives via its cellular network interface from a central server. This server is assumed to become aware of the event appearance and disappearance instantaneously. One way for the server to detect the appearance of an event is utilizing the sensor-capabilities of the vehicles at the event's location, which then share the event information, like type and location, with the server. The same holds for disappearing events, as previously notified vehicles at the event location can detect the absence of an event while passing by the event's location.

The transmission of messages from the server to the vehicles uses cellular bandwidth, which is considered to be limited. To improve communication efficiency, the server needs to be aware of the planned path of every vehicle in the network: Based on the server's knowledge about the state of an event and the planned paths of the vehicles, it can determine the vehicles whose original planned path is affected by the event; these vehicles are the *affected vehicles*. This is done by using the updated road segment cost information that also captures the impact of the event on various paths. The server will then provide Critical Routing Update Information (CRUI) by



(a) Each decision-point describes the costs of the optimal path from the decision-point to the destination if the event is active.

(b) Each decision-point describes the costs of the optimal path from the decision-point to the destination if the event goes inactive.

Fig. 1: Decision-making graph for a vehicle. The solid line is the planned path of the vehicle. Each ellipse is a possible decision-point. The red warning sign depicts the event.

communicating to affected (only) vehicles: (i) either the costs of all road segments that are relevant to the destination of the affected vehicle and the vehicle will calculate optimal new paths also considering any relevant private information; or (ii) the new optimal path for the affected vehicle, if all relevant private information is available to the server.

IV. Vehicular Decision-Making

In this section, we develop our innovative approach to determine optimal paths (minimizing the costs to their destination) for vehicles whose original path is adversely impacted by a (severe) road event. The path is in principle reassessed at the end of each road segment just before each upcoming road intersection (decision-point), taking into account updated information about the road event and an estimate of the remaining lifetime of the road event (if still active). Figure 2 shows an example of decision-points (A, B, C, D) associated with a road network and the costs (in [·]) to get to the destination from these points, in case of an event (Figure 4a) or its absence (Figure 4b). In this section, we assume that the vehicle has received the all relevant road update information related to the event and describe the proposed decision-making mechanism that establishes its optimal path given the presence of the event.

A. Path Cost Calculation

In the following, we will describe the calculation of the costs to reach the destination starting from a certain decision-point and given an active event. For a given vehicle type, destination, and road segment, a set of recursive equations are developed yielding the cost of the best path from the specific decision-point (vertex in the graph) to the destination. This cost is denoted as $c^+(e_s^t, s_i, v, s_d)$, where e_s^t is the event triggering the decision, s_i is the road segment terminated by the respective decision-point, v are the vehicle/driver properties influencing the decision, and s_d is the current destination of the vehicle. The cost may, for example, be measured in travel time, fuel consumption, or traveled distance. Additionally, there might be other cost functions which consider other driver-related aspects.

The calculation of the cost $c^+(e_s^t, s_i, v, s_d)$ depends on the state (active/inactive) of the event in the future. As the event lifetime l_t is a random variable, the cost $c^+(e_s^t, s_i, v, s_d)$ is also a random variable; let $\overline{c}^+(e_s^t, s_i, v, s_d)$ denote the expectation. A vehicle finds an optimal path by reevaluating its decision at every possible decision-point and selecting the decision with the lowest expected costs $\overline{c}^+(e_s^t, s_i, v, s_d)$. The decision taken at each such decision-points balances the relation between the lower costs (if the event stays active) and the costs for an unnecessary detour (if the event goes inactive) optimally for the given event-specific lifetime.

As events may turn inactive at any decision point, the cost of reaching the destination from any such point given that the event turned inactive will also be needed; this is denoted $c^-(s_i, v, s_d)$ at the decision-point s_i (when clear from context, we will omit e_s^t and v). Both $c^-(s_i, v, s_d)$ and $c^+(e_s^t, s_i, v, s_d)$ capture the costs to get from the current decision-point s_i to the vehicles destination s_d.

While traveling to its destination at s_d, the vehicle will possibly traverse several decision-points. The two costs $c^-(s_i, s_d)$ and $c^+(s_i, s_d)$ describe the costs that the vehicle encounters when traversing segment s_i and taking the optimal path from there. In this case, the optimal path is defined as the path with the lowest expected costs $c^+(s_i, s_d)$ (if the event is active) or $c^-(s_i, s_d)$ (if the event is inactive). As the cost at a decision-point s_i depends on the future decisions of the vehicle, the two cost functions are recursive functions based on the accessible roads from s_i. We need two assumptions for the cost of an edge: First, we define that both costs for the vehicle are 0 if it has reached its destination at s_d. Thus, $c^-(s_d, s_d) = 0$ and $c^+(s_d, s_d) = 0$. Without loss of generality, we assume that destination-leading segment s_d is not associated with the event under consideration. Second, we need to account for the higher costs of a segment with an active event. The cost of this edge (if the event is active) is increased by the event-type-specific cost value C^t. The calculation of this event-specific cost value is discussed in Section IV-C. Thus, $c^+(s_e, s_d) = c^-(s_e, s_d) + C^t$ (notice that s_e is the segment affected by the event e_s^t). When investigating our graph of the road network, in which the segments refer to the vertices, this increases the costs by C^t (which is assumed to be large) every time the affected vertex (segment) is traversed.

We start with the definition of $c^-(s_i, s_d)$ for the general segment s_i, as $c^+(s_i, s_d)$ depends on it. That is, as the event might turn inactive in the future if its lifetime is exceeded. However, as we cannot be sure about the exact lifetime of the event, we consider the probability of the event to disappear while we are traveling towards the event's location. Additionally, we assume that an inactive event cannot become active again. Equation 1 shows the calculation of $c^-(s_i, s_d)$. The costs are the sum of the costs $R(s_i, s_j) > 0$ to get to the next edge s_j and the costs $c^-(s_j, s_d)$. The function "neighbors(s_i)" returns the set of accessible segments from the decision-point s_i.

$$c^-(s_i, s_d) = \min_{s_j \in \text{neighbors}(s_i)} [R_(s_i, s_j) + c^-(s_j, s_d)] \quad (1)$$

As mentioned, the calculation of $c^+(s_i, s_d)$ is more complex than the calculation of $c^-(s_i, s_d)$, as an event might turn inactive while the vehicle is traveling. Thus, the calculation of $c^+(s_i, s_d)$ contains costs of traversing subsequent segments in both cases with their respective probability, i. e., the event is still active or event turns inactive. We employ the indicator function $I_{\{l_t < T(s_i,s_j)\}}$ and $I_{\{l_t \geq T(s_i,s_j)\}}$ associated with the lifetime of the event to capture the two possibilities. If the event turns inactive before reaching the subsequent decision-point s_j and the vehicle is notified, the costs $c^-(s_j, v, s_d)$ need to be applied, $c^+(e_s^t, s_j, v, s_d)$ otherwise. Equation 2 describes the costs at a decision-point s_i given that the event is active.

$$\begin{aligned} c^+(s_i, s_d) = \min_{s_j \in \text{neighbors}(s_i)} &[R(s_i, s_j) + \\ &c^-(s_j, s_d) \cdot I_{\{l_t < T(s_i,s_j)\}} + \\ &c^+(s_j, s_d) \cdot I_{\{l_t \geq T(s_i,s_j)\}}] \end{aligned} \tag{2}$$

Equation 2 describes the costs as a function of the random variable remaining lifetime l_t and is thus also a random variable. By taking expectations, we obtain the average values of the costs involved in the expression while replacing the indicator functions with the probabilities of the indicated events. Notice that $\overline{c}^-(s_i, v, s_d)$ is not influenced by the lifetime of the event, as there is no event to be considered. Additionally, we assume an exponential lifetime distribution. Thus, we can calculate the probability of $l_t < T(s_i, s_j)$ using the average event lifetime $\overline{l_t}$ as shown in Equation 3.

$$P(l_t < T(s_i, s_j)) = 1 - \exp\left[-\frac{T(s_i, s_j)}{\bar{l}_t}\right] \tag{3}$$

For better readability, we will use P^+ for $P(l_t \geq T(s_i, s_j))$ and P^- for $P(l_t < T(s_i, s_j))$. Due to the memorylessness property of the exponential distribution, we can calculate the probability of the event to disappear for a path from s_i to s_j individually. Thus, Equation 2 leads to Equation 4.

$$\begin{aligned} \overline{c}^+(s_i, s_d) = \min_{s_j \in \text{edges}(s_i)} &[R(s_i, s_j) + \\ \overline{c}^-(s_j, s_d) \cdot P^- + &\overline{c}^+(s_j, s_d) \cdot P^+] \end{aligned} \tag{4}$$

B. Finding the Lowest-cost Path

To find the path with the lowest cost, the vehicle needs to solve the recursive function for $\overline{c}^+(e_s^t, s_v, v, s_d)$ or $\overline{c}^-(s_v, v, s_d)$ for the its current position s_v based on its current knowledge. We find a solution for the recursive functions using a graph-based representation of the costs from each decision-point in the network to the destination s_d of the vehicle, but stop the executing when the impact of the event vanishes. As the costs under the assumption that the event is active or inactive differ, we use two costs graphs, in which the shortest (cost-minimal) path from the vehicle's location to its destination is determined. However, compared to the shortest-path search in conventional graphs, we might have to switch between the graphs, as the event might turn inactive.

To address this issue, we first search the shortest-paths in the graph based on $\overline{c}^-(s_i, v, s_d)$, as this graph is independent of time. We find the costs $\overline{c}^-(s_i, v, s_d)$ for each edge using the Bellmann-Ford shortest-path algorithm. Based on these costs, we aim to determine the costs $\overline{c}^+(e_s^t, s_i, v, s_d)$. Importantly, the event might turn inactive after a random duration, in which we need to switch to the graph of $\overline{c}^-(s_i, v, s_d)$. Thus, a conventional shortest-path search algorithm is not applicable.

Algorithm 1: Calculation of path costs starting from any segment s. In each round, the costs $\overline{c}_i^+(s, s_d)$ are recalculated based on the current values of $\overline{c}_{i-1}^+(s, s_d)$.

Input : Planned destination s_d of the vehicle
Output : Costs for all edges assuming the event was active

for $s \in S$ **do**
  $\overline{c}_0^+(s, s_d) \leftarrow \infty$;
  $\rho(s) \leftarrow 0$
end
$\overline{c}_0^+(e_s^t, s_d, v, s_d) \leftarrow 0$;
for $i \in \{1, \ldots, n\}$ **do**
  for $s_1 \in S$ **do**
    $\overline{c}_i^+(s_1, s_d) = \overline{c}_{i-1}^+(s_1, s_d)$;
    for $s_2 \in \textit{neighbors}(s_1)$ **do**
      if $\overline{c}_i^+(s_1, s_d) > R(s_i, s_j) + \overline{c}^-(s_j, s_d) \cdot P^- + \overline{c}_{i-1}^+(s_j, s_d) \cdot P^+ \ \& \ loopfree$ **then**
        $c_i^+(s_1, s_d) \leftarrow R(s_1, s_2) + \overline{c}^-(s_2, s_d) \cdot P^- + \overline{c}_{i-1}^+(s_2, s_d) \cdot P^+$;
        $\rho(s_1) \leftarrow s_2$;
      end
    end
  end
end

To calculate the costs $\overline{c}^+(e_s^t, s_i, v, s_d)$ of each segment, we use a modified version of Bellmann-Ford algorithm described in Algorithm 1. The algorithm performs n steps, which states the maximum number of segments in a shortest path that the approach can detect. At first, the costs c_0^+ of all segments are set to infinity, and the predecessor $\rho(s)$ is set to 0. In each step i, the algorithm sets the costs $c_i^+(s_i, s_d)$ of each segment to the current minimum given the currently known costs $c_{i-1}^+(s_i, s_d)$ of the accessible segments. The costs $c_i^+(s_i, s_d)$ are combined as shown in Equation 4, i. e., contain the costs $c^-(s_i, s_d)$ with the probability that the event goes inactive.

Theorem 1. Algorithm 1 finds the loop-free path of maximum length n with the lowest expected costs given the event e_s^t is currently active.

Proof. We use full induction to show that the path found is optimal:

First, we need to show that the statement holds for our base case. At the destination of the vehicle, the costs for $\overline{c}^+(s_d, s_d) = 0$ is optimal, as the vehicle's path terminates at s_d. As $R(s_i, s_j) \geq 0, \forall s_i, s_j \in S$, holds for the additional movement costs, the costs cannot get lower by traversing the graph. Thus, our algorithm finds the optimal path of length 0, as no edges are traversed (the vehicle stays at segment s_d). Notice, that s_d does not contain the event by assumption. For this calculation, the number of involved edges in the graph is 0. Notice that segments of the road network refer to vertices in our graph.

Second, we show that the inductive step holds. For a maximum number of edges k, we can assume that all shortest paths with a number of edges of at most $k-1$ have already been found. For this purpose, $\overline{c}^{-}_{k-1}(s_2, s_d), \forall s_2 \in \text{neighbors}(s_1)$, is calculated using Bellmann-Ford shortest path algorithm if the event is inactive. Due to the assumption in the induction step, $\overline{c}^{+}_{k-1}(e^t_s, s_2, v, s_d), \forall s_2 \in \text{neighbors}(s_1)$, reflects the lowest-cost path of a maximum length $k-1$ if the event is active. Thus, we can calculate $\overline{c}^{+}_{k}(e^t_s, s_1, v, s_d)$ based on these values (Equation 4). Notice that paths that would produce a loop are not considered in the calculation. The calculation of $\overline{c}^{+}_{k}(e^t_s, s_1, v, s_d)$ increases the path length by 1, as one more step is added and is optimal for this path length, as all values $\overline{c}^{+}_{k-1}(e^t_s, s_2, v, s_d), \forall s_2 \in \text{neighbors}(s_1)$, are optimal. Additionally, the segment s_2 used for the calculation is stored to restore the path. Thus, we can find the shortest path of maximum number of edges k using the path lengths of all paths with $k-1$. □

C. Determining of the Event-specific Cost Value C^t

In this work, we determine the costs for a vehicle by considering factors influencing the driver's comfort and safety. That is, factors like the type of event and the driver's personal preferences need to be considered. As an example of the influence of the event type, an accident impacts the safety and comfort of a driver more severely than a traffic jam. Thus, the basis for the calculation of C^t is the type of event, which generally has the highest impact on C^t. High values of C^t generally will lead to a lot of vehicles/drivers preventing an encounter with the event. Additionally to the type of event, a driver might observe events to be of a different impact. As an example, a driver could hate getting stuck in a traffic jam and might prefer driving a longer path to prevent from getting stuck. If the driver communicates this information to the server, the server will utilize it to provide more relevant information.

V. Efficient Information Dissemination

The decision-making of vehicles is strongly connected to the available information and, thus, to the communication between the vehicles. If a vehicle cannot improve its decision-making by receiving an event, it is generally not reasonable for the server to provide information about the event to the vehicle. The decision for or against the dissemination of this information to a specific vehicle is discussed in Section V.

As already mentioned in the previous chapter, vehicles can make decisions at multiple decision-points. We aim to use knowledge about the vehicle's decision-making to optimize server-based information dissemination. The underlying idea is that a vehicle only needs to know about an event if this knowledge changes the vehicle's decision. If the vehicle is too far away from the event and, thus, would not change its decision, it is not reasonable for the server to transmit the event-related information to this vehicle. Thus, we aim to estimate the benefit of a detour induced by received event-related information, i. e., reduction in costs compared to the current path of the vehicle, and derive the message utility. This message utility states the best time to share the event with the vehicle, which is generally at the decision-point where the vehicle would make a decision. That way, redundant message transmissions are reduced, and the vehicle receives the most up-to-date event-related information for its decision. Notice that our server can store events and provide them to the vehicles at any time.

In the previous section, each vehicle aimed to find the optimal (cost-minimal) path based on its available knowledge of the road network. While the decision-making is generally performed on the vehicle, the server might mirror it to improve the information dissemination. To this end, the server needs to estimate the expected utility of a message $\overline{u}(e^t_s, s_v, v, \vec{\phi})$ for a particular event and vehicle (when clear from context, we will omit e^t_s and v). This utility depends on the benefit of next possible decisions a vehicle can take, e. g., if the vehicle would stay on its planned path at the next decision-point or if it would detour. To exactly mirror the decision, the server requires the event, the vehicle's position, destination, and preferences. We assume that the server has already stored the event e^t_s and aims to distribute this event efficiently. The position of the vehicle is known through continuous context-updates performed by the vehicle. As stated previously, we assume that the planned path $\vec{\phi}$ is known to the server. Additionally, the vehicle may share its preferences v with the server once it starts its path.

Based on the planned path $\vec{\phi}$, the server can calculate the expected utility of the message for a vehicle at position s_v. The utility of this message is only non-zero if the vehicle would detour if it receives the message. The benefit of a decision at the current position of the vehicle s_v is shown in Equation 5. Notice that the destination s_d can be determined from $\vec{\phi}$.

$$\begin{aligned}\overline{u}(s_v, \vec{\phi}) = \overline{c}^{+}\left(\left[s \middle| \left(s \in \vec{\phi}\right) \wedge \left(s \in \text{neighbors}(s_v)\right)\right], s_d\right) \\ - \min_{s \in \text{neighbors}(s_v)} \overline{c}^{+}(s, s_d)\end{aligned} \tag{5}$$

The message utility (and benefit of a decision) is equal to the difference in cost of the planned and optimal path based on the event and the vehicle's location s_v. If the vehicle's decision is not to detour, the utility of a message by the server informing about the event is zero. Thus, the message should not be sent to the vehicle to save communication resources. However, the utility might be non-zero, but very small, for example, if the vehicle is far away from the event. In this case, it may also not be worth sending the message and incurring the communications/service cost. Thus, we use a threshold to determine if an event should be transmitted or not. This threshold is set by each vehicle individually and aims to adapt to the current networking conditions of the vehicle filtering unimportant events and hence freeing bandwidth for essential events.

VI. Evaluation

We evaluate our approach using the event-based Simonstrator framework in combination with our vehicular extension presented in [18]. This evaluation aims to analyze the performance in comparison with state-of-the-art approaches under varying environmental conditions, i. e., to obtain an insight into the influence factors of the performance. We performed each parameter setup with 20 seeds to ensure the statistical significance of our results.

A. Baseline approach

As a baseline, we use an approach where the vehicles only rely on their local perception, which is limited by environmental conditions. We will refer to this approach with *VI*. Thus, the vehicle can only detour if the driver can perceive the event in-time to change the path. In the following, we will assume a vision range of the driver of $500m$. For this approach, we search the shortest path using Dijkstra's shortest-path algorithm, in which the cost of the affected segment is increased by C^t.

B. Reference approaches

For comparison, we additionally implement three baseline approaches, a *latest-possible* decision approach, a *static*-information approach, and a *full*-information approach which will be explained in the following.

a) Latest-Possible: We refer to this approach with *LP*. The vehicles are notified by a central server about events, but always choose the last possible option to detour (which might be rather inefficient). The last detour is defined as the vehicle taking the last possible exit from the planned path before the event would be encountered. We expect this approach to perform well for events with a small lifetime. The vehicle's path is again determined using Dijkstra's shortest-path algorithm with high costs C^t for the road with the event.

b) Static: We refer to this approach with *SI* in the following. The vehicle receives all events in the road network from a central server but assumes that the events will always be active. However, the vehicle will receive an update if the event becomes inactive. Thus, the vehicle can search for the shortest path based on the permanent events it is provided by the server which can be considered to be up-to-date. Again, the vehicle uses Dijkstra's shortest-path algorithm to find the optimal path in the network employing Dijkstra's algorithm with the road segment costs as shaped by the presence of the permanently active event. The roads with active events have a higher edge costs, which are then utilized by Dijkstra in the finding of the shortest path. This path search is performed frequently to account for changes in the road network's state.

c) Full-Information: We refer to this approach with *FI* in the following. The vehicle receives all events in the road network from a central server including the exact times at which the events will disappear. This approach is unfeasible to be used in practice, but we use it for reference. This information is then used in the path-planning to detour only if the event will still be active by the time the vehicle arrives at the event location. Notice that the travel-time of the vehicle to the event location is only estimated, which induces a slight deviation from the optimal. For this purpose, we use a slightly modified version of Dijkstra's shortest-path algorithm. Once a road with an active event is encountered, the vehicle checks based on its estimated time to arrive there if the event will still be active. If the event will be active, the vehicle will consider the high edge costs in the path-planning, while it will use the standard edge costs if the event will be inactive. However, this estimated time to arrive is only a prediction, i. e., this approach might suffer from wrong predictions. This path search is performed frequently to account for changes in the road network's state.

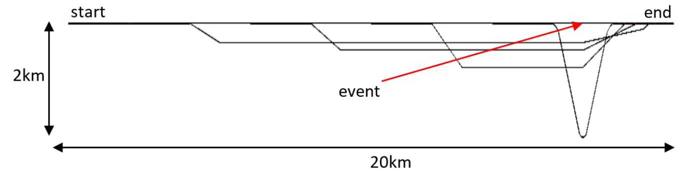


Fig. 2: Overview of the considered road scenario. The ratio of the scenario has been changed to improve readability.

d) Dynamic: We refer to this approach with *DI* in the following. The vehicle receives all events in the road network including their expected lifetime $\bar{l}_t$, which are considered to be relevant to it as proposed in Section V. That is, the vehicle communicates its planned path to the central server which uses this knowledge to estimate the vehicle's best decision. This approach is performed periodically to account for changes in the road network.

C. Scenario

We designed a scenario in which we can showcase the advantages of our *DI*-approach. Such a scenario contains multiple alternative paths towards a potential destination, i. e., a vehicle can detour at multiple locations. However, detouring late leads to a longer detour compared to an early detour. This scenario is realistic, as a vehicle generally has more options if its distance to the destination is farther.

The scenario we designed is depicted in Figure 2. It has a total size of $20km$, and the vehicles are driving with a speed of $60km/h$. The investigated traffic starts at the upper left and travels to the upper right. There are five possible paths towards the destination, out of which the shortest path is used as a default. The additional length of the detours decreases with increasing distance to the event location. After $20min$ of simulation time, the event edge is jammed, which blocks the previously shortest path. For the exact location, a point on this edge is chosen randomly. Similarly, the lifetime of the event is chosen randomly based on the exponential distribution.

D. Metrics

We analyze the performance of our approach compared to the baselines using the additional costs induced to the network for every second the event is active. We divide this metric into necessary costs, which captures the costs for detours that were required as the vehicle would otherwise

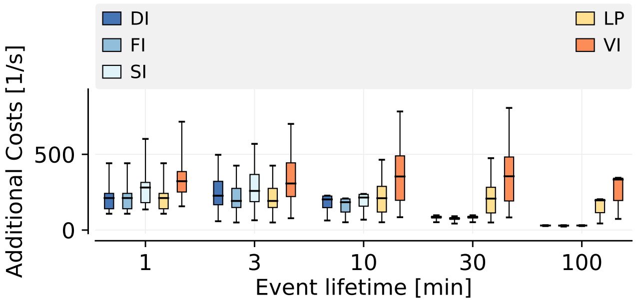
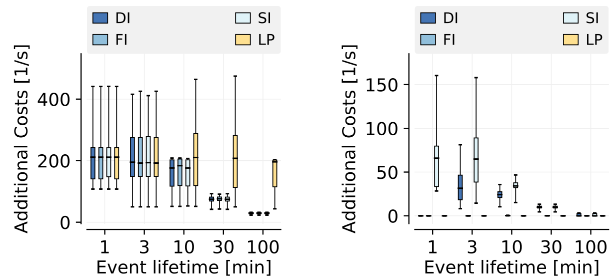


Fig. 3: Total additional edge costs of the network per second of event lifetime. Our approach adapts to the event lifetime.



(a) Costs for necessary detours. (b) Costs for unnecessary detours.

Fig. 4: Costs of the different approaches depending on the lifetime of the event.

encounter the event, and unnecessary costs, which captures the costs for detours that were not required, as the event would have turned inactive by the time the vehicle arrived. Additionally to the performance of the routing approaches, we analyzed the necessary communication demand of the different approaches by evaluating their influence on the communication infrastructure.

E. Evaluation Results

In the following, we provide detailed insights into the performance of our approach under varying environmental settings. That is, we investigate the quality of the decision-making as proposed in Section IV using cost-based metrics. Additionally, we investigate the produced network traffic for the different approaches to validate the low traffic requirements of our approach as proposed in Section V.

1) Produced Costs of the Routing Approaches: Figure 3 displays the total costs for rerouted vehicles depending on the event lifetime. This includes both vehicles which were required to reroute and vehicles that did not need to reroute. The performance of the *VI*-approach is bad for all event lifetimes, as the vision-range of the vehicle might not be able to detect the event and, thus, the vehicle cannot detour. The *LP*-approach has a similar behavior as the *VI*-approach, but uses communication and is not affected by the sensor-range. Thus, it performs well for a small event duration, as vehicles farther away do not need to be rerouted due to the small lifetime. However, the performance drops drastically for large event lifetimes, as the inefficient detour reduces the performance. It can clearly be observed that the costs of the *SI*-approach are comparably high for short event lifetimes. That is, as the *SI*-approach immediately tries to reroute all vehicles in the network, which leads to a lot of unnecessary detours for small event lifetimes. Interestingly, the costs for the static approach at $3m$ are slightly higher than at $1m$, that is justified by the higher number of vehicles that can decide during the time the event is active. Our dynamic approach adapts to the event lifetime and reacts accordingly. Thus, our approach reduces the average costs for the vehicles in the network by up to 26% compared to the *SI*-approach for a short event lifetime and by up to 80% compared to the *LP*-approach for a long event lifetime. Additionally, the average costs of our approach is a maximum of 18.6% higher compared to the *FI*-approach, which relies on perfect knowledge. For most event lifetimes, our approach performs almost similar compared with the *FI*-approach.

Figure 4 shows a more detailed view of the performance of the different approaches. As the behavior of the *VI*-approach and *LP*-approach is similar, we omit the *VI*-approach approach to increase readability. While the *SI*- and the *DI*-approach detour vehicles at earlier times, the risk of unnecessary detours increases. However, the efficiency of the necessary detours is also increased. Figure 4a shows the costs of the necessary detours. That is, the costs for the *DI*-, *SI*-, and *FI*-approach are very close to similar in this metric, while the performance of the *LP*-approach decreases for large event lifetimes. This is expected, as the *LP*-approach utilizes long detours to prevent unnecessary detours at all costs. This is also visible in Figure 4b. The *LP*-approach does not induce any additional costs for non-concerned vehicles, similar to the *FI*-approach. In contrast, the *SI* and the *DI*-approach induce unnecessary costs as vehicles are rerouted without need, i. e., the event has expired by the time the vehicle would have arrived there. However, these unnecessary detours reduce the costs of the necessary detours, such that our *DI*-approach outperforms the *LP*-approach significantly. That is, our *DI*-approach does not unnecessarily detour vehicles for an event lifetime of $1m$, and reduces the costs for unnecessary detours by 45.6% for an event lifetime of $3m$.

2) Necessary Amount of Communication: Figure 5 displays the necessary communication demands, that is required by the different approaches. We omit the *VI*-approach, as it does not communicate. We can clearly see that our *DI*-approach utilizes much less bandwidth compared to the *SI*-approach. That is, as vehicles in the *DI*-approach generally reroute later if the event lifetime is small and thus do not need to be notified. This effect is reduced for events with a long lifetime, in which the *SI*- and the *DI*-approach perform almost similarly. It consumes only slightly more bandwidth than the *FI*-approach, which is justified by the unpredictability of event disappearance. As already shown in Figure 4b, some vehicles detour unnecessarily and, thus, require the message. Compared to the *LP*-approach, our *DI*-approach generally consumes slightly more bandwidth.

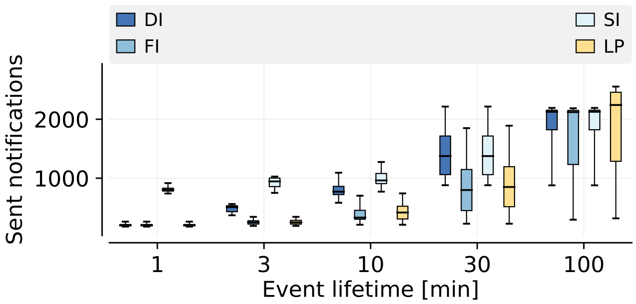


Fig. 5: Number of transmitted messages for the different approaches. Our *DI*-approach significantly reduces the number of transmitted messages for a short event lifetime compared to the *SI*-approach.

However, for very long event lifetimes, the *LP*-approach consumes slightly more messages in certain cases. This effect can be explained by the longer detours of the vehicles in case of the *LP*-approach. As these vehicles stay longer in the system than if they chose an efficient detour, more vehicles receive the notification that the event has disappeared compared to the *DI*-approach.

VII. Conclusion

In this work, we propose an innovative approach to determine optimal routes of a vehicle to its planned destination. For this purpose, consider the impact of road events on the costs associated with a certain road segment. As these events might turn inactive in the future, the route finding is reassessed at every intersection. Compared to state-of-the-art approaches, we consider the expected lifetime of events in our route-finding approach. That is, if an event will most likely turn inactive by the time the vehicle arrives at the event location, the vehicle does not necessarily need to consider it. To decide if a vehicle should detour, we use a set of recursive functions to determine the expected costs of every possible route for the vehicle. This information is then used in the information dissemination to notify only concerned vehicles.

In the evaluation, we show that our innovative approach adapts to the lifetime of the events well and significantly reduces the additional costs induced by route events. Additionally, our approach reduces the load on the cellular network significantly, as only relevant information are shared with the vehicles. In our future work, we aim to extend the assessment of road events to other properties like measuring accuracy and evaluate our work in a large-scale scenario.

References

[1] A. Tigadi, R. Gujanatti, A. Gonchi, and B. Klemsscet, "Advanced driver assistance systems," *iJERGS*, vol. 4, no. 3, pp. 151–158, 2016.

[2] S. Ancona, R. Stanica, and M. Fiore, "Performance boundaries of massive Floating Car Data offloading," in *Proc. of WONS*. IEEE, apr 2014, pp. 89–96.

[3] M. Wang, H. Shan, R. Lu, R. Zhang, X. Shen, and F. Bai, "Real-Time Path Planning Based on Hybrid-VANET-Enhanced Transportation System," *Transactions on Vehicular Technology*, vol. 64, no. 5, 2015.

[4] S. Gopinath, L. Wischhof, C. Ponikwar, and H. Hof, "Hybrid solutions for data dissemination in vehicular networks," in *Proc. of WD*, March 2016, pp. 1–4.

[5] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A Review of Motion Planning Techniques for Automated Vehicles," *Intelligent Transportation Systems*, vol. 17, no. 4, p. 1135, 2016.

[6] J. Ziegler, M. Werling, and J. Schröder, "Navigating car-like robots in unstructured environments using an obstacle sensitive cost function," *Proc. of IV*, pp. 787–791, 2008.

[7] P. Chen, Y. Guo, and W. Chen, "Fuel-saving navigation system in vanets," in *VTC Fall*, Sep. 2010, pp. 1–5.

[8] A. H. Khosroshahi, P. Keshavarzi, Z. D. KoozehKanani, and J. Sobhi, "Acquiring real time traffic information using vanet and dynamic route guidance," in *Proc. of CCIE*, vol. 1, Aug 2011, pp. 9–13.

[9] M. Kimura, Y. Taoda, Y. Kakuda, S. Inoue, and T. Dohi, "A novel method based on vanet for alleviating traffic congestion in urban transportations," in *Proc. of ISADS*, March 2013, pp. 1–7.

[10] E. Akyol and U. Mitra, "Power-distortion metrics for path planning over gaussian sensor networks," *IEEE Transactions on Communications*, vol. 64, no. 3, pp. 1220–1231, March 2016.

[11] C. Guo, D. Li, G. Zhang, and M. Zhai, "Real-time path planning in urban area via vanet-assisted traffic information sharing," *Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 5635–5649, July 2018.

[12] Y. Chung, "Development of an accident duration prediction model on the Korean Freeway Systems," *Accident Analysis & Prevention*, vol. 42, no. 1, pp. 282–289, 2010.

[13] F. Jomrich, A. Sharma, T. Rückelt, D. Burgstahler, and D. Böhnstedt, "Dynamic map update protocol for highly automated driving vehicles," in *Proc. of VEHITS*. SCITEPRESS, Apr 2017, pp. 68–78.

[14] I. Leontiadis, P. Costa, and C. Mascolo, "A hybrid approach for content-based publish/subscribe in vehicular networks," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 697–713, 2009.

[15] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. of SECON*, June 2011, pp. 350–358.

[16] M. Jerbi, R. Meraihi, S.-M. Senouci, and Y. Ghamri-Doudane, "Gytar: improved greedy traffic aware routing protocol for vehicular ad hoc networks in city environments," *Proc. of VANET*, 2006.

[17] G. Li, L. Boukhatem, and J. Wu, "Adaptive quality-of-service-based routing for vehicular ad hoc networks with ant colony optimization," *Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3249–3264, April 2017.

[18] T. Meuser, D. Bischoff, R. Steinmetz, and B. Richerzhagen, "Simulation platform for connected heterogeneous vehicles," in *Proc. of VEHITS*. SCITEPRESS, May 2019.